

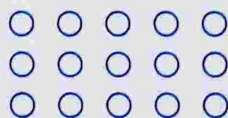
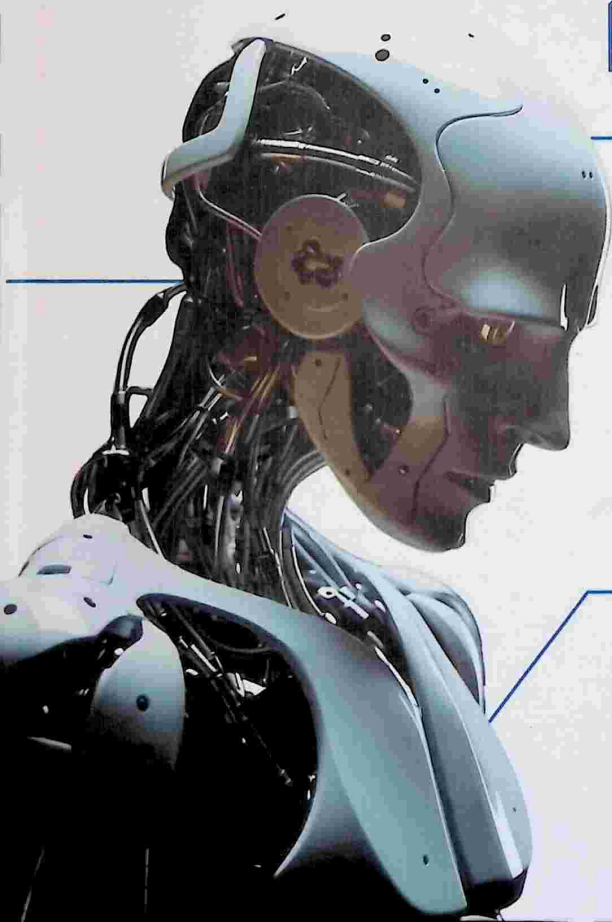
DEVID L. PUL,
ALAN K. MAKVORT

+ +

SUN'IY INTELLEKT

HISOBLASH AGENTLARINING ASOSLARI

II JILD

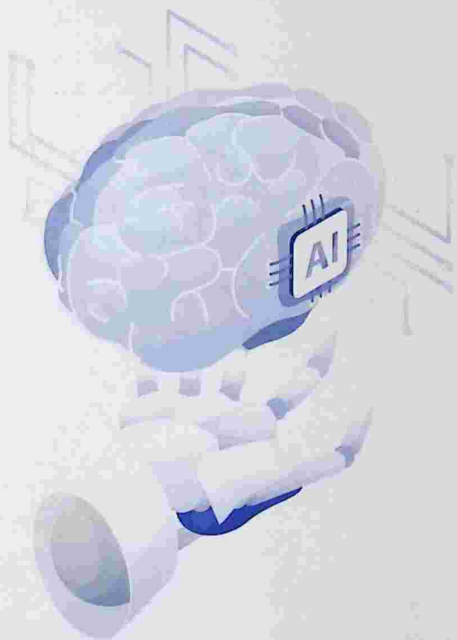


DEVID L. PUL, ALAN K. MAKVORT

SUN'IY INTELLEKT

HISOBLASH AGENTLARINING
ASOSLARI

II JILD



Toshkent – «Donishmand ziyosi» – 2025

**O'zbekiston Respublikasi Oliy ta'lim, fan va innovatsiyalar vazirligi
huzuridagi Oliy ta'limni rivojlantirish tadqiqotlari markazining
buyurtmasiga asosan nashrga tayyorlandi va chop etildi.**

Ingliz tilidan **Fotima Safarova** tarjimai

Devid L. Pul, Alan K. Makvort.

Sun'iy intellekt (Hisoblash agentlarining asoslari) II jild [Matn]: darslik /
Devid L. Pul, Alan K. Makvort.. — Toshkent: Donishmand ziyosi, 2025. — 336 b.

Darslik bakalavriat va magistratura bosqichi talabalari uchun mo'ljallangan bo'lib, zamonaviy sun'iy intellekt va uning ijtimoiy ta'sirini tushuntiradi hamda nazariya va amaliyotni birlashtiradi. Ushbu kengaytirilgan yangi nashr chuqur o'rganish bo'yicha bo'limlarni, shu jumladan generativ sun'iy intellekt, sun'iy intellektning ijtimoiy ta'sirlari va sababini o'z ichiga oladi.

Mazkur xorijiy o'quv adabiyoti professor-o'qituvchilar, doktorantlar, magistrantlar va talabalarning o'quv va ilmiy faoliyatida foydalanish uchun mo'ljallangan. Shuningdek, ilmiy izlanishlar olib borayotgan mustaqil tadqiqotchilar va keng ilmiy jamoatchilik uchun tavsiya qilinadi.

Ushbu xorijiy o'quv adabiyotidan tijorat maqsadida foydalanish qat'iyan man etiladi.



IV QISM. NOANIQLIK SHAROITIDA REJALASHTIRISH VA HARAKAT QILISH

Qanday qilib agent o'z harakatlarining oqibatlari va dunyoning joriy holatini biladi degan taxminni chetlab o'tib, rejalashtirishi va harakat qilishi mumkin?

12-bob. NOANIQLIK SHAROITIDA REJALASHTIRISH

O'tmishga nazar solsak... shuni ta'kidlash qiziqki, tadqiqotimni boshlagan dastlabki muammo, ya'ni vaqt o'tishi bilan dinamik rejalashtirish yoki jadvallash muammosi, ayniqsa noaniqlik sharoitida dinamik rejalashtirish hali ham yechimini topmagan. Agar bunday muammoni muvaffaqiyatli hal etish mumkin bo'lsa, u pirovardida yaxshiroq rejalashtirish orqali dunyoning farovonligi va barqarorligiga o'z hissasini qo'shishi mumkin.

– Jorj B. Dansig (1991)

Hamma narsadan xabardor bo'lmagan agent 6-bobda taxmin qilinganidek, qat'iy qadamlar ketma-ketligini oddiygina rejalashtira olmaydi. Rejalashtirish shuni hisobga olishi kerakki, haqiqiy dunyodagi agent u harakat qilganda aslida nima sodir bo'lishini va kelajakda nimani kuzatishini bilmaydi. Agent atrof-muhitiga mos ravishda harakat qilishni rejalashtirishi kerak.

Agent har qanday vaqtda nima qilish kerakligi, kelajakda nima qilishiga bog'liq. Misol uchun, tibbiy vaziyatda tahlillar ba'zan bemorlarga zarar yetkazadi, ammo ular foydalidir, chunki ular kelajakda harakat qilish imkonini beradi. Agar agent o'z harakatlarining oqibatlarini aniq bashorat qila olmasa, kelajakda nima qilishi uning hozir nima qilayotganiga va harakat qilishdan oldin nimani kuzatishiga bog'liq.

Noaniqlik sharoitida agent odatda o'z maqsadlariga erishishini kafolatlay olmaydi va hatto maqsadga erishish ehtimolini maksimal darajaga oshirishga harakat qilish ham asossiz bo'lishi mumkin. Masalan, avtomobil halokatida jarohatlanish ehtimolini minimal darajaga tushirish maqsadiga ega bo'lgan agent avtomobilga chiqmasligi yoki trotuarda yurmasligi yoki hatto binoning pastki qavatiga tushmasligi kerak, chunki ularning har biri, garchi oz bo'lsa ham, avtomobil halokatida jarohatlanish ehtimolini oshiradi. Maqsadga erishishni kafolatlamaydigan agent ko'p jihatdan bir-biridan yomonroq muvaffaqiyatsizliklarga uchrashi mumkin. Ushbu bob rejalashtirish, reaksiya, kuzatish, muvaffaqiyat va muvaffaqiyatsizlikni bir vaqtda qanday hisobga olish haqidadir. Yuqoridagi iqtibosda

ko'rsatilganidek, chiziqli dasturlash ixtirochisi Jorj Dansig noaniqlik sharoitida rejalashtirish aqlli agent uchun juda muhim ekanligini ta'kidlaydi.

Agentning har qanday vaqtda nima qilishga qaror qilishi (2.10-rasmga qarang) quyidagilarga bog'liq:

- *Agentning qobiliyati.* Agent o'zi uchun mavjud bo'lgan harakatlardan tanlashi kerak.

- *Agentning ishonchi va kuzatishlari.* Agent o'z harakatlarini dunyoda sodir bo'layotgan voqealarga bog'lashni xohlashi mumkin, lekin u dunyoga faqat sensorlari orqali kirish imkoniyatiga ega. Agent nima qilishni hal qilishi kerak bo'lsa, u faqat eslab qolgan va kuzatgan narsalaridan foydalanish huquqiga ega. Dunyo haqidagi ma'lumotlar agentning ishonchlarini yangilaydi. Ishonchlar va kuzatishlar har qanday vaqtda agent uchun mavjud bo'lgan dunyo haqidagi yagona ma'lumotdir.

- *Agentning tanlovlari.* Agent noaniqlik bilan mulohaza yuritganida, u nafaqat ehtimollikni, balki nima bo'lishi mumkinligini ham hisobga olishi kerak. Ba'zi mumkin bo'lgan natijalar boshqalariga qaraganda ancha yomon oqibatlariga olib kelishi mumkin. 6-bobda ko'rib chiqilgan oddiy *maqsad* tushunchasi noaniqlik sharoitida mulohaza yuritishda yetarli emas, chunki agentning dizayneri sodir bo'lishi mumkin bo'lgan turli natijalar o'rtasida muvozanat saqlashi kerak. Masalan, bir harakat ko'pchilik hollarda yaxshi natijaga olib keladigan bo'lsa-yu, lekin ba'zan falokatli natijaga olib kelishi mumkin bo'lsa, bu harakat yaxshi natijaga kamdan kam erishadigan va ko'p hollarda o'rtacha natijaga olib keladigan muqobil harakat bilan taqqoslanishi kerak. Qarorlar nazariyasi natijalarning istalganligini ularning ehtimolliklari bilan qanday muvozanat qilishni belgilaydi.

12.1. Afzalliklar va foyda

Agentning nima qilishga qaror qilishi uning **xohishiga** bog'liq bo'lishi kerak. Ushbu bo'limda afzalliklarning ba'zi intuitiv xususiyatlari va ushbu xususiyatlarining ba'zi oqibatlari keltirilgan. Xususiyatlar **ratsionallik aksiomalari** hisoblanadi. Har bir aksioma **ratsional agent** uchun qabul qilinishi kerakmi yoki yo'qligini ko'rib chiqishingiz kerak. Agar siz ularning hammasini asosli deb qabul qilsangiz, siz ularning oqibatlarini ham qabul qilishingiz lozim. Agar oqibatlarni qabul qilmasangiz, bir yoki bir nechta aksiomadan voz kechishingiz talab etiladi.

12.1.1. Ratsionallik aksiomalari

Agent **natijalarga** asoslangan harakatlarni tanlaydi. Natijalar agent **afzal ko'rgan** narsalardir. Agar agent biron-bir natijani boshqa natijadan afzal ko'rmasa, agentning nima qilishi muhim emas. Dastlab, bog'liq harakatlarni hisobga ol-

masdan natijalarni ko'rib chiqaylik. Faqat cheklangan sonli natijalar mavjud deb taxmin qilaylik. Keling, natijalar bo'yicha **afzalliklar munosabatini** belgilaylik. Aytaylik, o_1 va o_2 natijalardir. Agar o_1 natija hech bo'lmaganda o_2 natijasi kabi ma'qul bo'lsa, o_1 natijasi $o_1 \geq o_2$ deb yozilgan o_2 natijasiga nisbatan **zaif afzaldir**.

Biz $o_1 < o_2$ ni $o_1 \geq o_2$ ni inkor etish ma'nosida yozamiz, ya'ni o_1 natija o_2 dan zaifroq afzal ko'rilmaydi.

$o_1 \sim o_2$ ni $o_1 \geq o_2$ va $o_2 \geq o_1$ deb belgilash shuni anglatadiki, $o_1 \sim o_2$ natijalar o_1 va o_2 bir xil afzal ko'riladi. Bu holda, agent o_1 va o_2 o'rtasida befarqdir.

$o_1 > o_2$ ni $o_1 \geq o_2$ va $o_2 < o_1$ deb aniqlang. Ya'ni, agent o_1 natijasini o_2 natijasiga zaifroq afzal ko'radi, lekin o_2 ni o_1 dan zaifroq afzal ko'rilmaydi va ular o'rtasida befarq emas. Bu holda, o_1 natijasi o_2 natijasiga nisbatan qat'iy afzal ko'riladi.

Odatda, agent o'z harakatlarining natijasini bilmaydi. **Lotereya** natijala bo'yicha cheklangan taqsimot deb belgilangan,

$$[p_1 : o_1, p_2 : o_2, \dots, p_k : o_k]$$

deb yoziladi, bu yerda har bir o_i natija va har bir p_i manfiy bo'lmagan haqiqiy son bo'lib,

$$\sum_i p_i = 1.$$

Lotereya o_i natija ni p_i ehtimoli bilan sodir bo'lishini belgilaydi.

Keyingi barcha holatlarda natijalar lotereyalarni o'z ichiga olishi mumkin deb faraz qiling. Bu, shuningdek, natijalar ham lotereyalar bo'lgan lotereyalarni (lotereyalar ustidan lotereyalar deb ataladi) o'z ichiga oladi.

12.1-aksioma (To'liqlik) Agent barcha natijalar juftliklari orasida tanlovlar-
ga ega:

$$o_1 \geq o_2 \text{ or } o_2 \geq o_1.$$

Ushbu aksioma uchun asos shundaki, agent harakat qilishi kerak; agar unga mavjud bo'lgan harakatlar natijalari o_1 va o_2 bo'lsa, u harakat qilish orqali ochiq yoki yopiq ravishda bir natijani boshqasidan afzal ko'radi.

Ushbu aksiomaning asosi quyidagicha: agar agent u uchun mavjud bo'lgan harakatlar o_1 va o_2 natijalariga ega bo'lsa, u holda harakat qilish orqali u aniq yoki bilvosita bir natijani boshqasidan afzal ko'radi hamda harakat qilishi kerak.

12.2-aksioma (Tranzitivlik) Tanlovlar tranzitivdir:

$$\text{if } o_1 \geq o_2 \text{ and } o_2 > o_3 \text{ then } o_1 > o_3.$$

Buning asosli ekanligini ko'rish uchun, agar bu noto'g'ri bo'lsa, ya'ni $o_1 \leq o_2$ va $o_2 > o_3$ va $o_3 > o_1$ deb faraz qiling. Chunki o_2, o_3 dan qat'iy ravishda afzal ko'rilgan, agent o_3 dan o_2 ga o'tish uchun bir miqdor pul to'lashga tayyor bo'lishi kerak.

Agent o_2 natijasiga ega bo'lsa, unda o_1 kamida shuncha yaxshi bo'lganligi sababli, agent o_1 ni olishga tayyor bo'ladi. o_3 kamida o_1 kabi yaxshi, shuning uchun agent o_1 ni olishga tayyor bo'ladi. Agent o_3 ga ega bo'lgach, u yana o_2 ga o'tish uchun pul to'lashga tayyor bo'ladi. Bu afzalliklar siklidan o'tish va pul to'lashni o'z ichiga olgan siklmoney pump (pul pompasi) sifatida tanilgan, chunki sikldan yetarlicha ko'p marta o'tish orqali agent to'lashi kerak bo'lgan pul miqdori har qanday cheklangan miqdordan oshishi mumkin. Natijalar to'plami orqali aylanish uchun pul to'lashga tayyor bo'lish oqilona emas deb ta'kidlash asosli ko'rinadi; demak, ratsional agent transitiv afzalliklarga ega bo'lishi kerak.

Tranzitivlik va to'liqlik aksiomalaridan kelib chiqadiki, tranzitivlik $>$ va $\geq$ aralashmalari uchun amal qiladi, shuning uchun agar $o_1 \geq o_2$ va $o_2 \geq o_3$ bo'lsa, u holda $o_1 \geq o_3$ va agar tranzitivlik aksiomasidagi bir yoki ikkala afzallik qat'iy bo'lsa, u holda xulosa qat'iy. Demak, agar $o_1 > o_2$ va $o_2 \geq o_3$, u holda $o_1 > o_3$. 12.1-mashqqa qarang.

12.3-aksioma (Monotonlik) *Boshqa sharoitlar teng bo'lsa, agent yaxshiroq natijaga erishish uchun kichikroq imkoniyatdan ko'ra yaxshiroq natijaga erishish imkoniyatini afzal ko'radi. Ya'ni, agar $o_1 > o_2$ va $p > q$ bo'lsa, u holda*

$$[p : o_1, (1-p) : o_2] > [q : o_1, (1-q) : o_2].$$

E'tibor bering, ushbu aksiomada natijalar orasidagi $>$ belgi agentning tanlovlarini ifodalaydi, p va q orasidagi $>$ belgisi esa sonlar o'rtasidagi tanish taqqoslashni ifodalaydi.

Quyidagi aksiomada aytilishicha, lotereyalar ustidan lotereyalar faqat natijalar va ehtimolliklarga bog'liq.

12.4-aksioma (Parchalanish) *(«qimor o'ynash qiziqarli emas»).* Agent qaysi lotereyalar bir xil natijalar uchun bir xil ehtimolliklarga ega ekanligiga ahamiyat bermaydi, hatto biri yoki ikkalasi ham lotereyalar ustidan lotereyalar bo'lsa ham. Masalan:

$$\begin{aligned} & [p : o_1, (1-p) : [q : o_2, (1-p) : o_3]] \\ & \sim [p : o_1, (1-p) * q : o_2, (1-p) * (1-p) : o_3]. \end{aligned}$$

Shuningdek, $o_1 \sim [1 : o_1, 0 : o_2]$ har qanday o_1 va o_2 natijalar uchun amal qiladi.

Ushbu aksioma shuni belgilaydiki, lotereya faqat natijalar va ularning ehtimollari bilan aniqlanadi. Agar agent qimor o'yinlarini afzal ko'rsa, bu natijalar maydonining bir qismi bo'lar edi.

Ushbu to'rtta aksioma natijalar va lotereyalar o'rtasidagi afzalliklarning ba'zi tuzilishini nazarda tutadi. Faraz qilaylik, $o_1 > o_2$ va $o_2 > o_3$. Agent quyidagi afzalliklardan qaysi birini afzal ko'rishini ko'rib chiqing:

$$o_2 \text{ yoki lotereya } [p : o_1, (1-p) : o_3]$$

turli qiymatlar uchun $p \in [0, 1]$. Agar $p = 1$ bo'lsa, agent lotereyani afzal ko'radi (chunki lotereyaning parchalanishi o_1 va $o_1 > o_2$ ga teng). Agar $p = 0$ bo'lsa, agent o_2 ni afzal ko'radi (chunki lotereya o_3 ga va $o_2 > o_3$ ga teng). Ba'zi bir bosqichda, p o'zgarganda, agentning afzalliklari $o_2 o_{-2} o_2$ ni afzal ko'rishdan lotereyani afzal ko'rishga o'zgarib turadi. 12.1-rasmda (521-bet) p o'zgarganda afzalliklar qanday o'zgarishi ko'rsatilgan. X o'qida p va Y o'qida o_2 yoki lotereyaning qaysi biri afzal ekanligi ko'rsatilgan. Quyidagi taklif ushbu intuitsiyani rasmiylashtiradi.

12.1-taklif. Agar agentning afzalliklari to'liq, tranzitiv bo'lsa va monotonlik aksiomasiga amal qilsa va agar $o_1 > o_2$ va $o_2 > o_3$ bo'lsa, u holda $0 \leq p_2 \leq 1$ soni mavjud, shundayki:

$$\begin{aligned} & (i.e., o_2[p : o_1, (1-p) : o_3]) \\ & p > p_2 \text{ uchun agent lotereyani afzal ko'radi ya'ni,} \\ & (i.e., [p : o_1, (1-p) : o_3] > o_2) \end{aligned}$$

Isbot. Agar $o_{22}[p:o_1,(1-p):o_3]$ har qanday p uchun bo'lsa, unda

$$P' < p, o_2 > [p' : o_1, (1-p') : o_3]$$

Shuningdek, agar $[p : o_1, (1-p) : o_3] \geq o_2$ har qanday p uchun bo'lsa, unda

$$p' > p, [p' : o_1, (1-p') : o_3] > o_2$$

To'liqlik bo'yicha, har bir p qiymati uchun, bu holatlarning biri to'g'ri bo'lishi kerak:

$$o_2 > [p : o_1, (1-p) : o_3], o_2 \sim [p : o_1, (1-p) : o_3]$$

yoki $[p:o_1,(1-p):o_3] > o_2$. Agar ba'zi p uchun $o_2 \sim [p : o_1, (1-p) : o_3]$ bo'lsa, u holda teorema to'g'ri. Aks holda, har qanday p parametri bilan lotereya yoki o_2 ni afzal ko'rish barcha katta yoki kichik qiymatlar uchun afzalliklarni bildiradi. Afzalliklari bizga noma'lum bo'lgan hududni qayta-qayta bo'lish orqali biz oxir-oqibat p_2 mezoniga javob beradigan qiymatga yaqinlashamiz.

Yuqoridagi taklif agentning p_2 nuqtasida qanday afzalliklari borligini belgilamaydi. Quyidagi aksioma agentning bu nuqtada befarqligini belgilaydi.

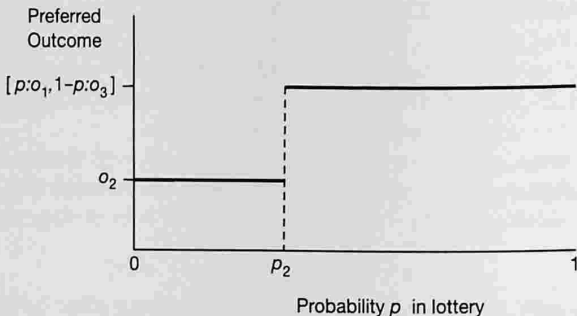
12.5-aksioma (Uzluksizlik) Agar $o_1 > o_2$ va $o_2 > o_3$, u holda $ap_2 \in [0, 1]$ mavjud bo'lishi kerak, shunda

$$o_2 \sim [p_2 : o_1, (1-p_2) : o_3]$$

12.6-aksioma (O'zaro almashinuvchanlik) Agar $o_1 \geq o_2$ bo'lsa, agent boshqa teng sharoitlarda, o_2 o'rniga o_1 ni o'z ichiga olgan lotereyalarni zaif afzal ko'radi. Ya'ni har qanday p soni va o_3 natijasi uchun:

$$[p : o_1, (1-p) : o_3] \geq [p : o_2, (1-p) : o_3]$$

Buning bevosita natijasi shundaki, agent befarq bo'lgan natijalar imtiyozlarni o'zgartirmasdan bir-birini almashtirishi mumkin.



12.1-rasm. o_2 va lotereya o'rtasidagi afzallik p funksiyasi sifatida

12.2-taklif. Agar agent substitutability (o'zaro almashinuvchanlik) aksiomasiga rioya qilsa va $o_1 \sim o_2$, agent faqat o_1 va o_2 bilan farqlanadigan lotereyalar o'rtasida befarq bo'ladi. Ya'ni har qanday p raqam va natija o_3 uchun quyidagi befarqlik munosabati mavjud:

$$[p : o_1, (1-p) : o_3] \sim [p : o_2, (1-p) : o_3]$$

Bu shundan kelib chiqadiki, $o_1 \sim o_2$, $o_1 \leq o_2$ va $o_2 \leq o_1$ ga teng bo'lib, biz o'zaro almashinuvchanlikni ikkala holat uchun ham qo'llashimiz mumkin.

Agar agent to'liqlik, tranzitivlik, monotonlik, parchalanish, uzluksizlik va o'zaro almashinuvchanlik aksiomalariga rioya qilsa, agent **ratsional** deb ta'riflanadi.

Ushbu ratsionallikning texnik ta'rifi sizning ratsionallikning intuitiv tushunchangizga mos keladimi yoki yo'qligini aniqlash o'zingizga bog'liq. Ushbu bo'limning qolgan qismida biz ushbu ta'rifning boshqa oqibatlarini ko'rsatamiz.

Garchi afzalliklar murakkab bo'lib tuyulishi mumkin bo'lsa-da, quyidagi teorema ratsional agentning natijaga bo'lgan qiymatini haqiqiy raqam bilan o'lchash mumkinligini ko'rsatadi. Ushbu qiymat o'lchovlari ehtimollik bilan birlashtirilishi mumkin, shuning uchun noaniqlik bilan afzalliklarni matematik kutish yordamida taqqoslanadi. Bu ikki sababga ko'ra hayratlanarli:

Afzalliklar bitta raqam bilan modellashtirish uchun juda ko'p qirrali bo'lib tuyulishi mumkin. Masalan, garchi afzalliklarni dollar bilan o'lchashga harakat qilish mumkin bo'lsa-da, hamma narsa sotilmaydi yoki osongina dollar va sentlar-ga aylantirilmaydi.

Qiymatlar ehtimollar bilan birlashtirilishi mumkin deb o'ylamaslik kerka. Agar agent har qanday pul qiymatlari x va y uchun va har qanday $p \in [0,1]$ uchun $\$(px + (1-p)y)$ va $[p : \$x, (1-p) : \$y]$ lotereyasi o'rtasida befarq bo'lsa, u kutilgan pul qiymati (EMV) agenti sifatida taniladi. Aksariyat odamlar EMV agentlari emas, chunki ular, masalan, $\$1,000,000$ va $[0.5 : \$0, 0.5 : \$2,000,000]$ lotereyasi o'rtasida aniq afzalliklarga ega (Qaysi birini afzal ko'rishingizni o'ylab ko'ring: bir million dollar yoki tanga uloqtirish, agar tanga old tomoni bilan tushsa, hech narsa olmaysiz yoki tanga orqa tomoni bilan tushsa, ikki million olasiz). Pul oddiygina ehtimollar bilan birlashtirilishi mumkin emas, shuning uchun birlashtirilishi mumkin bo'lgan qiymat mavjud bo'lishi ajablanarli bo'lishi mumkin.

12.3-taklif. *Agar agent ratsional bo'lsa, har bir oi natija uchun $u(o_i)$ deb nomlangan haqiqiy son mavjud bo'lib, u oining foydasi deyiladi, ya'ni:*

$o_i > o_j$ faqat va faqat $u(o_i) > u(o_j)$ bo'lsa;

foydalar ehtimollar bilan chiziqli bo'ladi:

$$u([p_1 : o_1, p_2 : o_2, \dots, p_k : o_k]) = p_1 u(o_1) + p_2 u(o_2) + \dots + p_k u(o_k)$$

Isbot: Agar agentda qat'iy afzal ko'rishlar bo'lmasa (ya'ni agent barcha natijalarga nisbatan befarq bo'lsa), unda barcha o natijalar uchun $u(o) = 0$ deb belgilang.

Aks holda, eng yaxshi natijani, o_{best} va eng yomon natijani, o_{worst} ni tanlang va har qanday natija o uchun oning foydasini p qiymati sifatida belgilang, ya'ni:

$$o \sim [p : o_{best}, (1-p) : o_{worst}]$$

Taklifning birinchi qismi o'zaro almashinuvchanlik va monotonlik xususiyatlaridan kelib chiqadi.

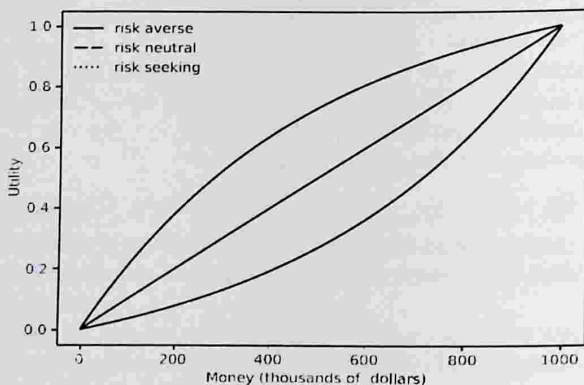
Ikkinchi qismini isbotlash uchun har qanday lotereya o_{best} va o_{worst} o'rtasidagi bitta lotereyaga qisqartirilishi mumkin, har bir o_i ni o_{best} va o_{worst} o'rtasidagi ekvivalent lotereya bilan almashtirib va parchalanish yordamida uni quyidagi shaklga keltirish mumkin:

$[p : o_{best}, (1-p) : o_{worst}]$ bu yerda $pp_1 u(o_1) + p_2 u(o_2) + \dots + p_k u(o_k)$ ga teng. Tafsilotlar mashq sifatida qoldiriladi.

Shunday qilib, aksiomalarga mos keladigan **ordinal preference (ordinal afzallik) cardinal preference (cardinal afzallik)**dir), unda foydalilik taqqoslanadigan qiymatlarni belgilaydi.

Ushbu isbotda natijalarning foydaliligi $[0,1]$ oralig'ida bo'lsa-da, har qanday chiziqli masshtablash bir xil natijani beradi. Ba'zida ehtimolliklardan farq qilish uchun $[0,100]$ yaxshi o'lchov bo'lishi mumkin va ba'zida natijalar xarajatlarga ega bo'lganda manfiy raqamlar ishlatilishi mumkin. Umuman olganda, dastur foydalanuvchi uchun intuitiv bo'lgan har qanday masshtabni qabul qilishi kerak. Pul va foydalilik o'rtasida odatda chiziqli bog'liqlik mavjud emas, hatto natijalarning pul qiymati bo'lsa ham. Odamlar ko'pincha pulga nisbatan **xavfga moyil emas**: ular \$n dollar olishni kutishgan, lekin ko'proq yoki kamroq olishlari mumkin bo'lgan tasodifiy sxemadan ko'ra qo'llarida \$n dollar bo'lishini afzal ko'rishadi.

12.1-misol. 12.2-rasmda uchta agent uchun pul va foydalilik o'rtasidagi mumkin bo'lgan munosabatlar ko'rsatilgan. Eng yuqori chiziqli xavfga moyil bo'lmagan, konveks foyda funksiyasiga ega agentni ifodalaydi. To'g'ri chiziqli grafik bilan ko'rsatilgan agent xavfga nisbatan neytral. Eng past chiziqli konveks foyda funksiyasiga ega bo'lgan xavfni yoqtiradigan agentni ifodalaydi.



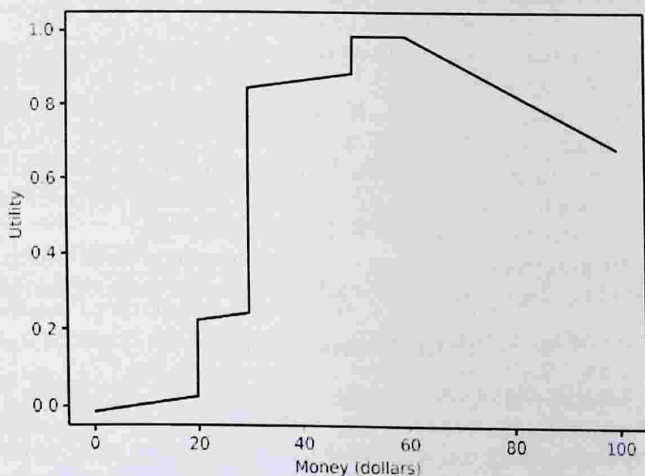
12.2-rasm. Har xil xavf profililariga ega agentlar uchun pul va foydalilik munosabatlari

12.2-rasmdagi xavfga moyil bo'lmagan agent \$300 000 dollarni hech narsaga ega bo'lmaslik yoki \$1 000 000 dollar olish imkoniyatining 50% dan ko'ra afzal ko'radi, lekin \$275 000 dollar olishdan ko'ra million dollarni qimorga tikishni afzal ko'radi. Buni foydalilik qiymati $utility = 0.5$ bo'lganini tekshirish orqali ko'rish mumkin. Bundan tashqari, ular yarim million dollarlik qimor o'yinini afzal ko'rish uchun million dollar yutib olish maqsadida 73% dan ko'proq imkoniyatga ega bo'lishlari kerak edi.

Xavfga moyil bo'lmagan agent uchun, $u(\$999,000) \approx 0.9997$. Shunday qilib, ushbu foydalilik funksiyasiga ko'ra, xavfga moyil bo'lmagan agent barcha pulni yo'qotish ehtimoli 0.03% ni yo'q qilish uchun \$1000 to'lashga tayyor bo'ladi. Bu **sug'urta** kompaniyalarining mavjudligining sababidir. Sug'urta kompaniyasiga, masalan, \$600 to'lash orqali xavfga moyil bo'lmagan agent o'zlari uchun \$999,000 qiymatidagi lotereyani \$1,000,000 qiymatidagi lotereyaga o'zgartirishi mumkin va sug'urta kompaniyalari, o'rtacha, taxminan \$300 to'lashni kutadilar va shuning uchun \$300 ishlab olishni kutadilar. Sug'urta kompaniyasi yetarlicha uylarni sug'urtalash orqali o'zining kutilgan qiymatini olishi mumkin. Bu ikkala tomon uchun ham foydalidir.

Ratsionallik foydalilik funksiyasining qanday ko'rinishda bo'lishiga hech qanday shart qo'ymaydi.

12.2-misol. 12.3-rasmda Kris uchun pul-foydalilik munosabatining mumkin bo'lgan shakli ko'rsatilgan. Kris 30 dollarlik o'yinchog'ni juda xohlaydi, lekin 20 dollarlikni ham yoqtiradi va ikkalasini ham olishni afzal ko'radi. Bulardan tashqari, pul Kris uchun unchalik ahamiyatga ega emas. Kris xavf-xatarni qabul qilishga tayyor. Masalan, agar Krisda 29 dollar bo'lsa, Kris adolatli garovda, masalan, tanga tashlashda boshqa agentning bir dollariga 9 dollar garov tikishdan juda xursand bo'ladi. Bu o'rinli, chunki bu 9 dollar Kris uchun unchalik foydali emas, lekin qo'shimcha dollar Krisga 30 dollarlik o'yinchoqni sotib olishga imkon beradi. Kris 60 dollardan ko'proq pulni xohlamaydi, chunki keyin Kris uni yo'qotish yoki o'g'irlanishidan xavotirga tushadi.



12.3-rasm. 12.2-misoldagi mumkin bo'lgan pul va foydalilik munosabatlari

Kutilgan foydalilik bilan bog'liq qiyinchiliklar

Kutilgan foydalilik nazariyasi oldida bir qator qiyinchiliklar mavjud edi. 1953-yilda taqdim etilgan Allais paradoksi [Allais va Hagen, 1979] quyidagicha. Quyidagi ikki variantdan qaysi birini afzal ko'rasiz?

A: \$1 million

B: lotereya [0.10 : \$2.5 million, 0.89 : \$1 million, 0.01 : \$0].

Xuddi shunday, quyidagi ikki variantdan qay birini tanlaysiz?

C: lotereya [0.11 : \$1 million, 0.89 : \$0]

D: lotereya [0.10 : \$2.5 million, 0.9 : \$0].

Ko'pchilik odamlar A ni B ga nisbatan va D ni C ga nisbatan afzal ko'rishadi. Ushbu tanlov ratsionallik aksiomalariga mos kelmaydi. Nima uchunligini ko'rish uchun, ikkala tanlovni ham bir xil shaklda ko'rish mumkin:

A, C: lotereya [0.11 : \$1 million, 0.89 : X]

B, D: lotereya [0.10 : \$2.5 million, 0.01 : \$0, 0.89 : X].

A va B da, X million dollar. C va D da, X nol dollar. Faqat farqli alternativalar qismlariga e'tibor qaratish intuitiv ko'rinadi, lekin odamlar aniq narsalarga afzallik berishadi.

Tverskiy va Kahneman [1974] bir qator insoniy tajribalarda odamlarning foydalilik nazariyasidan tizimli ravishda og'ishini ko'rsatdilar. Bunday og'ishlardan biri muammo taqdimotining **ramkalash effektidir**. Quyidagilarni ko'rib chiqing.

Kasallik 600 kishini nobud qilishi kutilmoqda. Ikki muqobil dastur taklif etilgan:

Dastur A: 200 kishi saqlanadi

Dastur B: 1/3 ehtimollik bilan 600 kishi saqlanadi va 2/3 ehtimollik bilan hech kim saqlanmaydi. Qaysi dasturni afzal ko'rasiz?

Kasallik 600 kishini nobud qilishi kutilmoqda. Ikki muqobil dastur taklif etilgan:

Dastur C: 400 nobud bo'ladi

Dastur D: 1/3 ehtimollik bilan hech kim o'lmaydi va 2/3 ehtimollik bilan 600 kishi nobud bo'ladi. Qaysi dasturga afzal ko'rasiz?

Tverskiy va Kahneman o'z tajribalarida odamlarning 72% Dastur A ni Dastur B dan afzal ko'rganini va 22% Dastur C ni Dastur D dan afzal ko'rganini ko'rsatdilar. Biroq, bu tanlovlar bir xil, faqat boshqacha tavsiflangan.

Istiqbollar nazariyasi, Kahneman va Tverskiy tomonidan ishlab chiqilgan, kutilgan foydalilikka muqobil bo'lib, inson xatti-harakatlariga yaxshiroq mos keladi.

12.1.2. Faktorli foydalilik

Agent uchun foydalilik natijalar yoki holatlar funksiyasi hisoblanadi. Foydaliliklarni xususiyatlar yoki o'zgaruvchilar nuqtayi nazaridan ifodalash, odatda, yanada ixcham ifodalar bilan natijalarni olishni osonlashtiradi va tabiiyroq bo'ladi.

Har bir natijani X_1, X_n xususiyatlar nuqtayi nazaridan tavsiflash mumkin deb faraz qilaylik. **Additiv foydalilikni** yig'indiga ajratish mumkin:

$$u(X_1, \dots, X_n) = f_1(X_1) + \dots + f_n(X_n)$$

Bunday dekompozitsiya **additiv mustaqillik** taxminini kiritadi. Buni amalga oshirish mumkin bo'lsa, bu **afzalliklarni aniqlashni** juda soddalashtiradi – foydalanuvchidan afzalliklarni aniqlash muammosini hal qiladi. Bu dekompozitsiya noyob emas, chunki bir terminni o'zgartirish va boshqasidan kamaytirish bir xil foydalilikni beradi. Additiv foydalilikning **kanonik ifodasi** noyob dekompozitsiyaga ega. Kanonik shakllarni qo'lga kiritish osonroq, chunki har bir raqamni boshqa raqamlarni hisobga olmasdan qo'lga kiritish mumkin. Additiv foydalilikni kanonik shaklga keltirish uchun, har bir X_i xususiyati uchun lokal foydalilik funksiyasini ta'riflang, $u_i(X_i)$, bu X_i qiymatining eng yomon natijasida 0 va eng yaxshi natijasida 1 qiymatiga ega bo'ladi va ijobiy real og'irlik w_i bilan. Og'irliklar yig'indisi 1 ga teng bo'lishi kerak. O'zgaruvchilar funksiyasining foydaliligi:

$$u(X_1, \dots, X_n) = w_1 * u_1(X_1) + \dots + w_n * u_n(X_n)$$

Bunday foydalilik funksiyasini aniqlash uchun har bir lokal foydalilik funksiyasini aniqlash va og'irliklarni baholash kerak. Har bir xususiyat, agar u muhim bo'lsa, agent uchun eng yaxshi qiymat va eng yomon qiymatga ega bo'lishi kerak. Lokal funksiyalar va og'irliklarni baholash quyidagicha amalga oshirilishi mumkin. Faqat X_j ni ko'rib chiqing; boshqa xususiyatlar o'xshash tarzda ko'rib chiqilishi mumkin. X_j xususiyati uchun X_j uchun qiymatlar x_j va x'_j va $X_2, \dots, X_n$ uchun fiksatsiyalangan qiymatlar $x_2, \dots, x_n$:

$$u(x_1, x_2, \dots, x_n) - u(x'_1, x_2, \dots, x_n) = w_1 * (u_1(x_1) - u_1(x'_1)). \quad 12.1$$

Og'irlik w_j eng yaxshi natija x_j va eng yomon natija x'_j bo'lganda hosil qilinishi mumkin (chunki u holda $u_1(x_j) - u_1(x'_j) = 1$) X_j domenidagi boshqa qiymatlar uchun u_j qiymatlari (12.1) tenglamasidan foydalangan holda hisoblanishi mumkin, x'_1 eng yomon natija deb hisoblanadi (chunki $u_1(x'_1) = 0$)

Additiv mustaqillikni qabul qilish kuchli mustaqillik taxminini o'z ichiga oladi. Xususan, (12.1) tenglamasida, foydaliliklardagi farq $X_2, \dots, X_n$ qiymatlari uchun bir xil bo'lishi kerak.

Additiv mustaqillik ko'pincha yaxshi taxmin emas. X va Y ikki xususiyatlari mavjudligini ko'rib chiqing, domenlari $\{x_\theta, x_j\}$ va $\{y_\theta, y_j\}$.

Agar ikkalasiga ega bo'lish, alohida-alohida ikkitasiga ega bo'lishdan yaxshiroq bo'lsa, X va Y ikki qiymatlari **o'zaro to'ldiruvchi** bo'ladi. Formal ravishda, x_j va y_j qiymatlari o'zaro to'ldiruvchi bo'lsa, birini olish boshqa biriga ega bo'lganda, agent boshqa biriga ega bo'lmagandan ko'ra ko'proq qadrlı bo'ladi:

$$u(x_1, y_1) - u(x_0, y_1) > u(x_1, y_0) - u(x_0, y_0)$$

Ikki qiymat **o‘rinbosarlar** bo‘lishadi, agar ikkalasi ham har birining qiymatlari yig‘indisi kabi qimmatli bo‘lmasa. Formal ravishda, x_1 va y_1 qiymatlari o‘rinbosarlar bo‘lsa, birini olish, agent boshqa biriga ega bo‘lganida, boshqa biriga ega bo‘lmagandan ko‘ra kam ahamiyatli bo‘ladi:

$$u(x_1, y_0) - u(x_0, y_0) > u(x_1, y_1) - u(x_0, y_1)$$

12.3-misol. Safar domenidagi xaridor agenti uchun, ma‘lum bir kunga samolyot va o‘sha kunga mehmonxona bronlash uchun foydalilik funksiyasini ko‘rib chiqing:

<i>Plane</i>	<i>Hotel</i>	<i>Utility</i>
<i>true</i>	<i>true</i>	100
<i>true</i>	<i>false</i>	0
<i>false</i>	<i>true</i>	10
<i>false</i>	<i>false</i>	20

Demak, $u(\text{plane}, \text{hotel}) - u(\neg \text{plane}, \text{hotel}) = 90 > u(\text{plane}, \neg \text{hotel}) - u(\neg \text{plane}, \neg \text{hotel}) = -20$.

Shunday qilib, samolyot bronlash va mehmonxona bronlash bir-birini to‘ldiradi: biri boshqasiz yaxshi natija bermaydi.

Agar dam olishga chiqqan kishi bir sayohatdan zavqlansa, lekin bir kunda ikkitasini yoqtirmasa, bir kunda ikkita sayohat o‘rnini bosuvchi bo‘ladi.

Additiv foydalilik bir-birini to‘ldiruvchi yoki o‘rin bosuvchi bo‘lmagan holatlarni qabul qiladi. O‘zaro ta‘sir mavjud bo‘lsa, yanada murakkab model talab qilinadi, masalan, **umumiy additiv mustaqillik** modeli, bu foydalilikni bir necha o‘zgaruvchilar ustida omil bo‘lishi mumkin bo‘lgan terminlar yig‘indisi sifatida ifodalaydi. Umumiy additiv mustaqillik modelini aniqlash additiv modelni aniqlashdan ancha murakkab, chunki o‘zgaruvchi ko‘plab omillarda paydo bo‘lishi mumkin.

Boolean xususiyatlari uchun, ehtimollikning **kanonik ifodasi** bilan analogi-yani ishlatib, ixtiyoriy foydalilikni ifodalash mumkin. Bu additiv foydalilikni atomlarning birikmalari uchun og‘irliklarni (bu *true* 1 va *false* 0 bo‘lganda mahsulotga to‘g‘ri keladi) shu jumladan, og‘ish atamasini qo‘shish uchun kengaytiradi. O‘zaro to‘ldiruvchilar birikma uchun ijobiy og‘irlikka ega va o‘rnini bosuvchilar birikma uchun salbiy og‘irlikka ega bo‘ladi.

12.4-misol. 12.3-misolni ko‘rib chiqing. Ma‘lum kun uchun samolyot va mehmonxona bronlash uchun foydalilik, jadvalda ko‘rsatilgandek, quyidagi formula yordamida ifodalanishi mumkin:

$$\text{utility}(\text{Plane}, \text{Hotel}) = w_0 + w_1 * \text{Plane} + w_2 * \text{Hotel} + w_3 * \text{Plane} * \text{Hotel}$$

bu yerda $w_0 = 20$, $w_1 = 20$, $w_2 = 10$, $w_3 = 110$

12.3-misoldagi ikkita sayohat uchun, odam ikkalasini ham xohlamasligi holda, mahsulot termini uchun og'irlik salbiy bo'ladi.

Odatda, faqat oddiyroq o'zaro ta'sirlarni ifodalash, masalan, faqat alohida atomlarning og'irliklarini va ba'zi bir juft mahsulotlarni ifodalash va kerak bo'lganda faqat ko'proq atomlarning mahsulotlarini kiritish bilan boshlanadi. Agar barcha birikmalar kiritilgan bo'lsa, n takliflar uchun $2n$ og'irliklar bo'ladi, bu barcha haqiqat qiymatlari kombinatsiyalarining jadvaliga teng. Ko'p og'irliklar nol bo'lganida, kanonik tasvirlash foydali bo'ladi va ularni umuman ifodalash kerak emas.

12.1.3. Istiqbollar nazariyasi

Foydalilik nazariyasi ratsional agentlarning **normativ nazariyasi** bo'lib, bir qator aksiomalar bilan asoslanadi. **Istiqbollar nazariyasi** esa odamlarning qanday qarorlar qabul qilishini tasvirlashga intiladigan **tasviriy nazariya**. Tasviriy nazariya odamlarning xatti-harakatlarini kuzatish va boshqariladigan psixologik tajribalar o'tkazish orqali baholanadi.

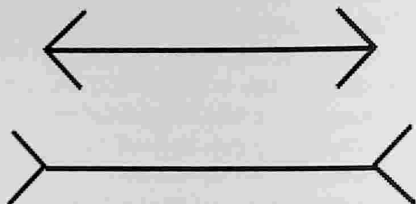
Natijalarga nisbatan afzalliklarni belgilash o'rniga, istiqbollar nazariyasi afzalliklar kontekstini ko'rib chiqadi. Psixologiyada odamlar mutlaq qiymatlarni emas, balki kontekstdagi qiymatlarni qabul qilishlari haqida yaxshi asoslangan g'oya mavjud. Myuller-Lyer illyuziyasini ko'rib chiqing. Gorizontaal chiziqlar teng uzunlikda, ammo boshqa chiziqlar kontekstida ular turlicha ko'rinadi. Boshqa misol, agar bir qo'lingizni sovuq suvga, ikkinchisini issiq suvga solib, keyin ikkalasini ham iliq suvga solsangiz, iliq suv har bir qo'lga har xil tuyuladi. Odamlarning afzalliklari ham kontekstga bog'liq. Istiqbollar nazariyasi, odamlarning afzalliklari natijalarga emas, balki tanlovning hozirgi vaziyatdan qanchalik farq qilishiga bog'liq ekanligini kuzatishga asoslanadi.

Pul va qiymat o'rtasidagi munosabat istiqbollar nazariyasi tomonidan bashorat qilinadi (12.5-rasm). Bu grafikda x -o'qida mutlaq boylik o'rniga hozirgi boylikdan farq ko'rsatilgan. X -o'qining boshlanish nuqtasi – shaxsning hozirgi boylik holati. Bu holat **referens nuqtasi** deb ataladi. Istiqbollar nazariyasi quyidagilarni bashorat qiladi:

Foyda uchun odamlar tavakkalni yomon ko'radilar. Buni hozirgi boylik ustidagi egri chiziq concave ekanligi ko'rsatadi.

Yo'qotishlar uchun odamlar tavakkalni yaxshi ko'radilar. Buni hozirgi boylik ostidagi egri chiziq qavariq ekanligi ko'rsatadi.

Yo'qotishlar taxminan foydadan ikki barobar yomonroqdir. Yo'qotishlar uchun qiyalik foydalardan ko'ra tikroq.



12.4-rasm. Insonning uzunlikni idrok etishi kontekstga bog'liq

Bu munosabat faqat pul bilan emas, balki qiymatga ega bo'lgan boshqa narsalar bilan ham bo'lishi mumkin. Istiqbollar nazariyasi odamlarning qanday harakat qilishlarini foydalilik nazariyasidan farqli tarzda bashorat qiladi, quyidagi Kahnemannning [2011] misollarida ko'rsatilgandek.

12.5-misol. Entoni va Betti haqida o'ylab ko'ring:

Entonining hozirgi boyligi 1 million dollar.

Bettining hozirgi boyligi 4 million dollar. Ular ikkalasi ham tavakkal va aniq narsani tanlash imkoniyatiga ega.

Tavakkal: 1 million yoki 4 million dollar egasi bo'lish imkoniyati teng.

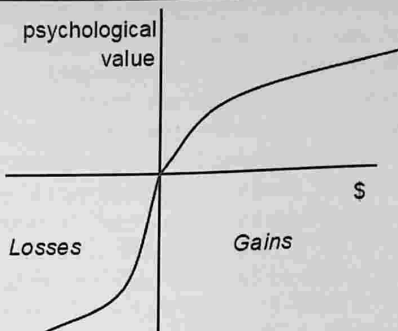
Aniq narsa: 2 million dollar egasi bo'lish.

Foydalilik nazariyasi, agar ularning foydalilik egri chiziqlari bir xil bo'lsa, Entoni va Betti bir xil tanlov qilishini bashorat qiladi, chunki natijalar bir xil. Foydalilik nazariyasi hozirgi boylikni hisobga olmaydi. Istiqbollar nazariyasi Entoni va Betti uchun turli bashoratlar qiladi. Entoni foyda qilmoqda va shuning uchun tavakkalni yomon ko'radi va ehtimol aniq narsani tanlaydi. Betti yo'qotish qilmoqda va shuning uchun tavakkalni yaxshi ko'radi va tavakkalni tanlaydi. Entoni 2 million dollardan mamnun bo'ladi va baxtsiz bo'lishni xavfini istamaydi. Betti 2 million dollardan mamnun bo'lmaydi va tavakkal qilish orqali baxtli bo'lish imkoniyatiga ega bo'ladi.

12.6-misol. Egizaklar Endi va Bobbi, ikkisi ham bir xil didli va bir xil ish boshlagan. Ikkita ish mavjud, ular bir xil, faqat:

Ish A 10,000 dollarlik ish haqini oshirishni beradi.

Ish B har oy bir kunlik qo'shimcha ta'til beradi.



12.5-rasm. Istiqbol nazariyasi uchun pul-qiyamat munosabatlari

Endi va Bobbi ikkalasi ham natijalariga befarq va tanga otish orqali qaror qabul qiladilar. Endi A ishini tanlaydi va Bobbi B ishini tanlaydi.

Endi kompaniya ularga \$500 bonus bilan ishlarini almashtirishni taklif qiladi. Foydalilik nazariyasi ularning ishlarini almashtirishlarini bashorat qiladi. Ular befarq edilar va endi ishlarini almashtirish orqali \$500 bilan yaxshiroq holatda bo'lishlari mumkin.

Istiqbollar nazariyasi ularning ishlarini almashtirmasliklarini bashorat qiladi. Endi ularning ishlarini qabul qilganliklari uchun, ularning referens nuqtalari farq qiladi. Endi \$10,000 yo'qotish haqida o'ylaydi. Bobbi 12 kun ta'tilni yo'qotish haqida o'ylaydi. Yo'qotish \$500 qo'shimcha pul va maosh yoki ta'tildan ko'ra yomonroq. Ularning har biri o'z ishlarini afzal ko'radi.

Empirik dalillar shuni ko'rsatadiki, istiqbollar nazariyasi odamlarning qarorlarini bashorat qilishda foydalilik nazariyasidan yaxshiroq. Biroq, bu sun'iy agent uchun eng yaxshi ekanligini anglatmaydi. Biroq, odamlar bilan o'zaro ta'sir qilishi kerak bo'lgan sun'iy agent odamlarning qanday fikr yuritishini hisobga olishi kerak. Ushbu bobning qolgan qismida biz sun'iy agentning qaror qabul qilish va rejalashtirish asosida foydalilik nazariyasini qabul qilamiz.

12.2. Yagona qarorlar

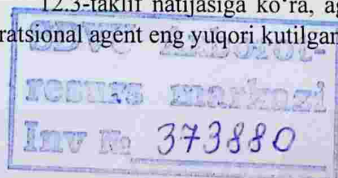
Aqli agentlarga asoslangan asosiy qarorlar nazariyasi quyidagi farazlarga asoslanadi:

Agentlar bajarishi mumkin bo'lgan harakatlar haqida biladilar.

Har bir harakatning ta'siri natijalarga nisbatan ehtimollik taqsimoti sifatida tasvirlanishi mumkin.

Agentning afzalliklari natijalarining foydaliligi bilan ifodalanadi.

12.3-taklif natijasiga ko'ra, agar agentlar faqat bir qadam harakat qilishsa, ratsional agent eng yuqori kutilgan foydaga ega bo'lgan harakatni tanlashi kerak.



12.7-misol. Yetkazib beruvchi robot muammosini ko'rib chiqing, bunda uning harakatlari natijasida noaniqlik mavjud. Xususan, 3.1-rasmda ko'rsatilgan o109 pozitsiyasidan pochta pozitsiyasiga o'tish muammosini ko'rib chiqing, bu yerda robot yo'ldan chiqib, zinapoyadan yiqilib tushishi ehtimoli mavjud. Robot himoya yostiqlachalarini olishi mumkin, bu hodisa ehtimolini o'zgartirmaydi, ammo hodi-saning og'irligini kamaytiradi. Afsuski, himoya yostiqlachalari qo'shimcha og'irlik qo'shadi. Robot, shuningdek, uzoq yo'ldan borishi mumkin, bu hodisa ehtimolini kamaytiradi, ammo sayohatni ancha sekinlashtiradi. Shunday qilib, robot himoya yostiqlachalarni kiyishi va qaysi yo'ldan borishni (uzoq yoki qisqa yo'ldan) qaror qilishi kerak. Bevosita boshqarish ostida bo'lmagan narsa-hodisa sodir bo'lish eh-timoli mavjud bo'lib, bu ehtimol uzoq yo'ldan borish orqali kamaytirilishi mum-kin. Agentning tanlovlari va hodisa sodir bo'lishi natijasida og'ir zarardan tortib himoya yostiqlachalarsiz tezda yetib kelishga qadar har bir kombinatsiya uchun na-tija mavjud.

Kimning qadriyatlar?

Har qanday kompyuter dasturi yoki maslahatchi harakat qilishi yoki mas-lahat berishi uchun qandaydir qiymat tizimidan foydalanishi kerak.

Alisa shunday davom etdi: «Iltimos, menga ayta olasizmi, bu yerdan qa-yerga borishim kerak?» Bu «qayerga borishni xohlashingizga bog'liq», dedi Mushuk. «Menga baribir», dedi Alisa. «Unda qaysi yo'lga borishingizning ahamiyati yo'q», dedi Mushuk.

– Luis Kerrol,

«Alisa mo'jizalar mamlakatida», 1865

Albatta, barchamiz kompyuterlarning bizning qiymat tizimimizga muvo-fiq ishlashini xohlaymiz, lekin ular hammaga bir xil qoidalarga asoslangan tarzda harakat qila olmaydilar. Dasturlar laboratoriyada ishlaganda bu odatda muammo emas. Dastur dizaynerning, ya'ni foydalanuvchining maqsadlari va qadriyatlariga muvofiq harakat qiladi. Tizimning bir nechta foydalanuvchisi bo'lganda, qaysi qiymat tizimi dasturga kiritilganligini bilishingiz kerak. Agar kompaniya shifokorga tibbiy diagnostika dasturini sotsa, dastur beradigan maslahatlar jamiyatning, kompaniyaning, shifokorning, bemorning yoki kim to'layotgan bo'lsa, shuning qadriyatlarini aks ettiradimi (bularning barchasi juda turli qiymat tizimlariga ega bo'lishi mumkin)? Bu shifokorning yoki be-morning qadriyatlarini aniqlaydimi? Avtonom avtomobillar uchun harakatlar egasining foydasinimi yoki jamiyatning foydasini aks ettiradimi? Yo'lni kesib o'tayotgan odamlarni jarohatlash yoki piyodalarni chetlab o'tish uchun oila

a'zolarini jarohatlash o'rtasida tanlov qilish haqida o'ylab ko'ring. Turli *n* va *m* qiymatlari uchun hayotlarning qiymatlari qanday savdolashadi va jarohat olish yoki o'lish imkoniyatlari qanday farq qiladi? O'z oilasini eng ko'p himoya qilishni xohlaydigan haydovchilar piyodalardan farqli savdolar qilgan bo'lar edilar. Bu holat **trolley muammolari** yordamida o'rganilgan, bu yerda savdolar aniq qilib berilgan va odamlar o'zlarining axloqiy fikrlarini bildiradilar. 2.4-mavzuga qarang.

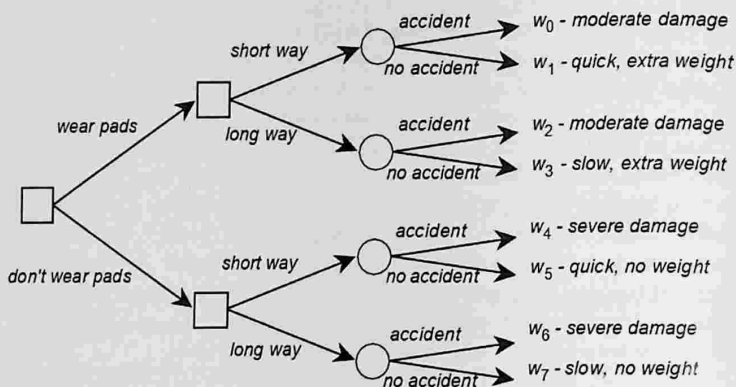
Agar kingadir maslahat beradigan tizim qurmoqchi bo'lsangiz, haqiqatni ham, ularning qadriyatlarini ham aniqlashingiz kerak. Masalan, tibbiy diagnostika tizimida tegishli muolaja nafaqat bemorlarning simptomlariga, balki ularning tanlovlariga ham bog'liq. Ular atrofdagi narsalarni ko'proq sezish uchun biroz og'riqqa chidashga tayyormi? Ular uzoqroq yashash uchun ko'p noqulaylikka bardosh berishga tayyormi? Ular qanday xatarlarga tayyor? Agar nima qilishni xohlayotganingizni so'ramasa, sizga nima qilish kerakligini aytadigan dastur yoki shaxsdan doimo ehtiyot bo'ling! Nima qilish yoki maslahat beradigan dasturlarni yaratishda harakatlar yoki maslahatlar asosida kimning qadriyat tizimlari kiritilganligini bilishingiz kerak. Agar odamlar ta'sir qilsa, ularning afzalliklarini hisobga olish kerak yoki hech bo'lmaganda ular qaysi afzalliklar qarorlar asosida ishlatilayotganini bilishlari kerak.

Yagona qaror qabul qilish jarayonida **qaror o'zgaruvchisi** agentning tanlovini modellash uchun ishlatiladi. Qaror o'zgaruvchisi tasodifiy o'zgaruvchiga o'xshaydi, lekin u bilan bog'liq ehtimollik taqsimoti mavjud emas. Buning o'rniga, agent qaror o'zgaruvchisi uchun qiymat tanlash huquqiga ega bo'ladi. **Mumkin bo'lgan dunyo** tasodifiy va qaror o'zgaruvchilari uchun qiymatlarni belgilaydi. Har bir mumkin bo'lgan dunyo bilan bog'liq bir foydalilik mavjud. Qaror o'zgaruvchilari uchun qiymatlar kombinatsiyasi uchun tasodifiy o'zgaruvchilar uchun ehtimollik taqsimoti mavjud. Ya'ni har bir qaror o'zgaruvchisi uchun qiymat tayinlash uchun bu tayinlashni qanoatlantiradigan dunyolar o'lchovlari 1 ga teng.

12.6-rasm agent uchun mavjud bo'lgan turli tanlovlarni va bu tanlovlarning natijalarini ko'rsatadigan **qarorlar daraxtini** tasvirlaydi. Qarorlar daraxtini o'qish uchun ildizdan boshlang (bu rasmda chap tomonda). Kvadratlar bilan ko'rsatilgan qarorlar tugunlari uchun agent qaysi shoxni olishni tanlaydi. Har bir tasodifiy tugun uchun, doira bilan ko'rsatilgan, agent qaysi shox olinishini tanlay olmaydi; aksincha, bu tugundan shoxlar bo'ylab ehtimollik taqsimoti mavjud. Har bir barg bir dunyoni bildiradi, agar o'sha bargga borish yo'li kuzatilsa, bu **natija** bo'ladi.

12.8-misol. 12.7-misolda ikkita qaror o'zgaruvchisi mavjud, biri robotning himoya yostiqlchalarni kiyishi qarori va biri qaysi yo'ldan borish qarori bilan bog'liq.

Bir tasodifiy o'zgaruvchi mavjud, ya'ni hodisa sodir bo'lishi yoki bo'lmasligi. 12.6-rasmdagi qarorlar daraxtidagi sakkiz yo'lga mos keladigan sakkizta mumkin bo'lgan dunyo mavjud. Agentning nima qilishi kerakligi, tez yetib borish qanchalik muhimligi, himoya yostiqchalarning og'irligi qanchalik muhimligi, zararlarning og'irlikdan o'rtacha darajaga kamayishi qanchalik qimmatga tushishi va hodisa ehtimoli qanchalik yuqori ekanligiga bog'liq.



12.6-rasm. Yetkazib beruvchi robot uchun qarorlar daraxti. Kvadrat qutilar robot qabul qilishi mumkin bo'lgan qarorlarni ifodalaydi. Doiralalar robot qaror qabul qilishdan oldin kuzata olmaydigan tasodifiy o'zgaruvchilarni ifodalaydi

12.3-taklifning isboti natijalarning istalganligini qanday o'lchashni aniqlaydi. Aytaylik, biz foydalilik qiymatlarni $[0,100]$ diapazonda bo'lishiga qaror qildik. Avval, eng yaxshi natijani tanlang, bu w_5 bo'ladi va unga 100 foydalilik qiymati beriladi. Eng yomon natija w_0 , shuning uchun unga 0 qiymat bering. Boshqa dunyolar uchun w_6 va w_3 orasidagi lotereyani ko'rib chiqing. Masalan, w_0 ning foydalilik qiymati 35 bo'lishi mumkin, ya'ni agent w_0 va $[0.35 : w_5, 0.65 : w_6]$ o'rtasida befarqdir, bu esa w_2 dan biroz yaxshiroq, u 30 foydalilik qiymatga ega bo'lishi mumkin. w_1 ning foydalilik qiymati 95 bo'lishi mumkin, chunki u faqat w_5 dan biroz yomonroq.

12.9-misol. Tibbiy diagnostikada qaror o'zgaruvchilari turli davolash va tahlillarga to'g'ri keladi. Foydalilik tahlillar va davolash xarajatlariga va bemorning tuzalishi, kasal bo'lib qolishi yoki o'lishi, qisqa muddatli yoki surunkali og'riq his qilishiga bog'liq bo'lishi mumkin. Bemorning natijalari bemor qabul qiladigan davolashga, bemorning fiziologiyasiga va kasallikning tafsilotlariga bog'liq bo'ladi, bu esa aniqlik bilan ma'lum bo'lmasligi mumkin.

Samolyotlar kabi artefaktlarning diagnostikasi uchun ham xuddi shu yondashuv qo'llaniladi; muhandislar komponentlarni sinab ko'radilar va ularni tuza-tadilar. Samolyotlarda, ehtimol, foydalilik funksiyasi avariylarni minimallashti-rish (xavfsizlikni maksimal darajada oshirish) bo'lishiga umid qilasiz, lekin bun-day qaror qabul qilishda kiritilgan foydalilik ko'pincha kompaniya uchun foydani maksimal darajada oshirish uchun bo'ladi va avariyalarga shunchaki hisobga oli-nadigan xarajatlar sifatida qaraladi.

Yagona qaror qabul qilish jarayonida agent har bir qaror o'zgaruvchisi uchun qiymatni bir vaqtning o'zida tanlaydi. Bu barcha qaror o'zgaruvchilarini bitta kompozit qaror o'zgaruvchisi, D sifatida ko'rib model qilish mumkin. Ushbu qa-ror o'zgaruvchisining domeni individual qaror o'zgaruvchilarining domenlarining o'zaro mahsulidir.

Har bir dunyo ω qaror o'zgaruvchisi D uchun qiymat tayinlash va har bir taso-difiy o'zgaruvchi uchun qiymat tayinlashni belgilaydi. Har bir dunyo foydalilikka ega, u o'zgaruvchi u tomonidan beriladi. Yagona qaror qaror o'zgaruvchisi uchun qiymat tayinlashdir. Yagona qaror $D = \text{dining}$ kutilgan foydaliligi $E(u \mid D = d_i)$ bo'lib, bu qarorga bog'liq foydalilikning kutilgan qiymati. Bu dunyolarning o'rta-cha foydaliligi bo'lib, dunyolar ehtimollikka qarab o'lchanadi:

$$E(u \mid D = d_i) \sum_{\omega: D(\omega) = d_i} P(\omega)$$

bu yerda $D(\omega)$ dunyosida D o'zgaruvchisining qiymatini bildiradi, $u(\omega)$ ω dunyosidagi foydalilik qiymati va $P(\omega)$ ω dunyosining ehtimoli. Optimal yagona qaror kutilgan foydaliligi maksimal bo'lgan qarordir. Ya'ni, $D = d_{\max}$ optimal qaror bo'ladi, agar:

$$E(u \mid D = d_{\max}) = \max_{d_i \in \text{domain}(D)} E(u \mid D = d_i)$$

bu yerda $\text{domain}(D)$ qaror o'zgaruvchisi D ning domeni. Shunday qilib:

$$d_{\max} = \arg \max_{d_i \in \text{domain}(D)} E(u \mid D = d_i)$$

12.10-misol. 12.7-misoldagi yetkazib beruvchi robot muammosi robotning *Wear pads* Yostiqchalarni kiyish va *Which way* Qaysi yo'lga borishni tanlash uchun qiymatlarni aniqlashni o'z ichiga oladi. Ushbu murakkab qaror o'zgaruv-chisi, *Wear pads*, *Which way* ni aniqlaydi. Har bir qaror o'zgaruvchisiga qiymat tayinlashning kutilgan qiymati bor. Masalan, $\text{Wear pads} = \text{true} \wedge \text{Which way} = \text{short}$ qarorining kutilgan foydasi quyidagicha aniqlanadi:

$$\begin{aligned} & E(u \mid \text{wear_pads} \wedge \text{Which_way} = \text{short}) \\ &= P(\text{accident} \mid \text{wear_pads} \wedge \text{Which_way} = \text{short}) * u(w_0) \\ &+ (1 - P(\text{accident} \mid \text{wear_pads} \wedge \text{Which_way} = \text{short})) * u(w_1) \end{aligned}$$

bu yerda $u(w_i)$ dunyodagi w_i foyda qiymatini bildiradi, w_0 va w_1 olamlari 12.6-rasm-da ko'rsatilgan va yostiqchalarni kiyish $Wear\ pads = true$ degan ma'noni anglatadi.

12.2.1. Bir bosqichli qaror tarmoqlari

Qaror daraxti holatga asoslangan tasvir bo'lib, ildizdan barggacha bo'lgan har bir yo'l holatni bildiradi. Ammo ko'pincha xususiyatlar nuqtayi nazaridan to'g'ri-dan to'g'ri ifodalash va mulohaza yuritish tabiiy va samaraliroqdir, bu o'zgaruv-chilar sifatida ifodalanadi.

Bir bosqichli qaror tarmog'i uch turdagi tugunlar bilan kengaytirilgan ish-onch tarmog'idir:

Qaror tugunlari, to'rtburchaklar shaklida chizilgan, qaror o'zgaruvchilari-ni ifodalaydi. Agent har bir qaror o'zgaruvchisi uchun qiymat tanlaydi. Agar bir nechta qaror o'zgaruvchilari bo'lsa, biz qaror tugunlari uchun umumiy tartibni taxmin qilamiz va ushbu umumiy tartibdagi D qaror tugunidan oldingi qaror tu-gunlari D ning asl to'plamlari bo'ladi.

Tasodifiy tugunlar, oval shaklida chizilgan, tasodifiy o'zgaruvchilarni ifoda-laydi. Bular ishonch tarmog'idagi tugunlar bilan bir xil. Har bir tasodifiy tugun-ning o'ziga xos domeni va asl to'plamlariga nisbatan shartli ehtimoli bor. Ishonch tarmog'ida bo'lgani kabi, tasodifiy tugunning asl to'plamlari shartli bog'liqlikni ifodalaydi: o'zgaruvchi, asl to'plamlariga nisbatan, o'z avlodlari bo'lmaganlari-dan mustaqildir. Qaror tarmog'ida tasodifiy tugunlar va qaror tugunlari tasodifiy tugunning asl to'plamlari bo'lishi mumkin.

Bir dona foydalilik tuguni, olmos shaklida chizilgan, foydani ifodalaydi. Foydalilik tugunining asl to'plamlari foydalilik bog'liq bo'lgan o'zgaruvchilardir. Tasodifiy tugunlar va qaror tugunlari foydalilik tugunining parents (asl to'plamlari) bo'lishi mumkin.

Har bir tasodifiy o'zgaruvchi va har bir qaror o'zgaruvchisining domeni bor. Foydalilik tuguni uchun domen mavjud emas. Tasodifiy tugunlar tasodifiy o'zga-ruvchilarni ifodalasa, qaror tugunlari qaror o'zgaruvchilarini ifodalaydi, lekin foy-dalilik o'zgaruvchisi yo'q.

Qaror tarmog'iga tegishli bo'lgan har bir imkoniyat tuguni uchun asl to'plam-lariga qarab shartli ehtimollik (ishonch tarmog'ida bo'lgani kabi) va foydalilik tugunining parents (asl to'plamlari)ning funksiyasi sifatida foydalilik mavjud. Tarmoqni belgilashda qaror bilan bog'liq funksiyalar yo'q (garchi algoritm funk-siyani tuzsa ham).

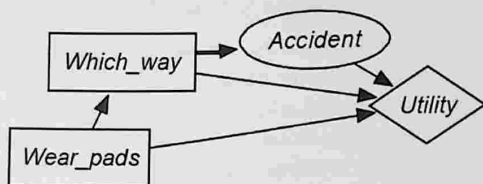
12.11-misol. 12.7-rasmda 12.7-misolning qaror tarmog'i tasviri berilgan. Qaysi yo'ldan borish va qaysi yostiqni kiyish qarorlarini qabul qilish kerak. Agentning halokatga uchrashi faqat qaysi yo'ldan borishiga bog'liq. Foydalilik esa uchta o'zgaruvchiga bog'liq.

Ushbu tarmoq ikkita omilni talab qiladi: shartli ehtimollikni ifodalovchi omil, $P(Accident \mid Which\ way)$ va foydalilikni $Which\ way, Accident$ va $Wear\ pads$

funksiyasi sifatida ifodalovchi omil. Ushbu omillar uchun jadvallar 12.7-rasmda ko'rsatilgan.

Bir bosqichli qaror tarmog'i uchun **tamoyil** har bir qaror o'zgaruvchisi uchun qiymat tayinlashni ifodalaydi. Har bir tamoyilning kutilgan foydasi bor. Optimal tamoyil eng katta kutilgan foydaga ega bo'lgan tamoyildir. Ya'ni bu shunday tamoyilki, hech qanday boshqa tamoyil undan yuqori kutilgan foydaga ega emas. Qaror tarmog'ining qiymati tarmoqning optimal tamoyilining kutilgan foydasi hisoblanadi.

12.8-rasmda bitta bosqichli qaror tarmog'ida optimal tamoyilni topish uchun o'zgaruvchilarni istisno qilish ko'rsatilgan. Tegishli bo'lmagan tugunlarni kesib tashlash va barcha tasodifiy o'zgaruvchilarni jamlagandan so'ng, har bir qaror o'zgaruvchilari kombinatsiyasining kutilgan foydasini ifodalovchi bitta omil bo'ladi. Bu omil barcha qaror o'zgaruvchilaridagi omil bo'lishi shart emas.



	Which way	Accident	Value
	short	true	0.2
	short	false	0.8
	long	true	0.01
	long	false	0.99

Wear pads	Which way	Accident	Utility
true	short	true	35
true	short	false	95
true	long	true	30
true	long	false	75
false	short	true	3
false	short	false	100
false	long	true	0
false	long	false	80

12.7-rasm. Yetkazib beruvchi robot uchun bir bosqichli qaror tarmog'i

Biroq, kiritilmagan qaror o'zgaruvchilari qaror uchun ahamiyatga ega emas.

12.12-misol. 12.7-rasmdagi qaror tarmog'ida *VE_SSDN* ni ishga tushirishni ko'rib chiqing. Hech qanday tugunlarni qisqartirib bo'lmaydi, shuning uchun u

yagona tasodifiy o'zgaruvchi, ya'ni *Accident* (Halokat)ni chiqarib tashlaydi. Buning uchun ikkala omil ham *Accident* ni o'z ichiga olgani uchun ularni ko'paytiradi va *Accident* ni chiqarib tashlab, quyidagi omilni beradi:

<i>Wear_pads</i>	<i>Which_way</i>	Value
<i>true</i>	<i>short</i>	$0.2 * 35 + 0.8 * 95 = 83$
<i>true</i>	<i>long</i>	$0.01 * 30 + 0.99 * 75 = 74.55$
<i>false</i>	<i>short</i>	$0.2 * 3 + 0.8 * 100 = 80.6$
<i>false</i>	<i>long</i>	$0.01 * 0 + 0.99 * 80 = 79.2$

Shunday qilib, maksimal qiymatga ega bo'lgan tamoyil – optimal tamoyil – qisqa yo'ldan borish va yostiqchalarni kiyish bo'lib, kutilgan utility (foyda) 83 ni tashkil qiladi.

12.3. Ketma-ket qarorlar

Umuman olganda, agentlar olam haqida biror narsani kuzatmasdan qarorlar qabul qilmaydi va faqat bitta qaror qabul qilmaydi. Ko'proq odatiy vaziyatda agent kuzatishlar qiladi, qaror qabul qiladi, bu qarorni amalga oshiradi, natijadagi dunyoda kuzatishlar qiladi, so'ngra kuzatishlar asosida yana bir qaror qabul qiladi va hokazo. Keyingi harakatlar kuzatilgan narsalarga va kuzatilgan narsalar oldingi harakatlarga bog'liq bo'lishi mumkin. Bunday vaziyatda, harakatni amalga oshirishning yagona sababi kelajakdagi harakatlar uchun ma'lumot olishdir. Faqat ma'lumot olish uchun amalga oshirilgan harakatlar **ma'lumot izlash harakatlari** deb ataladi. Bunday harakatlar faqat qisman kuzatiladigan muhitlarda zarur. Formalizm ma'lumot izlash harakatlarini boshqa harakatlardan ajratish zarur emas. Odatda, harakatlar ham ma'lumot natijalariga, ham dunyoga ta'sir qiladi.

-
- 1: **procedure** *VE_SSDN(DN)*
 - 2: **Inputs**
 - 3: *DN* a single-stage decision network
 - 4: **Output**
 - 5: An optimal policy and the expected utility of that policy.
 - 6: Prune all nodes that are not ancestors of the utility node.
 - 7: Sum out all chance nodes.
 - 8: – at this stage there is a single factor *F* that was derived from utility
 - 9: Let *v* be the maximum value in *F*
 - 10: Let *d* be an assignment that gives the maximum value
 - 11: return *d, v*

12.8-rasm. Bir bosqichli qaror tarmog'i uchun o'zgaruvchini istisno qilish

Ketma-ket qaror muammosi quyidagilarni modellashtiradi: har bir bosqichda agent uchun mavjud bo'lgan harakatlar, agent harakat qilganda mavjud bo'lgan yoki bo'lishi mumkin bo'lgan ma'lumotlar, harakatlarning ta'siri va bu ta'sirlarning maqbulligi.

12.13-misol. Tashxisning oddiy holatini ko'rib chiqing, bu yerda shifokor avval ba'zi tahlillarni tanlaydi va keyin tahlil natijalarini hisobga olgan holda bemorni davolaydi. Shifokor tahlil o'tkazishga qaror qilishining sababi shundaki, keyingi bosqichda davolash amalga oshirilganda ba'zi ma'lumotlar (tahlil natijalari) mavjud bo'ladi. Tahlil natijalari davolash qaror qilinganda mavjud bo'lgan ma'lumot bo'ladi, ammo tahlil o'tkazishga qaror qilinganda emas. Tahlilning o'zi bemorga zarar yetkazishi mumkin bo'lsa ham, ko'pincha tahlil o'tkazish tavsiya etiladi.

Mavjud harakatlar mumkin bo'lgan tahlillar va mumkin bo'lgan davolash usullaridir. Tahlil o'tkazish qarori qabul qilinganda, mavjud bo'lgan ma'lumotlar bemorning simptomlari bo'ladi. Davolash qarori qabul qilinganda, mavjud bo'lgan ma'lumotlar bemorning simptomlari, qaysi tekshiruvlar amalga oshirilganligi va tahlil natijalari bo'ladi. Tahlilning ta'siri, qaysi tekshiruvlar bajarilganiga va bemorga nima bo'lganiga bog'liq. Davolashning ta'siri esa, davolash funksiyalari-ga va bemorning sog'lig'ida nima muammo borligiga bog'liq bo'ladi. Foydalilik, masalan, tahlillar va davolashning narxlarini, bemorning qisqa muddatli og'riqlari va noqulayliklarini hamda uzoq muddatli prognozni o'z ichiga olishi mumkin.

12.3.1. Qaror tarmoqlari

Qaror tarmog'i (shuningdek, **ta'sir diagrammasi** deb ham ataladi) cheklan-gan ketma-ket qaror muammosining grafik tasviridir. Qaror tarmoqlari ishonch tarmoqlarini qaror o'zgaruvchilari va foydalilik bilan kengaytiradi. Qaror tarmog'i bir bosqichli qaror tarmog'ini ketma-ket qarorlar uchun kengaytiradi va qaror tugunlarining asl to'plamlari bo'lishi uchun tasodifiy tugunlar va qaror tugunlariga ruxsat beradi.

Xususan, qaror tarmog'i tasodifiy tugunlar (oval shaklida chizilgan), qaror tugunlari (to'rtburchaklar shaklida chizilgan) va foydalilik tuguni (olmos shaklida chizilgan) bo'lgan yo'naltirilgan asiklik grafdir (DAG). Yoylarning ma'nosi quyidagicha:

Qaror tugunlariga kiruvchi yoylar qaror qabul qilinayotganda mavjud bo'lgan ma'lumotlarni ifodalaydi.

Tasodifiy tugunlarga kiruvchi yoylar ehtimoliy bog'liqlikni ifodalaydi.

Foydalilik tuguniga kiruvchi yoylar foydalilik nima bilan bog'liq ekanligini ifodalaydi.

12.14-misol. 12.9-rasmda agent tashqariga chiqqanda soyabon olishi yoki olmasligi to'g'risida qaror qabul qilish kerak bo'lgan oddiy qaror tarmog'i ko'rsa-

tilgan. Agentning foydasi ob-havo va soyabon olishiga bog'liq. Agent ob-havoni kuzatmaydi; u faqat ob-havo ma'lumotlarini kuzatadi. Ob-havo ma'lumotlari ob-havoga ehtimoliy bog'liq.

Ushbu tarmoqning bir qismi sifatida dizayner har bir tasodifiy o'zgaruvchi va har bir qaror o'zgaruvchisi uchun domenni belgilashi kerak. Tasodifiy o'zgaruvchi *Weather* (Ob-havo) domeni $\{norain, rain\}$, ($\{yomg'ir\}$ yo'q, $yomg'ir\}$), tasodifiy o'zgaruvchi *Forecast* (Ob-havo ma'lumoti) domeni $\{sunny, rainy, cloudy\}$ ($\{quyoshli, yomg'irli, bulutli\}$) va qaror o'zgaruvchisi *Umbrella* (Soyabon) domeni $\{take\ it, leave\ it\}$ ($\{olish, qoldirish\}$). Foydalilik tuguniga hech qanday domen qo'shilmaydi. Dizayner, shuningdek, tasodifiy o'zgaruvchilarning ehtimolliklarini ularning asl to'plamlari bilan birga belgilashi kerak. Tasavvur qiling, $P(Weather)$ quyidagicha belgilanadi:

$$P(Weather = rain) = 0.3$$

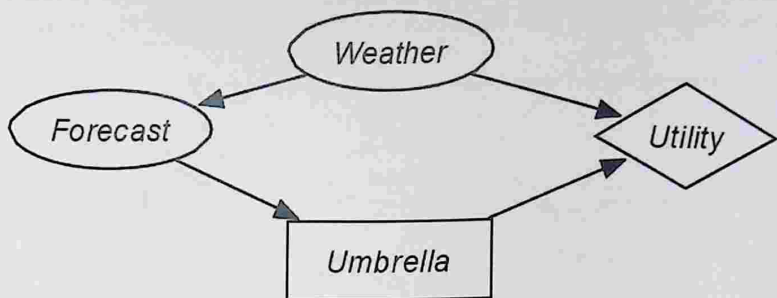
$P(Forecast|Weather)$ quyidagicha beriladi:

P(Forecast Weather) is given by		
Weather	Forecast	Probability
norain	sunny	0.7
norain	cloudy	0.2
norain	rainy	0.1
rain	sunny	0.15
rain	cloudy	0.25
rain	rainy	0.6

Tasavvur qiling, foydalilik funksiyasi, $u(Weather, Umbrella)$, quyidagicha:

Weather	Umbrella	Utility
norain	take_it	20
norain	leave_it	100
rain	take_it	70
rain	leave_it	0

Umbrella qaror o'zgaruvchisi uchun hech qanday jadval belgilanmagan. Rejalashtiruvchining vazifasi, ob-havo ma'lumoti funksiyasi sifatida *Umbrella* uchun qaysi qiymatni tanlashni aniqlashdir.



12.9-rasm. Soyabon olish to'g'risida qaror qabul qilish uchun qaror qabul qilish tarmog'i

12.15-misol. 12.10-rasmda 12.13-misoldagi idealizatsiya qilingan tashxis sse-nariysini ifodalovchi qaror tarmog'i ko'rsatilgan. Simptomlar kasallikka bog'liq. Qaysi tahlilni o'tkazish simptomlarga asoslanadi. Tahlil natijasi kasallik va ba-jarilgan tekshiruvga bog'liq. Davolash qarori simptomlar, o'tkazilgan tekshiruv va tahlil natijasiga asoslanadi. Natija kasallik va davolashga bog'liq. Foyda tahlillarning narxi va natijaga bog'liq.

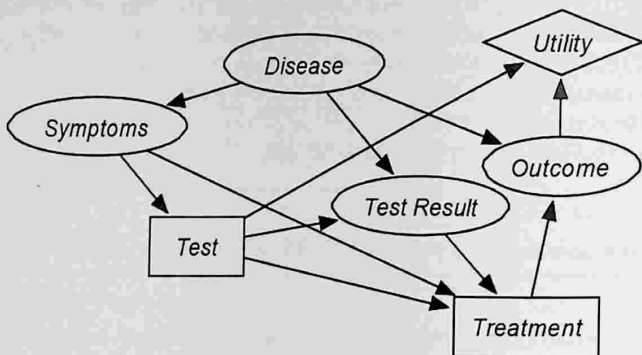
Natija tahlilga bog'liq emas, faqat kasallik va davolashga bog'liq, shuning uchun tahlilning yon ta'siri yo'q. Davolash foydaga to'g'ridan to'g'ri ta'sir qil-maydi; davolashning har qanday narxi natijaga kiritilishi mumkin. Agar barcha tahlillar bir xil narxga ega bo'lmasa, foyda tahlilga bog'liq bo'lishi kerak.

Tahlillar va davolashlar to'g'risida qaror qabul qiladigan diagnostik yordam-chi agar tahlil natijasi qat'iy bo'lmasa, bemorda qanday kasallik borligini hech qachon aniqlamay olmaydi.

12.16-misol. 12.11-rasmda ishonch tarmog'ining kengaytmasi bo'lgan qa-ror tarmog'i ko'rsatilgan. Agent binodan odamlarning chiqib ketishi haqida hi-sobot olishi mumkin va o't o'chirish xizmatiga qo'ng'iroq qilish yoki qilmaslik to'g'risida qaror qabul qilishi kerak. Qo'ng'iroq qilishdan oldin agent tutunni tekshirishi mumkin, lekin bu bilan bog'liq xarajatlar mavjud. Foydalilik agent qo'ng'iroq qilganmi, yong'in bormi va tutunni tekshirish bilan bog'liq xarajat-larga bog'liq.

Ushbu ketma-ket qaror muammosida ikkita qaror qabul qilinishi kerak. Bi-rinchidan, agent tutunni tekshirish yoki tekshirmaslikka qaror qilishi kerak. Ushbu qaror qabul qilinganida mavjud bo'lgan ma'lumotlar binodan odamlarning chiqib ketishi haqida hisobot mavjudmi yoki yo'qligini aniqlaydi. Ikkinchidan, agent o't

o'chirish xizmatiga qo'ng'iroq qilish yoki qilmaslikka qaror qilishi kerak. Ushbu qarorni qabul qilishda, agent hisobot bo'lganmi, tutunni tekshirganmi va tutunni ko'ra oladimi, yo'qmi bilib oladi. Hamma o'zgaruvchilar ikkilik deb taxmin qiling.



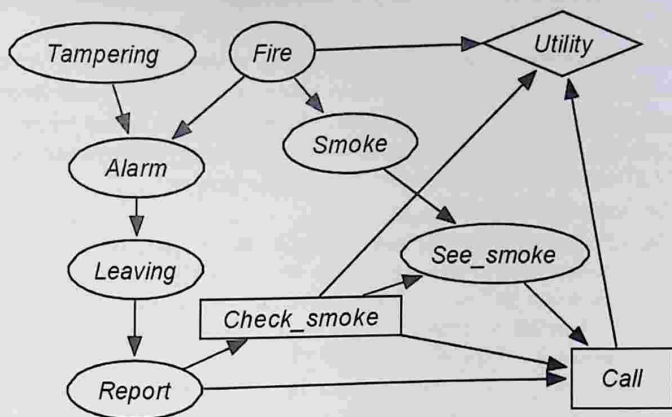
12.10-rasm. «Tahlil qilish – davolash» ideal diagnostika ssenariysi uchun qaror qabul qilish tarmog'i

Qaror tarmog'i uchun zarur bo'lgan ma'lumotlar ishonch tarmog'ining shartli ehtimolliklarini va

$P(\text{See smoke} \mid \text{Smoke}, \text{Check smoke})$ – agent tutunni izlashiga va tutun bor-yo'qligiga qarab tutunni ko'rish ehtimoli. Agentning tutun uchun mukammal sensori bor deb faraz qiling. U tutunni faqatgina tutunni izlaganda va tutun bo'lganda ko'radi. 12.9-mashqqa qarang.

$u(\text{Check smoke}, \text{Fire}, \text{Call})$ – foydalilik agent tutunni tekshirishi, yong'in borligi va o't o'chirish xizmatiga qo'ng'iroq qilishiga qarab qanday bog'liq ekanligini ifodalaydi. 12.12-rasmda bu foydalilik ma'lumotlari keltirilgan. Ushbu foydalilik funksiyasi qo'ng'iroq qilishning 200 qiymatida, tekshirishning 20 qiymatida, ammo yong'in bo'lganda qo'ng'iroq qilmaslik 5000 qiymatida ekanligini ifodalovchi xarajat tuzilmasini ifodalaydi. Foyda xarajatning manfiy qiymatidir

Unutmaydigan agent – bu qarorlari vaqt bo'yicha to'liq tartiblangan agent bo'lib, agent o'zining avvalgi qarorlarini va avvalgi qarorga mavjud bo'lgan har qanday ma'lumotni eslaydi.



12.11-rasm. Yong'in signalizatsiyasi muammosini hal qilish uchun qaror qabul qilish tarmog'i

Check smoke	Fire	Call	Utility
yes	true	yes	-220
yes	true	no	-5020
yes	false	yes	-220
yes	false	no	-20
no	true	yes	-200
no	true	no	-5000
no	false	yes	-200
no	false	no	0

12.12-rasm. Yong'in signalizatsiyasi to'g'risida qaror qabul qilish tarmog'i uchun foydalilik

Unutmaydigan qaror tarmog'i bu qaror tugunlari to'liq tartiblangan qaror tarmog'i bo'lib, agar D_i qaror tuguni umumiy tartibda D_j dan oldin kelsa, unda D_i D_j ning to'plamlari hisoblanadi va D_i ning har qanday asl to'plamlari ham D_j ning asl to'plamlari hisoblanadi. Shunday qilib, D_i uchun mavjud bo'lgan har qanday ma'lumot keyingi har qanday qaror uchun ham mavjud bo'ladi va D_i uchun tanlangan harakat keyingi qarorlar uchun mavjud ma'lumotlarning bir qismidir. Unutmaslik sharti quyidagi ta'riflar mantiqan to'g'ri va quyidagi algoritmlar ishlashini ta'minlash uchun yetarli.

12.3.2. Tamoyillar

Tamoyil agentning har qanday favqulodda vaziyatda nima qilishini belgilaydi. Tamoyil har bir qaror o'zgaruvchisi uchun qaror funksiyasidan iborat. Qaror o'zgaruvchisi uchun **qaror funksiyasi** – bu qaror o'zgaruvchisi uchun uning asl to'plamlariga har bir qiymat tayinlash uchun qiymatini belgilaydigan funksiya.

12.17-misol. 12.14-misolda ba'zi tamoyillar berilgan:

Har doim soyabon olish.

Faqat ob-havo ma'lumoti «yomg'irli» bo'lsa, soyabon olish.

Faqat ob-havo ma'lumoti «quyoshli» bo'lsa, soyabon olish.

Uchta mumkin bo'lgan ob-havo ma'lumoti borligi sababli, har bir ob-havo ma'lumoti uchun ikki xil tanlov mavjud bo'lib, jami sakkizta turli tamoyil mavjud.

12.18-misol. 12.16-misolda tamoyil *Check smoke* (Tutunni tekshirish) va *Call* (Qo'ng'iroq qilish) qarorlari uchun qaror funksiyasini belgilaydi. Ba'zi tamoyillar:

Hech qachon tutunni tekshirmaslik va faqat hisobot bo'lsa qo'ng'iroq qilish.

Har doim tutunni tekshirish va faqat tutunni ko'rsa qo'ng'iroq qilish.

Hisobot bo'lsa tutunni tekshirish va faqat hisobot bo'lsa va tutunni ko'rsa qo'ng'iroq qilish.

Hisobot bo'lmasa tutunni tekshirish va tutunni ko'rmasa qo'ng'iroq qilish.

Har doim tutunni tekshirish va hech qachon qo'ng'iroq qilmaslik.

Check smoke (Tutunni tekshirish) uchun to'rtta qaror funksiyasi mavjud. *Call* (Qo'ng'iroq qilish) uchun 2^8 qaror funksiyasi mavjud; *Call* asl to'plamlarining qiymatlari uchun sakkizta topshiriqning har biri uchun agent qo'ng'iroq qilish yoki qilmaslikni tanlashi mumkin. Shunday qilib, $4 \cdot 2^8 = 1024$ turli tamoyil mavjud.

Tamoyilning kutilgan foydasi

Har bir tamoyil ushbu tamoyilga amal qiladigan agent uchun kutilgan foydaga ega. Ratsional agent kutilgan foydasini maksimal darajada oshiradigan tamoyilni qabul qilishi kerak.

Mumkin bo'lgan olam har bir tasodifiy o'zgaruvchi va har bir qaror o'zgaruvchisi uchun qiymatni belgilaydi. Agar har bir qaror o'zgaruvchisi D bo'lsa, $D(\omega)$ mumkin bo'lgan dunyoda D ning parents (asl to'plamlari)ning qiymatlari berilgan tamoyil tomonidan belgilangan qiymatga ega bo'lsa, mumkin bo'lgan dunyo ω tamoyil π ni qondiradi.

Mumkin bo'lgan dunyo to'liq tarixga mos keladi va barcha kuzatilgan o'zgaruvchilarni o'z ichiga olgan barcha tasodifiy va qaror o'zgaruvchilarining qiymatlarini belgilaydi. Mumkin bo'lgan dunyo ω tamoyil π ni qondiradi, agar ω agent tamoyil π ga amal qiladigan tarixning mumkin bo'lgan bir qismi bo'lsa.

π tamoyilning **kutilgan foydasi** quyidagicha ifodalanadi:

$$E(u|\pi) = \sum u(\omega) * P(\omega)$$

ω satisfies π

bu yerda $P(\omega)$, ω dunyosining ehtimolligi, ω da asl to'plamlarining qiymatlari berilgan tasodifiy tugunlarning qiymatlarining ehtimolliklarining ko'paytmasi va $u(\omega)$ dunyodagi u foydasining qiymatidir.

12.19-misol. 12.14-misolda π tamoyili, agar ob-havo ma'lumoti bulutli bo'lsa soyabon olish va aks holda uyda qoldirish tamoyili bo'lsin. Ushbu tamoyilni qondiradigan dunyolar:

<i>Weather</i>	<i>Forecast</i>	<i>Umbrella</i>
<i>norain</i>	<i>sunny</i>	<i>leave_it</i>
<i>norain</i>	<i>cloudy</i>	<i>take_it</i>
<i>norain</i>	<i>rainy</i>	<i>leave_it</i>
<i>rain</i>	<i>sunny</i>	<i>leave_it</i>
<i>rain</i>	<i>cloudy</i>	<i>take_it</i>
<i>rain</i>	<i>rainy</i>	<i>leave_it</i>

Qaror o'zgaruvchisi uchun qiymat tamoyil tomonidan tanlangan qiymat ekanligiga e'tibor bering. Bu faqat ob-havo ma'lumotiga bog'liq.

Ushbu tamoyilning kutilgan foydasi ushbu tamoyilni qondiradigan dunyolar-ning o'rtacha foydasini olish orqali hisoblanadi:

$$\begin{aligned}
 E(u | \pi_1) = & P(norain) * P(sunny | norain) * u(norain, leave_it) \\
 & + P(norain) * P(cloudy | norain) * u(norain, take_it) \\
 & + P(norain) * P(rainy | norain) * u(norain, leave_it) \\
 & + P(rain) * P(sunny | rain) * u(rain, leave_it) \\
 & + P(rain) * P(cloudy | rain) * u(rain, take_it) \\
 & + P(rain) * P(rainy | rain) * u(rain, leave_it)
 \end{aligned}$$

bu yerda *norain* (yomg'ir yo'q) degani *Weather* = *norain*, *sunny*, (quyoshli) degani *Forecast* = *sunny* va boshqa qiymatlar uchun ham shunday.

Optimal tamoyil bu π^* tamoyil bo'lib, $E(u|\pi^*) \geq E(u|\pi)$ barcha π tamoyillar uchun. Ya'ni optimal tamoyil bu kutilgan foydasi barcha tamoyillar bo'yicha maksimal bo'lgan tamoyil.

Agar ikkilik qaror tugunida n ikkilik asl to'plamlari bo'lsa, asl to'plamlariga qiymatlar tayinlash uchun 2^n xil variant va natijada bu qaror tuguni uchun 2^{2^n} xil qaror funksiyalari mavjud. Tamoyillar soni har bir qaror o'zgaruvchisi uchun qa-

ror funksiyalari sonining ko'paytmasidir. Hatto kichik misollarda ham tamoyillar soni juda katta bo'lishi mumkin. Eng yaxshisini topish uchun oddiygina tamoyillarni sanab o'tadigan algoritim juda samarasiz bo'ladi.

12.3.3. Qidiruv yordamida qaror tarmoqlarini optimallashtirish

Ishonch tarmoqlari uchun rekursiv shartlash algoritmi qaror tarmoqlariga quyidagicha kengaytirilishi mumkin:

Bu kontekstni, tasodifiy o'zgaruvchilarning shartli taqsimotlari va foyda uchun omillar to'plamini va qaror o'zgaruvchilari to'plamini oladi.

Bu kontekst va optimal qaror funksiyalari uchun berilgan optimal tamoyil qiymatini qaytaradi. Qaror funksiyalari (*context*, $d = v$) juftliklar to'plami sifatida ifodalanadi, bu esa kontekstda qaror o'zgaruvchisi d ning optimal qarori v qiymatini olishini anglatadi.

Ajratish tartibi har bir qaror tugunining asl to'plamlari tugundan oldin kelishini va ular faqat qaror tugunidan oldin keladigan tugunlar ekanligini talab qiladi. Xususan, bu barcha qolgan qaror tugunlarining asl to'plamlari bo'lmagan o'zgaruvchini tanlay olmaydi. Shunday qilib, bu faqat unutmaydigan qaror tarmoqlariga tegishli. Agar qaror o'zgaruvchisi d ning asl to'plamlari bo'lmagan X o'zgaruvchisini ajratmoqchi bo'lsa, agent X ning qiymatini bilmagan holda d uchun qiymatni tanlashi mumkin emas.

Foyda tugunini ehtimolliklarni aniqlovchi omillardan farqli ravishda ko'rib chiqish kerak emas; foyda faqat ehtimolliklar bilan ko'paytiriladigan qiymat beradi.

Qaror tugunida ajratish uchun algoritim har bir tayinlash uchun qaytarilgan qiymatlarni maksimal darajada oshiradigan qiymatni tanlaydi.

12.13-rasm 9.9-rasmdagi sodda qidiruv algoritmini qaror tarmoqlarini hal qilish uchun kengaytiradi. Bu algoritimni mahsulotlar yig'indisini hisoblaydigan, qaror tugunlarida maksimal qiymatni tanlaydigan algoritim sifatida o'ylash osonroq. Rekursiv chaqiruv tomonidan qaytarilgan qiymat ehtimollik va foydaning aralashmasi hisoblanadi.

12.20-misol. 12.16-misolda algoritim avval qaror tugunlarining barcha asl to'plamlari bo'lgan *Report* (Hisobot) tugunida ajratadi. *False* tarmog'ini ko'rib chiqaylik. Keyin DN_dfs ni $\{Report=false\}$ konteksti bilan chaqiradi va qaror o'zgaruvchisini *Check smoke* (Tutunni tekshirish) ajratiladi. Rekursiv chaqiruv, hisobot yolg'on bo'lganda tekshirish yaxshimi yoki yo'qligini aniqlaydi. Keyin *See smoke* (Tutunni ko'rish) va *Call* (Qo'ng'iroq qilish) ga ajratiladi. Qolgan omillar har qanday tartibda istisno qilinishi mumkin.

DN_dfs ni $\{Report=false\}$ konteksti bilan chaqirishi

$$\langle P(Report = false) * utility(\pi), \pi \rangle$$

juftligini qaytaradi, bu yerda π $Report=false$ bo'lganda optimal tamoyildir.

12.1-jadval bir o'zgaruvchi tartibi uchun qanday omillar baholanishi mumkinligini ko'rsatadi. Rekursiv chaqiruvlar tomonidan qaytarilgan raqamlarni tushunish qiyin. Ba'zilar faqat ehtimollik mahsulotlari (foydalanish omili baholanganidan keyingi rekursiv chaqiruvlar), ba'zilar esa ehtimollik va foydalanish aralashmasidir. Bu haqda o'ylashning eng oson yo'li, algoritim qaror tugunlarida tanlov uchun maksimal bo'lgan raqamlarning mahsulotining yig'indisini hisoblayotganidir.

12.13-rasm algoritmi grafik tuzilmaning mustaqilligidan foydalanmaydi, lekin rekursiv shartlashning barcha yaxshilanishlari, ya'ni o'chirilgan komponentlarini tan olish va ehtiyotkor keshlarni o'zgartirmasdan qo'llashi mumkin.

12.3.4. Qaror tarmoqlari uchun o'zgaruvchini istisno qilish

O'zgaruvchini istisno qilishni optimal tamoyilni topishga moslashtirilishi mumkin.

```

1: procedure DN dfs (Con, Fs, Ds)
2: # Con: contest, Fs: set of factors, Ds: set of decision variables
3: # Returns a value, and set of context, d = v pairs
4: if Fs = {} then
5:   return (1, {})
6: else if f ∈ Fs can be evaluated in Con then
7:   (v, p) := DN dfs(Con, Fs \ {f}, Ds)
8:   return (eval(f, Con) * v, p)
9: else if Con assigns all parents of decision node d ∈ Ds then
10:  max value := -∞; opt_decn := ⊥
11:  for val in domain(d) do
12:    (v, p) := DN dfs(Con ∪ {d = val}, Fs, Ds \ {d})
13:    if v > max value then
14:      max_value := v; opt_decn := {(Con, d = val)} ∪ p
15:  return (max_value, opt_decn)
16: else
17:  select variable var in vars(Fs) \ vars(Con) that is a parent of all d ∈ Ds
18:  sum := 0; policy := {}
19:  for val in domain(var) do
20:    (v, p) := DN dfs(Con ∪ {var = val}, Fs, Ds)
21:    sum := sum + v; policy := policy ∪ p
22:  return (sum, policy)

```

12.13-rasm. Qaror tarmoqlari uchun qidiruvga asoslangan optimallashtirish algoritmi

Birinchi navbatda, *oxirgi* qarorni ko'rib chiqish, asl to'plamlarining har bir qiymati uchun optimal qarorni topish va ushbu maksimal qiymatlarning omilini ishlab chiqarish kerak. Bu rekursiv ravishda hal qilinishi mumkin bo'lgan bitta kamroq qaror bilan yangi qarorlar tarmog'iga olib keladi.

12.14-rasm qaror tarmoqlari uchun o'zgaruvchini istisno qilishni qanday ishlatishni ko'rsatadi. Asosan, bu optimal qarorning kutilgan foydasini hisoblaydi. Bu tasodifiy o'zgaruvchilarni, qaror tugunining asl to'plamlari bo'lmaganlarni, ularni ba'zi bir istisno qilish tartibida yig'ish orqali istisno qiladi. Tasodifiy o'zgaruvchilarni istisno qilish tartibi to'g'rilikka ta'sir qilmaydi, shuning uchun samadorlik uchun tanlanishi mumkin.

Qaror tugunining asl to'plamlari bo'lmagan barcha tasodifiy o'zgaruvchilarni yo'q qilgandan so'ng, unutmaydigan qaror tarmog'ida bitta qaror o'zgaruvchisi D qoldirilishi kerak, u F omilida bo'ladi, bu yerda F dagi D dan tashqari barcha o'zgaruvchilar D ning asl to'plamlari hisoblanadi. Bu qaror D qarorlar tartibida oxirgi qaror hisoblanadi. Ushbu qaror tugunini yo'q qilish uchun, VE_DN qaror qiymatlarini tanlaydi va bu qiymatlar maksimal foyda keltiradi. Ushbu maksimalashtirish qolgan o'zgaruvchilar uchun yangi omil va istisno qilinayotgan qaror o'zgaruvchisi uchun qaror funksiyasini yaratadi. Maksimallashtirish orqali yaratilgan bu qaror funksiyasi optimal tamoyildagi qaror funksiyalaridan biridir.

12.21-misol. 12.14-misolda uchta boshlang'ich omil mavjud: $P(Weather)$, $P(Forecast | Weather)$ va $u(Weather, Umbrella)$. Birinchi navbatda, 12.2-jadvalning chap tomonida ko'rsatilgan *Forecast* va *Umbrella* koeffitsiyentini hosil qilish uchun barcha uch omilni ko'paytirish va *Weather* yig'ish orqali *Weather* ni istisno qiladi. *Umbrella* ni maksimallashtirish uchun, *Forecast*ning har bir qiymati uchun, VE_DN *Umbrellaning* qiymatini tanlaydi va bu qiymat omilning qiymatini maksimal qiladi. Masalan, ob-havo *quyoshli* bo'lganda, agent soyabonni uyda qoldirishi kerak, bu esa 49.0 qiymatni beradi. Hosil bo'lgan qaror funksiyasi 12.2-jadvalning markazida ko'rsatilgan va hosil bo'lgan omil 12.2-jadvalning o'ng tomonida ko'rsatilgan.

Endi u *Forecast* ni ushbu nisbat asosida jamlaydi, natijada 77,0 qiymatiga erishiladi. Bu optimal tamoyilning kutilgan qiymatidir.

Variable	Factor(s) Evaluated
<i>Report</i>	-
<i>Check_smoke</i>	-
<i>See_Smoke</i>	-
<i>Call</i>	-
<i>Tampering</i>	$P(Tampering)$
<i>Fire</i>	$P(Fire), Utility(Fire, See\ smoke, Call)$
<i>Smoke</i>	$P(Fire Smoke), P(See\ smoke Smoke)$
<i>Alarm</i>	$P(Alarm Tampering, Fire)$
<i>Leaving</i>	$P(Leaving Alarm), P(Report Leaving)$

12.20-misolda bo'linadigan o'zgaruvchilar va baholangan omillar ko'rsatilgan.

12.22-misol. 12.16-misolga qarang. Dastlabki omillar 12.3-jadvalda ko'rsatilgan. Kutilayotgan foyda, agar mos harakatlar tanlangan bo'lsa, ehtimollik va foyda mahsulotidir.

12.3-jadval (pastki qismida) bitta istisno qilish tartibi uchun olib tashlangan va yaratilgan omillarni ko'rsatadi. Qaror qabul qilish tugunining asl to'plamlari bo'lmagan tasodifiy o'zgaruvchilar birinchi navbatda umumlashtiriladi.

$f16$ omili optimal tamoyilning kutilayotgan foydasini ifodalaydigan raqamdir. Chiqarib tashlagandan so'ng, bitta $f12$ omili qoladi, uning bir qismi (ikki kasrgacha) quyidagicha:

- 1: **procedure** $VE\ DN(DN)$;
- 2: **Inputs**
- 3: DN a decision network
- 4: **Output**
- 5: An optimal policy and its expected utility
- 6: **Local**
- 7: DFs : a set of decision functions, initially empty
- 8: Fs : a set of factors
- 9: Remove all variables that are not ancestors of the utility node
- 10: Create a factor in Fs for each conditional probability
- 11: Create a factor in Fs for the utility
- 12: **while** there are decision nodes remaining **do**
- 13: Sum out each random variable that is not a parent of a decision node
- 14: Let D be the last decision remaining
- 15: # D is only in a factor $F(D, V_1, \dots, V_k)$ where $V_1, \dots, V_k$ are parents of D
- 16: Add $\max_D F$ to Fs .
- 17: Add $\arg \max_D F$ to DFs .
- 18: Sum out all remaining random variables
- 19: Return DFs and the product of remaining factors

12.14-rasm. Qaror qabul qiluvchi tarmoqlar uchun o'zgaruvchilarni istisno qilish

12.2-jadval

12.21-misolda yaratilgan omillar va qaror qabul qilish funksiyalari

Forecast	Umbrella	Value	Forecast	Umbrella	Forecast	Value
sunny	take_it	12.95	sunny	leave_it	sunny	49.0
sunny	leave_it	49.0	cloudy	leave_it	cloudy	14.0
cloudy	take_it	8.05	rainy	take_it	rainy	14.0
cloudy	leave_it	14.0				
rainy	take_it	14.0				
rainy	leave_it	7.0				

<i>Report</i>	<i>See_smoke</i>	<i>Check_smoke</i>	<i>Call</i>	Value
<i>true</i>	<i>true</i>	<i>yes</i>	<i>yes</i>	-1.33
<i>true</i>	<i>true</i>	<i>yes</i>	<i>no</i>	-29.30
<i>true</i>	<i>true</i>	<i>no</i>	<i>yes</i>	0
<i>true</i>	<i>true</i>	<i>no</i>	<i>no</i>	0
<i>true</i>	<i>false</i>	<i>yes</i>	<i>yes</i>	-4.86
<i>true</i>	<i>false</i>	<i>yes</i>	<i>no</i>	-3.68

Ushbu omildan *Call* uchun optimal qaror funksiyasini yaratish mumkin, *Report*, *see_smoke* va *Check_smoke* uchun har bir topshiriqning qiymatini maksimal darajada oshiradigan *Call* qiymatini tanlash orqali.

Report=true, *See smoke=true* va *Check smoke=yes* bo'lgan holatni ko'rib chiqing. -1.33 va -29.3ning maksimumi -1.33 bo'ladi, shuning uchun bu holatda optimal harakat *Call=yes* va qiymati -1.33 bo'ladi. Bu maksimallashtirish *Report*, *See smoke* va *Check smokening* boshqa qiymatlari uchun takrorlanadi.

Call uchun optimal qaror funksiyasi quyidagicha:

Initial factors:

Meaning	Factor
<i>P(Tampering)</i>	$f_0(\text{Tampering})$
<i>P(Fire)</i>	$f_1(\text{Fire})$
<i>P(Alarm Tampering, Fire)</i>	$f_2(\text{Tampering, Fire, Alarm})$
<i>P(Smoke Fire)</i>	$f_3(\text{Fire, Smoke})$
<i>P(Leaving Alarm)</i>	$f_4(\text{Alarm, Leaving})$
<i>P(Report Leaving)</i>	$f_5(\text{Leaving, Report})$
<i>P(See smoke Check smoke, Smoke)</i>	$f_6(\text{Smoke, See smoke, Check smoke})$
<i>u(Fire, Check smoke, Call)</i>	$f_7(\text{Fire, Check smoke, Call})$

Bir istisno qilish tartibida olib tashlanadigan va yaratiladigan omillar:

12.3-jadval

12.22-misoldagi asl va yaratilgan omillar

Variable	How	Removed	Factor Created
<i>Tampering</i>	sum	f_0, f_2	$f_8(\text{Fire, Alarm})$
<i>Fire</i>	sum	f_1, f_8, f_3, f_7	$f_9(\text{Alarm, Smoke, Check_Smoke, Call})$
<i>Alarm</i>	sum	f_4, f_9	$f_{10}(\text{Smoke, Check Smoke, Call, Leaving})$
<i>Smoke</i>	sum	f_{10}, f_6	$f_{11}(\text{Check Smoke, Leaving, See Smoke, Call})$
<i>Leaving</i>	sum	f_5, f_{11}	$f_{12}(\text{Report, See Smoke, Check Smoke, Call})$
<i>Call</i>	max	f_{12}	$f_{13}(\text{Check Smoke, See Smoke, Report})$
<i>See_smoke</i>	sum	f_{13}	$f_{14}(\text{Check Smoke, Report})$
<i>Check_smoke</i>	max	f_{14}	$f_{15}(\text{Report})$
<i>Report</i>	sum	f_{15}	f_{16}

<i>Report</i>	<i>See_smoke</i>	<i>Check_smoke</i>	<i>Call</i>
<i>true</i>	<i>true</i>	<i>yes</i>	<i>yes</i>
<i>true</i>	<i>true</i>	<i>no</i>	<i>yes</i>
<i>true</i>	<i>false</i>	<i>yes</i>	<i>no</i>
...	...	...	...

Report=true, *See smoke=true* va *Check smoke=no* bo'lganda *Call* uchun qiymat ixtiyoriy. Agent bu holatda nima qilishni rejalashtirishi ahamiyatga ega emas, chunki bu holat hech qachon yuz bermaydi. Algoritm buni maxsus holat sifatida ko'rib chiqishi shart emas.

*Call*ni maksimallashtirishdan kelib chiqqan omil *Report*, *sees smoke* va *Check smokening* har bir kombinatsiyasi uchun maksimal qiymatlarni o'z ichiga oladi:

<i>Report</i>	<i>See_smoke</i>	<i>Check_smoke</i>	<i>Value</i>
<i>true</i>	<i>true</i>	<i>yes</i>	- 1.33
<i>true</i>	<i>true</i>	<i>no</i>	0
<i>true</i>	<i>false</i>	<i>yes</i>	- 3.68

See smoke ni umumlashtirish quyidagi omilni beradi:

<i>Report</i>	<i>Check_smoke</i>	<i>Value</i>
<i>true</i>	<i>yes</i>	-5.01
<i>true</i>	<i>no</i>	-5.65
<i>false</i>	<i>yes</i>	-23.77
<i>false</i>	<i>no</i>	-17.58

*Report*ning har bir qiymati uchun *Check smoke* ni maksimallashtirish qaror funksiyasini beradi:

<i>Report</i>	<i>Check smoke</i>
<i>true</i>	<i>yes</i>
<i>false</i>	<i>no</i>

va omil

<i>Report</i>	<i>Value</i>
<i>true</i>	-5.01
<i>false</i>	-17.58

Report umumlashtirish kutilayotgan foydani -22.60 (yaxlitlash xatolarini hisobga olgan holda) beradi.

Shunday qilib, qaytarilgan tamoyil qoidalar sifatida ko'rilishi mumkin:

$check_smoke \leftarrow report.$

$call \leftarrow see_smoke.$

$call \leftarrow report \wedge \neg check_smoke \wedge \neg see_smoke.$

Ushbu qoidalar orasida so'ngisi hech qachon ishlatilmaydi, chunki optimal tamoyilga amal qiladigan agent *report* bo'lsa, *smoke* ni tekshiradi. U *VE_DN*, *Call* ni optimallashtirganda *Check_smoke* uchun optimal tamoyilni aniqlamaganligi sababli tamoyilda qoladi.

Shuni ham yodda tutingki, bu holda tutunni tekshirish darhol salbiy natijaga olib kelsa ham, tutunni tekshirish mantiqan to'g'ri keladi, chunki olingan ma'lumotlar qimmatlidir.

Quyidagi misol VE algoritmi qarorni optimallashtirganda qaror o'zgaruvchisini o'z ichiga olgan omil asl to'plamlarining kichik to'plamini o'z ichiga olishi mumkinligini ko'rsatadi.

12.23-misol. 12.14-misolni ko'rib chiqing, ammo *Weather* dan *Umbrella* ga qo'shimcha yoy bilan. Ya'ni agent ham ob-havo, ham ob-havo ma'lumotini kuzatadi. Bu holatda, umumlashtiriladigan tasodifiy o'zgaruvchilar yo'q va qaror tuguni va asl to'plamlarining kichik to'plamini o'z ichiga olgan omil asl foyda omilidir. Keyin qaror funksiyasi va omilni berib, *Umbrella* ni maksimallashtirishi mumkin:

<i>Weather</i>	<i>Umbrella</i>	<i>Weather</i>	<i>Value</i>
<i>norain</i>	<i>leave_it</i>	<i>norain</i>	100
<i>rain</i>	<i>take_it</i>	<i>rain</i>	70

E'tibor bering, ob-havo ma'lumoti qarorga tegishli emas. Ob-havo ma'lumotini bilish agentga hech qanday foydali ma'lumot bermaydi. *Forecast* (Ob-havo ma'lumoti)ni umumlashtirish barcha qiymatlar 1 bo'lgan omilni beradi.

Weather ni umumlashtirish, bu yerda $P(Weather=norain) = 0.7$, kutilayotgan foydani beradi: $0.7 \cdot 100 + 0.3 \cdot 70 = 91$

12.14-rasmdagi qaror tarmoqlari uchun o'zgaruvchilarni istisno qilishning murakkabligi 12.13-rasmdagi chuqurlik bo'yicha birinchi qidiruv bilan rekursiv shartlashning kombinatsiyasi bilan bir xil. Ular bir xil hisob-darsliklarni turli tartibda bajaradilar. Xususan, rekursiv shartlashning keshida saqlangan qiymatlar o'zgaruvchilarni istisno qilish omillaridagi qiymatlar bilan bir xil. Rekursiv shartlash shartli ehtimollikning turli ko'rinishlaridan foydalanishga imkon beradi, lekin

qiymatlarni keshda saqlash tabulyar tasvirdan foydalanishga qaraganda kamroq samarali. Quyida ko'rsatilgandek, tarmoqlar cheksiz bo'lganda, murakkab alqoritmlar odatda biri yoki boshqasiga asoslanadi.

12.4. Ma'lumot va boshqaruvning qiymati

Agar ma'lumot agentlarga yaxshiroq qarorlar qabul qilishda yordam bersa, u qiymatli bo'lishi mumkin.

12.24-misol. 12.22-misolda, *Check smoke* harakati yong'in haqida ma'lumot beradi. *Smoke* ni tekshirish 20 birlik turadi va hech qanday bevosita rag'bat bermaydi; ammo, optimal tamoyilda, *report* mavjud bo'lganda *smoke* ni tekshirish foydalidir, chunki agent o'zining keyingi harakatlarini olingan ma'lumotlarga asoslanib shartlashi mumkin. Shunday qilib, *smoke* haqida ma'lumot agent uchun qimmatli, hatto *smoke* faqat yong'in borligi haqida mukammal bo'lmagan ma'lumot bersa ham.

Ushbu misoldan muhim xulosalardan biri shundaki, ma'lumot qidirish harakati, masalan, *Check smoke*, *Call* kabi boshqa harakatlar bilan bir xil tarzda ko'rib chiqilishi mumkin. Optimal tamoyil, agar keyingi harakatlar harakatning ba'zi ta'sirini keltirib chiqarishi mumkin bo'lsa, ko'pincha faqat ma'lumot olish maqsadiga ega bo'lgan harakatlarni o'z ichiga oladi.

Agar X tasodifiy o'zgaruvchi va D qaror o'zgaruvchisi bo'lsa, D qaror uchun X haqida ma'lumotning qiymati D qaror qabul qilinganda X qiymatini bilish orqali qanchalik qo'shimcha foyda olinishi mumkinligidir. Bu har bir qaror uchun nima boshqarilayotgani va nimani kuzatilayotgani, qaror tarmog'ida berilgan ma'lumotga bog'liq.

Unutmaydigan qaror tarmog'ida N uchun qaror D bo'yicha X haqida ma'lumotning qiymati:

X dan D ga yoy qo'shilgan holda va X dan D dan keyingi qarorlarga yoylar qo'shilgan holda tarmoq unutmaydigan qaror tarmog'i bo'lib qolishini ta'minlash uchun

D , X haqida ma'lumotga ega bo'lmagan va unutmaydigan yoylar qo'shilmagan holdagi N qaror tarmog'ining qiymatidan chiqarib tashlangan holda.

Bu faqat X , D ning vorisi bo'lmagan hollarda aniqlanadi, chunki bu siklga olib keladi. (X dan D ga yoy qo'shish siklga sabab bo'lganda yanada murakkab harakat bajarilishi kerak.)

12.25-misol. 12.14-misolda, yaxshiroq ob-havo ma'lumoti olish bo'lish qanchaga tushishi mumkinligini ko'rib chiqing. *Umbrella* olish qarori bo'yicha ob-havo haqida mukammal ma'lumot olish qiymati, 12.23-misolda hisoblanganidek, *Weather* dan *Umbrella* ga yoy qo'shilgan tarmoqning qiymati 91 va asl tarmoqning qiymati 77 o'rtasidagi farqdir. Shunday qilib, *Weather* haqida *Umbrella* qarori uchun ma'lumotning qiymati $91 - 77 = 14$.

Ma'lumotning qiymati ba'zi qiziqarli xususiyatlarga ega:

Ma'lumotning qiymati hech qachon salbiy emas. Eng yomon holatda agent ma'lumotni e'tiborsiz qoldirishi mumkin.

Agar X optimal qarorning qaysi qiymati kuzatilganidan qat'i nazar, bir xil ishni qilish bo'lsa, X ning ma'lumot qiymati nolga teng. Agar X ning ma'lumot qiymati nol bo'lsa, X ning qiymatiga bog'liq bo'lmagan optimal tamoyil mavjud (ya'ni X ning qaysi qiymati kuzatilganidan qat'i nazar, bir xil harakatni tanlash mumkin).

Ma'lumotning qiymati agent ma'lumot uchun (foydaning yo'qolishi jihatidan) to'lashga tayyor bo'lishi kerak bo'lgan miqdorga chegara hisoblanadi. Bu D qarorda X ning qiymati haqidagi mukammal bo'lmagan ma'lumot qanchalik qimmatli bo'lishi mumkinligiga yuqori chegara hisoblanadi. **Mukammal bo'lmagan** ma'lumot bu X ning shovqinli datchigidan olingan ma'lumotdir. X ning qiymati uchun D qarori bo'yicha ma'lumotning qiymatidan ko'proq to'lash kerak emas. X sensori, X ma'lumotidan foydalanishi mumkin bo'lgan eng erta qaror qabul qilish uchun X ma'lumotining qiymatidan ko'proq pul to'lash uchun arzimaydi.

12.26-misol. 12.22-misoldagi yong'in signalizatsiyasi muammosida agent nosozlikni aniqlashga urinish qanchalik foydali ekanligini bilishni xohlashi mumkin. Nosozlik sensori qancha qiymatga ega bo'lishi mumkinligini aniqlash uchun nosozlik haqidagi ma'lumotning qiymatini ko'rib chiqing. Quyida tarmoqning ba'zi variantlari uchun qiymatlar (optimal tamoyilning bir kasrgacha aniqlik bilan kutilayotgan foydasi) keltirilgan. N_0 ni asl tarmoq deb olaylik.

- Tarmoq N_0 ning qiymati -22.6.

- N_1 ni N_0 ga *Tampering* dan *Call* ga yoy qo'shilgan holati deb olaylik. N_1 ning qiymati -21.3.

- N_2 ni N_1 ga *Tampering* dan *Check smoke* ga ham yoy qo'shilgan holati deb olaylik. N_2 ning qiymati -20.9.

- N_3 ni N_2 ning *Report* dan *Check smoke* ga yoy qo'shilmagan holati deb olaylik. N_3 ning qiymati N_2 bilan bir xil.

Birinchi ikki qaror tarmog'i uchun optimal tamoyillarning qiymatlari farqi, ya'ni 1.3, N_0 tarmog'ida *Call* qarori uchun *Tampering* haqidagi ma'lumotning qiymatidir. N_0 tarmog'ida *Check smoke* qarori uchun *Tampering* haqidagi ma'lumotning qiymati 1.7. Shunday qilib, *Tampering* sensorini o'rnatish kutilgan foydani 1.7 ga oshirishi mumkin.

N_3 kontekstida, *check smoke* uchun *Tampering* haqidagi ma'lumotning qiymati 0. Ikkala yoyli tarmoq uchun optimal tamoyilda, *Alarm* haqidagi ma'lumot *Check smoke* uchun optimal qaror funksiyasida e'tiborga olinmaydi; agent optimal tamoyilda *Call* qarorini qabul qilayotganda *Smoke* ni hech qachon tekshirmaydi.

Nazoratning qiymati o'zgaruvchini nazorat qilish qanchalik qimmatli ekanligini belgilaydi. Eng oddiy shaklda, bu tasodifiy o'zgaruvchi qaror o'zgaruvchisi bilan almashtirilgan qaror tarmog'ining qiymatidagi o'zgarish va tarmoqni unutmaydigan tarmoq qilish uchun yoylar qo'shilgan holda. Agar bu amalga oshirilsa,

foydalilikdagi o'zgarish salbiy bo'lmaydi; natijada tarmoq har doim asl tarmoqdan teng yoki yuqori kutilgan foydaga ega bo'ladi.

12.27-misol. 12.11-rasmdagi yong'in signalizatsiyasi qaror tarmog'ida, siz *Tampering* ni nazorat qilish qiymati bilan qiziqishingiz mumkin. Bu, masalan, *Tampering* ni oldini olish uchun xavfsizlik qo'riqchilarini qo'shishning qancha qimmatga tushishini baholash uchun ishlatilishi mumkin. Buni hisoblash uchun, 12.11-rasmdagi qaror tarmog'ining qiymatini *Tampering* qaror tuguni bo'lgan va boshqa ikkita qaror tugunining asl to'plami bo'lgan qaror tarmog'i bilan solishtiring.

Nazorat qiymatini aniqlash uchun *Tampering* tugunini qaror tuguniga aylantirish va uni boshqa ikki qarorning parents (asl to'plami) ga aylantirish. Natijadagi tarmoqning qiymati – 20.7. Bu 12.26-misoldagi *N3* qiymati bilan solishtirilishi mumkin (bir xil yoylarga ega va *Tampering* qaror yoki tasodifiy tugun bo'lishidan farq qiladi), bu -20.9 edi. Nazorat ma'lumotdan ko'ra qimmatroq ekanligini ko'rsatib.

Yuqoridagi tavsif tasodifiy o'zgaruvchini nazorat qilayotganda uning asl to'plami qaror o'zgaruvchisining asl to'plami bo'lishini nazarda tutadi. Bu holda, nazoratning qiymati hech qachon salbiy emas. Ammo, agar qaror tugunining asl to'plami tasodifiy o'zgaruvchining barcha asl to'plami o'z ichiga olmasa, nazorat ma'lumotdan kamroq qimmatli bo'lishi mumkin. Umuman olganda, o'zgaruvchini nazorat qilayotganda qaysi ma'lumotlar mavjud bo'lishini aniq belgilash kerak.

12.28-misol. 12.11-rasmdagi tarmoqda *Smoke* o'zgaruvchisini nazorat qilishni ko'rib chiqing. Agar *Fire* qaror o'zgaruvchisi *Smoking* asl to'plami bo'lsa, u tarmoqni no-forgetting (unutmaydigan) tarmoq qilish uchun *Calling* asl to'plami bo'lishi kerak. *Smoke*, *Check smoke* dan oldin keladigan tarmoqning kutilgan foydasi -2.0. Bu holatda *Smoke* ni nazorat qilishning qiymati *Fire* ni kuzatish natijasidir. Natijadagi optimal qaror yong'in bo'lsa, qo'ng'iroq qilish va aks holda qo'ng'iroq qilmaslikdir.

Aytaylik, agent *Fire* ni kuzatmasdan *Smoke* ni nazorat qilishni xohladi. Ya'ni agent tutun chiqarish yoki tutunning oldini olishga qaror qilishi mumkin va *Fire* hech qanday qarorning asl to'plami emas. Bu holat *Smoke* qaror o'zgaruvchisi hech qanday asl to'plami bo'lmagan holda modellastirilishi mumkin. Bu holda, kutilgan foyda -23.20, bu dastlabki qaror tarmog'idan yomonroq, chunki *Smoke* ni ko'r-ko'rona nazorat qilish uning *Fire* uchun sensor sifatida ishlash qobiliyatini yo'qotadi.

12.5. Qaror qabul qilish jarayonlari

Eslatib o'tamiz, **rejalashtirish gorizonti** agent rejalashtirishda qancha oldinga qarashini anglatadi. 12.3.1-mavzudagi qaror tarmoqlari cheklangan bosqichli, qisman kuzatiladigan domenlar uchun edi. Ushbu bo'lim noaniq gorizont va cheksiz gorizont muammolarini ko'rib chiqadi.

Ko'pincha agent davom etayotgan jarayon haqida mulohaza yuritishi kerak yoki u qancha harakat qilish kerakligini bilmaydi. Jarayon abadiy davom etishi mumkin bo'lganida bular **cheksiz gorizont muammolari** yoki agent oxir-oqibat to'xtaydigan lekin, qachon to'xtashini bilmaydigan **noaniq gorizont muammolari** deb ataladi.

Davom etayotgan jarayonlar uchun faqat oxirgi foydani ko'rib chiqish mantiqqa to'g'ri kelmasligi mumkin, chunki agent hech qachon oxiriga yetmasligi mumkin. Buning o'rniga, agent rag'batlar ketma-ketligini olishi mumkin. **Rag'batlar** vaqt davomida foydani hisobga olish usulini ta'minlaydi, har bir vaqt uchun rag'bat olib, rag'batlarni to'plab, foydani aniqlash mumkin. Rag'batlar harakat xarajatlarini, shuningdek, berilishi mumkin bo'lgan har qanday sovrin yoki jazolarni o'z ichiga olishi mumkin. Salbiy rag'batlar **jazolar** deb ataladi. Noaniq gorizont muammolari to'xtash holatidan foydalanib modellashtirilishi mumkin. **To'xtash holati** yoki **yutilish holati** – bu holatda barcha harakatlar ta'sir qilmaydi; ya'ni agent bu holatda bo'lsa, barcha harakatlar darhol nol rag'bat l bilan shu holatga qaytadi. Maqsadga erishish bunday to'xtash holatiga kirganlik uchun rag'bat l olish orqali modellashtirilishi mumkin.

Markov qaror qabul qilish jarayoni Markov zanjiri harakatlar va rag'batlar bilan boyitilgan yoki vaqt o'tishi bilan kengaytirilgan qaror tarmog'i sifatida ko'rinishi mumkin. Har bir bosqichda agent qaysi harakatni bajarishni tanlaydi; rag'bat va natijaviy holat avvalgi holat va bajarilgan harakatga bog'liq.

Agar ko'rsatilmagan bo'lsa, **statsionar modelni** faraz qiling, bunda holat o'tishlari va rag'batlar vaqtga bog'liq emas. Markov qaror jarayoni yoki MDP quyidagilardan iborat:

S , dunyo holatlarining to'plami

A , harakatlar to'plami

$P: S \times S \times A \rightarrow [0,1]$ **dinamikani** belgilaydi. Bu $P(s' | s, a)$ deb yoziladi, agent holatda s bo'lib, a harakatni qilganda holat s' ga o'tish ehtimolini bildiradi. Shunday qilib:

$$\forall s \in S \forall a \in A \sum_{s' \in S} P(s' | s, a) = 1.$$

$R: S \times A \times S \rightarrow R$ **rag'bat funksiyasi** bo'lib, bu $R(s, a, s')$, s holatdan a holatga harakatini bajarib o'tishdan keladigan kutilgan darhol rag'batni bildiradi. Ba'zan $R(s, a)$ ni, ya'ni holatda s bo'lgan holatda a harakatini bajarishning kutilgan qiymatini ishlatish qulay bo'ladi, bu quyidagicha aniqlanadi:

$$R(s, a) = \sum_{s'} R(s, a, s') * P(s' | s, a)$$

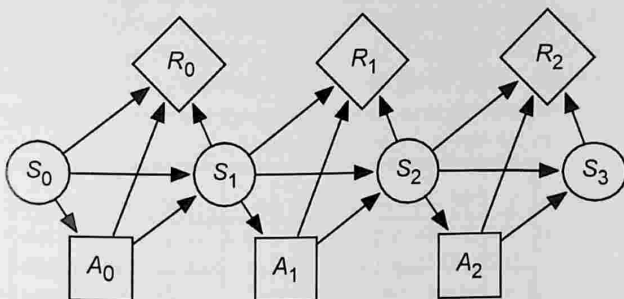
Markov qaror jarayonining cheklangan qismi 12.15-rasmdagi kabi qaror tarmog'i yordamida tasvirlanishi mumkin.

12.29-misol. Sem dam olish kunlari ziyofatda qatnashish yoki dam olish haqida ongli qaror qabul qilishni xohlaydi. Sem ziyofatda qatnashishni afzal ko'radi, lekin kasal bo'lib qolishdan xavotirda. Bunday muammo ikki holat: sog'lom va kasal va ikki harakat: dam olish va ziyofatda qatnashish bilan MDP sifatida modellashirilishi mumkin. Shunday qilib:

$$S = \{healthy, sick\}$$

$$A = \{relax, party\}.$$

Tajriba asosida, Sem dinamikani $P(s'|s,a)$ quyidagicha baholaydi:



12.15-rasm. MDPning cheklangan qismini ifodalovchi qarorlar tarmog'i

S	A	Probability of $s' = healthy$
healthy	relax	0.95
healthy	party	0.7
sick	relax	0.5
sick	party	0.1

Shunday qilib, agar Sem sog'lom bo'lsa va ziyofatda qatnashsa, kasal bo'lish ehtimoli 30%. Agar Sem sog'lom bo'lsa va dam olsa, sog'lom bo'lib qolish ehtimoli ko'proq. Agar Sem kasal bo'lsa va dam olsa, tuzalish ehtimoli 50%. Agar Sem kasal bo'lsa va ziyofatda qatnashsa, sog'lom bo'lish ehtimoli faqat 10%.

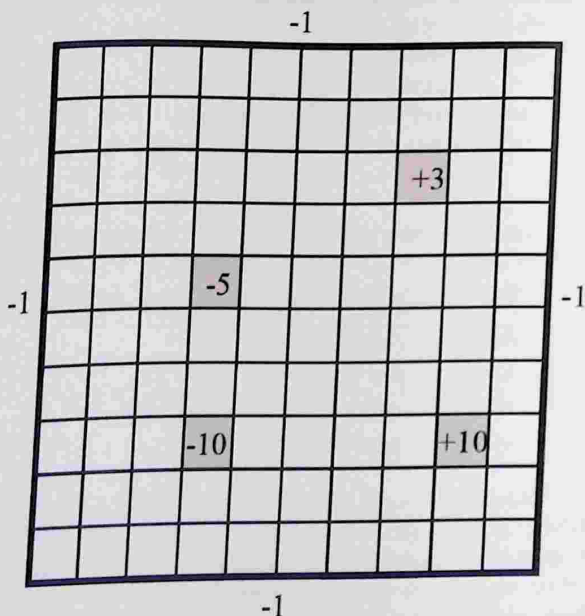
Sem, natijaviy holatdan qat'i nazar rag'batlarni quyidagicha baholaydi:

<i>S</i>	<i>A</i>	Reward
<i>healthy</i>	<i>relax</i>	7
<i>healthy</i>	<i>party</i>	10
<i>sick</i>	<i>relax</i>	0
<i>sick</i>	<i>party</i>	2

Shunday qilib, Sem har doim ziyofatda qatnashishni dam olishdan ko'ra ko'proq yoqtiradi. Biroq Sem umuman sog'lom bo'lganda o'zini ancha yaxshi his qiladi va ziyofatda qatnashish dam olishga qaraganda ko'proq kasal bo'lishga olib keladi.

Muammo Sem har hafta oxirida nima qilish kerakligini aniqlashdir.

12.30-misol. Tarmoq dunyosi muhitda bo'lgan robotning idealizatsiyasi. Har bir vaqtda, robot ba'zi joyda bo'lib, qo'shni joylarga harakatlanishi, rag'batlar va jazolarni to'plashi mumkin. Faraz qilaylik, harakatlar stoxastik bo'lib, holat va harakat berilgan natijaviy holatlar bo'yicha ehtimollik taqsimotiga ega.



12.16-rasm. 12.30-misoldan tarmoq dunyosi

12.16-rasm 10×10 tarmoq dunyosini ko'rsatadi, bu yerda robot to'rtta harakatdan birini tanlashi mumkin: yuqoriga, pastga, chapga yoki o'ngga. Agar agent ushbu harakatlardan birini amalga oshirsa, u kutilgan yo'nalishda bir qadam tashlash ehtimoli 0.7 va boshqa uch yo'nalishdan birida bir qadam tashlash ehtimoli 0.1 bo'ladi. Agar u tashqi devorga urilsa (ya'ni hisoblangan joy tarmoqdan tashqarida bo'lsa), 1 ta jazoga ega bo'ladi (ya'ni -1 rag'bat) va agent haqiqatan ham harakat qilmaydi. To'rt rag'batli holat mavjud (devorlardan tashqari), biri +10 (9, 8-pozitsiyada; 9 bo'ylab va 8 pastga), biri +3 (8, 3-pozitsiyada), biri -5 (4, 5-pozitsiyada) va biri -10 (4, 8-pozitsiyada) qiymatiga ega. Ushbu holatlarning har birida agent bu holatda harakatni amalga oshirgandan keyin mukofot oladi, holatga kirganida emas. Agent ijobiy mukofotga ega holatlardan biriga (ya'ni +3 yoki +10) yetganida, qanday harakatni bajarganidan qat'i nazar, keyingi qadamda tasodifan tarmoqning to'rtta burchagidan biriga tashlanadi.

E'tibor bering, ushbu misoldagi rag'bat ham dastlabki holatga, ham yakuniy holatga bog'liq. Agent devorga urildi va agar agent shu holatda qolsagina -1 miqdorida mukofot oldi. Faqat dastlabki holat va harakatni yoki faqat yakuniy holat va harakatni bilish rag'bat haqida xulosa chiqarish uchun yetarli ma'lumot bermaydi. Qaror tarmoqlaridagi kabi, dizayner agent nima qilishni hal qilganda qanday ma'lumotga ega ekanligini ham hisobga olishi kerak. Ikkita variant mavjud:

To'liq kuzatiladigan Markov qaror jarayoni (MDP): agent joriy holatni kuzatib, nima qilish kerakligini hal qilganda bu holatdan foydalanadi.

Qisman kuzatiladigan Markov qaror jarayoni (POMDP): MDP va **yashirin Markov modeli (HMM)** kombinatsiyasi. Har bir vaqtda agent holatga bog'liq bo'lgan ba'zi (noaniq va ehtimol shovqinli) kuzatishlarni amalga oshiradi. Agent qaror qabul qilayotganda faqat rag'batlar, kuzatishlar va oldingi harakatlar tarixiga ega bo'ladi. Joriy holatni to'g'ridan to'g'ri kuzatish imkoniyati yo'q.

Mukofotlar

Nima qilishni hal qilish uchun agent turli mukofotlar ketma-ketligini taqqoslaydi. Buning eng keng tarqalgan usuli – mukofotlar ketma-ketligini **daromad, jamlangan mukofot** yoki **qiymat** deb nomlangan raqamga aylantirishdir. Bu agentga joriy va kelajakdagi rag'batlarning foydasini ko'rsatadigan raqamdir. Daromad hisoblash uchun agent joriy mukofotni kelajakdagi boshqa mukofotlar bilan birlashtiradi. Faraz qilaylik, agent mukofotlar ketma-ketligini oladi:

$$r_1, r_2, r_3, r_4, \dots$$

Uchta umumiy mukofot kriteriyasi mukofotlarni qiymat V ga birlashtirish uchun ishlatiladi:

Umumiy mukofot $V = \sum_{i=1}^{\infty} r^i$ Bu holatda qiymat barcha mukofotlarning yig'indisidir. Bu yig'indining cheklanganligi kafolatlagan holatlarda ishlaydi; lekin agar yig'indi cheksiz bo'lsa, u qaysi mukofotlar ketma-ketligini afzalroq ekanligini taqqoslash imkoniyatini bermaydi. Masalan, 1 mukofotlar ketma-ketligi 100 mukofotlar ketma-ketligi bilan bir xil umumiy mukofotga ega (ikkalasi ham cheksiz). Umumiy mukofot cheklangan bo'lgan holat to'xtash holatlari mavjud bo'lganda va agent har doim to'xtash holatiga kirishning nolga teng bo'lmagan ehtimoliga ega bo'lganda sodir bo'ladi.

O'rtacha mukofot: $V = \lim_{n \rightarrow \infty} (r_1 + \dots + r_n)/n$. Bu holatda agentning qiymati mukofotlarning har bir vaqt davrida o'rtacha qiymati hisoblanadi. Mukofotlar cheklangan ekan, bu qiymat ham cheklangan bo'ladi. Biroq, umumiy mukofot cheklangan bo'lsa, o'rtacha mukofot nolga teng bo'ladi, shuning uchun o'rtacha mukofot agentni har xil nol o'rtacha mukofotga ega bo'lgan harakatlarni tanlashga imkon bermaydi. Ushbu kriteriyga ko'ra, agentning qayerda tugashi muhim. Har qanday yomon harakatlar ketma-ketligi chegaraga ta'sir qilmaydi. Masalan, 1,000,000 mukofotini olgandan so'ng 1 mukofotlari olingan ketma-ketlik 0 mukofotini olgandan so'ng 1 mukofotlari olingan ketma-ketlik bilan bir xil o'rtacha mukofotga ega (ikkalasi ham 1 o'rtacha mukofotga ega).

Chegirma mukofoti: $V = r_1 + \gamma r_2 + \gamma^2 r_3 + \dots + \gamma^{i-1} r_i + \dots$ bu yerda γ , chegirma omili, $0 \leq \gamma < 1$. oralig'ida bo'lgan raqam. Ushbu kriteriyga ko'ra kelajakdagi mukofotlar joriy mukofotdan kamroq qiymatga ega bo'ladi. Agar $\gamma = 1$ bo'lsa, bu umumiy mukofotga teng bo'ladi. Agar $\gamma = 0$ bo'lsa, agent barcha kelajakdagi mukofotlarini e'tiborsiz qoldiradi. $\gamma = 0 \leq \gamma < 1$ bo'lganida, mukofotlar cheklangan bo'lganda umumiy qiymat ham cheklangan bo'ladi.

Chegirma mukofoti quyidagicha yozilishi mumkin:

$$\begin{aligned} V &= \sum_{i=1}^{\infty} \gamma^{i-1} r_i \\ &= r_1 + \gamma r_2 + \gamma^2 r_3 + \dots + \gamma^{i-1} r_i + \dots \\ &= r_1 + \gamma(r_2 + \gamma(r_3 + \dots)) \end{aligned}$$

Faraz qilaylik, V_k vaqt k dan boshlab yig'ilgan mukofotdir:

$$\begin{aligned} V_k &= r_k + \gamma(r_{k+1} + \gamma(r_{k+2} + \dots)) \\ &= r_k + \gamma V_{k+1} \end{aligned}$$

V_k ning xususiyatlarini tushunish uchun, faraz qilaylik, $S = 1 + \gamma + \gamma^2 + \gamma^3 + \dots$ keyin $S = 1 + \gamma S$. S ni yechish $S = 1/(1-\gamma)$ ni beradi. Shunday qilib, chegirma mukofotida kelajakning barcha qiymati joriy mukofotga nisbatan $1/(1-\gamma)$ marta ko'p va $1/(1-\gamma)$ marta kam bo'ladi. Shunday qilib, hozirgi mukofot bilan taqqos-

laganda kelajakdagi mukofotning qiymati cheklangan bo'ladi, o'rtacha mukofotda esa hozirgi mukofot kelajakdagi mukofotning yig'indisidan ustunroq bo'ladi.

Iqtisodiyotda γ foiz stavkasiga bog'liq: hozirda \$1 olish bir yil ichida $\$(1 + i)$ olish bilan teng, bu yerda i foiz stavkasi. Siz ham chegirma stavkasini agentning omon qolish ehtimoli sifatida ko'rishingiz mumkin; γ agentning davom etish ehtimoli sifatida qaralishi mumkin.

Ushbu bobning qolgan qismida **daromad** yoki **qiymat** deb ataladigan chegirmali mukofotlar ko'rib chiqiladi.

12.5.1. Tamoyillar

To'liq kuzatiladigan Markov qaror jarayonida agent harakatni amalga oshirishdan oldin joriy holatini kuzatadi. Hozircha Markov qaror jarayoni to'liq kuzatiladi deb faraz qilamiz. MDP uchun tamoyil agent qaysi holatda bo'lsa, nima qilish kerakligini belgilaydi. Statsionar π tamoyil: $S \rightarrow A$. funksiyasidir. Statsionar bo'lmagan tamoyilda harakat holat va vaqtdan kelib chiqadi; biz tamoyillarning statsionar ekanligini faraz qilamiz.

Mukofot kriteriysini hisobga olgan holda, har bir holat uchun tamoyilning **kutilgan daromadi**, ko'pincha **kutilgan qiymat** deb ataladi. π tamoyiliga rioya qilishda s holatidagi kutilgan qiymatni $V\pi(s)$ deb belgilaymiz. Bu s holatidagi va tamoyil π ni kuzatayotgan agentning o'rtacha oladigan foydasi. π tamoyil har qanday boshqa π' tamoyil va har qanday s holat uchun **optimal tamoyil** hisoblanadi, agar

$V\pi(s) \geq V\pi'(s)$ bo'lsa. Ya'ni har bir holat uchun optimal tamoyil boshqa har qanday tamoyildan kamida teng yoki yaxshi.

12.31-misol. 12.29-misolda ikki holat va ikki harakat bilan, jami

$2^2=4$ tamoyil mavjud:

Doim dam olish.

Doim ziyofat qilish.

Sog'lom bo'lsa dam olish, kasal bo'lsa ziyofat qilish.

Sog'lom bo'lsa ziyofat qilish, kasal bo'lsa dam olish.

Ularning barchasi uchun umumiy mukofot cheksizdir, chunki agent hech qachon to'xtamaydi va hech qachon 0 mukofot ololmaydi. O'rtacha mukofotni aniqlash mashq sifatida qoldiriladi (12.15-mashq). Chegirma mukofotini qanday hisoblash keyingi qismda muhokama qilinadi.

12.32-misol. 12.30-misolda tarmoq dunyosi MDPda, 100 holat va 4 harakat mavjud, shuning uchun $4^{100} \approx 10^{60}$ statsionar tamoyillar mavjud. Har bir tamoyil har bir holat uchun harakatni belgilaydi.

Cheksiz gorizontli muammolar uchun statsionar MDP har doim optimal statsionar tamoyilga ega bo'ladi. Biroq, cheklangan bosqichli muammolar uchun statsionar bo'lmagan tamoyil barcha statsionar tamoyillardan yaxshiroq bo'lis-

hi mumkin. Masalan, agar agent n vaqtida to'xtashi kerak bo'lsa, ba'zi holatda agent kelajakdagi harakatlarni hisobga olmasdan eng katta darhol mukofotni olish uchun harakat qiladi, ammo keyinchalik katta mukofotni olish uchun ilgari qabul qilingan qarorlar uchun darhol pastroq mukofotni olishga qaror qilishi mumkin.

Chegirmalash asoslari

Chegirmalash mukofotlarida qandaydir maxsus narsa bormi yoki ular vaqt o'tishi bilan foydani aniqlashning ixtiyoriy usulimi, deb qiziqishingiz mumkin. Foydalilik 12.1.1-mavzudagi aksiomalardan kelib chiqadi. Chegirmalash quyidagi taxminlar to'plamidan kelib chiqishi isbotlanishi mumkin.

Cheksiz natijalar ketma-ketligi $(o_1, o_2, o_3, \dots)$ bilan quyidagi taxminlar qo'llaniladi:

Birinchi vaqt davri muhim, shuning uchun o_1, o_2, o_3 va o_1 mavjud:

$$\langle o_1, o_2, o_3, \dots \rangle > o_1', o_2', o_3', \dots$$

Additiv mustaqillik shakli, unda dastlabki ikki vaqt oralig'idagi afzalliklar kelajakka bog'liq emas:

$$\langle x_1, x_2, o_3, o_4 \dots \rangle > \langle y_1, y_2, o_3, o_4 \dots \rangle$$

faqat va faqat

$$\langle x_1, x_2, o_3, o_4 \dots \rangle > \langle y_1, y_2, o_3, o_4 \dots \rangle$$

Vaqtning statsionarligi, bu yerda birinchi natija bir xil bo'lsa, afzallik qoldiqqa bog'liq:

$$\langle o_1, o_2, o_3, \dots \rangle > o_1', o_2', o_3', \dots$$

agar va faqat agar

$$\langle o_2, o_3, \dots \rangle > o_2', o_3', \dots$$

Agentning faqat cheksiz vaqtning cheklangan pastki fazolari bilan qiziqishini ko'rsatadigan ba'zi qo'shimcha texnik shartlar:

faqat va faqat γ chegirma omili va r funksiyasi mavjud bo'lsa, shunda:

$$utility(o_1, o_2, o_3, \dots) = \sum_i \gamma^{i-1} r(o_i)$$

bu yerda r faqat natijaga bog'liq, vaqtga emas.

Tamoyilning qiymati

γ chegirma omili berilgan holda tamoyilning chegirma mukofotidan foydalanib, kutilgan qiymatni qanday hisoblashni ko'rib chiqing. Qiymat ikki o'zaro bog'liq funksiya yordamida aniqlanadi:

$V^\pi(s)$ agentning s holatida bo'lib, π tamoyiliga amal qilishi kutilgan qiymati agent uchun kutilgan qiymat bo'lib, u s holatda boshlanadi, so'ngra a amalni bajaradi va keyin π tamoyiliga amal qiladi. Bu π tamoyilining Q-qiymati deb ataladi.

$Q^\pi(s, a)$ va V^π bir-biriga nisbatan rekursiv tarzda aniqlanadi. Agar agent s holatida bo'lsa, a harakatini bajarsa va s' holatiga kelsa, u $R(s, a, s')$ darhol mukofotini va kelajakdagi chegirmalangan qaytishni $\gamma V^\pi(s')$ oladi. Agent rejalashtirish paytida haqiqiy natijaviy holatni bilmaydi, shuning uchun u natijaviy holatlar bo'yicha o'rtacha kutilgan qiymatdan foydalanadi:

$$\begin{aligned} Q^\pi(s, a) &= \sum_{s'} P(s' | s, a) (R(s, a, s') + \gamma V^\pi(s')) \\ &= R(s, a) + \gamma \sum_{s'} P(s' | s, a) V^\pi(s') \end{aligned} \quad (12.2)$$

Bu yerda $R(s, a) = \sum_{s'} P(s', a) R(s, a, s')$

$V^\pi(s) = \pi$ tomonidan belgilangan harakatni bajarib, so'ngra π ga amal qilish orqali olinadi:

$$V^\pi(s) = Q^\pi(s, \pi(s))$$

Optimal tamoyil qiymati

$Q^*(s, a)$, bu yerda s holat va a harakat bo'lsin, s holatida a ni bajarish va keyin optimal tamoyilga rioya qilishning kutilgan qiymati bo'lsin. $V^*(s)$ – bu yerda s holat, s holatidan optimal tamoyilga rioya qilishning kutilgan qiymati bo'lsin.

Q^*Q^* bilan o'xshash tarzda aniqlanishi mumkin:

$$\begin{aligned} Q^*(s, a) &= \sum_{s'} P(s' | s, a) (R(s, a, s') + \gamma V^*(s')) \\ &= R(s, a) + \gamma \sum_{s'} P(s' | s, a) V^*(s'). \end{aligned}$$

$V^*(s)$ har bir holatda eng yaxshi qiymatni beruvchi harakatni bajarish orqali olinadi:

$$V^*(s) = \max_a Q^*(s, a).$$

π^* optimal tamoyil har bir holat uchun eng yaxshi qiymatni beruvchi tamoyillardan biridir:

$$\pi^*(s) = \arg \max_a Q^*(s, a)$$

bu yerda $\arg \max_a Q^*(s, a)$

s holatining funksiyasi bo'lib, uning qiymati $Q^*(s, a)$ ning maksimal qiymatiga ega harakat a .

12.5.2. Qiymatni takrorlash

Qiymatni takrorlash – MDP (Markov qaror jarayoni) uchun optimal tamoyil va uning qiymatini hisoblash usulidir. Qiymatni takrorlash jarayoni «oxiridan» boshlanadi va keyin orqaga qarab ishlaydi, Q^* yoki V^* ni baholashni takomillash-tiradi. Aslida oxiri yo'q, shuning uchun tasodifiy bir oxirgi nuqtadan foydalani-ladi. K bosqich qolgan deb hisoblasak, V_k qiymat funksiyasi va Q_k Q -funksiyasi bo'lsin. Ularni rekursiv tarzda aniqlash mumkin. Qiymat iteratsiyasi tasodifiy V_0 funksiyasidan boshlanadi. Keyingi bosqichlar uchun, k bosqichdan $k+1$ bosqichga o'tish uchun quyidagi tenglamalardan foydalanadi:

$$Q_{k+1}(s, a) = R(s, a) + \gamma \sum_{s'} P(s' | s, a) * V_k(s')$$

$$V_k(s) = \max_a Q_k(s, a).$$

Qiymatni takrorlash $V[S]$ massivini yoki $Q[S, A]$ massivini saqlashi mumkin. V massivini saqlash kamroq saqlash maydonini talab qiladi, lekin optimal harakat-ni aniqlash qiyinroq bo'ladi va eng katta qiymatni beruvchi harakatni aniqlash uchun bir marta qo'shimcha takrorlash zarur bo'ladi.

12.17-rasm V massivi saqlanadigan qiymat takrorlash algoritmini ko'rsatadi. Ushbu tartib V_0 qiymatining boshlang'ich funksiyasi qanday bo'lishidan qat'i nazar birlashadi. V^* ni taxmin qiladigan dastlabki qiymat funksiyasi, taxmin qilin-maganidan tezroq birlashadi. MDPlar uchun ko'plab abstraksiya texnikalarining asosida V^* ni taxmin qilish uchun ba'zi bir evristik usuldan foydalanish va bu qiymatni qiymat takrorlash uchun dastlabki seed sifatida ishlatish yotadi.

12.33-misol. 12.29-misolida ko'rsatilgan ikki holatli MDPni $\gamma = 0.8$ chegir-masi bilan ko'rib chiqing. Qiymat funksiyasini [*healthy value*, *sick value*] deb yozamiz va Q -funksiyasini [*healthy relax*, *healthy party*], [*sick relax*, *sick party*] deb yozamiz. Dastlab, qiymat funksiyasi $[0, 0]$ deb faraz qilamiz. Keyingi Q -qiymat $[[7, 10], [0, 2]]$ bo'ladi, shuning uchun keyingi qiymat funksiyasi $[10, 2]$ bo'ladi (Sem ziyofat qilgan holda). Keyingi Q -qiymat quyidagicha bo'ladi:

State	Action	Value
healthy	relax	$7 + 0.8 * (0.95 * 10 + 0.05 * 2) = 14.68$
healthy	party	$10 + 0.8 * (0.7 * 10 + 0.3 * 2) = 16.08$
sick	relax	$0 + 0.8 * (0.5 * 10 + 0.5 * 2) = 4.8$
sick	party	$2 + 0.8 * (0.1 * 10 + 0.9 * 2) = 4.24$

Shunday qilib, keyingi qiymat funksiyasi [16.08, 4.8] bo'ladi. 1000 takrorlashdan so'ng, qiymat funksiyasi [35.71, 23.81] bo'ladi. Shunday qilib, Q -funksiyasi [[35.10, 35.71], [23.81, 22.0]] bo'ladi. Demak, optimal tamoyil sog'lom bo'lganda ziyofat qilish va kasal bo'lganda dam olishdir.

12.34-misol. 12.30-misolida ko'rsatilgan +10 mukofot atrofidagi to'qqizta kvadratni ko'rib chiqing. Chegirma $\gamma = 0.9$. Algoritm $V_0[s] = 0$ holatlari bilan boshlangan deb faraz qilamiz.

V_1 , V_2 va V_3 (bir kasrgacha aniqlik bilan) qiymatlari quyidagi to'qqizta katak uchun:

0	0	-0.1	0	6.3	-0.1	4.5	6.2	4.4
0	10	-0.1	6.3	9.8	6.2	6.2	9.7	6.6
0	0	-0.1	0	6.3	-0.1	4.5	6.1	4.4
V_1			V_2			V_3		

V_1 bosqichida qiymat takrorlashning birinchi qadamidan so'ng, tugunlar darhol kutilgan mukofotni oladi. Bu tasvirdagi markaziy tugun +10 mukofot holatidir. O'ng tugunlar -0.1 qiymatiga ega bo'lib, optimal harakatlar yuqoriga, chapga va pastga; har birida darhol kutilgan mukofot -1 bo'lgan 0.1 ehtimollik bilan devorga urilish xavfi mavjud.

V_2 - bu qiymatlarni takrorlashning ikkinchi bosqichidan keyingi qiymatlar. +10 mukofot holatidan chapdagi tugunni ko'rib chiqing. Uning optimal qiymati o'ngga o'tishdir; u quyidagi holatda 10 mukofot olish uchun 0.7 imkoniyatga ega, shuning uchun endi bu 9 ga teng (0.9 chegirmaning 10 barobari). Boshqa mumkin bo'lgan natijaviy holatlar uchun kutilgan mukofot 0. Shunday qilib, ushbu holatning qiymati $0.7 * 9 = 6.3$

+10 mukofot holatidan ikkinchi qadamdan keyin o'ngdagi tugunni ko'rib chiqing. Agentning optimal harakati chapga borishdir. Ushbu holatning qiymati:

Prob	Reward	Future Value	
0.7 * (0	+ 0.9 * 10)	<i>Agent goes left</i>	
+ 0.1 * (0	+ 0.9 * -0.1)	<i>Agent goes up</i>	
+ 0.1 * (-1	+ 0.9 * -0.1)	<i>Agent goes right</i>	
+ 0.1 * (0	+ 0.9 * -0.1)	<i>Agent goes down</i>	

Bu 6.173 ga teng bo'ladi, bu yuqoridagi V_2 da 6.2 ga yaqin. +10 mukofot holatining qiymati V_2 da 10 dan kamroq bo'ladi, chunki agent burchaklardan biriga kelib qoladi va bu burchaklar ushbu bosqichda yomon ko'rinadi.

Keyingi qiymat takrorlash bosqichidan so'ng, +10 mukofotning ta'siri bir bosqichga ilgarilaydi. Ayniqsa, ko'rsatilgan burchaklar uch bosqichdan mukofot olishni ko'rsatadi.

```

1: procedure Value_iteration ( $S, A, P, R$ )
2: Inputs
3:  $S$  is the set of all states
4:  $A$  is the set of all actions
5:  $P$  is the state transition function specifying  $P(s' | s, a)$ 
6:  $R$  is a reward function  $R(s, a)$ 
7: Output
8:  $\pi[S]$  approximately optimal policy
9:  $V[S]$  value function
10: Local
11: real array  $V_k[S]$  is a sequence of value functions
12: action array  $\pi[S]$ 
13: assign  $V_0[S]$  arbitrarily
14:  $k := 0$ 
15: repeat
16:  $k := k + 1$ 
17: for each state  $s$  do
18:  $V_k[s] = \max_a R(s, a) + \gamma \sum_{s'} P(s' | s, a) * V_{k-1}[s']$ 
19: until termination
20: for each state  $s$  do
21:  $\pi[s] = \arg \max_a R(s, a) + \gamma \sum_{s'} P(s' | s, a) * V_k[s']$ 
22: return  $\pi, V_k$ 

```

12.17-rasm. MDPlar uchun qiymat takrorlanishi, V ni saqlash

12.17-rasm qiymat takrorlash algoritmi har bir bosqich uchun massivni saqlaydi, lekin aslida faqat hozirgi va oldingi massivlarni saqlashi kerak. Bir massivni boshqa massivlardan qiymatlar asosida yangilashi mumkin.

Ushbu algoritmning umumiy qayta ishlash varianti **asinxron qiymat takrorlashidir**. Yangi qiymat funksiyasini yaratish uchun holatlarni ko'rib chiqish o'rni-ga, asinxron qiymat takrorlashi holatlarni birma-bir, istalgan tartibda yangilaydi va qiymatlarni bitta massivda saqlaydi.

Asinxron qiymat takrorlash $Q[s, a]$ massivini yoki $V[s]$ massivini saqlashi mumkin. 12.18-rasm asinxron qiymat takrorlashni Q -massivida saqlashda ko'rsatadi. U qiymatlarni takrorlashdan ko'ra tezroq birlashadi va ba'zi mustahkamlashni o'rganish algoritmlarining asosidir. Agent ma'lum bir xatolikni kafolatlashi kerak bo'lsa, yakunlanishni aniqlash qiyin bo'lishi mumkin, agar harakatlar va holatlar

qanday tanlanishini diqqat bilan ko'rib chiqmasa. Ko'pincha, bu jarayon **anytime algorithm** sifatida cheksiz ravishda ishlaydi, bu holda har doim so'ralsa, holatda optimal harakatning eng yaxshi taxminini beradi. Asinxron qiymat takrorlanishi faqat $V[s]$ massivini saqlash orqali ham amalga oshirilishi mumkin. Bunday holda, algoritim bir holatni tanlaydi va yangilanishni amalga oshiradi:

$$V[s] := \max_a R(s, a) + \gamma * \sum_s P(s' = s, a) * V[s'].$$

Ushbu parametr kamroq ma'lumotni saqlasa-da, tamoyilni ajratib olish qiyinroq. Maksimal qiymatni aniqlash uchun qaysi harakat a ni tanlashni belgilashda bitta qo'shimcha zaxira nusxasi talab qilinadi. Bu quyidagi formula yordamida amalga oshirilishi mumkin:

$$\pi[s] := \arg \max_a R(s, a) + \gamma * \sum_s P(s' = s, a) * V[s'].$$

12.35-misol. 12.34-misolda +10 mukofot holatidan bir qadam yuqorida va chapda bo'lgan holat faqat uchta qiymat takrorlashidan so'ng yangilandi, bu jarayonlarning har birida barcha holatlar bo'ylab ko'rib chiqishni o'z ichiga oladi.

Asinxron qiymat iteratsiyasida +10 mukofot holati birinchi tanlanishi mumkin. Keyin undan chapidagi tugun tanlanadi va uning qiymati $0.70.910 = 6.3$ bo'ladi. Keyin ushbu tugunning yuqoridagi tuguni tanlanadi va uning qiymati $0.70.96.3 = 3.969$ bo'ladi. Uning qiymati 300 ta holat emas, uchta holatni hisobga olgandan keyin +10 mukofotga yaqinligini aks ettiradi, bu qiymat takrorlashda bo'lgani kabi.

12.5.3. Tamoyilni takrorlash

Tamoyilni takrorlash tamoyildan boshlanadi va uni iterativ ravishda yaxshilaydi. U ixtiyoriy tamoyildan π_0 (optimal tamoyilga yaqinlashish eng yaxshi ishlaydi) boshlanadi va $i = 0$ dan boshlab quyidagi amallarni bajaradi.

Tamoyilni baholash: $V^m(S)$ ni aniqlash. V^m ning aniqlanishi $|S|$ noma'lumlar ichida $|S|$ chiziqli tenglamalar to'plamidir. Noma'lumlar $V^m(S)$ qiymatlaridir. Har bir holat uchun tenglama mavjud. Ushbu tenglamalarni chiziqli tenglama yechim usuli bilan (masalan, Gauss usuli) yoki iterativ ravishda yechish mumkin.

Tamoyilni yaxshilash: $\pi_{i+1}(s) = \arg \max_a Q^m(s, a)$ tanlash, bu yerda Q qiymati V ni 12.2-tenglamadan foydalanib olish mumkin.

1: **procedure** *Asynchronous value iteration* (S, A, P, R)

2: **Inputs**

3: S is the set of all states

4: A is the set of all actions

- 5: P is the state transition function specifying $P(s' | s, a)$
- 6: R is a reward function $R(s, a)$
- 7: **Output**
- 8: $\pi[s]$ policy
- 9: $Q[S, A]$ value function
- 10: **Local**
- 11: real array $Q[S, A]$
- 12: action array $\pi[S]$
- 13: assign $Q[S, A]$ arbitrarily
- 14: **repeat**
- 15: select a state s
- 16: select an action a
- 17: $Q[s, a] = R(s, a) + \gamma * \sum_s' P(s' | s, a) \max_a' Q[s', a]$
- 18: **until** termination
- 19: **for each** state s **do**
- 20: $\pi[s] = \arg \max_a Q[s, a]$
- 21: **return** π, Q

12.18-rasm. MDP uchun qiymatlarning asinxron takrorlanishi

Algoritm qachon yaqinlashishini aniqlash uchun u tamoyilni faqat ba'zi bir holat uchun yangi amal kutilgan qiymatni yaxshilagan taqdirdagina o'zgartirishi kerak; ya'ni, $\pi_{i-1}(s)$ ni $\pi_i(s)$ ga tenglashtirishi kerak, agar $\pi_i(s) Q_{\pi_{i-1}}^{s,a}$ ni maksimal qiladigan harakatlardan biri bo'lsa.

Agar tamoyilda hech qanday o'zgarish bo'lmasa, to'xtating, agar $\pi_{i+1} = \pi_i$ bo'lsa, aks holda i ni oshiring va takrorlang.

Algoritm 12.19-rasmda ko'rsatilgan. E'tibor bering, u faqat tamoyilning eng so'nggi versiyasini saqlaydi va agar u o'zgargan bo'lsa, sizni xabardor qiladi. Ushbu algoritm har doim, odatda, oz sonli takrorlashlardan keyin to'xtaydi. Afsuski, chiziqli tenglamalar tizimini yechish ko'pincha uzoq vaqt talab etadi.

Tamoyilni takrorlashning o'zgartirilgan variantini, **o'zgartirilgan tamoyil iteratsiyasi** deb ataladi, agent tamoyilni yaxshilash uchun uni baholash shart emasligini payqash orqali qo'lga kiritiladi; u shunchaki 12.2-tenglamadan (foydalanib bir nechta xazira qadamlarni bajarishi va keyin yaxshilashni amalga oshirishi mumkin.

- 1: **procedure** *Policy iteration* (S, A, P, R)
- 2: **Inputs**
- 3: S is the set of all states
- 4: A is the set of all actions
- 5: P is the state transition function specifying $P(s' | s, a)$


```

6:  $R$  is a reward function  $R(s, a)$ 
7: Output
8: optimal policy  $\pi$ 
9: Local
10: action array  $\pi[S]$ 
11: Boolean variable  $noChange$ 
12: real array  $V[S]$ 
13: set  $\pi$  arbitrarily
14: repeat
15:  $noChange := true$ 
16: Solve  $V[s] = R(s, a) + \gamma \sum_{s' \in S} P(s'|s, \pi[s]) * V[s']$ 
17: for each  $s \in S$  do
18:  $QBest := V[s]$ 
19: for each  $a \in A$  do
20:  $Qsa := R(s, a) + \gamma \sum_{s' \in S} P(s'|s, a) * V[s']$ 
21: if  $Qsa > QBest$  then
22:  $\pi[s] := a$ 
23:  $QBest := Qsa$ 
24:  $noChange := false$ 
25: until  $noChange$ 
26: return  $\pi$ 

```

12.19-rasm. MDP uchun tamoyil iteratsiyasi

Tamoyilning takrorlanishi MDP sifatida aniq ifodalanishi juda katta tizimlar uchun foydalidir. Bitta holat – bu harakatlar uchun katta joy mavjud bo'lganda va agent har har safar barcha harakatlarni ro'yxatlashni xohlamaydi. Algoritm, shuningdek, yaxshilovchi harakat topilmaguncha ishlaydi va u faqat ehtimollik bilan yaxshilovchi harakatni topishi kerak, masalan, hamma narsani emas, balki bir nechta istiqbolli harakatlarni sinab ko'rish orqali.

Faraz qilaylik, bir boshqaruvchida o'zgartirilishi mumkin bo'lgan ba'zi parametrlar mavjud. Parametr a ning ba'zi kontekst s da to'plangan diskontlangan mukofotning hosilasi bahosi, $Q(a, s)$ ning hosilasiga mos keladi, bundan parametrlarni yaxshilash uchun foydalanish mumkin. Bunday iterativ ravishda yaxshilovchi boshqaruvchi global maksimum emas, mahalliy maksimumga kirib qolishi mumkin. Holatga asoslangan MDPlar uchun tamoyil takrorlanishi optimal bo'lmagan mahalliy maksimumlarga olib kelmaydi, chunki boshqa holatlarga ta'sir qilmasdan holat uchun harakatni yaxshilash mumkin, parametrlarni yangilash esa bir vaqtning o'zida ko'plab holatlarga ta'sir qiladi.

Dinamik qaror tarmoqlari

Markov qaror jarayoni holat asosidagi ifodadir. Klassik rejalashtirishda xususiyatlar asosida mulohaza qilish to'g'ri ifodalar va samarali algoritmlarga imkon yaratishi kabi, noaniqlik ostidagi rejalashtirish ham xususiyatlar asosida mulohaza qilishdan foydalanishi mumkin. Bu **qarorlarni nazariy rejalashtirishning** asosini tashkil qiladi.

Dinamik qaror tarmog'i (DDN) bir necha turli usulda ko'rish mumkin:

- MDPlarning faktorlangan ifodasi, bu yerda holatlar xususiyatlar bilan tavsiflanadi;

- qaror qabul qilish tarmoqlarini kengaytirish, noaniq yoki cheksiz gorizontga ega bo'lgan vazifalar uchun takrorlanadigan tuzilmani yaratishga imkon beradi;

- dinamik ishonch tarmoqlarining kengaytmasi, harakatlar va mukofotlarni o'z ichiga oladi;

- faoliyat natijalaridagi mukofotlar va noaniqliklarni hisobga olish uchun funksional harakatlar ko'rinishini yoki CSP rejalashtirish ko'rinishini kengaytirish.

Dinamik qaror tarmog'i quyidagilardan iborat:

- holat xususiyatlari to'plami

- mumkin bo'lgan harakatlar to'plami

- har bir F xususiyati uchun F_0 va F_1 tasodifiy tugunlari (mos ravishda 0 va 1 vaqtidagi xususiyatlar uchun) va qaror tugunlari A_0 bo'lgan ikki bosqichli qarorlar tarmog'i, shunday qilib

- A_0 ning domeni barcha harakatlar to'plami

- A_0 ning asl to'plamlari 0 vaqtidagi xususiyatlar to'plamidir (bu yo'ylar odatda aniq ko'rsatilmaydi)

- 0-vaqt xususiyatlarining asl to'plamlari A_0 yoki 1-vaqt xususiyatlarini o'z ichiga olmaydi, ammo:

- boshqa 0-vaqt xususiyatlarini o'z ichiga olishi mumkin, agar natijada hosil bo'lgan tarmoq asiklik bo'lsa

- 1-vaqt xususiyatlarining asl to'plamlari A_0 va boshqa 0 yoki 1-vaqt xususiyatlarini o'z ichiga olishi mumkin, agar graf asiklik bo'lsa

- har bir F xususiyati uchun $P(F_0 \mid \text{parents}(F_0))$ and $P(F_1 \mid \text{parents}(F_1))$ uchun ehtimollik taqsimotlari mavjud

- mukofot funksiyasi harakat va 0 yoki 1 vaqtidagi xususiyatlarning har qanday qismiga bog'liq bo'lishi mumkin.

Dinamik ishonch tarmog'idagi kabi, dinamik qaror tarmog'i keyingi vaqt uchun xususiyatlar va harakatlarni takrorlash orqali qaror tarmog'iga **aylantirilishi** mumkin. Agar vaqt gorizonti n bo'lsa, har bir i vaqt uchun $0 \leq i \leq n$ bo'lgan har bir xususiyat F uchun F_i o'zgaruvchisi mavjud. Agar vaqt gorizonti n bo'lsa, har bir vaqt uchun $0 \leq i < n$ bo'lgan A_i o'zgaruvchisi mavjud.

Gorizont n cheksiz bo'lishi mumkin, bu bizga to'xtamaydigan jarayonlarni modellashtirish imkonini beradi.

Shunday qilib, agar vaqt gorizonti n bo'lsa va k ta xususiyat mavjud bo'lsa, yoyilgan tarmoqda $k * (n + 1)$ tasodifiy tugunlar (har biri tasodifiy o'zgaruvchini ifodalaydi) va k qaror tugunlari mavjud.

Aining asl to'plamlari tasodifiy o'zgaruvchilar F_i (shuning uchun agent holatni kuzatishi mumkin). Har bir F_{i+1} harakat A_i va i va $i + 1$ vaqtlaridagi xususiyatlarga bog'liq, xuddi F_i harakati bilan bir xil shartli ehtimolliklar bilan A_0 va 0 va 1 vaqtlaridagi xususiyatlarga bog'liq bo'lgani kabi. F_0 o'zgaruvchilari ikki bosqichli qaror tarmog'ida bevosita modellashtiriladi.

12.36-misol. 6.1-misol oddiy muhitda to'rta joyda qahva va pochta yetkazib beradigan robotni modellashtiradi. 6.1-misolning stoxastik versiyasini dinamik qaror tarmog'i sifatida ifodalashni ko'rib chiqing. Ushbu misoldagi bir xil xususiyatlardan foydalanamiz.

$RLoc$ xususiyati robotning joylashuvini modellashtiradi. $RLoc_i$ o'zgaruvchilarning asl to'plamlari $RLoc_0$ va A_i .

RHC xususiyati robotda qahva borligini bildiradi. RHC_i parents (asl to'plamlari) RHC_0 , A_0 va $RLoc_0$; robotda qahva bor yoki yo'qligi, unda avval qahva bo'lganmi-yo'qligiga, uning harakati va joylashuviga bog'liq. Ehtimollar robotning qahvani ko'tara olmasligi yoki yetkazib bera olmasligi, u qahvani tashlab yuborishi yoki kimdir unga qahvani boshqa holatda berishi (bu imkonsiz bo'lmisligi mumkin) ehtimolini kodlashi mumkin.

SWC o'zgaruvchisi Sem qahva xohlashini bildiradi. SWC_i asl to'plamlari SWC_0 , RHC_0 , A_0 va $RLoc_0$, RHC_i va SWC_i mustaqil bo'lishini kutmaysiz, chunki ular ikkalasi ham qahva muvaffaqiyatli yetkazib berilganiga bog'liq. Bu birining asl to'plamlari bo'lishi bilan modellashtirilishi mumkin.

1 vaqtdagi holat o'zgaruvchilarining harakat va boshqa holat o'zgaruvchilariga qanday bog'liqligini ko'rsatadigan ikki bosqichli ishonch tarmog'i 12.20-rasmda ko'rsatilgan. Ushbu rasm shuningdek, harakat, Sem qahvani xohlamasligi va pochta kutayotganini hisobga olgan holda mukofotni ham ko'rsatadi.

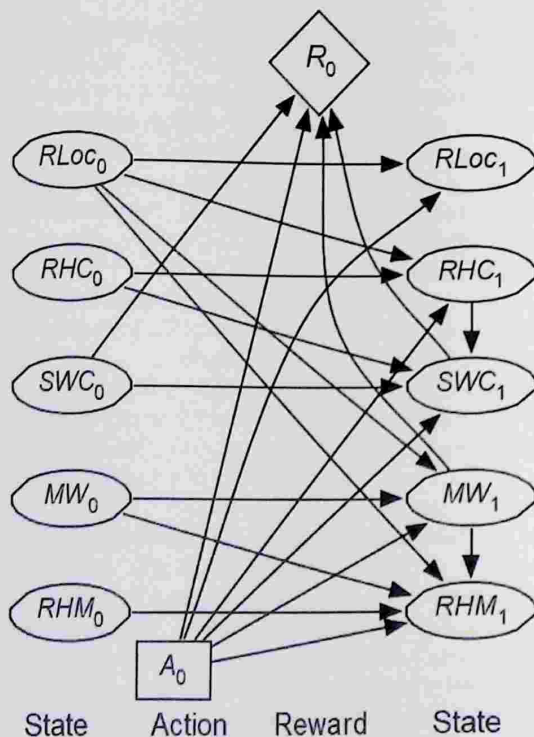
12.21-rasmda 3 gorizont uchun kengaytirilgan qarorlar tarmog'i ko'rsatilgan.

12.37-misol. RHC_i va SWC_i o'rtasidagi bog'liqlikni modellashtirishning muqobil usuli CSD_i deb nomlangan yangi o'zgaruvchini joriy etishdir, bu 1-vaqtda qahva muvaffaqiyatli yetkazib berilganligini bildiradi. Ushbu o'zgaruvchi RHC_i va SWC_i ning asl to'plamlari hisoblanadi. Sem qahvani xohlaydimi yoki yo'qmi, bu Sem ilgari qahvani xohlagan-xohlamaganligi va qahva muvaffaqiyatli yetkazib berilganiga bog'liq. Robotning qahvani olib ketishini modellashtirish, robot qahvaga ega yoki yo'qligi, harakati va joylashuviga bog'liq, Xuddi shunday, MW_i va RHM_i o'rtasidagi bog'liqlik MPU_i deb nomlangan o'zgaruvchini kiritish orqali modellashtirilishi mumkin, bu pochta muvaffaqiyatli olib ketilganligini bildiradi.

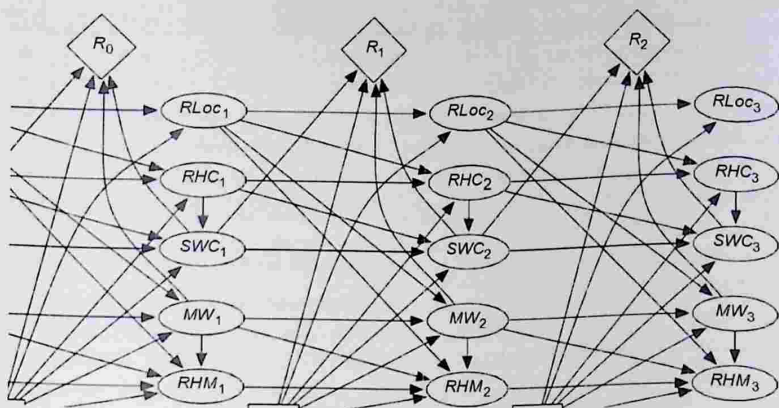
Natijada *DDN* 2-gorizontga yoyilgan, ammo mukofotni o'z ichiga olmagan holda, 12.22-rasmda ko'rsatilgan.

Olingan *DDN* 2-gorizonta kengaytiriladi, lekin 12.22-rasmda ko'rsatilganidek, mukofotni hisobga olmagan holda.

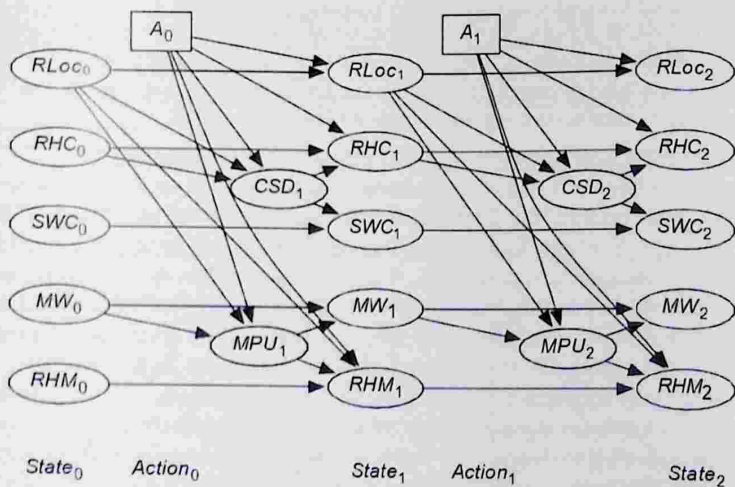
Agar mukofot faqat oxirida berilsa, qaror tarmoqlari uchun o'zgaruvchilarni istisno qilish, 12.14-rasmda ko'rsatilgandek, to'g'ridan to'g'ri qo'llanilishi mumkin. Qaror tarmoqlari uchun o'zgaruvchilarni istisno qilish qiymat takrorlanishiga mos keladi. To'liq kuzatiladigan qaror tarmoqlarida o'zgaruvchilarni istisno qilish no forgetting (unutmaydigan) shartini talab qilmasligini unutmang. Agent holatni bilgach, barcha oldingi qarorlar muhim emas. Agar mukofotlar har bir vaqt oralig'ida yig'ilsa, algoritim mukofotlarni qo'shish (va chegirmalash) imkonini berish uchun qo'shilishi kerak (12.19-mashqqa qarang).



12.20-rasm. Dinamik qarorlarni qabul qilishning ikki bosqichli tarmog'i



12.21-rasm. 3-gorizontda joylashtirilgan dinamik qaror qabul qilish tarmog'i



12.22-rasm. Mukofot tugunlarini hisobga olmagan holda, 2 ga teng gorizont uchun oraliq o'zgaruvchilar bilan dinamik qaror qabul qilish tarmog'i

Qisman kuzatiladigan qaror qabul qilish jarayonlari

Qisman kuzatiladigan Markov qaror qabul qilish jarayoni (POMDP) MDP va yashirin Markov modeli (HMM) kombinatsiyasidir. MDPdagi holat to'liq kuzatiladigan deb hisoblanadi, MPDdagi holat to'liq kuzatilishi mumkin deb hisoblansa-da, POMDPdagi muhit holati qisman kuzatilishi mumkin, ya'ni agent harakat qilishdan oldin atrof-muhitning qisman va/yoki shovqinli kuzatuvlarini oladi.

POMDP 12.23-rasmining qaror tarmog'ining cheksiz kengaytmasi sifatida ifodalanishi mumkin, bu agentning **ishonch holatini** aniq ko'rsatadi. Tarmoq cheksiz o'ng tomonga kengayadi.

POMDP quyidagi o'zgaruvchilar va omillardan iborat, bu agentning tashqi xatti-harakatlarini belgilaydi:

S , dunyo holatlarining to'plami

A , harakatlar to'plami

O , mumkin bo'lgan kuzatuvlar to'plami

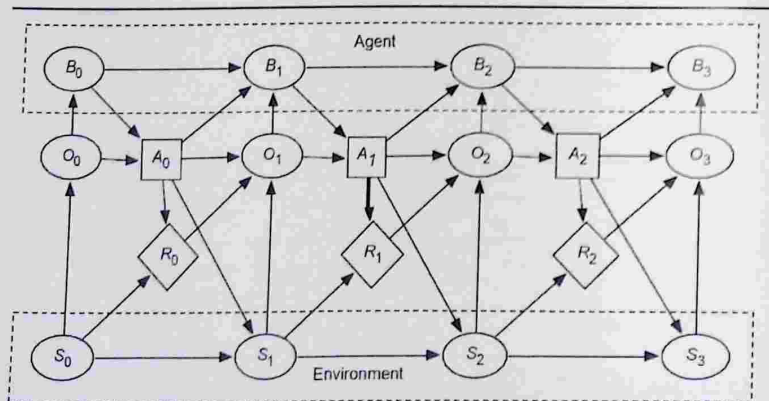
$P(S_0)$, boshlang'ich holatning ehtimollik taqsimoti

$P(S' | S, A)$, atrof-muhit **dinamikasi** S holatdan A harakatni bajarish orqali S' holatga kelish ehtimoli

$R(S, A)$, S holatida boshlab A harakatini bajarishning kutilgan **mukofoti**

$P(O | S, A, R)$, holat S , oldingi harakat A va mukofot R bo'lsa, O kuzatilish ehtimoli.

$P(S' | S, A)$ va $R(S, A)$, MDPdagi bilan bir xil. S_i dan O_i ga yoy agent kuzatuvlari holatga bog'liqligini bildiradi. $A_i - 1$ dan O_i ga yoy agentning harakatlari atrof-muhitga ta'sir qilmaydi, lekin uning kuzatuvlariga ta'sir qilishiga imkon beradi, masalan, atrof-muhitning bir qismiga e'tibor berish orqali.



12.23-rasm. POMDP aniq ishonch holatlari bilan

B_{i-1} dan O_i ga bo'lgan yoy agentning kuzatuvi mukofotga bog'liq bo'lishi mumkinligini ko'rsatadi; agent mukofotni to'g'ridan to'g'ri kuzatishi kerak deb hisoblanmaydi, chunki ba'zan mukofotlar faqat retrospektiv tarzda ko'rinadi. Mukofotni kuzatish ko'pincha agent qaysi holatni haqiqatda kuzata olmasligini ko'rsatishi mumkin.

Ichki jihatdan, agentda quyidagilar mavjud:

B_i , mumkin bo'lgan ishonch holatlari to'plami

$T(B, A, O)$, **ishonch holatiga o'tish funksiyasi**, bu agentning yangi ishonch holatini, avvalgi ishonch holati B , agent bajargan harakat A va kuzatuv O ni bergan holda belgilaydi; i bosqichdagi ishonch holati

$$B_i = T(B_{i-1}, A_{i-1}, O_i)$$

$\pi(A_i, B_i, O_i)$ bu **buyruq funksiyasi** yoki **tamoyil**, bu ishonch holati va kuzatuvning funksiyasi sifatida agent nima qilishini aniqlovchi **shartli rejani** belgilaydi.

Tamoyil tadqiqotga ruxsat berish yoki boshqa agentlarni chalkashtirish uchun stoxastik bo'lishi mumkin. Ishonch holati o'tish funksiyasi odatda deterministik bo'lib, harakat va yangi kuzatuvlar asosida yangilanadigan ehtimollik taqsimotlarini ifodalaydi. POMDP (Qisman kuzatiladigan Markov qaror qabul qilish jarayoni) da rejalashtirish ishonch holati o'tish funksiyasi va buyruq funksiyasini yaratishni o'z ichiga oladi. B_i o'zgaruvchilari maxsusdir, chunki dunyo bu o'zgaruvchilarning domenini yoki tuzilishini belgilamaydi; agent yoki uning dizayneri ishonch holatining tuzilishini tanlash va agent ishonch holati, oldingi harakati va so'nggi kuzatuvlari asosida qanday harakat qilishini aniqlash imkoniyatiga ega. Ishonch holati agentning o'z tarixini eslab qolgan hamma narsani kodlaydi. Ishonch holatini ifodalash va optimal tamoyilni topishning bir qancha usullari mavjud:

12.23-rasmdagi qarorlar tarmog'i 12.14-rasmda ko'rsatilgan, chegirmali mukofotlarni o'z ichiga olgan holda, qaror tarmoqlari uchun o'zgaruvchan istisno qilish yordamida hal qilinishi mumkin. Unutmaslik yoylarini qo'shish ishonch holati kuzatuvlar va harakatlar ketma-ketligi ekanligiga tengdir; B_i - bu B_{i-1} ketma-ketligiga A_{i-1} va O_i ning qo'shilishi. Bu yondashuvning muammosi shundaki, tarix cheksizdir va tamoyilning hajmi tarixning uzunligida eksponent hisoblanadi. Bu faqat tarix qisqa bo'lganda yoki ataylab qisqartirilganda amaliy hisoblanadi.

Ishonch holati muhit holatlarining ehtimollik taqsimoti bo'lishi mumkin. Ishonch holatini saqlab qolish keyin yashirin Markov modeli bilan bog'liq bo'lgan filtratsiya muammosi bo'ladi. Bu yondashuvning muammosi shundaki, n dunyo holatlari bilan, ishonch holatlari to'plami $(n-1)$ -o'lchovli haqiqiy makondir. Keyin agent bu ko'p o'lchovli haqiqiy makonni harakatlarga o'zgartirishning optimal funksiyasini topishi kerak. Harakatlar ketma-ketligining qiymati faqat dunyo holatlariga bog'liq; ishonch holati dunyo holatlarining ehtimollik taqsimoti bo'lgani uchun, ishonch holatining kutilgan qiymati dunyo holatlari qiymatlarining chiziqli funksiyasi hisoblanadi.

R^{n-1} fazodagi barcha nuqtalar ishonch holatlari bo'lishi mumkin emas, balki faqat kuzatuvlar ketma-ketligidan kelib chiqqan nuqtalar ishonch holatlari sifatida qaraladi. Tamoyillar **qaror daraxti** sifatida ifodalanishi mumkin, bunda shartlar kuzatuvlarning funksiyalari bo'ladi. Kuzatuvlar agent qaysi yo'lni tanlashini belgilaydi. Har qanday cheklangan oldindan ko'rish uchun eng yaxshi tamoyilni tanlayotgan agent uchun qiymat funksiyasi bo'lak chiziqli va konveksdir. Bu umumiy R^{n-1} funksiyasidan oddiyroq bo'lsa-da, shartli tamoyillarning soni O kabi o'sadi, bu yerda O kuzatuvlar to'plami t esa bosqichdir.

Umuman olganda, ishonch holati har qanday bo'lishi mumkin. Ishonch holati ba'zi o'zgaruvchilar ustidan ehtimollik taqsimotlarini, eslab qolingan kuzatuvlar va harakatlarni o'z ichiga olishi mumkin. Agent eng yaxshi boshqaruvchini topish uchun boshqaruvchilar makonini qidirishi mumkin. Shunday qilib, agent nimani eslab qolish (ishonch holati) va ishonch holati asosida nima qilish kerakligini qidiradi. Birinchi ikkita taklif ushbu yondashuvning misollari ekanligini e'tiborga oling: agentning ishonch holati uning barcha tarixi yoki agentning ishonch holati dunyo holatlarining ehtimollik taqsimoti bo'lib, ikkalasi ham boshqarib bo'lmaydigan hisoblanadi. Umumiy holat nima eslab qolish kerakligi bo'yicha cheklanmaganligi sababli, qidiruv makoni ulkan.

12.6. Ijtimoiy ta'sir

Ochiq va mas'uliyatli qarorlar qabul qilish foydaliliklarni aniq belgilanishini talab qiladi, shunda ular tekshirish uchun ochiq bo'ladi va odamlar ularning maqsadga muvofiq yoki mos emasligini muhokama qilishlari mumkin. Qaror taklif qilinganda, sezgirlik tahlili – qaror foydaliliklar o'zgarishi bilan qanday o'zgarishini o'rganish – foydalilikning qanchalik muhimligini aniqlashi mumkin. Foydaliliklarni aniq va ochiq qilish, ayniqsa, ko'p odamlarga ta'sir qiladigan jamoat sohasidagi qarorlar uchun juda muhim, lekin ko'pincha munozarali bo'ladi, chunki odamlar umumiy foydalilik funksiyaga ega emaslar.

Tamoyil kerakli xususiyatlarga ega bo'lishi uchun foydaliliklar va mukofotlarni loyihalash utility engineering (**foydalilik muhandisligi**) yoki reward engineering (**mukofot muhandisligi**) deb ataladi. Bu, ayniqsa, ijtimoiy-iqtisodiy ahvol, o'qituvchilar samaradorligi va qashoqlikni kamaytirish, ta'lim yoki jinoyatchilikka oid qarorlar qabul qilishda takroran jinoyat sodir etish xavfi kabi kuzatilmaydigan ko'rsatkichlar uchun qiyin. Bularni to'g'ridan to'g'ri o'lchash mumkin emas, balki o'lim darajasi, talabalar baholari yoki qayta hibsga olish kabi kuzatiladigan xususiyatlarning **o'lchash modelidan** xulosa chiqarish kerak. Biroq, o'lchash modelini optimallashtirayotgan agent aslida kerakli narsani optimallashtirmasligi mumkin. Jacobs va Wallach [2021] o'lchash modellarning **adolat** bilan o'zaro ta'sirini tahlil qiladilar.

Foydaliliklar ochiq ishlatilgan holatlardan biri sog'liqni saqlashda resurslarni taqsimlashdir, ayniqsa taqsimlash to'lov qobiliyati emas, ehtiyoj asosida bo'lgan yurisdiksiyalardir. Bir qator mamlakatlar (qimmat) dori-darmonlar yoki jarrohlik amaliyotlari kabi tibbiy aralashuvlarni baholash uchun foydali o'lchov sifatida **sifatga moslashtirilgan hayot yili (QALY)**dan foydalanadi. Har bir mumkin bo'lgan aralashuv uchun bir yil davomida sog'lom yashash uchun 1 va o'lim uchun 0 foydalilikdan foydalanadi. Natijalar eng yuqori va eng past foydaliliklar o'rtasidagi lotereya sifatida baholanadi. Aralashuvning foydaliligi bu qiymatning har bir kutilgan hayot yili bo'yicha yig'indisidir. Agar aralashuv davom etayotgan va kelajakda doimiy bo'lsa, foydalilik umr ko'rish davomiyligi va yillik qiymat bilan hisoblanadi. QALY hayotning miqdori va sifatini o'z ichiga oladigan o'lchovdir. Cheklangan resurslar mavjud bo'lganda, ko'pgina mamlakatlarda qaror qabul qilish uchun narx/QALY nisbati samaradorlik o'lchovi sifatida ishlatiladi.

Jamiyat barchaga ta'sir qiluvchi qarorlarni qabul qilishi kerak. Barchaga mos keladigan foydalilikni topish bahsli. Masalan, ko'pchilik ko'ruvchi odamlar uchun ko'r bo'lib qolish past foydalilikka ega bo'ladi; ular ko'r bo'lib qolishni juda yomon deb bilishadi, chunki ular dunyo bilan o'zaro aloqalarni qayta o'riganishlari kerak bo'ladi. Shunday qilib, bir yil davomida ko'rlikning foydaliligi past bo'ladi. Biroq, bu ko'r odamlar ko'radigan odamlardan kamroq baholanadi degan ma'noni anglatadi, bu nogironlarga nisbatan **kamsituvchi** taxmin. Istiqbollarni nazariyasida bo'lgani kabi, shaxsiy ma'lumot nuqtalarini kiritish bo'yicha takliflar mavjud.

Ba'zan jamiyat hayot-mamot qarorlarini qabul qilishi kerak. Masalan, davlat maktablarini zilzilaga chidamlilikka qancha mablag' sarflash kerakligini ko'rib chiqing. O'quvchilar maktabda bo'lganida kuchli zilzila ko'p o'limlarga olib kelishi mumkin. Muayyan joyda zilzila ehtimolini va o'quvchilar hozir bo'lganida muayyan tuzilmaning qulashi ehtimolini hisoblash mumkin. Qulash ehtimolini kamaytirish uchun pul sarflash mumkin. Pul sarflash yoki sarflamaslikka qaror qilish noaniqlik ostida qaror qabul qilishning klassik misolidir, bu pul bilan bolalar hayotini almashishni talab qiladi. Pul sarflash to'g'risida qaror qabul qilish noaniqlik sharoitida qaror qabul qilishning klassik namunasidir, bu bolalar hayoti uchun pul savdosini talab qiladi. Ko'pgina qaror qabul qiluvchilar ochiqchasiga pul va bolalar hayotini qurbon qilishni xohlamaydilar. Biroq, ular aniq murosaga kelmasa, u bolalar hayotini yetarlicha baholamaslikka moyil bo'ladilar.

AI vositalari keng tarqalgan va ko'proq qarorlar avtomatlashtirilganligi sababli, jamiyat foydaliliklarni belgilash bo'yicha noqulay suhbatlarni qabul qilishi kerak bo'ladi. Bu suhbatlar olib borilmasa, boshqa bironing foydaliliklari AI vositalariga o'rnatilganligini anglatadi.

12.7. Takrorlash

Foydalilik ehtimollik bilan birlashadigan ustunlik o'lchovi hisoblanadi.

Qarorlar tarmog'i cheklangan bosqichli, qisman kuzatiladigan ketma-ket qarorlar muammosini xususiyatlar orqali ifodalashi mumkin.

MDP cheksiz bosqichli yoki noma'lum bosqichli ketma-ket qarorlar muammosini holatlar orqali ifodalashi mumkin.

To'liq kuzatiladigan MDP qiymatni takrorlash yoki tamoyil iteratsiyasi yordamida yechilishi mumkin.

Dinamik qarorlar tarmog'i MDPni xususiyatlar orqali ifodalashga imkon beradi.

Davlat sohasida foydalilikka asoslangan qarorlar qabul qilish foydalilik o'lchovlari ochiq qilinganda yanada oshkora va hisobdor bo'lishi mumkin.

12.8. Qo'shimcha o'qish uchun manbalar

Bu yerda keltirilgan foydalilik nazariyasi Neumann va Morgenstern [1953] tomonidan ishlab chiqilgan va Savage [1972] tomonidan rivojlantirilgan. Keeney va Raiffa [1976] ko'p atributli (xususiyatlarga asoslangan) foydalilik funksiyalariga e'tibor qaratib, foydalilik nazariyasini muhokama qiladilar. Chegirmalash uchun aksiomalar Koopmans [1972] tomonidan keltirilgan; Bleichrodt va boshqalar. [2008] debugged versiyasi va dalilini taqdim etadi. Foydalilik va afzalliklarning grafik modellari o'rganish uchun Bacchus va Grove [1995] va Boutilier va boshqalar [2004]ni ko'ring. Walsh [2007] va Rossi va boshqalar [2011] Sida afzalliklardan foydalanishni umumlashtiradi.

Kahneman [2011] noaniqlik sharoitida odamlarning qanday qarorlar qabul qilishini tushuntiradigan psixologiyani muhokama qiladi va istiqbollar nazariyasini rag'batlantiradi. Wakker [2010] foydalilik va istiqbollar nazariyalarining darslik ko'rinishidagi umumlashtirishini taqdim etadi.

Qaror tarmoqlari yoki ta'sir diagrammalari Howard va Matheson [1984] tomonidan ixtiro qilingan. Ta'sir diagrammalarini yechish uchun dinamik dasturlashdan foydalanish usulini Shachter va Peot [1992] da topish mumkin. Ma'lumot va boshqaruv qiymati Matheson [1990] tomonidan muhokama qilingan.

MDPlar Bellman [1957] tomonidan ixtiro qilingan va Puterman [1994] va Bertsekas [2017] tomonidan muhokama qilingan. Mausam va Kolobov [2012] AIDA MDPlarni umumlashtiradi. Boutilier va boshqalar [1999] MDPlarni xususiyatlarga o'tkazishni, qaror nazariyasiga asoslangan rejalashtirish sifatida tanishtiradi.

Kochenderfer va boshqalar [2022] noaniqlik sharoitida rejalashtirishni tanishtiradi. Kochenderfer [2015] ko'plab real dunyo amaliyot holatlarini taqdim etadi.

Lehman va boshqalar [2018] mukofot funksiyalarining noto'g'ri aniqlanishining ta'siriga misollar keltiradi.

Sifatga moslashtirilgan hayot yili (QALY) Torrance [1970] va Fanshel va Bush [1970] tomonidan taqdim etilgan. Spencer va boshqalar [2022] QALY tarixini ko'plab manbalar bilan ko'rib chiqadi.

12.9. Mashqlar

12.1-mashq. To'liqlik va/yoki o'tkazuvchanlik aksiomalari quyidagi bayonotlarni isbotlayotganligini isbotlang. Sizning isbotlaringiz qaysi aksiomalar asosida bo'lishiga bog'liq.

$o_2 \geq o_1$ is equivalent to $o_1 > o_2$
if $o_1 > o_2$ and $o_2 > o_3$ then $o_1 > o_3$
if $o_1 > o_2$ and $o_2 \geq o_3$ then $o_1 > o_3$
if $o_1 \geq o_2$ and $o_2 \geq o_3$ then $o_1 \geq o_3$.

12.2-mashq. Quyidagi ikki muqobilni ko'rib chiqing:

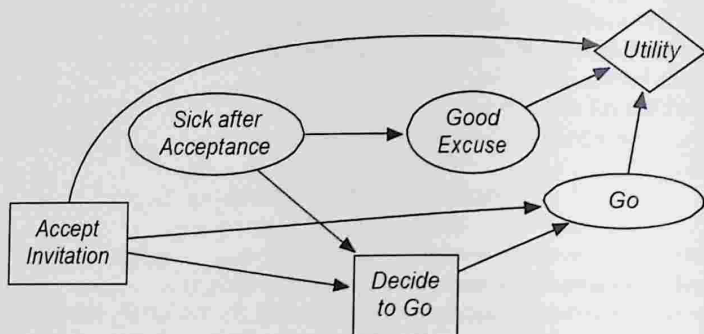
Sizda mavjud bo'lganiga qo'shimcha ravishda \$1000 berildi. Sizdan quyidagi variantlardan birini tanlash so'raladi: 50% ehtimollik bilan \$1000 yutish yoki aniq \$500 olish.

Sizda mavjud bo'lganiga qo'shimcha ravishda \$2000 berildi. Sizdan quyidagi variantlardan birini tanlash so'raladi: 50% ehtimollik bilan \$1000 yo'qotish yoki aniq \$500 yo'qotish. Ushbu muqobillar uchun foydalilik nazariyasi va istiqbollar nazariyasining bashoratlari qanday farq qilishini tushuntiring.

12.3-mashq. Hayotda qabul qilishimiz kerak bo'lgan qarorlardan biri biz tadbirga borishimiz mumkinligiga ishonchimiz komil bo'lmasa ham, taklifni qabul qilish-qilmaslikdir. 12.24-rasm bunday muammo uchun qaror tarmog'ini beradi. Barcha qaror va tasodifiy o'zgaruvchilar Boolean (ya'ni $\{true, false\}$ domeniga ega) deb faraz qiling. Siz taklifni qabul qilishingiz mumkin, lekin vaqt kelganda, siz baribir borishni yoki bormaslikni hal qilishingiz kerak bo'ladi. Siz taklifni qabul qilish va borishga qaror qilish o'rtasida kasal bo'lishingiz mumkin. Agar siz borishga qaror qilsangiz ham, taklifni qabul qilmagan bo'lsangiz, bora olmasligingiz mumkin. Agar kasal bo'lsangiz, bormaslik uchun yaxshi bahonangiz bor. Sizning foydaliligingiz roziligingiz, uzringiz bor-yo'qligingiz va haqiqatda borishingiz bilan bog'liq.

(a) Mumkin bo'lgan foydalilik funksiyasini ifodalovchi jadvalni taqdim eting. Yagona eng yaxshi natija siz taklifni qabul qilishingiz, sababli bahonangiz yo'q, lekin baribir borishingizdir. Yagona eng yomon natija siz taklifni qabul qilishingiz, sababli bahonangiz yo'q va bormasligingizdir. Boshqa foydalilik qiymatlarini asoslang.

(b) Taklifni qabul qilishdan oldin kasal yoki sog'lom ekanligingizni kuza-tishingiz mumkinligini faraz qiling. Bu taklifni qabul qilishdan keyin kasal yoki sog'lom ekanligingizdan boshqa o'zgaruvchi ekanligini unutmang. Ushbu vazi-yatni modellashtirish uchun tarmoqqa qo'shing. Foydalilik funksiyasini o'zgarti-rishingiz shart emas, lekin yangi kuzatish ijobiy ma'lumot qiymatiga ega bo'lishi kerak. Olingan tarmoq unutilmaydigan bo'lishi kerak.



12.24-rasm. Taklif to'g'risida qaror qabul qilish tarmog'i

(c) Tasavvur qiling, dastlabki taklifni qabul qilishga qaror qilganingizdan keyin va borishga qaror qilishdan oldin, siz yaxshiroq taklifni olganingizni bilib qolishingiz mumkin (bu dastlabki tadbir bilan to'qnashadigan voqea, shuning uchun ikkala tadbirga ham bora olmaysiz). Faraz qilaylik, siz keyingi taklifni dastlabki tadbirdan afzal ko'rasiz. (Qiyin qaror, dastlabki taklifni qabul qilish yoki yaxshiroq taklifni kutishdir, siz uni olmasligingiz ham mumkin.) Afsuski, boshqa taklifga ega bo'lish sababli bahona bermaydi. Tarmoqda «yaxshiroq taklif» tugunini va ushbu vaziyatni modellashtirish uchun barcha tegishli yoylarni qo'shing. [Siz (b) qismdagi tugun va yoylarni qo'shishingiz shart emas.]

(d) Agar sizda «yaxshiroq taklif» va «taklifni qabul qilish» orasida yoy bo'lsa, nima uchunligini tushuntiring (ya'ni bu yoyni mos qilish uchun dunyo qanday bo'lishi kerak). Agar sizda bunday yoy bo'lmasa, u hali ham oldingi hikoyaga mos kelishi uchun qaysi yo'nalishda ketishi mumkinligini; bu yoyni mos qilish uchun dunyoda nima bo'lishi kerakligini tushuntiring.

(e) Agar «yaxshiroq taklif» va «taklifni qabul qilish» orasida yoy bo'lmasa (siz shunday yoy chizgan bo'lsangiz ham), bu yoyni mos qilish uchun dunyoda nima to'g'ri bo'lishi kerakligini tushuntiring.

12.4-mashq. Talabalar har bir kurs uchun qancha o'qish kerakligini hal qilishlari kerak. Ushbu savolning maqsadi qaror tarmoqlaridan foydalanib, ularga bunday qarorlar qabul qilishda yordam berishdir.

Faraz qilaylik, talabalar avval oraliq nazorat uchun qancha o'qish kerakligini hal qilishlari kerak. Ular ko'p o'qishlari, oz o'qishlari yoki umuman o'qimasliklari mumkin. Oraliq nazoratdan o'tishlari kursning qanchalik qiyinligi va qancha o'qishlariga bog'liq. Taxminan, agar ular ko'p o'qisalar yoki kurs oson bo'lsa va ular biroz o'qishsa, o'tishadi. Oraliq nazorat bahosini olgandan so'ng, ular yakuniy imtihoniga qancha o'qish kerakligini hal qilishlari kerak. Yakuniy imtihon natijasi qancha o'qishlari va kursning qanchalik qiyinligiga bog'liq. Ularning yakuniy bahosi (A, B, C yoki F) qaysi imtihonlarni topshirganliklariga bog'liq; umuman olganda, agar ular ikkala imtihonni ham topshirsa, A olishadi, faqat yakuniy imtihonni topshirsa, B, faqat oraliq nazoratni topshirsa, C yoki ikkala imtihonni ham topshirmasa, F olishadi. Albatta, bu umumiy taxminlarda juda ko'p shovqin mavjud. Ularning foydaliligi ularning subyektiv umumiy sa'y-harakatlariga va yakuniy bahosiga bog'liq. Faraz qilaylik, ularning subyektiv umumiy sa'y-harakatlari (ko'p yoki oz) oraliq nazorat va yakuniy imtihon uchun o'qishlari bilan bog'liq.

(a) Ushbu hikoyaga asoslangan talaba qarorini qabul qilish uchun qaror tarmog'ini chizing.

(b) Har bir o'zgaruvchining domeni nima?

(c) Mos shartli ehtimollik jadvallarini keltiring.

(d) Eng yaxshi natija nima (unga 100 foydalilik bering) va eng yomon natija nima (unga 0 foydalilik bering)?

(e) Dangasa va shunchaki o'tishni xohlaydigan (F olmaslikni xohlaydigan) talaba uchun mos foydalilik funksiyasini keltiring. Umumiy sa'y-harakat bu yerda ular (o'ylashicha) ko'p yoki oz ishlaganini o'lchaydi. Eng yaxshi natija va eng yomon natijani tushuntiring. 12.4-jadvalning nusxasini to'ldiring; eng yaxshi natija uchun 100 va eng yomon natija uchun 0 dan foydalaning.

(f) Oldingi qism uchun foydalilik funksiyangizni hisobga olgan holda, yuqorida keltirilgan foydalilik funksiyasini aks ettiruvchi bir misol uchun yo'qolgan atamalar uchun qiymatlarni bering:

Natijani va lotereyani solishtirish [p : _____, $1-p$: _____] bo'lganda

p = _____ bo'lganda, natija lotereyadan ustun ko'riladi.

p = _____ bo'lganda, lotereya natijadan ustun ko'riladi.

(g) Ko'p ishlashdan bezovta bo'lmaydigan va A olishni chin dildan xohlaydigan, B yoki undan past bahodan juda hafsalasi pir bo'ladigan talaba uchun mos foydalilik funksiyasini keltiring. Eng yaxshi natija va eng yomon natijani tushuntiring. 12.4-jadvalning nusxasini to'ldiring; eng yaxshi natija uchun 100 va eng yomon natija uchun 0 dan foydalaning.

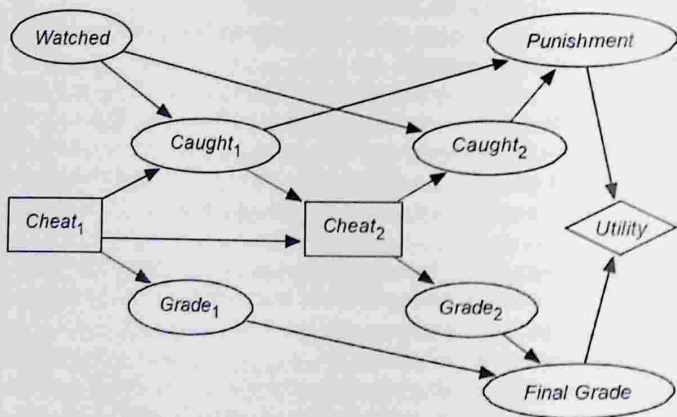
12.5-mashq. Ba'zi talabalar imtihonlarda ko'chirishni tanlaydi va o'qituvchilar ko'chirish foyda keltirmasligiga ishonch hosil qilishni xohlashadi. Ratsional model, ko'chirish qarori xarajatlar va foydalarga bog'liq ekanligini belgilaydi. Bu yerda biz bunday modelni ishlab chiqamiz va tanqid qilamiz.

12.25-rasmdagi qaror tarmog'ini ko'rib chiqing. Bu diagramma talabaning ikkita turli vaqtda ko'chirish yoki ko'chirmaslik qarorlarini modellashtiradi.

12.4-jadval

O'qish haqidagi qarorlar uchun foydalilik funksiyasi

Grade	Total Effort	Utility
A	lot	
A	little	
B	lot	
B	little	
C	lot	
C	little	
F	lot	
F	little	



12.25-rasm. Ko'chirish kerak yoki yo'qligi haqida qaror

Agar talabalar imtihonda ko'chirsalar, ular ko'chirishda ushlanishi, ammo yuqori baholarni ham olishlari mumkin. Jazolar (yoki imtihondan chetlashtirish, ko'chirishni stenogrammada qayd etish yoki hech biri) ular har ikkala imkoniyat-

da ham ushlanishlariga bog'liq. Ularning ushlanishi talabalar kuzatuv ostida yoki yo'qligi va ko'chirish ko'chirmasliklariga bog'liq. Foydalilik ularning yakuniy baholari va jazolariga bog'liq.

AI Python Fraud Decision Network (aipython.org) quyidagi savollarga javob berish uchun ishlatilishi mumkin bo'lgan ehtimolliklarni taqdim etadi.

(a) Optimal tamoyil nima? Optimal tamoyilning ingliz tilidagi tavsifini bering. (Tavsif SI yoki qaror nazariyasi jargonini ishlatmasligi kerak.) Optimal tamoyilning qiymati nima?

(b) Kuzatilish ehtimoli oshganda optimal tamoyilga nima bo'ladi? [«Watched» ehtimolini o'zgartiring.] Bir qator qiymatlarni sinab ko'ring. Nima sodir bo'layotganini va nima uchunligini tushuntiring.

(c) Ko'chirish uchun mukofotlar kamayganda optimal tamoyil nima bo'ladi? Bir qator turli parametrlarni sinab ko'ring.

(d) Talabalar ko'chirishda ushlangandan keyin ehtiyotkorlik bilan kuzatilishi uchun modelni o'zgartiring. [Maslahat: Ular birinchi imkoniyatda kuzatiladimi yoki yo'qmi, ikkinchi imkoniyatda kuzatilishidan farqli o'zgaruvchidir.] Natijadagi modelni (tuzilma va har qanday yangi parametrlar) ko'rsating va parametrlarning turli o'rnatishlari uchun tamoyillar va kutilgan foydaliliklarni bering.

(e) Hozirgi model ko'chirish kelajakdagi hisob-darsliklarga qanday ta'sir qilishi haqida nimani anglatadi? Modelni shunday o'zgartiringki, ko'chirish keyingi baholarga ta'sir qiladi. Yangi model buni qanday amalga oshirishini tushuntiring.

(f) Bu model qanday qilib yanada realistik bo'lishi mumkin (lekin hali ham oddiy bo'lishi kerak)? [Masalan: ehtimolliklar asoslimi? Foydaliliklar asoslimi? Tuzilma asoslimi?]

(g) Tasavvur qiling, universitet o'qituvchilar ko'chirishni faol tekshirmaydigan, lekin ko'chirish birinchi marta aniqlanganda qattiq jazo qo'llanilishiga asoslangan tizimni o'rnatishga qaror qildi. Bu qanday modellashtirilishi mumkin? Bu uchun modelni aniqlang va qanday qaror optimal ekanligini tushuntiring (bir nechta turli parametr sozlamalari uchun).

(h) Talabalar va o'qituvchilar ko'chirish muammosini o'yinda ratsional qaror sifatida ko'rib chiqishga rag'batlantirilishi kerakmi? Nima uchun kerak yoki kerak emasligini bitta paragrafda tushuntiring.

12.6-mashq. Faraz qilaylik, qaror tarmog'ida *Run* qaror o'zgaruvchisi *Look* va *See* dan asl to'plamlarga ega. VE yordamida optimal tamoyilni topmoqchi bo'lsangiz va barcha boshqa o'zgaruvchilarni istisno qilgandan keyin bitta omil bilan qolsangiz:

<i>Look</i>	<i>See</i>	<i>Run</i>	<i>Value</i>
<i>true</i>	<i>true</i>	<i>yes</i>	23
<i>true</i>	<i>true</i>	<i>no</i>	8
<i>true</i>	<i>false</i>	<i>yes</i>	37
<i>true</i>	<i>false</i>	<i>no</i>	56
<i>false</i>	<i>true</i>	<i>yes</i>	28
<i>false</i>	<i>true</i>	<i>no</i>	12
<i>false</i>	<i>false</i>	<i>yes</i>	18
<i>false</i>	<i>false</i>	<i>no</i>	22

(a) *Run* ni yo'q qilish natijasida hosil bo'lgan omil nima? [Ishora: Siz *Run* ni umumlashtirmaysiz, chunki bu qaror o'zgaruvchisi.]

(b) *Run* uchun optimal qaror funksiyasi nima?

(c) *See*, *Running* asl to'plamlari bo'lgan qaror tarmog'ida *Look* haqida ma'lumotning qiymati qanday? Ya'ni agar agent *See* haqida ma'lumotga ega bo'lsa, *Look* haqida ma'lumot qancha ko'proq qadrlil bo'ladi?

12.7-mashq. Faraz qilaylik, qaror tarmog'ida «contaminated specimen» («ifloslangan namuna») va «ijobiy test» tasodifiy o'zgaruvchilardan «namunani tashlab yuborish» qaror o'zgaruvchisiga yoyilar bor edi. Siz qaror tarmog'ini hal qildingiz va yagona optimal tamoyil mavjudligini aniqladingiz:

Contaminated Specimen	Positive Test	Discard Sample
<i>true</i>	<i>true</i>	<i>yes</i>
<i>true</i>	<i>false</i>	<i>no</i>
<i>false</i>	<i>true</i>	<i>yes</i>
<i>false</i>	<i>false</i>	<i>no</i>

Bu holatda ma'lumotning qiymati haqida nima deyishingiz mumkin?

12.8-mashq. 12.16-mashqdagi qaror tarmog'ining javoblari ehtimolliklarga qanchalik sezgir? Dasturda turli xil shartli ehtimolliklar bilan sinab ko'ring va bu javoblar natijalariga qanday ta'sir qilishini ko'ring. Optimal tamoyil va optimal tamoyilning kutilayotgan qiymatga sezgirlikini muhokama qiling.

12.9-mashq.

12.16-misolda, agar yong'in sensori 20% soxta musbat darajaga ega bo'lsa

$$P(\text{see smoke}|\text{report} \wedge \neg \text{smoke}) = 0.2$$

va 15% soxta manfiy darajaga ega bo'lsa

$$P(\text{see smoke}|\text{report} \wedge \text{smoke}) = 0.85.$$

hali ham tutun tekshirilishga arziydimi?

12.10-mashq. Bu mashq 12.16-misoldagi qaror tarmog'i uchun o'zgaruvchilarni istisno qilish va shartlashishni taqqoslashdir.

(a) 12.20-misoldagi qidiruv uchun o'zgaruvchilarni tartibga keltirishning *teskari* qidiruvi uchun (ya'ni, *Leaving* dan *Report* gacha) o'zgaruvchilarni istisno qilish ketma-ketligini ko'rsating, 12.22-misolga o'xshash jadvalda (faqat o'zgaruvchilar tartibi o'zgaradi).

(b) 12.22-misoldagi o'zgaruvchilar tartibiga *teskari* bo'lgan bo'linish tartibi uchun har bir bo'linish uchun qanday o'zgaruvchilar baholanishi mumkinligini ko'rsating (12.20-misolga o'xshash, lekin o'zgaruvchilar tartibi boshqacha. Shuningdek, 9.26-misoldagi kabi qanday o'zgaruvchilarni unutish mumkinligini ko'rsating. (c) Rekursiv shartlashuvdagi omillarning baholanishi, o'zgaruvchilar tartiblari bir-biriga *teskari* bo'lganda, o'zgaruvchilarni istisno qilish uchun yaratilgan omillarga qanday bog'liqligi haqida eng aniq ma'lumotni bering.

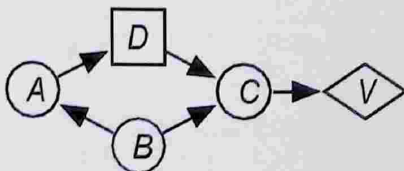
12.11-mashq. 9.10-mashqning ishonch tarmog'ini ko'rib chiqing. Signal kuzatilganda, reaktorni o'chirish yoki o'chirmaslik haqida qaror qabul qilinadi. Reaktorni o'chirish bilan bog'liq xarajat *cs* mavjud (yadro qizib ketganmi yoki yo'qmi, bog'liq emas), holbuki qizib ketgan yadro o'chirilmagan holda *cm* xarajatni olib keladi, bu esa *cs* dan ancha yuqori.

(a) Ushbu qaror muammosini modellastirish uchun qaror tarmog'ini chizing (ya'ni faqat bitta sensor bilan asl tizim uchun).

(b) Aniqlanishi kerak bo'lgan barcha yangi omillar uchun jadvalni ko'rsating (jadvalda tegishli joylarda *cs* va *cm* parametrlarini ishlatishingiz kerak). Foydalilik xarajatlarning manfiyligi deb faraz qiling.

(c) Optimal qarorni topish uchun o'zgaruvchilarni istisno qilish usulini qanday ishlatishni ko'rsating. Har bir o'zgaruvchini istisno qilish uchun qaysi o'zgaruvchini istisno qilish, qanday istisno qilish (umumlashtirish yoki maksimalizatsiya orqali), qaysi omillarni olib tashlash, qanday omil yaratilishi va bu omil qaysi o'zgaruvchilarga bog'liqligi ko'rsatiladi (12.22-misolga o'xshash). Jadvalni berish talab qilinmaydi.

12.12-mashq. Quyidagi qaror tarmog'ini ko'rib chiqing:



(a) Boshlang'ich omillar nima? (Har bir omil doirasidagi o'zgaruvchilarni bering va har bir omilning tegishli ma'nosini ko'rsating.)

(b) Qonuniy bo'linish tartibini va o'zgaruvchilarni baholash tartibini bering (12.20-misolga o'xshash).

(c) Qaror funksiyasini optimallashtirish va kutilgan qiymatni hisoblashda o'zgaruvchilarni istisno qilishda qanday omillar yaratilishini ko'rsating, qonuniy bo'linish tartiblaridan biri uchun. Har bir qadamda qaysi o'zgaruvchini istisno qilish, jamlanishi yoki maksimal darajaga ko'tarilishi, qaysi omillar birlashtirilganligi va qaysi omillar yaratilganligini tushuntiring (jadvallarda emas, ular bog'liq bo'lgan o'zgaruvchilarni ko'rsating).

(d) Aning D qaroridagi ma'lumot qiymati nol bo'lsa, optimal tamoyil qanday ko'rinishga ega? (Har qanday optimal tamoyil haqida eng aniq bayonotni bering.)

12.13-mashq. Asinxron qiymatlarni takrorlash va standart qiymatni takrorlash o'rtasidagi asosiy farq nima? Nega asinxron qiymat iteratsiyasi ko'pincha standart qiymat iteratsiyasidan yaxshiroq ishlaydi?

12.14-mashq. Nima uchun biz ko'pincha MDPlarda kelajakdagi mukofotlarni chegirmalashni ishlatamiz? Agar chegirma koeffitsiyenti 0,9 dan farqli o'laroq 0,6 bo'lsa, agent qanday harakat qiladi?

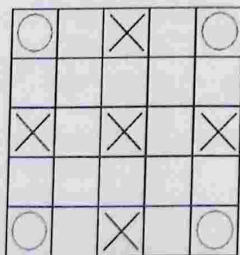
12.15-mashq. 12.31-misolidagi MDPni ko'rib chiqing.

(a) Chegirma miqdori 0 dan 1 gacha bo'lganligi sababli, optimal tamoyil qanday o'zgaradi? Chegirmaning o'zgartirish orqali olinishi mumkin bo'lgan turli xil tamoyillarni beradigan chegirmaga misol keltiring.

(b) Optimal tamoyil sog'lom bo'lganda dam olish va kasal bo'lganda ziyofat qilish bo'lishi uchun MDP va/yoki chegirma qanday o'zgartirilishi mumkin? Bu optimal tamoyilga ega bo'lish uchun imkon qadar kam ehtimolliklar, mukofotlar yoki chegirmalarni o'zgartiradigan MDP ni bering.

(c) 12.33-misolda hisoblangan optimal tamoyil sog'lom bo'lganda ziyofat qilish va kasal bo'lganda dam olish edi. Ushbu tamoyilga amal qilayotgan agent qanday holatlarga keladi? [Ishora: Tamoyil Markov zanjirini hosil qiladi, bu esa statsionar taqsimotga ega.] Ushbu tamoyilning o'rtacha mukofoti qanday? [Ishora: O'rtacha mukofot statsionar taqsimotga nisbatan darhol mukofotlarning kutilayotgan qiymatini hisoblash orqali olinishi mumkin.]

12.16-mashq. O'yin dunyosini ko'rib chiqing.

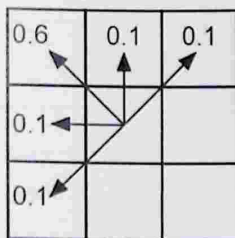


Robot tarmoqdagi 25 joydan birida bo'lishi mumkin. Burchaklardagi doiralarda xazina bo'lishi mumkin. Robot xazina bo'lgan burchakka yetib borganida, u 10 mukofot oladi va xazina yo'qoladi. Xazina yo'q bo'lganda, har bir vaqt bosqichida xazina paydo bo'lish ehtimoli $P1=0.2$ va u har bir burchakka teng ehtimollik bilan paydo bo'ladi. Robot o'zining pozitsiyasini va xazina joylashgan joyini biladi.

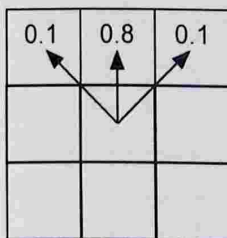
Belgilangana x bo'lgan kvadratchalarda maxluqlar bor. Har bir maxluq har bir vaqt bosqichida tasodifiy va mustaqil ravishda robotning uning kvadratchasida ekanligini tekshiradi. Agar robot maxluq tekshirayotgan kvadratchada bo'lsa, u -10 mukofot oladi (ya'ni, 10 ochko yo'qotadi). Markaziy nuqtada, maxluq har bir vaqt bosqichida $p2=0.4$ ehtimoli bilan tekshiradi; boshqa x bilan belgilangan to'rt kvadratchada maxluqlar har bir vaqt bosqichida $p3=0.2$ ehtimoli bilan tekshiradi.

Mukofotlar holatga kirish paytida darhol olinadi deb faraz qiling; ya'ni agar robot maxluq bo'lgan holatga kirs, holatga kirishda (salbiy) mukofot oladi va agar robot xazina bo'lgan holatga kirs, xazina kelgan vaqtda ham, holatga kirishda mukofot oladi.

Robot sakkizta qo'shni kvadratga mos keladigan sakkizta harakatga ega. Diagonal harakatlar shovqinli; tanlangan yo'nalishda $p4 = 0.6$ ehtimollik va kerakli yo'nalishga eng yaqin to'rtta qo'shni kvadratga teng imkoniyat bor. Vertikal va gorizontal harakatlar ham shovqinli; so'ralgan yo'nalishda borish uchun $p5 = 0.8$ imkoniyat va qo'shni diagonal kvadratlardan biriga borish uchun teng imkoniyat mavjud. Masalan, yuqoriga-chapga va yuqoriga harakatlar quyidagi natijalar- ega:



Action=up-left



Action=up

Agar harakat devorga urilishi bilan yakunlansa, robot -2 mukofot (ya'ni 2 ochko yo'qotadi) va harakatlanmaydi.

Chegirma omili $p6 = 0.9$ ga teng.

(a) Necha holatlar mavjud? (Yoki qancha kam holatlar bilan qutulishingiz mumkin?) Ular nimani ifodalaydi?

(b) Optimal tamoyil nima?

(c) Faraz qilaylik, o'yin dizayneri o'yinchi uchun noaniq optimal tamoyillar bilan turli xil o'yin instansiyalarini yaratmoqchi. Parametrlar $p1$ dan $p6$ ga uchta tayinlashni turli optimal tamoyillar bilan bering. Agar bu qadar ko'p turli xil optimal tamoyillar mavjud bo'lmasa, mavjud bo'lganlaricha bering va nima uchun ko'proq yo'qligini tushuntiring.

12.17-mashq. Faraz qilaylik, oldingi savolga o'xshash 5×5 matritsali o'yin bor. Agent 25 joydan birida bo'lishi va xazina burchaklardan birida yoki hech qayerda bo'lmasligi mumkin.

«Yuqoriga» harakat oldingi savoldagi dinamikaga ega, ya'ni agent 0.8 ehtimollik bilan yuqoriga, 0.1 ehtimollik bilan yuqoriga-chapga va 0.1 ehtimollik bilan yuqoriga-o'ngga harakat qiladi.

Xazina yo'q bo'lsa, xazina 0.2 ehtimollik bilan paydo bo'lishi mumkin. U paydo bo'lganda, tasodifiy burchaklardan birida paydo bo'ladi va har bir burchakda teng ehtimollik bilan paydo bo'ladi. Agent xazina joylashgan maydonga tushmaguncha, xazina turgan joyida qoladi. Bu sodir bo'lganda, agent +10 mukofot oladi va keyingi holatga o'tishida xazina yo'qoladi. Agent va xazina bir vaqtning o'zida harakat qiladi, shuning uchun agent xazina paydo bo'lgan paytda kvadratga yetib borsa, mukofotni oladi.

Faraz qilaylik, biz asinxron qiymat iteratsiyasini amalga oshiryapmiz va har bir holat uchun qiymat quyidagi matritsadagi kabi. Kvadratchalardagi raqamlar holatning qiymatini ifodalaydi va bo'sh kvadratchalarning qiymati nolga teng. Ushbu qiymatlar qanday kiritilganligi bu savol uchun ahamiyatsiz.

			2	
			*	

			7	○
				7

Chap matritsada xazina yo'q holatlar uchun qiymatlarni ko'rsatadi va o'ng matritsada yuqoridagi o'ng burchakda xazina bo'lgan holatlar uchun qiymatlarni ko'rsatadi. Xazina boshqa uch burchakda bo'lgan holatlar uchun qiymatlar hozirda nolga teng deb hisoblang.

Asinxron qiymat iteratsiyasining keyingi bosqichini ko'rib chiqing. $S13$ holati uchun, rasmda * bilan belgilangan va $a2$ harakati, ya'ni «yuqoriga», keyingi qiymat iteratsiyasida $Q(S_{13}, a_2)$ ga qanday qiymat tayinlanadi? Siz barcha ishingizni

ko'rsatishingiz kerak, lekin arifmetikani bajarishingiz shart emas (ya'ni ifoda sifati qoldiring). Ifodangizdagi har bir atamani tushuntiring.

12.18-mashq. Qaror tarmog'ida bir nechta foyda tugunlari mavjud bo'lib, ularning qiymatlari qo'shilishi kerak, deb faraz qilaylik. Bu umumiy qo'shimcha foyda funksiyasini ifodalash imkonini beradi. Qaror tarmoqlari uchun VE algoritmini, 12.14-rasmda ko'rsatilganidek, bunday foydalarni o'z ichiga olgan holda qanday o'zgartirish mumkin?

12.19-mashq. Qaror tarmoqlari uchun o'zgaruvchini istisno qilish, 12.14-rasmda ko'rsatilganidek, qo'shimcha chegirmali mukofotlarni o'z ichiga olgan holda qanday o'zgartirilishi mumkin? Ya'ni qo'shilishi va chegirmalanishi kerak bo'lgan bir nechta foyda (mukofot) tugunlari bo'lishi mumkin. Istisno qilinishi kerak bo'lgan o'zgaruvchilar eng so'nggi vaqt bosqichidan boshlab istisno qilinishini faraz qiling.

13-bob. KUCHAYTIRILGAN O'QITISH (REINFORCEMENT LEARNING)

Biz intellekt va unga aloqador qobiliyatlarni mukofotlarni maksimal darajada oshirishga xizmat qiladi, deb faraz qilamiz. Shu sababli, mukofot xulq-atvorni boshqarish uchun yetarli bo'lib, u tabiiy va sun'iy intellektida o'rganilgan qobiliyatlarni, jumladan bilim, o'rganish, idrok, ijtimoiy intellekt, til, umumlashma va taqlidni o'z ichiga oladi.

– Silver va boshq. [2021]

Kuchaytirilgan o'qitish (Reinforcement Learning – RL) agenti muhitda harakat qilib, uning holatini kuzatadi va mukofotlar oladi. Agent harakatlar oqimini kuzatib, keyingi holat va mukofotni ko'rib chiqib, to'plangan mukofotni maksimal darajada oshirish maqsadini hisobga olgan holda nima qilish kerakligini aniqlashi kerak. Ushbu bobda to'liq kuzatiladigan, yagona agentli Kuchaytirilgan o'qitishni ko'rib chiqiladi. Ko'p agentli Kuchaytirilgan o'qitish 14.7.2-qismda tasvirlangan. Bu o'tish va mukofot modellari aniq bo'lmagan holat uchun qarorlar nazariyasi rejalashtirishining kengaytmasidir.

Biz allaqachon foydaning umumiyligini ko'rsatuvchi natijalarni ko'rib chiqdik. Silver va boshq. [2021] foydani mukofotlarga ajratish aqlli harakatning asosini tashkil qilishi mumkinligini ta'kidlaydi.

13.1. Kuchaytirilgan o'qitish muammosi

Kuchaytirilgan o'qitish agenti quyidagicha tavsiflanadi:

O'rganish agentiga dunyoning mumkin bo'lgan holatlari va u bajarishi mumkin bo'lgan harakatlar to'plami beriladi.

Har safar agent dunyo holatini (muhit va agentni) va olingan mukofotni kuzatadi.

Holat va mukofotni kuzatgandan so'ng, agent harakatni amalga oshiradi.

Agentning maqsadi, ba'zi chegirma omili γ uchun chegirma qilingan mukofotni maksimal darajada oshirishdir.

Kuchaytirilgan o'qitish **Markov qaror jarayonlari** (MDP) nuqtayi nazaridan rasmiylashtirilishi mumkin, bunda agent dastlab faqat mumkin bo'lgan holatlar to'plamini va mumkin bo'lgan harakatlar to'plamini biladi. Dinamika, $P(s' | a, s)$ va mukofot funksiyasi, $R(s, a)$, agentga berilmagan. MDPda bo'lgani kabi, har bir harakatdan so'ng, agent o'z holatini kuzatadi va mukofot oladi.

13.1-misol. 13.1-rasmda ko'rsatilgan domenni ko'rib chiqing. Agent bo'lishi mumkin bo'lgan olti holat mavjud bo'lib, ular $S_0, \dots, S_5$ deb nomlangan. Agentda to'rtta harakat mavjud: *yuqoriga R*, *yuqoriga C*, *chap*, *o'ng*. Bu agent boshlashidan oldin biladigan hamma narsadir. U holatlar qanday tuzilganini, harakatlar nima qilishini yoki mukofotlar qanday qo'lga kiritilishini bilmaydi.

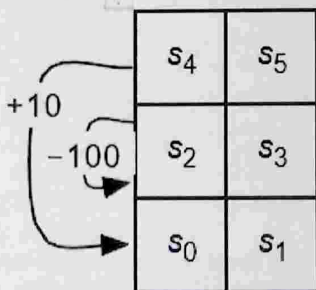
13.1-rasm olti holatning konfiguratsiyasini ko'rsatadi. Faraz qilaylik, harakatlarni quyidagicha ishlaydi:

right (o'ng) Agent S_0, S_2 va S_4 holatlarida o'ngga harakat qiladi, 0 mukofot oladi va boshqa holatlarda esa o'ngga harakat qilmaydi, -1 mukofot oladi.

left (chap): Agent 0 mukofot bilan S_1, S_3 va S_5 holatlarida bir holat chapga harakat qiladi. S_0 holatida esa u S_0 holatda qoladi va mukofot -1 oladi. S_2 holatida u -100 mukofot oladi va S_2 holatda qoladi. S_4 holatida u 10 mukofot oladi va S_0 holatiga o'tadi.

upC (yuqorigaC) (ya'ni «yuqoriga ehtiyotkorlik bilan»). Agent yuqoriga harakat qiladi, S_4 va S_5 holatlari bundan mustasno, bu yerda agent tushadi va joyida qoladi. U -1 miqdorida mukofot oladi, halokat hollari bundan mustasno, bu holda mukofot -2 ni tashkil qiladi.

upR (yuqorigaR) (ya'ni «yuqoriga tavakkal») 0,8 ehtimollik bilan u *upC* vazifasini bajaradi, faqat halokatga uchraganida mukofot -1, aks holda 0 ga teng. 0,1 ehtimollik bilan u *chap* va 0,1 ehtimollik bilan *o'ng* kabi harakat qiladi.



13.1-rasm. Kichkina kuchaytirilgan o'qitish muammosi muhiti

$\gamma = 0.9$ chegirma bilan, chegirma mukofoti mavjud. Bu agentning har qanday bosqichda 0,1 ehtimollik bilan o'yinni tark etish ehtimoli yoki agent kelajak uchun darhol mukofotni afzal ko'rishini ko'rsatish usuli sifatida tarjima qilinishi mumkin.

Agent+10 mukofotini olish uchun holatidan imkon qadar chapga harakat qilishga harakat qilishi kerak. S_0 holatidan S_4 holatiga borayotganda, S_2 holatidagi xavfli jarlik yonidan o'tib ketishi mumkin, bu yerda katta salbiy mukofot olish xavfi mavjud, yoki uzoqroq va xavfsiz yo'l bilan o'tishi mumkin. Dastlab, u buni bilmaydi, lekin buni o'rganishi kerak.

13.2-misol. 13.2-rasmda ko'rsatilgan murakkabroq o'yin domenini ko'rib chiqing. Agent bo'lishi mumkin bo'lgan 25 matritsada joylashuvi mavjud. Mukofot burchaklardan birida bo'lishi mumkin yoki hech qanday mukofot bo'lmasligi mumkin. Agent mukofotga tushganda, u 10 mukofot oladi va mukofot yo'qoladi. Mukofot bo'lmaganda, har bir vaqt bosqichida mukofot har qanday burchaklardan birida paydo bo'lish ehtimoli mavjud. Maxluqlar har qanday vaqtda «M» belgilangan joylardan birida paydo bo'lishi mumkin. Agar agent maxluq paydo bo'lgan kvadratda bo'lsa, agent shikastlanadi. Agar agent allaqachon shikastlangan bo'lsa, u -10 mukofot oladi. Agent tuzatish stansiyasiga tashrif buyurib (R bilan belgilangan) tuzatilib (endi shikastlanmaganda bo'ladi).

Ushbu misolda holat to'rtta komponentdan iborat: (X, Y, D, P) , bu yerda X agentning X -koordinatasi, Y agentning Y -koordinatasi, D Boolean va agent shikastlanganida true bo'ladi va P mukofotning pozitsiyasi ($P = 0$ agar mukofot bo'lmasa, $P = i$ agar mukofot pozitsiya P_i da bo'lsa). Chunki maxluqlar o'tkinchi bo'lib, maxluq paydo bo'lgan vaqtini bilish kelajak haqida hech qanday ma'lumot bermaydi, ularni holatning bir qismi sifatida kiritish shart emas. Shunday qilib, $5*5*2*5=250$ holat mavjud. Muhit to'liq kuzatiladigan, shuning uchun agent qaysi holatda ekanligini biladi. Agent holatlar ma'nosini bilmaydi; u to'rtta komponent haqida bilmaydi va dastlab shikastlanganlik yoki mukofot nima ekanligini bilmaydi.

Agentda to'rtta harakat mavjud: *yuqoriga*, *pastga*, *chapga* va *o'ngga*. Bular agentni bir qadam – odatda nom bilan ko'rsatilgan yo'nalishda bir qadam, lekin ba'zan boshqa yo'nalishlardan biriga siljitadi. Agar agent tashqi devorga yoki ichki devorlardan biriga (R joylashgan joy yaqinidagi qalin chiziqlar) urilsa, u o'z joyida qoladi va -1 mukofot oladi.

P_1	R		P_2
		M	
			M
M	M	M	
P_3			P_4

13.2-rasm. Maxluq o'yinining muhiti

Agent bu yerda berilgan hikoyani bilmaydi. U faqat 250 ta holat va to'rtta harakat mavjudligini, har safar qaysi holatda ekanligini va har safar qaysi mukofotni olganligini biladi.

Bu o'yin oddiy, lekin yaxshi boshqaruvchini yozish hayratlanarli darajada qiyin. Darslik veb-saytida siz o'ynashingiz va o'zgartirishingiz mumkin bo'lgan ilovalar mavjud. Buning uchun qo'lda boshqaruvchi yozishga harakat qiling; har 1000 qadam uchun taxminan 500 mukofot to'playdigan boshqaruvchini yozish mumkin. Bu o'yinni o'rganish ham qiyin, chunki R ga tashrif buyurish foydasiz ko'rinadi, to agent shikastlanish yomonligini va R ga tashrif buyurish uni tuzatishini bilib olmaguncha. U mukofotlarni to'plab olishga harakat qilayotganda bunga duch kelishi kerak. Mukofot mavjud bo'lmagan holatlar uzoq davom etmaydi. Bundan tashqari, u *zarar* haqida hech qanday tasavvurga ega bo'lmasdan buni o'rganishi kerak; u dastlab 250 ta holat va to'rtta harakat borligini biladi halos.

Kuchaytirilgan o'qitish bir qancha sabablarga ko'ra qiyin:

Kredit taqsimlash muammosi yoki **ayblash muammosi**, bu mukofot yoki jazoga qaysi harakat javobgar ekanligini aniqlash muammosi. Mas'ul harakat mukofotlar olishdan ancha oldin sodir bo'lgan bo'lishi mumkin. Bundan tashqari, bitta harakat emas, balki tegishli sharoitlarda amalga oshirilgan harakatlar kombinatsiyasi mukofot uchun javobgar bo'lishi mumkin. Masalan, siz agentga o'yinni yutganida yoki yutqazganida mukofot berib, o'yin o'ynashni o'rgatishingiz mumkin; u yutish uchun zarur bo'lgan mohirona harakatlarni aniqlashi kerak, bu odatda oxiridan ancha oldin sodir bo'ladi. Yana bir misol, siz itni uyga kelib, tartibsizlikni topganingizda «yomon it» deb aytish orqali o'rgatishingiz mumkin. It, barcha qilgan harakatlaridan, qaysi biri tanbeh uchun javobgar ekanligini aniqlashi kerak.

Agar dunyo dinamikasi o'zgarmasa ham, agentning harakati agentning kelajakda nima qilishiga bog'liq. Agent uchun dastlab yomon ish bo'lib tuyulishi mumkin bo'lgan narsa, agentning kelajakda qiladigan ishi tufayli eng yaxshi harakat bo'lishi mumkin. Bu rejalashtirish muammolarida keng tarqalgan, lekin Kuchaytirilgan o'qitish kontekstida murakkabdir, chunki agent o'z harakatlarining ta'sirini oldindan bilmaydi.

Izlanish-foydalanish dilemmasi: agar agent yaxshi harakatlar ketma-ketligini aniqlagan bo'lsa, u bu harakatlarni davom ettirishi kerakmi (aniqlagan narsalaridan foydalanish) yoki yaxshiroq harakatlarni topish uchun izlanishi kerakmi? Hech qachon izlanmaydigan agent, agar ilgari izlanganida yaxshiroq bo'lishi mumkin bo'lgan tarzda abadiy harakat qilishi mumkin. Doim izlanadigan agent hech qachon o'rgangan narsalaridan foydalanmaydi. Izlanish qaytarilmas zarar keltirishi mumkin. Ushbu dilemma 13.5-mavzuda batafsil muhokama qilinadi.

13.2. Evolyutsion algoritmlar

Reinforcement algoritmlarini hal qilishning bir usuli bu muammoni optimallashtirish masalasi sifatida ko'rib chiqish, ya'ni kutilgan chegirmali mukofotni maksimal darajada oshiradigan tamoyilni tanlashdir. Bu tamoyillar maydonida eng yaxshi tamoyilni topish uchun **tamoyilni qidirish** orqali amalga oshirilishi mumkin. Tamoyil – bu muhitda agentni boshqaradigan nazorat qilish tizimi bo'lib, u agentni muhitda ishlashini tekshirish orqali baholanadi.

Tamoyilni qidirish ko'pincha tamoyillar maydonida stoxastik lokal qidirish algoritmi sifatida hal qilinadi. Tamoyil muhitda bir necha marta ishga tushirish orqali baholanadi.

Dastlabki tamoyillar to'plami muhitda takroran baholanib, iterativ ravishda takomillashtirilishi mumkin. Ushbu jarayon **evolyutsion algoritmi** deb ataladi, chunki agent sifatida ko'rilgan har bir tamoyil qanchalik omon qolganligi bilan baholanadi. Bu ko'pincha genetik algoritmlar bilan birlashtiriladi, bu esa tamoyil komponentlarini birlashtirish va xilma-xil boshqaruvchilar to'plamini yaratish imkonini beradi. Ko'pincha bu boshqaruvchilar neyron tarmoqlar sifatida ifodalangani, bu esa **neyroevolyutsiyaning** asosidir.

Bu algoritmlar xilma-xil boshqaruvchilarni o'rganish imkoniyatiga ega bo'lish afzalligiga ega.

Tamoyillarni butunlay baholashga asoslangan algoritmlar bir qator muammolarga duch keladi. Birinchi navbatda, holat maydonining kattaligi. Agar n holatlar va m harakatlar bo'lsa, m^n tamoyillar mavjud bo'ladi. Masalan, 13.1-misolda ko'rsatilgan o'yin uchun $4^6 = 4096$ turli tamoyillar mavjud. 13.2-misolda 250 holatlar mavjud, shuning uchun $4^{250} \approx 10^{150}$ tamoyillar mavjud. Bu juda kichik o'yin, lekin u koinotdagi zarralardan ko'proq tamoyillarga ega.

Ikkinchidan, evolyutsion algoritmlar tajribalarni juda isrofli ishlatadi. Agar agent 13.1-misoldagi S_2 holatida bo'lganida va chapga siljigan bo'lsa, siz uni S_2 holatidan chapga siljish yomon ekanligini bilib olishini xohlaysiz. Lekin evolyutsion algoritmlar, taqdim etilganidek, agent tugagunga qadar kutib turadi va tamoyilni butunlay baholaydi. Stoxastik lokal qidirish tasodifiy ravishda S_2 holatida boshqa narsalarni sinab ko'radi va bu harakat yaxshi bo'lmaganligini aniqlashi mumkin, lekin bu juda bilvosita. Genetik algoritmlar biroz yaxshiroqdir, chunki S_2 holatida chapga siljigan agent tamoyillari yo'qolib ketadi, lekin bu ham juda bilvosita.

Uchinchidan, evolyutsion algoritmlarning ishlashi tamoyil ifodalanishiga juda sezgir bo'lishi mumkin. Genetik algoritmi uchun ifodalanish shunday bo'li-

shi kerakki, krossover tamoyilning yaxshi qismlarini saqlab qolsin. Ifodalanishlar ko'pincha maxsus domen uchun sozlanadi.

Ushbu bobning qolgan qismida har bir harakatdan so'ng o'rganishni davom ettirish muqobili ko'rib chiqiladi. Tamoyilning komponentlari butun tamoyil o'rniga o'rganiladi. Har bir holatda nima qilish kerakligini o'rganish orqali o'rganish holatlar soni bo'yicha eksponensial emas, balki holatlar soni bo'yicha chiziqli yoki polinomial vaqt va fazoviy murakkablikka ega bo'lishi mumkin.

Biroq, evolyutsion algoritmlar ba'zida boshqa usullardan qochadigan muammolarga ijodiy yechimlar topishi mumkin (13.9.2-mavzuga qarang).

13.3. Vaqtinchalik farqlar

Kuchaytirilgan o'qitish qanday ishlashini tushunish uchun, agentga ketma-ket ravishda keladigan qiymatlarni qanday o'rtacha hisoblashni ko'rib chiqing. A.1-qism vaqtinchalik farqlar asosini tashkil etuvchi o'rtacha harakatlarni qanday saqlash haqida muhokama qiladi.

Faraz qilaylik, $v_1, v_2, v_3, \dots$ raqamli qiymatlar ketma-ketligi mavjud va maqsad keyingisini bashorat qilishdir. Aylanuvchi o'rtacha A_k qiymati saqlanadi va A.1-qismda keltirilgan **vaqtinchalik farqlar tenglamasidan** foydalanib yangilanadi:

$$A_k = (1 - \alpha_k) * A_{k-1} + \alpha_k * v_k = A_{k-1} + \alpha_k * (v_k - A_{k-1}) \quad (13.1)$$

bu yerda $\alpha_k = \frac{1}{k}$.

Farq, $v_k - A_{k-1}$ **vaqtinchalik farq xatosi** yoki **TD xatosi** deb ataladi; bu yangi qiymat, v_k eski bashorat, A_k dan qanchalik farq qilishini ko'rsatadi. Eski baho, A_{k-1} yangi A_k bahosini olish uchun TD xatosining α_k marta yangilanadi.

Vaqtinchalik farqlar tenglamasining sifatli talqini shundan iboratki, agar yangi qiymat eski bashoratdan yuqori bo'lsa, bashorat qilinadigan qiymatni oshiring; agar yangi qiymat eski bashoratdan kam bo'lsa, bashorat qilinadigan qiymatni kamaytiring. O'zgarish yangi qiymat va eski bashorat orasidagi farqqa proporsionaldir. E'tibor bering, bu tenglama birinchi qiymat, $k = 1$, uchun ham amal qiladi, bu holda $A_1 = v_1$.

Kuchaytirilgan o'qitishda qiymatlar ko'pincha harakatlarning ta'sirlarini baholashdir; so'nggi qiymatlar avvalgi qiymatlardan aniqroq bo'ladi, chunki agent o'rganmoqda va shuning uchun ular ko'proq og'irlikka ega bo'lishi kerak. Keyinchalik misollarga ko'proq og'irlik berishning bir usuli (13.1) tenglamasidan foydalanishdir, lekin α doimiy ($0 < \alpha \leq 1$) deb belgilanishi kerak va k ga bog'liq emas. Bu qiymatlar ketma-ketligida o'zgarish bo'lganda o'rtacha qiymatga yaqinlashmaydi, lekin qiymatlar o'zgarishini yaratadigan asosiy jarayon o'zgarganda kuzatib borishi mumkin. A.1-mavzuga qarang.

Ko'proq so'nggi tajribalarga ko'proq og'irlik berishning bir usuli, lekin o'rtachaga yaqinlashish, $\alpha_k = (r + 1)/(r + k)$ deb belgilashdir, bu yerda $r > 0$. Birinchi tajribada $\alpha_1 = 1$, shuning uchun u oldingi A ni e'tiborsiz qoldiradi. Agar $r = 9$ bo'lsa, 11 tajribadan so'ng, $\alpha_{11} = 0.5$, shuning uchun bu tajriba barcha oldingi tajribalari bilan teng og'irlikda. r parametri domen uchun mos ravishda belgilanishi kerak.

O'rtachaga yaqinlashuvni kafolatlash, statsionar bo'lmagan dinamika yoki mukofotlar uchun asosiy qiymat yaratish jarayoni o'zgarganda aniqroq boshqartarlarni ishlab chiqarish uchun moslashish qobiliyati bilan mos kelmaydi.

13.4. Tajribalardan o'rganish

Kuchaytirilgan o'qitishda agent atrof-muhit bilan o'zaro ta'sir qilish tarixidan optimal tamoyilni o'rganishga harakat qiladi. Agent **tarixi** bu holat-harakat-mukofotlar ketma-ketligi:

$$\langle s_0, a_0, r_1, s_1, a_1, r_2, s_2, a_2, r_3, s_3, a_3, r_4, s_4 \dots \rangle$$

Bu shuni anglatadiki, agent s_0 holatida bo'lgan va a_0 harakatini amalga oshirgan, natijada r_1 mukofotini olgan va s_1 holatiga o'tgan; keyin a_1 harakatini qilgan, r_2 mukofotini olgan va s_2 holatiga o'tgan; keyin a_2 harakatini qilgan, r_3 mukofotini olgan va s_3 holatiga o'tgan va hokazo.

Biz bu o'zaro ta'sir tarixini tajribalar ketma-ketligi sifatida ko'rib chiqamiz, bunda *tajriba* to'rtlik $\langle s, a, r, s' \rangle$ dan iborat bo'lib, bu agent s holatida bo'lganligini, a harakatini qilganligini, r mukofotini olganligini va s' holatiga o'tganligini bildiradi. Bu tajribalar agentga nima qilishni o'rganish uchun ma'lumot beradi. Qaror-nazariy rejalashtirishda bo'lgani kabi, maqsad agentning qiymatini maksimal darajada oshirishdir, bu odatda chegirmali mukofotni anglatadi.

13.4.1. Q-o'rganish

Eslatib o'tamiz, $Q^*(s, a)$ bu yerda a harakat va s holat, s holatida a harakatini bajarish va keyin optimal tamoyilga rioya qilishning kutilayotgan qiymati (kumulativ chegirmali mukofot) hisoblanadi.

Q-o'rganish vaqtinchalik farqlarning $Q^*(s, a)$ qiymatini baholash uchun ishlatadi. Q-o'rganishda agent $Q/S/A$ jadvalini saqlaydi, bu yerda S holatlar to'plami va A harakatlar to'plami hisoblanadi. $Q/s/a$ hozirgi $Q^*(s, a)$ bahosini ifodalaydi.

Tajriba $\langle s, a, r, s' \rangle$ $Q(s, a)$ qiymati uchun bir ma'lumot nuqtasini beradi. Bu ma'lumot nuqtasi agent $r + \gamma V(s')$ ning kelajak qiymatini olganligini bildiradi, bu yerda $V(s') = \max_{a'} Q(s', a')$; bu hozirgi mukofot va chegirmali kelajak qiymatining bahosi. Ushbu yangi ma'lumot nuqtasi **daromad** deb ataladi. Agent

vaqtinchalik farq tenglamasidan (13.1) foydalanib $Q(s,a)$ bahosini yangilashi mumkin:

$$Q[s,a] := Q[s,a] + \alpha * (r + \gamma \max_{a'} Q[s',a']) - Q[s,a]$$

yoki teng ravishda:

$$Q[s,a] := (1 - \alpha) * Q[s,a] + \alpha * (r + \gamma \max_{a'} Q[s',a']).$$

13.3-rasmda Q -o'rganish boshqaruvchisi ko'rsatilgan bo'lib, agent harakat qilmoqda va bir vaqtning o'zida o'rganmoqda. 17-qatoridagi **do buyrug'i**, $do(a)$, boshqaruvchi tanaga yuboradigan **buyrug'ni** bildiradi. Mukofot va natijaviy holat boshqaruvchi tanadan olgan **idroklarni** bildiradi.

13.3-rasm algoritmi, shuningdek, $N[s,a]$ massivini saqlaydi, bu holatda a harakati necha marta bajarilganligini hisoblaydi. $Alpha_fun$ funksiyasi hisoblashlar uchun α ni hisoblaydi. $alpha_{fun(c)} = 10/(9 + c)$ ko'pincha yaxshi ishlaydi; 13.6-mashqqa qarang. Agar a o'zgarmas bo'lsa, N massivi saqlanishi shart emas (lekin ba'zi kashfiyot strategiyalari uchun ham ishlatiladi; quyida qarang).

Q -o'rganish agent yetarlicha kashf qilsa va har qanday holatda harakatlarni necha marta sinab ko'rishiga hech qanday cheklov bo'lmasa (ya'ni u har doim bir holatda bir xil harakatlar to'plamini qilmasa) optimal Q -funksiyani (yoki uning taxminiy qiymatini) o'rganadi.

13.3-misol. 12.29-misoldagi ikki holatli MDPni ko'rib chiqing. Agent ikkita holat {healthy,sick} ({sog'lom, kasal}) va ikkita harakat {relax, party} ({dam olish, ziyofat qilish}) mavjudligini biladi. U modelni bilmaydi va s,a,r,s' tajribalardan o'rganadi. Chegirma bilan, $\gamma = 0.8$, $\alpha = 0.3$ va boshlang'ich Q qiymati dastlab 0 ga teng ekanligini hisobga olsak, quyida mumkin bo'lgan yo'nalish (bir nechta muhim raqamlar qisqartirilgan holatlar va harakatlar bilan):

s	a	r	s	$Update=(1-\alpha)*Q[s,a] + \alpha(r + \gamma \max_{a'} Q[s',a'])$
he	re	7	he	$Q[he, re] = 0.7*0 + 0.3*(7 + 0.8*0) = 2.1$
he	re	7	he	$Q[he, re] = 0.7*2.1 + 0.3*(7 + 0.8*2.1) = 4.07$
he	pa	10	he	$Q[he, pa] = 0.7*0 + 0.3*(10 + 0.8*4.07) = 3.98$
he	pa	10	si	$Q[he, pa] = 0.7*3.98 + 0.3*(10 + 0.8*0) = 5.79$
si	pa	2	si	$Q[si, pa] = 0.7*0 + 0.3*(2 + 0.8*0) = 0.06$
si	re	0	si	$Q[si, re] = 0.7*0 + 0.3*(0 + 0.8*0.06) = 0.014$
si	re	0	he	$Q[si, re] = 0.7*0.014 + 0.3*(0 + 0.8*5.79) = 1.40$

```

1: controller Q-learning(S, A,  $\gamma$ , alpha_fun)
2:   Inputs
3:     S is a set of states
4:     A is a set of actions
5:      $\gamma$  the discount
6:     alpha_fun is a function to compute step size from counts
7:   Local
8:     real array Q[S, A]
9:     integer array N[S, A]
10:    states s, s
11:    action a
12:    initialize Q[S, A] arbitrarily
13:    initialize N[S, A] to 0
14:    observe current state s
15:    repeat
16:      select an action a
17:      do(a)
18:       $N[s, a] := N[s, a] + 1$ 
19:       $a := \text{alpha\_fun}(N[s, a])$ 
20:      observe reward r and state s
21:       $Q[s, a] := Q[s, a] + \alpha * (r + \gamma * \max_c Q[s, c] - Q[s, a])$ 
22:      s := s
23:    until termination()

```

13.3-rasm. Q-o'rganishning boshqaruvchisi

Agar α o'zgarmas bo'lsa, Q-qiymatlar yaqinlashadi, lekin 12.33-misoldagi qiymatlarni takrorlash yordamida olingan qiymatlarga yaqinlashmaydi. α qancha kichik bo'lsa, u haqiqiy Q-qiymatlarga shuncha yaqinroq bo'ladi, lekin yaqinlashuv shunchalik sekin bo'ladi.

13.5. Tadqiq qilish va foydalanish

Q-funksiyaning taxmini agent nimani qilishini (13.3-rasmning 16-qatorida qaysi harakat tanlanganini) aniqlash uchun yetarli emas. Agent nima qilishini belgilash uchun ikkita raqobatlashayotgan maqsadlar mavjud:

U topilgan bilimlardan **foydalanishi** va mukofotlarni ko'paytirishi mumkin, ya'ni *s* holatda, $Q[s, a]$ ni maksimal darajada oshiradigan harakatlardan birini bajarishi mumkin.

U *Q*-funksiyaning yaxshiroq bahosini qurish uchun **tadqiq** qilishi mumkin, masalan, har safar tasodifiy harakatni tanlash orqali.

Agent bir holatda va harakatda ko'p marta sinov o'tkazganida *Q*-qiymatlarining yaxshi bahosini oladi. Agar agent doimiy ravishda foydalanadigan bo'lsa, u bir holatda bir xil harakatni sinovdan o'tkazadi va joriy eng yuqori *Q*-qiymatga ega bo'lgan harakatdan yaxshiroq bo'lishi mumkin bo'lgan harakatni topish uchun yetarli darajada tadqiq qilmaydi.

Tasodifiy tadqiq qilish agent topgan narsalardan foydalanishga imkon bermaydi, to u o'rganishni tugatgunga qadar, bu esa hech qachon sodir bo'lmasligi

mumkin. E'tibor bering, Q -o'rganish tasodifiy tadqiq qilish bilan ham optimal Q -funksiyaga yaqinlashadi. Tasodifiy tadqiq qilish holat maydonining qiziqarli qismlarini topish uchun haddan tashqari ko'p vaqt talab qilishi mumkin – agent bo'lishni xohlaydigan holat makonining qismlari – va shuning uchun agar maqsad o'rganish va keyin foydalanish bo'lsa, maqsadliroq tadqiq qilish yaxshiroq bo'lishi mumkin.

Foydalanish va tadqiq qilishni birlashtirishning bir qancha usullari mavjud:

Noaniqlik qarshisida optimizmda. Q -qiymatlar tadqiqotni rag'batlantiradigan qiymatlarga boshlang'ich qiymat sifatida beriladi. Agar Q -qiymatlar yuqori qiymatlar bilan boshlansa, tadqiq qilinmagan joylar yaxshi ko'rinadi, shuning uchun sof foydalanish tadqiq qilishni rag'batlantiradi. Bu tadqiqotni rag'batlantiradi; ammo, agent ba'zi holat-harakat juftliklari yaxshi deb hali mavjud bo'lma-gan dalil borligiga uzoq vaqt ishonib qolishi mumkin. Bir holat faqat uning barcha harakatlari yomon ko'rinadigan holatgacha yomon ko'rinishga ega bo'ladi; ammo bu harakatlarning barchasi yaxshi ko'rinadigan holatlarga olib kelsa, haqiqiy qiymatlarning real ko'rinishini olish uzoq vaqt talab qilishi mumkin. Bu holatda eski Q -qiymatlar haqiqiy Q -qiymatlarning yomon baholari va ular uzoq vaqt davomida yomon baholar bo'lib qolishi mumkin. Tez yaqinlashish uchun boshlang'ich qiymatlar iloji boricha yakuniy qiymatlarga yaqin bo'lishi kerak; ularni ortiqcha baholashga urinish yaqinlashishni sekinlashtiradi. Shovqinli muhitlarda, harakatning ta'siri biroz tasodifiy bo'lsa, boshqa tadqiqot mexanizmlari bo'lmagan noaniqlik sharoitida optimizm yaxshi harakat hech qachon ko'proq o'rganilmasligini anglatishi mumkin. chunki tasodif bilan u past Q -qiymat oladi va undan qutula olmaydi. Noaniqlik qarshisida optimizm boshqa usullar bilan birgalikda ishlatilishi mumkin.

ϵ -greedy (ochko'z) tadqiqot strategiyasida, bu yerda $0 \leq \epsilon \leq 1$, agent vaqtning ϵ qismini tasodifiy harakatini tanlaydi va aks holda $Q[s,a]$ ni maksimal darajada oshiradigan harakatni tanlaydi. ϵ vaqt davomida o'zgarishi mumkin. Intuitiv ravishda, agentning hayoti boshida, u ko'proq tasodifiy harakat qilishi kerak va boshlang'ich tadqiqotni rag'batlantirishi kerak va vaqt o'tishi bilan ko'proq greedy harakat qilishi kerak (ϵ ni kamaytirish).

ϵ -greedy strategiyaning muammosi shundaki, u eng yaxshisidan tashqari barcha harakatlarga teng munosabatda bo'ladi. Agar bir nechta yaxshi ko'rinadigan harakatlar va boshqa harakatlar bo'lsa, ular kamroq istiqbolli ko'rinadigan bo'lsa, yaxshi harakatlarni aniqlashga ko'proq kuch sarflash va kamroq istiqbolli ko'rinadigan harakatlarni tadqiq qilishga kamroq kuch sarflash oqilona bo'lishi mumkin. Buning usullaridan biri $Q[s,a]$ qiymatiga qarab ehtimollik bilan a harakatini tanlashdir. Bu **softmax** harakat tanlash deb nomlanadi. Keng tarqalgan usul **Gibbs** yoki **Boltsman taqsimotidan** foydalanishdir, bunda s holatda a harakatni tanlash ehtimoli $e^{Q[s,a]/\tau}$ ga proporsionaldir. Shunday qilib, s holatda agent a harakatini quyidagi ehtimol bilan tanlaydi:

$$\frac{e^{Q[s,a]/\tau}}{\sum_a e^{Q[s,a]/\tau}}$$

$\tau > 0$ qiymatlar qanday tasodifiy tanlanishi kerakligini belgilaydigan **harorat-dir**. Haroratning ehtimollikka qanday ta'sir qilishi 4.1-rasmda keltirilgan; masalan, 1 haroratda, agar $a1$ harakati $a2$ harakatidan ikki barobar kamroq foydaga ega bo'lsa, u $a1$ tanlanadigan vaqtning taxminan 14% vaqtda tanlanadi. τ yuqori bo'lganda, harakatlar deyarli teng miqdorda tanlanadi. Harorat kamayishi bilan yuqori qiymatli harakatlar tanlanishi ehtimoli oshadi va $\tau \rightarrow 0$ chegarasida eng yaxshi harakat har doim tanlanadi. Ko'pincha qidiruv davomida haroratni kamaytirish mumkin.

Oldingi usullar bilan bog'liq muammo shundaki, ular agent ko'p marta sinab ko'rgan harakatlarni, ular uchun haqiqiy Q -qiymatini yaxshi baholagan, ko'p sinab ko'rilmagan, bahosi juda yomon bo'lgan harakatlardan ajrata olmaydi. Boshqa usullar guruhi nafaqat hozirgi kutilayotgan qiymatni, balki kutilayotgan qiymat bahosining noaniqligini ham modellash tiradi. Agent ko'proq ma'lumot olgan sari, kutilayotgan qiymatning noaniqligi kamayadi va u kutilayotgan qiymatga nisbatan ishonchliroq bo'lishi mumkin. **Yuqori ishonch chegarasi** kutilayotgan qiymatning yuqori bahosidir; maqsad shuki, haqiqiy qiymat bundan yuqori bo'lishi juda dargumon bo'lgan qiymatni tanlashdir. Yuqori ishonch chegarasi ikki atamaning yig'indisi, Q bahosi va harakat qancha marta tanlanganligiga bog'liq bo'lgan ishonch chegarasidir. $N[s, a]$ holatda a harakati necha marta tanlanganligini va $N(s) = \sum_a N[s, a]$ holat necha marta tashrif buyurilganligini bildiradi. $N(s)$ saqlanishi yoki talabga binoan hisoblanishi mumkin. **Yuqori ishonch chegarasi (UCB1)** (bu nom Auer va boshqalar [2002] tomonidan taklif qilingan bo'lib, bir nechta algoritmlar tahlil qilingan) quyidagicha:

$$UCB1(s,a) = Q[s,a] + C * \sqrt{\frac{\log N(s)}{N[s,a]}}$$

bu yerda C , Q -qiymatlar kattaligiga bog'liq bo'lgan doimiy; agar qiymatlar $[0,1]$ oralig'ida bo'lsa va namunalarning mustaqil bo'lsa, $C = \sqrt{2}$ yaxshi nazariy xususiyatlarga ega. Kuchaytirilgan o'qitishda namunalar kamdan kam hollarda mustaqildir. s holatda agent eng yuqori $UCB1(s,a)$ qiymatiga ega bo'lgan a harakatini tanlash orqali tadqiqot va foydalanishni muvozanatlashi mumkin.

$Q(s, a)$ ning ketma-ket qiymatlari uning o'rtacha qiymatining yanada aniqlash tirilgan bahosini beradi. Ushbu baholar o'rtacha qiymat bo'yicha taqsimotni yaratish uchun ishlatilishi mumkin. Yuqori ishonch chegarasini tanlashning muqobili ushbu taqsimotdan tanlashdir. **Tompson namunasida**, har bir a harakati uchun va

qiymati o'rtacha taqsimotdan tanlanadi; bajarish uchun eng yuqori va qiymatiga ega harakat tanlanadi.

Agar qiymatlar hammasi 0 yoki 1 bo'lsa (masalan, agar mukofot oxirida faqat g'alaba yoki mag'lubiyat bo'lsa), 0 va 1 sonining soni bilan ifodalangan beta taqsimoti mos keladi.

Agar qaytarilgan natija haqiqiy son bo'lsa, uni o'rtacha va dispersiya bilan parametrlangan Gauss soni deb taxmin qilish odatiy holdir. Har bir holat uchun, quyidagi α harakatni tanlang:

$$Q[s, a] + C * \frac{randn()}{\sqrt{N[s, a]}}$$

bu yerda $randn()$ standart normal taqsimotdan (o'rtacha 0, dispersiya 1) tasodifiy son qaytaradi. C, Q -qiymatlari o'lchamini aks ettirish uchun tanlanadi.

Ba'zan beta yoki normal taqsimotlar mos kelmaydi, bu holda boshqa usullar bo'yicha keng adabiyotlar mavjud.

Tadqiqot – foydalanish o'rtasidagi kelishuv bo'yicha nazariy ishlarning ko'p qismi yagona holat uchun, **multi-armed bandit** (ko'p qo'lli bandit) muammosi deb ataladi; «**bandit**» – mukofotlarni har safar tasodifiy va mustaqil ravishda beradigan slot mashinasining jargonidir. Tadqiqot – foydalanish o'rtasidagi kelishuv bandit muammosidan ko'ra murakkabroq, chunki mukofotlar kechikishi mumkin va shuning uchun Q -qiymatining dastlabki baholari tasodifiy namuna bo'lmashligi mumkin va baholar mustaqil emas. Bu shuni anglatadiki, biz odatda amaliyotda nima ishlashini sinashimiz kerak; masalan, 13.4-mashqqa qarang.

13.6. RL algoritmlarini baholash

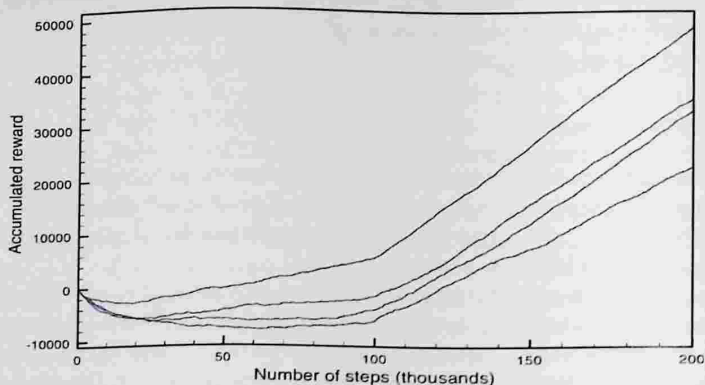
Biz kuchaytirilgan o'qitishda algoritmini qanday yaxshi tamoyil topishini yoki o'rgangan vaqtda qancha mukofot olishini baholaymiz. Qaysi biri muhimroq, agent qanday joylashtirilishiga bog'liq. Agar agentni xavfsiz o'rganish uchun yetarli vaqt bo'lsa, yakuniy tamoyil eng muhim bo'lishi mumkin. Agar agent o'rganish vaqtida joylashtirilishi kerak bo'lsa, u hech qachon tadqiqotdan chiqib ketish bosqichiga yetmasligi mumkin va agent o'rganish davomida oladigan mukofotni maksimallashtirish kerak.

Kuchaytirilgan o'qitishda algoritmining ishlashini ko'rsatishning bir usuli – kumulyativ mukofotni (shu kungacha olingan barcha mukofotlarning yig'indisi) qadamlar soni bo'yicha chizish. Bir algoritm, agar uning sxemasi doimiy ravishda ikkinchisidan yuqori bo'lsa boshqasidan ustun turadi.

13.4-misol. 13.4-rasmda Q -o'rganuvchisining to'rt xil o'yinlari 13.2-misol bo'yicha solishtirilgan.

Grafiklar α ning o'zgarishini, Q -funksiyaning boshlang'ich qiymatlari va harakat tanlashdagi tasodifiylikka qarab o'zgaradigan o'yinlar uchun keltirilgan.

Ularning barchasi birinchi 100,000 qadamda 80% greedy (ya'ni $\epsilon=0.2$) va keyingi 100,000 qadamda 100% greedy (ya'ni $\epsilon=0.0$) bo'lgan. Yuqori grafik boshqalaridan ustun keldi.



Qadamlar soni (minglab)

13.4-rasm. Qadamlar sonining umumiy mukofoti funksiyasi

Har bir algoritm turli xil ishga tushirishlarda juda farq qilishi mumkin, shuning uchun ushbu algoritmlarni taqqoslash uchun bir xil algoritm bir necha marta bajari-lishi kerak. Ushbu grafikda muhim bo'lgan uchta statistik ko'rsatkich mavjud:

Asimptotik nishab algoritm barqarorlashgandan keyin tamoyilning qanchalik yaxshi ekanligini ko'rsatadi.

Egri chiziqning minimumi yaxshilanishni boshlashdan oldin qancha mukofot qurbon qilinishini ko'rsatadi.

Nol kesish algoritmnining o'qish xarajatlarini tiklash uchun qancha vaqt keti- shini ko'rsatadi.

Oxirgi ikkita statistik ma'lumotlar ijobiy va salbiy mukofotlar mavjud bo'l- ganda qo'llaniladi va bu muvozanatli xatti-harakatlar oqilona hisoblanadi. Boshqa hollarda, jamlangan mukofotni domenga mos keladigan oqilona xatti-harakatlar bilan solishtirish kerak (13.3-mashqqa qarang).

Shuningdek, o'rtacha mukofotni (vaqt bosqichida to'plangan mukofot) reja- lashtirish mumkin. Bu oxir-oqibat o'rganilgan tamoyilning qiymatini va algoritm o'rganishni to'xtatganligini (u tekis bo'lganda) aniqroq ko'rsatadi, lekin ko'pin- cha dastlabki vaqtlarda katta o'zgarishlarga ega.

Kumulyativ mukofot rejasi haqida ta'kidlash kerak bo'lgan bir narsa shundaki, u umumiy mukofotni o'lchaydi, ammo algoritmlar har bir qadamda chegirmali mukofotni optimallashtiradi. Umuman olganda, siz domen uchun eng mos keladigan optimallik mezonidan foydalanib, algoritmingizni optimallashtirishingiz va baholashingiz kerak.

13.7. On-policy (Tamoyilga asoslangan o'rganish)

Q-o'rganish tamoyildan tashqari o'rganishdir. Off-policy (tamoyildan tashqari) o'rganish agentning harakatlaridan mustaqil ravishda optimal tamoyilning qiymatini o'rganadi, agar u yetarlicha tadqiq qilsa. Off-policy (tamoyildan tashqari) o'rganish agent tasodifiy harakat qilayotgan bo'lsa ham optimal tamoyilni o'rganishi mumkin. Tadqiqot qilayotgan off-policy (tamoyildan tashqari) o'rganish agent kuzatadigan tamoyilning qiymatini o'rganmaydi, chunki bu qadamlar tadqiqot bosqichlarini o'z ichiga oladi.

Ba'zi hollarda, masalan, katta salbiy mukofotlar mavjud bo'lganda, agentning haqiqatan nima qilayotganini e'tiborsiz qoldirish xavfli bo'lishi mumkin. Shu bilan bir qatorda agent haqiqatda amalga oshiradigan tamoyilning qiymatini o'rganishdir, bu tamoyilni takroriy takomillashtirish uchun qidiruv bosqichlarini o'z ichiga oladi. Shunday qilib, o'rganuvchi qidiruv ishlari bilan bog'liq xarajatlarni hisobga olishi mumkin. **Tamoyilni o'rganuvchi** agent tamoyilining qiymatini, jumladan, tadqiqot bosqichlarini bilib oladi.

SARSA (holat-harakat-mukofot-holat-harakat tajribalaridan foydalanganligi uchun shunday nomlangan) agent kuzatayotgan tamoyilning qiymatini baholaydigan *tamoyil asosida o'rganish* algoritmidir. SARSA dagi tajriba $\langle s, a, r, s', a' \rangle$ shaklida bo'ladi, ya'ni agent s holatida bo'lgan, a harakatini bajargan, r mukofotini olgan va s' holatida tugagan, undan a' harakatini qilishga qaror qilgan. Bu $Q(s, a)$ ni yangilash uchun yangi tajriba beradi. Ushbu tajribaning beradigan yangi qiymati

$$r + \gamma Q(s', a')$$

ni tashkil qiladi.

SARSA algoritmi 13.5-rasmda ko'rsatilgan. SARSA hisoblaydigan *Q*-qiymatlar hozirgi tadqiqot tamoyiliga bog'liq bo'lib, masalan, tasodifiy qadamlar bilan greedy bo'lishi mumkin. SARSA tadqiq qilish katta jarimalarga olib kelishi mumkin bo'lgan hollarda *Q*-o'rganishdan farqli tamoyilni topsa bo'ladi. Masalan, robot zinapoya tepasiga yaqinlashganda, hatto bu optimal tamoyil bo'lsa ham, bu tadqiqot bosqichlari uchun xavfli bo'lishi mumkin. SARSA buni kashf qiladi va robotni zinapoyalardan uzoqroq tutadigan tamoyilni qabul qiladi. Bu tamoyilda mavjud bo'lgan tadqiqotni hisobga olgan holda optimal tamoyilni topadi.

13.5-misol. 13.1-misolda optimal tamoyil s_0 holatidan yuqoriga o'tishdir (13.1-rasm). Ammo, agar agent tadqiq qilayotgan bo'lsa, bu harakat yomon bo'lishi mumkin, chunki s_2 holatidan tadqiq qilish juda xavfli.

Agar agent tamoyilni amalga oshirayotgan bo'lsa, unda «s holatda bo'lganda, vaqtning 80% $Q[s,a]$ ni maksimal darajada oshiradigan a harakatini va 20% tasodifiy harakatni tanlaydi», s_0 holatidan yuqoriga o'tish optimal emas. Tamoyilga asoslangan o'rganish agent kuzatayotgan tamoyilni optimallashtirishga harakat qiladi, bu esa optimal tamoyilni emas, tadqiqotni o'z ichiga oladi.

```

1: controller SARSA( $S, A, \gamma, \alpha$ )
2:   Inputs
3:      $S$  is a set of states
4:      $A$  is a set of actions
5:      $\gamma$  the discount
6:      $\alpha$  is the step size
7:   Local
8:     real array  $Q[S, A]$ 
9:     state  $s, s'$ 
10:    action  $a, a'$ 
11:    initialize  $Q[S, A]$  arbitrarily
12:    observe current state  $s$ 
13:    select an action  $a$  using a policy based on  $Q[s, a]$ 
14:    repeat
15:      do( $a$ )
16:      observe reward  $r$  and state  $s'$ 
17:      select an action  $a'$  using a policy based on  $Q[s', a']$ 
18:       $Q[s, a] := Q[s, a] + \alpha * (r + \gamma * Q[s', a'] - Q[s, a])$ 
19:       $s := s'$ 
20:       $a := a'$ 
21:    until termination()
```

13.5-rasm. SARSA: tamoyil bo'yicha kuchaytirilgan o'qitish

SARSAda optimal tamoyilning Q -qiymatlari Q -o'rganishdagi qiymatlarga nisbatan pastroq. 13.1-misol sohasidagi bir nechta holat-harakat juftliklari uchun Q -o'rganish va SARSA uchun qiymatlar (tadqiqot tezligi qavs ichida) quyidagicha:

Algorithm	$Q[s_0, right]$	$Q[s_0, up]$	$Q[s_2, upC]$	$Q[s_2, up]$	$Q[s_4, left]$
Q -learning	19.48	23.28	26.86	16.9	30.95
SARSA (20%)	9.27	7.9	14.8	4.43	18.09
SARSA (10%)	13.04	13.95	18.9	8.93	22.47

SARSA yordamida 20% tadqiqot bilan optimal tamoyil S_0 holatida o'ngga borish, ammo 10% tadqiqot bilan optimal tamoyil S_0 holatida yuqoriga ko'tarilishdir. 20% tadqiqot bilan bu optimal tamoyildir, chunki tadqiqot xavfli. 10% tadqiqot bilan S_2 holatiga kirish kamroq xavfli. Shunday qilib, agar tadqiqot tezligi kamaytirilsa, optimal tamoyil o'zgaradi. Biroq, kamroq tadqiqot bilan optimal tamoyilni topish ko'proq vaqt oladi. Q -o'rganish qiymati tadqiqot tezligiga bog'liq emas.

SARSA dunyoda tadqiqot qilayotgan agentni joylashtirishda foydalidir. Agar siz offlayn o'rganishni amalga oshirmoqchi bo'lsangiz va keyin bu tamoyilni tadqiq qilmaydigan agentda qo'llamoqchi bo'lsangiz, Q -o'rganish mos kelishi mumkin.

13.8. Modelga asoslangan RL

Kuchaytirilgan o'qitishning ko'plab ilovalarida har bir harakatni hisoblash uchun ko'p vaqt mavjud. Misol uchun, jismoniy robotda har bir harakat o'rtasida bir necha soniya bo'lishi mumkin. Har bir harakat uchun faqat bitta xazina nusxasini yaratadigan Q -o'rganish mavjud hisoblash vaqtidan to'liq foydalanmaydi.

Har bir harakatdan keyin bitta Q -qiymat yangilanishini amalga oshirishning o'rniga, tajribalarni modelni o'rganish uchun ishlatish alternativasi mavjud. Agent aniq $P(s' | s, a)$ va $R(s, a)$ ni o'rganishi mumkin. Muhitda agent harakat qilgan har bir harakat uchun, agent asinxron qiymat iteratsiyasining bir nechta qadamlarini bajarishi mumkin va Q -funksiyaning yaxshiroq bahosini berishi mumkin.

13.6-rasm modelga asoslangan kuchaytirilgan o'qitishning umumiy ko'rsatkichini ko'rsatadi. Boshqa kuchaytirilgan o'qitish dasturlari singari, u $Q[S, A]$ ni kuzatib boradi, lekin bu yerda T s sifatida taqdim etilgan dinamika modelini ham qo'llab – quvvatlaydi, bu yerda $T[s, a, s']$ agent s holatida a harakatini necha marta amalga oshirgani va s' holatida tugagan holatlar sonidir. Bu, shuningdek, agent s holatida a harakatini amalga oshirgan holatlar sonini ifodalovchi $C[s, a]$ ni saqlaydi. E'tibor bering, $C[s, a] = \sum_{s'} T[s, a, s']$ shuning uchun biz C ni saqlamasdan, kerak bo'lganda uni hisoblash orqali xotira tejashimiz mumkin. $R[s, a]$ massivi agent s holatida a harakatini amalga oshirganida olingan o'rtacha mukofotni saqlaydi.

Har bir harakatdan keyin agent mukofot r va natijaviy holat s ni kuzatadi. Keyin u o'tish hisoblagichlari matritsalarini T va C ni, shuningdek, o'rtacha mukofot R ni yangilaydi. Keyin u yangilangan ehtimollik modelidan olingan va yangilangan mukofot modelidan foydalanib, asinxron qiymat iteratsiyasining bir nechta bosqichlarini bajaradi.

Ushbu algoritmnin aniqlanmagan uchta asosiy qismi mavjud:

Qaysi Q -qiymatlar yangilanishi kerak? Algoritm hech bo'lmaganda $Q[s, a]$ m yangilashi mantiqan ko'rinadi, chunki o'tish ehtimolligi va mukofot bo'yicha ko'proq ma'lumot olingan. U yerdan u tasodifiy yangilanishlar amalga oshirishi yoki qaysi Q -qiymatlar eng ko'p o'zgarishini aniqlashi mumkin. Q -qiymatlar orasida eng ko'p o'zgarish potentsiali bo'lgan elementlar $Q[s_i, a_i]$ bo'lib, bu Q -qiymatning eng ko'p o'zgargan ehtimoli eng yuqori bo'lgan holat-harakat juftliklaridir (masalan, $Q[s_2, a_2]$ eng ko'p o'zgargan joyda). Bu holatlar uchun ko'rib chiqiladigan Q -qiymatlar navbatini saqlash orqali amalga oshirilishi mumkin. Nolgabo'lish xatosi yo'qligini ta'minlash uchun u faqat $C[s_1, a_1] \neq 0$ bo'lgan holat-harakat juftliklarini tanlashi yoki o'tishlar uchun psevdohisoblarni kiritishi kerak.

```

1: controller RL Model learner(S, A, γ)
2:   Inputs
3:     S is a set of states
4:     A is a set of actions
5:     γ the discount
6:   Local
7:     real array Q[S, A]
8:     real array R[S, A]
9:     integer array T[S, A, S]
10:    integer array C[S, A]
11:    initialize Q[S, A] arbitrarily
12:    initialize R[S, A] arbitrarily
13:    initialize T[S, A, S] to zero
14:    initialize C[S, A] to zero
15:    observe current state s
16:    select action a
17:    do(a)
18:    repeat
19:      observe reward r and state s'
20:       $T[s, a, s'] := T[s, a, s'] + 1$ 
21:       $C[s, a] := C[s, a] + 1$ 
22:       $R[s, a] := R[s, a] + \frac{r - R[s, a]}{C[s, a]}$ 
23:      s := s'
24:      select action a
25:      do(a)
26:      repeat
27:        select state s1, action a1 such that  $C[s_1, a_1] \neq 0$ 
28:         $Q[s_1, a_1] := R[s_1, a_1] + \gamma * \sum_{s_2} \frac{T[s_1, a_1, s_2]}{C[s_1, a_1]} * \max_{a_2} Q[s_2, a_2]$ 
29:      until an observation arrives
30:    until termination()

```

13.6-rasm. Model asosida kuchaytirilgan o'rganuvchi

Harakatlar orasida asinxron qiymat iteratsiyasining necha qadamlarini bajarish kerak? Agent harakatga majbur bo'lguncha yoki yangi ma'lumot olguncha Q-yangilanishlarni davom ettiradi. 13.6-rasmda agentning harakatini va keyin kuzatuv kelguniga qadar Q-yangilanishlarini amalga oshirishini nazarda tutadi. Kuzatuv kelganda, agent iloji boricha tezroq harakat qiladi. Buning bir nechta variantlari mavjud, jumladan, belgilangan miqdordagi yangilanishlarni amalga oshirish, bu har qanday vaqtda harakat qilishi mumkin bo'lgan o'yinlarda mos bo'lishi

mumkin. Monitoring va harakatlar bilan parallel ravishda yangilanishlarni ishga tushirish ham mumkin.

$Q[S,A]$ uchun dastlabki qiymatlar qanday bo'lishi kerak? Q ni yangilayotganda ilgari hech qachon duch kelmagan o'tishlar uchun qandaydir qiymat kerak bo'ladi. Agar u noaniqlik sharoitida optimizm strategiyasini ishlatayotgan bo'lsa, u R ning boshlang'ich qiymati sifatida mumkin bo'lgan maksimal mukofotni R_{max} dan foydalanishi mumkin, bu esa tadqiqotni rag'batlantiradi. Biroq, qiymatlarni takrorlashda bo'lgani kabi, agar u Q ni yakuniy Q -qiymatga iloji boricha yaqin qilib boshlashtirsa, algoritim tezroq yaqinlashadi.

Ushbu algoritim mukofotlar boshlang'ich holat va harakatga bog'liq deb hisoblaydi. Agar holatda bo'lish uchun alohida harakatlar xarajatlari va mukofotlar mavjud bo'lsa va agent xarajatlar va mukofotlarni alohida kuzatishi mumkin bo'lsa, mukofot funksiyasi $C[A]$ va $R[S]$ ga ajratilishi mumkin, bu esa samaraliroq o'rganishga olib keladi.

Modelga asoslangan va modeldan foydalanmasdan kuchaytirilgan o'qitishni to'g'ridan to'g'ri taqqoslash qiyin. Odatda, modelga asoslangan o'rganishlar tajriba jihatidan juda samaralidir; yaxshi o'rganish uchun ancha kamroq tajriba kerak bo'ladi. Biroq, modeldan foydalanmaydigan usullar kamroq xotira va ko'pincha kamroq hisoblash vaqtini talab qiladi. Agar tajriba arzon bo'lsa, masalan, kompyuter o'yinida, robot uchun tajriba qimmat bo'lganda boshqacha taqqoslash kerak bo'ladi.

13.9. Umumlashtirilgan kuchaytirilgan o'qitish

Odatda, holatlar soni juda ko'p bo'lgani uchun ularni aniq tasavvur qilish qiyin. Holatlar bo'yicha aniq fikr yuritishning o'rniga, xususiyatlar bo'yicha fikr yuritish muqobil variant sifatida qaraladi, bu xususiyatlar aniq berilishi yoki o'rganilishi mumkin.

13.7-rasm nazoratli o'rganuvchini o'z ichiga olgan umumiy mustahkamlash asosidagi tamoyil bo'yicha o'rganuvchini ko'rsatadi. Bu o'rganuvchi quyidagi operatsiyalarni amalga oshirishi mumkinligini taxmin qiladi:

$add(x,y)$ – yangi misolni ma'lumotlar to'plamiga qo'shadi, kirish qiymati x va maqsad qiymati y bo'lgan holda.

$predict(x)$ – kirish qiymati x bo'lgan misol uchun maqsadning nuqtaviy bashoratini beradi.

SARSA bilan umumlashtirishda, o'rganish uchun x kiritish holat-harakat juftligi va (s,a) juftligi uchun maqsad $Q^*(s,a)$ ning bahosidir.

7 va 8-boblarda ko'rib chiqilgan o'rganish usullaridan yagona farq shundaki, o'rganish qobiliyatiga ega bo'lishi kerak va olingan misollar asosida prognoz qilishni davom ettirishi kerak. Yangi misollar ko'pincha eski misollarga qaraganda yaxshiroq taxminlar beradi va algoritmlar buni hisobga olishlari kerak.

10-qatordagi navbatdagi harakat a' ni sof ekspluatatsiya bilan tanlash, *Learner.predict((s', a'))* ni maksimal darajada oshiradigan a' ni tanlashni anglatadi; tadqiqot 13.5-mavzudagi tadqiqot texnikalaridan biri yordamida amalga oshirilib mumkin.

Ushbu algoritmda umumlashtirishni o'rganuvchi orqali amalga oshiriladi. O'rganuvchi, masalan, chiziqli funksiya (keyingi bo'limga qarang), qarorlar daraxti o'rganuvchisi yoki neyron tarmoq bo'lishi mumkin. SARSA – bu algoritmining namunasi bo'lib, u yerda o'rganuvchi eslab qoladi, lekin umumlashtirmaydi.

Chuqur Kuchaytirilgan o'qitishida chuqur neyron tarmoq o'rganish vositasi sifatida ishlatiladi. Xususan, Q -funksiyani, qiymat funksiyasini va/yoki tamoyilni ifodalash uchun neyron tarmoqdan foydalanish mumkin. Chuqur o'rganish katta hajmdagi ma'lumotlarni va o'rganish uchun ko'p takrorlanishni talab qiladi va taqdim etilgan arxitekturaga sezgir bo'lishi mumkin. Go or Chess kabi o'yinlarda juda muvaffaqiyatli ekanligini isbotlagan bo'lsa-da (14.7.3-mavzuga qarang), bu ishni bajarish juda qiyin va hisoblash uchun juda zich. Kichikroq muammolar uchun chiziqli funksiya odatda yaxshiroqdir.

1: **controller** SARSA with Generalization(Learner, γ)

2: **Inputs**

3: Learner with operations *Learner.add(x, y)* and *Learner.predict(x)*.

4: $\gamma \in [0, 1]$: discount factor

5: observe current state s

6: select action a

7: **repeat**

8: *do(a)*

9: observe reward r and state s_{new}

10: select action a_{new} based on *Learner.predict((s_{new}, a_{new}))*

11: *Learner.add((s, a), $r + \gamma * \text{Learner.predict}((s_{\text{new}}, a_{\text{new}}))$)*

12: $s := s_{\text{new}}$

13: $a := a_{\text{new}}$

14: **until** *termination()*

13.7-rasm. Umumlashtirish bilan SARSA

13.9.1. SARSA algoritmlarining chiziqli funksiya bilan yaqinlashuvi

Holat va harakat xususiyatlarining chiziqli funksiyasi bo'lgan umumlashtirish bilan SARSA misolini ko'rib chiqing. Qaror daraxti yoki neyron tarmoq kabi murakkab muqobillar mavjud bo'lsa-da, chiziqli funksiya ko'pincha yaxshi ishlaydi, ammo **xususiyatlarni, funksional imkoniyatlarni ishlab chiqishni** talab qiladi.

Xususiyatga asoslangan o'rganishlar hozirgacha ko'rib chiqilgan Kuchaytirilgan o'qitish usullariga qaraganda domen haqida ko'proq ma'lumot talab qiladi.

Oldingi kuchaytirilgan o'qitish usullari faqat holatlar va mumkin bo'lgan harakatlardan bilan ta'minlangan bo'lsa, xususiyatga asoslangan o'rganishlar xususiyatlar bo'yicha qo'shimcha domen bilimlarini talab qiladi. Ushbu yondashuv xususiyatlarni diqqat bilan tanlashni talab qiladi; dizayner Q -funksiyani ifodalash uchun yetarli xususiyatlarni topishi kerak.

SARSA algoritmi chiziqli funksiyani yaqinlashtirish, *SARSA LFA*, Q -funkt-siyaga yaqinlashish uchun xususiyatlarning chiziqli funksiyasidan foydalanadi. U har bir misoldan keyin parametrlarni yangilab turuvchi stoxastik gradiyent tushish varianti bo'lgan o'sish gradiyentiga asoslangan. $F_1, \dots, F_n$ holat va harakatning raqamli xususiyatlari deb hisoblaylik. $F_i(s, a)$ holat s va harakat a uchun i -xususiyat qiymatini taqdim etadi. Ushbu xususiyatlar chiziqli Q -funksiyani ifodalash uchun ishlatiladi:

$$Q_{\bar{w}}(s, a) = w_0 + w_1 * F_1(s, a) + \dots + w_n * F_n(s, a)$$

Bu yerda, o'rganilishi kerak bo'lgan og'irliklar to'plami $\bar{w} = \langle w_0, w_1, \dots, w_n \rangle$ mavjud.

Faraz qilaylik, doimo qiymati 1 bo'lgan qo'shimcha xususiyat $F_0(s, a)$ mavjud, shuning uchun w_0 maxsus holat emas.

SARSA algoritmidagi tajriba $\langle s, a, r, s', a' \rangle$ shaklida bo'ladi (agent s holatida bo'lgan, a harakatini qilgan, r mukofotni olgan va s' holatiga kelgan, bu yerda a' harakatini tanlagan). Bu tajriba $Q_{\bar{w}}(s, a)$ ni yangilash uchun yangi baho $r + \gamma * Q_{\bar{w}}(s', a')$ ni beradi. Bu tajriba **chiziqli regressiya** uchun ma'lumot nuqtasi sifatida ishlatilishi mumkin.

$\delta = Q_{\bar{w}}(s, a) - (r + \gamma * Q_{\bar{w}}(s', a'))$ deb belgilaylik. Tenglama (7.4)ga muvofiq, og'irlik w_i quyidagicha yangilanadi:

$$w_i := w_i - \eta * \delta * F_i(s, a).$$

Keyin ushbu yangilanish SARSAga kiritilishi mumkin, bu 13.8-rasmda ko'rsatilgan algoritmnin beradi.

Ushbu dasturni amalga oshirish oddiy bo'lsa-da, **xususiyatga asoslangan o'rganishlar** – qanday xususiyatlarni o'z ichiga olishini tanlash – muhim vazifa hisoblanadi. Chiziqli funksiya nafaqat bajariladigan eng yaxshi harakatni ifodalashi kerak, balki kelajakdagi holatlar qanday foydali ekanligi haqida ma'lumot ham berishi kerak.

13.6-misol. 13.2-misolning maxluq o'yinini ko'rib chiqing. Domenni tushunish va uni faqat qora quti sifatida emas, balki ba'zi mumkin bo'lgan xususiyatlar-ni hisoblash va foydali bo'lishi mumkin bo'lgan xususiyatlar:

$F_1(s, a)$ qiymati 1 bo'lsa, a harakati agentni maxluq paydo bo'lishi mumkin bo'lgan joyga olib boradi va aks holda qiymati 0 bo'ladi.

$F_2(s,a)$: Agar a harakati agentni devorga olib borishi ehtimoli bo'lsa, qiymati 1, aks holda 0.

$F_3(s,a)$: Agar a qadam agentni mukofotga olib borishi ehtimoli bo'lsa, qiymati 1.

$F_4(s,a)$: Agar agent s holatida shikastlangan bo'lsa va a harakati uni ta'mirlash stansiyasiga olib boradigan bo'lsa, qiymati 1.

$F_5(s,a)$: Agar agent shikastlangan bo'lsa va a harakati uni maxluq paydo bo'lishi mumkin bo'lgan joyga olib borishi ehtimoli bo'lsa, qiymati 1, aks holda 0. Bu $F_1(s,a)$ bilan bir xil, lekin faqat agent shikastlangan bo'lsa amal qiladi.

$F_6(s,a)$: Agar agent s holatida shikastlangan bo'lsa, qiymati 1, aks holda 0.

$F_7(s,a)$: Agar agent s holatida shikastlanmagan bo'lsa, qiymati 1, aks holda 0.

$F_8(s,a)$: Agar agent shikastlangan bo'lsa va a yo'nalishida mukofot mavjud bo'lsa, qiymati 1.

$F_9(s,a)$: Agar agent shikastlanmagan bo'lsa va a yo'nalishida mukofot mavjud bo'lsa, qiymati 1.

$F_{10}(s,a)$: Agar s holatida P_0 joylashuvida mukofot bo'lsa, s holatidagi x -qiymatini oladi. Bu, ya'ni chap devordan masofa.

$F_{11}(s,a)$: Agar s holatida P_0 joylashuvida mukofot bo'lsa, qiymati $4-x$ bo'lib, x gorizontal joylashuvdir. Bu, ya'ni o'ng devordan masofa.

$F_{12}(s,a)$ dan $F_{29}(s,a)$ gacha bo'lgan to'rtta devorning har biridan mukofot joylashuvi va masofaning turli kombinatsiyalari uchun F_{10} va F_{11} ga o'xshaydi. Sovrin P_0 nuqtasida bo'lgan holatda, y masofa devorni hisobga olishi mumkin.

1: **controller** *SARSA LFA*(F, γ, η)

2: **Inputs**

3: $F = \{F_1, \dots, F_n\}$: a set of features. Define $F_0(s, a) = 1$.

4: $\gamma \in [0, 1]$: discount factor

5: $\eta > 0$: step size for gradient descent

6: **Local**

7: weights $w = \{w_0, \dots, w_n\}$, initialized arbitrarily

8: observe current state s

9: select action a

10: **repeat**

11: *do*(a)

12: observe reward r and state s

```

13: select action  $a_-$  (using a policy based on  $Q_-$ )
14:  $\delta := Qw(s, a) - (r + \gamma * Qw(s_-, a_-))$ 
15: for  $i = 0$  to  $n$  do
16:  $w_i := w_i - \eta * \delta * Fi(s, a)$ 
17:  $s := s_-$ 
18:  $a := a_-$ 
19: until termination()

```

13.8-rasm. SARSA LFA: chiziqli funksiyaning yaqinlashuvi bilan SARSA

Chiziqli funksiyaga misol:

$$\begin{aligned}
 Q(s, a) = & 2.0 - 1.0 * F_1(s, a) - 0.4 * F_2(s, a) - 1.3 * F_3(s, a) - \\
 & - 1.2 * F_5(s, a) - 1.6 * F_6(s, a) + 3.5 * F_7(s, a) + 0 \\
 & + 0.6 * F_9(s, a) - 0.0 * F_{10}(s, a) + 1.0 * F_{11}(s, a) +
 \end{aligned}$$

Bu SARSA LFA algoritmining 13.8-rasmida bir marta ishlashi uchun olingan qiymatlar (bir kasrgacha aniq).

AI Python (aipython.org) saytida ushbu maxluq o'yin uchun bu algoritmnin ochiq manbali Python implementatsiyasi mavjud. Algoritmni individual qadamlar bo'yicha sinab ko'ring, har bir qadam qanday qilib har bir parametрни yangilashini tushunishga harakat qiling. Endi uni bir necha qadam davomida ishga tushiring. 13.6-mavzuda ko'rsatilgan baholash o'lchovlari yordamida natijalarni ko'rib chiqing. O'rganilgan parametrlarning qiymatlarini tushunishga harakat qiling.

Ushbu algoritm hozirgi tajribalarga haddan tashqari moslashishga va eski tajribalarni unutishga moyil, shuning uchun u yaqin orada tashrif buyurmagan holat makonining bir qismiga qaytganida, u yana qayta o'rganishga majbur bo'ladi. Bu hodisa «**halokatli unutish**» deb ataladi. Bir modifikatsiya eski tajribalarni $((s, a, r, s')$ to'plamlarini) eslab qolish va ba'zi vazn yangilanishlarini amalga oshirish orqali **harakatlarni takrorlashning** ba'zi bosqichlarini bajarish. Og'irliklarni yangilash tajriba sodir bo'lganda amalda bo'lgan tamoyilga emas, balki joriy tamoyilga muvofiq tanlanishi kerak bo'lgan keyingi a amalidan foydalanishni talab qiladi. Xotira hajmi muammoga aylanganda, ba'zi eski tajribalar bekor qilinishi mumkin.

13.9.2. Lokal optimallashtirishdan qochish

Holatga asoslangan MDPlar va holatga asoslangan kuchaytirilgan o'qitish algoritmlari, masalan, Q -o'rganish, SARSA va modelga asoslangan kuchaytirilgan o'qitish algoritmlari mahalliy maksimumga ega emas, chunki har bir holat alohida optimallashtirilishi mumkin; bir holat uchun tamoyilni yaxshilash boshqa holatga salbiy ta'sir ko'rsatmaydi.

Biroq, umumlashtirish bo'lganda, bir holatni yaxshilash boshqa holatlarni yomonlashtirishi mumkin. Bu degani, algoritmlar eng yaxshi bo'lmagan qiymat bilan mahalliy optimaga yaqinlashishi mumkin. Mahalliy optimadan qochishning qandaydir yo'li mavjud bo'lganda ular yaxshiroq ishlashi mumkin. Mahalliy maksimaldan qochishning standart usuli tasodifiy algoritmlardan foydalanish, masalan, zarrachalarni filtrlashga o'xshash populyatsiyaga asoslangan usullardan foydalanish, bunda bir nechta boshlang'ich ishga tushirish parallel ravishda amalga oshiriladi va eng yaxshi tamoyil tanlanadi. **Evolutsion algoritmlar** yordamida topilgan ba'zi e'tiborga molik, ijodiy yechimlar mavjud bo'lib, bunda individual ishga tushirishlar genetik algoritim yordamida birlashtiriladi.

13.10. Ijtimoiy ta'sir

Kuchaytirilgan o'qitish (RL)ning real dunyoda qo'llanilishi bilan bog'liq asosiy muammolardan biri bu tasodiflar: agentning xatti-harakatlari natijasida kutilmagan yoki zararli ta'sirlar bo'lib, bu yomon dizayn natijasi bo'lishi mumkin.

Amodei va boshqalar [2016] RL agentlarini real dunyoda qo'llashdagi beshta asosiy muammoni ta'kidlaydi.

Salbiy yon ta'sirlardan qochish. Yon ta'sir – bu agentning harakati natijasida yuzaga keladigan kutilmagan hodisa. Masalan, yetkazib beruvchi robot gilamga qahva to'kib yubormasligi yoki mushuklarni bosib ketmasligi kerak, lekin binodan tashqarida yetkazib berishda chumolilarni bosib o'tishdan qocholmasligi mumkin. Agar siz yetkazib berishni iloji boricha tezroq bajarishni belgilasangiz, maqsadga erishish uchun robot ma'lum yon ta'sirlarga ega bo'lishi mumkin. Robotning noto'g'ri ishlashi uchun juda ko'p mumkin bo'lgan yo'llar mavjud, shuning uchun har bir vazifa uchun ularning barchasini ko'rsatib bo'lmaydi.

Xakerlik uchun mukofotdan qochish. Xakerlik uchun mukofot agent mukofotni kimningdir niyatiga zid tarzda optimallashtirganda sodir bo'ladi. Yetkazib beruvchi roboti yetkazilmagan xatlar qolmasligi uchun barcha xatlarni axlat qutisiga tashlashi mumkin yoki uning maqsadi tugallanmagan topshiriqlar qolmasligini ta'minlash bo'lsa, unga yangi vazifalar berilmasligi uchun yashirinishi mumkin. Chiqindilarni tozalash uchun mukofotlangan tozalash roboti tozalashi mumkin bo'lgan tartibsizlikni yaratish orqali eng katta mukofotni olishi mumkin.

Ko'lamli kuzatuv. Bu agent to'g'ri ish qilayotganiga ishonch hosil qilish uchun inson mukofot berayotganda sodir bo'ladi, lekin inson cheksiz fikr-mulohaza bera olmaydi. Yuqoridagi misollarda bo'lgani kabi, agent nimanidir noto'g'ri qilganda, uni tuzatish kerak. Ammo bunday nazorat doimiy bo'lishi mumkin emas; inson joylashtirilgan robotni doimiy ravishda baholashni xohlamaydi.

Xavfsiz tadqiq qilish. Qahva yetkazib berishning optimal yo'li xavfsiz bo'lsa ham, tadqiq qilish paytida u odamlarni urish yoki qahvani uloqtirish kabi xavfli harakatlarni sinab ko'rishi mumkin, shunchaki salbiy mukofot olish uchun. Agent-

ning harakatlari hatto tadqiqot paytida ham xavfsiz bo'lishi kerak; u xavfsizlik chegaralarida izlanish imkoniyatiga ega bo'lishi kerak.

Ta'minlashdagi taqsimot o'zgarishiga chidamlilik. Ishga tushirish paytidagi muhit agent o'qitilgan muhitdan shubhasiz farq qiladi. Muhitlar vaqt o'tishi bilan o'zgaradi va agentlar bu o'zgarishlarga bardosh bera olishlari yoki ularga tez moslashishlari kerak. Turli natijalar ehtimollik taqsimoti o'zgarishi mumkin va agent bu o'zgarishlarga bardosh bera olishi yoki ularga tez moslashishi kerak.

Amodei va boshqalar [2016] ushbu masalalarni ko'rib chiqqan boshqa tadqiqotchilarga havolalar beradi.

13.11. Takrorlash

Quyida siz ushbu bobdan o'rganishingiz kerak bo'lgan asosiy fikrlar keltirilgan:

Markov qaror jarayoni Kuchaytirilgan o'qitish uchun mos rasmiylikdir. Odatda qo'llaniladigan usul har bir holatda amal qilishning qiymatini $Q(S,A)$ funksiyasi orqali, taxmin qilishdir.

Kuchaytirilgan o'qitishida agent o'z bilimidan foydalanish va bilimini yaxshilash uchun tadqiq qilish o'rtasida muvozanatni saqlashi kerak.

Off-policy o'rganish (Tamoyildan tashqari o'rganish) (masalan, Q -o'rganish) optimal tamoyilning qiymatini o'rganadi. On-policy o'rganish (Tamoyil bo'yicha o'rganish) (masalan, SARSA) agent tomonidan haqiqiy bajarilayotgan tamoyilning (shu jumladan tadqiq qilish) qiymatini o'rganadi.

Modelga asoslangan kuchaytirilgan o'qitish dinamikalarini va mukofot modelarini o'rganishni qaror nazariyasi asosidagi rejalashtirishdan ajratadi.

Katta holat yoki harakat bo'shliqlari uchun mustahkamlovchi o'rganish algoritmлари qiymat funksiyasini, Q -funksiyani va/yoki tamoyilni ifodalash uchun neyron tarmoqlar kabi umumlashtruvchi o'quvchilardan foydalanish uchun mo'ljallangan bo'lishi mumkin.

13.12. Qo'shimcha o'qish uchun manbalar

Kuchaytirilgan o'qitishga kirish uchun Sutton va Barto [2018], Szepesvári [2010], Kochenderfer va boshq. [2022] va Powell [2022] asarlarini ko'ring. Silver va boshq. [2021] ko'p muammolarni mukofotni maksimallashtirish va kuchaytirilgan o'qitish doirasida ko'rish mumkinligini ta'kidlaydi. Langton [1997], De Jong [2006], Salimans va boshq. [2017], Such va boshq. [2017] evolyutsion hisoblashni, jumladan, uni kuchaytirilgan o'qitishda qanday ishlatilishini ko'rib chiqadi. Stanley va boshq. [2019] **neyro-evolyutsiyani** ko'rib chiqadi. Such va boshq. [2017] genetik algoritmlar kuchaytirilgan o'qitish algoritmлари uchun qiyin bo'lishi mumkinligini ko'rsatadi. Lehman va boshq.

[2018] evolyutsion algoritmlarning kreativligi haqida ko'plab misollar keltiradi. Vaqtinchalik-farqni o'rganish Sutton [1988] tomonidan yozilgan. *Q*-o'rganish Watkins va Dayan [1992] tomonidan yozilgan.

Bandit muammolari uchun yuqori ishonch oralig'ini ishlatish Auer va boshq. [2002] tomonidan tahlil qilingan. Russo va boshq. [2018] Thompson [1933] namunasini o'rgatish bo'yicha qo'llanma beradi. Kearns va boshq. [2002] daraxt qidiruvini tahlil qiladi, Kocsis va Szepesvári [2006] esa daraxt qidiruvini aqlli izlanish bilan birlashtiradi.

Schmidhuber [1990] neyron tarmoqlar bir vaqtning o'zida model va tamoyilni qanday o'rganishi mumkinligini ko'rsatadi. Mnih va boshq. [2015] kuchaytirilgan o'qitishning neyron tarmoqlar bilan birlashtirilishi klassik Atari kompyuter o'yinlarini qanday hal qilganini tavsiflaydi. Silver va boshq. [2016] o'rganish Go o'yini uchun qanday ishlatilishi mumkinligini ko'rsatadi va Silver va boshq. [2017] AlphaZero shaxmat va shogi o'yinlarida ham insondan ustun natijalarni qayd etganini tavsiflaydi. François-Lavet va boshq. [2018], Li [2018] chuqur kuchaytirilgan o'qitishni ko'rib chiqadi. Ijtimoiy ta'sir bo'limi Amodei va boshq. [2016] asariga asoslangan.

13.13. Mashqlar

13.1-mashq. *Q*-o'rganish agent arxitekturasining 2.1.1-mavzu bilan qanday mos kelishini tushuntiring. Faraz qiling, *Q*-o'rganish agenti chegirma omili γ , qadam hajmi α va bir-greedy izlanish strategiyasini amalga oshirmoqda.

- (a) *Q*-o'rganish agentining ishonch holatining komponentlari qanday?
- (b) Qabul qilish elementlari nima?
- (c) *Q*-o'rganish agentining buyruq funksiyasi nima?
- (d) *Q*-o'rganish agentining ishonch holatini o'tkazish funksiyasi nima?

13.2-mashq. *Q*-o'rganish agenti, o'zgarmas α va chegirma γ bilan, 34-holatda edi, 7-amalni bajardi, 3-mukofot oldi va 65-holatga tushdi. Qaysi qiymat(lar) yangilanadi? Yangi qiymat uchun ifoda bering. (Iloji boricha aniqroq bo'lsin.)

13.3-mashq. 13.4-rasmdagi vaqtga nisbatan umumiy mukofotning grafikasi uchun, minimum va nol kesish statistikalar faqat ijobiy va salbiy mukofotlarni muvozanatlashtirish oqilona bo'lsa ma'noli bo'ladi. Agar nol mukofot oqilona xatti-harakatlarning to'g'ri ta'rifi bo'lmasa, ushbu statistikani nima bilan almashtirish kerakligini taklif qiling. [Maslahat: Faqat ijobiy mukofot yoki faqat salbiy mukofot hollari haqida o'ylang.]

13.4-mashq. 13.2-misol (AIPython (aipython.org)dagi «maxluq o'yini») uchun *Q*-o'rganishning turli parametr sozlamalarini solishtiring. Xususan, quyidagi holatlarni solishtiring:

- (i) *qadam hajmi* (c) = $1/c$ va *Q*-qiymatlar 0,0 ga teng bo'lgan holda boshlanadi.
- (ii) *qadam hajmi* (c) = $10/(9+c)$ o'zgaradi va *Q*-qiymatlar 0,0 ga teng bo'lgan holda boshlanadi.

(iii) α o'zgaradi (i va ii variantlaridan qaysi biri yaxshi bo'lsa foydalaning) va Q -qiymatlar 5,0 ga teng bo'lgan holda boshlanadi.

(iv) α 0,1 ga o'zgaras qilib belgilangan va Q -qiymatlar 0,0 ga teng bo'lgan holda boshlanadi.

(v) α 0,1 ga o'zgaras qilib belgilangan va Q -qiymatlar 5,0 ga teng bo'lgan holda boshlanadi.

(vi) Boshqa parametr sozlamalari.

Ushbu holatlarning har biri uchun bir nechta bajarishlarni amalga oshiring va quyidagilarni solishtiring:

(a) minimal qiymatlar taqsimoti

(b) nol kesish

(c) tadqiq qilish tamoyilini o'z ichiga olgan tamoyil uchun asimptotik nishab

(d) tadqiq qilish tamoyilini o'z ichiga olmaydigan tamoyil uchun asimptotik nishab (buni sinash uchun, algoritm izlagandan so'ng, foydalanish parametrini 100% ga sozlang va qo'shimcha qadamlarni bajarang).

Qaysi sozlamalarni tavsiya qilasiz? Nima uchun?

13.5-mashq. Quyidagi kuchaytirilgan o'qitish algoritmlari uchun:

(i) O'zgaras α va 80% foydalanish bilan Q -o'rganish.

(ii) $ak = 1/k$ va 80% foydalanish bilan Q -o'rganish.

(iii) $ak = 1/k$ va 100% foydalanish bilan Q -o'rganish.

(iv) $ak = 1/k$ va 80% foydalanish bilan SARSA o'rganish.

(v) $ak = 1/k$ va 100% foydalanish bilan SARSA o'rganish.

(vi) Xususiyatga asoslangan SARSA o'rganish va softmax harakat tanlovi bilan.

(vii) Modelga asoslangan mustahkamlashni o'rganuvchi va 50% foydalanish bilan.

(a) Qaysi kuchaytirilgan o'qitish algoritmlari yetarlicha vaqt berilganida optimal tamoyilni topadi?

(b) Qaysilari haqiqatan ham optimal tamoyilga amal qiladi?

13.6-mashq. Q -o'rganishda ak qiymatini k dan olishning to'rtta turli usuli ni ko'rib chiqing (eslatma: ak o'zgaruvchi bo'lganda, har bir holat-harakat jufti uchun alohida k bo'lishi kerak).

$ak = 1/kbo'lsin.$

$ak = 10/(9+k)$ bo'lsin.

$ak = 0.1$ bo'lsin.

$ak =$ birinchi 10,000 qadam uchun 0.1, keyingi 10,000 qadam uchun 0.01, keyingi 10,000 qadam uchun 0.001, keyingi 10,000 qadam uchun 0.0001 va hokazo bo'lsin.

(a) Qaysi usul nazariyada haqiqiy Q -qiymatga yaqinlashadi?

(b) Qaysi usul amaliyotda haqiqiy Q -qiymatga yaqinlashadi (masalan, maqbul qadamlar sonida)? Uni bir nechta domen uchun sinab ko'ring.

(c) Qaysi usul atrof-muhit sekin o'zgarganda moslashish qobiliyatiga ega?

13.7-mashq. Modelga asoslangan mustahkamlovchi o'rganish noaniqlik sharoitida optimizmning boshqa shaklini beradi. Algoritm har bir holatni «nirvana» holatiga o'tish bilan boshlashi mumkin, bu juda yuqori Q -qiymatga ega (lekin amalda unga hech qachon yetib borilmaydi, shuning uchun ehtimollik nolga qisqaradi).

(a) Bu barcha Q -qiymatlarni yuqori qiymatga sozlashdan boshqacha ishlaydimi? Bu yaxshiroqmi, yomonmi yoki bir xilmi?

(b) Nirvana holati uchun Q -qiymat qanchalik yuqori bo'lishi kerak? Qaysi qiymat yaxshi bo'lishi mumkinligini taklif qiling va sinab ko'ring.

(c) Bu usul boshqa RL algoritmlarida ishlatilishi mumkinmi? Qanday qilib yoki nima uchun Shunday emasligini tushuntiring.

13.8-mashq. 13.6-misoldagi matritsali o'yinida joriy xazinagacha bo'lgan x -masofasini va joriy xazinagacha bo'lgan y -masofasini aniqlash funksiyalarini o'z ichiga oladi. Kris bu xususiyatlar amalga bog'liq bo'lmagani uchun foydali emas deb o'yladi. Ushbu xususiyatlar farq qiladimi? Nima uchun farq qilishi yoki qilmasligi mumkinligini tushuntiring. Ular amalda farq qiladimi?

13.9-mashq. SARSAda chiziqli funktsiyani taxmin qilishda, $r + \gamma Q_{\pi}(s', a') - Q_{\pi}(s, a)$ ni minimallashtirish uchun chiziqli regressiyadan foydalanish 13.8-rasmdagi algoritmdan boshqacha algoritmni beradi. Nima olishingizni tushuntiring va matnda tavsiflangan narsa nima uchun afzal bo'lishi mumkin (yoki bo'lmاسligi mumkinligini) tushuntiring. [Maslahat: og'irliklar nima uchun yaxshiroq baholash uchun moslashtirilishi kerak?]

13.10-mashq. 13.6-misolda ba'zi xususiyatlar mukammal korrelyatsiyaga ega (masalan, F_6 va F_7). Ushbu korrelyatsiyalangan xususiyatlar funksiyalarni ifodalash qobiliyatiga ta'sir qiladimi? Bu o'rganish tezligiga qanday ta'sir qiladi? Buni ba'zi misollarda empirik tarzda sinab ko'ring.

14-bob. KO'P AGENTLI TIZIMLAR

Tasavvur qiling, shaxsiy dasturiy ta'minot agenti sizning nomingizdan elektron tijorat bilan shug'ullanadi. Faraz qilaylik, bu agentning vazifasi vaqt o'tishi bilan turli onlayn savdo maydonlarida sotuvga qo'yilgan mahsulotlarni kuzatish va ularning ba'zilarini sizning nomingizdan jozibador narxda sotib olishdir. Muvaffaqiyatga erishish uchun, agentingiz sizning mahsulotlarga bo'lgan xohish-istaklaringiz, byudjetingizni va umumiy holda u ishlaydigan muhit haqidagi bilimlaringizni o'z ichiga olishi kerak. Bundan tashqari, agent u bilan hamkorlik qiladigan boshqa shunga o'xshash agentlar (masalan, kimoshdi savdosida u bilan raqobatlashishi mumkin bo'lgan yoki do'kon egalarini ifodalovchi agentlar) haqidagi bilimlaringizni, jumladan, o'z imtiyozlari va bilimlarini o'zida mujassamlashtirishi kerak bo'ladi. Bunday vositalar to'plami ko'p agentli tizimni hosil qiladi.

– Yoav Shoham va Kevin Leyton-Braun [2008]

O'z maqsadlari va xohish-istaklariga ega bo'lgan, nima qilish haqida mulohaza yuritadigan boshqa agentlar mavjud bo'lganda, agent nima qilishi kerak? Aqlli agent boshqa agentlarni e'tiborsiz qoldirmasligi yoki ularni muhitdagi shovqin sifatida ko'rmasligi kerak. Ushbu bobda o'z foydalariga ega bo'lgan boshqa agentlarni ham o'z ichiga olgan muhitda agent nima qilishi kerakligini aniqlash muammolari ko'rib chiqiladi.

14.1. Ko'p agentli tizimlar

Ushbu bobda bir nechta agentlarni o'z ichiga olgan muhitlar ko'rib chiqiladi va ularda quyidagi taxminlar mavjud:

Agentlar mustaqil ravishda harakat qilishlari mumkin, ularning har biri dunyo va boshqa agentlar haqidagi o'z ma'lumotlariga ega.

Natija barcha agentlarning harakatlariga bog'liq bo'lishi mumkin.

Har bir agentning natijaga bog'liq o'ziga xos foydasi bor. Agentlar o'z foydalarini maksimal darajaga yetkazish uchun harakat qilishadi.

Mexanizm har bir agent uchun qanday harakatlar mavjudligini va agentlarning harakatlari yaxshi natijalarga qanday olib kelishini belgilaydi. Agent o'z maqsadlari yoki foydalariga asoslanib, nima qilishni strategik ravishda hal qilgan-da harakat qiladi.

Ba'zan **tabiat** agent sifatida ko'riladi. Tabiat hech qanday afzalliklarga ega bo'lmagan va strategik ravishda harakat qilmaydigan maxsus agent sifatida ta'riflanadi. U shunchaki stoxastik tarzda ta'sir qiladi. 1.4-rasmda ko'rsatilgan agent arxitekturasini nuqtaiy nazaridan, tabiat va boshqa agentlar agent uchun muhitni

tashkil qiladi. Strategik harakat qilmaydigan agentlar tabiatning bir qismi sifatida ko'riladi. Strategik agent boshqa strategik agentlarni tabiatning bir qismi sifatida ko'rmasligi kerak, balki boshqa strategik agentlar bilan muvofiqlashtirish, hamkorlik va ehtimoliy muzokaralarga tayyor bo'lishi kerak.

Ko'p agentli tizimlarni o'rganishda ikkita ekstremal holat mavjud:

to'liq **hamkorlikda**, agentlar bir xil foyda funksiyasini baham ko'radi;

to'liq **raqobatdosh**, bir agent faqat boshqa agent yutqazganda yutishi mumkin;

nol-yig'indi o'yinlarda, har bir natija uchun agentlarning foydalari yig'indisi nolga teng.

Ko'pgina o'zaro ta'sirlar bu ikki ekstremalning orasida bo'lib, agentlarning foydalari ba'zi jihatlarida sinergetik, ba'zi jihatlarida raqobatdosh va boshqa jihatlarida mustaqildir. Masalan, do'konlari bir-biriga qo'shni bo'lgan ikkita tijorat agenti ko'cha hududini toza va jozibali qilish maqsadiga ega bo'lishi mumkin; ular mijozlar uchun raqobatlashishlari mumkin, lekin boshqa agent do'konining tafsilotlari bo'yicha imtiyozlarga ega bo'lmasligi mumkin. Ba'zan ularning harakatlari bir-biriga xalaqit bermaydi, ba'zan esa xalaqit beradi.

Ko'pincha agentlar hamkorlik va muzokaralar orqali o'z harakatlarini muvofiqlashtirsalar, yaxshiroq bo'ladi.

Ko'p agentli o'zaro ta'sirlar asosan Neumann va Morgenstern [1953] tomonidan olib borilgan ishlarga asoslangan o'yinlar terminologiyasidan foydalangan holda o'rganilgan. Agentlar o'rtasidagi o'zaro ta'sirlar muammolarini o'yinlar nuqtayi nazaridan o'rganish mumkin. Hatto kichik o'yinlar ham chuqur muammolarni yoritadi. Biroq, o'yinlarni o'rganish nafaqat sun'iy o'yinlar, balki umumiy ko'p agentli o'zaro ta'sirlar haqida bo'lishi kerak.

Ko'p agentli tizimlar sun'iy intellektli robotlar futbolida, interaktiv video-o'yinlarda keng tarqalgan bo'lib, murakkab iqtisodiy tizimlarda faoliyat yurituvchi agentlar uchun o'yinlar sun'iy intellektning ajralmas qismidir. O'yinlar Sining dastlabki ilovalari bo'lgan. Birinchi operatsion shashka dasturi 1952-yilga borib taqaladi. Samuel [1959] tomonidan ishlab chiqilgan dastur 1961-yilda Konnektikut shtati shashka chempionini mag'lub etgan. 1997-yilda Deep Blue [Campbell va boshq., 2002] jahon shaxmat chempionini mag'lub etganida va 2016-yilda AlphaGo [Silver va boshq., 2016] dunyoning eng yaxshi Go o'yinchilaridan birini mag'lub etganida katta shov-shuv bo'lgan. Garchi katta bo'lsa-da, bu o'yinlar konseptual jihatdan oddiy, chunki agentlar dunyo holatini mukammal kuzatishadi. Ko'pgina real dunyoning o'zaro ta'sirlarida, dunyo holati faqat qisman kuzatiladi. Hozirgi vaqtda poker kabi qisman kuzatiladigan o'yinlarga katta qiziqish bor, bu yerda atrof-muhit bashorat qilinadi (kartalarning nisbati ma'lum, garchi berilgan kartalar noma'lum bo'lsa ham) va robot futboli, bu yerda atrof-muhit ancha kam bashorat qilinadi. Ammo bu o'yinlarning barchasi insonlarning kundalik hayotlarida amalga oshiradigan ko'p agentli o'zaro ta'sirlardan ancha soddaroq, bozorlarda yoki internetda savdolashish uchun zarur bo'lgan strategiyalar haqida gapirma-

sa ham bo'ladi, bu yerda qoidalar kamroq aniqlangan va foydaliliklar ancha ko'p qirrali bo'ladi.

14.2. O'yinlarni ifodalash

Mexanizm har bir agent uchun mavjud bo'lgan harakatlarni va ularning birgalikdagi harakatlari natijalarini (taqsimotini) ifodalaydi. Iqtisodiyot va sun'iy intellektida (SI) o'yinlar va ko'p agentli o'zaro aloqalarni ifodalash uchun ko'plab usullar taklif qilingan. SIDA, ushbu tasvirlar odatda dizaynerga hisoblash samaradorligini oshirish uchun o'yinlarning ayrim jihatlarini modellashtirish imkonini beradi.

Uchta ifodalanish usuli taqdim etiladi; ulardan ikkitasi iqtisodiyotdan olingan klassik ifodalardir. Birinchisi, agentlar siyosatining butun tuzilishini mavhumlashtiradi. Ikkinchisi o'yinlarning ketma-ket tuzilishini modellashtiradi va stol o'yinlarini ifodalash bo'yicha ko'pgina ishlarning asosidir. Uchinchisi holatga asoslangan ifodalanishdan voz kechib, o'yinlarni xususiyatlar nuqtayi nazaridan ifodalash imkonini beradi.

14.2.1. Normal shakldagi o'yinlar

O'yinlarning eng asosiy ifodalanishi normal shakldagi o'yin yoki strategik shakldagi o'yin deb nomlanadi. Normal shakldagi o'yin quyidagilardan iborat:

Agentlarning cheklangan I to'plami, odatda $I = \{1, \dots, n\}$ butun sonlar bilan belgilangan.

Har bir agent uchun harakatlar to'plami A_i , bu yerda $i \in I$.

Har bir agent uchun foyda funksiyasi u_i , bu berilgan agentlarning har biriga amal tayinlanganda, i agenti uchun kutilgan foydani qaytaradi; har bir agent o'z foydasini maksimal darajada oshirishga harakat qiladi.

Harakatlar profili – bu $(a_1, \dots, a_n)$ bo'lib, unda $i \in I$ agenti a_i harakatini amalga oshiradi, bu yerda $a_i \in A_i$. Harakatlar profili **natija** beradi. Har bir agent har bir natija ustidan foydaga ega. Agent uchun foyda uning qiziqqan hamma narsasini, shu jumladan adolat, altruizm va jamiyat farovonligini o'z ichiga olishi kerak.

14.1-misol. «Tosh-qog'oz-qaychi» o'yini bolalar o'ynaydigan odatiy o'yindir va hatto dunyo chempionati ham mavjud. Faraz qilaylik, ikkita agent (o'yinchi), Alisa va Bob bor. Har bir agent uchun uchta harakat mavjud, shuning uchun

$$A_{\text{Alisa}} = A_{\text{Bob}} = \{\text{rock, paper, scissors}\}.$$

Alisa va Bob uchun har bir harakatlar kombinatsiyasi uchun Alisa va Bob uchun foyda mavjud. Bu 14.1-rasmdagi jadvalda ko'rsatiladi. Bu jadval **to'lov matritsasi** deb ataladi. Bir vaqtda Alisa bir qatorni, Bob esa ustunni tanlaydi. Bu sonlar juftligini beradi: birinchi raqam qator o'yinchisi (Alisa) uchun to'lovni va ikkinchi raqam ustun o'yinchisi (Bob) uchun to'lovni beradi. Har birining foydasi nima qilganlariga bog'liq ekanligiga e'tibor bering. Bir ha-

rakatlar profili misoli $\{scissors_{Alice}, rock_{Bob}\}$, bo'lib, unda Alisa qaychi, Bob esa toshni tanlaydi. Bu harakatlar profilida Alisa -1 va Bob 1 foyda oladi. Bu o'yin nol-summa o'yini, chunki bir kishi faqat boshqasi yutqazganda yutadi.

Bu o'yinning ifodasi juda cheklangandek ko'rinishi mumkin, chunki har bir agent tomonidan bir vaqtning o'zida tanlangan yagona harakatlarga asoslangan bir martalik to'lovni beradi. Biroq, ta'rifdagi harakatning talqini juda umumiydir.

Odatda, «harakat» shunchaki oddiy tanlov emas, balki **strategiya**: agentning turli sharoitlarda nima qilishi kerakligini belgilashdir. Normal shakl, asosan, agentlarning qilishi mumkin bo'lgan strategiyalarini inobatga olgan holda foydani belgilashdir. Shuning uchun uni o'yinning strategik shakli deb atashadi.

Umuman olganda, normal shakldagi o'yin ta'rifdagi «harakat» agent uchun **boshqaruvchi** bo'lishi mumkin. Shunday qilib, har bir agent boshqaruvchini tanlaydi va foyda har bir agent uchun muhitda boshqaruvchilar ishlashining kutilgan natijasini beradi. Keyingi misollar oddiy harakatlar uchun bo'lsa-da, umumiy holda har bir agent uchun juda ko'p miqdordagi mumkin bo'lgan harakatlar (mumkin bo'lgan boshqaruvchilar) mavjud.

14.2.2. O'yinning keng ko'lamli shakli

O'yinning normal shakli boshqaruvchilarni yagona birliklar sifatida ifodalagan bo'lsa, o'yinning vaqt davomida qanday rivojlanishini aniqlash ko'pincha tabiiyroq bo'ladi. O'yinlarning keng ko'lamli shakli – bu yagona agentli **qarorlar daraxtining** kengaytmasidir. Keling, o'yin to'liq kuzatiladigan (o'yin nazariyasi-da mukammal ma'lumot) deb faraz qilaylik.

Mukammal ma'lumotli keng ko'lamli o'yin yoki o'yin daraxti, ichki tugunlar holatlari va yoylar agentlar tomonidan amalga oshirilgan harakatlarga mos keladigan cheklangan daraxtdir.

Xususan:

Har bir ichki tugun agent (yoki *tabiat*) bilan belgilangan. Agent tugunni boshqaradi deb aytiladi.

	rock	paper	scissors
rock	0,0	-1,1	1,-1
paper	1,-1	0,0	-1,1
scissors	-1,1	1,-1	0,0

14.1-rasm. Tosh-qog'oz-qaychi o'yini uchun normal shakl

Agent i bilan belgilangan tugunning har bir yoyi agent i uchun harakatga to'g'ri keladi.

Tabiat bilan belgilangan har bir ichki tugun uning avlodlari ustidan ehtimollik taqsimotiga ega.

Barglar oxirgi natijalarni ifodalaydi va har bir agent uchun foyda bilan belgilangan.

O'yinlarning keng ko'lamli shakli o'yinning aniq rivojlanishini belgilaydi. Bargga bo'lgan har bir yo'l harakatlanish deb ataladi, agentlarning tanlovi va tabiatiga bog'liq ravishda o'yin qanday rivojlanishi mumkinligini aniq bir tarzda ko'rsatadi.

i agenti uchun **strategiya** i agenti tomonidan boshqariladigan tugunlardan harakatlarga o'tuvchi funksiyadir. Ya'ni strategiya i agenti boshqaradigan har bir tugun uchun keyingi avlodni tanlaydi. **Strategiya profili** har bir agent uchun strategiyadan iborat.

i agenti uchun strategiya, i agenti tomonidan boshqariladigan tugunlardan harakatlarga o'tish funksiyasidir. Ya'ni strategiya i agenti boshqaradigan har bir tugun uchun keyingi bolani tanlaydi. Strategiya profili har bir agent uchun strategiyadan iborat.

14.2-misol. Ikki agent Endi va Barb bo'lgan almashinuv o'yinini ko'rib chiqamiz va ular bir-biriga o'xshash ikki narsani taqsimlashlari kerak. Endi avval ularni qanday taqsimlashni tanlaydi: Endi ikkala obyektни ham o'zida saqlab qoladi, ular ulashadi va har bir kishi bir obyektни oladi yoki u ikkala obyektни Barbga beradi. Keyin Barb taqsimotни rad etadi, ikkalasi ham hech narsa olmaydi yoki taqsimotни qabul qiladi va ikkalasi ham taqsimlangan miqdorni oladi.

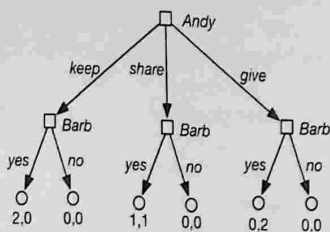
Almashinuv o'yinining keng ko'lamli shakli 14.2-rasmda ko'rsatilgan. Endida 3 ta strategiya bor. Barb $2^3 = 8$ ta strategiyaga ega; u boshqaradigan tugun tayinlashlarining har bir kombinatsiyasi uchun bittadan. Natijada 24 ta strategiya profili paydo bo'ladi.

Strategiya profili berilganida, har bir tugunda har bir agent uchun foyda mavjud. Tugundagi agent uchun foyda pastdan yuqoriga rekursiv ravishda aniqlanadi:

Bargdagi har bir agent uchun foyda bargning bir qismi sifatida berilgan.

Agentning o'zi boshqaradigan tugundagi foydalilik, agentning strategiyasi tanlangan avlod tugunining foydaliligidir.

Boshqa i agent tomonidan boshqariladigan tugundagi j agentning foydaliligi i agent strategiyasi tomonidan tanlangan kichik tugunning j agenti uchun foydaliligidir.



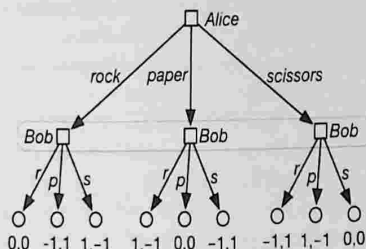
14.2-rasm. Almashish o'yining keng ko'lamli shakli

Tabiat tomonidan boshqariladigan tugundagi i agent uchun foyda avlodning foydasining kutilgan qiymatidir. Ya'ni $u_i(n) = \sum_c P(c) u_i(c)$ bu yerda n tugunning cavlodlari yig'indisi va $P(c)$ – tabiat c avlodni tanlashi ehtimolligi.

14.3-misol. Almashish o'yinida, quyidagi strategiya profilini ko'rib chiqing: Endi, Barb nima qilishini bilmagan holda, *saqlashni* tanlaydi. Barb har bir tugun uchun «yo'q», «ha», «ha»ni tanlaydi. Ushbu strategiya profilida, eng chap ichki tugunda Endi uchun foyda 0, markaziy ichki tugunda Endi uchun foyda 1 va eng o'ng ichki tugunda Endi uchun foyda 0. Ushbu strategiya profilida, ildizda Endi uchun foyda 0.

O'yinning keng ko'lamli shakli ta'rifining oldingi qismi agentlar dunyo holatini kuzata olishini taxmin qiladi (ya'ni har bir bosqichda ular qaysi tugunda ekanligini bilishadi). Bu o'yinning holati to'liq kuzatilishi kerakligini anglatadi. **Qisman kuzatiladigan o'yin** yoki **mukammal bo'lmagan ma'lumotli o'yinda**, agentlar nima qilish kerakligini hal qilishi kerak bo'lgan vaqtda dunyo holatini bilmasliklari mumkin. Bu **bir vaqtning o'zida** bir nechta agentlar nima qilish kerakligini hal qilishi kerak bo'lgan o'yinlarni ham o'z ichiga oladi. Bunday hollarda, agentlar o'yin daraxtidagi qaysi tugunda ekanligini bilishmaydi. Ushbu o'yinlarni modellashirish uchun o'yinlarning keng ko'rinishi ma'lumot to'plamlarini o'z ichiga olgan holda kengaytiriladi. **Ma'lumot to'plami** bitta agent tomonidan boshqariladigan va mavjud harakatlarning bir xil to'plamiga ega bo'lgan tugunlar to'plamidir. Gap shundaki, agent ma'lumot to'plamidagi elementlarni ajrata olmaydi. Agent faqat o'yin holati ma'lumot to'plamidagi tugunlardan birida ekanligini biladi, lekin qaysi tugunda ekanligini emas. Strategiyada agent har bir ma'lumot to'plami uchun bir harakatni tanlaydi; ma'lumot to'plamidagi har bir tugunda bir xil harakat amalga oshiriladi. Shunday qilib, keng ko'lamli shaklda strategiya ma'lumot to'plamlaridan harakatlarga o'tuvchi funksiyani belgilaydi.

14.4-misol. 14.3-rasmda 14.1-misoldagi tosh-qog'oz-qaychi o'yinining keng ko'rinishi keltirilgan. Ma'lumot to'plamining elementlari yumaloq to'rtburchak ichida. Strategiyada, Bob ma'lumotlar to'plamidagi har bir tugunga bir xil munosabatda bo'lishi kerak. Bob o'z harakatini tanlaganda, Alisa qaysi harakatni tanlaganini bilmaydi.



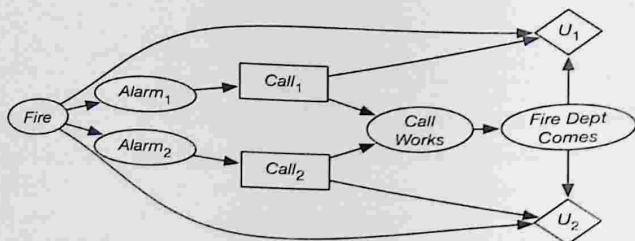
14.3-rasm. Tosh-qog'oz-qaychi o'yinining keng ko'lamli shakli

14.2.3. Ko'p agentli qaror tarmoqlari

O'yinning keng ko'lamli shakli holatga asoslangan o'yinning ifodasidir. Ko'pincha holatlarni xususiyatlar nuqtayi nazaridan tavsiflash ancha qisqacha bo'ladi. Ko'p agentli qaror tarmog'i bu ko'p agentli qaror muammosining faktori-zatsiyalangan ifodasidir. Bu qaror tarmog'iga o'xshaydi, faqat har bir qaror tuguni agent bilan belgilangan bo'lib, agent tugun uchun qiymatni tanlash imkoniga ega bo'ladi. Har bir agent uchun foyda tuguni mavjud bo'lib, agent uchun foydani belgilaydi. Qaror tugunining asl to'plamlari agentning harakat qilishi kerak bo'lganida mavjud bo'ladigan ma'lumotni belgilaydi.

14.5-misol. 14.4-rasmda yong'in signalizatsiyasi misoli uchun ko'p agentli qaror tarmog'i keltirilgan. Ushbu ssenariyda ikkita agent bor, Agent 1 va Agent 2. Har birining yong'in bor-yo'qligini aniqlaydigan o'zining shovqinli sensori bor. Biroq, agar ikkalasi ham qo'ng'iroq qilsalar, ularning qo'ng'iroqlari bir-biriga xalaqit qilishi va ikkalasi ham ishlamasligi mumkin. Agent 1 qaror o'zgaruvchisi $Call_1$ uchun qiymatni tanlash imkoniga ega va faqat $Alarm_1$ o'zgaruvchisi uchun qiymatni kuzatadi. Agent 2 qaror o'zgaruvchisi $Call_2$ uchun qiymatni tanlash imkoniga ega va faqat $Alarm_2$ o'zgaruvchisi uchun qiymatni kuzatadi. Qo'ng'iroq ishlash-ishlamasligi $Call_1$ va $Call_2$ qiymatlariga bog'liq. Yong'in bo'limi kelish-kelmasligi qo'ng'iroq ishlash-ishlamasligiga bog'liq. Agent 1 uchun foyda yong'in bor-yo'qligiga, yong'in bo'limining kelishiga va ularning qo'ng'iroq qilgan-qilmaganiga bog'liq – xuddi shunday Agent 2 uchun ham.

Ko'p agentli qaror tarmog'ini normal shakldagi o'yinga aylantirish mumkin; ammo, strategiyalar soni juda katta bo'lishi mumkin. Agar qaror o'zgaruvchisi d holatga va n ikkilik asl to'plamlari holatiga ega bo'lsa, asl to'plamlar uchun qiymatlar tayinlanishining 2^n kombinatsiyasi mavjud bo'ladi va shunday qilib d^{2^n} strategiyalar mavjud bo'ladi. Bu faqat bitta qaror tuguni uchun; yanada murakkab tarmoqlar normal shaklga aylantirilganda yanada kattaroq bo'ladi. Shunday qilib, strategiyalarni sanashga asoslangan algoritmlar eng kichik ko'p agentli qaror tarmoqlari uchun ham amaliy emas.



14.4-rasm. 14.5-misol uchun ko'p agentli qarorlar tarmog'i

Boshqa ifodalar ko'p agentli sozlamalarda turli tuzilmalardan foydalanadi. Masalan, agentning foydasi boshqa agentlarning biron-bir harakatni bajarishiga bog'liq bo'lishi mumkin, lekin ularning shaxsiyatlariga bog'liq emas. Agentning foydasi faqat bir necha boshqa agentlarning nima qilishiga bog'liq bo'lishi mumkin, boshqa agentlarning harakatlariga to'g'ridan to'g'ri bog'liq emas. Agentning foydasi faqat qo'shni joylardagi agentlarning nima qilishiga bog'liq bo'lishi mumkin va bu agentlarning shaxsiyatlariga yoki boshqa agentlarning nima qilishiga bog'liq emas.

14.3. Mukammal ma'lumotli o'yinlarni hal etish

To'liq kuzatiladigan ko'p agentli tizimlar odatda **mukammal ma'lumotli** deb ataladi. Mukammal ma'lumotli o'yinlarda agentlar ketma-ket harakat qiladi va agent harakat qilishi kerak bo'lganda, u dunyoning holatini kuzatish imkoniga ega bo'ladi. Har bir agent o'z foydasini maksimal darajada oshirish uchun harakat qiladi.

Mukammal ma'lumotli o'yinni keng ko'lamli shakldagi o'yin sifatida ifodalash mumkin, bu yerda barcha ma'lumot to'plamlari bitta tugundan iborat. Ular, shuningdek, ko'p agentli qaror tarmog'i sifatida ifodalanishi mumkin, bu yerda qaror tugunlari to'liq tartiblangan va har bir qaror tugunining asl to'plamlari ol-

dingi qaror tuguni va ularning barcha asl to'plamlarini o'z ichiga oladi (shuning uchun ular no-forgetting (unutmaydigan) qaror tarmoqlarining ko'p agentli namunasidir).

Mukammal ma'lumotli o'yinlar to'liq kuzatiladigan yagona agentli tizimlarga o'xshash tarzda yechilishi mumkin. Ular oxirgi qarorlardan birinchisigacha orqaga qarab, dinamik dasturlash yordamida yoki oldinga qarab qidiruv yordamida yechilishi mumkin. Ko'p agentli algoritmlar har bir agent uchun foydani saqlaydi va har bir harakat uchun harakatni tanlaydi, bu harakatni amalga oshirayotgan agentning foydasini maksimal darajada oshiradi. Dinamik dasturlash varianti, **teskari induksiya** deb ataladi, o'yinning oxiridan boshlanadi, qiymatlarni hisoblaydi va har bir tugun uchun har bir agent uchun reja tuzadi va kesh qiladi.

14.5-rasmda mukammal ma'lumotli o'yin uchun o'yin daraxtini baholash bo'yicha yuqoridan pastga, chuqurlik-birinchi qidiruv algoritmi keltirilgan. Bu butun daraxtni sanab chiqadi. Har bir ichki tugunda, tugunni boshqaradigan agent o'zi uchun eng yaxshi avlodni tanlaydi. Bu agentning ballini maksimal darajada oshiradigan birinchi avlodni ixtiyoriy ravishda tanlaydi (> 12 -qatorda), lekin quyidagi misol ko'rsatganidek, qaysi biri tanlanishi foydaga ta'sir qilishi mumkin.

14.6-misol. 14.2-rasmdagi almashish o'yinini ko'rib chiqamiz. Rekursiv qidiruvlarda, Barb o'z foydasini maksimal darajada oshiradigan qiymatni tanlaydi. Shunday qilib, u o'zi boshqaradigan o'ng ikki tugun uchun «ha»ni va chapdagi tugun uchun «yo'q»ni tanlaydi. Agar u ushbu tugun uchun «yo'q»ni tanlagan bo'lsa, Endi o'z harakatlaridan birini tanlashi mumkin: saqlash uchun foyda 0, ulashish uchun foyda 1 va berish uchun foyda 0, shuning uchun u ulashishni tanlaydi. Agar Barb chapdagi tugun uchun «ha»ni tanlaganida, Endi saqlashni tanlaydi va shunday qilib Endi ikkala obyektini ham oladi.

14.3.1. Raqiblar o'yinlari

Ikki agent raqobatlashayotgan, ya'ni bir agent uchun ijobiy mukofot ikkinchisi uchun salbiy mukofot oladigan holat, ikki agentli **nol-summa** o'yin deb ataladi. Bunday o'yinni yaxshi o'ynaydigan agent **raqibona mulohaza yuritishni** talab qiladi. Bunday o'yinning qiymatini bitta raqam bilan tavsiflash mumkin: bir agent uni maksimal qilishga, ikkinchisi esa minimal qilishga harakat qiladi, bu esa **minimaks** qidiruv maydoniga olib keladi. Har bir tugun yoki MAX tuguni bo'ladi, agar u maksimal qilishga harakat qilayotgan agent tomonidan boshqarilsa yoki MIN tuguni bo'ladi, agar u minimal qilishga harakat qilayotgan agent tomonidan boshqarilsa.

14.5-rasmda nol-summa mukammal ma'lumotli o'yinlar uchun minimaksalgoritmi amalga oshiriladi, bunda MAX agenti maksimal qiymatga ega bo'lgan harakatni tanlaydi va MIN agenti qiymatning *inkor qilishning* maksimal darajasiga ega bo'lgan harakatni tanlaydi.

14.5-rasmdagi o'yin-daraxti qidiruvi butun o'yin daraxtini kesib o'tadi. Daraxtning bir qismi hech qachon optimal o'yinning bir qismi bo'lmashligini ko'rsatib, minimaks o'yinlari uchun qidiruv daraxtining bir qismini kesish mumkin.

14.7-misol. 14.6-rasmdagi o'yin daraxtida qidiruvni ko'rib chiqing. Ushbu rasmda kvadrat MAX tugunlari maksimallashtiruvchi vosita tomonidan, yumaloq MIN tugunlari esa minimallashtiruvchi vosita tomonidan boshqariladi.

```
procedure GameTreeSearch(n)
```

```
  Inputs
```

```
  n a node in a game tree
```

```
  Output
```

```
  A pair of a value for each agent for node n, path that gives this value
  if n is a leaf node then
```

```
    return {i:avaluated(i,n)},None
```

```
  else if n is controlled by agent i then
```

```
    max_score_for_i:= - $\infty$ 
```

```
    for each child c of n do
```

```
      if score[i] > max score for i then
```

```
        max_score_for_i:= score[i]
```

```
      best: c: path
```

```
    return score, best
```

14.5-rasm. Mukammal ma'lumotli o'yin daraxtini baholash (tabiatsiz)

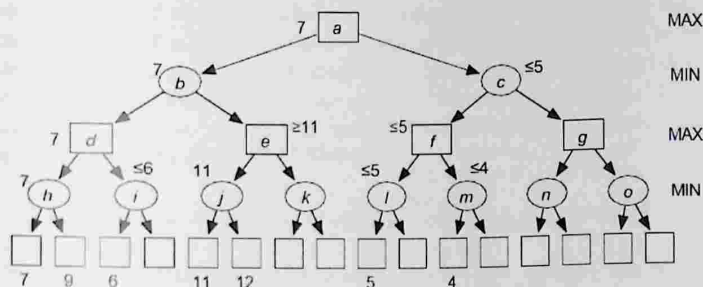
Barg tugunlarning qiymatlari berilgan yoki o'yinning ta'rifi berilgan holda hisoblab chiqiladi deb taxmin qilamiz. Pastdagi raqamlar bu qiymatlarning ba'zilarini ko'rsatadi. Qolgan qiymatlar, bu yerda ko'rsatilganidek, ahamiyatsizdir. Daraxtni chapdan birinchi bo'lib, chuqurlik bo'yicha kesib o'tayotganingizni taxmin qiling. *h* tugunining qiymati 7, chunki u 7 va 9 ning minimalidir. *i* tugunining eng chapdagi avlodi qiymati 6 bo'lgan holda, siz *i* tugunining qiymati 6 dan kam yoki unga teng ekanligini bilasiz. Shuning uchun, *d* tugunida, maksimal qiluvchi agent chapga boradi. Siz *i* tugunining boshqa avlodini baholashingiz shart emas. Xuddi shunday, *j*ning qiymati 11, shuning uchun *ening* qiymati kamida 11 va shuning uchun *b* tugunidagi minimal qiluvchi agent chapga borishni tanlaydi. *l* ning qiymati 5 dan kam yoki unga teng va *m* ning qiymati 4 dan kam yoki unga teng, shuning uchun *f*ning qiymati 5 dan kam yoki unga teng, shuning uchun *c*ning qiymati 5 dan kam yoki unga teng bo'ladi. Shunday qilib, *a* da, maksimal qiluvchi agent chapga borishni tanlaydi. Ushbu argument raqamlanmagan barglarning qiymatlariga bog'liq emasligini ko'rib chiqing. Bundan tashqari, u tekshirilmagan kichik daraxtlarning hajmiga bog'liq emas edi.

Yuqoridagi misol nimani qisqartirish mumkinligini tahlil qildi. Minimaks **alpha-beta (α - β) kesish** – chuqurlik bo'yicha qidiruv algoritmi bo'lib, α va β parametrlar nuqtayi nazaridan kesish ma'lumotlarini pastga yuborish orqali kesishadi. Ushbu chuqurlik bo'yicha qidiruvda tugun (ba'zi) avlodlaridan olingan ballga ega.

α parametri MIN tugunlarini kesish uchun ishlatiladi. Dastlab, bu joriy tugunning barcha MAX ajdodlari uchun eng yuqori joriy qiymatdir. Joriy qiymati uning α qiymatiga teng yoki undan kam bo'lgan har qanday MIN tuguni bundan keyin o'rganilishi shart emas. Ushbu kesish oldingi misolda l , m va c tugunlarining boshqa avlodlarini kesish uchun ishlatilgan.

β parametri MAX tugunlarini kesish uchun o'xshash tarzda ishlatiladi.

α - β kesish minimaks algoritmi 14.7-rasmda berilgan. U dastlab *Minimax Alpha Beta*($R, -\infty, \infty$) bilan chaqiriladi, bu yerda R ildiz tugunidir.



14.6-rasm. Qaysi tugunlarni kesish mumkinligini ko'rsatadigan nol-summalı o'yin daraxti

U n tugun uchun bir juft qiymat va bu qiymatga olib keladigan tanlov yo'lini qaytaradi. (E'tibor bering, bu yo'l n ni o'z ichiga olmaydi.) 13-qator β kesishni amalga oshiradi; bu bosqichda algoritim joriy yo'l hech qachon tanlanmasligini biladi va shuning uchun joriy ballni qaytaradi. Xuddi shunday, 22-qator α kesishni amalga oshiradi. 17-qator va 26-qator c yo'li bilan tutashadi, chunki u agent uchun eng yaxshi yo'lni topgan. Ushbu algoritmda «None» yo'li ba'zida barg bo'lmagan tugunlar uchun qaytariladi; bu faqat algoritim ushbu yo'ldan foydalanmasligini aniqlaganda sodir bo'ladi.

14.8-misol. 14.6-rasmdagi daraxtda *Minimax Alpha Beta*-ni ishga tushirishni ko'rib chiqing. Rekursiv chaqiruvlarni (va qaytarilgan qiymatlarni, ammo yo'llarni emas) ko'rib chiqing. Dastlab, u quyidagi chaqiruvni amalga oshiradi:

Minimax Alpha Beta ($a, -\infty, \infty$)

```

1: procedure Minimax_alpha_beta( $n, \alpha, \beta$ )
2:   Inputs
3:      $n$  a node in a game tree
4:      $\alpha, \beta$  real numbers
5:   Output
6:     A pair of a value for node  $n$ , path that gives this value
7:    $best := None$ 
8:   if  $n$  is a leaf node then
9:     return evaluate( $n$ ), None
10:  else if  $n$  is a MAX node then
11:    for each child  $c$  of  $n$  do
12:       $score, path := MinimaxAlphaBeta(c, \alpha, \beta)$ 
13:      if  $score \geq \beta$  then
14:        return  $score, None$ 
15:      else if  $score > \alpha$  then
16:         $\alpha := score$ 
17:         $best := c : path$ 
18:    return  $\alpha, best$ 
19:  else
20:    for each child  $c$  of  $n$  do
21:       $score, path := MinimaxAlphaBeta(c, \alpha, \beta)$ 
22:      if  $score \leq \alpha$  then
23:        return  $score, None$ 
24:      else if  $score < \beta$  then
25:         $\beta := score$ 
26:         $best := c : path$ 
27:    return  $\beta, best$ 

```

14.7-rasm. α - β kesish bilan minimaks

bu esa, o'z navbatida quyidagi qidiruvlarni amalga oshiradi:

Minimax Alpha Beta ($b, -\infty, \infty$)

Minimax Alpha Beta ($d, -\infty, \infty$) *Minimax Alpha Beta* ($h, -\infty, \infty$).

Ushbu oxirgi qidiruv ikkala avlodining minimumini topadi va 7 ni qaytaradi.

Keyin protsedura quyidagi qidiruvni amalga oshiradi:

Minimax Alpha Beta ($i, 7, \infty$) bu esa, *ining* birinchi vakilining qiymatini oladi, bu qiymat 6. Chunki $\alpha \geq \beta$, u 6 ni qaytaradi. d ga chaqiruv keyin 7 ni qaytaradi va quyidagi chaqiruv amalga oshiriladi:

Minimax Alpha Beta ($e, -\infty, 7$). e tugunining birinchi farzandi 11 ni qaytaradi va $\alpha \geq \beta$ bo'lgani uchun u 11 ni qaytaradi. Keyin b 7 ni qaytaradi va a ga chaqiruv quyidagi chaqiruvni amalga oshiradi:

Minimax Alpha Beta ($c, 7, \infty$)

bu esa, o'z navbatida, quyidagi chaqiruvni amalga oshiradi:

Minimax Alpha Beta ($f, 7, \infty$) bu oxir-oqibat 5 ni qaytaradi, shuning uchun c ga chaqiruv 5 ni qaytaradi va butun protsedura 7 ni qaytaradi.

Qiymatlarni kuzatib borish orqali maksimallashtiruvchi agent a nuqtada chapga yurishni biladi, keyin minimallashtiruvchi agent b nuqtada chapga yuradi va hokazo.

Ushbu algoritm tomonidan taqdim etilgan kesish miqdori har bir tugunning avlodlarining tartibiga bog'liq. U MAX tugunining eng yuqori qiymatli avlodi birinchi bo'lib tanlansa va MIN tugunining eng past qiymatli avlodi birinchi bo'lib qaytarilganda, eng yaxshi ishlaydi. Haqiqiy o'yinlarni amalga oshirishda, bu tartibni ta'minlashga harakat qilish uchun ko'p kuch sarflanadi.

Ko'pgina haqiqiy o'yinlar minimaks qidiruvni, hatto α - β kesish bilan ham amalga oshirish uchun juda katta. Ushbu o'yinlar uchun barg tugunlarida to'xtash o'rniga, har qanday tugunda to'xtash mumkin. Algoritm to'xtagan tugunda qaytarilgan qiymat ushbu tugun uchun qiymatning taxmini bo'ladi. Qiymatni baholash uchun ishlatiladigan funksiya **baholash funksiyasi** deb ataladi. Yaxshi baholash funksiyalarini topish uchun ko'p mehnat qilinadi. Baholash funksiyasini hisoblash uchun talab qilinadigan hisoblash miqdori va har qanday vaqtda qidirish mumkin bo'lgan qidiruv maydonining hajmi o'rtasida muvozanat mavjud. Murakkab baholash funksiyasi va katta qidiruv maydoni o'rtasidagi eng yaxshi muvozanat empirik savol hisoblanadi.

14.4. Mukammal bo'lmagan ma'lumotlar bilan mulohaza qilish

Mukammal bo'lmagan ma'lumotli o'yinda yoki **qisman kuzatiladigan o'yinda** agent dunyo holatini to'liq bilmaydi yoki agentlar bir vaqtning o'zida harakat qilishadi.

Ko'p agentli holat uchun qisman kuzatilish to'liq kuzatiladigan ko'p agentli holatdan yoki qisman kuzatiladigan yagona agentli holatdan ko'ra murakkabroqdir. Quyidagi oddiy misol hatto bir nechta tanlovlari bo'lgan ikki agent holatida ham muhim masalalarni ko'rsatadi.

14.9-misol. 14.8-rasmda tasvirlanganidek futboldagi penalti tepishni ko'rib chiqing. Agar hujumchi o'zining o'ng tomoniga tepsa va darvozabon o'zining o'ng tomoniga sakrasa, gol ehtimoli 0,9 va boshqa harakatlar kombinatsiyalari uchun ham shunday (rasmda berilganidek).

Hujumchi nima qilishi kerak, agar u gol ehtimolini maksimal qilishni va darvozabon gol ehtimolini minimal qilishni xohlasa? Hujumchi o'ng tomoniga tepgani yaxshiroq deb o'ylashi mumkin, chunki o'zining o'ng tomoniga tepish uchun sonlar jufti chap tomoniga qaraganda yuqoriroq. Keyin darvozabon agar hujumchi o'ngga tepsa, u chapga sakrashi kerakligi haqida o'ylashi mumkin. Biroq, agar hujumchi darvozabon chapga sakraydi deb o'ylasa, u chapga tepishi kerak. Ammo keyin darvozabon o'ngga sakrashi kerak. Keyin hujumchi o'ngga tepishi kerak...

Har bir agent boshqa agent nima qilishi to'g'risida cheksiz regressiyaga duch kelishi mumkin. Ularning mulohazalarining har bir bosqichida agentlar o'z qarorlarini bekor qiladi. Buni biron-bir chuqurlikda kesishni tasavvur qilish mumkin; ammo, keyin harakatlar faqat ixtiyoriy chuqurlik funksiyasi bo'ladi. Yanada yomoni, agar darvozabon hujumchi uchun mulohaza qilish chuqurligi cheklovini bilsa, u ushbu bilimdan foydalanib, hujumchi nima qilishi mumkinligini aniqlashi va o'z harakatini tegishli ravishda tanlashi mumkin.

Buning o'rniga agentlar tasodifiy harakatlarni tanlashlari mumkin. Tasavvur qiling, hujumchi va darvozabon har birisi yashirincha tanga tashlab, nima qilishni hal qilmoqda. Tangalar adolatli bo'lishi kerakmi yoki yo'qmi, deb o'ylab ko'ring. Faraz qiling, hujumchi o'zining o'ng tomoniga tepishni pk ehtimoli bilan va darvozabon o'zining o'ng tomoniga sakrashni pg ehtimoli bilan tanladi. Gol ehtimoli quyidagicha bo'ladi:



14.8-rasm. Gol urish ehtimoli.

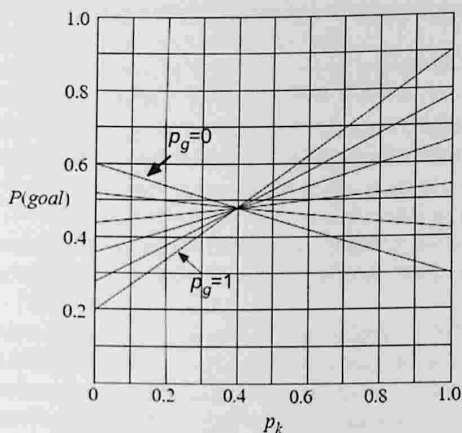
Futbolda jarima zarbasi. Hujumchi chap yoki o'ng tomonga tepishi mumkin. Darvozabon chapga yoki o'ngga sakrashi mumkin $P(goal) = 0.9pkpg + 0.3pk(1-pg) + 0.2(1-pk)pg + 0.6(1-pk)(1-pg)$ bu yerda (0.9, 0.3 va hokazo) raqamlar 14.8-rasmdan keladi.

14.9-rasmda pk ning funksiya sifatidagi gol ehtimoli ko'rsatilgan. Turli chiziqlar pg ning turli qiymatlariga mos keladi.

$Pk = 0.4$ qiymatida o'ziga xoslik bor. Ushbu qiymatda gol ehtimoli 0.48, pg qiymatidan qat'i nazar. Ya'ni darvozabon nima qilmasin, hujumchi 0.48 ehtimol bilan gol urishni kutadi. Agar hujumchi $pk = 0.4$ dan chetga chiqsa, darvozabon nima qilishiga bog'liq holda yaxshiroq yoki yomonroq natijaga erishishi mumkin.

Pg uchun chizma ham shunga o'xshash bo'lib, barcha chiziqlar $pg = 0.3$ da uchrashadi. Yana, $pg = 0.3$ da gol ehtimoli 0.48.

$P_k = 0.4$ va $p_g = 0.3$ strategiyasi shunchalik o'ziga xosdirki, agentlarning hech biri bu strategiyadan birtomonlama chetga chiqish orqali yaxshiroq ish qila olmaydi. Biroq, bu ular yaxshiroq ish qila olmaydi degani emas; agar agentlardan biri ushbu muvozanatdan chetga chiqsa, boshqa agent ham muvozanatdan chetga chiqish orqali yaxshiroq ish qilishi mumkin. Biroq, ushbu muvozanat agent uchun xavfsizdir, chunki boshqa agent agentning strategiyasini bilgan taqdirda ham, boshqa agent agent uchun yomonroq natijaga erisha olmaydi. Ushbu strategiyadan foydalanish agentning boshqa agentni taxmin qilish haqida tashvishlanmasligini anglatadi. Ushbu o'yinda har bir agent o'zi kafolatlaydigan eng yaxshi yutuqni oladi. Shunday qilib, endi strategiya ta'rifini tasodifiy strategiyalarni o'z ichiga olgan holda kengaytiramiz.



14.9-rasm. Maqsadga erishish ehtimoli harakat ehtimoli funksiyasi sifatida

O'yinning oddiy shaklini ko'rib chiqaylik, bu yerda har bir agent harakatni bir vaqtning o'zida tanlaydi. Har bir agent boshqa agentlarning tanlovlarini bilmasdan harakatni tanlaydi.

Agent uchun **strategiya** bu agent uchun harakatlar bo'yicha ehtimollik taqsimoti hisoblanadi. **Sof strategiyada** ehtimolliklardan biri 1 va qolganlari 0 bo'ladi. Shunday qilib, sof strategiyaga amal qilayotgan agent deterministik harakat qiladi. Sof strategiyaning muqobili **stoxastik strategiya** bo'lib, unda hech qanday ehtimollik 1 emas va shuning uchun bir nechta harakatlar nol bo'lmagan ehtimollikka ega. Strategiyadagi nol bo'lmagan ehtimollikka ega harakatlar to'plami strategiyaning **qo'llab-quvvatlash to'plami** hisoblanadi.

Strategiya profili bu har bir agentga strategiyani tayinlashdir. Agar σ strategiya profili bo'lsa, σ_i agentning σ dagi strategiyasi va $\sigma-i$ boshqa agentlarning strategiyalari bo'ladi. Shunda σ bu $\sigma_i\sigma-i$ bo'ladi. Agar strategiya profili sof strategiyalardan iborat bo'lsa, odatda bu **harakat profili** deb ataladi, chunki har bir agent ma'lum bir harakatni o'ynamoqda.

σ strategiya profilining har bir agent uchun kutilgan foydasi bor. $Utility(\sigma, i)$ agent i uchun σ strategiya profilining kutilgan foydasini bildiradi. Stoxastik strategiya profilining foydasi profilni tashkil etuvchi asosiy harakatlarning ehtimolliklari berilgan holda, ularning foydalarini o'rtacha qilib hisoblash orqali topilishi mumkin.

Agent i uchun boshqa agentlarning $\sigma-i$ strategiyalariga **eng yaxshi javob** bu agent uchun maksimal foydaga ega strategiya hisoblanadi. Ya'ni agar agent i uchun boshqa strategiyalar uchun quyidagi tenglik bajarilsa, σ_i , $\sigma-i$ ga eng yaxshi javobdir:

$$utility(\sigma_i, \sigma_{-i}) \geq utility(\sigma_i', \sigma_{-i}).$$

Agar har bir agent i uchun σ_i strategiyasi, σ_{-i} ga eng yaxshi javob bo'lsa, σ strategiya profili **Nash muvozanati** deb ataladi. Ya'ni Nash muvozanati shunday strategiya profiliki, hech bir agent ushbu profilni birtomonlama o'zgartirish orqali yaxshiroq natijaga erisholmaydi.

O'yin nazariyasining katta yutuqlaridan biri Nash [1950] tomonidan isbotlangan bo'lib, har bir cheklangan o'yin kamida bitta Nash muvozanatiga ega ekanini ko'rsatadi.

14.10-misol. 14.9-misolda $pk = 0.4$ va $pg = 0.3$ bo'lgan yagona Nash muvozanati mavjud. Bu shunday xususiyatga egaki, agar hujumchi $pk = 0.4$ o'ynayotgan bo'lsa, darvozabon nima qilayotgani muhim emas; darvozabon bir xil foydaga ega bo'ladi va shuning uchun $pg = 0.3$ eng yaxshi javobdir (boshqa har qanday strategiya kabi). Xuddi shunday, agar darvozabon $pg = 0.3$ o'ynayotgan bo'lsa, hujumchining nima qilayotgani muhim emas va shuning uchun har qanday strategiya, shu jumladan $pk = 0.4$, eng yaxshi javobdir.

Agent ikkita harakat o'rtasida tasodifiy tanlashni faqat harakatlar bir xil kutilgan foydaga ega bo'lsa ko'rib chiqadi. Ikki harakatning barcha ehtimollik aralashmalari bir xil foydaga ega. Aralashma ehtimolligi uchun ma'lum bir qiymatni tanlashning sababi boshqa agentning og'ishdan foydalanishiga to'sqinlik qilishdir.

O'yinlarda bir nechta Nash muvozanati bo'lishi mumkin. Quyidagi ikki agentli, ikki harakatli o'yinni ko'rib chiqing.

14.11-misol. Faraz qilaylik, ikkita agent bir resurs uchun kurashmoqchi. Har bir agent qirg'iy yoki kabutar bo'lib harakat qilishni tanlaydi. Faraz qilaylik, resurs R birlikka teng, bu yerda $R > 0$. Agar ikkala agent ham kabutar bo'lib harakat qilsa, ular resursni bo'lishadi. Agar bir agent qirg'iy bo'lib harakat qilsa va boshqa agent kabutar bo'lib harakat qilsa, qirg'iy agent resursni oladi va kabutar agent

hech narsa olmaydi. Agar ikkisi ham qirg'iy bo'lib harakat qilsa, resurs yo'q bo'ladi va ikkalasi ham $-D$ mukofotini oladi, bu yerda $D > 0$. Bu quyidagi foyda matritsasi bilan tasvirlangan:

2-agent

	kaptar	qirg'iy
kaptar	$R/2, R/2$	$0, R$
qirg'iy	$R, 0$	$-D, -D$

1-agent

Bu matritsada, 1-agent qatorni tanlaydi, 2-agent ustunni tanlaydi va katakdagi foyda juftligi 1-agent va 2-agent uchun mukofotni bildiradi. Har bir agent o'z mukofotini maksimallashtirishga harakat qiladi.

Ushbu o'yinda uchta Nash muvozanati mavjud:

Bir muvozanatda, 1-agent qirg'iy bo'lib, 2-agent kabutar bo'lib harakat qiladi. 1-agent chetga chiqishni xohlamaydi, chunki unda ular resursni baham ko'rishlari kerak. 2-agent chetlanishni xohlamaydi, chunki resurs yo'q bo'ladi.

Ikkinchi muvozanatda, 1-agent kabutar bo'lib harakat qiladi va 2-agent qirg'iy bo'lib harakat qiladi.

Uchinchi muvozanatda, ikkala agent ham tasodifiy harakat qiladi. Ushbu muvozanatda resurs yo'q bo'lish ehtimoli bor. Qirg'iy bo'lib harakat qilish ehtimoli resursning R qiymati oshishi bilan ortadi va yo'q bo'lishning D qiymati oshgani sayin kamayadi (14.3-mashqqa qarang).

Ushbu misolda siz har bir agent o'zi uchun foydali muvozanatni ta'minlash uchun nima qilishini ko'rsatish uchun biroz harakat qilayotganini tasavvur qilishingiz mumkin.

Quyidagi misoldan ko'rinib turibdiki, bir nechta Nash muvozanatiga ega bo'lish raqib bo'lish bilan bog'liq emas.

14.12-misol. Faraz qilaylik, birga bo'lishni xohlaydigan ikki kishi bor. 1-agent ikkalasi ham futbol o'yiniga borishini afzal ko'radi va 2-agent ikkalasi ham do'konga borishini afzal ko'radi. Ular birga bo'lmaganlarida ikkalasi ham baxtsiz bo'ladi. Faraz qilaylik, ikkalasi ham bir vaqtda nima qilishni tanlashlari kerak. Bu quyidagi foyda matritsasi bilan tasvirlangan:

2-agent

	futbol	shopping
futbol	$2, 1$	$0, 0$
shopping	$0, 0$	$1, 2$

1-agent

Bu matritsada, 1-agent qatorni tanlaydi va 2-agent ustunni tanlaydi. Ushbu o'yinda uchta Nash muvozanati mavjud. Bitta muvozanat ikkisi ham xarid qilishga boradi, ikkinchi muvozanat ikkisi ham futbol o'yiniga boradi va uchinchi muvozanat tasodifiy strategiyadir.

Bu **muvofiglashtirish** muammosidir. Muvozanatlar to'plamini bilish haqiqatan ham har qanday agentga nima qilish kerakligini anglatmaydi, chunki bir agent nima qilishi kerakligi boshqa agent nima qilishiga bog'liq. Ushbu misolda ular qaysi muvozanatni tanlashni aniqlash uchun suhbatlarni tasavvur qilishingiz mumkin.

Garchi o'yinda yagona Nash muvozanati mavjud bo'lsa ham, bu Nash muvozanati har bir agentga maksimal foydani kafolatlamaydi. Quyidagi misol mashhur **mahbus dilemmasining** bir variantidir.

14.13-misol. Faraz qilaylik, siz notanish bir kishi bilan bir o'yin namoyishdasisiz va uni boshqa hech qachon ko'rmaysiz. Siz har bir tanlovingizda quyidagilarni tanlashingiz mumkin:

O'zingiz uchun \$100 olish

Boshqa kishiga \$1000 berish

Bu quyidagi foyda matritsasi bilan tasvirlangan:

2-o'yinchi

	olish	berish
olish	100,100	1100,0
berish	0,1100	1000,1000

1-o'yinchi

Boshqa agent nima qilmasin, har bir agent bergandan ko'ra olsa yaxshi bo'ladi. Biroq, ikkala agent ham olishdan ko'ra, ikkalasi ham bersa, yaxshiroq bo'ladi.

Shunday qilib, yagona Nash muvozanati mavjud bo'lib, bu yerda ikkala agent ham oladi. Bu strategiya profilida har bir o'yinchi \$100 oladi. Har ikkala o'yinchi beradigan strategiya profilida esa har biri \$1000 oladi. Biroq, ushbu strategiya profilida har bir agent og'ish uchun mukofotlanadi.

Mahbus dilemmasi bo'yicha ko'plab tadqiqotlar olib borilmoqda, chunki har bir agent o'zi uchun eng yaxshisini qilishga harakat qilsa, ochko'zlikunchalik oqilona ko'rinmaydi, natijada hammaga yomonroq bo'ladi. O'yin bir necha marta o'ynaladigan holatda berish afzal bo'ladi. Bu **ketma-ket mahbus dilemmasi** deb ataladi. Ketma-ket mahbus dilemmasini hal qilish strategiyalaridan biri bu **bir-biriga qarshi**: har bir o'yinchi avval beradi, so'ngra har bir bosqichda boshqa agentning oldingi harakatini bajaradi. Agar ikkala o'yinchi ham biladigan oxirgi harakat bo'lmasa, bu strategiya Nash muvozanatidir (14.8-mashqqa qarang).

Bir nechta Nash muvozanatiga ega bo'lish faqat qisman kuzatuvchanlikdan kelib chiqmaydi. Mukammal ma'lumotli o'yinda ham ko'p muvozanatlar mavjud bo'lishi mumkin, hatto cheksiz ko'p Nash muvozanatlari ham mavjud bo'lishi mumkin, quyidagi misol buni ko'rsatadi.

14.14-misol. Bo'lish o'yini 14.2-misolda ko'rib chiqamiz. Bu o'yinda Nash muvozanatlari cheksiz ko'p. Shunday muvozanat to'plami mavjudki, unda Endi ulashadi, Barb esa markazni tanlash uchun «ha» deydi va boshqa tanlovlar o'rtasida tasodifiy bo'lishi mumkin, chunki chapdagi tanlovda «ha» deyish ehtimoli kichik yoki 0,5 ga teng. Ushbu Nash muvozanatlarida ikkalasi ham 1 ni oladi. Endi oladigan va Barb o'z tanlovlari o'rtasida tasodifiy tanlaydigan, chap tarmoqli tanlovda *ha* aytish ehtimoli 0,5 dan katta bo'lgan boshqa Nash muvozanatlari to'plami mavjud. Ushbu muvozanatlarda Barb 0 ni oladi va Endi Barbnings ehtimoliga qarab [1, 2] oralig'ida qiymat oladi. Uchinchi to'plam Nash muvozanatlari mavjud, unda Barb chap tarafdagi tugunda 0,5 ehtimol bilan «ha» ni tanlaydi, markaziy tugunda «ha» ni tanlaydi va Endi har qanday ehtimollik bilan saqlash va bo'lishish o'rtasida tasodifiy tartibni o'rnatadi.

Aytaylik, ulashish o'yini biroz o'zgartirildi, shunda Endi Barbga *ha* deyishi uchun kichik pora taklif qildi. Bu 2,0 to'lovini 1,9, 0,1 ga o'zgartirish bilan amalga oshirilishi mumkin. Endi o'ylashi mumkin: «0,1 yoki 0 olish o'rtasida tanlov berilganda, Barb 0,1 olishni tanlaydi, shuning uchun men *saqlashim* kerak». Lekin Barb shunday deb o'ylashi mumkin: «Men 0,1 ga *yo'q* deyishim kerak, shunda Endi ulashadi va men 1 ni olaman».

Bu misolda (hatto o'ng tarafdagi tugunni inobatga olmaganida ham), bir nechta sof Nash muvozanati mavjud. Biri shundaki, Endi *saqlaydi* va Barb chap tomondagi tugunda «ha» deydi. Bu muvozanatda Endi 1,9, Barb esa 0,1 ni oladi. Boshqa Nash muvozanati shundaki, Barb chap tomondagi tanlov tugunida «yo'q» va markaziy tugunda «ha» deydi va Endi ulashishni tanlaydi. Bu muvozanatda, ular ikkalasi ham 1 ni oladi. Bu Barbga ma'qul bo'lib tuyulishi mumkin. Biroq, Endi Barb shunchaki **tahdid** qilmoqda deb o'ylashi mumkin. Agar u haqiqatan ham *saqlashni* tanlagan bo'lsa, Barb o'z foydasini oshirish uchun harakat qilganda, aslida «yo'q» demasligi mumkin.

Teskari induksiya algoritmi oldingi misoldagi o'zgartirilgan ulashish o'yinida faqat bitta muvozanatni topadi. Bu algoritm har bir tugunda agentlar uchun eng katta foyda keltiradigan harakatni tanlashlari faraz qilingan **kichik o'yin mukammal muvozanatni** hisoblaydi. U, agentlar o'z manfaatlariga mos kelmaydigan tahdidlarni amalga oshirmaydilar deb faraz qiladi. Oldingi misoldagi o'zgartirilgan ulashish o'yinida, Barb kichik poraga «ha» deydi deb faraz qilinadi. Biroq, haqiqiy raqiblar bilan muomala qilganda, ular siz mantiqsiz deb o'ylashingiz mumkin bo'lgan tahdidlarni amalga oshirishlari mumkinligini bilishingiz kerak. Darhaqiqat, agent uchun oqilona bo'lmaslik yoki shunday ko'rinmaslik yaxshiroq bo'lishi mumkin!

14.4.1. Nash muvozanatlarini hisoblash

Oddiy shakldagi o'yin uchun Nash muvozanatini hisoblash uchun uchta bosqich mavjud:

Ustun bo'lgan strategiyalarni istisno qiling.

Qo'llab-quvvatlash to'plamini aniqlang, bu nolga teng bo'lmagan ehtimolliklarga ega bo'lgan harakatlar to'plamidir.

Qo'llab-quvvatlash to'plamidagi harakatlar uchun ehtimolliklarni aniqlang.

Ma'lum bo'lishicha, ulardan ikkinchisi eng qiyini bo'lib chiqadi.

Ustun bo'lgan strategiyalarni istisno qilish

A agentiga s_1 strategiyasi A uchun s_2 strategiyasidan **ustun turadi**, agar boshqa agentlarning har bir harakati uchun A agenti uchun s_1 foydaliligi A agenti uchun s_2 foydaliligidan past bo'lsa. Quyida bu rasmiylashtirilgan. Har qanday boshqa strategiya tomonidan ustun qilingan sof strategiya ko'rib chiqishdan istisno qilinishi mumkin. Ustun turuvchi strategiya tasodifiy strategiya bo'lishi mumkin. Ustun qilingan strategiyalarni istisno qilish tashlash takroriy ravishda amalga oshirilishi mumkin.

14.15-misol. Quyidagi yutuq matritsasini ko'rib chiqing, bunda birinchi agent qatorni va ikkinchi agent ustunni tanlaydi. Har bir qator va ustun kombinatsiyasi uchun har bir agentning foydalari juftlik sifatida berilgan: birinchi agent uchun foyda va ikkinchi agent uchun foyda. Birinchi agentning harakatlari $\{a_1, b_1, c_1\}$. Ikkinchi agentning harakatlari $\{d_2, e_2, f_2\}$.

2-agent

	d_2	e_2	f_2
a_1	3,5	5,1	1,2
b_1	1,1	2,9	6,4
c_1	2,6	4,7	0,8

1-agent

(Yechimga qarashdan oldin, har bir agent nima qilishini aniqlashga harakat qiling.)

c_1 harakati a_1 harakati tomonidan ustun qilinishi sababli olib tashlanishi mumkin: 1-agent, a_1 harakatini amalga oshirish imkoniyati bo'lsa, hech qachon c_1 harakatini bajarmaydi. E'tibor bering, 1-agent uchun har qanday boshqa agent nima qilsa ham, a_1 harakatini bajarish c_1 harakatini bajarishdan ko'ra ko'proq foyda beradi.

c_1 amali istisno qiliganidan so'ng, f_2 harakati 2-agent uchun $0.5d_2 + 0.5e_2$ tasodifiy strategiya tomonidan ustun qilinishi sababli olib tashlanishi mumkin.

c_1 va f_2 istisno qiliganidan so'ng, b_1, a_1 tomonidan ustun qilinadi, shuning uchun 1-agent, a_1 harakatini amalga oshiradi. 1-agent, a_1 harakatini amalga oshiradi, deb hisoblagan holda, 2-agent, d_2 harakatini amalga oshiradi. Shunday qilib, yagona Nash muvozanati 1-agent a_1 harakatini amalga oshirishi va 2-agent d_2 harakatini amalga oshirishi bilan bog'liq bo'lib, 1-agent uchun 3 va 2-agent uchun 5 foyda beradi.

Agar barcha boshqa agentlarning σ_i harakat profillari uchun:

$$utility(s_1, \sigma_{-1}, i) > utility(s_2, \sigma_{-1}, i)$$

bo'lsa, s_1 strategiyasi Agent i uchun s_2 strategiyasini **qat'iy ustun qiladi**. Agar s_2 har qanday sof strategiya bo'lib, s_1 strategiyasi tomonidan qat'iy ustun qilinsa, s_2 hech qachon hech qanday Nash muvozanatining qo'llab-quvvatlash to'plamida bo'lishi mumkin emas. Bu s_1 stoxastik strategiya bo'lsa ham amal qiladi. Qat'iy ustun qilingan strategiyalarni takroran istisno qilish, qat'iy hukm qilingan strategiyalarni olib tashlash tartibidan qat'i nazar, bir xil natijani beradi.

Bundan tashqari, ustunlikning zaif tushunchalari ham mavjud bo'lib, oldingi formuladagi katta belgini katta yoki teng belgisiga almashtiradi. Agar ustunlikning zaif tushunchasi qo'llanilsa, har doim ustun qilinmagan strategiyalarni qo'llab-quvvatlaydigan Nash muvozanati mavjud. Biroq, ba'zi Nash muvozanatlari yo'qolishi mumkin. Qaysi muvozanatlar yo'qolishi ustun qilingan strategiyalarni olib tashlash tartibiga bog'liq bo'lishi mumkin.

Tasodifiy strategiyalarni hisoblash

Agent faqat boshqa agentlarning strategiyalari berilgan holda barcha harakatlari uchun bir xil foyda bo'lsa, harakatlarni tasodifiy ravishda tanlaydi. Bu g'oya Nash muvozanatini hisoblash uchun yechilishi mumkin bo'lgan cheklovlar to'plamiga olib keladi. Agar bu cheklovlar (0,1) oralig'idagi raqamlar bilan yechilishi mumkin bo'lsa va har bir agent uchun hisoblangan aralash strategiyalar boshqa strategiya tomonidan ustun qilinmasa, bu strategiya profili Nash muvozanati hisoblanadi.

Eslatib o'tamiz, qo'llab-quvvatlash to'plami Nash muvozanatida har birida nol bo'lmagan ehtimollik mavjud bo'lgan sof strategiyalar to'plamidir.

Ustun qilingan strategiyalar istisno qilinganidan so'ng, agent qo'llab-quvvatlash to'plamlari Nash muvozanatini tashkil etishini aniqlash uchun ularni qidirishi mumkin. E'tibor bering, agar agent uchun mavjud bo'lgan n harakatlar mavjud bo'lsa, $2n - 1$ bo'sh bo'lmagan to'plamlar mavjud va biz turli agentlar uchun qo'llab-quvvatlash to'plamlarini birlashtirib qidirishimiz kerak. Bu faqat ustun

qilinmagan harakatlar kam bo'lsa yoki Nash muvozanatlari kichik qo'llab-quvvatlash to'plamlariga ega bo'lsa amalga oshiriladi. Soddalashgan (qo'llab-quvvatlash to'plamidagi harakatlar soni bo'yicha) muvozanatlarni topish uchun agent kichikroq qo'llab-quvvatlash to'plamlaridan kattaroqlarigacha qidirishi mumkin.

Faraz qilaylik, i agenti har biri nolga teng bo'lmagan ehtimolga ega bo'lgan Nash muvozanatida $a^1, \dots, a^i$ ta'sirni randomizatsiyalagan bo'lsin. Agar agent i harakati a^i ni qiladigan ehtimollik p_i bo'lsa, boshqa agentlar uchun strategiyalar ularning ehtimollariga bog'liq bo'lgan $\sigma-i$ funksiyasi bo'lsin. Bu Nash muvozanati ekanligi quyidagi cheklavlarni beradi:

$$p^i > 0, \sum^i p^i = 1,$$

va barcha $j \neq i$ uchun:

$$utility(a^i, \sigma_{-i}) = utility(a^j, \sigma_{-i}).$$

Shuningdek, a^j harakatini bajarishning foydasi qo'llab-quvvatlash to'plamidan tashqaridagi harakatni bajarishning foydasidan kam emasligini talab qilishinigiz kerak. Shunday qilib, barcha $a^i \in \{a^1, \dots, a^i\}$ uchun:

$$utility(a^i, \sigma_{-i}) \geq utility(a^j, \sigma_{-i})$$

14.16-misol. 14.9-misolda, darvozabon pg ehtimoli bilan o'ng tomonga sakraydi va hujumchi pk ehtimoli bilan o'ng tomonga zarba beradi deb taxmin qiling.

Agar darvozabon o'ng tomonga sakrasa, gol ehtimolligi:

$$0.9pk + 0.2(1-pk).$$

Agar darvozabon chap tomonga sakrasa, gol ehtimolligi:

$$0.3pk + 0.6(1-pk).$$

Darvozabon faqat shular teng bo'lsa tasodifiy tanlaydi; ya'ni agar

$$0.9pk + 0.2(1-pk) = 0.3pk + 0.6(1-pk).$$

pk uchun yechim $pk = 0.4$ ni beradi.

Xuddi shunday, hujumchi tasodifiy bo'lishi uchun, hujumchi chap yoki o'ng zarbasidan qat'i nazar, gol ehtimoli bir xil bo'lishi kerak:

$$0.2pg + 0.6(1-pg) = 0.9pg + 0.3(1-pg).$$

pg uchun yechim $pg = 0.3$ ni beradi.

Shunday qilib, yagona Nash muvozanati $pk = 0.4$ va $pg = 0.3$ ga ega.

14.5. Guruh bo'lib qaror qabul qilish

Ko'pincha, odamlar guruhi nima qilish kerakligi haqida qaror qabul qilishlari kerak bo'ladi. Guruh nima xohlayotganini aniqlash uchun ovoz berish yaxshi yo'l

bo'lib tuyulishi mumkin va eng afzal tanlov aniq bo'lganda, bu shunday bo'ladi ham. Biroq, quyidagi misolda ko'rsatilganidek, aniq afzal qilingan tanlov bo'lmasa, ovoz berishda katta muammolar mavjud.

14.17-misol. Bir guruh odamlar uchun ularning xohish-istaklaridan kelib chiqqan holda dam olish joyini belgilashi kerak bo'lgan xarid agentini ko'rib chiqamiz.

Uchta odam, Elis, Bob va Kori va uchta manzil, X , Y va Z bor deb faraz qiling. Agentlar quyidagi afzalliklarga ega bo'lsin, bu yerda $\succ$ kuchli afzallikni bildiradi:

Alice: $X \succ Y \succ Z$

Bob: $Y \succ Z \succ X$

Cory: $Z \succ X \succ Y$

Bu afzalliklarni hisobga olgan holda, juft-juft ovoz berishda, $X \succ Y$, chunki uchta odamdan ikkitasi X ni Y dan afzal ko'radi. Xuddi shunday, ovoz berishda, $Y \succ Z$ va $Z \succ X$. Shunday qilib, ovoz berish natijasida olingan afzalliklar tranzitiv emas. Bu misol **Kondorset paradoksi** deb nomlanadi. Darhaqiqat, bu holatda guruh natijasi nima bo'lishi kerakligi aniq emas, chunki bu natijalar orasida simmetriya mavjud.

Ijtimoiy afzallik funksiyasi guruh uchun afzalliklar munosabatini beradi. Biz ijtimoiy afzallik funksiyasi guruhdagi shaxslarning Afzal ko'rishliklariga bog'liq bo'lishini xohlaymiz. Kondorset paradoksi faqat juft-juft ovoz berish bilan bog'liq muammo bo'lib tuyulishi mumkin; ammo, Arrow [1963] tomonidan keltirilgan quyidagi natija shuni ko'rsatadiki, bunday paradokslar har qanday ijtimoiy afzallik funksiyasi bilan sodir bo'ladi.

14.1-taklif (Arrovning imkonsizlik teoremasi). *Agar uchta yoki undan ko'p natijalar mavjud bo'lsa, quyidagi xususiyatlar bir vaqtning o'zida biron-bir ijtimoiy afzallik funksiyasi uchun amal qila olmaydi:*

Ijtimoiy afzallik funksiyasi to'liq va tranzitivdir.

• To'liq va o'tish xususiyatiga ega bo'lgan har bir shaxsiy afzallikka ruxsat beriladi.

Agar har bir kishi o_2 natijadan ko'ra o_1 natijasini afzal ko'rsa, guruh o_2 natijasidan o_1 ni afzal ko'radi.

o_1 va o_2 natijalari o'rtasidagi guruh afzalligi faqat o_1 va o_2 ning individual afzalliklariga bog'liq, boshqa natijalarning individual afzalliklariga emas.

Hech bir shaxs yakka tartibda natijani hal qilmaydi (nodiktatorlik).

Shaxsiy afzalliklarni olib, ijtimoiy afzalliklarni beradigan agentni yaratishda, ushbu intuitiv va xohlagan xususiyatlarning barchasiga ega bo'la olmasligingizni bilishingiz kerak. Istalmagan xususiyatlarga ega bo'lgan guruhga afzallik berish o'rniga, shaxslarga o'z afzalliklarini muvofiqlashtira olmasligini ko'rsatish yaxshiroq bo'lishi mumkin.

14.6. Mexanizmlarni loyihalash

Agentlarning harakatlarini tanlashga bag'ishlangan avvalgi muhokamalarda har bir agent oldindan belgilangan o'yinda qatnashishini faraz qildik. **Mexanizmlarni loyihalash** vazifasi turli agentlar uchun kerakli xususiyatlarga ega o'yinni ishlab chiqishdir.

Mexanizm har bir agent uchun mavjud bo'lgan harakatlar va har bir harakat profilining natijalarini belgilaydi. Biz agentlar natijalar ustidan foydalarga ega ekanligini faraz qilamiz.

Mexanizm uchun zarur bo'lgan ikkita umumiy xususiyat mavjud:

Mexanizm agentlar uchun oson bo'lishi kerak. Agentning foydasiga ko'ra, agent nima qilish kerakligini aniqlashi oson bo'lishi kerak. Boshqa agentlar nima qilishidan qat'i nazar, **ustuvor strategiya** agent uchun eng yaxshi strategiyadir. Agar agentda dominant strategiya bo'lsa, u eng yaxshi harakatini murakkab strategik mulohazasiz bajarishi mumkin. Mexanizm har bir agent uchun dominant strategiyaga ega bo'lsa va dominant strategiyada agentning eng yaxshi strategiyasi o'zining haqiqiy afzalliklarini bayon qilish bo'lsa, bu mexanizm **haqiqiy ustuvor strategiya** deb ataladi. Haqiqiy **ustuvor** strategiya mexanizmida agent o'z haqiqiy afzalliklarini bayon qiladi; agent mexanizmini o'z foydasi uchun manipulyatsiya qilishga urinish orqali yaxshiroq natijaga erisholmaydi.

Mexanizm eng yaxshi natijani agentlar ustidan jamlangan holda berishi kerak. Agar tanlangan natija agentlarning umumiy foydalarini maksimal darajada oshirsa, mexanizm **iqtisodiy jihatdan samarali** hisoblanadi.

14.18-misol. Siz uchrashuv jadvalini yaratmoqchi bo'lsangiz, foydalanuvchilar qachon mavjudligini kiritib, rejalashtiruvchi uchrashuv vaqtini tanlaydi. Bir mexanizm foydalanuvchilarning qachon mavjud yoki mavjud emasligini belgilashi va rejalashtiruvchi eng ko'p odam mavjud bo'lgan vaqtni tanlashi mumkin. Ikkinchi mexanizm foydalanuvchilarning turli vaqtlardagi foydalarini belgilashi va rejalashtiruvchi umumiy foydalarni maksimal darajada oshiradigan vaqtni tanlashi mumkin. Bu mexanizmlarning hech biri haqiqiy dominant strategiya emas.

Birinchi holda, foydalanuvchilar o'zlari uchun qulay bo'lgan vaqtni ko'rsatish uchun ma'lum bir vaqtda mavjud emasligini aytishlari mumkin. Ma'lum bir vaqtda mavjud bo'lish yaxshi aniqlangan bo'lishi aniq emas; ba'zi bir bosqichda, foydalanuvchilar o'sha vaqtda mavjud emasligini aytishdan ko'ra, ma'lum bir vaqtda nima qilishlarini qayta rejalashtirish osonroq yoki yo'qligini hal qilishlari kerak. Turli odamlar boshqa faoliyatlarni qachon ko'chirish mumkinligi bo'yicha turli xil chegaralarga ega bo'lishi mumkin.

Ikkinchi mexanizm uchun, uch kishi, Elis, Bob va Kori bor deb faraz qiling va ular dushanba, seshanba yoki chorshanba kunlari uchrashish kerakligini hal qilishlari kerak. Ularning uchrashuv kunlari uchun quyidagi foydaliklari bo'lsin:

	Dushanba	Seshanba	Chorshanba
Elis	0	8	10
Bob	3	4	0
Kori	11	7	6

Iqtisodiy jihatdan samarali natija seshanba kuni uchrashishdir. Biroq, agar Elis seshanba kunini 2 ga o'zgartirsa, mexanizm chorshanbani tanlaydi. Shunday qilib, Elis o'z qiymatlarini noto'g'ri bayon qilish uchun rag'batga ega. Elis uchun rostgo'y bo'lish foydali emas.

Dominant strategiyalarga ega bo'lgan mexanizm mavjud bo'lsa, haqiqiy dominant strategiyaga ega mexanizm ham mavjud. Bu revelation (oshkor qilish printsipli) deb nomlanadi. Haqiqiy ustuvor strategiyani mexanizmni amalga oshirish uchun, biz, asosan, agentdan uning haqiqiy afzalliklarini qabul qiladigan va manba mexanizmiga ushbu agent uchun maqbul ma'lumotlarni taqdim etadigan dastur yozishimiz mumkin. Ushbu dastur agent uchun optimal ravishda yolg'on gapirgan bo'lar edi.

Haqiqiy dominant strategiya bo'lgan oqilona mexanizmni yaratish amalda deyarli imkonsiz ekanligi ma'lum bo'ldi. Gibbard [1973] va Satterthwaite [1975], agar tanlash mumkin bo'lgan uchta yoki undan ko'p natijalar mavjud bo'lsa, dominant strategiyaga ega bo'lgan yagona mexanizmlar **diktatorga** ega ekanligini isbotladilar: imtiyozlari natijani belgilaydigan bitta agent bor. Bu **Gibbard-Satterthwaite teoremasi** deb nomlanadi.

Haqiqiy dominant strategiya mexanizmlariga erishishning bir yo'li pulni kiritishdir. Pulni foydaga qo'shish mumkin deb faraz qiling, shunda har qanday ikkita natija o_1 va o_2 uchun, har bir agent uchun o_1 va $o_2 + d$ natijalariga nisbatan befarq bo'lish uchun ba'zi (ehtimol manfiy) miqdor d mavjud. Agentlarga ular afzal ko'rmagan natijani qabul qilganliklari yoki istalgan natija uchun to'lashlari uchun pul to'lashga ruxsat berish orqali biz agent yolg'on gapirishdan foyda ko'rmasligiga ishonch hosil qilishimiz mumkin.

VCG (Vickrey – Clarke – Groves) mexanizmida, agentlar har bir natija uchun o'z qiymatlarini bayon qiladilar. E'lon qilingan qiymatlar yig'indisini maksimal darajada oshiradigan natija tanlanadi. Agentlar ularning ishtiroki natijaga qanday ta'sir qilganiga qarab to'laydilar. Agent i , agar i ishtirok etmagan bo'lsa, boshqa agentlar uchun qiymatlar yig'indisini to'laydi, agar i ishtirok etgan bo'lsa, boshqa agentlar uchun qiymatlar yig'indisini olib tashlaydi. Shunday qilib, agent uning ishtiroki boshqalarga tushadigan xarajat miqdorni to'lashi kerak.

14.19-misol. 14.18-misol qiymatlarini ko'rib chiqing. Berilgan qiymatlarni dollar ekvivalenti sifatida talqin qilish mumkin deb taxmin qiling; masalan, Elis dushanba yoki seshanba kuni uchrashish va 8,00 dollar to'lash o'rtasida befarq

(u uchrashuvni dushanbadan seshanbaga ko'chirish uchun 7,99 dollar to'lashga tayyor, lekin 8,01 dollar emas).

Ushbu e'lon qilingan qiymatlarga ko'ra, uchrashuv kuni uchun seshanba tanlanadi. Agar Elis ishtirok etmaganida, dushanba tanlangan bo'lardi va shuning uchun boshqa agentlar 3 sof zarar olgan bo'lishadi, shuning uchun Elis \$3.00 to'lashi kerak. Uning sof qiymati u holda 5 ga teng; seshanba uchun foyda 8 minus 3 to'lov. E'lonlar, to'lovlar va sof qiymatlar quyidagi jadvalda keltirilgan:

	Dushanba	Seshanba	Chorshanba	To'lov	Sof qiymati
Elis	0	8	10	3	5
Bob	3	4	0	1	3
Kori	11	7	6	0	7
Jami	14	19	16		

Agar Elis seshanba kungi bahosini 2 ga o'zgartirganida nima bo'lishini o'ylab ko'ring. Bu holda, chorshanba tanlangan kun bo'lardi, lekin Elis sof qiymati 2 ga teng bo'lgan 8,00 dollar to'lashi kerak edi va shuning uchun ham ahvol yomonroq bo'lar edi. Elis mexanizmga yolg'on gapirib foyda ololmaydi. To'lov haqida o'ylashning bir yo'li shundaki, Elis mexanizmini uning yoqtirgan tanloviga moslash uchun pora berishi kerak.

Agar agentlar boshqa agentlarning foydaliligi yoki boshqa agentlarning to'lovlari haqida emas, balki faqat o'zlarining foydaliligi haqida qayg'uradilar deb hisoblasak, VCG mexanizmi iqtisodiy jihatdan samarali va haqiqiy dominant strategiya.

Biror narsani yoki narsalar to'plamini sotish uchun umumiy mexanizmlardan biri bu **auksiondir**. Biror narsani sotish uchun keng tarqalgan auksion turi oshib boruvchi auksiondir, bu yerda narsaning avvalgi taklif narxi qabul qilinganida, hozirgi taklif narxi oldindan belgilangan miqdorda oshiriladi. Buyumni joriy narxda sotib olishni taklif qilish taklif deyiladi. Faqat bir kishi ma'lum bir narx uchun taklif qo'yishi mumkin. Buyum eng yuqori taklifni qo'ygan shaxsga o'tadi va u bu taklif miqdorini to'laydi.

Yagona narsani sotish uchun VCG mexanizmini ko'rib chiqing. Bir qator odamlar bor, har biri narsaga qancha qiymat berishini taklif qiladi. To'lovlarni maksimal darajada oshiradigan natija narsani eng yuqori taklif kiritgan kishiga berishdir. Agar ular ishtirok etmaganida, narsa ikkinchi eng yuqori taklif qiluvchiga o'tgan bo'lardi. Shunday qilib, VCG mexanizmi bo'yicha, eng yuqori taklif kiritgan kishi narsani olishi va ikkinchi eng yuqori taklif qiymatini to'lashi kerak. Bu **ikkinchi narx auksioni** deb ataladi. Ikkinchi narx auksioni oshib boruvchi auksionga tengdir (taklif o'sishiga qadar), bu yerda odamlar qancha to'lashni

xohlashlarini proksi taklif sifatida belgilaydilar va proksi takliflarni haqiqiy takliflarga aylantiradigan agent mavjud. Ikkinchi narx auksionida taklif kiritish oson, chunki agentlar murakkab strategik mulohaza qilishlari shart emas. G'olib va mos to'lovni aniqlash ham oson.

14.7. Ko'p agentli mustahkamlashni o'rganish

14.7.1. Mukammal ma'lumotli o'yinlar

Mukammal ma'lumotli o'yinda, agentlar harakat qilishdan oldin dunyoning holatini navbatma-navbat kuzatib boradilar va har bir agent o'z foydasini maksimal darajada oshirish uchun harakat qiladi, 13-bobdagi yagona agent uchun mo'ljallangan mustahkamlashni o'rganish algoritmlari o'zgarishsiz ishlashi mumkin. Har bir agent boshqa agentlar atrof-muhitning bir qismi deb taxmin qilishi mumkin. Bu raqib o'zining optimal strategiyasini o'ynayotgan bo'lsa yoki o'rganayotgan bo'lsa ham ishlaydi. Buning sababi shundaki, o'yin daraxtining joriy tuguni uchun agentning qiymatini ifodalovchi yagona Nash muvozanati mavjud. Bu strategiya boshqa agentlarga nisbatan eng yaxshi javobdir.

Agar raqib o'zining optimal strategiyasidan foydalanmasa yoki optimal strategiyaga yaqinlashmasa, o'quv agenti optimal bo'lmagan strategiyaga o'tishi mumkin. Raqib yomon o'ynash orqali o'rganayotgan agentni optimal bo'lmagan strategiyani bajarishga o'rgatishi va keyin raqib boshqa strategiyaga o'tib, agentning optimal bo'lmagan strategiyasidan foydalanishi mumkin. Biroq, o'quv agenti keyinchalik (hozirgi) eng yaxshi raqibdan o'rganishi mumkin.

O'yinda ikkala o'yinchining harakatlarini simulyatsiya qilish va ikkalasini ham o'rganish uchun mustahkamlovchi o'rganishdan foydalanish mumkin. Ikki o'yinchili, nol yig'indili, mukammal ma'lumotli o'yinlar uchun, minimaksda bo'lgani kabi o'yin bitta qiymat bilan tavsiflanishi mumkin, uni bir agent minimallashtirishga, ikkinchisi esa maksimallashtirishga harakat qiladi. Bunday holda, agent $Q(s, a)$ ni o'rganishi mumkin, bu qiymat s holatida bo'lish va a harakatini bajarish uchun taxminiy hisoblanadi. Algoritmlar asosan o'zgarmasdan qolishi mumkin, lekin qaysi o'yinchining navbati ekanligini bilishi kerak va Q -qiymati keyin qaysi o'yinchining navbati ekanligiga qarab maksimal yoki minimal qilib yangilanadi.

14.7.2. Stoxastik tamoyillar bilan mustahkamlashni o'rganish

Mukammal bo'lmagan ma'lumotli ko'p agentli o'yinlar uchun, shu jumladan, bir vaqtning o'zida harakat qilish o'yinlari uchun, bir nechta Nash muvozanati mavjud bo'lishi va hech qanday deterministik strategiya optimal bo'lmasiligi mumkin. Bir nechta muvozanatlar mavjud bo'lganligi sababli, ko'plab hollarda agent nima qilishini aniqlash qiyin, hatto u o'yin natijalarini va boshqa agentlarning foy-

dalarini bilsa ham. Biroq, aksariyat real strategik to'qnashuvlar ancha murakkab, chunki agentlar boshqa agentlarning natijalari yoki foydalari haqida bilishmaydi.

Nash muvozanatlari faqat tasodifiy strategiyalardan iborat bo'lgan o'yinlar uchun, masalan, futbolda penaltini tepishda (14.8-rasm), har qanday deterministik tamoyilni o'rganish yaxshi fikr emas, chunki har qanday deterministik strategiya boshqa agent tomonidan ishlatilishi mumkin. Bir nechta Nash muvozanatlariga ega bo'lgan boshqa o'yinlar uchun, masalan, 14.4-mavzuda o'rganilganlar kabi, agentlar strategiya profilida muvofiqlashtirish uchun bir nechta o'yin o'ynashlari kerak bo'lishi mumkin, bu yerda hech biri chekinishni xohlamaydi.

Mustahkamlashni o'rganishni bunday holatlarga **stoxastik tamoyillarni** aniq ifodalash orqali kengaytirish mumkin, bu tamoyillar harakatlar ustidan ehtimollik taqsimotini belgilaydi. Bu avtomatik ravishda tadqiq qilish shaklini ta'minlaydi, chunki nolga teng bo'lmagan ehtimollik bilan har qanday harakat takroran sinovdan o'tkaziladi.

Ushbu bo'lim har bir o'yinchi harakatlar uchun Dirixlet taqsimoti kabi sanash orqali stoxastik siyosatni ifodalovchi SARSA variantini taqdim etadi. Agent olgan mukofotlar asosida harakatlarning sanog'ini yangilaydi.

14.10-rasmning **stoxastik tamoyil iteratsiya** boshqaruvchisi SARSA algoritmiga o'xshaydi, lekin **stoxastik siyosatni**, har bir holat uchun harakatlar ustidan ehtimollik taqsimotini o'rganadi.

Dirixlet taqsimotida bo'lgani kabi, tamoyil raqam sifatida π bilan ifodalanadi, bunda harakat eng katta Q qiymatiga ega (shuningdek, psevdohisoblar). Stoxastik siyosat π massivini normallashtirish orqali olinadi, shunda s holatida a harakatini qilish ehtimoli $\pi[s, a] / \sum_a \pi[s, a]$ bo'ladi.

Algoritm SARSA bilan bir xil, faqat π ni eng yuqori baholangan Q -qiymatiga ega harakat uchun qiymatga bitta qo'shib yangilaydi.

Tadqiq qilish va foydalanish o'rtasidagi kelishuv uchun, P nolga yaqin bo'lmagan hisoblar bilan ishga tushiriladi (masalan, quyidagi misollarda 5 ishlatiladi), bir siyosatning tezda dominant bo'lib qolishiga yo'l qo'ymaslik va/yoki qidiruvni rag'batlantirish uchun Q yuqori qiymat bilan boshlanadi. Ushbu o'rganish boshqaruvchisining ochiq manbali ilovasi AIPython (aipython.org)da mavjud.

Agar sof strategiyalarda yagona Nash muvozanati mavjud bo'lsa va barcha agentlar ushbu algoritmdan foydalansa, ular ushbu muvozanatga yaqinlashadi. Dominant bo'lgan strategiyalarning ehtimoli nolga yaqinlashadi. 14.15-misolda, u Nash muvozanatini topadi. Xuddi shunday, 14.13-misoldagi mahbus dilemma-sida, u ikkala agent qabul qiladigan yagona muvozanatga yaqinlashadi. Shunday qilib, bu algoritm **hamkorlik** qilishni o'rganmaydi, bu yerda hamkorlik qiluvchi agentlar o'zlarining to'lovlarini maksimal darajada oshirish uchun mahbusning dilemmasidan *voz kechishadi*.

Agar bir nechta sof muvozanatlar mavjud bo'lsa, ushbu algoritm ulardan biriga yaqinlashadi. Agentlar shunday qilib **muvofiqlashtirishni** o'rganadilar. Ma-

salan, 14.12-misolda futbol-do'kon o'yinida, u ikkalasi ham do'konga yoki ikkalasi ham futbol o'yiniga borishning muvozanatlaridan biriga yaqinlashadi.

```

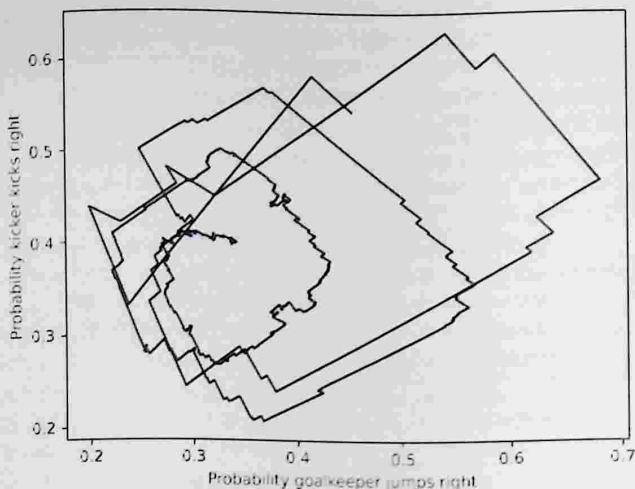
1: controller Stochastic_policy_iteration( $S, A, \gamma, \alpha, q\_init, \pi\_init$ )
2:   Inputs
3:      $S$  set of states
4:      $A$  set of actions
5:      $\gamma$  discount factor
6:      $\alpha$  step size
7:      $q\_init$  initial  $Q$  value
8:      $\pi\_init$  initial  $\pi$  value (must be greater than 0)
9:   Local
10:     $\pi[S, A]$  unnormalized probability of doing  $A$  in state  $S$ 
11:     $Q[S, A]$  an estimate of the value of doing  $A$  in state  $S$ 
12:     $\pi[s, a] := \pi\_init$  for all  $s \in S, a \in A$ 
13:     $Q[s, a] := q\_init$  for each  $s \in S, a \in A$ 
14:    observe state  $s$ , choose action  $a$  at random
15:    repeat
16:       $do(a)$ 
17:      observe reward  $r$ , state  $s'$ 
18:      select action  $a'$  based on  $\pi[s', a'] / \sum_{a'} \pi[s', a']$ 
19:       $Q[s, a] := Q[s, a] + \alpha (r + \gamma \cdot Q[s', a'] - Q[s, a])$ 
20:       $a\_best := \arg \max_a (Q[s, a])$ 
21:       $\pi[s, a\_best] = \pi[s, a\_best] + 1$ 
22:       $s := s'; a := a'$ 
23:    until termination()

```

14.10-rasm. Stoxastik tamoyillar bilan mustahkamlashni o'rganish

14.20-misol. 14.9-misoldagi penaltini tepish o'yinini ko'rib chiqing. 14.11-rasm ikki o'yinchi uchun o'quv algoritmi ishlatadigan bir ko'rsatkichni ko'rsatadi. Ushbu rasm penaltini tepish o'yinida darvozabonning o'ngga sakrashi va zarb beruvchining o'ngga tepishi ehtimollarini bir o'quv algoritmi uchun ko'rsatadi. Bu normalizatsiyalangan P funksiyasining o'ng qiymatini ko'rsatadi. Ushbu bosqichda, Q_1 ga o'rnatiladi va P dagi hisoblar 5 ga o'rnatiladi. Muvozanatda, zarb beruvchi o'ngga 0.4 ehtimol bilan tepadi va darvozabon o'ngga 0.3 ehtimol bilan sakraydi.

Bitta tasodifiy Nash muvozanatiga ega bo'lgan ikki agentli raqobatli o'yinni ko'rib chiqing. Agar A va B agentlar raqobatlashayotgan bo'lsa va agent B Nash muvozanatini o'ynayotgan bo'lsa, agent A ning yordamchi to'plamidagi qaysi harakat bajarilishi ahamiyatga ega emas; ularning barchasi A uchun bir xil qiymatga ega. Ushbu algoritm Nash muvozanatini topishga moyil va keyin bir agent muvozanatdan chetlanadi, chunki boshqa agent muvozanat strategiyasini o'ynayotganida bunday qilish uchun hech qanday jazo yo'q. Agent chetlanganidan keyin, ikkinchisi og'ishdan foydalanish uchun strategiyasini o'zgartirishi mumkin. Bu siklik xatti-harakatlarning izohlaridan biridir.



14.11-rasm. Futboldagi penalti misolida o'rganish

Ikki o'yinchining 14.10-rasmdagi stoxastik mustahkamlashni o'rganish algoritmidan foydalangan holda tamoyilning grafigi (S bitta elementga ega, $\gamma = 0$, $\alpha = 0.1$, $q_{init} = 1$, $p_{i init} = 5$) 14.9-misol uchun futbol penaltisi. Birinchi harakatdan so'ng, tamoyil pozitsiyasi (0.455, 0.545) bo'lib, chiziq bo'ylab davom etadi va (0.338, 0.399) nuqtasida tugaydi. Nash muvozanati 14.16-misolda keltirilganidek (0.3, 0.4) nuqtasidir.

Ushbu algoritmning ko'plab variantlari mavjud, shu jumladan yaqinda olingan tajribalarni ko'proq og'irlik bilan baholash, ehtimolliklarni biror konstanta bilan sozlash (ehtimolliklar 0 va 1 orasida bo'lishi va 1 ga teng bo'lishiga ishonch hosil qilish kerak) (14.9-mashqqa qarang).

Ushbu algoritmdan foydalanayotgan agent o'yinni yoki mukofotlarni bilishi shart emas. Agar agent o'yinni bilsa, boshqa oddiy strategiya – bu **xayoliy o'yin** bo'lib, unda agent boshqa o'yinchining statistikalarini to'playdi va bu statistikalar boshqa agentning stoxastik tamoyilini aks ettiradi deb hisoblab, eng yaxshi javobni o'ynaydi. Ikkala o'yinchi xayoliy o'yindan foydalansa, ko'p turdagi o'yinlar uchun (shu jumladan, ikki o'yinchi uchun nol summali o'yinlarni o'z ichiga oladi) Nash muvozanatiga erishadi.

E'tibor berib, hech qanday mukammal o'rganish strategiyasi mavjud emas. Agar qarshi agent agent Aning strategiyasini (o'rganuvchi bo'lsa yoki bo'lmasa ham) aniq bilsa va agent A nima qilishini oldindan ayta olsa, u bu bilimdan foydalanishi mumkin.

14.21-misol. Ikki agent penaltini tepish o'yinini (14.9-misol) o'ynayotganini tasavvur qiling va bitta agent (1-agent) xayoliy o'yindan foydalanmoqda. 1-agent deterministik strategiyaga ega (bir nechta holatlardan tashqari). Agar qarshi o'yinchi, 2-agent, 1-agentning fiktiv o'yindan foydalanayotganini bilsa, 1-agent nima qilishini oldindan aytib, eng yaxshi javobni bera oladi. Ular deyarli har doim 1-agentni yutishlari mumkin bo'ladi.

2-agent 1-agentni mag'lub etishi kafolatlanmaydigan yagona vaqt, 2-agentning tarixi Nash muvozanatiga mos keladigan vaqtdir, bu holda 1-agent stoxastik bo'lishi mumkin. Bunday holda, keyingi bosqichda 2-agent boshqa nisbatga ega bo'ladi va shuning uchun muvozanat strategiyasini o'ynamaydi va keyin 1-agentni mag'lub eta oladi.

Bu xususiyat xayoliy o'yinning o'ziga xosligi emas. Aslida, agar 1-agent 2-agentni xayoliy o'yindan foydalanayotganiga ishonтира olsa, unda 1-agent, 2-agent nima qilishini oldindan aytishi va shunga qarab o'z strategiyasini o'zgartirishi mumkin. Agar 2-agent, 1-agent fiktiv o'yindan foydalanmayotganligini ko'rsa, u o'z strategiyasini o'zgartirishi kerak.

14.7.3. Eng zamonaviy o'yinchilar

2016-yilda AlphaGo kompyuter dasturi dunyoning eng yaxshi Go o'yinchilaridan biri Li Sedolni beshta o'yinli Go o'yinida mag'lub etdi. AlphaZero, o'yinlar bo'yicha oldindan dasturlashtirilgan nonga ega bo'lgan davomiy dastur – o'yin simulyatori bo'lishdan tashqari – jahon darajasidagi shaxmat, shogi va Go o'ynaydi. AlphaZeroning asosiy xususiyatlari:

Bu modifikatsiyalangan tamoyil iteratsiyasini amalga oshiradi, unda chuqur neyron tarmog'i yordamida boshqaruv pozitsiyasi bilan kirish va chiqish qiymat funksiyasi (yutish uchun +1, durang uchun 0 va yutqazish uchun -1) va stoxastik tamoyil (softmax orqali mumkin bo'lgan harakatlar ustidan ehtimollik taqsimoti, yuqori ishonch chegarasini o'z ichiga oladi) hisoblanadi. Neyron tarmoq parametrlari qiymat funksiyasining kvadrat yo'qotishining yig'indisi, stoxastik tamoyilni ifodalovchi taqsimotning log yo'qotishi va L2 regulatsiya termini bo'yicha optimallashtiriladi. Har bir bosqichda, kuzatuvlar mavjud bo'lmaganligi sababli, qolgan o'yinning stoxastik simulyatsiyasini – oldindan namuna olishni amalga oshiradi va stoxastik siyosatga amal qiladi. Bu stoxastik simulyatsiya har bir harakatning kutilgan qiymatini baholash uchun ishlatiladi. Bu **Monte Karlo daraxtini qidirish (MCTS)** deb ataladi. E'tibor bering, MCTS modelga tayanadi; u har qanday nuqtadan qidirishni qayta boshlash imkoniyatiga ega bo'lishi kerak.

U birinchi va ikkinchi avlod tenzor protsessorlaridan apparat sifatida foydalanilgan holda o'n millionlab o'yinlarni o'ynab, o'ziga qarshi o'ynashga o'rgatilgan.

14.8. Ijtimoiy ta'sir

Mahbus dilemmasining ko'p agentli versiyasi **umumiy foydalanish resurslari fojiasidir**, chunki umumiy resurslar (biz nafas olayotgan havo kabi) quyidagi raqamli misolda bo'lgani kabi, o'z foydasini maksimal darajada oshirishga intilayotganlar tomonidan birinchi o'ringa qo'yilmaydi, bu esa barchaning ahvolini yomonlashtiradi.

14.22-misol. 100 kishi bor deb faraz qiling va har bir kishi tanlashi mumkin bo'lgan harakat (masalan, ichki yonuv dvigateli bo'lgan mashinani haydash) ularning 10 birlik foyda olishiga olib keladi, lekin atrof-muhitga 100 zarar keltiradi, bu esa hammaga tarqatilgan holda har bir kishi uchun -1 ga teng bo'ladi; aks holda har bir agent 1 birlik foyda oladigan va atrof-muhitga 0 zarar keltiradigan harakatni amalga oshirishi mumkin (masalan, piyoda yurish). Har bir kishi mashina haydashni ko'rib chiqqanda, ularning mukofoti 10 dan atrof-muhit ulushini ayirib tashlagan holda, bu -1 ga teng, shuning uchun mashina haydash ularga 9 birlik foyda beradi, shuning uchun ular mashina haydashni afzal ko'radilar. Bu boshqalar nima qilayotganidan mustaqil. Shunday qilib, hamma mashina haydaydi, bu esa 1000 mukofot minus 10,000, bu atrof-muhit uchun foyda, jami -9000 mukofot bo'ladi, har bir kishi uchun -90. Hech kim mashina haydamaganida, hammaga ancha yaxshi bo'lardi, hatto har bir kishi uchun mashina haydash katta foyda keltirgan bo'lsa ham.

Buni hal qilishning bir yo'li VCG mexanizmini amalga oshirishdir, bu yerda har bir kishi soliq to'laydi. Bu holda, mexanizm har bir kishiga o'z harakatining boshqalarga bo'lgan qiymatiga mos keladigan 99 birlik soliq to'lashini belgilaydi. Shunda har bir kishi uchun mashina haydamaslik asosli bo'ladi.

O'tgan misoldagi ssenariy bilan bog'liq muammo odamlarni soliqni joriy etishga ishonitirishdir. Axir, ularning mashina haydamasligiga faqat soliq sabab bo'ladi va ular mashina haydamoqchi. Yana bir katta muammo atrof-muhitni o'zgartirishning narxini hisoblashdir, bu narx uzoq vaqt davomida paydo bo'lishi mumkin, shuning uchun kelajakdagi qiymatlar qanday chegirmaga ega ekanligiga bog'liq va ta'sir boshqa joylarda bo'lishi mumkin va shuning uchun boshqalarning azoblanishi va yashash joyini yo'qotish qanday o'lganishi kerakligiga bog'liq. Bu issiqxona gazlarini chiqarishni cheklash bo'yicha sa'y-harakatlarning asosiy muammosidir. Bu individual darajada ham, mamlakat darajasida ham sodir bo'ladi, mamlakatlar uglerod chiqindilarini kamaytirmaydi, chunki ular buni xohlamaydilar.

SI turli yo'llar bilan yordam berishi mumkin. U kelajakdagi xarajatlarni yaxshiroq bashorat qilishi mumkin, ammo bu kelajakdagi qiymatlar qanday chegirmaga ega bo'lishini aniqlay olmaydi, chunki bu jamiyatning qadriyatlariga bog'liq (va har bir kishi kelajakni har xil baholashi mumkin). SI, ekstremal qarashlarning rivojlanishiga olib kelishi ma'lum bo'lgan faollikni maksimal darajada oshirish

oʻrniga [Acemoglu va boshq., 2021] ijtimoiy tarmoqlarda haqiqat ekanligi maʼlum boʻlgan postlarni targʻib qilish, yolgʻon ekanligi maʼlum boʻlgan xabarlarini targʻib qilmaslik va faqat mos qoʻshimchalarni oʻz ichiga olgan xabarlarini targʻib qilish orqali yordam berishi mumkin, Biroq, postning haqiqat yoki notoʻgʻri ekanligini aniqlash juda qiyin.

Perrault va boshqalar [2020] murakkab jamiyat muammolarini hal qilish uchun koʻp agentli tizimlar boʻyicha tadqiqot kun tartibini tasvirlaydi. Ular uchta dastur sohasida koʻplab muvaffaqiyatli amalga oshirilgan tizimlarni ifodalaydi: jamoat xavfsizligi, yovvoyi tabiatni muhofaza qilish va resurslari kam boʻlgan jamoalarda sogʻliqni saqlash. Ularning koʻp ishlari **Stackelberg xavfsizlik oʻyinlari** kontekstida tasvirlangan. Ikki oʻyinchi Stackelberg xavfsizlik oʻyini himoyachi va hujumchi oʻrtasida oʻynaladi. Cheklangan aralashuv resurslarini ishlatib, himoyachi birinchi boʻlib harakat qiladi va hujumchidan bir qator maqsadlarni himoya qiladi. Perro va boshqalar [2020] Sining ijtimoiy adolatsizlikka qarshi kurashda va jamiyatni yaxshilashda qanday muhim rol oʻynashini koʻrsatadi.

14.9. Takrorlash

Ushbu bobda koʻp agentli tizimlar bilan bogʻliq baʼzi muammolar koʻrib chiqildi.

Quyida eslab qolish kerak boʻlgan asosiy fikrlar mavjud:

Koʻp agentli tizim bir nechta agentlardan iborat boʻlib, ular mustaqil ravishda harakat qilishadi va natijalar boʻyicha oʻz foydasiga ega. Natijalar barcha agentlarning harakatlariga bogʻliq. Agentlar raqobatlashishi, hamkorlik qilishi, muvofiqlashishi, muloqot qilishi va muzokara qilishi mumkin.

Oʻyinning strategik shakli yoki oddiy shakli har bir agent uchun boshqaruvchilar berilgan holda kutilayotgan natijani aniqlaydi.

Oʻyinning keng qamrovli shakli vaqt davomida oʻyinchilarning harakatlari va maʼlumotlarini oʻyin daraxtlari nuqtayi nazaridan modellaydi.

Koʻp agentli qarorlar tarmogʻi ehtimoliy bogʻliqlik va maʼlumot mavjudligini modellaydi.

Mukammal maʼlumotli oʻyinlarni oʻyin daraxtlaridagi qiymatlarni zaxiralash yoki α - β kesish bilan minimax yordamida oʻyin daraxtini qidirish orqali hal qilish mumkin.

Qisman kuzatiladigan domenlarda baʼzan stoxastik harakat qilish optimal hisoblanadi.

Nash muvozanati – bu har bir agent uchun strategiya profili boʻlib, hech bir agent strategiya profilidan birtomonlama chetga chiqish orqali oʻz foydasini oshira olmaydi.

Toʻlovlarni kiritish orqali haqiqiy dominant strategiyaga asoslangan va iqtisodiy jihatdan samarali mexanizmi loyihalash mumkin.

O'yin nazariyasiga asoslangan SI atrof-muhitdagi ijtimoiy xulq-atvorni modellashtirish va rag'batlantirish uchun ishlatilishi mumkin.

14.10. Qo'shimcha o'qish uchun manbalar

Ko'p agentli tizimlar haqida umumiy ma'lumotlar uchun Leyton-Brown va Shoham [2008], Shoham va Leyton-Brown [2008], Wooldridge [2009], Vlassis [2007], Stone, Veloso [2000] va Jackson [2011]ning ishlarini ko'ring. Shuningdek, Kochenderfer va boshqalar [2022]ga qarang. Ko'p agentli qaror tarmoqlari Koller va Milch [2003]ning MAID(Bir nechta agentlarning ta'sir jadvallari)lariga asoslangan. Genesereth va Thielscher [2014] **umumiy o'yin o'ynashni** tasvirlaydi, bu o'yinlar uchun mantiqiy ifodalarni ishlatadi.

Minimax α - β saralash birinchi marta Hart va Edwards [1961] tomonidan nashr etilgan. Knuth, Moore [1975] va Pearl [1984] α - β saralash va boshqa o'yin daraxtlarini qidirish usullarini tahlil qiladi. Ballard [1983] minimaxni imkoniyat tugunlari bilan birlashtirishni muhokama qiladi.

1997-yil may oyida **shaxmat** bo'yicha jahon chempioni Garri Kasparovni mag'lub etgan **Deep Blue** shaxmat kompyuteri Kempbell va boshqalar [2002] tomonidan tasvirlangan. **AlphaGo** dasturi, 2016-yilda yuqori darajadagi Go o'yinchisini mag'lub etgani, Silver va boshqalar [2016] tomonidan tasvirlangan. **AlphaZero** – Silver va boshqalarning ishi [2017]. **Pluribus** [Braun va Sandholm, 2019] olti o'yinchi cheksiz Texas hold'em pokerida eng yaxshi professionallarni mag'lub etdi; u ikki o'yinchi nol summali o'yinni o'ynamagan bu g'ayritabiyy o'yinchilarning yagonasidir. **Cicero** [Baxtin va boshqalar, 2022] **Diplomatiya** o'yini uchun anonim onlayn blits ligada o'ynab, inson o'yinchilardan ikki baravar ko'proq ball to'pladi va eng yaxshi 10% ishtirokchilar qatoriga kirdi. Faqat bitta inson o'yinchi bu kompyuter dasturi deb gumon qildi.

Robot futboli Mackworth [1993] tomonidan taklif qilingan va amalga oshirilgan bo'lib, embodied AI (tanlangan sun'iy intellekt) muammosi sifatida tanilgan. Bu mavzu bo'yicha Busoni va boshqalar [2008] ko'p agentli mustahkamlashni o'rganishini ko'rib chiqadilar.

Mexanizm dizayni Shoham, Leyton-Braun [2008] va Nisan [2007] tomondan tasvirlangan. Guruh bo'lib qaror qabul qilish va o'yin nazariyasini Ordeshook [1986] tasvirlaydi.

Hardin [1968] **umumiy foydalanish resurslari fojiasi** tushunchasini kiritdi. Ostrom [1990] esa umumiy resurslar boshqarilishini va boshqarilishi mumkinligini ko'rsatdi.

Gal va Grosz [2022] texnik va axloqiy muammolarni tasvirlaydi. Perrault va boshqalar [2020] ko'p agentli tizimlarning jamoat xavfsizligi, yovvoyi tabiatni muhofaza qilish va jamoat salomatligiga qanday ijtimoiy ta'sir ko'rsatayotganini tasvirlaydi.

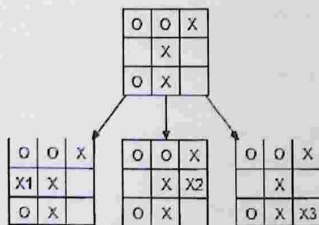
Bu bob faqat **hamkorlikda bo'lmagan o'yinlarni** qamrab oldi, bu yerda agentlar harakatlarni muvofiqlashtirmasdan, izolyatsiyada qaror qabul qiladilar. U **hamkorlikdagi o'yinlarini** qamrab olmadi, bu yerda agentlar muloqot qilishi, muzokara qilishi va ehtimol to'lovlar va bajarilishi mumkin bo'lgan mojaralarda ishtirok etishi mumkin. Yuqoridagi manbalarning ko'plari hamkorlikdagi o'yinlarini qamrab oladi. Kramár va boshqalar [2022] **Diplomatiya** o'yinini o'rganish uchun muzokara qilishni talab qiladigan SI tizimini tasvirlaydi.

Siu va boshqalar [2021] SI tizimlari insonlar bilan hamkorlikda **Hanabi** hamkorlik kartalar o'yinini qanchalik yaxshi o'ynashini baholadilar va shunday xulosaga kelishdi: «Biz insonlar qoidaga asoslangan SI hamkasbini (SmartBot) eng zamonaviy o'rganish asosidagi SI hamkasbiga (Other-Play) nisbatan deyarli barcha subyektiv metrikalarda aniq afzal ko'rishlarini va o'rganishga asoslangan agentni umumiy holda salbiy baholashlarini aniqladik, o'yin ballida statistik farq bo'lmasa ham».

14.11. Mashqlar

14.1-mashq. 14.5-rasmni tabiiy harakatlarni kiritish uchun o'zgartiring. Tabiiy harakatlarni o'z ichiga olgan mukammal ma'lumotli oddiy o'yinni sinab ko'ring (masalan, tanga yoki shoshqol toshi tashlash). Esingizda bo'lsin, o'yinning keng qamrovli shaklida *tabiat* bilan belgilangan ichki tugun o'z avlodlari bo'yicha ehtimollik taqsimotiga ega.

14.2-mashq. Tic-Tac-Toe o'yinini ko'rib chiqing, o'yin ikki o'yinchi tomonidan o'ynaladi: biri «X» o'yinchisi va ikkinchisi «O» o'yinchisi. Ular navbat bilan o'z belgilarini 3x3 o'lchamdagi o'yin taxtasidagi bo'sh joylarga qo'yadilar. O'yinchining maqsadi bir qator, ustun yoki diagonal bo'ylab uchta belgi qo'yish orqali g'alaba qozonishdir; o'yin o'yinchi g'alaba qozonganida yoki taxta to'lganida tugaydi. Quyida ko'rsatilgan o'yinda, O o'yinchisi o'zining uchinchi yurishini amalga oshirdi. Endi X o'zining to'rtinchi yurishini amalga oshirishi kerak. O'ynash agenti mavjud bo'lgan uchta yurishdan qaysi birini tanlashi kerakligini aql bilan hal qilishi kerak: X1, X2 yoki X3.



O'yin daraxtining qolgan qismini chizing. Daraxt qiymatini X g'alabasi uchun $+1$, O g'alabasi uchun -1 va durang uchun 0 deb oling. Har bir tugun uchun qiymatlarni ko'rsating. X nima qilishi kerak? G'alaba qozonishi, yutqazishi yoki durang bo'lish ehtimoli qanchalik? α - β kesish (pruning) daraxtning bir qismini kesib olishi mumkinmi?

14.3-mashq. 14.11-misoldagi kalxat-kabutar o'yini uchun, $D > 0$ va $R > 0$ bo'lganda, har bir agent o'z foydasini maksimallashtirishga harakat qiladi. Tasodifiy strategiya bilan Nash muvozanati bormi? Ehtimollar qanday? Har bir agentning kutilayotgan foydasi qanday? Bularni R va D funksiyalari sifatida ifodalang. Hisob-darsliklaringizni ko'rsating.

14.4-mashq. Quyidagi normal ko'rinishdagi o'yinlardan qaysi birida sof strategiyalardan tashkil topgan Nash muvozanati mavjud? Sof strategiyali Nash muvozanatini belgilash uchun, ularni aniqlang. Sof strategiyali Nash muvozanati mavjud bo'lmaganlari uchun, nima uchun sof strategiyali Nash muvozanati yo'qligini tushuntiring.

2-o'yinchi

1-o'yinchi

	a2	b2
a1	10,10	110,0
b1	0,110	100,100

2-o'yinchi

	a2	b2
a1	10,10	11,20
b1	0,11	20,1

1-o'yinchi

2-o'yinchi

	a2	b2
a1	10,20	5,10
b1	7,11	20,12

1-o'yinchi

14.5-mashq. 14.12-misolda, tasodifiy qilingan strategiyalar bilan Nash muvozanati qanday? Ushbu muvozanatda har bir agentning kutilayotgan qiymati-chi?

14.6-mashq. Quyidagi normal shakldagi o'yinni ko'rib chiqing, unda qator o'yinchisi A , B yoki C harakatlarini va ustun o'yinchisi D , E yoki F harakatlarini tanlashi mumkin:

	D	E	F
A	40,40	120,10	60,30
B	30,60	110,60	90,90
C	30,110	100,120	

bu yerda juftliklar qator o'yinchisi uchun natija qiymatini, so'ngra ustun o'yinchisi uchun qiymatni beradi.

(a) Ustun strategiyalarni istisno qilishda, qanday strategiyalar (agar bo'lsa) istisno qilinadi? Nimalar istisno qilinganini va nima uchun istisno qilinmaganini tushuntiring.

(b) Ushbu o'yin uchun Nash muvozanatini aniqlang. (Tasodifiy qilingan strategiya uchun faqat randomizatsiya qilingan harakatlarni ko'rsating; ehtimollarni ko'rsatishingiz shart emas.) Nima uchun bu Nash muvozanati ekanligini tushuntiring.

(c) Bir nechta Nash muvozanati bormi? Agar shunday bo'lsa, boshqa birini ko'rsating. Agar bo'lmasa, nima uchun boshqa yo'qligini tushuntiring.

(d) Agar agentlar hamkorlik qilsa, Nash muvozanatidan yaxshiroq natijaga erisha oladimi? Nima uchun yoki nima uchun emasligini tushuntiring.

14.7-mashq. Quyidagi o'yinlar uchun oldingi mashqdagi kabi savollarga javob bering:

(i)	2,1110,103,12				(ii)	A	80,130	20,10	130,80
A B	5,7	12,1	6,5		B		130,80	30,20	80,130
C	6,5	13,2	4,6		C		20,10	100,100	30,20

14.8-mashq. Ketma-ket mahbus dilemmasini ko'rib chiqing.

(a) Agar agentlar belgilangan miqdordagi vaqt davomida (masalan, uch marta) o'ynasa, ikkita yoki undan ko'p muvozanat bo'lsa, ikkita muvozanatni ko'rsating, aks holda, yagona muvozanatni ko'rsating va nima uchun faqat bitta ekanligini tushuntiring. [Maslahat: Oxirgi vaqtni avval ko'rib chiqing.]

(b) Agar har bir bosqichda to'xtash ehtimoli γ bo'lsa, γ chegirma faktoriga ega, tit-for-tat (bir-biriga qarshi) harakat Nash muvozanati bo'ladimi? Agar shunday bo'lsa, buni isbotlang. Agar bo'lmasa, qaysi γ qiymatlarining Nash muvozanati ekanligini aniqlang.

14.9-mashq. 14.10-rasmdagi stoxastik tamoyil iteratsiyasi algoritmidagi P ehtimolini yangilashning quyidagi muqobil usullarini ko'rib chiqing.

(i) So'nggi tajribalar ko'proq og'irlikka ega bo'lishi uchun P dagi qiymatlarni $(1-\beta)$ ga ko'paytirib, keyin eng yaxshi harakatga 1 qo'shib qo'yning kichik β kabi, (masalan, 0.01).

(ii) Eng yaxshi harakat ehtimoliga kichik bir qiymat qo'shish (masalan, 0.01 yoki 0.001) va boshqa harakatlardan qiymatlarni olib tashlash orqali, ehtimollarining manfiy bo'lmashligini va 1 ga teng bo'lishini ta'minlang.

(a) 14.15-misol o'yinida, yagona Nash muvozanati mavjud bo'lsa-da, boshqa strategiya profilining ikkala agent uchun ham yaxshiroq bo'lganida, (i) yoki (ii) qaysi biri eng yaxshi natijalarga ega?

(b) 14.9-misoldagi penalti tepish o'yinida boshqalarga qarshi o'ynaganida qaysi biri eng yaxshi natijaga ega?

(c) Qaysi asl va muqobil strategiyalar (agar mavjud bo'lsa) o'ynashida Nash muvozanatiga erishadi (o'ynagan barcha harakatlar bo'yicha o'rtacha hisoblanadi)? (Buni eksperimental ravishda qiling, kuzatishlaringizdan farazlarni yarating va keyin farazlaringizni isbotlashga harakat qiling.)

14.10-mashq. 14.10-rasmdagi stoxastik tamoyil iteratsiyasi algoritmi SARSA asosida yaratilgan. Buni Q -o'rganish kabi tamoyildan tashqari bo'lishi uchun qanday o'zgartirish mumkin? [Maslahat: Q -o'rganish eng yaxshi harakatdan foydalanib yangilanadi, SARSA tamoyildan foydalanib yangilanadi va stoxastik tamoyil iteratsiyasi kutilgan qiymatdan foydalanib yangilanadi.] Bu misol domenlari uchun yaxshiroq ishlaydimi?

V BO'LIM. SHAXSLAR VA MUNOSABATLARNI IFODALASH

Agent murakkab muhitlarda shaxslar (narsalar, subyektlar, obyektlar) va ularning orasidagi murakkab munosabatlarda qanday aql yuritishi, o'rganishi va harakat qilishi mumkin?

15-bob. SHAXSLAR VA MUNOSABATLAR

Haqiqiy dunyo o'zining haqiqiy tuzilmasiga ega. Ong dasturi bu dunyo bilan o'zaro keng qamrovli aloqada bo'lishga o'rgatilgan va shuning uchun dunyo tuzilishini aks ettiruvchi va undan qanday foydalanishni biladigan kodni o'z ichiga oladi. Ushbu kod dunyodagi haqiqiy obyektlarni aks ettiradi va ularning o'zaro ta'sirini ifodalaydi. Kod asosan modulli bo'lib ..., turli xil obyektlar bilan ishlash uchun modullar va ko'plab turdagi obyektlar bo'ylab umumlashmalarni o'z ichiga oladi... Modullar haqiqiy dunyoni aks ettiradigan va uning qanday rivojlanishi haqida aniq bashorat qiladigan tarzda o'zaro ta'sir qiladi...

Siz qaror qabul qilish va harakat qilish uchun dunyo tuzilishidan foydalanasiz. Kategoriyalar orasidagi chegarani belgilaganingizda, siz uchun alohida obyekt yoki obyektlar sinfini nima tashkil etishini, tasniflashni amalga oshiradigan ongingiz dasturi belgilaydi. Bu tasnif tasodifiy emas, balki dunyoning ixcham tavsifini, xususan, dunyo tuzilishidan foydalanish uchun foydali tavsifni aks ettiradi.

– Eric B. Baum [2004, 169–170-betlar]

Ushbu bob shaxslar (narsalar, subyektlar, obyektlar) va ularning orasidagi munosabatlarni qanday ifodalash haqida. Baum yuqoridagi iqtibosda ta'kidlaganidek, haqiqiy dunyo obyektlarni va ularning ixcham tasvirlarini o'z ichiga oladi, shuningdek, bu obyektlar o'rtasidagi munosabatlar ular haqida mulohaza yuritishni osonlashtirishi mumkin. Bunday tasvirlar faqat xususiyatlar asosida ifodalgan tasvirlarga qaraganda ancha ixcham bo'lishi mumkin. Ushbu bob mantiqiy tasvirlarni ko'rib chiqadi va bunday tasvirlarni ma'lumotlar bazalariga tabiiy til interfeyslari uchun qanday foydalanishni batafsil misollarini keltiradi. Keyingi boblar bilim graflari, ontologiyalar va ma'no almashish va munosabatlarni o'rganish va noaniqlik sharoitida o'rganish va mulohaza yuritish bilan munosabatlarni birlashtirishni ko'rib chiqadi.

15.1. Munosabatlar tuzilmasidan foydalanish

Sining asosiy darslaridan biri shundaki, muvaffaqiyatli agentlar dunyoning tuzilmasidan foydalanadilar. Oldingi boblar holatlarni qanday ifodalash, o'rganish va xususiyatlar nuqtayi nazaridan aql yuritish mumkinligini ko'rsatdi. Domenlarni xususiyatlar yordamida ifodalash, holatlarni aniq ifodalashdan ancha ixchamdir

va bu ixchamlikdan foydalangan algoritmlar ancha samarali bo'lishi mumkin. Biroq, odatda, ifodalash va xulosa chiqarish uchun xususiyatlardan tashqari foydalanish mumkin bo'lgan ko'proq tuzilma mavjud. Xususan, ushbu bob shaxslar va munosabatlar nuqtayi nazaridan aql yuritishni ko'rib chiqadi:

Shaxslar dunyodagi narsalar bo'lib, ular aniq shaxslar, masalan, odamlar va binolar, tasavvuriy shaxslar, masalan, afsonaviy maxluqlar va Tyuring testidan muvaffaqiyatli o'ta oladigan dasturlar, jarayonlar, masalan, darslik o'qish yoki dam olish, yoki mavhum tushunchalar, masalan, pul, kurs (ma'lum bir universitetdagi MCS-224 kursi kabi), kursning taklifi (masalan, 2026-yil bahorida taklif etilgan MCS-224 kursi) va vaqt kabi mavhum tushunchalar bo'lishi mumkin. Shaxslar, shuningdek, **subyektlar**, **obyektlar** yoki **narsalar** deb ataladi.

Munosabatlar ushbu shaxslar haqida nima to'g'ri ekanligini belgilaydi. Bu iloji boricha umumiy bo'lishi kerak va bir shaxsga nisbatan to'g'ri yoki noto'g'ri bo'lgan **birtomonlama munosabatlarni**, shaxslaridan mustaqil ravishda to'g'ri yoki noto'g'ri bo'lgan **takliflarni**, shuningdek, bir nechta shaxslar o'rtasidagi munosabatlarni o'z ichiga oladi.

15.1-misol. 5.8-misoldagi elektr domenini tasvirlashda *up_s2*, *up_s3* va *ok_s2* takliflar ichki tuzilishga ega emas. *up_s2* va *up_s3* takliflar bir xil munosabat bilan, lekin turli shaxslar bilan bog'liq ekanligi yoki *up_s2* va *ok_s2* bir xil tugma haqida ekanligi tushunchasi yo'q. Shaxslar va munosabatlar tushunchasi yo'q.

Boshqa bir variant individual tugmalar *s1*, *s2*, *s3* va xususiyatlar *up* va *ok* ni aniq ifodalashdir. Ushbu tasvirlash usulida «s2 tugmasi yuqorida» *up(s2)* deb ifodalanadi. *up* va *s1* nimani ifodalashini bilib, *up(s1)* ni alohida aniqlash talab qilinmaydi. *Connected to* kabi ikkitomonlama munosabat ikki shaxsni bog'lash uchun ishlatilishi mumkin, masalan, *connected to(w1,s1)*.

Shaxslar va munosabatlar nuqtayi nazaridan modellashtirish faqat xususiyatlardan foydalanishga nisbatan bir qator afzalliklarga ega:

- Bu ko'pincha tabiiy ifodadir. Ko'pincha xususiyatlar shaxslarning xususiyatlari bo'lib, bu ichki tuzilma xususiyatlarga aylantirilganda yo'qoladi.

- Agent ma'lum bir domenni qanday shaxslar mavjudligini yoki ularning nechta bo'lishini bilmasdan modellashtirishi kerak bo'lishi va shu sababli, qanday xususiyatlar mavjudligini ham bilmasligi mumkin. Muhit bilan o'zaro ta'sir qilayotganda, agent qaysi shaxslar aynan shu muhitda ekanligini aniqlagach, ushbu xususiyatlarni o'zi yaratishi mumkin.

- Agent ba'zi xulosalarni aniq shaxslarni aniqlashga ehtiyoj sezmasdan chiqara oladi. Misol uchun, u biron-bir shaxsni aniqlamasdan, biror narsa barcha shaxslarga tegishli degan xulosaga kelishi mumkin. Agent boshqa shaxslarni aniqlamasdan, ba'zi xususiyatlarga ega bo'lgan ba'zi bir shaxs borligi haqida xulosa chiqarishi mumkin. Agent shaxslarni farqlamasdan javob berishi mumkin bo'lgan ba'zi so'rovlar bo'lishi mumkin. Masalan, tozalash robotlari tartibsizlikni kim qilganini bilmasdan, kimdir yoki nimadir tartibsizlik yaratganini aniqlay olishi mumkin.

• Shaxslarning mavjudligi harakatlarga bogʻliq boʻlishi yoki noaniq boʻlishi mumkin. Masalan, ishlab chiqarish kontekstida rejalashtirishda ishchi komponentning mavjudligi koʻplab boshqa kichik komponentlarning ishlashi va toʻgʻri yigʻilishiga bogʻliq boʻlishi mumkin; ularning baʼzilari agentning harakatlariga bogʻliq boʻlishi mumkin, baʼzilari esa agentning nazorati ostida boʻlmasligi mumkin. Shunday qilib, agent bu yerda qanday xususiyatlar borligini yoki qanday xususiyatlar boʻlishini bilmasdan harakat qilishi mumkin.

• Koʻpincha agent fikr yuritadigan cheksiz koʻp shaxslar va shuning uchun cheksiz koʻp xususiyatlar mavjud boʻladi. Masalan, agar shaxslar jumlar boʻlsa, agent faqat juda cheklangan jumlar toʻplami haqida mulohaza yuritishi kerak boʻlishi mumkin (misol uchun, inson tomonidan aytilishi mumkin boʻlgan jumlar yoki mantiqan yaratilishi mumkin boʻlgan jumlar), hatto cheksiz koʻp jumlar mavjud boʻlishi mumkin va shuning uchun cheksiz koʻp xususiyatlar mavjud boʻladi.

15.2. Belgilar va semantika

Mantiqdan foydalanishning asosiy gʻoyasi shundan iboratki, agar bilim bazasini ishlab chiquvchilar belgilashni istagan maʼlum bir dunyoga ega boʻlsalar, ular ushbu dunyoni **belgilangan talqin** sifatida tanlashlari, talqinga nisbatan belgilarining maʼnolarini tanlashlari va oʻsha dunyoda nima toʻgʻri ekanligini aksioma sifatida yozishlari mumkin. Tizim bilim bazasining mantiqiy natijasini hisoblab chiqqanda, belgilar maʼnosini biladigan foydalanuvchi bu javobni belgilangan talqinga nisbatan talqin qilishi mumkin. Belgilangan talqin model boʻlgani uchun va mantiqiy natija barcha modellarda toʻgʻri boʻlganligi sababli, mantiqiy natija belgilangan talqinda toʻgʻri boʻlishi kerak. Ushbu bob shaxslar va munosabatlar haqida fikr yuritish imkonini berish uchun taklif mantiqini kengaytiradi. Atom takliflar endi munosabatlar va shaxslar nuqtayi nazaridan ichki tuzilishga ega.

15.2-misol. 15.1-rasmda semantikaning shaxslar va munosabatlar bilan umumiy gʻoyasi tasvirlangan. Bilim bazasini loyihalash bilan shugʻullanadigan shaxs belgilar maʼnosi haqida tasavvurga ega. Shaxs *kim*, *r123* va *in* belgilarining domeidagi maʼnolarini biladi va kompyuterga ifoda tilidagi jumlar bazasini taqdim etadi. Bilimlar *«kim»123da* va *«r123 cs binosining bir qismi»* kabi aniq faktlarni va *«agar har qanday Z, Y ning bir qismi boʻlsa va har qanday X, Z da boʻlsa, X, Y da boʻladi»* degan umumiy qoida kiritilgan jumalarni oʻz ichiga oladi, bu yerda *X, Y* va *Z* mantiqiy oʻzgaruvchilar hisoblanadi. Ushbu jumlar bu shaxs uchun maʼnoga ega. Ular bu belgilar yordamida oʻziga xos maʼnoga ega boʻlgan soʻrovlar berishi mumkin. Kompyuter bu jumla va soʻrovlarni qabul qiladi va javoblarni hisoblaydi. Kompyuter belgilar nima maʼnoga ega ekanligini bilmaydi. Biroq, maʼlumotni taqdim etgan shaxs belgilar bilan bogʻliq maʼnodan foydalangan holda javobni dunyo bilan bogʻlab talqin qilishi mumkin.

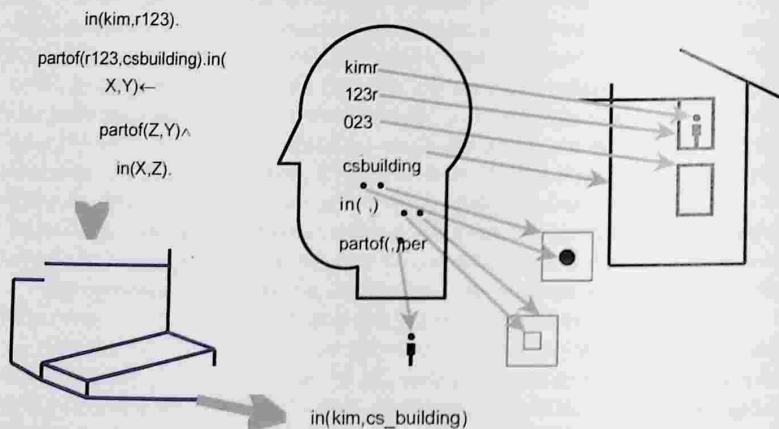
Ongdagi belgilar va shaxslar o'rtasidagi o'zaro bog'liqlik va ushbu belgilar bilan belgilangan munosabatlar konseptualizatsiya deb ataladi.

Ushbu bob konseptualizatsiyaning foydalanuvchining ongida yoki norasmiy ravishda izohlarda yozilgan deb faraz qiladi. Konseptualizatsiyani oydinlashtirish formal ontologiyaning vazifasidir.

To'g'ri javob qanday hisoblanishidan qat'i nazar aniqlanadi. Bilim bazasining to'g'riligi so'rovlarni isbotlash uchun maxsus algoritmi bilan emas, semantika bilan belgilanadi. Xulosa qilish algoritmi semantikaga mos kelsa, samaradorlikni oshirish uchun uni optimallashtirish mumkin. Ma'no va hisoblashning bu ajratilishi agentga aniqlikni saqlab qolgan holda samaradorlikni optimallashtirish imkonini beradi.

15.3. Predikatlar hisobi

Predikativ (Obyektlarning xususiyatlarini yoki obyektlar orasidagi munosabatlarni ifodalaydi) **hisoblash**, ko'pincha **predikat mantiq** deb ataladi, propozitsional hisoblashni ikki yo'nalishda kengaytiradi:



15.1-rasm. Semantikaning roli. Belgilarning ma'nosi foydalanuvchining ongida mavjud. Kompyuter belgilarni qabul qiladi va belgilarni chiqaradi. Chiqish foydalanuvchi tomonidan belgilarni qanday ma'noda qo'llaganiga qarab talqin qilinishi mumkin

Atomlar tuzilishga ega bo'lib, mantiqiy o'zgaruvchilarni o'z ichiga olishi mumkin. Mantiqiy o'zgaruvchilarni keltirish.

Ushbu darslik, o'zgaruvchilar bosh harf bilan boshlanadigan **Prolog** sintaksisi qoidalariga amal qiladi. Matematikada o'zgaruvchilar odatda x , y va z ko'rinishda

bo'ladi. Prologda, boshqa dasturlash tillarida bo'lgani kabi, uzun nomlar odatda kodni o'qish uchun qulayroq qilish uchun ishlatiladi. Katta harflar o'zgaruvchilarni ta'kidlash uchun ishlatiladi.

Predikativ hisobining **sintaksisi** quyidagicha harflar, raqamlar yoki pastki chiziq («_») ketma-ketligi sifatida aniqlangan **belgi** orqali propozitsional hisoblardan kengayadi:

- **Mantiqiy o'zgaruvchi** katta harf yoki «_» bilan boshlanadigan belgi. Masalan, *X, Room, B4, Raths* va *The_big_guy* barchasi o'zgaruvchilar.

- **Konstanta** kichik harf bilan boshlanadigan, belgi yoki son doimiysi yoki satrdir.

- **Predikativ belgi** kichik harf bilan boshlanadigan belgi. Konstanta va predikativ belgilari bilim bazasidagi kontekst bilan farqlanadi. Masalan, *kim, r123, f.grandfather* va *borogroves* konstanta yoki predikativ belgisi bo'lishi mumkin; 725 esa konstantadir.

- **Termin** o'zgaruvchi yoki konstantadir.

Masalan, *X, kim, cs422, mome*, yoki *Raths* termin bo'lishi mumkin.

- **Atomik belgi** yoki oddiygina **atom** shakli p yoki $p(t_1, \dots, t_n)$ bo'lib, bu yerda p predikativ belgisi va har bir t_i termin. Har bir t_i predikativ uchun **argument** deyiladi. Masalan, *teaches (sue, cs422)*, *in (kim, r123)*, *father (bill, Y)*, *happy (C)*, *outgrabe (mome, Raths)* va *sunny* atomlar bo'lishi mumkin. Atom *outgrabe (mome, Raths)* kontekstida *outgrabe* predikativ belgisi va *mome* konstanta bo'ladi.

- **Mantiqiy formula** atomlardan propozitsiyaviy hisob bog'lovchilari va miqdor ko'rsatkichlari yordamida quriladi. Quyidagilar ham mantiqiy formulalardir:

- $\forall Xp$, bu yerda X o'zgaruvchi va p mantiqiy formula (odatda X ni o'z ichiga olgan), «**barcha uchun** X, p » deb o'qiladi. X o'zgaruvchisi **universal miqdor** deb ataladi.

- $\exists Xp$, bu yerda X o'zgaruvchi va p mantiqiy formula, « p shunday X mavjud» deb o'qiladi. X o'zgaruvchisi **ekzistensial miqdoriy** deb aytiladi.

15.3-misol. Quyidagi mantiqiy formulalar:

$$\forall G(\text{grandfather}(\text{sam}, G) \leftarrow \exists P(\text{father}(\text{sam}, P) \wedge \text{parent}(P, G)))$$
$$\text{in}(X, Y) \leftarrow \text{partof}(Z, Y) \wedge \text{in}(X, Z)$$
$$\text{slithy}(\text{toves}) \leftarrow \forall R(\text{Raths}(\text{mimsy} \wedge \text{borogroves} \wedge \text{outgrabe}(\text{mome}, R)))$$

Matndan kelib chiqqan holda, *sam, toves* va *mome* konstantalar; *grandfather, father, parent, in, part of, slithy, mimsy, borogroves* va *outgrabe* predikativ belgilar; *G, P, X, Y, Z* va *Raths* o'zgaruvchilardir.

Birinci ikkita formulalar, hatto formulalarning ma'nosi uchun aniq ko'rsatilgan rasmiy spetsifikatsiyasiz ham intuitiv ma'noga ega bo'lishi mumkin. Biroq, mnemonik nomlarning tavsiya etilishidan qat'i nazar, kompyuterga kelsak, birinchi ikkita formula uchinchidan ko'ra ko'proq ma'noga ega emas. Ma'no faqat semantika orqali ta'minlanadi.

Ifoda hech qanday o'zgaruvchisi bo'lmasa, **asosiy** hisoblanadi. Masalan, *teaches(chris, cs322)* asosiy, ammo *teaches(Prof, Curse)* asosiy emas.

Keyingi bo'limlarda semantika aniqlanadi. Avval asosiy iboralarni ko'rib chiqamiz, so'ngra o'zgaruvchilarni kiritish uchun semantikani kengaytiramiz.

15.3.1. Asosiy mantiqiy formulalarning semantikasi

Predikativ hisoblash semantikasini berishning birinchi bosqichi – asosiy (o'zgaruvchisiz) holat uchun semantikani berishdir. **Talqin** uchlik shaklida ifodalanadi: $I = \langle D, \varphi, \pi \rangle$:

- D – bo'sh bo'lmagan to'plam, u **domen** deb ataladi. D elementlari **shaxslar** deb ataladi.

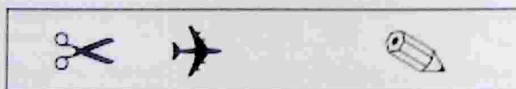
- φ – har bir konstantaga D elementini tayinlovchi funksiya;

- π – har bir n -ary predikativ belgisi uchun D^n dan $\{true, false\}$ qiymatlariga xarita beruvchi funksiyadir.

φ dunyoda nomlardan shaxslarga o'tish funksiyasi. c konstantasi $\varphi(c)$ shaxsni ifodalaydi. Bu yerda c bir belgi, lekin $\varphi(c)$ har qanday narsa bo'lishi mumkin: haqiqiy jismoniy shaxs (masalan, kishi yoki virus), abstrakt tushuncha (masalan, kurs, sevgi, 2 raqami) yoki belgi.

$\pi(p, n)$ -ary predikativ belgisi bilan belgilangan munosabat har bir n -juftlik uchun to'g'ri yoki noto'g'ri ekanligini aniqlaydi. Agar predikativ belgisida hech qanday argument bo'lmasa, $\pi(p)true$ yoki $false$ bo'ladi. Shunday qilib, argumentlari bo'lmagan predikativ belgilari uchun bu semantika, taklif hisobining semantikasiga qisqaradi.

15.4-misol. Jadvaldagi uchta individdan iborat dunyoni ko'rib chiqamiz:



Ushbu obyektlar haqiqiy dunyodagi narsalar bo'lgani uchun shu tarzda tasvirlangan, ya'ni ular oddiygina belgilar emas. ✂ Bir juft qaychi, ✈ samolyot modeli va ✎ qalam mavjud. Har bir obyektning o'ziga xos xususiyatlari va vazifalari bor va ular mantiqiy tizimlarda modellashirilganda real dunyodagi mavjudliklari aks ettiriladi.

Tasavvur qilaylik, bizning tilimizdagi konstantlar – *plane*, *pencil* va *model* va predikativ belgilar – *noisy* va *left of*. *Noisy* – bir argumentli predikativ (unary predicate) va *left of* – ikki argumentli predikativ (binary predicate) bo'lsin.

Jadvaldagi individlarni ifodalovchi misol talqini:

- $D = \{ \text{✂}, \text{✈}, \text{✎} \}.$

$$\bullet \phi(plane) = \rightarrow, \phi(pencil) = \rightarrow, \phi(model) = \rightarrow.$$

$$\bullet \pi(noisy): \begin{array}{|c|c|c|} \hline \langle \rightarrow, \rightarrow \rangle & false & \langle \rightarrow, \rightarrow \rangle & true & \langle \rightarrow, \rightarrow \rangle & false \\ \hline \end{array}$$

$$\pi(left_of): \begin{array}{|c|c|c|} \hline \langle \rightarrow, \rightarrow \rangle & false & \langle \rightarrow, \rightarrow \rangle & true & \langle \rightarrow, \rightarrow \rangle & true \\ \hline \langle \rightarrow, \rightarrow \rangle & false & \langle \rightarrow, \rightarrow \rangle & false & \langle \rightarrow, \rightarrow \rangle & true \\ \hline \langle \rightarrow, \rightarrow \rangle & false & \langle \rightarrow, \rightarrow \rangle & false & \langle \rightarrow, \rightarrow \rangle & false \\ \hline \end{array}$$

• *Noisy* bir argumentli predikat bo'lgani uchun, u har bir individ uchun haqiqiy qiymatiga ega. *Left_of* ikki argumentli predikat bo'lgani uchun, u ikki individni qabul qiladi va birinchi element ikkinchi elementning chap tomonida joylashgan bo'lganda to'g'ri (true) bo'ladi. Masalan, $\pi(left_of)(\langle \rightarrow, \rightarrow \rangle) = true$, chunki qaychi samolyotning chap tomonida; $\pi(left_of)(\langle \rightarrow, \rightarrow \rangle) = false$, chunki qalam o'zining chap tomonida joylashgan emas.

• E'tibor bering, *D* real dunyodagi narsalar to'plamini ifodalaydi. Munosabatlar individlar o'rtasida bo'lib, ularning nomlari o'rtasida emas, $\phi plane$ va *model* bir xil individga tegishli ekanligini belgilasa, bu interpretatsiyada ular haqida aynan bir xil bayonotlar to'g'ri bo'ladi.

15.5-misol. 15.1-rasmdagi talqinni ko'rib chiqing. *D* to'rt elementdan iborat to'plam: Kim shaxs, xona 123, xona 023 va CS binosi. Bu to'rtta belgilar to'plami emas, balki haqiqiy shaxs, haqiqiy xonalar va haqiqiy bino to'plamidir. Bu to'plamni yozib qo'yish qiyin va baxtimizga, hech qachon kerak emas. Ma'noni eslab qolish va boshqalarga yetkazish uchun bilim bazasi dizaynerlari odatda *D*, ϕ va π ni jismoniy shaxslarni ko'rsatish yoki ularni tasvirlash (15.1-rasmda bo'lgani kabi) yoki tabiiy tilda ma'noni tavsiflash orqali ta'riflaydi. Konstantalar *kim*, *r123*, *r023* va *cs_building*. ϕ xaritasi 15.1-rasmdagi kulrang yoylar bilan har bir konstantadan dunyodagi shaxsga o'tkaziladi. Predikativ belgilar *person*, *in* va *part_of*. Ularning ma'nolari yoylar orqali tasvirlanadi.

Shunday qilib, Kim deb atalgan shaxs *r123* xonasida va CS binosida va bu *in* munosabatining to'g'ri bo'lgan yagona misollari. Shunga o'xshab, *r123* va *r023* xonalar CS binosining bir qismidir va bu talqinda boshqa qism munosabatlari yo'q. Har bir asosiy termin talqinda shaxsni bildiradi. Konstanta *c* talqinda $\phi(c)$ shaxsini bildiradi. Asosiy atom talqin qilishda to'g'ri yoki noto'g'ri bo'ladi. Atom $p(t_1, \dots, t_n)$ uchun **haqiqat**, agar $\pi(p)(t_1, \dots, t_n) = true$ bo'lsa, bu yerda t_i termining belgilangan shaxsidir va aks holda *I* uchun **yolg'ondir**.

15.6-misol. *in(kim, r123)* atomi 15.5-misoldagi talqinda haqiqat, chunki *kim* tomonidan bildirgan shaxs haqiqatan ham *r123* tomonidan bildirgan xonada. Shunga o'xshash, *person(kim)* haqiqat, shuningdek, *part_of(r123, cs_building)* kabi. *in(cs_building, r123)* va *person(r123)* atomlari bu talqinda yolg'on.

Mantiqiy bog'lamlar taklif hisoblashda qanday ma'noga ega bo'lsa, xuddi shunday ma'noga ega; mantiqiy bog'lamlar uchun rostlik jadvallari 5.1-rasmda berilgan.

15.3.2. O'zgaruvchilarni talqin qilish

O'zgaruvchilar bilan mantiqiy formulalarning semantikasini aniqlash uchun o'zgaruvchilar tayinlash, ρ . D domenidagi o'zgaruvchilar funksiyasidir. Shunday qilib, o'zgaruvchilar tayinlash har bir o'zgaruvchiga domen elementini tayinlaydi. D , φ , π talqini va ρ o'zgaruvchilar tayinlash berilgan holda, har bir termin domen ichidagi shaxsni bildiradi. Agar termin konstanta bo'lsa, shaxs φ tomonidan beriladi. Agar termin o'zgaruvchi bo'lsa, shaxs ρ tomonidan beriladi. Barcha o'zgaruvchilar uchun talqin va o'zgaruvchilar tayinlash berilgan holda, har bir atom oldingi aniqlashdan foydalangan holda yoki haqiqat yoki yolg'on bo'ladi.

Kvantifikatsiya semantikasi quyidagicha:

Universal miqdoriy formula $\forall X p$ talqinda to'g'ri hisoblanadi, agar p , X ning har bir shaxsga o'zgaruvchilar tayinlashlari uchun haqiqat bo'lsa. Shunday qilib, formula yolg'on bo'lgan X ni belgilaydigan o'zgaruvchi mavjud bo'lganda, formula yolg'on talqin qilinadi. Formula p , X ning **doirasi** hisoblanadi.

Ekzistensial miqdoriy formula $\exists X p$ talqinda to'g'ri hisoblanadi, agar p , X ning biror-bir o'zgaruvchilar tayinlashi uchun haqiqat bo'lsa. Mavjud bo'lgan shaxs $\exists X p$ ning doirasida bo'lgan umumiy miqdoriy o'zgaruvchilarga bog'liq bo'lishi mumkin. Masalan, $\forall Y (\exists X p)$ ifodasida, X o'zgaruvchisi Y ga bog'liq bo'lishi mumkin; har bir Y raqam uchun shunday bir X mavjudki, $X = Y + 1$. Ifodadagi $X (Y p)$, da, X , Y ga bog'liq bo'lishi mumkin emas, chunki X , Y ning doirasida emas; shunday X mavjud emaski, har bir raqam Y uchun $Y, X=Y+1$ bo'lsin.

Ichki kvantifikatsiyaga ega bo'lmagan o'zgaruvchini o'z ichiga olgan formula **ochiq** deb ataladi va rostlik qiymatiga ega emas, chunki barcha o'zgaruvchilar uchun o'zgaruvchilar tayinlash mavjud emas.

15.7-misol. Quyidagi formula

$$\forall X \forall Y \text{partof}(X, Y) \leftarrow \text{in}(X, Y).$$

15.5-misoldagi talqinda yolg'on, chunki X , Kim va Y , Room 123 ni bildirgan o'zgaruvchilar tayinlashi ostida $\text{in}(X, Y)$ rost va $\text{part_of}(X, Y)$ yolg'on bo'ladi.

Quyidagi formula

$$\forall X \forall Y \forall Z \text{in}(X, Y) \leftarrow \text{partof}(Z, Y) \wedge \text{in}(X, Z)$$

bu talqinda to'g'ri, chunki $\text{in}(X, Y)$ ning $\text{part_of}(Z, Y) \wedge \text{in}(X, Z)$ qismi to'g'ri bo'lgan barcha o'zgaruvchilarni belgilashda to'g'ri bo'ladi.

Mantiqiy natija 5.1.2-mavzuda taklif hisobi uchun belgilanganidek, KB ning **mantiqiy natijasi** sifatida aniqlanadi: g , KB ning mantiqiy natijasi bo'lsa, bu $KB \models g$, shaklida yoziladi, agar g , KB ning har bir modelida to'g'ri bo'lsa.

15.8-misol. Tasavvur qilingki, *KB* bilim bazasi quyidagi formulalar birlashmasidan iborat:

In(kim, r123)

part of(r 123, cs building).

$\forall XY \forall Z \text{ in } (X,Y) \text{ part of } (Z,Y) \wedge \text{in}(X,Z).$

15.5-misolda ta'riflangan interpretatsiya *KB* modelidir, chunki har bir formula bu interpretatsiyada haqiqatdir.

KB |= *in(kim,r123)*, chunki bu bilimlar bazasida aniq ko'rsatilgan. Agar *KB* dagi har bir formula talqinda rost bo'lsa, u holda *in(kim,r123)* o'sha talqinda to'g'ri bo'lishi kerak. *KB* |= *in(kim,r023)*. 15.5-misolda ta'riflangan talqin *KB* modeli bo'lib, unda *in(kim,r023)* yolg'ondir.

KB |= *part_of(r023,cs_building)*. Garchi *part_of(r023,cs_building)* 15.5-misolda ta'riflangan talqinda haqiqat bo'lsa-da, boshqa *KB* modeli mavjud, unda *part_of(r023,cs_building)* yolg'ondir. Xususan, 15.5-misolda ta'riflangan talqin-ga o'xshash, lekin unda

$\pi(\text{part_of})(\varphi(r023), \varphi(cs_building)) = \text{false}$

bo'lgan talqin *KB* modeli bo'lib, unda *part_of(r023,cs_building)* yolg'ondir.

KB |= *in(kim,cs_building)*. Agar *KB* formulasi *I* talqinida to'g'ri bo'lsa, unda *in(kim,cs_building)I* da haqiqat bo'lishi kerak, aks holda, *KB*ning uchinchi formulasi-ning bir nusxasi *I* da yolg'on bo'ladi – *I* ning *KB* modeli ekanligiga ziddiyatdir.

Semantikaga inson nuqtai nazari

Semantikaning rasmiy ta'rif-i bizga nima uchun semantika qiziqarli ekanligini yoki u qanday qilib intellektual tizimlar qurish uchun asos bo'lishi mumkinligini anglatmaydi. Proposition mantiq dasturlari uchun semantikadan foydalanish metodologiyasi predikativ mantiqqa kengaytirilishi mumkin:

1-qadam. Taqdim etish uchun vazifa sohasi yoki dunyoni tanlang. Bu haqiqiy dunyoning ba'zi jihatlarini bo'lishi mumkin, masalan, universitetdagi kurslar strukturalari va talabalar yoki ma'lum bir vaqtda laboratoriya muhiti, tasavvur qilingan dunyo, masalan, «Alice in Wonderland» («Alisa mo'jizalar mamlakatida») dunyosi yoki bir tugma ishlamas elektr muhiti holati, mavhum dunyo, masalan, pul, raqamlar va to'plamlar dunyosi. Ushbu dunyo doirasida, domen *D* bu siz istagan barcha shaxslar yoki narsalar to'plami bo'lsin. Shuningdek, qaysi munosabatlarni ifodalashni tanlang.

2-qadam. Tilning konstantalarini siz nomlamoqchi bo'lgan dunyodagi shaxslar bilan bog'lang. Har bir *D* elementi uchun siz nom bilan murojaat qilmoqchi bo'lsangiz, tilda bir konstantani belgilang. Masalan, siz «kim» nomini ma'lum bir professorni va «cs322» nomini ma'lum bir kirish SI kursini belgilash uchun tanlashingiz mumkin. Har bir nom dunyodagi mos shaxsni belgilaydi.

3-qadam. Siz vakil qilmoqchi bo'lgan har bir munosabat uchun tilning predikativ belgisi bilan bog'lang. Har bir n -ary predikativ belgisi D^n dan $\{true, false\}$ ga funksiyani belgilaydi, bu munosabat to'g'ri bo'lgan D^n ning pastki to'plamini belgilaydi. Masalan, ikki argumentli «teaches» predikativ belgisi (o'qituvchi va kurs) birinchi argument tomonidan belgilangan shaxs ikkinchi argument tomonidan belgilangan kursni o'qitganida to'g'ri bo'lgan ikkilik munosabatga mos kelishi mumkin. Belgilarni ularning ma'nolari bilan bog'lash **mo'ljallangan talqinni** shakllantiradi. Bu belgilar nimani anglatishini belgilaydi.

4-qadam. Mo'ljallangan talqinda to'g'ri bo'lgan formulalarni yozing. Bu **domenni aksiomatizatsiyalash** deb ataladi, bu yerda berilgan formulalar domenning **aksiomalari** yoki **bilimlar bazasi** hisoblanadi. Masalan, *kim* belgisi bilan belgilangan shaxs *cs322* belgisi bilan belgilangan kursni o'qitgan bo'lsa, siz mo'ljallangan talqinda to'g'ri bo'lgan *teaches (kim, cs322)* formulasini tasdiqlashingiz mumkin. Hamma haqiqatni yozib qo'yish shart emas; boshqa aksiomalardan kelib chiqadigan narsalarni bayon qilish shart emas.

5-qadam. Mo'ljallangan talqin haqida savollar bering. Tizim belgilar uchun belgilangan ma'no asosida siz tushunishingiz mumkin bo'lgan javoblarni beradi. Ushbu metodologiyaga rioya qilgan holda, bilimlar bazasini loyihalovchi aslida kompyuterga hech narsa anglatmaydi to'rtinchi bosqichgacha. Birinchi uch bosqich dizaynning boshida amalga oshiriladi. Albatta, dizayner o'z bilim bazasini boshqalarga tushunarli qilish uchun, har bir belgi qanday ma'noga ega ekanligini eslab qolish va formulalarning to'g'riligini tekshirish uchun denotatsiyalarni hujjatlashtirishi kerak.

Dunyoning o'zi shaxslar qanday bo'lishi kerakligini belgilamaydi.

15.9-misol. Domenning bir tushunchasida, *pink* bitta argumentli predikativ belgi bo'lishi mumkin, bu argument tomonidan belgilangan shaxs pushti rang bo'lganda to'g'ri. Boshqa bir tushunchada, *pink* pushti rang bo'lgan shaxs bo'lishi mumkin va bu ikkilik predikativ *color* uchun ikkinchi argument sifatida ishlatilishi mumkin, bu birinchi argument tomonidan belgilangan shaxs ikkinchi argument tomonidan belgilangan rangga ega ekanligini bildiradi. Shu bilan bir qatorda, kimdir dunyoni har xil *red* ranglar ajratib bo'lmaydigan darajada tasvirlamoqchi bo'lishi mumkin, shuning uchun *pink* rangi kiritilmasligi mumkin. Boshqa birov dunyoni yanada batafsil tasvirlashi va *pink* juda umumiy atama deb hisoblab, *coral* va *salmon* terminlaridan foydalanishi mumkin. Domenning shaxslari haqiqiy jismoniy narsalar bo'lganda, denotatsiyani jismonan ko'rsatmasdan belgilash odatda qiyin bo'ladi. Shaxs mavhum shaxs bo'lganda – masalan, universitet kursi yoki muhabbat tushunchasi – denotatsiyani yozish deyarli mumkin emas. Biroq, bu tizimni bunday tushunchalarni ifodalash va ularning ustida mantiq yuritishdan to'xtatmaydi.

15.4. Datalog: relyatsion qoidalar tili

Datalog (Ma'lumotlar jurnali) mantiqiy aniqliklar tilini kengaytirib, shaxslar va munosabatlarni o'z ichiga oladi. Bu predikativ mantiqning cheklangan shakli sifatida ko'rilishi mumkin, bu yerda:

- Yagona ruxsat etilgan formulalar **aniq jumlarlar** bo'lib, shakli $h \leftarrow a_1 \wedge \dots \wedge a_m$ $h \leftarrow a_1 \wedge \dots \wedge a_m$ bo'lgan, bu yerda h bosh va $a_1 \wedge \dots \wedge a_m$ tanadir. Agar $m > 0$ bo'lsa, aniqlik **qoida** deb ataladi. Agar $m = 0$ bo'lsa, « $\leftarrow$ » e'tiborga olinmaydi va aniqlik **fakt** deb ataladi.

- Hech qanday aniq miqdor yo'q; barcha o'zgaruvchilar jumladan tashqarida universal miqdor sifatida qabul qilinadi.

$h(X) \leftarrow b(X, Y)$ aniqligi $\forall X \forall Y (h(X) \leftarrow b(X, Y))$ degan ma'noni anglatadi, bu $\forall X (h(X) \leftarrow \exists Y b(X, Y))$ ga tengdir. Datalog qiziqarli, chunki u aloqalarni aniqlash va so'rash uchun ma'lumotlar bazasi tilidir.

15.10-misol. Munosabatlar ko'pincha jadval shaklida yoziladi:

Course	Section	Time	Room
cs111	7	830	dp101
cs422	2	1030	cc208
cs502	1	1230	dp202

Bu bilimlar bazasi faktlarini o'z ichiga olgan *scheduled* (*Course*, *Section*, *Time*, *Room*) munosabati yordamida ifodalanishi mumkin:

scheduled(cs111, 7, 830, dp101).

scheduled(cs422, 2, 1030, cc208).

scheduled(cs502, 1, 1230, dp202).

Tasavvur qiling, *enrolled* (*StudentNum*, *Course*, *Section*) munosabati ham faktlar bilan aniqlangan. Munosabat *busy* (*Student Num*, *Time*), bu munosabatlarning birlashmasidan iborat, talaba soni va vaqtiga proyeksiyalangan, qoida yordamida ifodalanishi mumkin:

busy(*StudentNum*, *Time*) $\leftarrow$

enrolled(*StudentNum*, *Course*, *Section*) $\wedge$

scheduled(*Course*, *Section*, *Time*, *Room*).

Talaba bir vaqt ichida band bo'ladi, agar u belgilangan vaqtda rejalashtirilgan kurs bo'limiga yozilgan bo'lsa.

15.11-misol. 5.2-rasmdagi elektr muhitini takliflar yordamida ifodalash mumkin. Shaxslar va munosabatlardan foydalanish ifodalashni intuitivroq qilishi mumkin, chunki tugmalar qanday ishlashi haqida umumiy bilimlar ma'lum bir uy haqidagi bilimlardan aniq ajralib turadi.

Bu domenni ifodalash uchun birinchi qadam, domen ichida qanday shaxslar mavjudligini aniqlashdir. Quyida har bir tugma, har bir chiroq va har bir rozetka alohida deb faraz qilamiz. Har bir tugma va chiroq o'rtasidagi sim ham individuallidir. Kimdir simlar juftliklari bog'langan va elektr oqimi Kirchhoff qonunlariga rioya qilishi kerak deb da'vo qilishi mumkin. Boshqalar, hatto elektronlar oqimi ham modellastirilishi kerak deb qaror qilishlari mumkin. Ammo, foydali bo'lgan abstraksiya darajasi, topshiriqqa mos bo'lgan darajadir. Uyning aholisi simlarning individual tolalari orasidagi ulanishlar yoki hatto kuchlanish haqida bilmasligi mumkin. Shuning uchun, elektr energiyasi uyning tashqarisidan chiroqlarga o'tadigan modelini qabul qilaylik. Ushbu model chiroq yoqilishi kerakmi yoki yo'qmi aniqlash uchun mos, ammo boshqa vazifalar uchun mos bo'lmasiligi mumkin.

Keyingi qadam, siz nomlamoqchi bo'lgan har bir shaxsga nom berishdir. Bu 5.2-rasmda komponent yonida nom yozish orqali amalga oshiriladi. Masalan, shaxs w_0 chiroq l va tugma s_2 o'rtasidagi simdir.

Keyin, qaysi munosabatlarni ifodalashni tanlang. Quyidagi predikatlarni va ularning maqsadli talqinlarini qabul qiling:

- $light(L)$ agar L chiroq bo'lsa, to'g'ri bo'ladi.
- $lit(L)$ agar chiroq L yoqilgan va yorug'lik chiqarayotgan bo'lsa, to'g'ri bo'ladi.
- $live(W)$ agar W ga elektr kelayotgan bo'lsa, to'g'ri bo'ladi.
- $up(S)$ agar tugma S yuqoriga ko'tarilgan bo'lsa, to'g'ri bo'ladi.
- $down(S)$ agar tugma S pastga tushirilgan bo'lsa, to'g'ri bo'ladi.
- $ok(E)$ agar E nosoz bo'lmasa, to'g'ri bo'ladi; E yo o'chirgich yoki chiroq bo'lishi mumkin.
- $connected_to(X,Y)$ agar komponent X , Y ga ulanib, tok Y dan X ga o'tishi mumkin bo'lsa, to'g'ri bo'ladi.

Bu bosqichda kompyuterga hech narsa aytilmagan. U predikatlar nima ekanligini ham bilmaydi, ularning ma'nolarini ham bilmaydi. Qaysi shaxslar mavjudligi yoki ularning nomlari haqida hech narsa bilmaydi.

Aniq bir uy haqida hech narsa ma'lum bo'lmasdan oldin, tizimga umumiy qoidalar aytilishi mumkin:

$$lit(L) \leftarrow light(L) \wedge live(L) \wedge ok(L).$$

Bu chiroq yoqilgan bo'lishini, agar u nosoz bo'lmasa, elektr energiyasiga ega ekanligini bildiradi.

Rekursiv qoidalar nimani nimadan ulanganligini aytishga imkon beradi:

$$live(X) \leftarrow connected_to(X,Y) \wedge live(Y).$$

$$live(outside).$$

Bu uyga elektr kirayotganini va agar Y ga elektr kirayotgan va X,Y ga ulangan bo'lsa, X ga ham elektr kirayotganini bildiradi.

Aynan uyning konfiguratsiyasi va komponentlari haqida quyidagi faktlar tizimga aytilishi mumkin:

$light(l_1).light(l_2).down(s_1).$
 $up(s_2).ok(cb_1).$
 $connected_to(w_0, w_1) \leftarrow up(s_2).$
 $connected_to(w_0, w_2) \leftarrow down(s_2).$
 $connected_to(w_1, w_3) \leftarrow up(s_1).$
 $Connected_to(w_3, outside) \leftarrow ok(cb_1).$

Bu qoidalar va atomik iboralar kompyuterga aytiladigan hamma narsadir. U bu belgilar ma'nosini bilmaydi. Ammo, endi u bu uy haqida so'rovlar javobini bera oladi.

15.4.1. O'zgaruvchilar bilan so'rovlar

So'rovlar, ma'lumotlar bazasi asosida qandaydir bayonot mantiqiy natija ekanligini so'rash uchun ishlatiladi. Taklif so'rovlar bilan, foydalanuvchi faqat «ha-yo'q» so'rovlarini so'rashi mumkin. O'zgaruvchilar bilan **so'rovlar** foydalanuvchilarga so'rovni haqiqiy qiladigan shaxslarni so'rash imkonini beradi.

So'rovning **misoli**, so'rovdagi o'zgaruvchilar uchun terminlar almashtirilishi bilan olinadi. So'rovdagi har bir turli o'zgaruvchining har bir ko'rinishi bir xil termin bilan almashtirilishi kerak; agar formulada X o'zgaruvchisi bo'lsa, X ning har bir ko'rinishi bir xil termin bilan almashtiriladi.

Erkin o'zgaruvchiga ega bo'lgan so'rov mavjud bo'lganda, **javob** bilimlar bazasining mantiqiy natijasi bo'lgan so'rovning nusxasi yoki «yo'q»dir, ya'ni so'rovning hech bir nusxasi bilimlar bazasidan mantiqiy ravishda kelib chiqmaydi. So'rovning misollari o'zgaruvchilar uchun qiymatlar berish orqali belgilanadi. So'rovning qaysi misollari ma'lumotlar bazasidan kelib chiqishini aniqlash, **javob chiqarish** deb ataladi.

«Yo'q» javobi, maqsadli talqinda so'rov noto'g'ri ekanligini anglatmaydi; bu faqat so'rovning hech qanday misoli mantiqiy natija emasligini bildiradi.

15.12-misol. Janubiy Amerikadagi ba'zi tumanlar haqida faktlarni ko'rib chiqing (15.2-rasm). Kompyuter geografiya yoki Janubiy Amerika haqida hech narsa bilmaydi. U faqat berilgan iboralarni biladi, ammo u mantiqiy natijalarni hisoblab chiqishi mumkin. Mamlakatlar va tillarni belgilaydigan konstantalar kichik harf bilan boshlanadi, chunki bu ishlatilgan til konvensiyasi; ingliz tilida esa aksincha, atqli otlar katta harf bilan boshlanadi.

tili(chili, ispan)

% country (C) agar C mamlakat bo'lsa, to'g'ri

country(argentina).country(brazil).

country(chile).

country(paraguay).

country(peru).

% *area* (C, A) to'g'ri, agar C mamlakat bo'lsa va A uning kvadrat kilometrdagi maydoni bo'lsa.

area(argentina, 2780400).

area(brazil, 8515767).

area(chile, 756102).

area(paraguay, 406756).

area(peru, 1285216).

% *language* (C, L) to'g'ri, agar L , C mamlakatining asosiy tili bo'lsa.

language(argentina, spanish).

language(brazil, portuguese).

language(chile, spanish).

language(paraguay, spanish).

language(peru, spanish).

% *borders* (C_1, C_2) to'g'ri, agar C_1 va C_2 mamlakatlari bir-biriga chegara bo'lsa va C_1 alifbo bo'yicha C_2 dan oldinda bo'lsa,

*borders*₀(chile, peru).

*borders*₀(argentina, chile).

*borders*₀(brazil, peru).

*borders*₀(brazil, argentina).

*borders*₀(brazil, paraguay).

*borders*₀(argentina, paraguay).

% *borders* agar C_1 mamlakati C_2 mamlakati bilan chegaradosh bo'lsa, to'g'ri

borders(X, Y) $\leftarrow$ *borders*₀(X, Y).

borders(X, Y) $\leftarrow$ *borders*₀(Y, X).

15.2-rasm. Janubiy Amerika haqidagi ba'zi faktlar

Foydalanuvchi quyidagi so'rovni so'rashi mumkin:

language (chile, spanish) va javob *ha*.

Foydalanuvchi quyidagi so'rovni berishi mumkin:

language (venezuela, spanish) va javob *yo'q*. Bu uning noto'g'ri ekanligini emas, balki ma'lumotlar bazasida *venezuelaning* asosiy tilini aniqlash uchun yetarli ma'lumot yo'qligini anglatadi. So'rov:

language(C , spanish).

to'rt javobga ega. $X = \text{chile}$ javobi *language*(chile, spanish) jumlarining mantiqiy natijasi ekanligini anglatadi. *borders* munosabati ikki mamlakat chegaradosh bo'lganda to'g'ri bo'ladi. Bu munosabat simmetrik; agar *borders*(X, Y) bo'lsa va faqat agar *borders*(Y, X) bo'lsa. Buni ifodalash uchun, 15.2-rasmda juftlardan biri (tasodifan, birinchi argument sifatida alifbo bo'yicha birinchi kelgan) *borders*₀ yordamida ifodalangan va keyin chegaradan hosil qilingan. Bu, agar *borders* to'g'ridan to'g'ri ifodalangan bo'lsa, ikki qoidaning qo'shimcha xarajatlari bilan ikki barobar ko'p faktlar ishlatilgan bo'lar edi.

So'rov:

ask *borders*(paraguay, X).

javoblarga ega emas. So'rov:

ask *borders*(paraguay, X).

ikki javobga ega: $X=argentina$ and $X=brazil$.

So'rov:

ask *borders*(X, Y).

12 javobga ega.

Chiliga chegaradosh bo'lgan va maydoni ikki million kvadrat kilometrdan ortiq bo'lgan mamlakatni so'rash uchun quyidagi so'rovni berish mumkin:

ask *borders*(chile, Y) $\wedge$ *area*(Y, A) $\wedge$ A > 2000000.

Bu yerda >bu argumentlari sonlar bo'lgan va chap argument o'ng argumentdan katta bo'lsa to'g'ri bo'lgan predikativ. Matematikada an'anaviy bo'lganidek, bu predikat infiks belgisi yordamida yoziladi.

15.13-misol. 15.3-rasmda Janubiy Amerikadagi ba'zi mamlakatlar va ularning poytaxtlari haqida ko'proq faktlar berilgan. Bu mamlakat nomini mamlakatning o'zidan qanday farqlashni ko'rsatadi. *chile* konstantasi mamlakatni va «Chile» konstantasi mamlakat nomi bo'lgan satrni anglatadi. Bu juda farqli narsalardir; masalan, «Chile» besh belgidan iborat, ammo *chile* (shimoliydan janubga) 4270 kilometr uzunlikda.

Chilining poytaxt shahrining nomini so'rash uchun quyidagi so'rovni berishingiz mumkin:

ask *capital*(chile, C) $\wedge$ *name*(C, N).

15.5. Tasdiqlash va almashtirish

5.3.2-mavzudagi pastdan yuqoriga va yuqoridan pastga qarab taklifni isbotlash protseduralari Datalog uchun kengaytirilishi mumkin. O'zgaruvchilarga nisbatan qo'llaniladigan isbotlash tartibi jumladagi erkin o'zgaruvchi jumlaning barcha misollari haqiqat ekanligini anglatishini hisobga olishi kerak. Isbot, bitta isbotda bir xil jumlaning turli misollaridan foydalanishi mumkin.

15.5.1. Isbotlash va almashtirish

Jumlaning **misoli**, jumladagi o'zgaruvchilar uchun terminlarni bir xil ravishda almashtirish orqali olinadi. Ma'lum bir o'zgaruvchining barcha ko'rinishlari bir xil termin bilan almashtiriladi.

Qaysi qiymat har bir o'zgaruvchiga tayinlanishini ko'rsatuvchi spetsifikatsiya **almashtirish** deb ataladi. **Almashtirish** $\{V_1/t_1, \dots, V_n/t_n\}$ shaklida bo'lib, har bir V_i turli o'zgaruvchilar va har bir t_i terminlardan iborat. V_i/t_i elementi V_i o'zgaruvchi-

si uchun **bog'lovchi** hisoblanadi. Agar biron-bir t_j da V_i ko'rinmasa, almashtirish **normal shaklda** bo'ladi.

15.14-misol. Masalan, $\{X/Y, Z/chile\}$ normal shakldagi almashtirish bo'lib, X ni Y ga va Z ni $chile$ ga bog'laydi. $\{X/Y, Z/X\}$ almashtirish normal shaklda emas, chunki X har ikkala tayinlashning chap va o'ng tomonida ham paydo bo'ladi.

% capital (A, B) to'g'ri. Agar B , Aning poytaxti bo'lsa

capital(argentina, beunosaires).

capital(chile, santiago).

capital(peru, lima).

capital(brazil, brasilia).

capital(paraguay, asuncion).

agar N qatori alohida E nomi bo'lsa, % name (E, N) to'g'ri qiymatiga ega

name(beunosaires, «BuenosAires»).

name(santiago, «Santiago»).

name(lima, «Lima»).

name(brasilia, «Brasilia»).

name(asuncion, «Asunción»).

name(argentina, «Argentina»).

name(chile, «Chile»).

name(peru, «Peru»).

name(brazil, «Brazil»).

name(paraguay, «Paraguay»).

15.3-rasm. Janubiy Amerika poytaxtlari haqida ba'zi faktlar

e ifodasiga $\sigma = \{V_1/t_1, \dots, V_n/t_n\}$ almashtirish **qo'llanilishi**, yozma $e\sigma$, bu ibora asl e ifodasi bilan bir xil bo'lib, e dagi V_{ining} har bir ko'rinishi tegishli t_i bilan almashtiriladi. $e\sigma$ ifodasi **enings misoli** deyiladi. Agar hech qanday o'zgaruvchi bo'lmasa, u **enings asosiy misoli** deyiladi.

15.15-misol. Ba'zi o'rinbosarlarning qo'llanilishi:

$borders(peru, X)\{X/chile\} = borders(peru, chile).$

$borders(Y, chile)\{Y/peru\} = borders(peru, chile).$

$borders(peru, X)\{Y/peru, Z/X\} = borders(peru, X).$

$p(X, X, Y, Y, Z)\{X/Z, Y/brazil\} = p(Z, Z, brazil, brazil, Z).$

Almashtirishlar jumalarga, atomlarga nisbatan qo'llanilishi mumkin. Masalan, $\{x/Y, Z/peru\}$ almashtirishni jumlada qo'llash natijasi:

$p(X, Y) \leftarrow q(peru, Z, X, Y, Z)$

bu jumla

$p(Y, Y) \leftarrow q(peru, peru, Y, Y, peru).$

σ almashtirish e_1 va e_2 ifodalarni **birlashtiruvchisi** hisoblanadi, agar $e_1\sigma$ va $e_2\sigma$ bir xil bo'lsa; ya'ni, e_1 va e_2 ifodalari **birlashtirilgan** deyiladi. Ya'ni ikki ifodani birlashtiruvchi, har bir ifodaga qo'llanganda bir xil ifodani hosil qiladigan almashtirishdir.

15.16-misol. $\{X/\text{chile}, Y/\text{peru}\}$ almashtirish $borders(\text{peru}, X)$ va $borders(Y, \text{chile})$ ifodalari uchun birlashtiruvchi hisoblanadi. Bu almashtirishni har ikkisi qo'llash $borders(\text{peru}, \text{chile})$ ifodasini hosil qiladi. $\{X/a, Y/b\}$ $t(a, Y, c)$ va $t(X, b, c)$ ifodalarning birlashtiruvchi sihisoblanadi, chunki

$$t(a, Y, c)\{X/a, Y/b\} = t(X, b, c)\{X/a, Y/b\} = t(a, b, c).$$

Ifodalar ko'plab birlashtiruvchilarga ega bo'lishi mumkin.

15.17-misol. $p(X, Y)$ va $p(Z, Z)$ atomlari ko'plab birlashtiruvchilarga ega, jumladan

$$\{X/b, Y/b, Z/b\}, \{X/c, Y/c, Z/c\}, \{X/Z, Y/Z\} \text{ va } \{Y/X, Z/X\}.$$

Oxirgi ikkita birlashtiruvchi birinchi ikkitasiga qaraganda umumiyroqdir, chunki birinchi ikkitasida X, Z bilan bir xil, Y esa Z bilan bir xil, ammo ular bu qiymatlar qanday bo'lishiga ko'proq bog'liq.

σ almashtirish, $e1$ va $e2$ ifodalarning **eng umumiy birlashtiruvchisi (MGU)** hisoblanadi, agar:

σ ikki ifodani birlashtiruvchisi bo'lsa, va

agar almashtirish σ ham $e1$ va $e2$ ni birlashtiruvchisi bo'lsa, unda har qanday e ifodasi uchun e ning misoli bo'lishi kerak.

15.18-misol. $\{X/Z, Y/Z\}$ va $\{Z/X, Y/X\}$ ikkisi ham $p(X, Y)$ va $p(Z, Z)$ uchun eng umumiy birlashtiruvchilar (MGU) hisoblanadi. $\{X/a, Y/a, Z/a\}$ ularni birlashtiruvchi hisoblanadi, ammo eng umumiy birlashtiruvchi emas. Natijadagi qo'llanishlar quyidagicha bo'ladi:

$$p(X, Y)\{X/Z, Y/Z\} = p(Z, Z)\{X/Z, Y/Z\} = p(Z, Z)$$

$$p(X, Y)\{Z/X, Y/X\} = p(Z, Z)\{Z/X, Y/X\} = p(X, X)$$

$$p(X, Y)\{X/a, Y/a, Z/a\} = p(Z, Z)\{X/a, Y/a, Z/a\} = p(a, a).$$

Shuni unutmangki, $p(a, a)$, $p(Z, Z)$ va $p(X, X)$ ning misoli hisoblanadi, ammo ular $p(a, a)$ ning misollari emas. $p(Z, Z)$ va $p(X, X)$ bir-birlarining misollari hisoblanadi. MGUning ta'rifi «all expressions e »ga tegishli bo'lib, $p(X, Y)$ va $p(Z, Z)$ uchun $\{X/Z, Y/Z, W/a\}$ kabi birlashtiruvchi eng umumiy birlashtiruvchi bo'lishiga yo'l qo'ymaydi, chunki bu boshqa ifodalarga, masalan $r(W)$ ga ta'sir qiladi.

$\setminus e1$ ifoda, agar u faqat o'zgaruvchilar nomlarida farqlansa, $e2$ ning **qayta nomlanishi** hisoblanadi. Bu holda, ular bir-birlarining misollari hisoblanadi.

Agar ikkita ifoda birlashtiruvchiga ega bo'lsa, ularda kamida bitta MGU mavjud. MGU ni ifodalarga qo'llash natijasidagi ifodalar bir-birlarining qayta nomlanishlari hisoblanadi. Ya'ni agar σ va σ' ikkisi ham $e1$ va $e2$ ifodalarning MGU lari bo'lsa, unda $e1\sigma$, $e1\sigma'$ ning qayta nomlanishi hisoblanadi.

15.5.2. Ma'lumotlarni yozib olishning pastdan yuqoriga tartibi

Taklifni pastdan yuqoriga isbotlash tartibi jumalarning asosiy misollaridan foydalangan holda Datalogga kengaytirilishi mumkin. Gapning asosiy misoli jumladagi o'zgaruvchilar uchun konstantalarni bir xilda almashtirish orqali olinadi.

si uchun **bog'lovchi** hisoblanadi. Agar biron-bir tj da Vi ko'rinmasa, almashtirish **normal shaklda** bo'ladi.

15.14-misol. Masalan, $\{X/Y, Z/chile\}$ normal shakldagi almashtirish bo'lib, X ni Y ga va Z ni $chile$ ga bog'laydi. $\{X/Y, Z/X\}$ almashtirish normal shaklda emas, chunki X har ikkala tayinlashning chap va o'ng tomonida ham paydo bo'ladi.

% capital (A, B) to'g'ri, Agar B , Aning poytaxti bo'lsa

capital(argentina, beunosaires).

capital(chile, santiago).

capital(peru, lima).

capital(brazil, brasilia).

capital(paraguay, asuncion).

agar N qatori alohida E nomi bo'lsa, % name (E, N) to'g'ri qiymatiga ega

name(beunosaires, «BuenosAires»).

name(santiago, «Santiago»).

name(lima, «Lima»).

name(brasilia, «Brasilia»).

name(asuncion, «Asunción»).

name(argentina, «Argentina»).

name(chile, «Chile»).

name(peru, «Peru»).

name(brazil, «Brazil»).

name(paraguay, «Paraguay»).

15.3-rasm. Janubiy Amerika poytaxtlari haqida ba'zi faktlar

e ifodasiga $\sigma = \{V1/t1, \dots, Vn/tn\}$ almashtirish **qo'llanilishi**, yozma $e\sigma$, bu ibora asl e ifodasi bilan bir xil bo'lib, e dagi $Vining$ har bir ko'rinishi tegishli ti bilan almashtiriladi. $e\sigma$ ifodasi ening **misoli** deyiladi. Agar hech qanday o'zgaruvchi bo'lmasa, u ening **asosiy misoli** deyiladi.

15.15-misol. Ba'zi o'rinbosarlarning qo'llanilishi:

$borders(peru, X)\{X/chile\} = borders(peru, chile).$

$borders(Y, chile)\{Y/peru\} = borders(peru, chile).$

$borders(peru, X)\{Y/peru, Z/X\} = borders(peru, X).$

$p(X, X, Y, Y, Z)\{X/Z, Y/brazil\} = p(Z, Z, brazil, brazil, Z).$

Almashtirishlar jumalarga, atomlarga nisbatan qo'llanilishi mumkin. Masalan, $\{x/Y, Z/peru\}$ almashtirishni jumlada qo'llash natijasi:

$p(X, Y) \leftarrow q(peru, Z, X, Y, Z)$

bu jumla

$p(Y, Y) \leftarrow q(peru, peru, Y, Y, peru).$

σ almashtirish e_1 va e_2 ifodalarni **birlashtiruvchisi** hisoblanadi, agar $e_1\sigma$ va $e_2\sigma$ bir xil bo'lsa; ya'ni, e_1 va e_2 ifodalari **birlashtirilgan** deyiladi. Ya'ni ikki ifodani birlashtiruvchi, har bir ifodaga qo'llanganda bir xil ifodani hosil qiladigan almashtirishdir.

15.16-misol. $\{X/\text{chile}, Y/\text{peru}\}$ almashtirish $\text{borders}(\text{peru}, X)$ va $\text{borders}(Y, \text{chile})$ ifodalari uchun birlashtiruvchi hisoblanadi. Bu almashtirishni har ikkisi qo'llash $\text{borders}(\text{peru}, \text{chile})$ ifodasini hosil qiladi. $\{X/a, Y/b\}$ $t(a, Y, c)$ va $t(X, b, c)$ ifodalarning birlashtiruvchi sihisoblanadi, chunki

$$t(a, Y, c)\{X/a, Y/b\} = t(X, b, c)\{X/a, Y/b\} = t(a, b, c).$$

Ifodalar ko'plab birlashtiruvchilarga ega bo'lishi mumkin.

15.17-misol. $p(X, Y)$ va $p(Z, Z)$ atomlari ko'plab birlashtiruvchilarga ega, jumladan

$$\{X/b, Y/b, Z/b\}, \{X/c, Y/c, Z/c\}, \{X/Z, Y/Z\} \text{ va } \{Y/X, Z/X\}.$$

Oxirgi ikkita birlashtiruvchi birinchi ikkitasiga qaraganda umumiyroqdir, chunki birinchi ikkitasida X, Z bilan bir xil, Y esa Z bilan bir xil, ammo ular bu qiymatlar qanday bo'lishiga ko'proq bog'liq.

σ almashtirish, $e1$ va $e2$ ifodalarning **eng umumiy birlashtiruvchisi (MGU)** hisoblanadi, agar:

σ ikki ifodani birlashtiruvchisi bo'lsa, va

agar almashtirish σ ham $e1$ va $e2$ ni birlashtiruvchisi bo'lsa, unda har qanday e ifodasi uchun e ning misoli bo'lishi kerak.

15.18-misol. $\{X/Z, Y/Z\}$ va $\{Z/X, Y/X\}$ ikkisi ham $p(X, Y)$ va $p(Z, Z)$ uchun eng umumiy birlashtiruvchilar (MGU) hisoblanadi. $\{X/a, Y/a, Z/a\}$ ularni birlashtiruvchi hisoblanadi, ammo eng umumiy birlashtiruvchi emas. Natijadagi qo'llanishlar quyidagicha bo'ladi:

$$p(X, Y)\{X/Z, Y/Z\} = p(Z, Z)\{X/Z, Y/Z\} = p(Z, Z)$$

$$p(X, Y)\{Z/X, Y/X\} = p(Z, Z)\{Z/X, Y/X\} = p(X, X)$$

$$p(X, Y)\{X/a, Y/a, Z/a\} = p(Z, Z)\{X/a, Y/a, Z/a\} = p(a, a).$$

Shuni unutmangki, $p(a, a)$, $p(Z, Z)$ va $p(X, X)$ ning misoli hisoblanadi, ammo ular $p(a, a)$ ning misollari emas. $p(Z, Z)$ va $p(X, X)$ bir-birlarining misollari hisoblanadi. MGUning ta'rifi «all expressions e »ga tegishli bo'lib, $p(X, Y)$ va $p(Z, Z)$ uchun $\{X/Z, Y/Z, W/a\}$ kabi birlashtiruvchi eng umumiy birlashtiruvchi bo'lishiga yo'l qo'ymaydi, chunki bu boshqa ifodalarga, masalan $r(W)$ ga ta'sir qiladi.

$e1$ ifoda, agar u faqat o'zgaruvchilar nomlarida farqlansa, $e2$ ning **qayta nomlanishi** hisoblanadi. Bu holda, ular bir-birlarining misollari hisoblanadi.

Agar ikkita ifoda birlashtiruvchiga ega bo'lsa, ularda kamida bitta MGU mavjud. MGU ni ifodalarga qo'llash natijasidagi ifodalar bir-birlarining qayta nomlanishlari hisoblanadi. Ya'ni agar σ va σ' ikkisi ham $e1$ va $e2$ ifodalarning MGU lari bo'lsa, unda $e1\sigma$, $e1\sigma'$ ning qayta nomlanishi hisoblanadi.

15.5.2. Ma'lumotlarni yozib olishning pastdan yuqoriga tartibi

Taklifni pastdan yuqoriga isbotlash tartibi jumalarning asosiy misollaridan foydalangan holda Datalogga kengaytirilishi mumkin. Gapning asosiy misoli jumladagi o'zgaruvchilar uchun konstantalarni bir xilda almashtirish orqali olinadi.

Kerakli konstantalar bilimlar bazasida yoki so'rovda paydo bo'lganlardir. Agar bilimlar bazasida yoki so'rovda doimiylar bo'lmasa, ularni yaratish kerak bo'ladi.

15.19-misol. Bilim bazasi quyidagicha bo'lsin:

$$q(a). \quad q(b). \quad r(a).$$

$$s(W) \leftarrow r(W). p(X, Y) \leftarrow q(X) \wedge s(Y).$$

Barcha asosiy misollar to'plami:

$$q(a). \quad q(b). \quad r(a).$$

$$s(a) \leftarrow r(a). \quad s(b) \leftarrow r(b). \quad p(a, a) \leftarrow q(a) \wedge s(a).$$

$$p(a, b) \leftarrow q(a) \wedge s(b). p(b, a) \leftarrow q(b) \wedge s(a). p(b, b) \leftarrow q(b) \wedge s(b).$$

5.3.2-mavzuning taklifga asoslangan pastdan yuqoriga qarab isbotlash tartibi $q(a)$, $q(b)$, $r(a)$, $s(a)$, $p(a, a)$ va $p(b, a)$ kabi asoslangan misollarni hosil qilish uchun qo'llanilishi mumkin, ular mantiqiy natijalardir.

15.20-misol. Bilim bazasi quyidagicha bo'lsin:

$$p(X, Y).$$

$$g \leftarrow p(W, W).$$

«ask g» so'rovi uchun pastdan yuqoriga qarab isbotlash tartibi yangi doimiy belgini, masalan c ni kiritishi kerak. Barcha asoslangan misollar to'plami keyin quyidagicha bo'ladi:

$$p(c, c).$$

$$g \leftarrow p(c, c).$$

Taklifga asoslangan pastdan yuqoriga qarab isbotlash tartibi $p(c, c)$ va g ni hosil qiladi. Agar so'rov «ask $p(b, d)$ » bo'lsa, asoslangan misollar to'plami b va d doimiylarini aks ettirishi uchun o'zgaradi.

Bilim bazasini asoslashga qo'llaniladigan pastdan yuqoriga qarab isbotlash tartibi to'g'ri, chunki har bir qoida misolining har bir modeli uchun har doim haqiqatdir. Ushbu tartib asosan o'zgaruvchilarsiz holatga o'xshash, ammo u gaplarning asoslangan misollarini ishlatadi, ularning hammasi haqiqatdir, chunki gapdagi o'zgaruvchilar umumiy miqdordlar bilan aniqlangan.

Datalog uchun ushbu pastdan yuqoriga qarab isbotlash tartibi oxir-oqibat to'xtaydi, chunki faqat cheklangan miqdordagi asoslangan atomlar mavjud va har safar sikl orqali bir asoslangan atom natija to'plamiga qo'shiladi.

Ushbu tartib, shuningdek, asoslangan atomlar uchun to'liq hisoblanadi. Ya'ni agar asosiy atom bilimlar bazasining natijasi bo'lsa, u hosil qilinadi. Buni isbotlash uchun, taklifga asoslangan holatda bo'lgani kabi, maxsus umumiy modelni tuzing. Model domenni, konstantalarning nimani anglatishini va nimaning haqiqiyligini aniqlaydi. **Herbrand talqini**, domen ramziy bo'lib, tilning barcha konstantalaridan iborat bo'lgan talqindir. Agar bilim bazasida yoki so'rovda hech qanday konstanta bo'lmasa, yangi konstanta ixtiro qilinadi. Herbrand talqinida har bir konstanta o'zini o'zi anglatadi. Shunday qilib, talqinni aniqlashda, D va φ berilgan dastur uchun belgilangan va faqat predikat belgilarini aniqlaydigan π aniqlanishi kerak.

Pastdan yuqoriga qarab isbotlash tartibi orqali oxir-oqibat hosil qilingan munosabatlarning asosiy misollari haqiqiy atomlar bo'lgan Herbrand talqinini ko'rib chiqing. Ko'rinib turibdiki, Herbrand talqini ushbu qoidalarining namunasidir. O'zgaruvchisiz holatda bo'lgani kabi, bu minimal modeldir, chunki u har qanday modeldagi haqiqiy atomlarning eng kam soniga ega. Agar asosiy atom g uchun $KB \models g$ bo'lsa, u holda g minimal modelda to'g'ri bo'ladi va natijada xulosa chiqariladi.

15.5.3. Birlashtirish

Kompyuter asoslanish orqali propozitiv darajada mulohaza yuritishi shart emas. Buning o'rniga, u bir nechta asosiy misollar o'rniga o'zgaruvchilar nuqtayi nazaridan mulohaza yuritishi mumkin. Ba'zan, u o'zgaruvchilarni boshqa o'zgaruvchilar yoki konstantalar bilan almashtirishi kerak.

Birlashtirish muammosi quyidagicha: ikki atom yoki termin berilgan bo'lsa, ularning birlashtirilishini aniqlash va agar ular birlashtirilsa, ularni birlashtiruvchini qaytarish. Birlashtirish algoritmi ikki atomning **eng umumiy birlashtiruvchisini** (MGU) topadi yoki agar ular birlashtirilmasa, $\perp$ ni qaytaradi.

```

1: procedure Unify( $t_1, t_2$ )
2:   Inputs
3:      $t_1, t_2$ : atoms or terms
4:   Output
5:     most general unifier of  $t_1$  and  $t_2$  if it exists or  $\perp$  otherwise
6:   Local
7:      $E$ : a set of equality statements
8:      $S$ : substitution
9:      $E \leftarrow \{ t_1 = t_2 \}$ 
10:     $S \leftarrow \{ \}$ 
11:    while  $E \neq \{ \}$  do
12:      select and remove  $\alpha = \beta$  from  $E$ 
13:      if  $\beta$  is not identical to  $\alpha$  then
14:        if  $\alpha$  is a variable then
15:          replace  $\alpha$  with  $\beta$  everywhere in  $E$  and  $S$ 
16:           $S \leftarrow \{ \alpha/\beta \} \cup S$ 
17:        else if  $\beta$  is a variable then
18:          replace  $\beta$  with  $\alpha$  everywhere in  $E$  and  $S$ 
19:           $S \leftarrow \{ \beta/\alpha \} \cup S$ 
20:        else if  $\alpha$  is  $p(\alpha_1, \dots, \alpha_n)$  and  $\beta$  is  $p(\beta_1, \dots, \beta_n)$  then
21:           $E \leftarrow E \setminus \{ \alpha = \beta \}$ 
22:           $E \leftarrow E \cup \{ \alpha_1 = \beta_1, \dots, \alpha_n = \beta_n \}$ 
23:        else
24:          return  $\perp$ 
25:    return  $S$ 

```

15.4-rasm. Datalog uchun birlashtirish algoritmi

Birlashtirish algoritmi 15.4-rasmda keltirilgan. E , $t1$ va $t2$ ni birlashtirish bilan bog'liq va ulardan kelib chiqadigan tenglik bayonotlari to'plamidir. S oddiy ko'rinishdagi almashtirishdir; agar α/β almashtirish S da mavjud bo'lsa, α , S yoki E ning boshqa joylarida uchramaydigan o'zgaruvchi bo'lishi kerak. 20-qatorida α va β bir xil predikatga va bir xil argumentlar soniga ega bo'lishi kerak; aks holda birlashtirish muvaffaqiyatsiz bo'ladi.

15.21-misol. $Unify(p(X,Y,Y), p(a,Z,b))$ chaqiruvini ko'rib chiqing. Dastlab $E \{p(X,Y,Y)=p(a,Z,b)\}$ bo'ladi. Birinchi marta while sikli orqali, $E \{X=a, Y=Z, Y=b\}$ ga aylanadi. Keyin $X=a$ tanlanadi. Shunda $S \{X/a\}$ ga aylanadi va $E \{Y=Z, Y=b\}$ ga aylanadi. Keyin $Y=Z$ tanlanadi. Shunda Y, S va E da Z bilan almashtiriladi. $S \{X/a, Y/Z\}$ ga aylanadi va $E \{Z=b\}$ ga aylanadi. Nihoyat, $Z=b$ tanlanadi, Z, b bilan almashtiriladi, $S \{X/a, Y/b, Z/b\}$ ga aylanadi va E bo'sh bo'ladi. $\{X/a, Y/b, Z/b\}$ almashtiruvchisi MGU sifatida qaytariladi.

$p(a,Y,Y)$ ni $p(Z,Z,b)$ bilan birlashtirishni ko'rib chiqing. $E \{p(a,Y,Y)=p(Z,Z,b)\}$ bilan boshlanadi. Keyingi bosqichda $E \{a=Z, Y=Z, Y=b\}$ ga aylanadi. Keyin Z, E da a bilan almashtiriladi va $E \{Y=a, Y=b\}$ ga aylanadi. Keyin $Y E$ da a bilan almashtiriladi va $E \{a=b\}$ ga aylanadi, shunda $\perp$ qaytariladi va bu birlashtiruvchining yo'qligini bildiradi.

15.5.4. O'zgaruvchilar bilan aniq o'lcham

Taklifning aniq bandlari uchun **yuqoridan pastga isbotlash tartibi** hosilalashtirishda qoidalar misollaridan foydalanishga ruxsat berish orqali o'zgaruvchilarni boshqarish uchun kengaytirilishi mumkin.

Umumlashtirilgan javob bandi quyidagi shaklda bo'ladi:

$$yes(t_1, \dots, t_k) \leftarrow a_1 \wedge a_2 \wedge \dots \wedge a_m$$

bu yerda $t_1, \dots, t_k$ atamalar va $a_1, \dots, a_m$ atomlardir. *Yes* so'zi **javobni ajratib** olishga imkon beradi: so'rov o'zgaruvchilarining qaysi misollari bilim bazasining mantiqiy natijasi ekanligini aniqlash.

Dastlab, so'rov q uchun umumlashtirilgan javob bandi quyidagicha bo'ladi:

$$yes(V_1, \dots, V_k) \leftarrow q$$

bu yerda $V_1, \dots, V_k$ so'rov q da paydo bo'ladigan o'zgaruvchilardir. Intuitiv ravishda bu, agar so'rovning tegishli misoli to'g'ri bo'lsa, $yes(V_1, \dots, V_k)$ misoli to'g'ri ekanligini anglatadi.

Isbotlash tartibi joriy umumlashtirilgan javob bandini saqlab turadi. Har bir bosqichda, algoritm umumlashtirilgan javob bandining tanasidagi atomni tanlaydi. Keyin u bilim bazasidagi bosh qismi atom bilan birlashadigan bandni tanlaydi.

SLD o'lchami, bu yerda SLD umumiy javob bandining *chiziqli aniqligini* bildiradi:

$$yes(t_1, \dots, t_k) \leftarrow a_1 \wedge a_2 \wedge \dots \wedge a_m$$

bu yerda a_j va a eng umumiy birlashtiruvchi σ ga ega, javob bandi quyidagicha bo'ladi:

$$a \leftarrow b_1 \wedge \dots \wedge b_p$$

bu yerda tanlangan bandning tanasi javob bandida a_j ni almashtiradi va MGU σ butun javob bandiga qo'llaniladi.

SLD hosil qilish $\gamma_0, \gamma_1, \dots, \gamma_n$ ketma-ketligidir, bunda:

γ_0 asl so'rovga mos keladigan javob bandi. Agar so'rov q bo'lsa, bo'sh o'zgaruvchilar $V_p, \dots, V_k$ bilan, dastlabki umumlashtirilgan javob bandi γ_0 quyidagicha bo'ladi:

$$yes(V_1, \dots, V_k) \leftarrow q.$$

γ_i quyidagicha olinadi: γ_{i-1} tanasidagi a_i atomini tanlash; ma'lumotlar bazasidagi $a_i b_1 \dots b_p$ band nusxasini tanlash; a_i ni $b_1 \wedge \dots \wedge b_p$ tanasi bilan almashtirish va unifikatori butun natijaviy javob bandiga qo'llash.

```

1: non-deterministic procedure Prove_datalog_TD(KB, q)
2:   Inputs
3:     KB: a set of definite clauses
4:     Query q: a set of atoms to prove, with variables  $V_1, \dots, V_k$ 
5:   Output
6:     substitution  $\theta$  if  $KB = q\theta$  and fail otherwise
7:   Local
8:     G is a generalized answer clause
9:   Set G to generalized answer clause  $yes(V_1, \dots, V_k) \leftarrow q$ 
10:  while G is not an answer do
11:    Suppose G is  $yes(t_1, \dots, t_k) \leftarrow a_1 \wedge a_2 \wedge \dots \wedge a_m$ 
12:    select atom  $a_i$  in the body of G
13:    choose clause  $a \leftarrow b_1 \wedge \dots \wedge b_p$  in KB/
14:    Rename all variables in  $a \leftarrow b_1 \wedge \dots \wedge b_p$  to have new names
15:    Let  $\sigma$  be Unify( $a_i, a$ ). Fail if Unify returns  $\perp$ .
16:     $G := (yes(t_1, \dots, t_k) \leftarrow b_1 \wedge \dots \wedge b_p \wedge a_2 \wedge \dots \wedge a_m)\sigma$ 
17:  return  $\{V_1 = t_1, \dots, V_k = t_k\}$  where G is  $yes(t_1, \dots, t_k) \leftarrow$ 

```

15.5-rasm. Datalog uchun yuqoridan pastga ma'lum bir shart bilan isbotlash tartibi

Bu va taklifning yuqoridan pastga isbotlash tartibi o'rtasidagi asosiy farq shundaki, o'zgaruvchilar bilan bandlar uchun isbotlash tartibi bilim bazasidan bandlarning nusxalarini olishi kerak. Nusxa olish banddagi o'zgaruvchilarni yangi nomlar bilan qayta nomlaydi. Bu o'zgaruvchilar orasidagi nom ziddiyatlarini

bartaraf etish va bitta isbot turli bandlarning turli misollarini ishlatishi mumkinligi uchun zarurdir.

γ n javob. Ya'ni u quyidagi shaklda bo'ladi:

$yes(t_1, \dots, t_k) \leftarrow$

Bu sodir bo'lganda, algoritim quyidagi javobni qaytaradi:

$V_i = t_1, \dots, V_k = t_k.$

Javob qanday olinayotganiga e'tibor bering; *yes* argumentlari muvaffaqiyatli isbotga olib keladigan dastlabki so'rovda o'zgaruvchilarning misollarini kuzatib boradi.

15.5-rasm SLD hosil qilishlarini qidirish orqali so'rovlarga javob beradigan nodeterministik algoritimni beradi. Bu barcha hosil qilishlarni muvaffaqiyatsizlikka uchramaydigan tegishli tanlovlarni amalga oshirish orqali topish mumkin degan ma'noda nodeterministikdir. Agar barcha tanlovlar muvaffaqiyatsiz bo'lsa, algoritim muvaffaqiyatsiz bo'ladi va hech qanday hosil qilish mavjud emas.

13-qatordagi «tanlash» 5.12-misolga o'xshab qidiruv yordamida amalga oshiriladi. Eslatib o'tamiz, *Unify* (*ai*, *a*) *ai* va *a* ning MGU sini qaytaradi, agar mavjud bo'lsa va agar ular birlashmasa, $\perp$ qaytaradi. *Unify* algoritmi 15.4-rasmda keltirilgan.

15.22-misol. Qo'shiqchi Shakira Kolumbiyaning Atlantiko departamenti poytaxti Barrankilyada tug'ilgan. Shunday qilib, u Kolumbiyada tug'ilgan va shuning uchun Janubiy Amerikada ham tug'ilgan. Quyidagi takliflarni (kengroq bilim bazasining bir qismi sifatida) ko'rib chiqing:

born_in(*shakira*, *barranquilla*)

born_in(*P*, *L*) $\leftarrow$ *part_of*(*S*, *L*) $\wedge$ *born_in*(*P*, *S*).

par_of(*barranquilla*, *atlantico*).

part_of(*atlantico*, *Colombia*).

part_of(*colombia*, *south_america*)

Kolumbiyada kim tug'ilganligini so'rash uchun so'rov quyidagicha bo'ladi:

ask_born_in(*P*, *Colombia*).

15.6-rasmda $P = shakira$ javobi bilan muvaffaqiyatli hosil bo'lish ko'rsatilgan. Ushbu hosil qilishda quyidagi qoidaning ikkita misoli ishlatilganligini e'tiborga oling:

born_in(*P*, *L*) $\leftarrow$ *part_of*(*S*, *L*) $\wedge$ *born_in*(*P*, *S*).

Bir holatda *S* o'rniga *kolumbiya* so'zi, ikkinchi holatda esa *S* o'rniga *atlantico* so'zi almashtirilgan.

15.6. Funksional belgilar va ma'lumot tuzilmalari

Datalog tizim muhokama qiladigan har bir shaxs uchun doimiy ravishda kons-tantadan foydalangan holda nom talab qiladi. Ko'pincha, har bir shaxs uchun alo-hida konstanta talab qilishdan ko'ra, shaxsni uning tarkibiy qismlari nuqtayi naza-ridan aniqlash osonroq.

15.23-misol. Ko'p sohalarida siz vaqtni shaxs sifatida ko'rsatishni xohlaysiz. Masalan, biror kurs soat 11:30 da o'tkazilishini aytishingiz mumkin. Buni amalga oshirish uchun har bir mumkin bo'lgan vaqt uchun alohida doimiy kerak emas, garchi bu mumkin bo'lsa ham. Vaqtni yarim tundan o'tgan soatlar soni va soatdan o'tgan daqiqalar soni orqali aniqlash yaxshiroq. Xuddi shunday, siz ma'lum sanalar haqida mulohaza yuritishni xohlaysiz. Kelajakda cheksiz ko'p mumkin bo'lgan sanalar mavjudligi sababli, har bir sana uchun ixtiyoriy doimiy berishingiz mumkin emas. Yil, oy va kun orqali sanani aniqlash osonroq.

Har bir shaxsni nomlash uchun doimiy ishlatish bilim bazasi faqat cheklangan miqdordagi shaxslarni ifodalashini anglatadi va bilim bazasi yaratilganda shaxslar soni belgilanadi. Ammo, siz cheksiz shaxslar to'plami haqida mulohaza yuritishni xohlashingiz mumkin.

15.24-misol. Faraz qiling, siz ingliz tilidagi savollarni qabul qilib, ularga onlayn ma'lumotlar bazasiga murojaat qilib javob beradigan tizim qurmoqchisiz. Bu holda, har bir gap shaxs sifatida qabul qilinadi. Har bir gapga o'z nomini berishni xohlamaysiz, chunki juda ko'p inglizcha gaplar mavjud. Har bir gapni nomlashdan ko'ra, so'zlarni nomlab, keyin gapni uning tarkibidagi so'zlar ketma-ketligi orqali aniqlash yaxshiroq bo'ladi. Bu yondashuv ko'proq amaliy bo'lishi mumkin, chunki nomlash uchun gaplarga qaraganda kamroq so'zlar mavjud va har bir so'z o'zining tabiiy nomiga ega. Siz, shuningdek, so'zlarni harflar yoki ularning tarkibiy qismlari orqali aniqlashni xohlashingiz mumkin.

$yes(P) \leftarrow born\ in(P, colombia)$

resolve with: $born\ in(P_1, L_1) \leftarrow part\ of(S_1, L_1) \wedge born\ in(P_1, S_1)$

substitution: $\{P_1/P, L_1/colombia\}$

$yes(P) \leftarrow part\ of(S_1, colombia) \wedge born\ in(P, S_1)$

select leftmost conjunct

resolve with: $part\ of(atlantico, colombia)$

substitution: $\{S_1/atlantico\}$

$yes(P) \leftarrow born\ in(P, atlantico)$

resolve with: $born\ in(P_2, L_2) \leftarrow part\ of(S_2, L_2) \wedge born\ in(P_2, S_2)$

substitution: $\{P_2/P, L_2/atlantico\}$

$yes(P) \leftarrow part\ of(S_2, atlantico) \wedge born\ in(P, S_2)$

resolve with: $part\ of(barranquilla, atlantico)$

substitution: $\{S_2/barranquilla\}$

$yes(P) \leftarrow born\ in(P, barranquilla)$

resolve with: $born\ in(shakira, barranquilla)$

substitution: $\{P/shakira\}$

$yes(shakira) \leftarrow$

15.6-rasm. Born in(P,colombia) so'rov uchun ma'lumot.

Funksional belgilar sizga shaxslarni bilvosita tasvirlashga imkon beradi. Shaxsni tavsiflash uchun doimiy ishlatish o'rniga, shaxs boshqa shaxslar orqali tasvirlanadi. Funksional belgilar tilga boy **ma'lumotlar tuzilmalarini** ifodalash imkonini beradi; funksional belgi boshqa obyektlarning tuzilgan to'plamini beradi.

Sintaktik jihatdan **funksional belgi** kichik harf bilan boshlanadigan so'zdir. Atamaning ta'rifi kengaytirilgan bo'lib, atama o'zgaruvchi, doimiy yoki $f(t1, \dots, tn)$ shaklidagi bo'ladi, bu yerda f funksional belgi va har bir ti atamadir. Atamalar ta'rifini kengaytirishdan tashqari, til o'zgarishsiz qoladi.

Atamalar faqat predikat belgilar ichida ko'rinadi. Siz atamalarni anglatuvchi jumalarni yozmaysiz. Biroq, siz funksional belgilaridan foydalanadigan atomlarni o'z ichiga olgan jumalarni yozishingiz mumkin.

Datalog semantikasi yangi sintaksisni aks ettirish uchun kengaytirilishi kerak. ϕ ta'rifi shunday kengaytirilganki, ϕ har bir n -arry funksiya belgisiga D^n dan D ga funksiya tayinlaydi. Doimiy 0-arry (ya'ni hech qanday argumentga ega bo'lmag) funksional belgi sifatida qaralishi mumkin. Shunday qilib, ϕ har bir asosiy atama tomonidan qaysi shaxs belgilanishini aniqlaydi.

15.25-misol. Faraz qiling, siz sanalarni, masalan, 1969-yil 20-iyulni aniqlamoqchisiz. Bu inson o'ya birinchi marta qadam qo'ygan sana. $Ce(Y, M, D), Y$ yili, M oyi va D kuni bilan sanani ifodalashi uchun siz ce (common era) funksional belgidan foydalanishingiz mumkin. Misol uchun, $ce(1969, iyul, 20)$ 1969-yil 20-iyulni ifodalashi mumkin. Xuddi shunday, miloddan avvalgi sanani ifodalash uchun bce belgisini belgilashingiz mumkin.

Funksional belgidan foydalanishning yagona yo'li, funksional belgidan foydalangan holda munosabatlarni aniqlovchi jumalarni yozishdir. « ce » funksiyasini *aniqlash* degan tushuncha yo'q; sanalar kompyuterda shaxslar kabi mavjud emas.

Funksional belgilaridan foydalanish uchun, siz funksiya belgisining argumentlari ustidan miqdoriy ifodalovchi jumalarni yozishingiz mumkin. Misol uchun, 15.7-rasmda $before(D_1, D_2)$ munosabati aniqlangan, agar D_1 sana D_2 sanasidan bir kun oldin bo'lsa, bu to'g'ri bo'ladi. Bu « $<$ » predikati butun sonlar o'rtasidagi «kamroq» munosabatini ifodalaydi deb taxmin qiladi. Bu bandlar orqali ifodalanishi mumkin, lekin ko'pincha oldindan belgilangan bo'ladi, xuddi Prologda bo'lgani kabi Oylar oyning birinchi uchta harfidan iborat doimiyalar bilan ifodalanadi.

$\%before(D_1, D_2)$ is true if date D_1 is before date D_2

$before(ce(Y_1, M_1, D_1), ce(Y_2, M_2, D_2)) \leftarrow$

$Y_1 < Y_2.$

$before(ce(Y, M_1, D_1), ce(Y, M_2, D_2)) \leftarrow$

$month(M_1, N_1) \wedge$

$month(M_2, N_2) \wedge$

$N_1 < N_2.$

$before(ce(Y, M, D_1), ce(Y, M, D_2)) \leftarrow$

$$D_1 < D_2.$$

$\%month(M,N)$ is true if month M is the N th month of the year

month(jan,1).	month(feb,2).	month(mar,3).
month(apr,4).	month(may,5).	month(jun,6).
month(jul,7).	month(aug,8).	month(sep,9).
month(oct,10).	month(nov,11).	month(dec,12).

15.7-rasm. Eramizdagi sanalar uchun «before» munosabatining aksiomatizatsiyasi

Funksional belgilarga ega bandlardan iborat bilim bazasi har qanday funktsiyani hisoblashga qodir. Shunday qilib, bilim bazasi dastur sifatida talqin qilinishi mumkin, bu **mantiqiy dastur** deb ataladi. Mantiqiy dasturlar raqamli kompyuterda hisoblash mumkin bo'lgan har qanday funktsiyani hisoblay oladi.

Tilning bu kengaytmasi katta ta'sirga ega. Bitta funksional belgi va bitta doimiy bilan til cheksiz ko'p asosiy atamalarni va cheksiz ko'p asosiy jummalarni o'z ichiga oladi. Cheksiz ko'p atamalar cheksiz shaxslarni ifodalash uchun ishlatilishi mumkin.

Funksional belgilar quyidagi misoldagi kabi ma'lumotlar tuzilmalarini qurish uchun ishlatiladi.

15.26-misol. Daraxt foydali ma'lumotlar tuzilmasidir. Siz tabiiy tilni qayta ishlash tizimi uchun gapning sintaktik tasvirini qurish uchun daraxtdan foydalanishingiz mumkin. Belgilangan daraxt $node(N, LT, RT)$ yoki $leaf(L)$ bo'lishi mumkin. Shunday qilib, $node$ – bu nom, chap daraxt va o'ng daraxtga aylantiruvchi funktsiya hisoblanadi. $leaf$ funksional belgi barg belgisidan daraxtga aylantiruvchi funktsiya hisoblanadi.

$leaf(L, T)$ munosabati, agar L yorlig'i T daraxtidagi barg yorlig'i bo'lsa, rost bo'ladi. Bu quyidagicha aniqlanishi mumkin:

$at_leaf(L, leaf(L)).$

$at_leaf(L, node(N,LT,RT)) \leftarrow$
 $at_leaf(L,LT)$

$at_leaf(L, node(N,LT,RT)) \leftarrow$
 $at_leaf(L,RT)$

Bu strukturaviy rekursiv dastur misolidir. Qoidalar daraxtlarni ifodalaydigan tuzilmalar uchun barcha holatlarni qamrab oladi.

$in_tree(L, T)$ munosabati, agar L yorlig'i T daraxtidagi ichki tugun yorlig'i bo'lsa, rost bo'ladi. Bu quyidagicha aniqlanishi mumkin:

$in_tree(L, node(L,LT,RT))$

$$\begin{aligned} in_tree(L, node(N, LT, RT)) &\leftarrow \\ in_tree(L, LT) \\ in_tree(L, node(N, LT, RT)) &\leftarrow \\ intree(L, RT). \end{aligned}$$

15.27-misol. Ro'yxat elementlarning ketma-ket tartibga solingan to'plamidir. Ro'yxatlar haqida faqat funksional belgilar va doimiylardan foydalanib mulohaza yuritishingiz mumkin, bunda ro'yxat tushunchasi avvaldan tilda belgilanmagan bo'ladi. Ro'yxat bo'sh bo'lishi yoki ro'yxatni davom ettiradigan element bo'lishi mumkin. Bo'sh ro'yxatni belgilash uchun doimiyni yaratishingiz mumkin. Faraz qiling, bo'sh ro'yxatni belgilash uchun *nil* doimiysini ishlatasiz. Ro'yxatning birinchi elementini *Hd* va qolgan qismini *Tl* bilan ifodalash uchun *cons(Hd, Tl)* funksional belgini tanlashingiz mumkin. *a*, *b*, *c* elementlari mavjud bo'lgan ro'yxat quyidagicha ifodalanadi:

$$cons(a, cons(b, cons(c, nil))).$$

Ro'yxatlardan foydalanish uchun, ularni nima qilish kerakligini yozishingiz kerak. Masalan, *X*, *Y* va *Z* ro'yxatlar bo'lganida, *Z*, *X* elementlarini va undan keyin *Y* elementlarini o'z ichiga olgan bo'lsa, rost bo'lgan *append(X, Y, Z)* munosabati quyidagicha rekursiv aniqlanishi mumkin:

$$append(nil, L, L).$$

$$append(cons(Hd, X), Y, cons(Hd, Z)) \leftarrow$$

$$append(X, Y, Z).$$

cons yoki *nil* haqida hech qanday maxsus narsa yo'q; siz xuddi shu tarzda *foo* va *bar* dan foydalanishingiz mumkin edi.

15.6.1. Funksional belgilar bilan isbot protseduralari

O'zgaruvchilar bilan isbot protseduralari funksiya belgilariga ega bo'lgan holat uchun amalga oshiriladi. Asosiy farq shundaki, atamalar sinfi funksional belgilarni o'z ichiga olish uchun kengaytirilgan.

Funksional belgilardan foydalanish cheksiz ko'p atamalarga olib keladi. To'liqlik uchun jummalarni to'g'ridan to'g'ri bog'lash jummalarni tanlash mezonining adolatli bo'lishini ta'minlashi kerak.

15.28-misol. Adolatning muhimligini ko'rish uchun quyidagi qoidalarini ko'rib chiqing:

$$num(0).$$

$$num(s(N)) \leftarrow num(N).$$

$$a \leftarrow b.$$

$$b.$$

Adolatsiz strategiya dastlab zanjirni yoʻnaltirish uchun birinchi jumlani tanlashi va har bir keyingi tanlovda ikkinchi jumlani tanlashi mumkin. Ikkinchi jumla har doim yangi natija uchun ishlatiladi. Ushbu strategiya hech qachon oxirgi ikkita jumlani tanlamaydi va shuning uchun hech qachon a yoki b ni hosil qilmaydi.

Baʼzi qoidalarni doimiy ravishda eʼtiborsiz qoldirish muammosi **starvation** (resurs ochligi) deb ataladi. **Adolatli** tanlash mezoni shundayki, tanlanishi mumkin boʻlgan har qanday jumla oxir-oqibat tanlanadi. Pastdan yuqoriga isbot protsedurasi cheksiz natijalar ketma-ketligini hosil qilishi mumkin va agar tanlash adolatli boʻlsa, har bir natija oxir-oqibat hosil qilinadi va shuning uchun isbot protsedurasi toʻliq boʻladi.

Pastdan yuqoriga isbot protsedurasi Datalog bilan bir xil (15.5-rasmga qarang). Birlashtirish (unification) murakkablashadi, chunki u atamalar tuzilmasiga rekursiv tarzda tushishi kerak. Birlashtirish algoritmda bitta oʻzgarish mavjud: oʻzgaruvchi X , X ning oʻzi boʻlmagan va X ning oʻzi boʻlmagan t atamasi bilan birlashmaydi. Agar X/t , S ga qoʻshilishi kerak boʻlsa, algoritim $\perp$ ni qaytarishi kerak. Bu holatni tekshirish **paydo boʻlishni tekshirish** (occurs check) deb ataladi. Baʼzan (masalan, Prologda) bu tekshiruv tushirib qoldiriladi, chunki uni olib tashlash isbot protsedurasini samaraliroq qiladi, garchi uni olib tashlash quyidagi misolda koʻrsatilganidek, isbotlash protsedurasini asossiz qilsa ham.

Birinchi darajali va ikkinchi darajali mantiq

Birinchi tartibli predikatlar hisobi – bu takliflar hisobini funksional belgilar va mantiqiy oʻzgaruvchilarga ega boʻlgan atomlarni oʻz ichiga oladigan mantiq. Barcha mantiqiy oʻzgaruvchilar «hamma uchun» ($\forall$) va «mavjud» ($\exists$) shartlari boʻyicha aniq miqdorga ega boʻlishi kerak. Birinchi darajali predikat hisoblashning semantikasi ushbu bobda taqdim etilgan mantiqiy dasturlar semantikasi bilan bir xil, lekin boyroq operatorlar toʻplami bilan.

Mantiqiy dasturlar tili birinchi darajali predikat hisoblashning amaliy kichik toʻplamini tashkil etadi, bu koʻpgina vazifalar uchun foydali boʻlganligi sababli ishlab chiqilgan. Birinchi darajali predikat hisoblash mantiqiy dasturlarga disyunksiya va aniq kvantifikatsiyani qoʻshuvchi til sifatida koʻrilishi mumkin.

Birinchi darajali mantiq *birinchi darajali* deb ataladi, chunki u domen ichidagi shaxslar ustidan miqdorni aniqlashga imkon beradi. Birinchi darajali mantiq, na oʻzgaruvchi sifatida predikatlarni, na predikatlar boʻyicha miqdorni aniqlashga imkon beradi

Ikkinchi darajali mantiq birinchi darajali munosabatlar ustida miqdorni aniqlash uchun ruxsat beradi va argumentlari birinchi darajali munosabatlar boʻlgan predikatlar ruxsat beradi. Bular ikkinchi darajali munosabatlardir. Masalan, ikkinchi darajali mantiq formulasi

$$\forall R \text{ symmetric}(R) \leftrightarrow (\forall X \forall Y R(X, Y) \rightarrow R(Y, X))$$

ikkinchi tartibli *simmetrik* munosabatni belgilaydi, agar uning argumenti simmetrik munosabat bo'lsa, to'g'ri bo'ladi.

Ko'pgina ilovalar uchun ikkinchi darajali mantiq zarur bo'lib tuyuladi, chunki tranzitiv yopilishni (transitive closure) birinchi darajada aniqlash mumkin emas.

Misol uchun, siz *before* munosabatini *next* munosabatining tranzitiv yopilishi bo'lishini xohlasangiz, bu yerda $(X, s(X))$ haqiqiy bo'ladi. *next* ni «keyingi millisekund» va *before*ni «oldin» deb tasavvur qiling. Tabiiy birinchi darajali ta'rif quyidagicha bo'ladi:

$$\forall X \forall Y \text{ before}(X, Y) \leftrightarrow (Y = s(X) \vee \text{ before}(s(X), Y)).$$

Ushbu ifoda ta'rifni aniq aks ettirmaydi, chunki masalan:

$$\forall X \forall Y \text{ before}(X, Y) \rightarrow \exists W Y = s(W)$$

formuladan mantiqan kelib chiqmaydi, chunki formula (15.1)ning nostandart modellari mavjud bo'lib, unda *Y cheksizlikni* bildiradi. Tranzitiv yopilishni aks ettirish uchun *before* minimal predikat bo'lishini bildiruvchi formulaga muhojiz. Buni ikkinchi darajali mantiq yordamida ifodalash mumkin.

Birinchi darajali mantiq **yarim qarorli (semi-decidable)**, bu degani, har qanday to'g'ri bayonotni isbotlaydigan to'g'ri va to'liq isbot protsedurasi mavjud, lekin u to'xtamasligi mumkin. Ikkinchi darajali mantiqni hal qilib bo'lmaydi: Turing mashinasida ishonchli va to'liq isbotlash protsedurasini amalga oshirish mumkin emas.

15.29-misol. Faqat bitta banddan iborat bilimlar bazasini ko'rib chiqing:

$$It(X, s(X)).$$

Faraz qilaylik, mo'ljallangan talqin butun sonlar sohasi bo'lib, unda *It* «kichik» degan ma'noni bildiradi va *s(X)* *X* dan keyingi butun sonni bildiradi. *It(Y, Y)* so'rovi bajarilmasligi kerak, chunki u mo'ljallangan talqinda noto'g'ri; o'zidan kichikroq raqam yo'q. Biroq, agar *X* va *s(X)* ni birlashtirish mumkin bo'lsa, bu so'rov muvaffaqiyatli bo'ladi. Bunday holda, isbotlash protsedurasini asossiz bo'ladi, chunki aksioma modelida yolg'on bo'lgan narsani chiqarish mumkin edi.

Quyidagi misol funksional belgilar bilan SLD o'Ichamining tafsilotlarini ko'rsatadi.

15.30-misol. Quyidagi jummalarni ko'rib chiqing:

$$\text{append}(c(A, X), Y, c(A, Z)) \leftarrow$$

$$\text{append}(X, Y, Z).$$

$$\text{append}(\text{nil}, Z, Z).$$

Hozircha bu nimani anglatishini e'tiborsiz qoldiring. Kompyuter singari, buni belgilarni boshqarish muammosi sifatida ko'rib chiqing. Quyidagi so'rovni ko'rib chiqing:

$$\text{ask append}(F, c(L, \text{nil}), c(l, c(i, c(s, c(t, \text{nil}))))).$$

Quyidagilar hosiladir:

$yes(F, L) \leftarrow append(F, c(L, nil), c(l, c(i, c(s, c(t, nil))))))$
 $resolvewithappend(c(A_1, X_1), Y_1, c(A_1, Z_1)) \leftarrow append(X_1, Y_1, Z_1)$
 $substitution: \{F/c(l, X_1), Y_1/c(L, nil), A_1/l, Z_1/c(i, c(s, c(t, nil)))\}$
 $yes(c(l, X_1), L) \leftarrow append(X_1, c(L, nil), c(i, c(s, c(t, nil))))$
 $resolvewithappend(c(A_2, X_2), Y_2, c(A_2, Z_2)) \leftarrow append(X_2, Y_2, Z_2)$
 $substitution: \{X_1/c(i, X_2), Y_2/c(L, nil), A_1/i, Z_2/c(s, c(t, nil))\}$
 $yes(c(l, c(i, X_2)), L) \leftarrow append(X_2, c(L, nil), c(s, c(t, nil)))$
 $resolvewithappend(c(A_3, X_3), Y_3, c(A_3, Z_3)) \leftarrow append(X_3, Y_3, Z_3)$
 $substitution: \{X/c(s, X_3), Y_3/c(L, nil), A_3/s, Z_3/c(t, nil)\}$
 $yes(c(l, c(i, c(s, X_3))), L) \leftarrow append(X_3, c(L, nil), c(t, nil))$

Bu bosqichda ikkala band ham qo'llanilishi mumkin. Birinchi bandni tanlash quyidagi natijani beradi:

$resolve with append(c(A_4, X_4), Y_4, c(A_4, Z_4)) \leftarrow append(X_4, Y_4, Z_4)$
 $substitution: \{X/c(t, X_4), Y_4/c(L, nil), A_4/t, Z_4/nil\}$
 $yes(c(l, c(i, c(s, X_3))), L) \leftarrow append(X_4, c(L, nil), nil).$

Umumlashtirilgan javob jumlasining tanasida boshi atom bilan birikadigan gaplar hozircha mavjud emas. Isbot amalga oshmadi.

Birinchi jumla o'rniga ikkinchi jumlaning tanlash quyidagi natijani beradi:

$resolve with append(nil, Z_5, Z_5) substitution: \{Z_5/c(t, nil), X_3/nil, L/t\}$
 $yes(c(l, c(i, c(s, nil))), t) \leftarrow$

Bu bosqichda isbot muvaffaqiyatli bo'ladi, natija:

$F=c(l, c(i, c(s, nil))), L=t.$

Ushbu bobning qolgan qismida ro'yxatlar uchun Prolog notatsiyasining «syntactic sugar» yozuvi qo'llaniladi. Bo'sh ro'yxat, *nil*, [] deb yoziladi. 15.27-misoldagi *cons* (*E*, *R*) bo'lgan birinchi *E* elementi va *R* ro'yxatining qolgan qismi bo'lgan ro'yxat [*E* | *R*] sifatida yoziladi. Belgilanishning yana bir soddalashtirilgani mavjud: [*X* | [*Y*], [*X*, *Y*]] shaklida yoziladi, bu yerda *Y* qiymatlar ketma-ketligi bo'lishi mumkin. Masalan, [*a* | []], [*a*] kabi yoziladi [*b* | [*a* | []]], [*b*, *a*] shaklida yoziladi. [*a* | [*b* | *C*]] atamasi [*a*, *b* | *C*] shaklida yoziladi.

15.31-misol. Ro'yxat notatsiyasidan foydalanib, oldingi misoldagi *append* quyidagicha yozilishi mumkin:

$append([A|X], Y, [A|Z]) \leftarrow$

$append(X, Y, Z).$

$append([], Z, Z).$

So'rov:

$ask append(F, [L], [l, i, s, t])$

quyidagi javobga ega: $F=[l, i, s], L=t$. Isbot xuddi oldingi misoldagidek. Isbot usuli bo'yicha hech narsa o'zgarmadi, faqat o'zgartirilgan funksional belgi va konstanta mavjud.

15.7. Tabiiy til ilovalari

Tabiiy tilni qayta ishlash bilim va mulohazalar nazariyalarini ishlab chiqish va baholash uchun qiziqarli va murakkab sohadir. Sun'iy intellektning ko'plab muammolari ushbu sohada yuzaga keladi; «tabiiy til muammosini» hal qilish «sun'iy intellekt muammosini» hal qilish kabi qiyin, chunki har qanday soha tabiiy tilda ifodalanishi mumkin. **Hisoblash tilshunosligi** sohasida ko'plab usullar va bilimlar mavjud. Ushbu darslikda faqat umumiy ma'lumot berilgan.

Tabiiy tilni qayta ishlashni o'rganish uchun kamida uchta sabab mavjud:

Foydalanuvchilar o'z shartlariga ko'ra muloqot qilishni xohlashadi va ko'pchilik tabiiy tilni sun'iy tildan yoki grafik foydalanuvchi interfeysidan afzal ko'radi. Bu ayniqsa tasodifiy foydalanuvchilar va yangi o'zaro ta'sir ko'nikmalarini o'rganishga vaqt yoki xohish-istagi bo'lmagan foydalanuvchilar, masalan, menejerlar va bolalar uchun muhimdir.

Tabiiy tilda yozilgan katta hajmdagi ma'lumotlar mavjud bo'lib, ulardan kompyuterlar yordamida foydalanish mumkin. Tvitarlar, bloglar, darsliklar, yangiliklar, biznes va hukumat hisobotlari va ilmiy maqolalar shaklida doimiy ravishda ma'lumotlar ishlab chiqarilmoqda, ularning ko'pi onlayn mavjud. Ko'p ma'lumotni talab qiladigan tizim kompyuterlarda mavjud bo'lgan ma'lumotlarning ko'pini olish uchun tabiiy tilni qayta ishlashga qodir bo'lishi kerak.

Sun'iy intellekt muammolarining aksariyati tabiiy tilni qayta ishlashda juda aniq va ochiq shaklda yuzaga keladi va shuning uchun umumiy nazariyalarni sinab ko'rish uchun yaxshi sohadir.

Tabiiy tilni qayta ishlashning ikkita an'anasi

Sun'iy intellektidagi tabiiy tilni qayta ishlash tizimlarida ikkita asosiy yondashuv mavjud:

Keng qamrovli tizimlar, tabiiy muhitda tilni tushunishga harakat qilmoqda, lekin ba'zi xatolarni qabul qilishga tayyor.

Cheklangan domenlarda ishlatiladigan tizimlar, bu yerda foydalanuvchilar boshqariladigan tabiiy tilni qo'llashlari kutiladi va natijalar aniq bo'lishi mumkin.

Birinchi turdagi tizimlar smartfonlarda oldindan bashoratli yozuvlar uchun va keng qamrovli til tarjimasini uchun ishlatiladi. Ikkinchi turdagi tizimlar, bir xil odamlar tizim bilan uzluksiz ravishda o'zaro aloqa qilganda, masalan, supermarket zaxiralari uchun ma'lumotlar bazasi so'rovlari, o'yinlar yoki til uslublashtirilganda qo'llaniladi, masalan, Jeopardy o'yinidagi savolni tahlil qilish! (Watson tizimi tomonidan amalga oshirilganidek; havolalarga qarang).

Asosiy farqlardan biri grammatik bo'lmagan jumlar yoki savollar, yoki notanish so'zlarga qanday munosabatda bo'lishda yotadi. Birinchi turdagi tizimlar bu jumalarni grammatik bo'lganlar kabi qabul qiladi. Biroq, ikkinchi turdagi

tizimlar tushuntirishni yoki qayta yozishni so'rashi yoki hatto bunday jumlar uchun ishlamay qolishi mumkin.

Keng qamrovli tizimlardan foydalanish noto'g'ri bo'lish xavfi past bo'lgan yoki siz bilan muloqot qilayotgan shaxs to'g'ri ma'noni topish uchun savol berishi va o'zaro aloqa qilishi mumkin bo'lgan hollarda yaxshi. Ammo, xatolar yoki noto'g'ri tushunish uchun katta xarajatlar mavjud bo'lgan hollarda, masalan, yuridik shartnomalar tuzishda, bunday tizimdan foydalanishni xohlamasligingiz mumkin.

Hozirgi zamonaviy yondashuv keng qamrovli tizimlarni ma'lumotlardan o'rganishdir, masalan, neyron tarmoqlar va chuqur o'rganish orqali.

Ushbu bobda ikkinchi turdagi tizimlar ko'rib chiqiladi, bu yerda hali ham qo'llaniladigan texnikalar keltirilgan.

Tabiiy tilning kamida uchta asosiy jihati mavjud:

Sintaksis. Sintaksis tilning shaklini tavsiflaydi. Tabiiy til mantiq va kompyuter dasturlari uchun ishlatiladigan rasmiy tillarga qaraganda ancha murakkab. Sintaksis odatda grammatika bilan belgilanadi. Ba'zi tabiiy til modellari tahlil qilish daraxtini aniq tuzmagan holda, har bir so'zni o'z kontekstidan bashorat qiluvchi neyron tarmoqlar yordamida taqdim etiladi (8.5-mavzuga qarang).

Semantika. Semantika tilning gaplari yoki jumlarining ma'nosini beradi. Umumiy semantik nazariyalar mavjud bo'lsa-da, tabiiy tilni qayta ishlash tizimlari ma'lum bir dastur uchun qurilganda, odatda eng oddiy tasvirlash usulidan foydalaniladi. Masalan, keyingi rivojlanish bilimlar bazasida so'zlar va tushunchalar o'rtasida qat'iy taqqoslashni o'z ichiga oladi, bu ko'plab sohalarga mos kelmaydi, ammo rivojlanishni osonlashtiradi. Xuddi shu so'z turli xil ma'nolarga ega bo'lishi mumkin (masalan, bank – Daryo bo'yi va pulni saqlash uchun bank). Neyron modellari odatda so'zlarni qo'shimchalar deb nomlangan sobit uzunlikdagi vektorlar yordamida ifodalaydi, ular har bir so'zni kontekstda yoki ular o'rgatilgan boshqa vazifalarda bashorat qilish uchun yetarli semantikani ifodalaydi.

Pragmatika. Pragmatik komponent gaplarning dunyo bilan qanday bog'liqligini tushuntiradi. Tilni to'liq tushunish uchun agent faqat jumlaning, balki jumlaning kontekstini, dunyo holatini, gapiruvchi va tinglovchining maqsadlarini, maxsus kelishuvlarni va boshqalarni hisobga olishi kerak.

Ushbu jihatlar orasidagi farqni tushunish uchun sun'iy intellekt darsligining boshida paydo bo'lishi mumkin bo'lgan quyidagi jummalarni ko'rib chiqing:

Bu darslik sun'iy intellekt haqida.

Yashil qurbaqalar tinch uxlaydi.

Rangsiz yashil g'oyalar g'azab bilan uxlaydi.

G'azab bilan uxlaydi g'oyalar yashil rangsiz.

Birinchi jumla bunday darslikning boshida juda mos keladi; u sintaktik, semantik va pragmatik jihatdan yaxshi shakllangan. Ikkinchi jumla sintaktik va semantik jihatdan yaxshi shakllangan, lekin sun'iy intellekt darsligi boshida juda g'alati ko'rinadi; bu kontekst uchun pragmatik jihatdan yaxshi shakllanmagan. Oxirgi ik-

kita jumla tilshunos Noam Chomsky [1957] tomonidan yozilgan. Uchinchi jumla sintaktik jihatdan yaxshi shakllangan, lekin semantik jihatdan bemaʼnilikdir. Toʻrtinchi jumla sintaktik jihatdan notoʻgʻri shakllangan; u sintaktik, semantik yoki pragmatik jihatdan hech qanday maʼnoga ega emas.

Keyingi boʻlim foydalanuvchilar rioya qilishlari kerak boʻlgan uslublashtirilgan tabiiy tilni qoʻllagan holda juda tor domenlarga tegishli boʻlgan tabiiy til soʻrovlariga javob beruvchi tizimni qanday yozishni koʻrsatadi. Bu yondashuv noaniqlik kam boʻlgan sohalar uchun mos boʻlishi mumkin. Boshqa tomondan, 9.36-misol va 10.5-misolda keltirilgan yordam tizimi kabi keng, ammo yuzaki tizimlar mavjud.

15.7.1. Kontekstsiz grammatika uchun aniq gaplardan foydalanish

Ushbu boʻlimda tabiiy tilning sintaksisi va semantikasining jihatlarini ifodalash uchun aniq boʻlaklardan qanday foydalanish koʻrsatilgan.

Tillar oʻzlarining qonuniy gaplari bilan aniqlanadi. **Gaplar** – bu ketma-ketlikdagi **tokenlar** boʻlib, odatda tilning soʻzlarini, umumiy iboralarni (masalan, «sunʼiy intellekt») ifodalaydi va koʻpincha tinish belgilarini oʻz ichiga oladi. Baʼzi modellar soʻzlarni ularning qismlari orqali ifodalaydi, masalan, «ing» va «er» kabi qoʻshimchalarni alohida belgilar sifatida ajratib olish yoki belgilar ketma-ketligini ishlatish (8.11-misolda boʻlgani kabi). Gaplar bu yerda roʻyxatlar yordamida ifodalanadi va belgilar qoʻsh tirnoq ichida yozilgan qatorlar sifatida koʻrsatiladi.

Qonuniy gaplar grammatikaga muvofiq belgilanadi. **Kontekstsiz grammatika** qayta yozish qoidalarini toʻplami bilan belgilanadi, bunda **noterminal** belgilar terminal va noterminal belgilar ketma-ketligiga aylanadi. Tilning gaplari bunday qayta yozish qoidalarini orqali hosil qilingan **terminal belgilar** ketma-ketligidan iborat. Masalan, quyidagi grammatika qoidasi:

sentence $\rightarrow$ noun_phrase, verb_phrase

bu degani **noterminal gap** belgisi ot soʻz birikmasidan keyin feʼl soʻz birikmasi boʻlishi mumkinligini bildiradi. « $\rightarrow$ » belgisi «qayta yozish mumkin» degan maʼnoni anglatadi.

Kontekstsiz grammatika baʼzi tabiiy tillar grammatikasining dastlabki taxminini beradi. Tabiiy tillar uchun terminal belgilar – bu tilning tokenlari. Agar tabiiy tilning jumlasini tokenlar roʻyxati sifatida ifodalangan boʻlsa, quyidagi aniq qoida shuni anglatadiki, agar soʻzlar roʻyxati ot iborasi bilan boshlanib, feʼl iborasi bilan davom etsa, u jumla hisoblanadi:

sentence(S) $\leftarrow$ noun_phrase(N) $\wedge$ verb_phrase(V) $\wedge$ append(N, V, S).

«Country» soʻzi ot ekanligini aytish uchun siz quyidagicha yozishingiz mumkin:

noun([«country»]).

Muayyan jumlarlar grammatikasi (DCG) deb nomlanuvchi grammatik qoidalarning muqobili, sodda ko'rinishi aniq qo'shimchalarni talab qilmasdan ma'lum jumladan foydalanadi. Kontekstsiz grammatikani ifodalash uchun har bir terminal bo'lmagan belgi s ikkita argumentli predikatga aylanadi, $s(L_1, L_2)$, bu L_2 ro'yxati L_1 ro'yxatining oxiri bo'lsa va L_1 ro'yxatidagi barcha so'zlar L_2 dan oldin s toifasidagi so'zlar ketma-ketligini tashkil qilsa, haqiqatdir. L_1 va L_2 ro'yxatlari birgalikda noterminal belgi bilan belgilangan sinfni tashkil etuvchi so'zlarning **farqlari ro'yxatini** tashkil qiladi, chunki ular o'rtasidagi farq sintaktik kategoriyani tashkil qiladi.

15.32-misol. Agar L_2 ro'yxat L_1 ro'yxatning oxiri bo'lib, L_2 dan oldingi L_1 dagi barcha so'zlar ot so'z birikmasini hosil qilsa, bu ifodada *noun phrase* (L_1, L_2) to'g'ri bo'ladi. $L_2 - L_1$ ning ot so'z birikmasidan keyingi qismi.

nounphrase([«large», «country», «bordering», «Paraguay», «borders», «Chile»])
[«bordering», «Paraguay», «borders», «Chile»])

Bunda «katta mamlakat» ot iborasini tashkil qilgani uchun haqiqatdir.

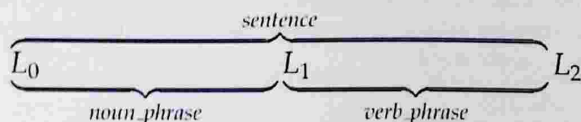
noun_phrase([«large», «country», «bordering», «Paraguay», «borders», «Chile»])
[«borders», «Chile»])

Bunda «katta mamlakat Paragvay bilan chegaradosh» ot iborasini tashkil qilgani uchun haqiqatdir.

Quyidagi grammatik qoida:

sentence $\rightarrow$ *noun_phrase*, *verb_phrase*

L_0 va L_1 o'rtasida ot ibora va L_1 va L_2 o'rtasida fe'l iborasi bo'lsa, L_0 va L_2 o'rtasida gap borligini anglatadi:



Bu grammatika qoidasini quyidagi aniqlangan qoida sifatida yozish mumkin:

sentence(L_0, L_2) $\leftarrow$

nounphrase(L_0, L_1) $\wedge$

verb_phrase(L_1, L_2).

Umuman olganda, qoida

$$h \rightarrow b_1, b_2, \dots, b_n$$

h ning b_i dan so'ng b_i , va shu ketma-ketlikda b_n bo'lishini bildiradi va muayyan jumla sifatida yoziladi:

$$h(L_0, L_n) \leftarrow$$

$$b_1(L_0, L_1) \wedge$$

$$b_2(L_1, L_2) \wedge$$

.

.

$$b_n(L_{n-1}, L_n).$$

Bu ifodada

$$\overbrace{L_0 \quad L_1 \quad L_2 \cdots L_{n-1}}^h L_n$$

$\underbrace{\hspace{1.5cm}}_{b_1} \quad \underbrace{\hspace{1.5cm}}_{b_2} \quad \underbrace{\hspace{2.5cm}}_{b_n}$

L_i unikal o'zgaruvchilardir.

Noterminal belgi h , $t_1, \dots, t_n$ terminal belgilarga o'zgartirilishini aytish uchun quyidagicha yoziladi:

$$h([t_1, \dots, t_n | T], T)$$

ni quyidagicha yozish mumkin:

$$\overbrace{t_1, \dots, t_n}^h T$$

shunday qilib, $h(L_1, L_2)$ haqiqat bo'ladi, agar $L_1 = [t_1, \dots, t_n | L_2]$ bo'lsa.

15.33-misol. Noterminal h noterminal a dan keyin noterminal b ga, undan keyin terminal belgilar c va d ga, so'ngra noterminal belgi e ga, keyin esa terminal belgi f va noterminal belgi g ga qayta yozilishi mumkinligini belgilovchi qoida quyidagicha yozilishi mumkin:

$$h \rightarrow a, b, [c, d], e, [f], g$$

va quyidagicha ifodalash mumkin:

$$h(L_0, L_6) \leftarrow$$

$$a(L_0, L_1) \wedge$$

$$b(L_1, [c, d | L_3]) \wedge$$

$$e(L_3, [f | L_5]) \wedge$$

$$g(L_5, L_6).$$

E'tibor bering, $L_2 = [c, d \mid L_3]$ va $L_4 = [f \mid L_3]$ tarjimalari qo'lda qilingan.

15.8-rasmda ingliz tilidagi savollarning oddiy grammatikasi ko'rsatilgan. E'tibor bering, % satrning qolgan qismi sharh ekanligini bildiradi. 15.9-rasmda ushbu grammatika bilan ishlatilishi mumkin bo'lgan so'zlar va ularning nutq qismlarining oddiy lug'ati berilgan. Birinchi qoida «What country borders Chile?» yoki «What large county bordering Paraguay borders Chile?» kabi savollar uchun, ikkinchi qoida esa «What is bordering Chile?» kabi savollar uchun ishlatilishi mumkin.

15.34-misol. «What large county bordering Paraguay borders Chile?» savolini ko'rib chiqing. Bu har bir so'z uchun alohida satrlar ro'yxati sifatida ifodalana-di. 15.8-rasmdagi grammatika, 15.9-rasmdagi lug'at va 15.3-rasmdagi *namening* ta'rif uchun so'rov

ask_nounphrase([«large», «country», «bordering», «Paraguay», «borders», «Chile»])

uning javobi

$R = ["bordering", "Paraguay", "borders", "Chile"]$

bunda «large country» ot iborasi ekanligini bildiradi.

Yana bir javob:

$R = ["borders", "Chile"]$

bunda «large county bordering Paraguay» ot iborasi ekanligini bildiradi.

%A noun phrase is a determiner (or article, such as «the» or «a») followed by

%adjectives followed by a noun followed by an optional modifying phrase

$noun_phrase(L_0, L_4) \leftarrow det(L_0, L_1) \wedge$

$adjectives(L_1, L_2) \wedge$

$noun(L_2, L_3) \wedge$

$omp(L_3, L_4).$

%Adjectives consist of a (possibly empty) sequence of adjectives

$adjectives(L, L).$

$adjectives(L_0, L_2) \leftarrow$

$adj(L_0, L_1) \wedge$

$adjectives(L_1, L_2).$

%A modifying phrase is a relation (verb or preposition) followed by a noun phrase

$mp(L_0, L_2) \leftarrow$

$reln(L_0, L_1) \wedge$

$noun_phrase(L_1, L_2).$

%An optional modifying phrase is a modifying phrase or nothing

$omp(L, L).$

$omp(L_0, L_1) \leftarrow mp(L_0, L_1).$

%Some simple questions are «What» and «What is» questions
 $question([\langle\langle What \rangle\rangle|L_0], L_2) \leftarrow nounphrase(L_0, L_1) \wedge mp(L_1, L_2). \leftarrow$
 $question([\langle\langle What \rangle\rangle, \langle\langle is \rangle\rangle|L_0], L_1) \leftarrow$
 $mp(L_0, L_1).$

15.8-rasm. Ingliz tilidagi oddiy savollar uchun kontekstsiz grammatika

15.7.2. Grammatikani kengaytirish

Kontekstsiz grammatika tabiiy tillarning, masalan, ingliz tili grammatikasi-ning murakkabligini yetarli darajada ifodalay olmaydi. Ushbu grammatikani yanada ifodalash uchun ikkita mexanizm qo'shilishi mumkin:

noterminal belgilariga qo'shimcha argumentlar
 qoidalarga ixtiyoriy cheklovlar.

Qo'shimcha argumentlar bizga bir nechta narsalarni, jumladan, sintaktik daraxt tuzish va ma'lumotlar bazasiga so'rov yozishni amalga oshirish imkonini beradi. Ixtiyoriy argumentlar va shartlardan foydalanish muayyan jumla grammatikasi kontekstsiz grammatikadan ko'proq narsani ifodalashi mumkinligini anglatadi; bu Tyuring mashinasi yordamida hisoblash mumkin bo'lgan hamma narsani ifodalashi mumkin (15.13-mashqqa qarang).

15.7.3. Ma'lumotlar bazasi uchun tabiiy til interfeysini qurish

Yuqoridagi grammatikani oddiy tabiiy til interfeysi sifatida ma'lumotlar bazasida qo'llash uchun kengaytirish mumkin. Subiboralarni tahlil daraxtlariga aylantirish o'rni, ularni bevosita so'z turkumi tegishli bo'lgan obyektga aylantirish mumkin. Buning uchun quyidagi soddalashtirilgan taxminlarni qabul qilaylik, ular har doim ham to'g'ri emas, lekin dastlabki yaqinlashuv uchun foydalidir:

Atoqli otlar (masalan, «Chili») shaxslarni ifodalaydi.

Otlar (masalan, «country») va **sifatlar** (masalan, «large») xususiyatlarga mos keladi.

Fe'llar (masalan, «borders») va **predloglar** (masalan, «next to») ikki shaxs, **subyekt** va **obyekt** orasidagi ikkilik munosabatni ifodalaydi.

$det(L, L).$	$adj([\langle\langle large \rangle\rangle L], L).$
$det([a L], L).$	$reln([\langle\langle bordering \rangle\rangle L], L).$
$det([the L], L).$	$reln([\langle\langle borders \rangle\rangle L], L).$
$noun([\langle\langle country \rangle\rangle L], L).$	$reln([\langle\langle next \rangle\rangle, \langle\langle to \rangle\rangle L], L).$
$noun([\langle\langle city \rangle\rangle L], L).$	$reln([\langle\langle the \rangle\rangle, \langle\langle name \rangle\rangle, \langle\langle of \rangle\rangle L], L).$
	$noun([N L], L):$ $-name(E\ N).$

15.9-rasm. Oddiy lug'at

Bunday holda, ot iborasi uni belgilaydigan xususiyatlar to'plamiga ega bo'lgan shaxsni ifodalaydi. Savolga javob berish uchun tizim ushbu xususiyatlarga ega

bo'lgan shaxsni topishi mumkin. O'zgartiruvchi ibora (masalan, taklif so'z birikmasi yoki nisbiy gap) boshqa shaxsga bo'lgan munosabat nuqtayi nazaridan shaxsni tavsiflaydi. Quyida faqat cheklangan tabiiy tildan ma'lumotlar bazasi so'rovlarini amalga oshirish uchun zarur bo'lgan lingvistik tuzilma nazarda tutilgan.

15.35-misol. «What large country borders Chile?» gapida «large country» iborasi «borders» fe'lining subyekt bo'lib, «Chile» obyekt hisoblanadi. Faraz qilaylik, bizda 15.2-rasmda ko'rsatilgan geografik ma'lumotlar bazasi mavjud. Subyekt bo'lgan S shaxs uchun $large(S)$ va $country(S)$ haqiqat bo'ladi. Obyekt – bu «chile» shaxsi va fe'l $borders(S, chile)$ ni ifodalaydi. Shunday qilib, «What large country borders Chile?» savoli quyidagicha so'rovga aylantirilishi mumkin:

$ask\ large(S) \wedge country(S) \wedge borders(S, chile)$

bu yerda $large$ – ikki million kvadrat kilometrdan ortiq mamlakatlar uchun to'g'ri bo'lishi mumkin bo'lgan predikat. «What is the name of the capital of a Spanish-speaking country that borders Argentina?» savoli « S qiymatini so'rash sifatida tarjima qilinishi mumkin

$ask\ name(C, S) \wedge capital(X, C) \wedge language(X, spanish)$

$\wedge borders(X, argentina).$

15.10-rasmda ingliz tilidagi savolni tahlil qiladigan va bir vaqtning o'zida javob beradigan oddiy grammatika ko'rsatilgan. Bu ingliz tili grammatikasining ko'p qismini, masalan, predloglar va fe'llar yoki aniqlovchilar va sifatlar o'rtasidagi farqlarni e'tiborsiz qoldiradi va savol grammatik bo'lmasa ham, ma'no haqida taxmin qiladi. Sifatlar, otlar va ot iboralari shaxsni anglatadi. Predikatlarga qo'shimcha argument – bu sifatlar va otlarni qanoatlantiradigan shaxs. Bu yerda mp o'zgartiruvchi ibora bo'lib, u predlog iborasi yoki nisbiy gap bo'lishi mumkin. $reln$ – bu ikki shaxs, ya'ni subyekt va obyekt o'rtasidagi munosabat bo'lib, bu $reln$ predikatining qo'shimcha argumentlari hisoblanadi.

15.36-misol. Aytaylik, $question(Q, A)A$ savoliga Q javob ekanligini bildiradi, bu yerda savol so'zlar ro'yxatidir. Quyida 15.10-rasmdagi jumalardan, hatto u yerda qo'llaniladigan lug'at juda cheklangan bo'lsa ham, savollar berishning ba'zi usullari keltirilgan. Quyida tinish belgilari e'tiborga olinmaydi.

Quyidagi qoida «What is next to Chile?» kabi savollarga javob berish uchun ishlatiladi:

$question([«What», «is»|L_0], Ind) \leftarrow$

$mp(L_0, [], Ind).$

E'tibor bering, $[]$ savolda o'zgartiruvchi iboradan keyin hech narsa yo'qligini bildiradi.

Quyidagi qoida «What is a large Spanish-speaking country next to Chile?» kabi savollarga javob berish uchun ishlatiladi:

$question([«What», «is»|L], Ind) \leftarrow$

% Ot iborasi aniqlovchi bilan boshlanadi, undan keyin sifatlar va ot keladi

% ixtiyoriy ravishda bir xil shaxsga tegishli aniqlovchi ibora keladi

noun_phrase(L_1, L_4, Ind) $\leftarrow$
adjectives(L_1, L_2, Ind) $\wedge$
noun(L_2, L_3, Ind) $\wedge$
omp(L_3, L_4, Ind).

%Adjectives consist of a sequence of adjectives about the same individual
adjectives(L_0, L_2, Ind) $\leftarrow$ *adj*(L_0, L_1, Ind) $\wedge$ *adjectives*(L_1, L_2, Ind).
adjectives(L, L, Ind).

%A modifying phrase/relative clause is a relation (verb or preposition) followed by a noun phrase

mp($L_0, L_2, Subject$) $\leftarrow$
reln($L_0, L_1, Subject, Object$) $\wedge$
noun_phrase($L_1, L_2, Object$).

%An optional modifying phrase is either a modifying phrase or nothing

omp(L_0, L_1, Ind) $\leftarrow$ *mp*(L_0, L_1, Ind).*omp*(L, L, Ind).

%*adj*(L_0, L_1, Ind) is true if $L_0 - L_1$ is an adjective that is true of *Ind*
adj(["large"] L , L, Ind) $\leftarrow$ *large*(*Ind*).

adj([*LangName*, «speaking»] L , L, Ind) $\leftarrow$
name(*Lang*, *LangName*) $\wedge$ *language*(*Ind*, *Lang*).

adj([*a*] L , L, Ind).

% *noun*(L_0, L_1, Ind) is true if $L_0 - L_1$ is a noun that is true of *Ind*
noun(["country"] L , L, Ind) $\leftarrow$ *country*(*Ind*).

noun([*N*] L , L, E) $\leftarrow$ *name*(*E*, *N*).

%*reln*(L_0, L_1, Sub, Obj) is true if $L_0 - L_1$ is a relation on individuals *Sub* and *Obj*

jreln(["borders"] L , L, Sub, Obj) $\leftarrow$ *borders*(*Sub*, *Obj*).

reln(["bordering"] L , L, Sub, Obj) $\leftarrow$ *borders*(*Sub*, *Obj*).

reln(["the", «capital», «of»] L , L, Sub, Obj) $\leftarrow$ *capital*(*Obj*, *Sub*).

reln(["the", «name», «of»] L , L, Sub, Obj) $\leftarrow$ *name*(*Obj*, *Sub*).

15.10-rasm. Savolga bevosita javob beradigan oddiy grammatika *nounphrase*($L, [], Ind$).

Quyidagi qoida «What large country bordering Paraguay borders Chile?» kabi savollarga javob berishga imkon beradi:

question(["What"] L_0, L_2, Ind) $\leftarrow$
nounphrase(L_0, L_1, Ind) $\wedge$
mp(L_1, L_2, Ind).

Oldingi grammatika to'g'ridan to'g'ri tabiiy til savoliga javob topdi. Bu uslubda savollarga javob berishdagi ikki muammo quyidagilardan iborat:

Dastur tilni tahlil qila olmagan holatlarni tilni tahlil qilishda hech qanday javob yo'q bo'lgan holatdan ajratish qiyin; har ikkala holatda ham javob «yo'q».

Noaniqlik bo'lsa, ya'ni bir nechta qonuniy tahlil va har bir tahlil uchun bir nechta javoblar bo'lsa, u barcha tahlillar uchun barcha javoblarni farqlamasdan beradi.

Bu xususiyatlar ushbu uslubdan foydalanadigan dasturlarni nosozliklarini tuzatishni qiyinlashtiradi. Muqobil variant – bilimlar bazasini so'rashdan oldin tahlil qilish paytida bilimlar bazasini to'g'ridan to'g'ri so'rashdan ko'ra, tabiiy tilning **mantiqiy shaklini** – nutqning ma'nosini bildiruvchi mantiqiy jumlani qurishdir. Semantik shakl boshqa vazifalar uchun, masalan, tizim ma'lumotlarini aytish, tabiiy tilni qayta ko'rib chiqish yoki hatto boshqa tilga tarjima qilish uchun ishlatilishi mumkin.

Siz ot iboralari shaxsni va ot iborasiga qo'yilgan cheklovlar ro'yxatini qaytarishga ruxsat berish orqali so'rovni tuzishingiz mumkin. 15.11-rasmda mos keladigan grammatika qoidalari berilgan va ular 15.12-rasmdagi lug'at bilan birga ishlatiladi.

noun_phrase(L_0, L_1, Ind, C_0, C_1)

L_0 ro'yxati L_1 ro'yxatining oxiri ekanligini va L_1 dan oldingi L_0 dagi so'zlar ot so'z birikmasini hosil qilishini bildiradi. Bu ot so'z birikmasi individual *Ind* ga ishora qiladi. $C_1 - C_0$ ning oxiri va C_0 dagi C_1 dan oldingi formulalar *Ind* shaxsiga ot iborasi qo'ygan cheklovlardir.

Jarayon jihatidan, L_2 tahlil qilinadigan so'zlar ro'yxati, L_1 esa ot iborasidan keyingi qolgan so'zlar ro'yxati. C_1 ot iborasiga kiradigan shartlar ro'yxati va C_0 – bu C_1 ro'yxati bo'lib, unga ot iborasi tomonidan qo'yilgan qo'shimcha shartlar qo'shilgan.

15.37-misol.

So'rov

ask noun_phrase ([*a, Spanish, speaking, country, bordering, Argentina*], [], E_1, C_0 , []).

quyidagicha qaytadi:

C_0 = [*language*(E_1, A), *name*($A, Spanish$), *country*(E_1), *borders*($E_1, argentina$)].

So'rov

ask mp([*the, name, of, the, capital, of, a, country, bordering, Argentina*], [], *Ind, C_0*, []).

quyidagicha qaytadi:

C_0 = [*name*(A, Ind), *capital*(B, A), *country*(B), *borders*($B, argentina$)].

Agar C_0 ro'yxatidagi elementlar ushbu munosabatlar va doimiyliklardan foydalanadigan ma'lumotlar bazasiga qarshi so'rov qilinsa, 15.2-rasmda bo'lgani kabi, ular Argentina bilan chegaradosh ispan tilida so'zlashuvchi mamlakat yoki

Argentina bilan chegaradosh mamlakat poytaxtining nomini qaytarishi mumkin. Prologda ichki oʻrnatilgan predikat *call* bir atomga soʻrov qiladi.

%A noun phrase is a determiner followed by adjectives followed by a noun
%followed by an optional modifying phrase

$nounphrase(L_0, L_4, Ind, C_0, C_4) \leftarrow$
 $det(L_0, L_1, Ind, C_0, C_1) \wedge$
 $adjectives(L_1, L_2, Ind, C_1, C_2) \wedge$
 $noun(L_2, L_3, Ind, C_2, C_3) \wedge$
 $omp(L_3, L_4, Ind, C_3, C_4).$

%Adjectives consist of a sequence of adjectives.

$adjectives(L, L, Ind, C, C).$
 $adjectives(L_0, L_2, Ind, C_0, C_2) \leftarrow$
 $adj(L_0, L_1, Ind, C_0, C_1) \wedge$
 $adjectives(L_1, L_2, Ind, C_1, C_2).$

%A modifying phrase is a relation followed by a noun phrase

$mp(L_0, L_2, Sub, C_0, C_2) \leftarrow$
 $relation(L_0, L_1, Sub, Obj, C_0, C_1) \wedge$
 $noun_phrase(L_1, L_2, Obj, C_1, C_2).$

%An optional modifying phrase is either nothing or a modifying phrase

$omp(L, L, Ind, C, C).$
 $omp(L_0, L_1, Ind, C_0, C_1) \leftarrow mp(L_0, L_1, Ind, C_0, C_1).$

15.11-rasm. Soʻrovni tuzuvchi grammatika qismi

15.7.4. Katta til modellari bilan taqqoslash

Yuqoridagi grammatikalar tabiiy til soʻrovlariga katta til modellarida muhokama qilingan usullardan (8.5.5-mavzu) farqli yondashuvlarni taklif qiladi.

GPT oilasi kabi katta til modellari, masalan, Janubiy Amerikaning geografiasini haqidagi savollarga javob berish uchun ishlatilishi mumkin. Asosiy farqlar quyidagilardan iborat:

GPT modellar keyingi soʻzni ehtimoliy prognoz qiladi. Keyingi soʻz namundan olinishi va tabiiy tilda javob berish uchun bashorat takrorlanishi mumkin. Oldingi boʻlimning grammatikasi toʻgʻridan toʻgʻri savolga javob beradi va keyingi tahlil uchun moʻljallangan ramziy javobni beradi.

Oldingi grammatika barcha javoblarni berishi va javoblar tugagach toʻxtashi mumkin, GPT modellar esa ehtimollik bilan javoblarni taxmin qiladi yoki javoblar taqsimotidan namunalarni oladi.

GPT modellar har bir tahlil uchun har bir javobni, tahlillarni ajratmasdan taqdim etadi, bu 15.10-rasmda koʻrsatilgan savollarga bevosita javob beruvchi grammatika bilan oʻxshash. 15.11-rasmdagi grammatika har bir tahlil uchun alohida

tahlillarni va javoblarni taqdim etishi mumkin, bu ba'zi dasturlar uchun foydali bo'lishi mumkin.

det(L, L, O, C, C).

det([«a»|T], T, O, C, C).

det([«the»|T], T, O, C, C).

noun([«country»|L], L, Ind, [country(Ind)|C], C).

noun([«city»|L], L, Ind, [city(Ind)|C], C).

noun([NL], L, Ind, C, C): -name(Ind, N).

adj([«large»|L], L, Ind, [large(Ind)|C], C).

adj([LangName, «speaking»|L], L, Ind,

[language(Ind, Lang), name(Lang, LangName)|C], C).

reln([«borders»|L], L, Sub, Obj, [borders(Sub, Obj)|C], C).

reln([«bordering»|L], L, Sub, Obj, [borders(Sub, Obj)|C], C).

reln([«next», «to»|L], L, Sub, Obj, [borders(Sub, Obj)|C], C).

reln([«the», «capital», «of»|L], L, Sub, Obj, [capital(Obj, Sub)|C], C).

reln([«the», «name», «of»|L], L, Sub, Obj, [name(Obj, Sub)|C], C).

15.12-rasm. So'rovni tuzish uchun lug'at

Zamonaviy katta til modellari, masalan, SQL yoki Datalogda realistik ko'rinishdagi so'rovlarni yaratishi mumkin. Ma'lumotlar bazasi bilan o'zaro aloqada bo'lish uchun yaratilgan so'rovdan samarali foydalanish uchun til modeli parametrlarni, ularning mavjudligi va qiymatini va boshqalarni aniqlash uchun ma'lumotlar bazasi sxemasiga kirishni talab qiladi, masalan, «Chili» kabi doimiylarning noaniqligini bartaraf etish uchun ma'lumotlar bazasining o'ziga kirishi kerak (bu so'z mamlakatni, futbol jamoasini yoki chili qalampirining yozilishi mumkinligini anglatishi mumkin).

Oldingi grammatika, agar foydalanuvchi tegishli grammatika va lug'atni ishlatmasa, muvaffaqiyatsizlikka uchrashi mumkin, GPT modellar esa tilni ishlatishda juda kechirimlidir.

GPT modellar kengroq qamrovga ega va har qanday mavzuda suhbatlashishi mumkin, oldingi grammatikalar esa har bir soha uchun maxsus ishlab chiqilishi kerak.

Sizning ehtiyojlaringizga, har bir domen uchun tizimni loyihalashga tayyor bo'lishingizga va foydalanuvchilarni qabul qilingan grammatikadan foydalanishga o'rgatish imkoniyatiga qarab, bu usullardan biri foydaliroq bo'lishi mumkin.

15.8. Tenglik

Ba'zida bitta shaxsni nomlash uchun bir nechta atamadan foydalanish foydali bo'lishi mumkin. Masalan, 4^*4 , 2^4 , 273-257 va 16 sonlarni bitta son bilan ifodalash mumkin. Ba'zida har bir nom boshqa shaxsга tegishli bo'lishini xohlaymiz. Masalan, universitetda turli kurslar uchun noyob nomlar kerak bo'lishi mumkin.

Baʼzida ikkita nom bitta shaxsni bildiradimi yoki yoʻqmi, bilmaymiz – masalan, ertalab soat 8:00 yetkazib beruvchi va tushdan keyin soat 13:00 yetkazib beruvchi bitta shaxsmi yoki yoʻqmi.

Bu boʻlim **tenglikning** rolini koʻrib chiqadi, bu bizga ikkita atama dunyoda bitta shaxsni anglatishi yoki anglatmasligini koʻrsatish imkonini beradi. Eʼtibor bering, ushbu bobda ilgari keltirilgan muayyan jumla tilida barcha javoblar atamalar bir xil shaxslarni anglatishi yoki anglatmasligidan qatʼi nazar, barcha javoblar toʻgʻri edi.

Tenglik – bu maʼlum bir maʼnoga ega boʻlgan maxsus predikat belgisi.

$t1$ atamasi $t1 = t2$ deb yozilgan $t2$ atamaga **teng**, agar $t1$ va $t2$ I da bir xil shaxsni bildirsa, I talqinida toʻgʻri boʻladi.

Tenglik oʻxshashlikni anglatmaydi. Agar a va b konstantalar boʻlsa va $a = b$ boʻlsa, bu ikkita oʻxshash yoki hatto bir xil narsa borligini anglatmaydi. Aksincha, bitta narsa ikki nomga ega boʻlishi mumkin.

15.38-misol. 15.13-rasmdagi dunyoni koʻrib chiqing. Ushbu dunyoda $chair1 = chair2$ haqiqat emas, garchi ikkita stul barcha jihatdan bir xil boʻlishi mumkin; stullarning aniq joylashuvini ifodalamasdan, ularni ajratish mumkin emas. Bu $chairOnRight = chair2$ boʻlishi mumkin. Oʻng tomonda joylashgan stul $chair2$ bilan oʻxshash emas. Bu $chair2$.

15.8.1. Tenglikni tasdiqlashga ruxsat berish

Gaplarning boshida tenglikka ruxsat berilmasa, barcha talqinlarda faqat oʻzi-ga teng boʻlgan narsa bu oʻzidir.

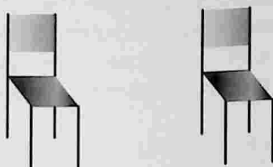
Koʻpincha, ikki atama bitta shaxsni bildirganini tasdiqlash yoki xulosa qilish foydali boʻladi, masalan, $chairOnRight = chair2$. Buning uchun vakillik va mulohaza yuritish tizimi jumla sarlavhalarida tenglikka ega boʻlgan jummalarni oʻz ichiga olgan bilimlar bazasidan kelib chiqadigan xulosalar chiqarishi kerak. Buning uchun ikkita usul mavjud. Birinchisi, boshqa predikatlar kabi tenglikni aksiomatizatsiya qilishdir. Yana biri tenglik uchun maxsus xulosa mashinasini qurishdir.

Agar $t_1 = t_2$ boʻlsa, t_1 ning har qanday paydo boʻlishi t_2 bilan almashtirilishi mumkin. Shuning uchun, tenglikni **qayta yozish qoidasi** sifatida, tenglar uchun tenglarni almashtirish mumkin. Ushbu yondashuv, agar siz har bir shaxs uchun **kanonik koʻrinishni** tanlasangiz yaxshi ishlaydi, yaʼni bu shaxs uchun boshqa tasavvurlarni aks ettirishi mumkin boʻlgan atamadir.

Klassik misollardan biri raqamlarning ifodasidir. Bir xil sonni ifodalovchi koʻplab atamalar mavjud (masalan, $4 * 4$, $13 + 3$, $273 - 257$, 2^4 , $4^2 16$), lekin odatda raqamlar ketma-ketligi (oʻnlik sanoq sistemasida) sonning kanonik ifodasi sifatida ishlatiladi.

Universitetlar har bir talaba uchun kanonik koʻrinishni taʼminlash uchun talabalar raqamlarini ishlab chiqdilar. Bir xil ismga ega boʻlgan turli xil talabalar

ajralib turadi va bir xil shaxsning turli xil ismlarini o'sha shaxsning talaba raqamiga solishtirish mumkin. Har bir qayta yozish qoidasi ta'rifni kanonik ko'rinishga yaqinlashtirishi kerak. Teoremlarni isbotlashda tenglikni qayta yozish qoidalarini sifatida ishlatish **paramodulyatsiya** deb ataladi. Kanonik ko'rinishni aniqlash bo'zan tavsifning **identifikatsiyasini** aniqlash (masalan, ismsiz topshiriq topshirgan talabani aniqlash) deb ham ataladi.



15.13-rasm. Ikki stul

15.8.2. Yagona nomlar taxmini

Har bir termin tengligini e'tiborsiz qoldirib, foydalanuvchidan qaysi ismlar bir xil shaxsni va qaysi biri turli shaxslarni ifodalashini aksiomatlashtirishini kutish o'rniga, turli tayanch terminlar turli shaxslarni ifodalaydi degan qoidaga ega bo'lish osonroq bo'ladi.

15.39-misol. Talabalar ma'lumotlar bazasiga misol keltiramiz, unda talaba ikkita fan bo'yicha tanlov kurslariga ega bo'lishi kerak. Faraz qilaylik, talaba *math302* va *psyc303* ni topshirgan; keyin siz $302 = \text{psyc}303$ ekanligini bil-sangizgina u ikkita kursni o'taganligini bilib olasiz. Ya'ni, *math302* va *psyc303* o'zgarmlari turli kurslarni ifodalaydi. Shunday qilib, qaysi kurs raqamlari tu-kurslarni ifodalashini bilishingiz kerak. n ta shaxs uchun $n * (n-1) / 2$ tengsizl aksiomalarini yozish o'rniga, har bir kurs raqami boshqa kursni bildiradi, degar, konvensiyaga ega bo'lish va shuning uchun tengsizlik aksiomalaridan foydala-nishdan qochish yaxshidir. **Yagona nomlar taxmini** (unique names assumption) bo'yicha, turli tayanch terminlar turli shaxslarni ifodalaydi. Ya'ni har bir turli ta-yanch terminlar juftligi t_1 va t_2 uchun, $t_1 \neq t_2$ faraz qilinadi, bu yerda « $\neq$ » «teng emas» degan ma'noni anglatadi.

Yagona nomlar taxmini (UNA) ma'lumotlar bazasi dasturlarida juda foydali. Siz, masalan, $kim \neq sam$ va $kim \neq chris$ va $chris \neq sam$ va shunga o'xshash har bir subyekti juftligi uchun buni bayon qilishingiz shart emas.

Yagona nomlar taxmini aniq gaplar tili semantikasidan kelib chiqmaydi. Bu semantikaga ko'ra, turli tayanch terminlar t_1 va t_2 bir xil shaxsni ifodalashi yoki turli shaxslarni ifodalashi mumkin edi.

Yagona nomlar taxmini bilan tengsizlik ($\neq$) jumlarlar tanasida bo'lishi mum-kin.

Asosiy terminlar faqat ularni birlashtirish mumkin bo'lmasa, farq qiladi. Bu asosiy bo'lmagan terminlar uchun to'g'ri emas. Masalan, $a \neq X$ ba'zi hollarda to'g'ri bo'ladi – masalan, X ning qiymati b bo'lganida – va bir holatda noto'g'ri bo'ladi, ya'ni X ning qiymati a bo'lganida.

Ba'zida yagona nomlar taxmini noto'g'ri bo'ladi – masalan, $2 + 2 \neq 4$ noto'g'ri va clark kent $\neq$ superman deganda bunday bo'lmashligi mumkin.

Noyob nomlar taxmini uchun yuqoridan pastga isbotlash tartibi

Noyob nomlartaxminini o'z ichiga olgan yuqoridan pastga isbotlash tartibi tengsizlikni faqat boshqa predikat sifatida ko'rmasligi kerak, asosan har bir berilgan shaxs uchun juda ko'p turli shaxslar mavjud bo'lgani uchun.

Agar $t_1 \neq t_2$ shaklidagi kichik maqsad mavjud bo'lsa, t_1 va t_2 terminlari uchun uchta holat mavjud:

1. t_1 va t_2 birlashmaydi. Bu holatda, $t_1 \neq t_2$ muvaffaqiyatli bo'ladi.

Masalan, tengsizlik $f(X, a, g(X)) = f(t(X), X, b)$ muvaffaqiyatli bo'ladi, chunki ikki termin birlashmaydi.

2. t_1 va t_2 bir xil, shu jumladan, bir xil pozitsiyadagi bir xil o'zgaruvchilarga ega bo'lsa. Bu holatda, $t_1 \neq t_2$ muvaffaqiyatsiz bo'ladi.

Masalan, $f(X, a, g(X)) = f(X, a, g(X))$ muvaffaqiyatsiz bo'ladi. E'tibor bering, har qanday *tayanch* terminlar juftligi uchun bu ikkita holatdan biri bo'lishi kerak.

3. Aks holda, $t_1 \neq t_2$ ning muvaffaqiyatli va muvaffaqiyatsiz bo'lgan ba'zi holatlari mavjud. Masalan, $f(W, a, g(Z)) = f(t(X), X, Y)$ kichik maqsadini ko'rib chiqing. $f(W, a, g(Z))$ va $f(t(X), X, Y)$ ning eng umumiy birlashmasi $\{X/a, W/t(a), Y/g(Z)\}$. Tengsizlikning ba'zi misollari, masalan, birlashtiruvchiga mos keladigan asosiy misollar, muvaffaqiyatsiz bo'lishi kerak. Birlashtiruvchi bilan mos bo'lmagan har qanday holat muvaffaqiyatli bo'lishi kerak. Boshqa maqsadlardan farqli o'laroq, siz har bir muvaffaqiyatli misolni sanab o'tishni xohlamaysiz, chunki bu X ni a dan boshqa har bir funksiya va doimiy bilan birlashtirishni va Y va Z uchun har bir juftlik qiymatni sanab o'tishni anglatadi, bu yerda $Y g(Z)$ dan farq qiladi.

Yuqoridan pastga isbotlash tartibi noyob nomlar taxminini qo'shish uchun kengaytirilishi mumkin. Birinchi turdagi tengsizlik muvaffaqiyatli va ikkinchi turdagi tengsizliklar muvaffaqiyatsiz bo'lishi mumkin. Uchinchi turdagi tengsizlik **kechiktirilishi** mumkin, keyingi maqsadlar o'zgaruvchilarni birlashtirish uchun bu ikki holatdan biri sodir bo'lguncha kutib turadi. 15.5-rasmdagi isbotlash tartibidagi maqsadni kechiktirish uchun, javob gapining G tanasida bir atomni tanlashda, algoritmi kechiktirilayotgan atomlardan birini tanlashi kerak. Agar tanlash uchun boshqa atomlar bo'lmasa va birinchi ikkita holat qo'llanilmasa, so'rov muvaffaqiyatli bo'lishi kerak. Har doim muvaffaqiyatli bo'lgan tengsizlik misoli mavjud, ya'ni har bir o'zgaruvchiga boshqa joyda paydo bo'lmaydigan boshqa o'zgarmas qiymat beriladigan holat.

Bu sodir bo'lganda, foydalanuvchi javobda erkin o'zgaruvchilarni talqin qilishda ehtiyoat bo'lishi kerak. Javob har bir erkin o'zgaruvchining misoli uchun to'g'ri degani emas, balki ba'zi bir misol uchun to'g'ri ekanligini anglatadi.

15.40-misol. Talaba kamida ikkita kursdan o'tganligini aniqlovchi qoidalarni ko'rib chiqing:

passed_two_courses(S) ←

C₁ = C₂ ∧

passed(S, C₁) ∧

passed(S, C₂), passed(S, C) ←

grade(S, C, M) ∧

M ≥ 50.

grade(sam, engl101, 87).

grade(sam, phys101, 89).

ask_passed_two_courses(sam) so'rovi uchun kichik maqsad $C_1 \neq C_2$ aniqlanmasligi mumkin va shuning uchun kechiktirilishi kerak. Yuqoridan pastga isbotlash tartibi, buning o'rniga, *passed(sam, C₁)* ni tanlashi mumkin, bu C_1 ni *engl101* ga bog'laydi. Keyin u *passed(sam, C₂)* ni chaqirishi mumkin, bu o'z navbatida *grade(sam, C₂, M)* ni chaqiradi, bu esa $\{C_2/\text{phys101}, M/89\}$ o'rnini bosish bilan muvaffaqiyatli bo'lishi mumkin. Bu bosqichda, kechiktirilgan tengsizlikning o'zgaruvchilari tenglik muvaffaqiyatsiz bo'lishi kerakligini aniqlash uchun yetarli darajada bog'langan.

Boshqa qoidalar *grade(sam, C₂, M)* uchun tanlanishi mumkin, bu esa $\{C_2/\text{phys101}, M/89\}$ o'rnini bosadi. Kechiktirilgan tenglik o'zgaruvchilari tenglikni tekshirish uchun yetarlicha bog'langan va bu safar tenglik muvaffaqiyatli bo'ladi. Keyin u $89 > 50$ ekanligini isbotlashda davom etishi mumkin va maqsad muvaffaqiyatli bo'ladi.

Ushbu misoldan kelib chiqadigan savollardan biri: «nega tengsizlikni oxirgi yechimga aylantirmaslik kerak, chunki keyin kechiktirishga hojat qolmaydimi?» bo'lishi mumkin. Buning ikkita sababi bor. Birinchidan, kechiktirish samaraliroq bo'lishi mumkin. Ushbu misolda kechiktirilgan tengsizlikni $87 > 50$ ekanligini tekshirishdan oldin sinab ko'rish mumkin. Ushbu o'ziga xos tengsizlikni tekshirish tez bo'lishi mumkin bo'lsa-da, ko'p hollarda, imkon qadar tezroq buzilgan tengsizliklarni payqash orqali muhim hisob-darsliklarni chetlab o'tish mumkin.

Ikkinchidan, agar subisbot qiymatni bog'lashdan oldin qaytarsa, isbot tartibi tenglik cheklovini eslab qolishi kerak, shunda har qanday kelajakdagi birlashtirish cheklovni buzgan holda muvaffaqiyatsiz bo'lishi mumkin.

15.9. To'liq bilimlar taxmini

To'liq bilimlar taxmini, 5.7-mavzuda muhokama qilinganidek, bilimlar bazasidan kelib chiqmaydigan har qanday bayonot noto'g'ri degan taxmindir. Shuningdek, **rad etish orqali muvaffaqiyatsizlikni** isbotlashga imkon beradi.

O'zgaruvchilar va funksiya belgilariga ega mantiqiy dasturlar uchun to'liq bilimlar taxmini tenglik uchun aksiomalarni, domenning yopilishi va to'ldirishning yanada murakkab tushunchasini talab qiladi.

15.41-misol. *Talaba* munosabati quyidagicha aniqlangan deb faraz qiling:

student(huan).

student(manpreet).

student(karan).

To'liq bilim haqidagi taxmin shuni ko'rsatadiki, bu uchasi yagona talabalar-dir:

$student(X) \leftrightarrow X = huan \vee X = manpreet \vee X = karan.$

Ya'ni agar *X*, *huan*, *manpreet* yoki *karan* bo'lsa, *X* – talaba, agar *X* – talaba bo'lsa, *X* bu uchadan biri bo'lishi kerak. Xususan, *kim* talaba emas.

$\neg student(kim)$ ni yakunlash uchun $kim \neq huan \wedge kim \neq manpreet \wedge kim \neq karan$ ni isbotlash kerak. Tengsizliklarni hosil qilish uchun yagona nomlar taxmini talab qilinadi.

To'liq bilimlar taxmini noyob nomlar taxminini o'z ichiga oladi.

Klark normal shakli qoidasi:

$p(t_1, \dots, t_k) \leftarrow B.$

quyidagicha:

$p(V_1, \dots, V_k) \leftarrow \exists W_1 \dots \exists W_m V_1 = t_1 \wedge \dots \wedge V_k = t_k \wedge B.$

bu yerda $V_1, \dots, V_k$ dastlabki jumlada ko'rinmagan k o'zgaruvchilar va $W_1, \dots, W_m$ esa jumladagi dastlabki o'zgaruvchilar. « $\exists$ » «mavjud» degan ma'noni anglatadi.

Jumlalar atomik bo'lsa, *B* rost bo'ladi.

Faraz qilaylik, *p* uchun barcha jumlarlar Klarkning normal shakliga kiritilgan va bir xil kiritilgan o'zgaruvchilar to'plami mavjud bo'lsa, quyidagi natijalarni beradi:

$p(V_1, \dots, V_k) \leftarrow B_1.$

$p(V_1, \dots, V_k) \leftarrow B_n.$

bu quyidagiga teng:

$p(V_1, \dots, V_k) \leftarrow B_1 \vee \dots \vee B_n.$

Ushbu implikasiya dastlabki jumlarlar to'plamiga mantiqiy jihatdan ekvivalentdir.

Clakning tugallanishi predikat *p* ni quyidagi ekvivalentlikda ifodalaydi:

$\forall V_1 \dots \forall V_k p(V_1, \dots, V_k) \leftrightarrow B_1 \vee \dots \vee B_n$

bu yerda tanalarda **rad etish orqali muvaffaqiyatsizlik** ($\sim$) standart mantiqiy inkor qilish ($\neg$) bilan almashtiriladi. Bu to'ldiruv shuni anglatadiki, $p(V_1, \dots, V_k)$ haqiqiy bo'ladi, agar kamida bitta B_i tana haqiqiy bo'lsa. **Klarkning** bilim bazasi- ni **to'ldirishi** har bir predikat belgining to'ldirilishini o'z ichiga oladi.

15.42-misol. Quyidagi gaplar uchun:

student(huan).

student(manpreet).

student(karan).

the Clark normal form is

student(V) $\leftarrow V = \text{huan}.$

student(V) $\leftarrow V = \text{manpreet}.$

student(V) $\leftarrow V = \text{karan}.$

which is equivalent to

student(V) $\leftarrow V = \text{huan} \vee V = \text{manpreet} \vee V = \text{karan}.$

The completion of the *student* predicate is

$\forall V \text{student}(V) \leftrightarrow V = \text{huan} \vee V = \text{manpreet} \vee V = \text{karan}$

Klarkning normal shakli:

bu quyidagiga teng:

talaba predikatining to'ldirilishi:

15.43-misol. Quyidagi rekursiv ta'rifni ko'rib chiqing:

Passed_each([], St, MinPass).

Passed_each([C|R], St, MinPass) $\leftarrow$

passed(St, C, MinPass) $\wedge$

passedeach(R, St, MinPass).

Klarkning normal shaklida, o'zgaruvchilarni qayta nomlash bilan bu quyidagicha yozilishi mumkin:

Passed_each(L, S, M) $\leftarrow L = [].$

passed_each(L, S, M) $\leftarrow$

$\exists C \exists R (L = [C|R] \wedge$

passed(S, C, M) $\wedge$ passed_each(R, S, M).

passed_each uchun Klarkning to'ldirilishi quyidagicha ifodalanadi:

$\forall L \forall S \forall M \text{passed_each}(L, S, M) \leftrightarrow L = [] \vee$

$\exists C \exists R (L = [C|R] \wedge$

passed(S, C, M) $\wedge$

passedeach(R, S,

To'liq bilim taxminiga ko'ra, endi faqat aniq bo'laklar yordamida aniqlab bo'lmaydigan munosabatlarni aniqlash mumkin.

15.44-misol. Tasavvur qiling, sizga *course(C)* ma'lumotlar bazasi berilgan, bu *C* kurs ekanligini bildiradi va *enrolled(S, C)* esa *S* talabasi *C* kursiga yozilganligini bildiradi. To'liq bilim taxminisiz *empty course(C)* ni aniqlay olmaysiz, bu

C kursiga hech kim yozilmagan bo'lsa haqiqatdir. Chunki ma'lumotlar bazasining har doim har bir kursda kimdir yozilgan bo'lgan modeli mavjud.

Rad etish orqali muvaffaqiyatsizlikni qo'llagan holda, *empty course*(C) quyidagicha aniqlanishi mumkin:

$$\text{empty_course}(C) \leftarrow \text{course}(C) \wedge \sim \text{has_enrollment}(C). \text{has_enrollment}(C) \leftarrow \text{enrolled}(S, C)$$

Buning to'ldirilishi:

$$\forall \text{Emptycourse}(C) \leftrightarrow \text{course}(C) \wedge \sim \text{hasenrollment}(C)$$

$$\forall \text{Chasenrollment}(C) \leftrightarrow \exists \text{Senrolled}(S, C)$$

Ogohlantirish sifatida, muvaffaqiyatsizlik sifatidagi inkorga erkin o'zgaruvchilarni kiritganingizda juda ehtiyot bo'lishingiz kerak. Ular odatda siz o'ylagan narsani anglatmaydi. *has_enrollment* predikati oldingi misolda muvaffaqiyatsizlik sifatidagi inkorga erkin o'zgaruvchini qo'shmaslik uchun kiritilgan (15.15-mashqqa qarang).

15.9.1. To'liq bilim taxminining isbot jarayonlari

Yuqoridan pastga isbotlash tartibi o'zgaruvchilar va funksiyalar bilan muvaffaqiyatsizlik sifatidagi inkor uchun, taklifli muvaffaqiyatsizlik sifatidagi inkor uchun yuqoridan pastga isbotlash tartibiga juda o'xshaydi. Noyob nomlarni taxmin qilishda bo'lgani kabi, rad etilgan maqsadlar uchun erkin o'zgaruvchilar mavjud bo'lganda muammo yuzaga keladi.

15.45-misol. Quyidagi jumalarga e'tibor bering:

$$p(X) \leftarrow \sim q(X) \wedge r(X).$$

$$q(a).$$

$$q(b).$$

$$r(d).$$

Semantika bo'yicha ask $p(X)$ so'rovi uchun faqat bitta javob bor, bu $X = d$. $r(d)$ ga ko'ra, $\sim q(d)$ ham rost bo'ladi va $p(d)$ bilim bazasidan mantiqiy ravishda kelib chiqadi. Yuqoridan pastga isbotlash tartibi $\sim q(X)$ ga duch kelganda, $q(X)$ ni isbotlashga urinmasligi kerak, bu (X/a almashtirish bilan) muvaffaqiyatli bo'ladi. Bu $p(X)$ maqsadini muvaffaqiyatsiz qiladi, holbuki u $X = d$ bilan muvaffaqiyatli bo'lishi kerak. Shunday qilib, isbot tartibi to'liq bo'lmaydi. Agar bilim bazasi $s(X) \leftarrow \sim q(X)$ ni o'z ichiga olgan bo'lsa, $q(X)$ ning muvaffaqiyatsizligi $s(X)$ ning muvaffaqiyatli bo'lishini anglatadi. Shunday qilib, muvaffaqiyatsizlik sifatida inkor qilish bilan, to'liqsizlik asossizlikka olib keladi.

Noyob nomlar taxminidagi kabi (15.8.2-mavzu), to'g'ri isbot tartibi inkor qilingan kichik maqsadni bo'sh o'zgaruvchi bog'langunga qadar kechiktirishi kerak.

Muvaffaqiyatsizlik sifatida inkor qilish uchun chaqiruvlar bo'lganida, murakkabroq yuqoridan pastga tartib talab qilinadi:

5.12-rasmdagi inkor etish tartib-qoidalarida erkin o'zgaruvchilarni o'z ichiga olgan muvaffaqiyatsizlik sifatida rad etish maqsadlari o'zgaruvchilar bog'lanmagancha tanlanmasligi kerak.

Agar o'zgaruvchilar hech qachon bog'lanmasa, maqsad muvaffaqiyatsiz bo'ladi. Bunday holda, maqsad haqida hech qanday xulosa qilolmaysiz. Quyidagi misol muvaffaqiyatsiz bo'lgan maqsadlar holatida nima qilish kerakligini ko'rsatadi.

15.46-misol. Quyidagi gaplarni ko'rib chiqing:

$$p(X) \leftarrow \sim q(X)$$

$$q(X) \leftarrow \sim r(X)$$

$$r(a)$$

va so'rovini

$$\text{ask } p(X).$$

Bilim bazasining to'ldirilishi:

$$p(X) \leftrightarrow \sim q(X)$$

$$q(X) \leftrightarrow \sim r(X)$$

$$r(X) \leftrightarrow X = a.$$

$X = a$ ni r o'rni qo'yish $q(X) \leftrightarrow \sim X = a$ ni beradi va shuning uchun $p(X) \leftrightarrow X = a$ bo'ladi. Shunday qilib, bitta javob bor, ya'ni $X = a$, lekin maqsadni kechiktirish uni topishga yordam bermaydi. Isbot protsedurasida maqsad muvaffaqiyatsiz bo'lgan holatlarni tahlil qilishi kerak. Biroq, bunday tartib ushbu darslikning doirasidan tashqarida.

15.10. Ijtimoiy ta'sir

Mantiqiy dasturlashning ko'plab qo'llanmalari mavjud. Bu Java virtual mashinasining tur tekshiruvchisida qo'llaniladi: «Tur tekshiruvchisi Prolog yordamida belgilangan tur qoidalarini amalga oshiradi. Ingliz tili matni natijalar tarzida tur qoidalarini tavsiflash uchun ishlatiladi, Prolog qoidalari esa rasmiy spesifikasiyani ta'minlaydi» [Lindholm va boshqalar, 2022]. Prolog IBM Watson tizimida savollarni tahlil qilish uchun ishlatilgan, bu tizim «Jeopardy!» televizion viktorina shousida inson chempionini mag'lub etdi [Lally va boshqalar, 2012].

Robot olim g'oyasi [King va boshqalar, 2004, 2009a; Sparkes va boshqalar, 2010] kompyuter farazlarni yaratishi va sinovdan o'tkazishi haqida. Robot olim **Adam** [King va boshqalar, 2009b] avtomatik ravishda achitqi haqida funksional genomika farazlarini yaratadi va farazlarni laboratoriyani avtomatlashtiradigan fizik robot yordamida sinaydi. Farazlar mantiqiy dastur shaklida ifodalanadi. Mantiqiy dastur shaklidagi faraz uni izohlash imkonini beradi va uning natijalari sinovdan o'tkazilishi mumkin. Robot faraz yaratadi, farazni sinash uchun eksperiment o'tkazadi, natijalarni izohlaydi va jarayonni takrorlaydi.

15.11. Takrorlash

Ushbu bobdan o'rganishingiz kerak bo'lgan asosiy fikrlar quyidagilardir:

Shaxslar va munosabatlar bilan tavsiflangan domenlarda, shaxslarni ifodalovchi konstantalar va munosabatlarni ifodalovchi predikat belgilaridan foydalangan holda, domen ichida nima rost ekanligini aniqlash mumkin.

Datalog konstantalar, universal miqdorli o'zgaruvchilar, munosabatlar va qoidalariga ega mantiqiy tildir.

Almashtirishlar atomlar va qoidalar holatlarini yaratish uchun ishlatiladi. Bir-lashtirish atomlarni isbotlashda foydalanish uchun bir xil qiladi.

Funksional belgilardan boshqa shaxslar asosida tasvirlangan cheksiz shaxs-lar to'plamini ifodalash uchun foydalaniladi. Funksional belgilardan ma'lumotlar strukturalarini yaratish uchun foydalanish mumkin.

Aniq-murakkab grammatikalar tabiiy til interfeyslari orqali ma'lumotlar ba-zasiga ega bo'lgan tilni nazorat qilish mumkin bo'lgan holatlarda moslashuvchan tabiiy tilni qayta ishlash uchun ishlatilishi mumkin.

Atamalar orasidagi tenglik, atamalar bir xil shaxsni ifodalashini anglatadi.

Klarkning to'liqlik farazi muvaffaqiyatsizlik sifatida inkor qilish semantikasi-ni belgilash uchun ishlatilishi mumkin.

Mantiqiy dasturlash xususiyatlar, tahlillar va ilmiy farazlarni yaratish uchun foydalidir.

15.12. Qo'shimcha o'qish uchun manbalar

Datalog va mantiqiy dasturlar haqida Kovalski [2014], Sterling va Shapiro [1994] va Garsia-Molina va boshqalar [2009] tomonidan tasvirlangan. Mantiqiy dasturlashning tarixi Kovalski [1988] va Colmerauer va Roussel [1996] tomoni-dan tasvirlangan. Muvaffaqiyatsizlik sifatida inkor qilish va noyob nomlar fara-zi Klarkning [1978] ishlariga asoslanadi. Mantiqiy dasturlash haqida umumiy va muvaffaqiyatsizlik sifatida inkor qilish haqida maxsus munosabatlar uchun Lloyd [1987]ning darslikiga qarang. Apt va Bol [1994] muvaffaqiyatsizlik sifatida inkor qilishni boshqarishning turli texnikalari bo'yicha tadqiqotlarni taqdim etadi. Juraf-ski va Martin [2023] kompyuter lingvistikasi bo'yicha mukammal kirish bo'limini taqdim etadi. Aniq qoidalar yordamida tabiiy tilni tasvirlash Dahl [1994] va Pere-ira va Shieber [2002] tomonidan tasvirlangan.

15.13. Mashqlar

15.1-mashq. Ikki shaxs ($\otimes$ and $\oplus$), ikki predikat belgisi (p va q) va uchta konstantaga ega bo'lgan domenni ko'rib chiqing (a , b va c). Bilim bazasi KB quyidagicha belgilanadi:

$$p(X) \leftarrow q(X).$$

$$q(a).$$

- (a) KB modeliga mos keladigan bitta talqinni ko'rsating.
 (b) KB modeliga mos kelmaydigan bitta talqinni ko'rsating.
 (c) Nechta talqin mavjud? Javobingiz uchun qisqacha asos keltiring.
 (d) Ushbu talqinlardan nechitasi KB modellari hisoblanadi? Javobingiz uchun qisqacha asos keltiring.

15.2-mashq. Til a , b va c konstantalariga, p va q predikat belgilariga va hech qanday funksiyalarga ega emas deb faraz qiling. Quyidagi bilim bazalari ushbu til asosida qurilgan:

KB_1	KB_2	KB_3
$p(a).$	$p(X) \leftarrow q(X).$	$p(X) \leftarrow q(X).$ $q(b).$

Til uchun mumkin bo'lgan talqinlarni $I=(D, \pi, \varphi)$ shaklida ko'rib chiqing, bu yerda $D = \{ \leftarrow, \wedge, \rightarrow, \neg \}$.

(a) Bizning oddiy tilimiz uchun to'rt domen elementi bilan nechta talqin mavjud? Javobingiz uchun qisqacha asos keltiring. [Maslahat: Konstantalar uchun mumkin bo'lgan φ tayinlovlari sonini aniqlang. Predikatlar p va q ning kengaytmalari nechta bo'lishi mumkinligini ko'rib chiqing va π tayinlovlari nechta bo'lishi mumkinligini aniqlang.] Barcha mumkin bo'lgan talqinlarni sanab o'tishga urinmang.

(b) Yuqorida keltirilgan talqinlardan nechitasi KB_1 modellari hisoblanadi? Javobingiz uchun qisqacha asos keltiring.

(c) Yuqorida keltirilgan talqinlardan nechitasi KB_2 modellari hisoblanadi? Javobingiz uchun qisqacha asos keltiring.

(d) Yuqorida keltirilgan talqinlardan nechitasi KB_3 modellari hisoblanadi? Javobingiz uchun qisqacha asos keltiring.

15.3-mashq. Quyidagi bilim bazasini ko'rib chiqing:

$$r(a). \quad r(e). \quad p(c).$$

$$q(b). \quad s(a,b). \quad s(d,b).$$

$$s(e,d). \quad p(X) \leftarrow q(X) \wedge r(X). \quad q(X) \leftarrow s(X,Y) \wedge q(Y).$$

Ushbu bilim bazasidan olinadigan asosiy atom natijalarini ko'rsating. Har bir iteratsiyada birinchi bo'lib tanlanadigan mos keladigan qoidani taxmin qilgan holda pastdan yuqoriga isbot tartibidan foydalaning. Shuningdek, agar bitta qoida uchun bir nechta mos keladigan konstantalar mavjud bo'lsa, «alifbo tartibida» tanlangan mos keladigan konstantalarni taxmin qiling; masalan, agar X/a va X/b

bir qoida uchun ba'zi iteratsiyalarda mos keladigan bo'lsa, birinchi navbatda $q(a)$ natijasini hosil qiling. Natijalar qaysi tartibda olinadi?

15.4-mashq. Quyidagi bilim bazasini ko'rib chiqing:

$has_access(X, library) \leftarrow student(X).$

$has_access(X, library) \leftarrow faculty(X).$

$has_access(X, library) \leftarrow has_access(Y, library) \wedge parent(Y, X).$

$has_access(X, office) \leftarrow has_keys(X).$

$faculty(diane), faculty(ming), student(william).$

$student(mary), parent(diane, karen), parent(diane, robyn).$

$parent(susan, sarah), parent(sarah, ariel), parent(karen, chelsey).$

$parent(karen, todd).$

(a) $has_access(todd, library)$ so'rovi uchun SLD hosila taqdim eting (15.6-rasmga o'xshash).

(b) $has_access(mary, library)$ so'rovi uchun ikkita SLD hosila mavjud. Ikkalasini ham ko'rsating, lekin tanlangan qoidalar yoki almashtirishlarni ko'rsatmang.

(c) $has_access(ariel, library)$ so'rovi uchun hosila mavjudmi? Nima uchun bor yoki yo'qligini tushuntiring.

(d) $has_access(X, office)$ so'roviga javoblar to'plami nima uchun bo'sh ekanligini tushuntiring.

(e) Quyidagi jumla bilim bazasiga qo'shilgan deb faraz qiling:

$has_keys(X) \leftarrow faculty(X)$

$has_access(X, office)$ so'rovi uchun javoblar qanday?

15.5-mashq. Quyidagi almashtirishlarning natijasi qanday?

(a) $f(A, X, Y, X, Y) \{A/X, Z/b, Y/c\}.$

(b) $yes(F, L) \leftarrow append(F, c(L, nil), c(l, c(i, c(s, c(t, nil))))))$

$\{F/c(l, X1), Y1/c(L, nil), A1/l, Z1/c(i, c(s, c(t, nil)))\}.$

(c) $append(c(A1, X1), Y1, c(A1, Z1)) \leftarrow append(X1, Y1, Z1)$

$\{F/c(l, X1), Y1/c(L, nil), A1/l, Z1/c(i, c(s, c(t, nil)))\}.$

15.6-mashq. Quyidagi ifodalarning juftlari uchun eng umumiy birlashtiruvchini keltiring:

$p(f(X), g(g(b)))$ and $p(Z, g(Y))$

$g(f(X), r(X), t)$ and $g(W, r(Q), Q)$

$bar(val(X, bb), Z)$ and $bar(P, P)$

15.7-mashq. Quyidagi atomlar juftlari uchun eng umumiy birlashtiruvchini bering yoki uning mavjud emasligini tushuntiring:

(a) $p(X, Y, a, b, W)$ and $p(E, c, F, G, F)$

(b) $p(Y, a, b, Y)$ and $p(c, F, G, F)$

(c) $foo(Z, [a, z/X], X)$ and $foo([a, m/W], W, [i, n, g])$

(d) $ap(F0, c(b, c(B0, L0)), c(a, c(b, c(a, emp))))$ and $ap(c(H1, T1), L1, c(H1, R1)).$

15.8-mashq. Quyidagi bilimlar bazasidan barcha asosiy atomik mantiqiy natijalarni sanab chiqing:

$q(Y) \leftarrow s(Y,Z) \wedge r(Z). p(X) \leftarrow q(f(X)). s(f(a), b).$
 $s(f(b), b). s(c, b). r(b).$

15.9-mashq. Quyidagi mantiqiy dastur haqida o'ylang:

$rd(cons(H, cons(H, T)), T).$
 $rd(cons(H, T), cons(H, R)) \leftarrow rd(T, R).$

Quyidagi so'rov uchun yuqoridan pastga qarab xulosani, barcha almashtirishlarni ko'rsating

$ask\ rd(cons(a, cons(cons(a, X), cons(B, cons(c, Z))))), W).$

Ushbu xulosa bilan bog'liq bo'lgan javob nima? Ikkinchi javob bormi? Ha bo'lsa, xulosani ko'rsating; bo'lmasa nima uchun yo'qligini tushuntiring.

15.10-mashq. Quyidagi mantiqiy dastur haqida o'ylang:

$ap(emp, L, L).$
 $ap(c(H, T), L, c(H, R)) \leftarrow ap(T, L, R).$
 $adj(A, B, L) \leftarrow ap(F, c(A, c(B, E)), L).$

(a) Quyidagi so'rov uchun bitta javob berish uchun yuqoridan pastga qarab (barcha almashtirishlarni o'z ichiga olgan holda) xulosani keltiring

$ask\ adj(b, Y, c(a, c(b, c(b, c(a, emp))))).$

(b) Boshqa javoblar bormi? Agar shunday bo'lsa, oldingi javobdagi xulosada boshqa tanlov qayerda amalga oshirilishi mumkinligini tushuntiring va xulosani davom ettiring, yana bir javobni ko'rsating. Agar boshqa javoblar bo'lmasa, nima uchun yo'qligini tushuntiring. [Siz bu mashqni kompyuterdek qilishingiz kerak, belgilarning nima anglatishini bilmadan. Agar siz bu dasturni anglatmoqchi bo'lsangiz, ap ni $append$, c ni $cons$, emp ni $empty$ va adj ni $adjacent$ deb o'qiysiz.]

15.11-mashq. Ushbu savolning maqsadi oddiy mantiqiy dasturlarni yozish amaliyotini olishdir.

(a) Agar R, L ro'yxatidagi E elementining bitta holatini olib tashlash natijasida hosil bo'lgan ro'yxat bo'lsa, $remove(E, L, R)$ munosabatini yozing. Agar E, L tarkibida bo'lmasa, munosabat noto'g'ri bo'ladi.

(b) Quyidagi so'rovlarga barcha javoblarni keltiring:

$ask\ remove(a, [b, a, d, a], R).$
 $ask\ remove(E, [b, a, d, a], R).$
 $ask\ remove(E, L, [b, a, d]).$
 $ask\ remove(p(X), [a, p(a), p(p(a)), p(p(p(a)))], R).$

(c) L_1 ro'yxatida L_2 elementlarining kichik to'plami bir xil tartibda bo'lsa, rost bo'ladigan $subsequence(L_1, L_2)$ munosabatini yozing.

(d) Quyidagi so'rovlardan har biri uchun necha xil dalillar bor:

$ask\ subsequence([a, d], [b, a, d, a]).$
 $ask\ subsequence([b, a], [b, a, d, a]).$
 $ask\ subsequence([X, Y], [b, a, d, a]).$

ask *subsequence*($S, [b, a, d, a]$).

Nima uchun shuncha ekanligini tushuntiring.

15.12-mashq. Ushbu savolda siz maxsus video taqdimotlarni loyihalash uchun ma'lum bir ma'lumot bazasini yozishingiz kerak.

Videoqa quyidagi munosabat yordamida izoh berilgan deb taxmin qiling
segment(*SegId*, *Duration*, *Covers*)

Bu yerda *SegId* segment uchun identifikator hisoblanadi. (Haqiqiy dasturda bu video segmentini olish uchun yetarli ma'lumot bo'ladi.) *Duration* segmentning davomiyligi (sekundlarda). *Covers* videosegmenti qamrab olgan mavzular ro'yxatidir. Video annotatsiyasining bir misoli – ma'lumotlar bazasi:

segment(*seg0*, 10, [*welcome*]).

segment(*seg1*, 30, [*skiing*, *views*]).

segment(*seg2*, 50, [*welcome*, *artificialintelligence*, *robots*]).

segment(*seg3*, 40, [*graphics*, *dragons*]).

segment(*seg4*, 50, [*skiing*, *robots*]).

Taqdimot segmentlar ketma-ketligidir. Taqdimotni segment identifikatorlari ro'yxati bilan ifodalang.

Predikatni aksiomatizatsiya qiling.

presentation(*MustCover*, *Maxtime*, *Segments*)

Agar *Segments*ning jami davomiyligi *Maxtime* soniyadan kam yoki unga teng bo'lsa va *MustCover* ro'yxatidagi barcha mavzular taqdimotdagi segmentlar tomonidan qamrab olingan bo'lsa, bu *Segments* taqdimoti to'g'ri bo'ladi. Ushbu predikatning maqsadi ma'lum bir vaqt oralig'ida ma'lum miqdordagi mavzularni qamrab oladigan taqdimotlarni loyihalashdir.

Masalan, quyidagi so'rov

ask *presentation*([*welcome*, *skiing*, *robots*], 90, *Segs*)

kamida quyidagi ikki javobni qaytarishi kerak (segmentlar boshqa tartibda bo'lishi mumkin):

presentation([*welcome*, *skiing*, *robots*], 90, [*seg0*, *seg4*])

presentation([*welcome*, *skiing*, *robots*], 90, [*seg2*, *seg1*]).

Ishlatilgan barcha belgilarni izohlang va AIPython (aipython.org) yoki Prologda aksiomangizni sinab ko'rganingizni (shu jumladan, so'rovingizga barcha javoblarni topish) ko'rsating. Har bir javob nima uchun javob ekanligini qisqacha tushuntiring.

(b) Agar sizda yaxshi foydalanuvchi interfeysi va taqdimotlarni haqiqatda ko'rish usuli bor deb hisoblasangiz, avvalgi dastur qilmagan *uchta* narsani sanab bering. (Bu qism uchun to'g'ri javob yo'q. To'liq ball olish uchun ijodiy bo'lishingiz kerak.)

15.13-mashq. Noaniq jumlar bilan grammatikadagi qo'shimcha dalillar uni kontekstsiz grammatikaga qaraganda qat'iy kuchliroq qiladi. $\{a^n b^n c^n \mid n > 0\}$ tili, ya'ni bir nechta a lar, keyin bir xil sonli b lar va undan keyin bir xil sonli c lardan

tashkil topgan gaplarni kontekstsiz grammatika bilan aniqlab bo'lmaydi. Bu tilni aniq jumla grammatikasidan foydalanib aniqlang. [Maslahat: Argumentlaridan birining n nusxasini yaratadigan noterminal uchun *copies* predikatini aniqlang va nol uchun 0 va n dan keyingi raqamlar uchun $s(N)$ dan foydalangan holda raqamlarni ifodalang.]

15.14-mashq. Geografik savollarga javob berish uchun 15.12-rasmga asoslangan bilimlar bazasi va lug'at yarating, masalan, 1.3-rasmga berilgan savollarga. Har bir so'rov uchun unga qanday javob berish mumkinligini ko'rsating yoki ushbu bobda taqdim etilgan vositalar bilan javob berish qiyin bo'lganligini tushuntiring.

15.15-mashq. 15.44-misolda nima bo'lishini ko'rib chiqing, agar *empty_course* quyidagicha aniqlangan bo'lsa:

$empty_course(C) \leftarrow course(C) \wedge \sim enrolled(S, C).$

Bilimlar bazasining qolgan qismi quyidagicha bo'lsa:

$course(cs422).$ $course(cs486).$ $course(cs987).$

$enrolled(huan, cs422).$ $enrolled(sally, cs486).$

(a) *empty_course* uchun Klarkning to'liqligi qanday?

(b) Qaysi qarama-qarshi misol to'ldirishning to'g'riligini rad etadi? Tana rost, boshi yolg'on bo'lgan gapga misol keltiring.

(c) Muvaffaqiyatsizlik bilan inkor qilingan dastur (prolog kabi) nimani beradi?

$emptycourse(C) \leftarrow \sim enrolled(S, C) \wedge course(C).$

bilan taqqoslang.

16-BOB. BILIM GRAFIKLARI VA ONTOLOGIYALAR

Hisoblash nazariyasini rivojlantirishda eng jiddiy muammolar ontologik jihatdan ham, semantik jihatdan ham mavjud. Semantik muammolar yo 'qolayotgani yo 'q; ular har doimgidek qiyin bo'lib qolmoqda. Ular faqat bir joyda, go'yoki sahnaning markazida, yanada talabchan ontologik muammolar bilan birlashadi.

– Brayan Kantvell Smith [1996, 14-b.]

Dunyodagi bilimlarni qanday qilib ko'rsatish kerakki, bu bilimlarni osonlik bilan olish, xatolarini tuzatish, saqlash, aloqa qilish, baham ko'rish va ular bilan mantiqiy fikr yuritish mumkin bo'lsin? Ushbu bob faktlar, bilim va ma'lumotlarni saqlash va ulardan foydalanish uchun mos usullarni o'rganadi, shuningdek, ontologiyalar yordamida bilim va ma'lumot almashish imkoniyatlarini ko'rib chiqadi. Smit ta'kidlaganidek, ontologiyaning muammolari aqlli hisoblash agentlarini yaratishda markaziy ahamiyatga ega.

16.1. Bilim grafiklari

16.1.1. Tripletlar (uchliklar)

Mantiqiy ko'rsatish tili, masalan, avvalgi bobda ishlab chiqilgan va fikr yuritish uchun bir dunyo berilganda, ma'lumotlar bazalari va bilim bazalarini loyihalovchilar qaysi shaxslar va aloqalarni ko'rsatishni tanlashlari kerak. Modelchilar dunyoda mavjud shaxslar va aloqalarga murojaat qilishlari mumkin deb o'ylash mumkin. Biroq, dunyo qaysi shaxslar borligini aniqlamaydi. Dunyoni shaxslar bo'yicha qanday taqsimlash, dunyoni model qilayotgan odam tomonidan ixtiro qilinadi. Modelchi dunyoni narsalarga bo'lib, agentning ma'lum vazifa uchun ma'noli qismga murojaat qilishini ta'minlaydi.

16.1-misol. «Qizil» narsa dunyodagi narsalarga xos xususiyat sifatida ko'rsatilishi mantiqiy ko'rinadi. Siz buni yetkazib berish robotiga qizil qadoqni olib kelishini aytish uchun qilishingiz mumkin. Dunyoda ba'zi chastotalarni so'raydigan va boshqa chastotalarni aks ettiruvchi yuzalar mavjud. Ba'zi foydalanuvchi ma'lum bir dastur uchun aks ettirish xususiyatlarining ma'lum to'plamini qizil deb atashga qaror qilishi mumkin. Boshqa bir modelchi spektrni boshqa bir xarita asosida belgilashi va pushti, qizil, zumrad va qizil ranglar terminlaridan foyda-

lanishi mumkin. Yana bir modelchi esa spektrni har qanday tilga mos kelmaydigan, lekin turli shaxslar toifalarini ajratish uchun eng foydali bo'lgan hududlarga bo'lishi mumkin.

Modelchilar qaysi shaxslarni ko'rsatishni tanlaganidek, ular qaysi aloqalarni ishlatishni ham tanlaydilar. Biroq, aloqalar va shaxslarni tanlash uchun foydali bo'lgan ba'zi yo'riqnomalar mavjud. Bu yo'riqnomalar bir qator misollar orqali ko'rsatiladi.

16.2-misol. Agar siz «qizil» shaxslarni tasniflash uchun mos toifa deb qaror qilsangiz, qizil nomini unar aloqa sifatida ko'rib, qadoq a qizil deb yozishingiz mumkin:

red(a).

Agar siz rang ma'lumotlarini bunday tarzda ko'rsatsangiz, unda siz osonlik bilan «Nima qizil rangda?» degan savolni berishingiz mumkin:

ask red(X).

Qaytarilgan X javob qizil shaxslar bo'ladi.

Bunday ko'rsatmada «Qadoq qanday rangda?» degan savolni berish qiyin. Aniq qoidalar sintaksisida siz

ask X(a).

deb so'rashingiz mumkin emas, chunki birinchi tartibli mantiqqa asoslangan tillarda predikat nomlari o'zgaruvchilar bo'la olmaydi. Ikkinchi tartibli yoki yuqori tartibli mantiqda bu *a* ning har qanday xususiyatini qaytarishi mumkin, faqat uning rangi emas.

Sizda *a* qadoqning rangini so'rash imkonini beruvchi alternativ ko'rsatmalar mavjud. Dunyoda «qizil»ni predikat qilishga majbur qiladigan hech narsa yo'q. Ranglar ham shaxslar ekanini aytishingiz va qizil rangni ko'rsatish uchun doimiy qizil deb ishlatishingiz mumkin. Agar qizil doimiy bo'lsa, siz rangni ifodalovchi predikatdan foydalanishingiz mumkin. Bu yerda *color (Ind, Val)* jismoniy shaxs *Ind*ning rangi *Val* ekanligini anglatadi. «Qadoq *a* qizil rangda» endi quyidagicha yozilishi mumkin:

color (a, red).

Dunyoni qayta anglab yetdingiz. Endi dunyo siz nom berishingiz mumkin bo'lgan alohida elementlar sifatida ranglardan iborat. Endi jismoniy obyektlar va ranglar o'rtasida yangi ikkilik «rang» munosabati mavjud. Ushbu yangi ko'rinishning bir qismi sifatida siz «Nima qizil rangga ega?»ni *ask color(X, red)* so'rovi orqali berishingiz mumkin.

ask color (a, C) so'rov orqali «*a* bloki qanday rangda?» ekanini so'rang.

Mavhum tushunchani obyektga aylantirish uni hayotga tatbiq etishni anglatadi. Oldingi misol uchun, qizil rang haqiqatga aylandi.

16.3-misol. Avvalgi misolda ranglarning yangi ko'rinishida hech qanday kamchilik yo'qdek tuyuladi. Ilgari qilish mumkin bo'lgan hamma narsani hozir qilish mumkin. *Color(x, red)* ni yozish *red(x)* ga qaraganda uncha qiyin emas, ammo endi

siz buyumlarning rangi haqida savollar berishingiz mumkin. Shunday qilib, savol tug'iladi: buni har bir munosabat bilan qila olasizmi va oxir-oqibat nima bo'ladi? Siz 16.2-misolda Red predikati kabi rang predikati uchun ham xuddi shunday tahlil qilishingiz mumkin. Rangni predikat sifatida ishlatadigan ko'rinish «a uchastkasining qaysi xususiyati qizil rangga ega?» Bu yerda mos javob «rang» bo'ladi. 16.2-misoldagi kabi konvertatsiya qilish orqali siz *rang* kabi xususiyatlarni alohida elementlarga aylantirishingiz mumkin, shuningdek, prop munosabatini o'ylab ko'ring va «a shaxs qizil rangga ega»ni

prop(a,color:red)

sifatida yozing. Ushbu ko'rinish ushbu va oldingi misollardagi barcha so'rovlarni bajarishga imkon beradi. Siz buni yana qilishingiz shart emas, chunki siz barcha munosabatlarni *prop* munosabati nuqtayi nazaridan yozishingiz mumkin.

Shaxs–xususiyat–qiymat taqdimoti bitta prop munosabati orqali ifodalanadi, bu yerda

prop (Ind, Prop, Val)

shaxs *Ind* xususiyati *Prop* uchun *Val* qiymatiga ega ekanligini anglatadi. Bu, shuningdek, **uchlik taqdimoti** deb ham ataladi, chunki barcha munosabatlar uchliklar sifatida ifodalanadi. Uchlikning birinchi elementi **subyekt**, ikkinchisi **fe'l** va uchinchisi **obyekt** deb ataladi, uchlik oddiy uch so'zli jumla analogiyasi bilan.

Uchlik ba'zan uch so'zli jumla sifatida yoziladi:

subject verb object.

bu atom

prop(subject, verb, object)

ma'nosini anglatadi. Yoki funksional notatsiyada

verb(subject, object)

sifatida yozilishi mumkin.

Uchlikning fe'li **xususiyatdir**. Xususiyat *p*ning **domeni**, *p* fe'l bo'lganda, uchlikning subyekt sifatida paydo bo'lishi mumkin bo'lgan shaxslar to'plamidir.

P xususiyatning diapazoni *p* fe'l sifatida paydo bo'ladigan uchlikning obyekti sifatida paydo bo'lishi mumkin bo'lgan qiymatlar to'plamidir. **Atribut** – bu xususiyat–qiymat juftligidir. Masalan, posilkaning atributi uning rangi qizil ekanligini bildirishi mumkin.

Ba'zi predikatlar uchlik taqdimoti uchun juda sodda bo'lib tuyulishi mumkin.

16.4-misol. *Parcel(a)(posilka)* ni ko'rib chiqing, bu *a* posilka ekanligini anglatadi. Uni uchliklardan foydalangan holda ifodalashning ikki yo'li bor.

Birinchi, posilka tushunchasini reifikatsiya qilish va *a* posilka ekanligini aytilish:

prop (a, type, parcel).

Bu yerda *type* – shaxsni sinfga bog'laydigan xususiyatdir. *Parcel* konstantasi barcha real yoki potensial posilkalar bo'lgan narsalar sinfini bildiradi. Ushbu uchlik shaxs aning *posilkalar* sinfiga tegishli ekanligini belgilaydi. *Type* xususiyati

ba'zida «**is a**» (yoki «bu shunaqa») deb yoziladi, bu holda uchlik «a posilka» degan ma'noni anglatadi.

Ikkinchisi, posilkani ni xususiyat qilib qo'yish va «a posilka»ni

prop (a, parcel, true)

deb yozishdir.

Ushbu taqdimotda *parcel* – bu posilkalar uchun to'g'ri bo'lgan mantiqiy (Boolean) xususiyatdir. Bu xususiyat CSP'larda va mashinada o'rganishda ishlatiladigan indikator o'zgaruvchiga mos keladi.

Mantiqiy xususiyat – bu diapazoni {*true, false*} bo'lgan xususiyat bo'lib, *true* va *false* tilning doimiy belgilaridir.

Ba'zi predikatlar uchlik taqdimoti uchun juda murakkab bo'lib tuyulishi mumkin:

16.5-misol. «Berish» fe'li uchta ishtirokchini talab qiladi: beruvchi agent, qabul qiluvchi va pacient (berilayotgan narsa). Agar munosabatda faqat ikkita ishtirokchi ko'rsatilsa, «Alex Chrisga darslik berdi» jumlasini ma'lumotlarni yo'qotadi. *Buni gave (alex, chris, book)* sifatida ifodalash mumkin. Uni uchliklar yordamida, masalan, *ga3545* degan berish harakatini ixtiro qilish orqali quyidagi uchliklar bilan ifodalash mumkin:

prop (ga3545, type, giving_act).

prop (ga3545, agent, alex).

prop (ga3545, patient, b342).

prop (ga3545, recipient, chris).

prop (b342, type, book).

Bu yerda, *ga3545* harakatni bildiruvchi reifikatsiyalangan subyekt va *b342* berilgan darslikdir. Agent – bu harakatni amalga oshiruvchi shaxs, bemor – bu harakat amalga oshiriladigan obyekt va recipient – bemorni qabul oluvchi shaxsdir. *Agent, bemor va recipient* xususiyatlari **semantik rollar** yoki jummalarga ma'no berish uchun ishlatiladigan **tematik munosabatlardir**. Reifikatsiya berish harakatining boshqa xususiyatlarini, masalan, sanani yoki to'langan narxni ko'rsatish imkonini beradi.

Reifikatsiya qilingan munosabatlarni ifodalovchi so'zlar va raqamlar juda keng tarqalgan. Masalan, bron qilish, rezervatsiya, nikoh, reys raqami, sotib olish buyurtmasi barchasi bir munosabatni bildiradi va ishtirokchilar va boshlanish vaqti kabi xususiyatlarga ega.

16.1.2. Shaxslar va identifikatorlar

Shaxslar noyob **identifikatorlar** bilan belgilangan. Masalan, universitetda talabalar raqami talaba uchun ishlatiladigan identifikator hisoblanadi. Ism yetarli emas, chunki bir xil ismli bir nechta talabalar bo'lishi mumkin va ba'zi talabalar o'z ismlarini o'zgartiradilar.

Yagona resurs identifikatori (uniform resource identifier(URI)) yoki uning Unicode kengaytmasi, **xalqaro resurs identifikatori (internationalized resource identifier (IRI))**, har qanday narsani aniqlash uchun ishlatilishi mumkin bo'lgan noyob nom hisoblanadi. **Resurs** – bu nomlanishi mumkin bo'lgan har qanday narsadir. IRI odatda yagona resurs manziliga (URL), veb-manzilga ega bo'lib, odatda <http://> yoki <https://> bilan boshlanadi, chunki URLlar noyobdir. IRI veb-saytni emas, balki subyektni bildiradi. Maqsad shundaki, kimdir IRI'dan foydalansa, ular IRI bilan belgilangan shaxsni nazarda tutadi. IRI ma'no kasb etadi, chunki odamlar uni shu ma'noda ishlatishadi. Rasmiy ta'rif kerak emas; uni izchil ishlatish kifoyadir.

Wikidata (<https://www.wikidata.org>) 1,25 milliardga yaqin uchliklarni va 100 million shaxslarni tavsiflovchi bepul, hamkorlikdagi bilimlar grafigidir (2022-yil holatiga ko'ra).

16.6-misol. Christine Sinclair – kanadalik assotsiatsiya futbolchisi, xalqaro o'yinlarda boshqa har qanday o'yinchidan ko'proq gol urgan. 2021-yildagi Wikidata'da Christine Sinclair «<http://www.wikidata.org/entity/Q262802>» identifikatori orqali ifodalangan, bu identifikatorni qisqartirib «Q262802» deb ataymiz. Wikidata bu identifikatorni futbolchini boshqa har qanday Christine Sinclair ismli shaxslardan ajratish uchun ishlatadi. «<http://www.wikidata.org/entity/Q262802>» identifikatori futbolchini bildiradi, veb-sahifani emas (futbol o'ynamaydiganlarni).

Wikidata xususiyatlar uchun noyob identifikatorlardan foydalanadi. Masalan, subyektning ismini beruvchi xususiyat uchun «<http://schema.org/name>» identifikatori ishlatiladi. Biz «name»dan foydalanamiz, lekin bu schema.org tomonidan belgilangan xususiyatning qisqartmasi ekanligini yodda tuting. Agar siz bu ma'noni xohlamasangiz, boshqa identifikatorni ishlatishingiz kerak. «name» xususiyati bilan uchlikning qiymati satr va til juftligidan iborat bo'ladi, masalan, («Christine Sinclair», en).

Wikidata «fuqarolik davlati» xususiyati uchun «<http://www.wikidata.org/prop/direct/P27>» identifikatoridan foydalanadi, bu yerda subyekt shaxs, obyekt esa ular fuqarosi bo'lgan mamlakatdir. Kanada «<http://www.wikidata.org/entity/Q16>» identifikatori bilan ifodalangani, shuning uchun «Christine Sinclair Kanadaning fuqarosi» deb aytish uchlik orqali quyidagicha ifodalangani:

/entity/Q262802 /prop/direct/P27 /entity/Q16

lekin to'liq IRI lar (shu jumladan «<http://www.wikidata.org>») bilan ishlatiladi.

16.1.3. Graf taqdimotlar

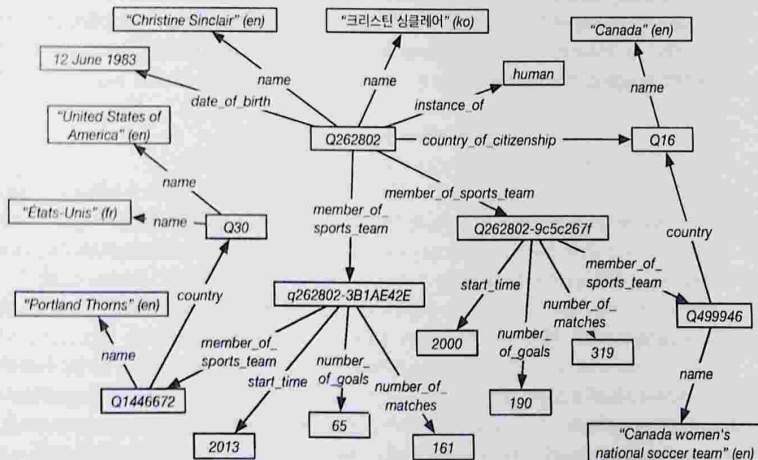
Siz *prop* munosabatini **bilimlar grafigi**, yo'naltirilgan belgilangan graf shaklida talqin qilishingiz mumkin, bu yerda tugunlar subyektlar (satrlar va raqamlar kabi boshqa qiymatlar) bo'ladi.

munosabati Ind dan *Val* ga *Prop* bilan belgilangan dumli yoyni belgilaydi. Bunday grafik **semantik tarmoq** deb ham ataladi.

16.7-misol. 16.1-rasmda Christine Sinclair (Q262802) haqida **Wikidata** bilimlar grafigining bir qismi ko'rsatilgan. 2022-yil avgust holatiga ko'ra, Wikidata uning haqida 3400 dan ortiq uchliklarni taqdim etadi, jumladan uning ismi 47 tilda, ingliz va koreys tillaridagi ko'rsatilgan.

Christine Sinclair Kanadaning fuqarosi. Wikidata nomi oʻrniga «<http://www.wikidata.org/prop/direct/P27>», *country_of_citizenship* koʻrsatilgan. U 2000 yilda beri Kanada ayollar milliy futbol jamoasida (Q499946 identifikatori) 319 ta oʻyinida oʻynagan va 190 ta gol urgan. U va jamoa oʻrtasidagi munosabat Q262802-9c-5c267f identifikatori yordamida reifikatsiya qilingan (haqiqiy identifikator bundan uzoqroq), u bilan «<http://www.wikidata.org/prop/P54>» (sport jamoasi aʼzosi) xususiyati bilan bogʻlangan. Reifikatsiyalangan subyekt va Q499946 oʻrtasidagi xususiyat «<http://www.wikidata.org/prop/statement/P54>», shuningdek, sport jamoasi aʼzosi sifatida koʻrsatilgan.

U 2013-yildan beri Portland Thorns (Q1446672 identifikatori) jamoasida 161 ta o'yinda o'ynab, 65 ta gol urgan.



16.1-rasm. Christine Sinclair haqida Wikidata bilimlar grafigining bir qismi. IRI lar emas, xususiyatlarning inglizcha nomlari ko'rsatilgan.

Avval ta'kidlangan moslashuvchanlikdan tashqari, grafik yozuvning bir qancha afzalliklari mavjud:

– Bu mantiqning ma'lum bir sintaksisini o'rganmasdan, odamga munosabatlarini ko'rishni osonlashtiradi. Grafik yozuv bilim bazalarini yaratishda bilimlarni tashkil etishda yordam beradi. Foydalanuvchilar shunchaki tugunlar va yoylarni chizadigan (ba'zida **bilim xaritalari** yoki **ong xaritalari** deb ataladigan) ko'plab bilim grafiklarini yaratish uchun vositalar mavjud.

– Bilim grafikasini ko'rib chiqayotgan kishi faqat ma'nosiz nomlarga ega tugunlarning yorliqlarini e'tiborsiz qoldirishi mumkin – masalan, 16.5-misoldagi ga3545 nomi yoki 16.1-rasmdagi Q262802-9c5c267f nomi. Taqdimot va vizualizatsiya ushbu tugunlarni bo'sh qoldirishi va agar ular aniq uchliklarga mos kelishi kerak bo'lsa, ixtiyoriy nom yaratishi mumkin.

– Uchliklar **uchlik do'konida** samarali saqlanishi mumkin. Relatsion ma'lumotlar bazasida dizayner ma'lumotlar bazasiga indeksatsiya qilish uchun ishlatiladigan kalitlarni belgilashi kerak; kalit berilgan bo'lsa, tegishli tuple'larni samarali topish mumkin. Uchlik do'konida har qanday berilgan pozitsiyalar kombinatsiyasi bilan bog'liq tuple'larni topish uchun sakkizta indeks kerak bo'ladi. Bir tomonda subyekt, fe'l va obyekt berilgan va so'rov ma'lum bir uchlik do'konda bor yoki yo'qligini aniqlashdir. Boshqa tarafda hech qanday pozitsiya belgilanmagan va maqsad uchliklarni sanashdir. Ular orasida esa, masalan, faqat subyekt va obyekt berilgan va maqsad ularni bog'laydigan fe'llarni sanash yoki faqat subyekt va fe'l berilgan va maqsad obyektlarni sanashdir. Sakkizta indeks talab qilinadi, chunki subyekt, fe'l va obyektning har biri berilishi yoki berilmasligi mumkin. Ushbu indekslar bilan dizayner uchliklarga samarali kirishni ta'minlash uchun kalitlarni belgilashga hojat qoldirmaydi.

16.2. Sinflar va xususiyatlar

Odatda, siz biror soha haqida faktlar bazasidan ko'proq narsani bilasiz; boshqa faktlar chiqarib olinishi mumkin bo'lgan umumiy qoidalarini bilishingiz mumkin. Qaysi faktlar aniq berilishi va qaysilari chiqarib olinishi ma'lumotlar bazasini loyihalash va qurishda tanlanadi.

Ibtidoiy bilim – bu aniq ko'rsatilgan bilimdir. **Egallangan bilim** – bu ibtidoiy bilim va boshqa hosil qilingan bilimlardan olinadigan bilimdir.

Qoidalaridan foydalanish bilimning yanada ixcham taqdimotiga imkon beradi. Olingan munosabatlar biror narsa sinfga tegishli ekanligini bilish asosida **umumlash-tirishga** imkon beradi. Bu muhim, chunki siz soha haqida hamma narsani bevosita kuzatib boravermaysiz. Sohaning ko'p ma'lum bo'lgan narsalari kuzatishlardan va umumiy bilimlardan, shu jumladan o'rganilgan bilimlardan kelib chiqadi.

Olingan bilimdan foydalanishning standart usuli sinfning barcha a'zolari uchun amal qiladigan atributlarni (xususiyat-qiymat juftliklarini) belgilashdir.

Shaxslar o'zlari bo'lgan sinflarga tegishli atributlarni meros qilib oladi. Shaxslarni sinflarga guruhlash har bir shaxs uchun alohida atributlarni ifodalashdan ko'ra ixchamroq taqdimotni ta'minlaydi. Ushbu masala ehtimoliy klassifikatorlar va nazoratsiz o'rganish kontekstida muhokama qilingan.

Sinf – bu sinf a'zolari bo'ladigan barcha mavjud va potensial shaxslarning to'plami. Bu odatda **intension to'plamdir**, ya'ni to'plam a'zolari uchun to'g'ri va boshqa shaxslar uchun noto'g'ri bo'lgan **xarakteristik funksiya** orqali aniqlanadi. Intension to'plamning alternativasi **ekstension (kengaytmali) to'plam** bo'lib, uning elementlarini sanab chiqish orqali aniqlanadi.

Masalan, *stul* sinfi bu barcha *stul* bo'ladigan narsalar to'plamidir. Ta'rif mavjud stullar to'plami emas, chunki hali qurilmagan stullar ham *stul* sinfiga kiradi. Ushbu ta'rifsiz agent *stul*ni loyihalay olmas edi, chunki *stul* loyihalash jarayonida mavjud emas (va hech qachon mavjud bo'lmasligi mumkin).

Ikki sinf bir xil a'zolariga ega bo'lganligi uchun teng emas. Masalan, yashil unicornlar sinfi va 124 metr balandlikdagi stullar sinfi turli sinflardir, hatto ular bir xil elementlarni o'z ichiga olsa ham. Ular ikkalasi ham bo'sh. Yashil unicorn – bu 124 metr balandlikdagi *stul* emas.

Sinf ta'rifi tasvirlanishi mumkin bo'lgan har qanday to'plamni sinf sifati-da ko'rishga imkon beradi. Masalan, 17 raqami, Agra shahridagi Toj Mahal va 2022-yilgi FIFA Jahon chempionati finalidagi birinchi gol to'plamidan iborat to'plam sinf sifatida qaralishi mumkin, lekin bu juda foydali emas. **Tabiiy tur** – bu shaxslarni sinfdan foydalanib tasvirlash oddiyroq bo'lgan sinfdir. Masalan, «mammal» tabiiy turdir, chunki sut emizuvchilarning umumiy xususiyatlarini tasvirlash «mammal»ni ishlatadigan bilim bazasini «mammal»ni ishlatmaydigan va har bir shaxs uchun xususiyatlarni takrorlaydigan bilim bazasiga qaraganda ixchamroq qiladi.

16.2.1. Sinf va xususiyatlar iyerarxiyalari

S sinf C sinfning subklassi, demak S Cning bir qismidir. Ya'ni S turidagi ha bir shaxs C turidandir.

16.8-misol. 16.8-misol 16.2-rasmda Braziliya milliy futbol terma jamoasi va Kanada ayollar futbol terma jamoasining tashkilot darajasiga qadar sinf tuzilishi ko'rsatilgan. Obyekt yoki sinfning inglizcha nomi qavs ichida Wikidata identifikatori bilan ko'rsatilgan.

Eng quyi darajada Braziliya milliy futbol jamoasi ingliz tilida «national association football team» deb nomlangan q6979593 sinfining namunasidir. Bu «milliy futbol jamoasi» va «assotsiatsiya futbol jamoasi» deb nomlangan sinflarning kichik sinfidir.

Turlar va kichik sinflar o'rtasidagi bog'liqlik ma'lum bir jumla sifatida tuzilishi mumkin:

$prop(E, type, C) \leftarrow$
 $prop(S, subClassOf, C) \wedge$
 $prop(E, type, S).$

Yoki funksional notatsiya yordamida, agar $subClassOf(S, C)$ va $type(E, S)$ bo'lsa, unda $type(E, C)$ bo'ladi.

Siz $type$ va $subClassOf$ ni **xususiyat merosxo'rligini** ta'minlaydigan maxsus xususiyatlar sifatida ko'rishingiz mumkin. Xususiyat merosxo'rligi xususiyat uchun qiymat sinf darajasida belgilanganida va sinf a'zolari tomonidan meros qilib olinganda sodir bo'ladi. Agar c sinfning barcha a'zolari p xususiyati uchun v qiymatiga ega bo'lsa, bu quyidagi aniq klausada yozilishi mumkin:

$prop(Ind, p, v) \leftarrow$

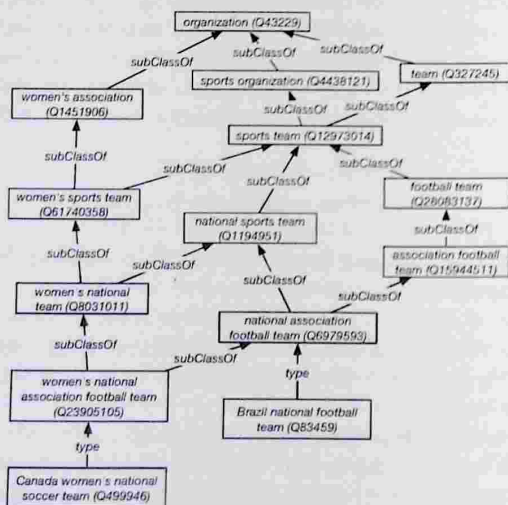
$prop(Ind, type, c).$

Ushbu qoidadan turlar va subclasslar o'rtasidagi munosabat bilan birga foydalanib, xususiyat merosxo'rligi uchun foydalanish mumkin.

p_1 xususiyat c xossaning **subxususiyati** bo'lsa, har bir p_1 orqali bog'langan juftlik p_2 orqali ham bog'langan bo'ladi. Funksional notatsiyada bu $p_1(x, y)p_2(x, y)$ ni anglatadi, yoki uchlik notatsiyada $(x, p_1, y)(x, p_2, y)$ ni anglatadi.

16.9-misol. Wikidatada ba'zi subxususiyatlar quyidagilar:

– «*member of sports team*» (P54) «*member of*» (P463)ning subxususiyati hisoblanadi.



16.2-rasm. Braziliya milliy futbol jamoasi va Kanada ayollar milliy futbol jamoasi uchun Wikidata sinf tuzilmasining bir qismi

- (P463) «a'zosi» (P1416) «bog'lanish»ning subxususiyati hisoblanadi.
- (P1416) «bog'lanish» (P361) «qismi»ning subxususiyati hisoblanadi.
- (P361) «qismi» (P1382) «qisman mos keluvchi»ning subxususiyati hisoblanadi.
- (P1382) «qisman mos keluvchi» (P1889) «farqli»ning subxususiyati hisoblanadi.

Xususiyatning **domeni** – bu sinf bo'lib, xususiyatga ega uchlikning subyekti (birinchi argument) ushbu sinfga tegishli bo'lishi kerak. Ya'ni «xususiyat p ning domeni C sinfidir» degani, agar $p(x, y)$ bo'lsa, unda $x \in C$. Agar p xususiyatning domeni C_1 va C_2 bo'lsa, unda p subyekti bo'lgan har bir individ C_1 va C_2 sinflariga tegishli bo'lishi kerak.

Xususiyatning **diapazoni** (range) – bu sinf bo'lib, xususiyatga ega uchlikning obyekti (so'nggi argument) ushbu sinfga tegishli bo'lishi lozim. Ya'ni p xususiyatning doirasi C sinfidir degani. Agar $p(x, y)$ bo'lsa, unda $y \in C$.

P xususiyat funksional bo'lsa, demak har qanday subyekt bilan faqat bitta obyekt bog'langan bo'ladi, ya'ni, agar $p(x, y_1)$ va $p(x, y_2)$ bo'lsa, unda $y_1 = y_2$. Bu qisman funksiyaning matematik ta'rifiga mos keladi.

Funksional xususiyatlar 17-bobda relyatsion o'rganish va relyatsion ehtimoliy modellarda alohida o'rin tutadi, chunki cheklovlar bilan bog'liq muammolar va mashinani o'rganishda foydalaniladigan xususiyatlar hamda tasodifiy o'zgaruvchilar funksional deb hisoblanadi (faqat bitta qiymatga ega). Bu yerda o'ziga xos jihati shundaki, «do'stga ega» kabi funksional bo'lmagan xususiyatlarga ruxsat berilgan.

16.10-misol. Quyidagilar hozirda Wikidata tarkibiga kirmaydi, lekin kiritilishi mumkin:

- (P54) «Sport jamoasi a'zosi»ning domeni (Q5) «inson»dir.
- (P54) «Sport jamoasi a'zosi»ning doirasi (Q12973014) «sport jamoasi»dir.
- (P54) «Sport jamoasi a'zosi» funksional emas, chunki kimdir bir va birinchi o'zida bir nechta sport jamoalarida bo'lishi mumkin. 16.1-rasmda ko'rsatilganidek, Christine Sinclair ikkita sport jamoasida edi.
- (P569) «Tug'ilgan sana» funksional, chunki har bir insonda faqat bitta tug'ilgan sana mavjud.

Ba'zi umumiy ko'rsatmalar nimani ibtidoiy va nimani hosila sifatida belgilash kerakligini aniqlash uchun foydali bo'lishi mumkin:

- Bir shaxsga atributni bog'lashda, shaxs kiradigan eng umumiy C sinfni tanlang, bu yerda C sinfining barcha a'zolari shu atributga ega bo'ladi va atributni C sinfi bilan bog'lang. Meros olishdan foydalanib, atributni shaxs va C sinfining boshqa barcha a'zolari uchun hosil qilish mumkin. Bu vakillik metodologiyasi bilim bazalarini ixchamroq qiladi va bu yangi shaxslarni kiritishni osonlashtiradi, chunki C sinfi a'zolari avtomatik ravishda atributni meros qilib oladilar. Masalan,

odamlarda umurtqa pog'onasi mavjud; bu umurtqali hayvonlar sinfi bo'lishi bilan ifodalanadi, ular orasida insonlar bir sinfga kiradi.

- Sinfning tasodifiy atributini sinf bilan bog'lamang. **Shartli atribut** – bu hollarlar o'zgarganda o'zgaradigan qiymatga ega bo'lgan atributdir. Masalan, futbol jamoasini murabbiyga ega bo'lishi bilan baholamang; murabbiy iste'foga chiqsa yoki vafot etsa, jamoa jamoa bo'lishdan to'xtamaydi.

16.2.2. Sinflarni loyihalash

Obyektlarni tasniflash, zamonaviy ontologiyalarning asosi, uzoq tarixga ega. Aristotel [miloddan avvalgi 350-yil] C sinfini quyidagilar asosida aniqlashni taklif qildi:

- **Genus:** C sinfining superklassidir. Genusning ko'pligi «genera».
- **Differensiya:** C sinfi a'zolarini C sinfining superklassidagi boshqa a'zolaridan farqlovchi atributlar.

U ta'rifda yuzaga keladigan ko'plab masalalarni oldindan ko'rgan:

Agar nasllar har xil va muvofiqlashtirilgan bo'lsa, unda ularning differentsiallari tabiatan farq qiladi. Misol tariqasida «hayvon» jinsini va «bilim» jinsini olaylik. «Oyoqlari bor», «ikki oyoqli», «qanotli», «suvli» – bu «hayvon» navlari; bilim turlari bir-biridan turli yo'llar bilan farq qiladi. Bilimning bir turi boshqasidan farq qilmaydi, chunki u «ikki oyoqli».

– Aristotel [miloddan avvalgi 350-yil]

E'tibor bering, bu yerda «muvofiqlashtirish» ularning hech biri boshqasiga bo'ysunmasligini anglatadi. Zamonaviy ontologiyalar uslubida biz «hayvon» sinf, «bilim» esa sinf deb aytamiz. «Ikki oyoqli» xususiyati «hayvon» domeniga ega. Agar biror narsa bilimning namoyandasi bo'lsa, u «ikki oyoqli» xossasiga qiymatga ega bo'lmaydi.

Narsalarni jins va turlarga ajratish san'ati juda muhim va bizning fikrimizga va xotiramizga katta yordam beradi... Bu odamga nafaqat xotirasidagi narsalarni saqlashga, balki ularni o'sha yerda topishga yordam beradi. Har xil tushunchalarni ma'lum rubrikalar yoki toifalarga birlashtirgan yozuvchilar juda foydali ish qildilar.

– Leybnits [1705]

Aristotel ta'riflariga asoslangan ontologiyani yaratish uchun:

Siz aniqlamoqchi bo'lgan har bir sinf uchun tegishli super sinfni aniqlang va keyin ushbu sinfni boshqa kichik sinflardan ajratib turadigan atributlarni tanlang. Har bir atribut xususiyat va qiymatni o'z ichiga oladi.

Har bir xususiyat uchun ma'noga ega bo'lgan eng umumiy sinfni aniqlang va xususiyatning domenini shu sinf qilib belgilang. Xususiyatning qamrovini boshqa bir sinf qilib belgilang, bu sinfni qiymatlarini sanab yoki Aristotel ta'rifidan foydalangan holda aniqlab bo'ladi.

Ma'lumotlar bazalaridagi sinflar va obyektga yo'naltirilgan dasturlash

Bilinga asoslangan tizimlarda «shaxslar» va «sinflar»dan foydalanish **Small-talk, Python** yoki **Java** kabi **obyektga yo'naltirilgan dasturlash (OOP (object-oriented programming))** tillarida «obyektlar» va «sinflar»dan foydalanishga juda o'xshaydi. Bunga ajablanmaslik kerak, chunki ular bir-biriga bog'liq tarixga ega. Ammo to'g'ridan to'g'ri analogiya qilish ko'pincha yordamdan ko'ra ko'proq chalkashlik tug'diradigan muhim farqlar mavjud:

- OOPdagi obyektlar hisoblash obyektlaridir. Ular ma'lumotlar tuzilmalari va ularga bog'liq dasturlardir. Java dasturidagi «shaxs» obyekt shaxs emas. Biroq, bilimlar bazasidagi (KB) individlar (odatda) haqiqiy dunyodagi narsalardir. KBdagi «shaxs» individ haqiqiy shaxs bo'lishi mumkin. «Stul» individi esa haqiqiy stul bo'lib, unga haqiqatda o'tira olasiz; unga urilganingizda u sizni jarohatlaydi. Java dasturidagi «stul» obyektiga xabar yuborishingiz va javob olishingiz mumkin, ammo haqiqiy dunyodagi stul siz unga nima aytasangiz, e'tibor bermaydi. KB odatda stul bilan o'zaro aloqada bo'lish uchun emas, balki stul haqida fikrlash uchun ishlatiladi. Haqiqiy stul o'z joyida turadi, faqat uni jismoniy agent ko'chirganda harakatlanadi.

- Bilimlar bazasida (KB) obyektning tasviri faqat bir (yoki bir necha) abstraksiya darajasida yaqinlashuv bo'lib, haqiqiy obyektlar odatda tasvirlangan narsalardan ancha murakkab bo'ladi. Masalan, stulning matosidagi har bir tolani odatda tasvirlamaysiz. Obyektga yo'naltirilgan dasturlash (OOP) tizimida esa faqat obyektning tasvirlangan xususiyatlari mavjud. Tizim Java obyektidagi har bir xususiyatni bilishi mumkin, ammo haqiqiy individ haqida emas.

- Java sinf tuzilmasi asosan loyihalangan obyektlarni ifodalash uchun mo'ljallangan. Tizim tahlilchisi yoki dasturchi dizayn yaratadi. Masalan, Javada obyektlar faqat bir eng past darajadagi sinfga a'zo bo'ladi. Bu yerda ko'p merosxo'rlik mavjud emas. Haqiqiy obyektlar bunday qattiq cheklovlarga duch kelmaydi. Bir xil shaxs futbol murabbiyi, matematik va ona bo'lishi mumkin.

- Kompyuter dasturi o'z ma'lumot tuzilmalari haqida noaniqlikka ega bo'la olmaydi; u foydalanadigan ma'lumot tuzilmalarini tanlash kerak. Ammo dunyodagi narsalarning turlari haqida noaniqlikka ega bo'lishingiz mumkin.

- KBdagi tasvirlar aslida hech narsa qilmaydi. OOP tizimida obyektlar hisoblash ishini bajaradi. KBda ular faqat dunyodagi obyektlarni ifodalaydi.

- UML kabi obyektga yo'naltirilgan model tuzish tili KB larni ifodalash uchun ishlatilishi mumkin, lekin bu eng yaxshi tanlov bo'lmاسligi mumkin. Yaxshi obyektga yo'naltirilgan model tuzish vositasi yaxshi dizaynlarni yaratishga yordam beradigan imkoniyatlarga ega. Biroq, modellashtirilayotgan dunyo yaxshi dizayn

bo'lmashligi mumkin. Toza dunyoga yaxshi dizayn paradigmasini majburlash produktiv bo'lmashligi mumkin.

Knublauch va boshqalar [2006] obyektga yo'naltirilgan dasturiy ta'minot dizayni va ontologiyalardan foydalanish o'rtasidagi batafsilroq taqqoslashni taqdim qiladi.

16.11-misol. 16.8-misolda milliy sport jamoasi «bir millatni sportda ifodalovchi jamoa» bo'lib, bu yerda «jamoas» umumiy tushuncha va «bir millatni sportda ifodalovchi» esa ajraluvchi xususiyatdir. Assotsiatsion futbol jamoasi sport jamoasi bo'lib, u futbol o'ynaydi. Shunday qilib, milliy assotsiatsion futbol jamoasi millatni assotsiatsion futbolda ifodalovchi jamoa bo'lib, ikkala sinfning ham kichik sinfi hisoblanadi. Jamoa «umumiy maqsadga birlashgan guruh» sifatida ta'riflanadi.

16.2-rasmda chap tomondagi sinflarni farqlovchi ajraluvchi xususiyat a'zolarining ayollardan iborat bo'lishi shartidir. Shuningdek, ayollar assotsiatsion futbol jamoasi, ayollar futbol jamoasi va ayollar jamoasi mavjud bo'lishi mumkin, ularning barchasiga Wikidatada identifikatorlar berilmagan. Shuni unutmangki, «eraklar futbol jamoasi» sinfini ta'riflaydi, lekin Wikidatada mavjud emas, chunki ko'pchilik jamoalar ayollarga ham ruxsat beradi, hatto ularda hech kim o'ynamasa ham.

Sinf iyerarxiyasi yo'naltirilgan graf bo'lib, unda pastki sinflardan darhol yuqoridagi sinflarga bog'lanishlar mavjud. Tsiklik ta'riflar foydasiz; masalan, x y ustida joylashgan deb ta'riflash va y x ostida joylashgan deb ta'riflash hech narsani aniqlamaydi; bu faqat tenglikni yaratadi. Shuning uchun, grafni yo'naltirilgan asiklik graf (DAG (directed acyclic graph)) deb hisoblash mantiqan to'g'ri, bu esa panjara hosil qiladi. Ushbu metodologiya, odatda, sinflar iyerarxiyasining daraxt shaklidagi iyerarxiyasini bermaydi. Obyektlar bir nechta sinflarda bo'lishi mumkin. Har bir sinf eng aniq yuqori sinfga ega bo'lmaydi.

16.12-misol.

To'rtburchak, romb va kvadratning ta'riflarini ko'rib chiqing:

- To'rtburchak – barcha ichki burchaklari to'g'ri (90°) bo'lgan to'rtburchakdir.
- Romb – to'rtta tomoni teng uzunlikdagi to'rtburchakdir.
- Kvadrat – to'rtta tomoni teng uzunlikdagi va barcha ichki burchaklari to'g'ri bo'lgan to'rtburchakdir.

Kvadrat ham to'rtburchak, ham romb hisoblanadi; ikkalasi ham eng aniq yuqori sinflardir. Kvadratni barcha tomonlari teng uzunlikdagi to'rtburchak deb ta'riflash mumkin. Shuningdek, uni ichki burchaklari to'g'ri bo'lgan romb deb ta'riflash ham mumkin. To'rtburchak to'rtta to'g'ri chiziqli tomondan tashkil topgan tekislik shakli hisoblanadi. Kvadratning ta'rifini kengaytirib, kvadratni to'rt teng uzunlikdagi to'g'ri chiziqli tomondan tashkil topgan va ichki burchaklari to'g'ri bo'lgan tekislik shakli deb ta'riflash mumkin.

Kamdan kam hollarda sinflarning tabiiy iyerarxiyasi tirik mavjudotlarning **Linney taksonomiyasida** eng yaxshi ma'lum bo'lgan daraxtni hosil qiladi. Bu

daraxtning sababi evolyutsiyadir. Boshqa sohalarida daraxtga o'xshash tuzilmani yaratishga urinishlar unchalik muvaffaqiyatli bo'lmadi.

Agar sinf tuzilishi asiklik bo'lsa va har bir sinf – biz thing deb ataydigan yuqori sinfdan tashqari – super sinf va differensiatsiyani tashkil etuvchi atributlar nuqtayi nazaridan aniqlansa, u holda har bir sinf har bir super sinfni uning ta'rif bilan almashtirish orqali atributlar bilan birlashtirilgan narsa sifatida normal shaklga ega. Agar atributlar mulk-qiymat juftlarini ifodalasa, unda bitta sinf boshqasining pastki sinfidir, agar uning odatiy shakli boshqasining yuqori to'plami bo'lsa; 16.3.1-mavzu sinflarning yanada ifodali konstruksiyalarini tavsiflaydi. 16.12-misolda kvadrat to'rtburchak va romb subklassidir, chunki kvadratni ajratib turadigan qo'shimcha atributdan tashqari barcha atributlar bir xil.

16.3. Ontologiyalar va bilim almashish

Bilimga asoslangan katta tizimlarni qurish juda qiyin:

- Bilim ko'pincha bir nechta manbadan keladi, jumladan odamlar, sensorlar va veb, bu manbalarni birlashtirish zarur. Bundan tashqari, bu manbalar dunyoni bir xil taqsimlamasligi mumkin. Ko'pincha bilim turli sohalaridan keladi, har birining o'ziga xos terminologiyasi bo'lib, dunyoni o'z ehtiyojlariga qarab taqsimlaydi.

- Tizimlar vaqt o'tishi bilan rivojlanadi va kelajakda amalga oshirilishi kerak bo'lgan barcha farqlarni oldindan bilish qiyin.

- Bilim bazasini loyihalashda ishtirok etayotgan odamlar qaysi shaxslar va munosabatlarni ifodalashni tanlashlari kerak. Dunyo shaxslar bo'lib taqsimlanmagan; bu aqlli agentlar tomonidan dunyoni tushunish uchun amalga oshiriladi. Bilim asosidagi tizimda ishtirok etayotgan turli odamlar dunyoni taqsimlashda kelishib olishlari kerak.

- O'z notatsiyangiz nima anglatishini eslab qolish ko'pincha qiyin, boshqalarning notatsiyasi nima anglatishini aniqlash esa yanada qiyin. Buning ikki jihati bor:

- kompyuterdagi belgi nima anglatishini aniqlash
- biror kishining ongida bir tushuncha berilgan bo'lsa, qaysi belgini ishlatishni aniqlash. Buning uch jihati bor:

- tushunchaning allaqachon belgilanganligini aniqlash;
- agar u belgilangan bo'lsa, qaysi belgi ishlatilgani haqida ma'lumot olish;
- agar hali belgilangan bo'lmasa, uni qanday qilib belgilash mumkinligini aniqlash uchun tegishli tushunchalarni topish.

Bilimni ulashish va muloqot qilish uchun umumiy leksika va ushbu leksikaning kelishilgan ma'nosini ishlab chiqish muhimdir. **Konseptualizatsiya yoki mo'ljallangan talqin** kompyuterdagi belgilar, leksika va dunyodagi shaxslar va munosabatlar o'rtasidagi xaritalashdir. Bu dunyoning ma'lum bir mavhumlashuvi va ushbu abstraksiya uchun notatsiyani taqdim etadi.

Semantik veb – bu mashinaga talqin qilnadigan bilimlarni jahon internet tarmog'ida tarqatish usulidir. Faqatgina insonlar o'qishi uchun mo'ljallangan HTML sahifalarini taqdim etish o'rniga, veb-saytlar kompyuterlar tomonidan foydalanilishi mumkin bo'lgan ma'lumotlarni ham taqdim etishi mumkin.

Eng oddiy darajada, **XML (the Extensible Markup Language)** (Kengaytiriladigan belgilash tili) mashinalar o'qishi uchun mo'ljallangan, ammo insonlar o'qishi mumkin bo'lgan sintaksisni taqdim etadi. Bu iyerarxik tarzda belgilangan matn asosidagi til. XML sintaksisi juda murakkab bo'lishi mumkin, lekin eng oddiy darajada, bir teginishning ko'lamidagi formatlar quyidagicha: `<tan .../>` yoki `<tan ...> ...</tan >`

IRI (internationalized resource identifier (xalqaro resurs identifikatori)) resursni noyob ravishda aniqlash uchun ishlatiladi. Resurs – bu noyob ravishda aniqlanishi mumkin bo'lgan har qanday narsa, shu jumladan shaxslar, klasslar va xususiyatlar. Odatda, IRIlar veb-manzil (URL) sintaksisini ishlatadi.

RDF ((the resource description framework) resursni tasvirlash ramkasi) – bu XML asosida qurilgan til, u shaxs-xususiyat-qiyamat uchliklari uchun mo'ljallangan.

RDFS (RDF sxemasi) resurslarni (klasslar va xususiyatlar) boshqa resurslar orqali (masalan, *subclassOf* va *subPropertyOf* dan foydalanib) aniqlashga imkon beradi. RDFS shuningdek, xususiyatlarning domeni va diapazonini cheklash imkonini beradi va konteynerlar taqdim etadi: to'plamlar, ketma-ketliklar va alternativalar.

RDF o'zining tilidagi gaplarni reifikatsiya qilishga imkon beradi. Bu tasodifiy mantiqiy formulalarni ifodalashga imkon beradi va shu sababli umuman hal qilinmaydi. Hal qilinmaslik har doim yomon narsa emas; bu shunchaki hisoblashning qancha vaqt olishini cheklay olmaysiz, deganidir. Funksional ramzlari bo'lgan mantiqiy dasturlar va deyarli barcha dasturlash tillaridagi dasturlar hal qilinmaydi.

OWL (the web ontology language (web ontologiya tili)) – bu jahon internet tarmog'i uchun ontologiya tilidir. U sinflar, xususiyatlar va shaxslarni ta'riflash uchun ishlatilishi mumkin bo'lgan ba'zi klasslar va xususiyatlarni aniq talqin qilish bilan belgilaydi. Shu bilan birga, shaxslar, sinflar va xususiyatlar tengligini ta'minlaydigan mexanizmlar, xususan, xususiyatlar domenlarini va diapazonlarini cheklash va boshqa xususiyatlar bo'yicha cheklovlar (masalan, transitivity, kardinalitet) kiritadi.

Katta universal ontologiyalarni yaratishga qaratilgan ba'zi tashabbuslar bo'lgan, masalan, **Cyc** (www.cyc.com), lekin semantik veb g'oyasi hamjamiyatlarga ontologiyalarga birlashishiga imkon berishdir. Har kim ontologiya qurishi mumkin. Bilim bazasini rivojlantirishni istagan odamlar mavjud ontologiyalardan foydalanishlari yoki o'z ontologiyalarini rivojlantirishlari mumkin, odatda, mavjud ontologiyalar asosida. Ularning semantik o'zaro ta'sirini ta'minlash manfa'atida, kompaniyalar va shaxslar o'z sohalari uchun standart ontologiyalarni bir-

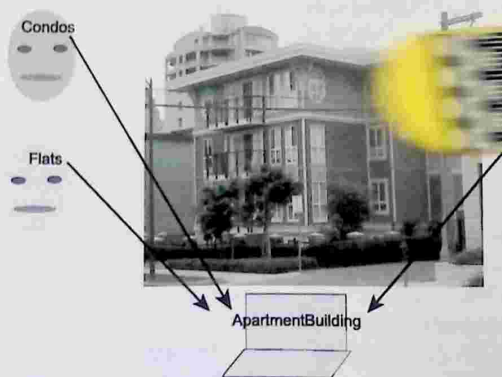
lashtirishga yoki o'z ontologiyalaridan boshqalar ontologiyalariga xaritalar ishlab chiqishga intiladilar.

Kichik bilim bazalari uchun konseptualizatsiya dizaynning boshida bo'lishi yoki hujjatda tabiiy til bilan belgilanishi mumkin. Ushbu norasmiy konseptualizatsiyaning belgilanishi katta tizimlarga mos kelmaydi, chunki konseptualizatsiya bo'lishishi kerak.

Falsafada **ontologiya** mavjud bo'lgan narsalarni o'rganishdir. Sun'iy intellektda **ontologiya** – bu ma'lumot tizimidagi ramzlarning ma'nolarini belgilovchi hujjatdir. Ya'ni bu konseptualizatsiyani belgilashdir. Bu nimalar va munosabatlar mavjud deb hisoblanganligini va ularga qanday atamalar ishlatilishini belgilovchi hujjatdir. Odatda, bu modelga olinadigan shaxslar turini belgilaydi, ishlatiladigan xususiyatlarni belgilaydi va ushbu lug'atni cheklovchi bir necha aksiomalarni beradi.

16.13-misol. Xaritada ko'rinishi mumkin bo'lgan shaxslar ontologiyasi «Apartment Building» ramzi kvartira binolarini anglatishini belgilashi mumkin. Ontologiya kvartira binosini aniqlamaydi, lekin uni boshqalar tushunishi uchun yetarlicha ta'riflaydi. Boshqalar, bunday binolarni «Condos», «Flats» or «Apartment Complex» deb atashi mumkin bo'lganida, ontologiyada mos ramzni topishlari kerak (16.3-rasm). Ya'ni berilgan bir tushuncha uchun odamlar ramzni topishlari va berilgan ramz uchun ular nima anglatishini aniqlashlari kerak.

Ontologiya ba'zi ramzlarning foydalanishini cheklovchi aksiomalarni ham berishi mumkin. Masalan, u kvartira binolari binolar, ya'ni inson tomonidan yaratilgan obyektlar ekanligini belgilashi mumkin. U, shuningdek, binolar hajmi bo'yicha ba'zi cheklovlar qo'yishi mumkin, shuning uchun kutilmagan narsalar, binolar yoki shaharlar bo'lmaydi.



16.3-rasm. Konseptualizatsiyadan ramzga o'tish

Unda bino bir vaqtning o'zida geografik jihatdan bir-biridan ajratilgan ikkita joyda bo'lishi mumkin emasligi ko'rsatilishi mumkin (shuning uchun agar siz binoning bir qismini buzib, uni boshqa joyga ko'chirsangiz, u endi bitta bino bo'lmaydi). Ko'p qavatli uylar binolar bo'lganligi sababli, ushbu cheklovlar ko'p qavatli uylarga ham tegishli.

Ontologiyalar odatda ma'lum bir dasturdan mustaqil ravishda yoziladi va ko'pincha belgilar ma'nosiga mos keladigan jamoaning ishtirokini o'z ichiga oladi. Ontologiya quyidagilardan iborat:

- * ma'lumotlar bazasida taqdim etilishi kerak bo'lgan obyektlar toifalarining lug'ati (sinflar va xususiyatlar).

- * kategoriyalarni tashkil qilish, masalan, subclassof yoki subPropertyOf (bu yerda s xususiyati p xususiyatining yordamchi xususiyati).

- * ba'zi belgilarning ta'rifini ularning taxmin qilingan ma'nosini yaxshiroq aks ettirish uchun cheklaydigan aksiomalar to'plami – masalan, ba'zi bir mulk tranzitiv yoki mulk doirasi va diapazoni yoki mulk har bir shaxs uchun qabul qilishi mumkin bo'lgan qiymatlar soniga cheklovlar. Ba'zida munosabatlar boshqa munosabatlar nuqtayi nazaridan aniqlanadi, ammo oxir-oqibat, bu munosabatlar aslida aniqlanmagan ibtidoiy munosabatlarga aylanadi.

Ontologiyada rivojlanish vaqtida noma'lum shaxslar ko'rsatilmagan. Masalan, binolarning ontologiyasi odatda haqiqiy binolarni o'z ichiga olmaydi. Ontologiyada belgilangan va birgalikda ishlatilishi kerak bo'lgan shaxslar, masalan, haftaning kunlari yoki ranglari ko'rsatilgan.

16.14-misol. Uy-joy topish uchun mo'ljallangan savdo agentini ko'rib chiqadi. Foydalanuvchilar qaysi turarjoyni xohlashlarini tasvirlash uchun bunday agentdan foydalanishlari mumkin. Savdo agenti tegishli turarjoyni topish uchun bir nechta bilim bazalarini qidirishi yoki har qanday mos turarjoy mavjud bo'lganda foydalanuvchilarga xabar berishi mumkin. Ontologiya foydalanuvchi uchun belgilarning ma'nosini aniqlash va bilim bazalarining o'zaro ta'sirini ta'minlash uchun zarurdir. Bu foydalanuvchi ehtiyojlari va bilim bazalari o'rtasida semantik aloqani ta'minlaydi.

Bunday hududda turarjoy binolari ham turarjoy binolari, ham ko'p qavatli uylar bo'lishi mumkin. Ko'p qavatli uyda uy yoki kvartirani ijaraga olishni taklif qilish oqilona bo'lishi mumkin bo'lsa-da, butun binoni ijaraga olmoqchi ekanligini ko'rsatmagan har bir kishiga ko'p qavatli uyni ijaraga berishni taklif qilish oqilona bo'lmazligi mumkin. «Uy-joy birligi» to'plam sifatida belgilanishi mumkin.

Turarjoy birligi ijara agentligi ijaraga beradigan narsa bo'lishi mumkin. Muayyan bosqichda dizayner kerak bo'lishi mumkin. Uyda ijaraga olingan xona turarjoy birligi yoki hatto alohida ijaraga olingan umumiy xonaning bir qismi turarjoy birligi ekanligiga qaror qiling. Ko'pincha chegara holatlari – dastlab kutilmasligi mumkin bo'lgan holatlar aniq belgilanmagan, ammo ontologiya rivojlanib borishi bilan aniqroq bo'ladi.

Ontologiya haqiqiy uylar yoki kvartiralarining tavsiflarini o'z ichiga olmaydi, chunki ontologiya aniqlangan paytda dizaynerlar ontologiyada qaysi uylar tasvirlanishini bilishmaydi. Ontologiya haqiqiy arzon uy-joylarga qaraganda ancha sekin o'zgaradi.

Ontologiyaning asosiy maqsadi belgilarning ma'nosini hujjatlashtirishdir – belgilar (kompyuterda) va tushunchalar (kimningdir boshida) o'rtasidagi xaritalashdir. Belgini olgandan so'ng, odam nimani anglatishini aniqlash uchun ontologiyadan foydalanishi mumkin. Agar kimdir taqdim etilishi kerak bo'lgan tushunchaga ega bo'lsa, ontologiya mos belgini topish yoki konsepsiya mavjud emasligini aniqlash uchun ishlatiladi. Aksiomalar yordamida erishilgan ikkinchi darajali maqsad – bu qiymatlarning ba'zi kombinatsiyasi mos kelmasligini aniqlashdir. Ontologiyani qurishda asosiy muammo tushunchalarni inson ularni kompyuterdagi belgilarga aylantirishi va kompyuter taqdim etilgan faktlardan yangi foydali bilimlarni chiqarishi mumkin bo'lgan tarzda tashkil etishdir.

16.3.1. Tavsif mantiqi

Zamonaviy ontologiya tillari, masalan, **OWL tavsif** mantiqlari asosida qurilgan. Tavsif mantiqi klasslar, xususiyatlar va shaxslarni tavsiflash uchun ishlatiladi. Tavsif mantiqining asosiy g'oyalardan biri quyidagilarni ajratishdir:

- **terminologik bilim bazasi (TBox)** terminologiyani tavsiflaydi; bu belgilar nima deganini aniqlaydi;

- **tasdiqlovchi bilim bazasi (ABox)** ba'zi bir vaqtda nima to'g'ri ekanini aniqlaydi.

Odatda, terminologik bilim bazasi tizimning dizayn vaqtida o'rnatiladi va ontologiyani aniqlaydi. U faqat terminologiyaning ma'nosi o'zgaradi, bu esa nisbatan kam uchraydi. Tasdiqlovchi bilim bazasi odatda vaziyatga xos bilimlarni o'z ichiga oladi va faqat ishga tushirish vaqtida ma'lum bo'ladi.

Tasdiqlovchi bilim bazasini aniqlash uchun odatda uchliklardan foydalaniladi va terminologik bilim bazasini aniqlash uchun OWL kabi til qo'llaniladi.

Veb ontologiya tili (web ontology language – OWL) domenlarni quyidagi tushunchalar asosida tavsiflaydi:

- **Shaxslar (Individuals)** – ta'riflanayotgan dunyodagi narsalar (masalan, ma'lum bir uy yoki ma'lum bir shaxs bo'lishi mumkin).

- **Sinflar (Classes)** – shaxslar to'plami. Klass – bu ushbu toifaga kiradigan barcha haqiqiy yoki potensial narsalar to'plami. Masalan, «Uy» sinfi, uy deb tasniflanadigan barcha narsalar to'plami bo'lishi mumkin, faqat qiziqish doirasida mavjud bo'lgan uylar emas.

– **Xususiyatlar** (Properties) – shaxslar va boshqa shaxslar yoki qiymatlar oʻrtasidagi ikkitemonlama munosabatlarni tavsiflash uchun ishlatiladi. **Maʼlumotlar turi** xususiyati oddiy maʼlumotlar turlari, masalan, butun sonlar, satrlar yoki sanalar kabi qiymatlarga ega. Masalan, «koʻcha nomi» koʻcha va satr orasidagi maʼlumotlar turi xususiyati boʻlishi mumkin. **Obyekt xususiyati** boshqa shaxslar qiymatlariga ega. Masalan, «yonma-yon» ikkita uy orasidagi xususiyat boʻlishi mumkin va «koʻchada» uy va koʻcha oʻrtasidagi xususiyat boʻlishi mumkin.

OWL bir necha variantlarda keladi, ularning har biri klasslar va xususiyatlar ustiga qoʻyilgan cheklovlar va qanday qilib samarali amalga oshirilishi mumkinligi bilan farqlanadi. Masalan, OWL-DLda klass shaxs yoki xususiyat boʻla olmaydi va xususiyat shaxs emas. OWL-Full da esa shaxslar, xususiyatlar va klasslar toifalari ajralmas boʻlishi shart emas. OWL-DL uchta profilga ega boʻlib, ular maʼlum dasturga moʻljallangan va kerak boʻlmagan tuzilmalarni qoʻllamaydi, bu esa xulosa chiqarishni sekinlashtiradi. OWL 2 EL katta biosalamatlik ontologiyalari uchun moʻljallangan boʻlib, boy tuzilma tavsiflarini qoʻllashga imkon beradi. OWL 2 QL maʼlumotlar bazasini soʻrov tillarining old qismi sifatida ishlatilish uchun moʻljallangan. OWL 2 RL esa qoidalar muhim boʻlgan holatlar uchun moʻljallangan tildir.

OWL noyob ismlar taxminini qabul qilmaydi; ikki nom albatta turli shaxslar yoki turli klasslarni ifodalaydi, degan maʼnoni anglatmaydi. Shuningdek, OWL toʻliq bilim taxminini qabul qilmaydi; barcha tegishli faktlar keltirilgan deb hisoblamaydi.

C_k – sinflar, p – xususiyat, I_k – shaxslar va n – butun sonidir. #S – S toʻplamidagi elementlar sonini anglatadi.

Sinf	Sinf oʻz ichiga oladi
owl: Thing	barcha shaxslar
owl: Nothing	hech qanday shaxs (boʻsh toʻplam)
owl: ObjectIntersectionOf($C_1, \dots, C_k$)	$C_1 \cap \dots \cap C_k$ dagi shaxslar
owl: ObjectUnionOf($C_1, \dots, C_k$)	$C_1 \cup \dots \cup C_k$ dagi shaxslar
owl: ObjectComplementOf(C)	C dagi boʻlmagan shaxslar
owl: ObjectOneOf($I_1, \dots, I_k$)	$I_1, \dots, I_k$ shaxslar
owl: ObjectHasValue(p, v)	p xususiyatida v qiymatiga ega boʻlgan shaxslar; yaʼni $\{x : p(x, v)\}$
owl: ObjectAllValuesFrom(p, C)	p xususiyatida barcha qiymatlari C dagi shaxslar; yaʼni $\{x : \forall y p(x, y) \rightarrow y \in C\}$
owl: ObjectSomeValuesFrom(p, C)	p xususiyatida C dagi baʼzi qiymatlarga ega boʻlgan shaxslar; yaʼni $\{x : \exists y \in C \text{ shartki } p(x, y)\}$
owl: ObjectMinCardinality(n, p, C)	p bilan bogʻlangan, kamida n shaxs C sinfiga tegishli boʻlgan shaxslar x ; yaʼni $\{x : \#\{y : p(x, y) \text{ va } y \in C\} \geq n\}$
owl: ObjectMaxCardinality(n, p, C)	p bilan bogʻlangan, eng koʻpi bilan n shaxs C sinfiga tegishli boʻlgan shaxslar x ; yaʼni $\{x : \#\{y : p(x, y) \text{ va } y \in C\} \leq n\}$
owl: ObjectHasSelf(p)	$p(x, x)$ boʻlgan shaxslar x ; yaʼni $\{x : p(x, x)\}$

16.4-rasm. Baʼzi OWL ichki sinflari va sinf konstruktorelari

16.4-rasm ba'zi primitiv sinflar va sinf konstruktorlarini ko'rsatadi. Bu rasmda sinfdagi individual to'plamlarini aniqlash uchun to'plam yozuvi (set notation) ishlatilgan. 16.5-rasmda OWLning ba'zi primitiv predikatlari berilgan. «owl:» prefiksi OWL uchun standart IRI qisqartmasi hisoblanadi.

Ushbu rasmlarda $p(x, y)$ bu triple. OWL ba'zi terminologiyani aniqlaydi, bu terminologiya predikatlarining ma'nosini aniqlash uchun ishlatiladi, nafaqat sintaksis. Predikatlar turli xil sintaksislarda ishlatilishi mumkin, masalan, XML, uchlik yoki funksional yozuvda.

16.15-misol. Funksional yozuvda sinf konstruktorining misoli:

ObjectHasValue (country of citizenship, Q16)

bu Kanada fuqarolarini (Q16) o'z ichiga olgan sinfdir.

OWL quyidagi predikatlarga ega bo'lib, ularga belgilangan talqin berilgan. Bu yerda C_k sinflar, p_k xususiyatlar va l_k individlar; x va y universal kvantorli o'zgaruvchilar:

Jumla	Ma'nosi
rdf: type (I, C)	$I \in C$ I individ C sinfiga tegishli
owl: ClassAssertion (C, I)	$I \in C$ (I individ C sinfiga tegishli)
rdfs: subclassOf (C_1, C_2)	$C_1 \subseteq C_2$ (C_1 sinfi C_2 sinfiga qisman kiradi)
owl: SubClassOf (C_1, C_2)	$C_1 \subseteq C_2$ (C_1 sinfi C_2 sinfiga qisman kiradi)
rdfs: domain (p, C)	Agar $p(x, y)$ bo'lsa, unda $x \in C$
owl: ObjectPropertyDomain (p, C)	Agar $p(x, y)$ bo'lsa, unda $x \in C$
rdfs: range (p, C)	Agar $p(x, y)$ bo'lsa, unda $y \in C$
owl: ObjectPropertyRange (p, C)	Agar $p(x, y)$ bo'lsa, unda $y \in C$
owl: EquivalentClasses ($C_1, C_2, \dots, C_k$)	$C_i \equiv C_j$ (har bir i va j uchun C_i va C_j ekvivalent)
owl: DisjointClasses ($C_1, C_2, \dots, C_k$)	$C_i \cap C_j = \emptyset$ (har bir i va j uchun C_i va C_j disjoint)
rdfs: subPropertyOf (p_1, p_2)	$p_1(x, y)$ $p_2(x, y)$ ni bildiradi
owl: EquivalentObjectProperties (p_1, p_2)	$p_1(x, y)$ faqat va faqat $p_2(x, y)$ bo'lsa
owl: DisjointObjectProperties (p_1, p_2)	$p_1(x, y)$ $p_2(x, y)$ emasligini bildiradi
owl: InverseObjectProperties (p_1, p_2)	$p_1(x, y)$ faqat va faqat $p_2(y, x)$ bo'lsa
owl: SameIndividual ($I_1, \dots, I_n$)	$\forall j \forall k I_j = I_k$ (barcha j va k uchun nI_j va I_k bir xil)
owl: DifferentIndividuals ($I_1, \dots, I_n$)	$\forall j \forall k j \neq k$ bo'lsa, $I_j \neq I_k$
owl: FunctionalObjectProperty (p)	Agar $p(x, y_1)$ va $p(x, y_2)$ bo'lsa, unda $y_1 = y_2$
owl: InverseFunctionalObjectProperty (p)	Agar $p(x_1, y)$ va $p(x_2, y)$ bo'lsa, unda $x_1 = x_2$
owl: TransitiveObjectProperty (p)	Agar $p(x, y)$ va $p(y, z)$ bo'lsa, unda $p(x, z)$
owl: SymmetricObjectProperty	Agar $p(x, y)$ bo'lsa, unda $p(y, x)$ bo'ladi
owl: AsymmetricObjectProperty (p)	$p(x, y)$ $p(y, x)$ emasligini bildiradi
owl: ReflectiveObjectProperty (p)	Barcha x uchun $p(x, x)$ bo'ladi
owl: IrreflectiveObjectProperty (p)	Barcha x uchun $p(x, x)$ emas bo'ladi

16.5-rasm. RDF, RDFS va OWLda ichki (built-in) predikatlar

Q113489728 – OECD (Iqtisodiy hamkorlik va taraqqiyot tashkiloti) ga a'zo bo'lgan mamlakatlar sinfi bo'lib,

ObjectSomeValuesFrom (country of citizenship, Q113489728) – ushbu sinf OECDga a'zo bo'lgan mamlakatlardan birining fuqarosi bo'lgan odamlarni qamrab oladi. Agar biror shaxsda bir nechta fuqarolik bo'lsa, ular orasida kamida bir mamlakat OECDga a'zo bo'lishi kerak.

MinCardinality (2, country of citizenship, Q113489728) – bu sinf OECDga a'zo bo'lgan ikki yoki undan ortiq mamlakat fuqarosi bo'lgan shaxslarni o'z ichiga oladi.

Sinf konstruktorlari qandaydir individning ushbu sinfga a'zo ekanligini aytish yoki bir sinfning boshqa sinfga teng ekanligini aytish uchun ishlatiladi. OWLda aniq hukmlar mavjud emas. Bir to'plamdagi barcha elementlar p predikati uchun v qiymatiga ega ekanligini aytish uchun, S to'plami qiymati v bo'lgan barcha narsalar to'plamining qism to'plami ekanligini aytamiz.

OWL va RDF yoki RDFS bayonotlarining ba'zilarida bir xil ma'no mavjud. Masalan, `rdf:type (I, C)` bilan `owl:ClassAssertion (C, I)` bir xil ma'noga ega bo'ladi va `rdfs:domain` bilan `owl:ObjectPropertyDomain` (obyekt xossalari uchun) bir xil ma'noga ega bo'ladi. Ba'zi ontologiyalar ikkala ta'rifni ham ishlatadi, chunki ontologiyalar uzoq vaqt davomida ishlab chiqilgan va turli konvensiyalarni qabul qilgan ishtirokchilar tomonidan ishlab chiqilgan.

16.16-misol. Ko'p qavatli uyning Aristotel ta'rifini ko'rib chiqing. Aytishimiz mumkinki, ko'p qavatli uy bu ijaraga beriladigan bir nechta kvartiralarga ega bo'lgan ko'p xonadonli turarjoy binosidir. (Bu kvartiralar alohida sotiladigan kondominiumdan yoki faqat bitta kvartira bo'lgan uydan farq qiladi.) Aytaylik, bizda kichik sinf bo'lgan turarjoy binolari sinfi bor binolar bor.

Quyidagi kod funksional obyekt xossasi *number Of Units* ni, uning domeni *Residential Building* va {one, two, moreThanTwo} diapazoni bilan belgilaydi:

`FunctionalObjectProperty(numberOfUnits)`

`ObjectPropertyDomain(numberOfUnits, ResidentialBuilding)`

`ObjectPropertyRange(numberOfUnits,`

`ObjectOneOf(two, one, moreThanTwo))`.

Shunga o'xshab, *Residential Building* domeni va {rental, owner Occupied coop} diapazoni bilan ko'chmas mulkka funksional egalik huquqini aniqlash mumkin.

Ko'p qavatli uy – bu *Residential Building* bo'lib, unda ko'chmas mulk birliklari soni ikkitadan oshadi va egalik qiymati ijara qiymatiga teng. Buni OWLda ko'rsatish uchun biz a ni aniqlaymiz. Mulk sinfi *number Of Units* mulki uchun ikkitadan ortiq qiymatga ega bo'lgan mulklar sinfi, ijara qiymati mulkka egalik sifatida belgilangan mulklar sinfi va *Apartment Building* ushbu sinflarning keshishiga teng deb aytamiz.

Owl funksional sintaksisida bu quyidagi ko'rinishga ega:

`EquivalentClasses(ApartmentBuilding,`

ObjectIntersectionOf(
ResidentialBuilding,
ObjectHasValue(numberOfunits, moreThanTwo),
ObjectHasValue(ownership, rental))).

Ushbu ta'rif kvartira binolari haqida savollarga javob berish uchun ishlatilishi mumkin, masalan, mulk egaligi va birliklar soni. Kvartira binolari barcha xususiyatlarni yashash binolaridan meros oladi.

Avvalgi misolda mulk egaligi aniq belgilamaydi. Tizim mulk egaligi aslida nima ekanligini bilmaydi. Umid qilamizki, foydalanuvchi bu ma'noni biladi. Ontologiyani qabul qilmoqchi bo'lgan har bir kishi, mulk va klassni ishlatishda boshqa ontologiya foydalanuvchilari bilan mos kelishini ta'minlashi kerak.

OWLda bir mulk konstruktor mavjud, 'owl: *ObjectInverseOf*(*p*)', bu *p* ning teskari mulkidir, ya'ni bu p^{-1} mulk bo'lib, $p^{-1}(x, y)$ faqatgina $p(x, y)$ bo'lganda to'g'ri. E'tibor bering, bu faqat obyekt xususiyatlariga tatbiq etiladi. Ma'lumot turi mulklarida teskari mulklar yo'q, chunki ma'lumot turlari uchlikning subyekti bo'lishi mumkin emas.

Ushbu rasm va bayonotlar ro'yxati to'liq emas. Ma'lumot turidagi mulklarga mos keluvchi ma'lumot turidagi klasslar mavjud, kerak bo'lganda. Masalan, owl: *Data Some Values From* va owl: *Equivalent Data Properties* mos keladigan obyekt belgilari bilan bir xil ta'riflarga ega, ammo ma'lumotlar turi xususiyatlariga ishora qiladi. OWLda xususiyatlarni, izohlarni, annotatsiyalarni, versiyalashni va boshqa ontologiyalarni import qilish uchun boshqa konstruksiyalar ham mavjud.

Domen ontologiyasi – bu ma'lum bir qiziqish domeni haqidagi ontologiyadir. Aksariyat mavjud ontologiyalar ma'lum dasturlar uchun yozilgan tor domenlarda joylashgan. Bilim almashishni ta'minlash uchun domen ontologiyalarini yozish maqsadida rivojlangan bir nechta qoida mavjud:

- Agar imkon bo'lsa, mavjud bo'lgan yaxshi o'rnatilgan ontologiyadan foydalaning. Bu sizning bilim bazangiz bir xil ontologiyadan foydalangan boshqa bilim bazalari bilan o'zaro aloqada bo'lishini anglatadi.

- Agar mavjud ontologiya sizning ehtiyojlaringizga to'liq mos kelmasa, uni import qiling va unga qo'shiling. Yangi boshlamang, chunki mavjud ontologiyani ishlatgan odamlar sizning ontologiyangizni ishlatishda qiynalishadi va boshqalar ontologiyani tanlashda birini yoki boshqasini tanlashga to'g'ri keladi. Agar sizning ontologiyangiz boshqasini o'z ichiga olsa va yaxshilasa, ontologiyani qabul qilmoqchi bo'lganlar siznikini tanlaydilar, chunki ularning dasturi har ikkala ontologiyadan foydalanadiganlar bilan o'zaro aloqada bo'lishi mumkin.

- Ontologiyangiz qo'shni ontologiyalar bilan integratsiyalashganligiga ishonch hosil qiling. Masalan, dam olish maskanlari haqidagi ontologiya oziq-ovqat, plyajlar, ochiq havoda mashg'ulotlar va boshqalar haqidagi ontologiyalar bilan o'zaro ta'sir qilishi mumkin. Ehtimol, bir xil maqsadlar uchun tegishli ontologiyalar bilan bir xil terminologiyadan foydalanganligiga ishonch hosil qilishga harakat qiling.

– Yuqori darajadagi ontologiyalar bilan moslashishga harakat qiling (quyida ko‘ring). Bu boshqalarga o‘z bilimlarini siznikiga integratsiyalashni osonlashtiradi.

– Agar yangi ontologiyani loyihalashtirishingiz kerak bo‘lsa, boshqa potensial foydalanuvchilar bilan keng maslahatlashishingiz kerak. Bu ontologiyani eng foydali va qabul qilinishi ehtimoli yuqori qiladi.

– Nomenklatura qoidalariga rioya qiling. Masalan, klassni a‘zolarining birlik nomi bilan chaqiring. Misol uchun, klassni «Resort» deb atang, «Resorts» emas. «ResortConcept» deb atashga intilmaslik kerak (bu faqat kurort tushunchasi degan fikrda). Sinf va mulklarni nomlashda ularning qanday foydalanish kerakligini o‘ylang. «r1 Resorts» deb aytgandan ko‘ra, «r1 Resort» deb aytish yaxshiroq, bu «r1 ResortConcept»dan afzaldir.

– Oxirgi variant sifatida, ontologiyalar o‘rtasidagi moslikni belgilab qo‘ying. Ba‘zan ontologiyalar mustaqil ravishda rivojlanganda moslikni belgilash zarur bo‘lishi mumkin. Moslikni oldini olish eng yaxshi yo‘ldir. Bu ontologiyalardan foydalanishda bilimlarni yanada murakkablashtiradi, chunki bir narsani aytishning bir nechta usullari mavjud.

OWL ko‘pchilik ko‘rsatishni yoki o‘qishni xohlaganidan pastroq darajadadir. U mashinada o‘qiladigan spetsifikatsiya sifatida ishlab chiqilgan. OWL ko‘rinishini tahrirlashga imkon beradigan ko‘plab muharrirlar mavjud. Misol **Protégé** sifatida (<http://protege.stanford.edu/>)ni keltirish mumkin. Ontologiya muharriri quyidagi narsalarni qo‘llab-quvvatlashi kerak:

– U foydalanuvchilarga ontologiyalarni eng mantiqiy mavhumlik darajasida kiritish imkoniyatini berishi kerak.

– Foydalanuvchi foydalanmoqchi bo‘lgan konsepsiyani hisobga olgan holda, ontologiya muharriri ushbu konsepsiya uchun terminologiyani topishni osonlashtirishi yoki tegishli atama mavjud emasligini aniqlashi lozim.

– Har qanday foydalanuvchi uchun atamaning ma‘nosini aniqlash oson bo‘lishi muhim.

– Ontologiyaning to‘g‘riligini tekshirish imkon qadar oson bo‘lishi kerak (ya‘ni foydalanuvchining atamalar uchun mo‘ljallangan talqiniga muvofiq kelishini ta‘minlash).

– Boshqa foydalanuvchilar foydalanishi mumkin bo‘lgan ontologiya yaratilishi zarur. Bu shuni anglatadiki, u standartlashtirilgan tildan maksimal darajada foydalanishi kerak.

16.3.2. Yuqori darajadagi ontologiyalar

16.16-misol xaritalarda ko‘rinishi mumkin bo‘lgan narsalarga murojaat qilmoqchi bo‘lgan odamlar uchun kvartira binosi uchun soha ontologiyasini belgilaydi. Har bir soha ontologiyasi unga mos keladigan yuqori darajadagi ontologiya-

ni bilvosita yoki aniq tarzda taxmin qiladi. Kvartira binosi ontologiyasi binolarni aniqlashni nazarda tutadi.

Yuqori darajadagi ontologiya *hamma narsani* juda mavhum darajada belgilaydi. Yuqori darajadagi ontologiyaning maqsadi boshqa ontologiyalarni asoslash uchun foydali tasniflashni ta'minlashdir. Soha ontologiyalari yuqori darajadagi ontologiyaga qanday mos kelishini aniq belgilash bu ontologiyalarning integratsiyasini osonlashtirishni va'da qiladi. Ontologiyalarni integratsiyalash turli ontologiyalardan foydaladigan bir nechta bilim bazalariga murojaat qilishga imkon berish uchun zarurdir.

Yuqorida **mohiyat** (entity) mavjud. OWL (Veb ontologiya tili) iyerarxiyaning yuqori qismini **jism** deb ataydi. Aslida, hamma narsa mohiyatdir.

Soha ontologiyalarini aniqlash uchun ishlatiladigan yuqori darajadagi xususiyatlarning ba'zilar quyidagilardan iborat:

- Konkret yoki mavhum: Jismoniy obyektlar va voqealar konkret, matematik obyektlar va vaqt esa mavhumdir.

- Davomli yoki davriy:

- **Davomli** bu vaqtning har qanday lahzasida mavjud bo'lgan va vaqt davomida mavjud bo'lishda davom etadigan narsadir. Misollar: odam, barmoq, mamlakat, tabassum, gulning hidi va elektron pochta. Davomli har qanday vaqtda mavjud bo'lganda, uning qismlari ham mavjud bo'ladi. Davomli xususiyatlar vaqt davomida o'z identifikatsiyasini saqlab qoladi.

- **Davriy** xususiyatlar vaqt bo'ylab bo'linadigan narsadir. Misollar: hayot, bolalik, tabassum qilish, gulning ochilishi va elektron pochta xabarini yuborish. Farqni tushunishning bir usuli – obyektning tarkibiy qismlarini ko'rib chiqish: barmoq insonning bir qismi, lekin hayotning bir qismi emas; go'daklik hayotning bir qismi, lekin shaxsiyatning bir qismi emas. Davomli xususiyatlar davriylarida ishtirok etadi. Uzoq vaqt davom etadigan jarayonlar va ma'lum bir vaqtda sodir bo'ladigan hodisalar ikkalasi ham o'z-o'zidan sodir bo'ladi.

Qit'a va sodir bo'layotgan narsalarning dioxotomiyasiga muqobil bu to'rt o'lchovli yoki doimiy ko'rinishdir. Bunda obyektlar fazo-vaqt ichida mavjud bo'ladi, shuning uchun inson makon va vaqtdagi trayektoriyadir va ular o'rtasida farq yo'q. Istalgan vaqtda odam to'rt o'lchovli trayektoriyaning oniy tasviridir.

Tobe yoki mustaqil: **mustaqil doimiylik** – bu o'z-o'zidan mavjud bo'lishi mumkin bo'lgan yoki boshqa mavjudotning bir qismi bo'lgan hodisa. Masalan, inson, yuz, qalam, gul, mamlakat va atmosfera mustaqil doimiylik. **Tobe doimiylik** faqat boshqa mavjudot tufayli mavjud va bu mavjudotning bir qismi emas. Masalan, tabassum, kulish qobiliyati yoki og'zingizning ichki qismi yoxud o'rtasidagi mulkchilik munosabati shaxs va telefon faqat boshqa obyekt yoki narsalarga nisbatan mavjud bo'lishi mumkin. E'tibor bering, boshqa obyektning bir qismi mustaqil kengaytmadir. Masalan, bir yurak tanasiz mavjud bo'lmaydi, lekin u tanadan ajratilishi mumkin va hali ham mavjud bo'ladi. Bu, mushukning tabassumidan farq qiladi; mushukning tabassumini ajratib olish mumkin emas.

Sinflar va tushunchalar

Ontologiyani belgilashda sinflarni **tushunchalar** deb atash juda oson bo'lishi mumkin, chunki belgilar tushunchalarni ifodalaydi. Ichki tasvirlardan obyektlar yoki belgilar ifodalaydigan munosabatlarga mos keladigan xaritalashdir.

Masalan, unicornlar sinfini «unicornConcept» deb atash oson bo'lishi mumkin, chunki unicornlar yo'q, faqat unicorn tushunchasi mavjud. Ammo unicornlar va unicorn g'oyasi juda boshqacha narsalardir; biri hayvon, ikkinchisi esa bilimning kichik sinfidir. Unicornning to'rtta oyog'i va boshidan chiqadigan shoxi bor. Unicorn tushunchasining esa oyoqlari yoki shoxi yo'q. Agar ontologiyalar haqidagi universitet darsida unicorn paydo bo'lgan bo'lsa, bu sizni juda hayratga solardi, ammo unicorn tushunchasi paydo bo'lgan bo'lsa, bu sizni ajablantirmasligi kerak. Unicornlar mavjud emas, ammo unicorn tushunchasining ko'plab misollari mavjud. Agar siz unicornni nazarda tutsangiz, «unicorn» so'zini ishlatishingiz kerak. Agar unicorn g'oyasini nazarda tutsangiz, «unicorn tushunchasi» so'zini ishlatishingiz kerak. Siz unicorn tushunchasi to'rtta oyoqqa ega deyolmaysiz, chunki bilimning misollarida oyoqlar bo'lmaydi; oyoqlarga ega bo'lganlar hayvonlar, mebellar va ba'zi robotlardir.

Yana bir misol: yer po'stining bir qismi bo'lgan tektonik plitani ko'rib chiqamiz. Plitalar million yillar davomida mavjud bo'lgan. Plita tushunchasi esa yuz yildan kamroq vaqt davomida mavjud bo'lgan. Kimdir o'z ongida tektonik plita tushunchasiga ega bo'lishi mumkin, ammo ular o'z ongida tektonik plitani joylashtira olmaydilar. Tektonik plita va tektonik plita tushunchasi juda boshqacha narsalar bo'lib, juda boshqacha xususiyatlarga ega. Agar siz «*tektonik plita*»ni nazarda tutsangiz, «tektonik plita tushunchasi»ni ishlatmang va aksincha.

Obyektlarni tushunchalar deb atash ontologiyalarni yaratishda keng tarqalgan xato hisoblanadi. Garchi narsalarni istalgan nom bilan atashda erkin bo'lsangiz ham, boshqa odamlar sizning ontologiyangizni qabul qilishi foydali bo'ladi. Agar bu ularga mantiqiy bo'lmasa, ular uni qabul qilmaydilar.

Mohiyatga tobe davriylik bu obyektga bog'liq voqea – bu jarayon yoki voqea-dir. **Jarayon** – bu vaqt o'tishi bilan sodir bo'ladigan, vaqtinchalik qismlarga ega bo'lgan va davom etayotgan narsalarga bog'liq bo'lgan narsa. Masalan, Joening hayoti go'daklik, bolalik, o'smirlik va kattalik kabi bo'laklarga ega bo'lib, doimiy bir entitet – Joega bog'liqdir. Dam olish kuni, elektron pochta yozish va laboratoriyani tozalayotgan robot – bularning hammasi jarayonlardir.

Voqea – bir lahzada sodir bo'ladigan narsa bo'lib, ko'pincha jarayon chegarasi hisoblanadi. Masalan, 2022-yilgi FIFA Jahon kubogi finalidagi birinchi gol – to'p chiziqidan o'tganda sodir bo'ladigan voqea bo'lib, u jamoa bilan bog'liq jarayonning yakuni sifatida ko'rilishi mumkin.

– *Bog'langan yoki tarqoq*: Yashayotgan qush – bir butun bog'langan narsa, lekin qushlar to'dasi bir necha qushlardan iborat bo'lgan tarqalgan entitet hisob-

lanadi. 2024-yil mart oyi – bog‘langan yagona narsa, lekin GMT vaqti bilan soat 3:00 dan 4:00 gacha bo‘lgan seshanba kunlari – tarqalgan vaqt mintaqasidir.

– *Moddiy yoki nomoddiy*: Mustaqil doimiylik **moddiy mohiyat** yoki **nomoddiy mohiyat** bo‘lishi mumkin. Moddiy mohiyatning bir qismi moddiy bo‘ladi. Moddiy mohiyatlar fazoda joylashgan va fazoda harakat qilishi mumkin. Ularga misollar: odam, futbol jamoasi, Everest tog‘i va Katrina dovuli. Nomoddiy mohiyat esa mavhumdir. Unga misollar: o‘tgan dushanba kuni yuborgan birinchi elektron pochtagiz, reja va eksperimental protokol.

E-pochta olish uchun sizga uning jismoniy ko‘rinishi kerak (masalan, smartfoningizda matn sifatida yoki nutq sintezatori orqali aytilgan holda), lekin e-pochta o‘sha jismoniy ko‘rinish emas; boshqa jismoniy ko‘rinish ham bir xil e-pochta bo‘lishi mumkin. Ushbu dikotomiya orqali turli toifalarni shakllantirish mumkin. Bitta bir butun va bir-biriga bog‘langan moddiy entitet obyekt hisoblanadi. Obyekt vaqt davomida o‘z kimligini saqlaydi, hatto u qismlarini qo‘lsa yoki yo‘qotib qolsa ham (masalan, sochlarini yo‘qotgan odam, bir e‘tiqod yoki hatto bir oyoq – bular hamon bir xil odamdir). Odam, stul, tort yoki kompyuter – bularning barchasi obyektlardir. Odamning chap oyoqi (agar u hali ham odamga bog‘langan bo‘lsa), futbol jamoasi yoki ekvator obyektlar emas. Agar robotga uchta obyekt topish so‘ralgan bo‘lsa, stul olib kelib, orqa, o‘tirish joyi va chap old oyoqni uchta obyekt deb da‘vo qilish ma‘noli bo‘lmaydi.

Eng yuqori darajadagi ontologiyani loyihalash qiyin. Bu har qanday kishi uchun qoniqarli bo‘lmashligi mumkin. Har doim muammoli holatlar bo‘lib qoladi. Ayniqsa, chegaraviy holatlar ko‘pincha yaxshi belgilab olinmaydi. Biroq, standart eng yuqori darajadagi ontologiyani ishlatish ontologiyalarni bir-biriga bog‘lashda yordam berishi kerak.

16.4. Ijtimoiy ta’sir

Ma’lumotlar bo‘yicha bashoratlar qilish uchun **ma’lumotlarning kelib chiqishi** yoki **ma’lumotlar tarixi** – ma’lumot qayerdan kelgan va qanday manipulyatsiya qilingan – yaxshi bashorat bilan almoqona bashorat o‘rtasida farq yaratishi mumkin. Kelib chiqish odatda metama’lumot sifatida yoziladi – ma’lumotlar haqidagi ma’lumotlar – quyidagilarni o‘z ichiga oladi:

- Har bir ma’lumotni kim yig‘di? Ularning vakolatlari qanday?
- Ma’lumotlarni kim yozib oldi?
- Ma’lumotlarni yig‘ish uchun qaysi protokol ishlatilgan? Ma’lumot tasodifiy ravishda tanlanganmi yoki qiziqish yoki boshqa sababga ko‘ra tanlanganmi?
- Nazoratlar nimalar edi? Nima manipulyatsiya qilingan va qachon?
- Qanday sensorlar ishlatilgan? Ularning ishonchliligi va ishlash doirasi qanday?
- Ma’lumotlar ustida qanday qayta ishlashlar amalga oshirildi?

Bunday metama'lumot atrof-muhit, geografik va ijtimoiy ma'lumotlar – Yer haqidagi ma'lumotlar – odamlar tomonidan yig'ilgan va atrof-muhit qaror qabul qilishda ishlatiladigan ma'lumotlar uchun zarurdir [Gil va boshq., 2019].

Agar ma'lumot FAIR (topish oson, ochiq va aniq) bo'lishi kerak bo'lsa, bu ayniqsa muhimdir [Wilkinson va boshq., 2016]:

– *Topish mumkin* – metama'lumotlar yagona barqaror identifikatorlardan, masalan, IRIlardan foydalanadi.

– *Kirish mumkin* – ma'lumotlar bepul va ochiq protokollar orqali mavjud va metama'lumotlar ma'lumotlar mavjud bo'lmagan taqdirda ham kirish mumkin.

– *O'zaro ishlash imkoniyati mavjud* – lug'at rasmiy bilimni ifodalash tillari (ontologiyalar) yordamida aniqlanadi.

– *Qayta ishlatish mumkin* – ma'lumotlar boy metama'lumotlardan, jumladan kelib chiqishdan foydalanadi va hamjamiyat ma'lumotlardan foydalanishi uchun mos ochiq litsenziyaga ega bo'ladi.

Ushbu tamoyillarga asoslangan ma'lumotlar omborlari ko'plab sohalar, jumladan, Yer kuzatuvlari [NASA, 2022], ijtimoiy fanlar [King, 2007], hisoblash jarayonlari [Goble va boshq., 2020] va barcha tadqiqot sohaları [Springer Nature, 2022] uchun mavjud. FAIR ma'lumotlari zamonaviy ma'lumotlarga asoslangan ilm-fanning muhim qismidir; ammo, tijorat yoki harbiy sabablarga ko'ra o'zlarini raqobatdosh sifatida ko'rishni xohlaydigan ba'zi tadqiqotchilar FAIR yo'riqnomalariga amal qilmaslik uchun rag'batga ega bo'lishi mumkin.

Stodden va boshq. [2016], Gil va boshq. [2017] va Sikos va boshq. [2021] ma'lumotlar fanida takrorlanishni oshirish usullarini ko'rib chiqadi. Gebru va boshq. [2021] ma'lumot to'plami va uni ishlab chiqish jarayoni haqidagi 57 ta savolni taklif qiladilar.

16.5. Sharhlar

Ushbu bobdan o'rganishingiz kerak bo'lgan asosiy fikrlar quyidagilardir:

– Individual–xususiyat–qiymat uchliklari munosabatlarni ifodalash uchun moslashuvchan va universal ko'rinishni tashkil etadi.

– Ontologiyalar semantik o'zaro ishlash va bilimlarni ulashish imkonini beradi.

– OWL ontologiyalari individualar, sinflar va xususiyatlardan tashkil topgan. Sinf – haqiqiy va potensial individuallarning to'plami.

– Eng yuqori darajadagi ontologiya domen ontologiyalarini o'zaro ishlashini ta'minlash uchun asos yaratadi.

– Ontologiyalar asosida kelib chiqishi ko'rsatilgan ma'lumotlar omborlari zamonaviy ma'lumotlarga asoslangan ilm-fanda keng qo'llaniladi.

16.6. Manbalar va qo‘shimcha adabiyotlar bilan ishlash

Sowa [2000], Brachman va Levesque [2004] bilimni ifodalash haqida umumiy ma’lumot beradi. Davis [1990] oddiy mantiqiy fikr yuritishdagi bilimni ifodalash masalalari bo‘yicha keng qamrovli kirishdir. Brachman va Levesque [1985] ko‘plab klassik bilimni ifodalash maqolalarini taqdim etadi. Semantik tarmoqlar haqida umumiy ma’lumot uchun Woods [2007]ga qarang.

Hogan va boshq. [2021], Chaudhri va boshq. [2022] bilim grafiklarga keng qamrovli kirish taqdim etadi.

Ontologiyalarning falsafiy va hisoblash jihatlarini haqida umumiy ma’lumot olish uchun Smith [2003] va Sowa [2011] ishlariga murojaat qilish lozim.

Semantik veb va uning texnologiyalari Berners-Lee va boshq. [2001], Hendler va boshq. [2002], Antoniou va van Harmelen [2008], Allemang va boshq. [2020] tomonidan ta’riflangan. Janowicz va boshq. [2015] katta ma’lumotda semantikaning rolini tushuntiradi. Kendall va McGuinness [2019] zamonaviy ontologiya muhandisligi haqida umumiy ma’lumot beradi. OWL tavsifi OWL-2 ga asoslangan. W3C OWL Ishchi Guruhi [2012], Hitzler va boshq. [2012], Motik va boshq. [2012]. Krotzsch [2012] OWL 2 profil-larini tasvirlaydi. Baader va boshq. [2007] tavsifiy mantiq haqida umumiy ma’lumot beradi. Heath va Bizer [2011] bog‘langan ma’lumotlar tasavvuri haqida umumiy ma’lumot beradi. DBpedia [Auer va boshq., 2007], YAGO [Suchanek va boshq., 2007; Hoffart va boshq., 2013; Mahdisoltani va boshq., 2015], Wikidata [Vrandečić va Krötzsch, 2014] (<http://www.wikidata.org/>) va Knowledge Vault [Gabrilovich va boshq., 2014] – millionlab mohiyatlar haqidagi faktlarni ifodalash uchun uchliklar va ontologiyalardan foydalana-digan katta bilim bazalaridir.

Eng yuqori darajadagi ontologiya BFO, ya’ni Basic Formal Ontology 2.0 ga asoslangan bo‘lib, bu Smith [2015] va Arp va boshq. [2015] hamda Sowa [2000] tomonidan tavsiflangan. Boshqa eng yuqori darajadagi ontologiyalar DOLCE [Gangemi va boshq., 2003], Cyc [Panton va boshq., 2006] va SUMO [Niles va Pease, 2001; Pease, 2011]ni o‘z ichiga oladi. Yengil va keng qo‘llaniladigan ontologiya esa <http://schema.org> saytida mavjud. SNOMED klinik terminlari (SNO-MED CT) [IHTSDO, 2016] klinik amaliyotda ishlatiladigan katta tibbiy ontologi-yadir. Uni <http://browser.ihtsdotools.org/> saytida o‘rganishingiz mumkin.

16.7. Mashqlar

16.1-mashq. Tasavvur qilishingiz mumkin bo‘lgan ko‘plab qarindoshlik rishtalari mavjud. Masalan, ona, ota, buvi, ikki marta uzoq qarindosh va tabiiy otalik amakisi.

Ularning ba’zilari boshqalar orqali aniqlanishi mumkin, masalan:
 $\text{brother}(X, Y) \leftarrow \text{father}(X, Z) \leftarrow \text{natural paternal uncle}(Y, Z).$

$sister(X, Y) \leftarrow parent(Z, X) \leftarrow parent(Z, Y) \leftarrow$
 $female(X) \leftarrow different(X, Y).$
 $aka-uka(X, Y) \leftarrow ota(X, Z) \wedge ota\ tarafd\an\ amaki\ (chin)\ (Y, Z).$
 $opa-singil(X, Y) \leftarrow ota-ona(Z, X) \wedge ota-ona(Z, Y) \wedge$
 $ayol(X) \wedge har\ xil\ (X, Y).$

Turli munosabatlarni primitiv munosabat sifatida qabul qilgan holda, qarindoshlik munosabatlarining ikki xil tasvirini bering.

Primitiv qarindoshlik munosabatini «children» (Bolalar) munosabati yordamida ifodalashni ko'rib chiqing:

children(Mother, Father, Listofchildren).

bolalar(Ona, Ota, Bolalar ro'yxati).

Ushbu ifodaning yuqoridagi ikkita variant bilan solishtirganda qanday afzalliklari yoki kamchiliklari bo'lishi mumkin?

16.2-mashq. Sayohat sayti foydalanuvchilardan kelib tushgan mulohazalar va mehmonxonalar haqida ma'lumotni aks ettiruvchi ma'lumotlar bazasidan foydalanadi.

hotel(HotelId, Name, City, Provinceorstate, Country, Address)

reportedclean(HotelId, RoomNumber, Cleanliness, day(Year, Month, Day)).

Quyidagi faktlarni izohli lug'atlardan foydalangan holda uchlik belgisi yordamida qanday ifodalash mumkinligini ko'rsating:

hotel(h345, "The Beach Hotel", victoria, bc,

canada, "300 Beach St").

reported_clean(h345, 127, clean, day(2023,01,25)).

16.3-mashq. Christine Sinclair (Q262802) Britaniya Kolumbiyasining Burnabi (Q244025) shahrida tug'ilgan. U tug'ilgan viloyatning nomini uning tug'ilgan joyi bilan bog'laydigan Wikidatada uchraydigan uchliklarni keltiring. Natija Q262802 subyekt bo'lgan va «British Columbia» (en) bilan yakunlanadigan uchliklar ketma-ketligi bo'lishi kerak, bunda har bir uchlikning obyekti keyingi uchlikning subyekti hisoblanadi. Iltimos, Wikidata uchliklarini hamda inglizcha tarjimasini keltiring. Ma'lumotlarni <https://www.wikidata.org/wiki/Q262802> sahifasidan topishingiz mumkin. Birinchi uchlik:

Q262802 P19 Q244025. «Christina Sinclairning tug'ilgan joyi Burnabi edi».

Keyingi uchlikda esa Q244025 subyekt bo'ladi.

16.4-mashq. Andres Iniesta (Q43729) tomonidan 116-daqiqada kiritilgan 2010-yilgi FIFA Jahon chempionati finalida (Q208401) birinchi gol bilan bog'liq 10 ta uchlikni keltiring. Munosabatlarni 16.1-rasmdagi kabi chizing yoki Wikidata nomlaridan foydalangan holda uchliklarni yozing hamda ularning inglizcha tarjimasini keltiring. Uchliklar o'zaro bog'liq bo'lishi kerak (16.1-rasmda bo'lgani

kabi). Ma'lumotlarni <https://www.wikidata.org/wiki/Q208401> sahifasidan topishingiz mumkin.

16.5-mashq. Sem har qanday n -munosabatni $P(X_1, X_2, X_3, \dots, X_n)$ ni $n - 1$ ikkilik munosabatlar sifatida qayta ifodalash mumkinligini taklif qildi, ya'ni

$$P_1(X_1, X_2), P_2(X_2, X_3), P_3(X_3, X_4), \dots, P_{n-1}(X_{n-1}, X_n).$$

Siz Semga nima uchun bu yaxshi g'oya bo'lmashligini tushuntiring. Agar Sem buni qilishga urinib ko'rsa, qanday muammolar paydo bo'ladi? Muammo qayerda yuzaga kelishini ko'rsatish uchun misol keltiring.

16.6-mashq. Stolingizda tez-tez uchraydigan obyektlar uchun ontologiya yozing, bu stolingizni tartibga solish uchun mo'ljallangan robot uchun foydali bo'lishi mumkin. (a) Robot ko'ra oladigan va (b) vazifa uchun ajratilishi kerak bo'lgan toifalarni o'ylab ko'ring.

16.7-mashq. Faraz qiling, «plyaj kurorti» bu kurort yaqinidagi mehmonlar foydalanishi mumkin bo'lgan plyaj bo'lgan kurortdir. Plyaj dengiz yoki ko'l yaqinida bo'lishi kerak, bu yerda suzishga ruxsat beriladi. Kurortda uxlash va ovqatlanish joylari bo'lishi kerak. OWLda plyaj kurorti ta'rifini yozing.

16.8-mashq. Hashamatli mehmonxona ijaraga beriladigan bir nechta xonaga ega bo'lib, ularning har biri qulay va ko'rinishga ega bo'lishi kerak. Mehmonxonada bir nechta restoran ham bo'lishi lozim. Restoranlarda vegetarianlar va go'sht iste'mol qiluvchilar uchun menyu elementlari bo'lishi kerak.

(a) OWLda bu ta'rifga asoslanib, hashamatli mehmonxona ta'rifini bering. Aniq ko'rsatilmagan joylarda asosli farazlar qiling.

(b) Hashamatli mehmonxonadan kutishingiz mumkin bo'lgan yana uchta xususiyatni taklif qiling. Har biriga tabiiy tilda ta'rif bering va OWL spetsifikatsiyasini yozing.

16.9-mashq. Quyidagilar uchun har birini 16.3.2-mavzu top-level ontologiyasi orqali qanday tasniflanishini tushuntiring:

- (a) teringiz
- (b) ushbu bo'limning birinchi gapining oxiridagi nuqta
- (c) bolada ta'tildan oldin paydo bo'lgan hayajon
- (d) ta'tildan uyga qaytish
- (e) kompyuter dasturi
- (f) yozgi ta'til
- (g) telefonning jiringlashi
- (h) stoldagi chang
- (i) ofisingizni tozalash vazifasi
- (j) bir kishida gripp tashxisi
- (k) Fransiya.

Bu tajribadan kelib chiqib, top-level ontologiyani qanday o'zgartirish kerakligini taklif qiling va asoslab bering. Maxsus bo'lmagan kategoriyalar yoki boshqa muhim farqlar haqida o'ylang.

17-bob. NISBIY TA'LIMNI O'RGANISH VA EHTIMOLIY FIKRLASH

Ong – tabiiy tanlanish natijasida shakllangan neyron kompyuter bo'lib, u o'simliklar, hayvonlar, predmetlar va insonlar haqida sababiy va ehtimoliy mulohaza yuritish uchun kombinatorik algoritmlar bilan ta'minlangan.

Qonuniyatlarga ega bo'lgan har qanday koinotda, o'tmish haqida ma'lumotga asoslangan qarorlar tasodifiy qabul qilingan qarorlardan ko'ra yaxshiroq bo'ladi. Bu har doim shunday bo'lib kelgan va biz organizmlar, ayniqsa odamlar singari axborot jihatidan faol mavjudotlar, ehtimollik haqida o'tkir sezgini rivojlantiradi, deb o'ylagan edik. Ehtimollik nazariyasi asoschilari, xuddi mantiq asoschilari kabi, ular shunchaki sog'lom fikrni shakllantirishyapti deb o'ylaganlar.

– Steven Pinker [1997, 524, 343-b.]

Mashina yordamida o'qitish modellarida va avvalgi boblarda ko'rib chiqilgan ehtimollik modellarida, dunyo belgilar va tasodifiy o'zgaruvchilardan tashkil topgan. Pinker ta'kidlaganidek, biz odatda narsalar to'g'risida fikr yuritamiz. Narsalar xususiyatlar yoki tasodifiy o'zgaruvchilar emas; muayan hayvonning ehtimolligi haqida gapirish mantiqsiz, biroq uning alomatlariga asoslanib, u kasal ekanligini taxmin qilib, fikr yuritish mumkin. Ushbu bobda narsalar yoki obyektlar haqida qanday o'rganish va ular to'g'risida bashoratlar qilish usullari haqida so'z yuritiladi.

Tasavvur o'lehami eng yuqori darajada, **shaxslar (obyektlar)** va munosabatlar nuqtayi nazaridan mulohaza yuritishni o'z ichiga oladi. Munosabatlar nuqtayi nazaridan fikr yuritish, ma'lum bir obyektlardan mustaqil ravishda qurilishi mumkin bo'lgan ixcham tasavvurlarni yaratish imkonini beradi. Bunday yondashuv har bir obyekt haqida o'rganish va mulohaza qilish uchun ham qo'llanilishi mumkin. Ushbu bobda o'qitish va ehtimoliy fikrlashda qo'llaniladigan xususiyatlarga asoslangan ifodalar qanday qilib obyektlar va munosabatlar bilan ishlash uchun kengaytirilishi mumkinligi bayon etilgan. Relyatsion model uning mohiyati ma'lum bo'lgunga qadar va, binobarin, uning xarakteristikallari ma'lum bo'lgunga qadar uni qurish imkoniyatidan foyda ko'rishi mumkin.

Statistik relyatsion sun'iy intellekt yoki neyrosimvolik sun'iy intellekt relyatsion ma'lumotlar bazasi va ehtimol, metama'lumotlardan tashkil topgan relyatsion ma'lumotlarga asoslanib, munosabatlar haqida bashoratlar qilishni ko'zda tutadi. Statistik relyatsion sun'iy intellekt ma'lumotlarga asoslangan relyatsion basho-

ratlar uchun qo'llaniladigan umumiy atama hisoblanadi. Neyrosimvolik sun'iy intellekt esa bashorat qilish maqsadida neyron tarmoqlari va boshqa o'rnatilgan usullardan foydalanishni nazarda tutadi.

Bunga quyidagilar kiradi:

- Obyektlarning xususiyatlarini ularning boshqa xususiyatlari va ular bog'liq bo'lgan obyektlarning xususiyatlari asosida bashorat qilish.

- Obyektlarning xususiyatlari va o'zaro munosabatlariga asoslanib, ular o'rta-sidagi aloqalarni oldindan aytib berish.

- Eng oddiy holat yagona ikkilik munosabatni o'rganish bo'lib (17.2.1-mavzu), bu kino tavsiyalari kabi domenlarida foydali hisoblanadi. Bu 17.2.2-mavzu-da berilgan uchliklarni o'rganishga va 17.3-mavzuda umumiyroq munosabatlarni o'rganishga taalluqlidir.

- **Shaxsni** aniqlash, ya'ni tavsiflar bir xil obyektни ifodalayotganini bashorat qilish – tavsiflarning tengligi, masalan, qaysi iqtiboslar bir xil maqolalarga ishora qilishi yoki ikkita tavsif bir xil shaxsга tegishli ekanligini aniqlash (17.4-mavzu).

- **Mavjudlikni** bashorat qilish, ya'ni biror ta'rifga mos keladigan narsa yoki shaxs mavjud yoki yo'qligini aniqlash. Masalan, ma'lum bir xonada odam bor-yo'qligini aniqlash (17.4-mavzu).

Ma'lum bir turdagi barcha obyektlar **populyatsiya** deb ataladi.

Ushbu bobda diapazoni ma'lum bir mohiyat (masalan, inson yoki film) bo'lgan **xususiyat – munosabat** deb ataladi. Agar qiymatlar doirasi belgilangan to'plam, masalan, Boolean yoki haqiqiy sonlar bo'lsa, **xususiyat** atamasi ishlatiladi.

17.1. Relyatsion xususiyatlar va tasodifiy o'zgaruvchilar

7 va 8-boblarning o'zlashtirilishi hamda 9-bobning ehtimoliy mulohazalari xususiyatlar va tasodifiy o'zgaruvchilar asosida o'rganilgan edi. Obyektlar, xususiyatlar yoki munosabatlar belgilar yoki tasodifiy o'zgaruvchilar emas, ammo ular yordamida tasodifiy o'zgaruvchilarni hosil qilish mumkin.

Afsuski, «o'zgaruvchi» atamasi tasodifiy miqdorlar va mantiqiy o'zgaruvchilar uchun ishlatiladi, ammo bu tushunchalar o'zaro bog'liq emas; mantiqiy o'zgaruvchi biror-bir obyektни ifodalaydi, tasodifiy o'zgaruvchi esa olamlar yoki holatlar ustida amal qiluvchi funksiya hisoblanadi. Ushbu bob doimo ikkalasini farqlaydi; boshqa manbalar uchun esa qaysi biri nazarda tutilganini matn mazmunidan aniqlab olishingiz zarur. Bilimlar grafigidan (16.1-mavzu) olingan tasodifiy o'zgaruvchilar quyidagicha aniqlanadi:

Funksional xususiyatlar uchun har bir obyekt-xususiyat juftligiga mos keladigan tasodifiy o'zgaruvchi va funksional munosabatlar uchun har bir obyekt-munosabat juftligi uchun tasodifiy o'zgaruvchi mavjud. Eslatib o'tamiz, agar har bir subyekt uchun yagona obyekt mavjud bo'lsa, **p funksional** hisoblanadi; ya'ni, agar (x, p, y_1) va (x, p, y_2) bo'lsa, u holda $y_1 = y_2$ ga teng bo'ladi. Misol uchun,

agar bo'ʻy xususiyati funksional bo'lsa, Kristin Sinklarning (Wikidatada Q262802) 20 yoshdagi bo'ʻyi santimetrlarda haqiqiy qiymatli tasodifiy miqdor hisoblanadi. Shaxsning tug'ilgan oyi o'sha shaxsning kategoriyali tasodifiy o'zgaruvchisi hisoblanadi. Muayyan shaxsning biologik onasi esa insonlar domeni bo'yicha tasodifiy o'zgaruvchi sanaladi.

Biologik ona kabi funksional munosabat uchun (har bir uchlikning obyekti hisoblanadi) bashorat obyektlar bo'yicha ehtimollar taqsimotidan iborat bo'ladi. Bunday ehtimolliklar taqsimotini aholidan mustaqil ravishda aniqlab bo'lmaydi;

Buning o'rniga, ma'lum bir aholi uchun shaxslarning ehtimolini o'rganish yoki shaxslar hamda aloqalarning joylashuv funksiyasini aniqlash zarur.

- Funksional bo'lmagan xususiyatlar uchun har bir «subyekt-xususiyat-qiymat» yoki «subyekt-munosabat-obyekt» uchligiga tegishli Boolean mantiqiy tasodifiy o'zgaruvchi bor. Masalan, «*participated-in*» funksional bo'lmagan munosabat hisoblanadi va Q262802 *participated-in* Q181278 kabi uchliklar uchun (Kristin Sinkler 2020-yilgi yozgi Olimpiya o'yinlarida ishtirok etganmi yoki yo'qmi) Boolean tasodifiy o'zgaruvchisi mavjud.

Yanada umumiyroq munosabatlar uchun $r(X_1, \dots, X_k)$:

- Agar bitta argument, aytaylik X_k , boshqa argumentlarning funksiyasi bo'lsa, u holda har bir $r(e_1, \dots, e^{k-1})$ kortej uchun tasodifiy miqdor mavjud bo'ladi. Mazkur tasodifiy miqdorning aniqlanish sohasi X_k qabul qilishi mumkin bo'lgan qiymatlar to'plamidan iborat bo'ladi. Masalan, *rated* (U, M, R) munosabati, ya'ni R – foydalanuvchi U tomonidan M filmiga berilgan (1 dan 5 gacha bo'lgan) reyting ekanligini anglatadigan nisbat, har bir foydalanuvchi-film juftligi uchun tasodifiy o'zgaruvchini beradi. Bu o'zgaruvchining domeni mumkin bo'lgan reytinglar to'plamidan iborat bo'lib, $\{1, 2, 3, 4, 5\}$ ko'rinishida ifodalanadi. Muayyan foydalanuvchi va film uchun reyting bashoratini ikki xil masala sifatida ko'rib chiqish mumkin: regressiya masalasi sifatida, bunda bashorat haqiqiy son ko'rinishida ifodalanadi, yoki tasniflash masalasi sifatida, bunda bashorat 1 dan 5 gacha bo'lgan sonlar bo'yicha ehtimollar taqsimoti hisoblanadi.

- Aks holda, har bir $r(e_1, \dots, e_k)$ kortej uchun Boolean tasodifiy o'zgaruvchisi mavjud bo'ladi.

Quyida keltirilgan tavsifda funksional holat Boolean bo'lmagan bashorat bilan $k - 1$ argumentning munosabati sifatida ko'rib chiqiladi.

K argumentlarning nisbati n ta obyektlar uchun nk ta tasodifiy miqdorni beradi, bunda n obyektlar sonini anglatadi. Bu argumentlarning qo'llanish sohalariga qarab kamroq bo'lishi mumkin: masalan, agar birinchi argument foydalanuvchilar, ikkinchi argument esa filmlar bo'lsa, u holda tasodifiy miqdorlar soni foydalanuvchilar sonining filmlar soniga ko'paytmasiga teng bo'ladi. Agar tasodifiy miqdorlar orasida ixtiyoriy o'zaro bog'liqlik mavjud bo'lsa, unda faqat bitta k argumentli Boolean munosabati uchun $2^{nk} - 1$ ehtimollikni belgilash zarur bo'ladi.

Tasodifiy o'zgaruvchilar sonining kombinatorik portlashini oldini olish lifted inference (bekor qilingan xulosa)ning masalasidir.

17.2. Embeddingga asoslangan modellar

17.2.1. Ikkiлик munosabatni o'rganish

Faraz qiling, sizda binar munosabatlar ma'lumotlar bazasi mavjud bo'lib, unda ikkala argument ham mohiyat hisoblanadi. Maqsad esa boshqa kortejlarning to'g'ri yoki noto'g'riligini bashorat qilishdan iborat. 7 va 8-boblarda tavsiflangan mashina yordamida o'qitish usullarini to'g'ridan to'g'ri qo'llab bo'lmaydi, chunki obyektlar ma'nosiz nomlar identifikatorlar yordamida ifodalangan. Boshqa munosabatlar aniqlanmagan, ammo juftlik uchun raqamli bashorat ham mumkin bo'lgan holat uchun bu ishini bajaradigan algoritm quyida tasvirlangan. Bu, foydalanuvchining film kabi biror mahsulotga beradigan bahosini oldindan aytib berishning aniq misoli orqali tushuntiriladi.

Tavsiya qilish tizimida foydalanuvchilar o'zlariga yoqishi mumkin bo'lgan narsalar haqida **shaxsiy tavsiyalarni** oladilar. Tavsiya tizimlarining bir usuli – foydalanuvchining ma'lum bir element bo'yicha bahosini bashorat qilish. Bu bashorat o'xshash foydalanuvchilar yoki o'xshash elementlarning reytinglariga asoslanib, birgalikda filtrlash (collaborative filtering) deb ataladigan usul orqali amalga oshiriladi.

17.1-misol MovieLens (<https://movielens.org/>) foydalanuvchilardan film haqidagi reytinglarini olib, ularni tavsiya qilish tizimidir. Baholash 1 ta yulduzchadan 5 ta yulduzchagacha bo'lib, 5 ta yulduzcha eng yaxshi hisoblanadi. 17.1-jadvalda bitta ma'lumotlar to'plamining dastlabki bir nechta qatori ko'rsatilgan. Unda har bir foydalanuvchiga hamda har bir elementga (filmga) yagona raqam beriladi. Vaqt belgisi – 1970-01-01 UTC dan boshlab soniyalarda o'lchanadigan Unix standartidir. Bunday ma'lumotlardan foydalanuvchiga yoqishi mumkin bo'lgan boshqa filmlarni bashorat qilish uchun tavsiyalar tizimi foydalanish mumkin.

Tavsiya tizimi uchun **tavsiyalarni shaxsiylashtirish** usullaridan biri foydalanuvchiga eng mashhur n ta elementni taqdim etishdir. Bu yerda n musbat butun son bo'lib, masalan, 10 ga teng bo'lishi mumkin. Bu ishini, avvalo har bir element uchun foydalanuvchi reytingini baholab, so'ngra eng yuqori bashorat qilingan reytingga ega elementlarni taqdim etish orqali amalga oshirish mumkin. Umuman olganda, bu uncha yaxshi yechim emas, chunki barcha asosiy elementlar bir-biriga juda o'xshash ketishi mumkin. Shuningdek, tizim elementlar to'plamini tanlashda xilma xillikni ham e'tiborga olishi kerak.

Yuqoridagi misolda tovarlar sifatida filmlar ko'rsatilgan edi, biroq bular iste'mol mollari, restoranlar, ta'tillar yoki boshqa mahsulotlar ham bo'lishi mumkin edi.

Faraz qilaylik, E_s – bu $\langle u, i, r \rangle$ uchliklaridan tashkil topgan ma'lumotlar to'plami bo'lsin. Bunda $\langle u, i, r \rangle$ u foydalanuvchining i elementga r baho berganini anglatadi (vaqt belgilarini e'tiborga olmaganda).

Aytaylik, $r(u, i)$ – i tovar bo'yicha u foydalanuvchining bashorat reytingi bo'lsin.

17.1-jadval

MovieLens ma'lumotlar to'plamining qismi

User	Item	Rating	Timestamp
196	242	3	881250949
186	302	3	891717742
22	377	1	878887116
244	51	2	880606923
253	465	5	891628467
...	...	...	...

Maqsad – kvadratlar yig'indisi xatosini optimallashtirishdir.

$$\sum_{(u,i,r) \in E_s} (\hat{r}(u,i) - r)^2$$

Ko'pgina mashina yordamida o'qitishda bo'lgani kabi, maqsad o'quv misollari uchun emas, balki sinov misollari uchun optimallashtirishdir.

Quyida $r(u, i)$ uchun tobora murakkab modellar ketma-ketligi hamda foydalanuvchining biror element uchun reytingini bashorat qilish keltirilgan.

Har bir model modelga qo'shimcha atamalarni qo'shadi, xuddi boosting funksiyalar va chuqur o'rganishdagi qoldiq ma'lumotlar kabi.

Faqat bitta bashorat qiling. Eng oddiy holat barcha foydalanuvchilar va tovarlar uchun bir xil reytingni bashorat qilishdir: $r(u, i) = \mu$, bu yerda μ – o'rtacha reyting. Eslatib o'tamiz, har bir holat uchun bir xil qiymatni bashorat qilishda o'rtqiymatni bashorat qilish kvadratlar yig'indisi xatosini minimallashtiradi.

Foydalanuvchi va element biaslarini qo'shish:

Ba'zi foydalanuvchilar boshqa foydalanuvchilarga qaraganda yuqori baho berishlari mumkin, ba'zi filmlar esa boshqa filmlarga qaraganda yuqori reytingga ega bo'ladi. Buni hisobga olish uchun quyidagilardan foydalani-shingiz mumkin:

$$\hat{r}(u, i) = \mu + b_1[u] + b_2[i]$$

bu yerda u foydalanuvchining biasi $b_1[u]$ ga va i elementning biasi $b_2[i]$ ga teng.

Kvadratlar yig'indisi xatosini minimallashtirish uchun $b_1[u]$ va $b_2[i]$ parametrlari tanlanadi. Agar n ta foydalanuvchi va m ta element bo'lsa, μ o'zgaruvchisini belgilangan deb hisoblasak, $n+mn$ ta parametrlarni sozlash kerak bo'ladi. Eng yaxshi parametrlarni topish optimizatsiya masalasi bo'lib, uni gradiyentli tusish kabi metodlar yordamida amalga oshirish mumkin, bu esa chiziqli o'rgatishda qo'llanilgan.

Netflix mukofoti

Netflix mukofoti bilan hamkorlikda filtrlash bo'yicha bir nechta tadqiqotlar o'tkazildi. Bu mukofot \$1,000,000 miqdorida bo'lib, Netflixning maxsus tizimini bashorat aniqligini, kvadratlar yig'indisi bo'yicha o'lchangan holda, 10% ga yaxshilay olgan jamoaga berildi. Har bir reyting foydalanuvchiga, reytingi 1 dan 5 ta yulduzgacha bo'lgan filmni taqdim etadi. Shu bilan bir qatorda reyting berilgan sana hamda vaqtni ham taqdim etadi. Ma'lumotlar to'plami taxminan 100 million bahodan iborat bo'lib, 480,189 ta anonim foydalanuvchi tomonidan 17,770 ta filmga berilgan va 7 yil davomida to'plangan. Mukofot 2009-yilda yuzlab bashoratchilardan iborat to'plamdan o'rtacha natija ko'rsatgan va ba'zi murakkab metodlardan foydalangan jamoa tomonidan yutib olindi. 3 yil davomida olib borilgan tadqiqotdan so'ng, g'olib jamoa boshqa bir jamoani atigi 20 daqiqaga ortda qoldirib, mukofotni qo'lga kiritdi. Ikkala jamoaning ham yechimlari asosan bir xil xatolikni ko'rsatdi, ular g'alaba qozonish arafasida edi. Qiziqarli jihati shundaki, ikkala yechimning o'rtacha natijasi alohida har birining alohida yechimidan yaxshiroq edi. Bu yerda taqdim etilgan algoritim eng katta yaxshilanishni ko'rsatgan asosiy algoritmdir. Maxfiylik sabablari tufayli Netflix ma'lumotlar to'plami endi foydalanish uchun mavjud emas. Foydalanuvchilar faqat raqam bilan belgilangan bo'lsa-da, boshqa ma'lumotlar bilan qo'shilganda, bu ayrim foydalanuvchilarning shaxsini aniqlashga yetarli bo'lishi mumkin edi.

$b_1[u]$ to'g'ridan to'g'ri foydalanuvchi u ning o'rtacha reytingi bilan i mahsulot elementning o'rtacha reytingi bilan bog'liq bo'lishi kerak deb o'ylash mumkin. Shu bilan birga, agar barcha reytinglar foydalanuvchi u uchun o'rtacha μ dan yuqori bo'lsa ham, $b_1[u] < 0$ bo'lishi mumkin. Bu holat foydalanuvchi faqat mashhur narsalarni baholab, ularga boshqalarga nisbatan pastroq baho qo'ygan paytda yuz berishi mumkin.

$b_1[u]$ va $b_2[i]$ parametrlarini optimallashtirish reytinglarni yanada aniqroq baholash imkonini berishi mumkin, lekin bu tavsiyalarni shaxsiylashtirish imkonini bermaydi, chunki filmlar har bir foydalanuvchi uchun baribir bir xil tarzda tanlanadi.

Yashirin (latent) xususiyatni qo'shish. Siz har bir foydalanuvchi va har bir filmning yanada aniqroq bashorat qilish imkonini beruvchi xususiyatlar, masalan,

har bir foydalanuvchining yoshi va har bir filmning yoshga mosligi kabi xususiyatlari borligini taxmin qilishingiz mumkin. Har bir obyekt uchun real baholana-digan **latent xususiyat (yashirin xususiyat)** ma'lumotlarga muvofiq sozlanishi mumkin.

Faraz qilaylik, foydalanuvchi u uchun yashirin xususiyatning qiymati $f_1[u]$ va item i uchun yashirin xususiyatning qiymati $f_2[i]$ bo'lsin. Ushbu qiymatlarning ko'paytmasi ushbu foydalanuvchi uchun ushbu elementning reytingini hisoblash uchun ishlatiladi:

$$\hat{r}(u, i) = \mu + b_u[u] + b_i[i] + f_1[u] * f_2[i].$$

Agar $f_1[u]$ va $f_2[i]$ ning ikkalasi ham musbat yoki ikkalasi ham manfiy bo'lsa, bu xususiyat bashoratning aniqligini oshiradi. Agar $f_1[u]$ yoki $f_2[i]$ funksiyalardan biri manfiy, ikkinchisi esa musbat bo'lsa, ushbu xususiyat bashoratni kamaytiradi.

17.2-misol. 17.1-rasm bitta yashirin xususiyatga bog'liq reytinglarning grafik tasvirini ko'rsatadi. Buning uchun eng ko'p baho olgan 20 ta filmni va ushbu filmlar uchun eng ko'p baho olgan 20 ta foydalanuvchini o'z ichiga olgan MovieLens 100k ma'lumotlar to'plamidan foydalaniladi. U bitta yashirin xususiyatga ega bo'lgan gradiyent tushishining 1000 ta iteratsiyasi uchun o'qitilgan. Buning uchun MovieLens 100k ma'lumotlar to'plamining kichik qismidan foydalaniladi. Ushbu qism eng ko'p baholangan 20 ta filmni va shu filmlarga eng ko'p baho bergan 20 ta foydalanuvchini o'z ichiga oladi. U bir yashirin xususiyat bilan gradiyent tushish usulining ming marta takrorlanishi davomida o'rgatildi.

x o'qida foydalanuvchilar $f_1[u]$ xususiyatidagi qiymatlari bo'yicha tartiblangan holda joylashtirilgan. y o'qi bo'ylab $f_2[i]$ qiymatiga ko'ra tartiblangan filmlar joylashtirilgan. Keyin, har bir reyting foydalanuvchi va filmga nisbatan chizmada belgilanadi, shunday qilib u, i, r uchligi (x, y) koordinatarida $(f_1[u], f_2[i])$ nuqtasiga r ni joylashtirish orqali tasvirlanadi. Sonlarning har bir vertikal ustuni bitta foydalanuvchiga, har bir gorizontal qatori esa filmga mos keladi. Agar ikki foydalanuvchining ma'lum bir obyekt bo'yicha qiymatlari juda o'xshash bo'lsa, ustunlar bir-birini qoplaydi. Xususiyatda juda o'xshash qiymatlarga ega bo'lgan filmlar uchun satrlar bir-birini qoplaydi. Bu xususiyat qiymatlari nolga yaqin bo'lgan foydalanuvchilar va filmlarga ta'sir ko'rsatmaydi, chunki bashorat qilish jarayonida xususiyat qiymatlarining ko'paytmasi qo'llaniladi.

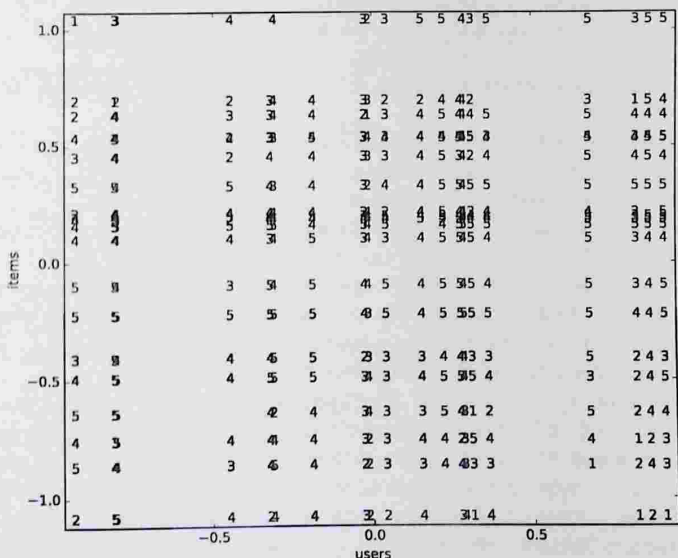
Odatda, yuqori reytinglar o'ng yuqori va chap quyi burchaklarda joylashadi, chunki bular mahsulotni ijobiy tavsiflovchi ko'rsatkichlardir. Past reytinglar esa, mahsulot uchun salbiy bo'lgani sababli, chap yuqori va o'ng quyi burchaklarda joylashtiriladi. Shuni esda tutingki, yuqori va past ko'rsatkichlar foydalanuvchining didi hamda filmning o'ziga xos xususiyatlariga bog'liq bo'ladi; bir film uchun yuqori deb hisoblangan ko'rsatkichlar boshqa film uchun farqli bo'lishi mumkin.

K ta yashirin xususiyatlarni qo‘shish. Faqat bitta xususiyatdan foydalanish o‘rniga, har bir foydalanuvchi va obyekt uchun k ta yashirin xususiyatni qo‘llash imkoniyatini ko‘rib chiqing. Har bir foydalanuvchi va $f \in \{0, \dots, k-1\}$ xususiyati uchun $E_1[u][f]$ qiymati mavjud (bu yerda matritsa uchun Python belgilanishi ishlatilgan). Har bir u foydalanuvchi va $f \in \{0, \dots, k-1\}$ xususiyati uchun $E_1[u][f]$ qiymati mavjud (matritsalar uchun Python belgilash tizimidan foydalanilgan holda). Uzunligi k bo‘lgan $E_1[u]$ vektor u foydalanuvchi uchun **foydalanuvchi joylashtirilishi** (user embedding) deb ataladi. Bu chuqur o‘rganishda qo‘llaniladigan **so‘z joylashtirishlariga** (word embeddings) o‘xshaydi. Shuningdek, har bir i elementi va f xususiyati uchun $E_2[i][f]$ qiymati mavjud bo‘lib, bunda $E_2[i]$ elementining joylashtirilgan (item embedding) ko‘rinishidir.

Alohida xususiyatlarning hissalari qo‘shib hisoblanadi. Bu quyidagini bashoratni beradi:

$$\hat{r}(u, i) = \mu + b_1[u] + b_2[i] + \sum_f E_1[u][f] * E_2[i][f].$$

Bu usul ko‘pincha **matritsa faktorizatsiyasi** usuli deb ataladi, chunki yig‘indi E_1 uchun **matritsa ko‘paytmasiga** va E_2 uchun esa transpozitsiyaga – ya‘ni argumentlarni o‘zgartirishga mos keladi.



17.1-rasm. Filmlarning birgina yashirin xususiyatga bog‘liq reytinglari; 17.2-misolga qarang

Regularize (tartibga solish) ortiqcha moslashuvning oldini olish uchun muntazamlashtirish (regularization) atamasini qo'shish mumkin. Bu atama parametrlarning haddan tashqari o'sishiga va berilgan ma'lumotlarga ortiqcha moslashishiga yo'l qo'ymaydi. Bu yerda optimallashtirish jarayoniga har bir parametr uchun L_2 regularizatori qo'shiladi.

Maqsad λ regularizatsiya parametri bo'lgan o'zgaruvchilarni tanlashdan iborat. Bu parametrni kesishma tekshirish (kross-validatsiya) yordamida sozlash mumkin.

$$\begin{aligned} & \text{minimize} \sum_{(u,i,r) \in E_S} (\hat{r}(u,i) - r)^2 \\ & + \lambda (\sum_u ([b_1]^2 + \sum_f E_1[u]^2) + \sum_i (b_2[i]^2 + \sum_f E_2[i][f]^2) \end{aligned} \quad (17.1)$$

b_1 , b_2 , E_1 va E_2 parametrlarini 17.2-rasmda ko'rsatilganidek stoxastik gradient tushish usuli yordamida optimallashtirish mumkin.

Undagi parametrlar har bir iteratsiyada sozlanadi, biroq **tartibga solish** ma'lumotlar to'plamidan o'tgandan so'ng amalga oshiriladi. Bu eng maqbul tanlov bo'lmashligi mumkin (17.3-mashqqa qarang). E'tibor beringki, $E_1[i][f]$ va $E_2[u][f]$ elementlari har bir xususiyatni farqli qilish uchun tasodifiy tarzda (bir xil qiymat bilan emas) initsializatsiya qilinishi kerak.

Nima o'rgatilayotganini anglash uchun quyidagi hayoliy (thought) tajribani ko'rib chiqing.

17.3-misol. Aytaylik, reyting misolida ba'zi odamlar haqiqatan ham yoqtiradigan, boshqalari esa yomon ko'radigan film janri bor. Faraz qilaylik, f xususiyati ushbu janrdagi filmlar uchun 1 qiymatini, qolgan filmlar uchun esa 0 qiymatiga ega. Bu janrni boshqa xususiyatlarga nisbatan kutilganidan ko'proq yoqtiradigan foydalanuvchilar f xususiyatining yuqori qiymatiga ega bo'ladi. Janrni unchalik yoqtirmaydiganlari salbiy qiymatga ega bo'lsa, qolganlarining qiymatlari nolga yaqin bo'ladi. Buni foydalanuvchilarni «yoqtirish», «yoqtirmaslik» va «befarq» toifalariga yumshoq klasterlash (soft clustering) sifatida ko'rish mumkin – xuddi kutilayotgan maksimallashtirish (EM) usulida bo'lgani kabi.

Endi shunday faraz qilaylik: foydalanuvchi xususiyati f ushbu qiymatlarga o'zgartirilmas qilib belgilangan, biroq filmga tegishli xususiyat o'rganish jarayonida shakllangan. Shu sababli, u janrga to'liq mos kelmasligi mumkin. Xususiyat uchun yuqori qiymatga ega bo'lgan filmlar – bu janrni yoqtiradigan odamlar tomonidan sevib tomosha qilinadigan va janrni yoqtirmaydiganlar tomonidan yoqtirilmaydigan filmlardir. Shunday qilib, u bu odamlarga qanday filmlar yoqishini yoki yoqmasligini bilib oladi. Bu esa filmlarning yumshoq klasterlanishini ta'minlaydi. Bir nechta yashirin xususiyatlar mavjud bo'lgan holda, u bir qancha yum-

shoq klasterlashtirish usullarini aniqlay oladi. Bunda ularning qiymatlari busting usuli va chuqur o'rganishdagi qoldiqlarga o'xshab jamlanadi.

O'zgaruvchan eng kichik kvadratlar algoritmi yuqoridagi tavsifga o'xshab ishlaydi, foydalanuvchi yoki element xususiyatlarini navbatma-navbat qayd qiladi va boshqasini optimallashtiradi, lekin tasodifiy topshiriqlardan boshlanadi. Foydalanuvchi yoki element parametrlari o'zgarmas qilib belgilangan holda (17.1) tenglamani optimallashtirish boshqa parametrlar bo'yicha chiziqli regressiyani hosil qiladi. 17.2-rasmdagi o'zgartirishlar ancha maydaroq, biroq natijalar bir-biriga o'xshash.

Ushbu algoritm ma'lum bir vaqtgacha bo'lgan ma'lumotlar bilan o'qitilishi va kelajakdagi ma'lumotlarda sinovdan o'tkazilishi orqali baholanadi. Baholash mezoni algoritmning kelajakdagi reytinglarni qanchalik aniq bashorat qila olishiga asoslanadi.

Algoritm elementlarning va foydalanuvchilarning kuzatilgan atributlarini hisobga olish uchun ba'zi xususiyatlarni qayd etish orqali o'zgartirilishi mumkin.

Masalan, Movielens ma'lumotlar to'plami 19 ta janrni o'z ichiga oladi. Bu janrlarni embedding maydonining 19 ta pozitsiyasi yordamida ifodalash mumkin, bunda har bir janrga bitta pozitsiya ajratiladi (bu embedding fazosining bir qismi uchun one-hot kodlashni hosil qiladi). Agar g pozitsiya ma'lum bir janrni ifodalasa, filmda ushbu janr mavjud bo'lsa, g pozitsiyaga 1 qo'yiladi, aks holda 0 qo'yiladi.

```
1: procedure Collaborative filter learner( $E_s, \eta, \lambda$ )
2: Inputs
3:  $E_s$ : set of  $\langle \text{user}, \text{item}, \text{rating} \rangle$  triples
4:  $\eta$ : gradiyent descent step size
5:  $\lambda$ : regularization parameter
6: Output
7: function to predict rating for a  $\langle \text{user}, \text{item} \rangle$  pair
8:  $\mu :=$  average rating
9: assign  $E1[u][f], E2[i][f]$  randomly
10: assign  $b1[u], b2[i]$  arbitrarily
11: define  $r(u, i) = \mu + b1[u] + b2[i] + \sum_f E1[u][f] * E2[i][f]$ 
12: repeat
13: # Update parameters from training data:
14: for each  $\langle u, i, r \rangle \in E_s$  do
15: error :=  $r(u, i) - r$ 
16:  $b1[u] := b1[u] - \eta * \text{error}$ 
17:  $b2[i] := b2[i] - \eta * \text{error}$ 
18: for each property  $f$  do
19:  $E1[u][f] := E1[u][f] - \eta * \text{error} * E2[i][f]$ 
20:  $E2[i][f] := E2[i][f] - \eta * \text{error} * E1[u][f]$ 
```

```

21: # Regularize the parameters:
22: for each item  $i$  do
23:  $b1[u] := b1[u] - \eta * \lambda * b1[u]$ 
24: for each property  $f$  do
25:  $E1[u][f] := E1[u][f] - \eta * \lambda * E1[u][f]$ 
26: for each user  $u$  do
27:  $b2[i] := b2[i] - \eta * \lambda * b2[i]$ 
28: for each property  $f$  do
29:  $E2[i][f] := E2[i][f] - \eta * \lambda * E2[i][f]$ 
30: until termination
31: return  $r$ 

```

17.2-rasm. Hamkorlikda filtrlash uchun gradiyent tushish

Har bir foydalanuvchi uchun g pozitsiyasida o'rganiladigan parametrlar mavjud bo'lib, u 17.3-misoldagi kabi, janrning shu foydalanuvchi uchun ijobiy yoki salbiy ekanligini ko'rsatadi. Foydalanuvchi xususiyatlari uchun turli xil pozitsiyalar qo'llaniladi. Foydalanuvchilar yoki obyektlar haqida o'rganish uchun yetarli baholash ma'lumotlari kam bo'lganda yoki «**sovuq boshlash**» muammosini (cold-start problem) hal qilishda – ya'ni yangi obyektlar yoki yangi foydalanuvchilar uchun tavsiyalar berishda, aniq xususiyatlarni qo'llash foydali hisoblanadi. Kuzatilgan xususiyatlarga asosanib, hech qanday reytinglarsiz bashoratlar qilish mumkin.

Ushbu algoritm vaqt belgisini hisobga olmaydi, biroq undan foydalanish foydali bo'lishi mumkin, chunki foydalanuvchilarning xohish istaklari o'zgarib turadi, mahsulotlar esa urfda bo'lishi yoki urfdan chiqishi mumkin. Kelajakdagi reytinglarni bashorat qilishda afzalliklar dinamikasi muhim (preferences becomes important) ahamiyat kasb etadi – bu odatda ko'zlangan maqsad hisoblanadi – biroq har bir foydalanuvchi uchun dinamikani o'rganishga yetarli ma'lumot deyarli mavjud emas.

Yuqorida keltirilgan misol regressiya uchun bo'lib, u foydalanuvchi uchun mahsulotning sonli reytingini bashorat qilishga qaratilgan edi. Masalan, Boolean klassifikatsiyasi uchun, reytingning 3 dan yuqori yoki pastligini aniqlashda, chiziqli funksiyaning sigmoid ehtimoliy bashoratni taqdim etadi. Binar (ikkilik) logarifmik yo'qotishni optimallashtirishni o'rgatganda, algoritm deyarli o'zgarishsiz qolishi mumkin. Faqatgina 17.2-rasmning 11-qatorida r ni aniqlashda sigmoid funksiyasi qo'shiladi, xolos. Reytingning 3 dan yuqori yoki past ekanligini aniqlovchi Boolean funksiyasi noodatiy hisoblanadi, chunki unda ham ijobiy, ham salbiy misollar keltirilgan. Aksariyat munosabatlar salbiy misollarni o'z ichiga olmaydi. Ba'zilar uchun yopiq dunyo taxmini – aytilmagan hamma narsa yolg'on, degan faraz o'rinli bo'lishi mumkin, biroq bu holda o'rganish uchun hech narsa qolmaydi. Ko'pchilik munosabatlarda salbiy misollar uchramaydi. Ba'zilar uchun yopiq

dunyo farazini qo'llash mumkin – ya'ni ko'rsatilmagan hamma narsa yolg'on deb hisoblanadi, biroq bu holda o'rganish uchun hech narsa qolmaydi. Faqat ijobiy misollar keltiriladigan munosabatlar uchun, masalan, reytingni ko'rsatishda haqiqiy bo'lgan reytingni bashorat qilish uchun yuqorida keltirilgan algoritmi hamma narsa to'g'ri degan xulosaga keladi, chunki bu bashorat (17.1) tenglamani minimal-lashtiradi. O'rtacha qiymat (17.2-rasm) aniqlanmagan. Bu holatni hal qilish uchun ikki usul mavjud: birinchisi, μ ni tartibga solinishi kerak bo'lgan parametrlar sifatida ko'rib chiqish; ikkinchisi esa salbiy misollarni taqdim etish zarur.

Musbat va manfiy misollarning nisbati hamda regularizatsiya parametrlari ehtimollik haqida dastlabki farazni taqdim etadi. Bu faraz ma'lumotlardan baholab bo'lmaydigan narsadir. Algoritmi shovqinni boshqara olgani tufayli, tasodifiy holatlarni salbiy misol sifatida tanlash maqsadga muvofiq, garchi ular ijobiy bo'lishi mumkin bo'lsa-da.

Salbiy misollarning yo'qligi baholash jarayonini qiyinlashtiradi va bu relyatsion bashorat usullarining ko'pincha har bir foydalanuvchi uchun **eng yuqori n ta elementni** bashorat qilish kabi reytinglar orqali baholanishiga sabab bo'ladi.

Tavsiyalar uchun qo'llanilganda, o'xshash mahsulotlarni taklif etishdan ko'ra, tavsiyalarning xilma-xilligini inobatga olish va butunlay turlicha mahsulotlarni tavsiya qilish ma'qulroq bo'lishi mumkin. Shu sababli, reytinglarni bashorat qilish foydali bo'lsa-da, bu to'g'ridan to'g'ri samarali tavsiyalarga olib kelavermaydi.

17.2.2. Bilimlar grafini o'rganish

Faraz qilaylik, maqsadimiz bilim grafining uchliklarini o'rganish bo'lsin. (s, r, o) uchligining to'g'riligini bashorat qilishni ko'rib chiqaylik, bunda s – subyekt, r – munosabat (fe'l) va o – uchlikning obyektidir. Uchliklarni bashorat qilish uchun matritsa faktorizatsiyasining oddiy kengaytmasi – poliadik parchalanish hisoblanadi. Bu usul tenzorni matritsalariga ajratishga asoslangan. Poliadik parchalanish, tenzorni bir nechta matritsa ko'rinishida ifodalaydi. U uch o'lchovli tenzorning elementi bo'lgan uchlikning ehtimollikni sozlanadigan parametrlar yordamida bashorat qilish uchun qo'llanilishi mumkin

$$\hat{p}((s, r, o)) = \text{sigmoid}(\mu + b_1[s] + b_2[r] + b_3[o] + \sum_f E_1[s][f] * E_2[r][f] * E_3[o][f])$$

global bias μ

har bir e obyekt uchun ikkita bias, ya'ni e birinchi holatda bo'lganda ishlatiladigan $b_1[e]$ va e uchinchi holatda bo'lganda ishlatiladigan $b_3[e]$

har bir r munosabat uchun $b_2[r]$ bilan berilgan bias

E_1 matritsasi obyektlar va xususiyatlar bilan indekslangan bo'lib, $E_1[e] - k$ uzunlikdagi vektorni ifodalaydi. Bu vektor e uchun **subyekt joylashtiruvchi** deb ataladi va e uchlikning birinchi elementi sifatida paydo bo'lganda qo'llaniladi. E_3 matritsasi esa har bir obyekt uchun obyekt joylashtiruvchini ta'minlaydi, qachonki u uchlikning uchinchi elementi sifatida namoyon bo'lsa.

- Munosabatlar va xususiyatlar bo'yicha indekslangan E_2 matritsasi, bunda $E_2[r] - r$ munosabatning **joylashtirilgan vektori** hisoblanadi

Barcha $E_1[e]$, $E_2[r]$ va $E_3[e]$ kiritmalar embeddings bir xil uzunlikka ega.

Ushbu modelni uchliklar to'plamida o'rgatish, L_2 kategorik logarifmik yo'qotishni tartibga solish usuli yordamida minimallashtirish, 17.2-rasmdagi algoritmgaga o'xshaydi. Biroq, bu holatda boshqacha bashorat qiluvchi va sozlash hamda tartibga solish uchun ko'proq parametrlar mavjud.

Faqat musbat misollarga ega bo'lgan uchliklar bilimlar grafigidagi, yuqoridagi ikkilik holat kabi, μ ni tartibga solmasdan, barcha uchliklarning to'g'ri deb hisoblanishi xatolikni eng kam darajaga tushiradi.

Yo manfiy misollarni qo'shish, yoki μ ni tartibga solish kerak bo'ladi. Ijobiy va salbiy misollar nisbati hamda μ ning tartibga solinishi, faqat ijobiy uchliklardan iborat ma'lumotlar to'plamidan olish mumkin bo'lmagan, to'g'ri deb taxmin qilinayotgan to'plamlar ulushi haqidagi dastlabki ma'lumotni beradi.

Biroq, poliadik ajratish yaxshi natija bermaydi, chunki **subyekt va obyekt joylashuvlari** o'zaro bog'liq emas.

17.4-misol. Aytaylik, u foydalanuvchi ma'lum bir d rejissyor suratga olgan filmlarni yoqtiradi va bu bilim (u , likes, m) va (m , directed by, d) ko'rinishida ifodalangan. Kingadir filmning yoqishi faqat uning **obyektiv joylashuvlariga** bog'liq bo'lsa, rejissyorlik esa faqat filmning **subyektiv joylashuvlariga** bog'liq bo'ladi. Bu kiritishlar o'zaro ta'sirlashmagani sababli, yuqoridagi kiritish modelida ularning o'zaro bog'liqligini aks ettirish imkoniyati mavjud emas.

Bu muammoning yechimi teskari munosabatlarni ifodalashdan iborat, bu yerda (m , likes⁻¹, u) $\leftrightarrow$ (u , likes, m) va (d , directed by⁻¹, m) $\leftrightarrow$ (m , directed by, d).

Inversiyali poliadik dekompozitsiya har bir r nisbat va uning r^{-1} inversiyasi uchun kiritmalarga ega, bu yerda (s , r , o) = (o , r^{-1} , s), bir xil kortejni tasvirlashning ikki usulidan bashorat qilishning o'rtacha qiymati hisoblanadi:

$$\widehat{p}(s, r, o) = \frac{1}{2} (\widehat{pd}(s, r, o) + \widehat{pd}(o, r^{-1}, s))$$

bu yerda pd poliadik parchalanishdan olingan bashoratni anglatadi. Bu subyekt va obyekt joylashtiruvlarining o'zaro bog'liqligini ta'minlaydi, natijada 17.4-misoldagi muammoni hal qilish imkoniyati paydo bo'ladi.

Inverslar bilan poliadik dekompozitsiya **to'liq ifodali** hisoblanadi, ya'ni u har qanday munosabatni logarifmik yo'qotish va $e > 0$ dan kichik bo'lgan aniqlik bilan yaqinlashtirishga qodir, hatto barcha vektorlar manfiy bo'lmagan qiymatlar bilan cheklangan va barcha vektor qiymatlari o'zgarmas bilan chegaralangan bo'lsa ham (bu o'zgarmas e ga bog'liq bo'lishi mumkin). Bu shuni anglatadiki, u har qanday munosabatni eslab qolishi mumkin, bu esa umuman olganda u qila oladigan eng yaxshi narsa hisoblanadi; **«bepul tushlik bo'lmaydi» yoki mushuk tekinga oftobga chiqmaydi** teoremasiga qarang.

Bu, shuningdek, taqdim etilishi mumkin bo'lgan narsalarga hech qanday ichki chekllovlar yo'qligini anglatadi. Subyekt va obyekt uchun bir xil joylashtirish usuli qo'llanilganda, faqat simmetrik munosabatlarni ifodalash yoki taxminiy tarzda ko'rsatish mumkin bo'ladi.

17.3-misol foydalanuvchi va elementlarning oldindan aytib berishda qo'llaniladigan o'rnatish qanday qilib o'rganilayotgan narsani tushunish uchun ishlatilishi mumkinligini tushuntiradi. Bu tushuntirish munosabatning noldan farqli qiymatga ega bo'lgan har bir joylashtirish o'rni uchun amal qiladi. Ayniqsa, barcha kiritilgan qiymatlar manfiy bo'lmagan holat uchun:

- Subyektни joylashtirishda har bir joylashuv o'rni obyektlarning umshoq klasterlashtirishini ifodalaydi. Klasterdagi o'sha obyektlar ma'lum bir pozitsiya uchun katta qiymatga ega bo'ladi, bunda «katta» so'zi boshqa qiymatlarga nisbatan nisbiy tushuniladi. Xuddi shunday holat obyektни joylashtirish uchun ham amal qiladi.

- Har bir pozitsiya uchun subyekt, munosabat va obyekt joylashuvlarining ko'paytmasi faqat uchala komponenti ham katta qiymatga ega bo'lgandagina katta bo'ladi. Biror pozitsiyada munosabat joylashuvi yuqori qiymatga ega bo'lsa, o'sha pozitsiya uchun subyekt klasteridagi elementlar tegishli obyekt klasteridagi elementlar bilan o'zaro bog'liq bo'ladi.

- $(\sum f)$ qo'shish bir nechta klasterlashlarni bir-biriga qo'shish imkonini beradi.

Umumiy holda, ko'paytiriladigan miqdorlarning juft soni manfiy bo'lsa, natija musbat chiqadi va yuqorida yuqorida ko'rsatilgandek ishlaydi. Qiymatlarning toq soni manfiy bo'lganida (ya'ni subyekt, munosabat va obyektning bittasi yoki uchtasi manfiy bo'lsa), ko'paytma manfiy bo'ladi va bu holat istisno vaziyat kabi ishlashi mumkin.

Ushbu usul hamda tegishli usullar – obyektlarni turli yo'llar bilan klasterlash bo'lib, eng yaxshi bashoratlarni ta'minlaydi. Olingan bilimlar aholining boshqa guruhlariga qo'llash mumkin bo'lgan umumiy bilimlar hisoblanmaydi. Agar maqsad bitta populyatsiyani, masalan, ijtimoiy tarmoqni modellashirish bo'lsa, bu yetarli bo'lishi mumkin. Ammo, agar maqsad yangi aholiga nisbatan qo'llanilishi mumkin bo'lgan bilimlarni o'rganish bo'lsa, bu usul o'z samarasini bermaydi.

Ko'p argumentli munosabatlarni matritsalariga ajratish (decomposing) usuli samarali natija bermaydi. Agar munosabat reifikatsiya yordamida uchlikka aylan-

tiriladigan bo'lsa, kiritilgan elementlar munosabatning argumentlari soniga teng bo'lgan juda oz miqdordagi namunaga ega bo'ladi – bu esa o'rganish jarayonini qiyinlashtiradi. Agar poliadik ko'rsatkich cho'kma bevosita qo'llanilsa, yuqorida teskari qiymatlarni keltirib chiqargan mustaqil holatlar muammosi har bir holat jufti uchun vujudga keladi.

Har bir argument bashorati juftligi orasidagi o'zaro ta'sirni modellashtirish haddan tashqari moslashuvga olib keladi.

Munosabatlarni buzuvchi o'rnatishga asoslangan modellar, foydalanuvchilarning yoshi kabi belgilangan oraliqli xususiyatlarni bashorat qilishda samarali emas. Buning sababi shuki, ular past o'lchamli tasvirlarga tayanadi, ammo bunday xususiyatlar uchun bu turdagi tasvirlar mavjud emas. Foydalanuvchi vektori bir yoki bir nechta joylashtirish pozitsiyalarida yoshni eslab qolish uchun ishlatilishi mumkin, bu esa uning umumlashtirilmasligini anglatadi.

17.3. Relyatsion o'zaro bog'liqlikni o'rganish

17.3.1. Relyatsion ehtimoliy modellar

Oldingi modellar subyektlar va munosabatlarning kiritilishini o'rganishga qaratilgan bo'lsa, bunga muqobil yondashuv munosabatlarning o'zaro bog'liqligini o'rganishdir. Bu yondashuv ba'zi munosabatlarni boshqalari orqali bashorat qiladigan modellarni yaratishni nazarda tutadi. Bu 9-bobda keltirilgan grafik modellar tasodifiy o'zgaruvchilarning o'zaro bog'liqligini ifodalashiga o'xshash usuldir. 9-bobdagi ishonch tarmoqlari tasodifiy o'zgaruvchilar orqali ta'riflangan edi. Ko'pgina sohalarni esa eng yaxshi tarzda shaxslar va ular o'rtasidagi munosabatlar nuqtayi nazaridan modellashtirish mumkin. Agentlar ko'pincha domenda qanday obyektlar borligini va shuning uchun qanday tasodifiy o'zgaruvchilar mavjudligini bilishdan oldin modellarni yaratishlari kerak. Ehtimolliklarni o'rganishda ko'pincha obyektlarning o'ziga xos xususiyatlari hisobga olinmaydi. Obyektlar va munosabatlarni o'z ichiga olgan ehtimollik modellari statistik relyatsion sun'iy intellekt sohasida o'rganiladi. Bu modellar o'zi relyatsion ehtimollik modellari, ehtimoliy relyatsion modellar yoki ehtimoliy mantiqiy modellar deb yuritiladi.

17.5-misol. O'quvchilarning o'zlari hali o'qimagan kurslarda qanday natija ko'rsatishlari mumkinligini bashorat qilish masalasini ko'rib chiqaylik. 17.3-rasmda ba'zi to'qima ma'lumotlar ko'rsatilgan bo'lib, ular nima qilish mumkinligini ko'rsatishga mo'ljallangan. s_3 va s_4 o'quvchilar kurslarning o'rtacha bali B bo'lgan fanlardan B o'rtacha balga ega. Biroq, ular qatnashgan kurslar haqidagi ma'lum ma'lumotlarga ega bo'lganingiz sababli, siz ularni farqlay olishingiz kerak. Relyatsion ehtimoliy model (RPM) – bu munosabatlarga asoslangan, ammo aniq obyektlardan mustaqil ravishda ehtimolliklar belgilanadigan modeldir. Obyektlar umumiy ehtimollik ko'rsatkichlariga ega bo'lib, ular ehtimollik parametrlarni o'zaro bo'lishadilar. Parametrlarni ham-

korlikda qo'llash, konvolyutsion neyron tarmoqlaridagi kabi, parametrlarni bo'lishish yoki vazn bo'g'lash deb ataladi.

Parametrlashtirilgan tasodifiy miqdor $R(t_1, \dots, t_k)$ ko'rinishida bo'lib, bu yerda har bir t_i – atama bo'lib, (mantiqiy o'zgaruvchi yoki o'zgarmas) hisoblanadi. Parametrlashtirilgan tasodifiy o'zgaruvchi unda mavjud bo'lgan mantiqiy o'zgaruvchilar yordamida parametrlashtirilgan deb hisoblanadi. Parametrlangan tasodifiy o'zgaruvchining asosiy namunasi, unda mavjud mantiqiy o'zgaruvchilar o'rniga doimiy qiymatlar qo'yish orqali hosil qilinadi. Parametrlangan tasodifiy o'zgaruvchining asosiy namunasi 17.1-mavzuda ko'rsatilganidek tasodifiy o'zgaruvchilarga mos keladi. Tasodifiy o'zgaruvchining domeni R diapazonidir. Mantiqiy Boolean parametrlangan tasodifiy o'zgaruvchi R predikat belgisiga mos keladi. Masalan, $Grade(S, C)$ parametrlangan tasodifiy o'zgaruvchi hisoblanadi, $Grade_{(s_3, c_4)}$ esa barcha mumkin bo'lgan baholar to'plamini o'z ichiga oluvchi tasodifiy o'zgaruvchidir. Biz mantiqiy o'zgaruvchilar bosh harf bilan boshlanadigan va o'zgarmaslar kichik harf bilan boshlanadigan Datalog konvensiyasidan foydalanamiz. Parametrlangan tasodifiy o'zgaruvchilar bosh harf bilan yoziladi, ularga tegishli takliflar esa kichik harflar bilan ifodalanadi (masalan, $Diff(c_i) = \text{true}$ – $\text{diff}(c_i)$ ko'rinishida, $Diff(c_i) = \text{false}$ esa – $\neg \text{diff}(c_i)$ ko'rinishida yoziladi). Bu tasodifiy o'zgaruvchilar uchun qo'llaniladigan konvensiyaga o'xshashdir.

Plastinka modeli quyidagilardan tashkil topgan:

- tugunlari parametrlashtirilgan tasodifiy o'zgaruvchilardan iborat bo'lgan yo'naltirilgan grafik

- har bir mantiqiy o'zgaruvchi uchun obyektlar populyatsiyasi va
- ota-onasi berilgan har bir tugunning shartli ehtimoli

To'g'ri to'rtburchak – **plastina** – umumiy mantiqiy o'zgaruvchiga ega bo'lgan parametrlashtirilgan tasodifiy miqdorlar atrofida chiziladi. Har bir mantiqiy o'zgaruvchi uchun alohida plastinka bor. Bu belgilash ortiqcha hisoblanadi, chunki mantiqiy o'zgaruvchilar ham jadvallar, ham argumentlarda ko'rsatib o'tilgan. Ba'zan ulardan biri tushirib qoldiriladi; ko'pincha jadvallardan plate (plastinka) xulosa chiqarish mumkin bo'lganda dalillar tushirib qoldiriladi.

17.6-misol 17.4-rasmda o'quvchilarning baholarini bashorat qilishning plate modeli keltirilgan. Kurslar uchun C plate va talabalar uchun S plate mavjud.

Parametrlangan tasodifiy miqdorlar bular:

- o'quvchi S ning aqlli yoki aqlli emasligini anglatuvchi $Int(S)$

Student	Course	Grade
s_1	c_1	A
s_2	c_1	C
s_1	c_2	B

s_2	c_3	B
s_3	c_2	B
s_4	c_3	B
s_3	c_4	?
s_4	c_4	?

17.3-rasm. Qaysi talabani c_4 kursida yaxshiroq natija ko'rsatishi mumkinligini taxmin qilish

- C kursning qiyinligini bildiruvchi $Diff(C)$
- talaba S ning C kursidagi bahosini ifodalovchi $Grade(S, C)$.

$P(Int(S))$, $P(Diff(C))$ va $P(Grade(S, C) | Int(S), Diff(C))$ ehtimolliklari aniqlanishi kerak. Agar I va D boolean bo'lsa (true va false qiymatlari bilan) va $Gr\{A, B, C\}$ oraliqda bo'lsa, unda ehtimollik taqsimotini belgilovchi 10 ta parametr mavjud.

Deylik, $P(Int(S)) = 0.5$ va $P(Diff(C)) = 0.5$ va $P(Grade(S, C) | Int(S), Diff(C))$

quyidagi jadval orqali aniqlanadi:

Int(S)	Diff(C)	Grade(S, C)		
		A	B	C
true	true	0.5	0.4	0.1
true	false	0.9	0.09	0.01
false	true	0.01	0.1	0.9
false	false	0.1	0.4	0.5

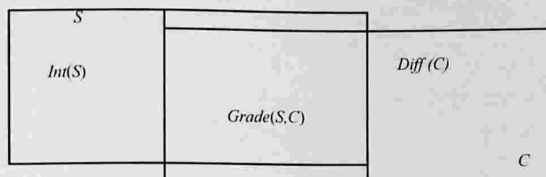
$P(Grade(S, C) | Int(S), Diff(C))$ ni aniqlash uchun sakkizta parametr talab qilinadi, chunki to'rtta holat mavjud va har bir holat ikkita raqamni aniqlashni talab qiladi; uchinchi, ehtimolliklar yig'indisi birga teng bo'lishi uchun chiqarilgan bo'lishi mumkin. Agar n ta o'quvchi va m ta kurs bo'lsa, asosda grounding n ta $Int(S)$, m ta $Diff(C)$ va n ta $Grade(S, C)$ holatlar misollari mavjud bo'ladi. Demak, asosda $n + m + n * m$ tasodifiy o'zgaruvchilar mavjud bo'ladi.

Parametrlangan tasodifiy o'zgaruvchilarni o'z ichiga olgan model ko'tarilgan model (lifted) deb ataladi.

Plate modeli uning asoslashi grounding degan ma'noni anglatadi – bu ishonchlar tarmog'i bo'lib, unda tugunlar parametrli tasodifiy o'zgaruvchilarning barcha asoslangan nusxalarini ifodalaydi (har bir mantiqiy o'zgaruvchi o'z populyatsiya-

sidagi obyekt bilan almashtiriladi) va yoylar saqlanadi. Ya'ni har bir plastinkadagi parametrli tasodifiy o'zgaruvchilar har bir obyekt uchun nusxalanadi. Asoslangan ishonch tarmog'idagi shartli ehtimolliklar plastinkalar modelining mos keluvchi nusxalari bilan bir xil bo'ladi.

17.7-misol. 17.4-rasmdagi plastinka modelini 17.3-rasmda berilgan ma'lumotlarga ko'ra shartlash va oxirgi ikki qatorga mos keladigan tasodifiy miqdorlarni so'rashni ko'rib chiqamiz. 4 ta kurs va 4 ta talaba bor, shuning uchun **groundda** 24 ta tasodifiy o'zgaruvchi bo'ladi. $\text{Grade}(S, C)$ ning barcha kuzatilmagan yoki so'ralmagan nusxalari qisqartirilishi yoki dastlab qurilmasligi mumkin, natijada esa 17.5-rasmdagi ishonch tarmog'i hosil bo'ladi.



17.4-rasm. Talabalar baholarini bashorat qilish uchun plastinkalar modeli

Ushbu tarmoqdan, obs shartlari ostida, 17.3-rasmdagi kuzatilgan baholarni va 17.6-misoldagi ehtimolliklarni ishlatgan holda quyidagi posterior ehtimolliklarni chiqarish mumkin:

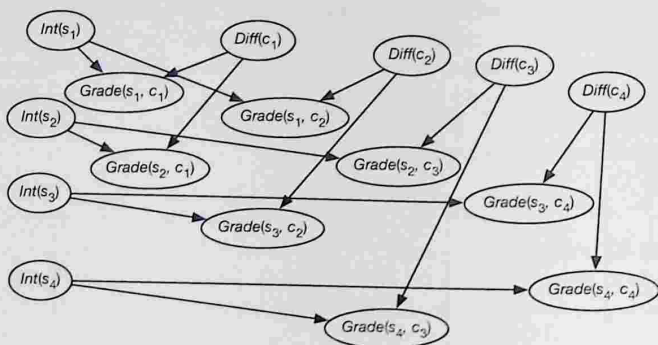
A	B	C
0.491	0.245	0.264
0.264	0.245	0.491

Shunday qilib, ushbu model s_3 talabasining s_4 talabasidan c_4 kursida yaxshiroq natija ko'rsatishi mumkinligini taxmin qiladi. 17.6-rasm yo'naltirilgan yoyning plastinka bilan qanday o'zaro ta'sir qilishining uchta holatini ko'rsatadi. Mantiqiy o'zgaruvchi X ning populyatsiyasi $\{x_1, \dots, x_n\}$ ga teng. Tasodifiy o'zgaruvchi $A(x)$ A_i ko'rinishda yoziladi va B uchun ham shunday. Parametrizatsiya plastinkalardan chiqarib olinishi mumkin.

17.6-rasmning (a) qismida A va B lar plastinkaning ichida bo'lib, shu sababli ular parametrli tasodifiy o'zgaruvchilar $A(X)$ va $B(X)$ ni ifodalaydi. Asoslashda, yoy har bir X qiymati uchun nusxalanadi.

$P(B_i | A_j)$ shartli ehtimollik har bir i uchun bir xil bo'lib, ko'tarilgan modelda $P(B | A)$ bilan modellashtirilgan. A_i va B_j bog'liq qiladigan boshqa yoy bo'lmasa A_j va B_j ga bog'liq emas.

17.6 (b) rasmda ota-ona (**parent**) plastinka tashqarisida joylashgan, shuning uchun u bosh populatsiyaning har bir elementi uchun takrorlanmaydi. $P(B_i | A)$ barcha i lar uchun bir xil bo'ladi.



17.5-rasm. 17.3-rasm ma'lumotlaridan taxmin qilish uchun yetarli bo'lgan asos. Kuzatilmagan yoki so'ralmagan $\text{Grade}(S, C)$ misollarini yaratish shart emas

Bu A kuzatilmaganda B_i ni bir-biriga bog'liq qiladi. 17.6 (a) va (b)-rasmdagi ko'tarilgan modellardagi parametrlar soni plitalarsiz tarmoq bilan bir xil bo'lganligi sababli populyatsiya soni n ga bog'liq emas.

17.6-misol(c)da bola o'zgaruvchan sonli ota-onalar bilan plastinkadan tashqarida joylashgan. Bu holda B n ta ota-elementi ega va shuning uchun $P(B | A)$ bo'lgan shartli ehtimollikdan foydalana olmaydi. Unga ixtiyoriy ko'plab ota-onalarning ehtimolligini hisoblash uchun agregator kerak. Indeksning ahamiyati bo'lmaganligi tufayli agregator simmetrik bo'ladi; ismlarni almashtirish hech qanday ta'sir ko'rsatmaydi.

Ushbu simmetriya almashinuvchanlik deb ataladi.

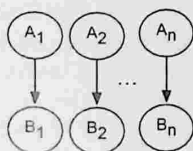
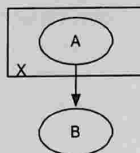
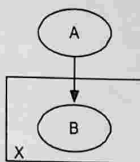
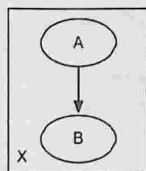
Ba'zi **agregatorlar** quyidagilar:

Agar model **Boolean** mantiqiy o'zgaruvchilar bilan **ehtimollik mantiqiy dasturi** sifatida aniqlangan bo'lsa, u holda **body of a clause** mantiqiy o'zgaruvchilarining **ekzistensial kvantizatsiyasi noisy-or** agregatsiyasiga olib keladi.

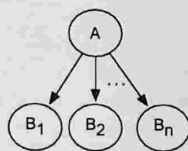
•17.6 (c)-rasmning agregatorida noisy-or $b \leftarrow (\exists X a(X) \wedge n(X)) \vee n_0$ ga teng, bu yerda n_0 va $n(x)$ lar X ning barcha x holatlari uchun mustaqil tasodifiy o'zgaruvchilaridir.

•Modelni birinchi tartib mantiqiy qo'shimchalar bilan kengaytirilgan mantiqiy vazn formulalari yordamida belgilash mumkin. Birinchi tartibli mantiqiy vazn formulasi birinchi tartibli mantiqiy formula va vazn juftligidir. Shartli ehtimol-

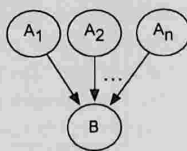
lik to'g'ri bo'lgan formulalar asoslari og'irliklari yig'indisining eksponensialiga proporsionaldir. Bu holda, agregatsiya **logistik regressiyaning** bir turi bo'lgan **relyatsion logistik regressiyaga** aylanadi.



(a)



(b)



(c)

17.6-rasm. Plitalar bilan yoylarning uchta holati: (a) plastinka ichidagi yoy; b) plastinka ichiga kiruvchi yoy; c) plastinkadan tashqariga chiqadigan yoy. Yuqorida plastinka modeli va uning ostida uning asoslanishi mavjud

Masalan, birinchi tartibli vazn formulalari ($b : w_0$), ($b \wedge a(X), w_1$) va ($b \wedge \neg a(X), w_2$) quyidagi natijaga olib keladi: $P(b | as) = \text{sigmoid}(w_0 + c_1 * w_1 + c_2 * w_2)$. Bu yerda as – ota-ona qiymatlariga belgilangan tayinlash bo'lib, u $A(e) = \text{true}$ bo'lgan c_1 ta obyekt va $A(e) = \text{false}$ bo'lgan c_2 ta obyektidan tashkil topgan.

- Ota-elementlarning ba'zi qiymatlarini o'rtacha, yig'indi yoki maksimum kabi standart ma'lumotlar bazasi agregatorlaridan foydalanish mumkin.

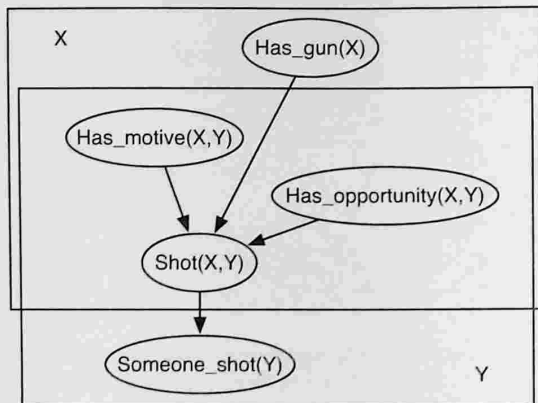
- Yanada murakkab agregatsiya yashirin **Dirichlet** taqsimoti (LDA), plastinkaning populyatsiyasi bolaning domeniga mos kelganda yuzaga keladi. $A(v)$ qiymatidan $P(B = v)$ ni hisoblash uchun foydalanish mumkin. Masalan, quyida berilgan softmaks yordamida (yoki ehtimol, agar $A(v) \geq 0$ bo'lsa, darajaga ko'tarmasdan).

$$P(B = v) = \frac{\exp(A(v))}{\sum_{v'} \exp(A(v'))}$$

Shunday qilib, agregatsiya transformerlarda ishlatiladigan mexanizmga o'xshash diqqat mexanizmini amalga oshiradi.

Bir-birini qoplaydigan plastinalar uchun qiymatlar quyidagi misolda ko'rsatilganidek, har bir plastina uchun alohida takrorlanadi.

17.8-misol. Faraz qilaylik, kimdir otib o'ldirilgan va muammo buni kim qilganligini aniqlashdan iborat. Otish uchun sabab, imkoniyat va qurol mavjud bo'lishi kerak deb faraz qiling.

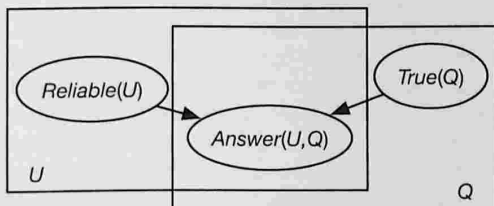


17.7-rasm. Odamga kim o'q uzganini aniqlaydigan plastina modeli

Buning uchun plita modeli 17.7-rasmda ko'rsatilgan bo'lib, unda Y otib o'ldirilgan odamni va X esa o'q uzgan odamni anglatadi. Muayyan bir odam y otib o'ldirilganligi kuzatilganda, maqsad ularni qaysi odam x otib o'ldirganligini aniqlashdir. Sabab va imkoniyat x va y ga bog'liq bo'lsa-da, o'q uzgan odam x ning quroli bor yoki yo'ligi y bog'liq emas. Bu yerda «noisy-or» (shovqinli-yoki) tegishli agregator hisoblanadi; agar kimdir y ni otgan bo'lsa, u otilgan hisoblanadi, ammo hech kim otmagan taqdirda ham uning otilgan bo'lishi ehtimoli mavjud (masalan, tasodifan qurol o'q uzib yuborgan bo'lishi mumkin). Ehtimol, $X = Y$ (ular o'zini o'zi otgan) va $X \neq Y$ (o'q uzgan shaxs va jabrlanuvchi turli odamlar bo'lgan) holatlar uchun balki turlicha ehtimolliklar mavjuddir. Agar 1000 kishi bo'lsa, asosda 3 002 000 ta tasodifiy o'zgaruvchi mavjud bo'lib, ular orasida «Has motive(X, Y)»ning millionta holati va «Has gun(X)»ning mingta holati bor. Agar barcha o'zgaruvchilar mantiqiy (Boolean) turdagi bo'lsa, 13 ta parametr belgilanishi lozim: «Shot» – jabrlanuvchi uchun sakkizta, «Someone shot» – qotil uchun ikkita va qolgan (parametrlashtirilgan) tasodifiy o'zgaruvchilarning har biri uchun bittadan qiymat belgilangan. Boshqa o'zgaruvchilarning har biri ota-elementlariga ega bo'lishi mumkin (17.5-mashqqa qarang).

17.3.2. Jamoaviy tasniflash va Crowd sourcing

Faraz qiling, siz javobini bilishni istagan ko'plab to'g'ri/noto'g'ri savollar mavjud. Javobni topishning bir yo'li ko'plab odamlardan **kraudsorsing**dan foydalanib so'rash bo'lishi mumkin. Bunda foydalanuvchilarga ishonchli javoblari uchun haq to'lanishi ehtimoli bor. Har bir kishi savollarning ma'lum bir qismiga javob beradi. Maqsad qaysi savollarning to'g'riligini aniqlash va kimga to'lov qilish kerakligini belgilashdan iborat. Ba'zi odamlar va botlar kam mehnat evaziga pul ishlash umidida tasodifiy javoblar berishga moyil bo'lishi mumkin. Kraudsorsing orqali haqiqatni aniqlash muammosi «haqiqatni kashf etish» (**truth discovery**) deb nomlanadi. Haqiqatni kashf etish algoritmlari sun'iy intellektda bilim grafikasidagi bashoratlarini baholashdan iborat. Bilim grafigida faqat ijobiy misollar mavjudligi sababli, baholash uchun alohida ajratib qo'yilgan test to'plamidan shunchaki foydalana olmaysiz. Ehtimoliy bashoratlarni baholash uchun, masalan, logarifmik yo'qotishni qo'llashda, salbiy misollar talab etiladi. Salbiy misollarni kraudsorsing yordamida olish mumkin. Bu muammoni ifodalashning bir usuli 17.8-rasmdagi plastinka modeli bo'lib, unda Answer (javob) (U, Q), ya'ni U foydalanuvchining Q savoliga bergan javobi, U ning ishonchliligi va Q ning haqiqiylikiga bog'liq bo'ladi.



17.8-rasm. Haqiqatni aniqlashning plastinka modeli

Shartli ehtimolliklar ishonchli foydalanuvchilar bergan javobning haqiqatga yaqin bo'lish ehtimoli yuqoriligini va ishonchsiz shaxs bergan javobning tasodifiy bo'lish ehtimoli ko'pligini ko'rsatib berishi mumkin. Taqdim etilgan javoblarga qarab, foydalanuvchilar ishonchli yoki ishonchsiz, savollar esa to'g'ri yoki noto'g'ri deb tasniflanadi. Ma'lum bir savolga ishonchli foydalanuvchilarning ozgina qismi javob berishi ham savolning to'g'riligiga ishonch hosil qilishimiz uchun yetarli bo'lishi mumkin. Biroq, tasodifiy javob beruvchi foydalanuvchilar javob haqida hech qanday foydali ma'lumot bermaydilar. Bu holatda bo'lgani kabi, bir vaqtning o'zida bir nechta obyektlarni tasniflash jarayoni **jamoaviy tasniflash (collective classification)** deb ataladi. Bu modelda aniq xulosa chiqarish imkonsiz, chunki Reliable, True tasodifiy o'zgaruvchilarni va kuzatilgan javobni o'z ic-

higa olgan asoslanish daraxtining kengligi tez sur'atda o'sib boradi (17.6-mashqqa qarang). Mantiqiy xulosani amalga oshirishning usullaridan biri Markov zanjiri Monte-Karlo (MCMC) usulidan foydalanish bo'lib, bunda haqiqiylik va ishonchlilik tasodifiy miqdorlari navbatma-navbat almashtiriladi. Foydalanuvchilarni ishonchli va ishonchsizlarga taxminiy ajratish hamda savollarning to'g'riligi aniqlangach, kutishni maksimallashtirish (EM – **expectation maximization**) usuli orqali ishonchli foydalanuvchining to'g'ri javob berish ehtimoli va foydalanuvchining ishonchli bo'lishi uchun dastlabki ehtimolini o'rganish mumkin bo'ladi. Ammo bunday usullar foydalanuvchilarning o'zaro til biriktirib, noto'g'ri ma'lumot berishlari natijasida buzilishi mumkin. Bu holatda noto'g'ri ma'lumot berish xato javoblar berishni anglatadi. Ba'zi foydalanuvchilar noto'g'ri ma'lumot tarqatish uchun til biriktirgan holatda, qaysi foydalanuvchilarning ishonchli ekanligini aniqlashda boshqa usullardan foydalanish mumkin. Siz haqiqatni bilgan ba'zi savollardan boshlashingiz mumkin; ishonchlilikni aniqlash uchun ehtimollikka asoslangan xulosa chiqarish usulidan foydalansa bo'ladi. Shu bilan bir qatorda, ba'zi foydalanuvchilar ishonchli deb hisoblanishi mumkin; ular bilan kelishgan foydalanuvchilarning ham ishonchli bo'lishi ehtimoli yuqoriroq. Shuningdek, Reliable domeni sifatida ikkitadan ortiq sinf bo'lishi mumkin. Qancha sinfdan yuklab olish masalasi ushbu darslik shaklidan tashqariga savol chiqsa, xitoycha restoran boshlash jarayoni bunga aniq darajada javob beradi.

Bir guruh odamlar barcha savollarga to'g'ri javob berib, faqat bittasiga noto'g'ri javob berishga kelishib olgan holatga qarshi kurashish juda mushkul vaziyatdir. Siz faqatgina til biriktirishmaydigan ishonchli foydalanuvchilar soni til biriktirayotganlarga qaraganda ko'proq bo'lishiga umid qilishingiz mumkin. Ba'zida eng yaxshi chora javoblarni tekshirib, ehtimol til biriktirish sodir bo'lganini aniqlashdir. Biroq, shuni yodda tutingki, odamlar biror narsaning to'g'riligiga kelishib olishlari, uning haqiqatan ham to'g'ri ekanligini anglatmaydi.

17.3.3. Til va mavzu modellari

9.6.6-mavzuda berilgan **ehtimoliy til modellari**, shu jumladan 9.38-rasmda tasirlangan **mavzu modellari** relyatsion modellar sifatida ko'rib chiqilganda ancha soddalashadi. Hujjatning mavzusi yoki mavzularini aniqlash undagi so'zlarni ma'no jihatdan farqlashga, shuningdek, hujjatlarni tasniflashga yordam beradi.

17.9-rasm **so'zlar to'plami** sifatida ifodalangan hujjat yagona mavzuga ega bo'lgan modelning plastinka ko'rinishidagi tasvirini aks ettiradi.

Ushbu modelda:

- D – barcha hujjatlarning to'plami
- W – barcha so'zlarning to'plami
- $A(W, D)$ shunday parametrlangan tasodifiy o'zgaruvchiki, agar w so'zi d hujjatda mavjud bo'lsa, $A(w, d)$ mantiqiy (Boolean) qiymati to'g'ri bo'ladi

- $T(D)$ shunday parametrlangan tasodifiy o'zgaruvchiki, uning $T(d)$ qiymati barcha mavzular to'plamidan olingan bo'lib, d hujjatning mavzusini ifodalaydi.
- $T(D)$ parametrizatsiyalangan tasodifiy o'zgaruvchi bo'lib, $T(d)$, barcha mavzularning aniqlanish sohasi bilan, d hujjatning mavzusi hisoblanadi.

Ushbu modelda hujjatning mavzusi barcha so'zlar bilan bog'langan. Plastinkadan chiquvchi yoylar bo'lmagani sababli, unda hech qanday ma'lumotlarni birlashtirish mavjud emas. Hujjatlar bir-biriga bog'liq emas. 17.10-rasm so'zlar to'plami sifatida ifodalangan hujjatning bir nechta mavzuga ega bo'lishi mumkinligini ko'rsatuvchi modelning plastinka tasvirini aks ettiradi. Domenli tasodifiy miqdor o'rniga barcha mavzular to'plami ishlatiladi – bu hujjat faqat bitta mavzuga ega bo'lishi mumkinligini anglatadi – har bir mavzu va hujjat uchun **Boolean** tasodifiy o'zgaruvchisi mavjud:

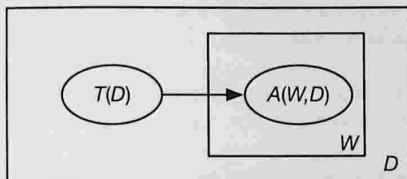
D – barcha hujjatlar to'plami

W – barcha so'zlar to'plami

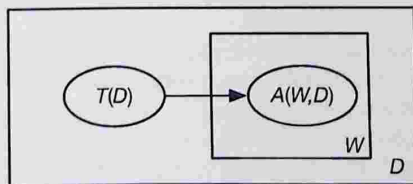
T – barcha mavzular to'plami (avvalgi modelga nisbatan belgilarning o'zgina o'zgarishi bilan)

Agar w so'zi d hujjatda uchrasa, $A(w, d)$ **Boolean** qiymati haqiqiy bo'ladi

Agar t mavzu d hujjatning predmeti bo'lsa, $S(t, d)$ **Boolean** qiymati haqiqiy bo'ladi.



17.9-rasm. Mavzu modeli: bitta mavzu va so'zlar to'plami



17.10-rasm. Plastinkalar yordamida mavzu modeli: bir nechta mavzular va so'zlar to'plami

Ushbu modelda hujjatning barcha mavzulari barcha soʻzlar bilan aloqador. Hujjatlar bir-biridan mustaqil hisoblanadi. Bitta hujjatning asoslanishi 9.29-rasmda koʻrsatilgan. Plastinadan tashqariga yoy chiqqanligi sababli, bu model maʼlumotlarni birlashtirish usulini talab qiladi.

Google Rephil'da Noisy-or parametridan foydalanilgan (9.38-misolga qarang).

17.11-rasmda, yuqoridagi misoldagi kabi, har bir hujjat soʻzlar toʻplami (unigram) koʻrinishida berilgan. Ushbu modelda:

- D – barcha hujjatlar toʻplami
- I – hujjatdagi soʻzlar indeksleri toʻplami; I ning qiymati hujjatdagi soʻzlar soniga qarab 1 dan boshlab oʻzgaradi
- T – barcha mavzular toʻplami
- $W(d, i)$ – d hujjatdagi i -inchi soʻz; W ning domeni barcha soʻzlar toʻplamidan iborat

• Agar t mavzu d hujjatning mavzusi boʻlsa, $S(t, d)$ ning qiymati toʻgʻri (true) boʻladi; S Boolean mantiqiy qiymatga ega boʻladi

17.9-misol. Faqat «mushuk gilamcha ustida oʻtirardi» soʻzlaridan iborat boʻlgan d_1 hujjatni koʻrib chiqaylik, u uchta mavzudan: hayvonlar, futbol va sunʼiy intellekt mavzularidan iborat deb faraz qilaylik. Soʻzlar toʻplami modelida har bir soʻz uchun mantiqiy tasodifiy oʻzgaruvchi bor. Ushbu hujjatni shartlash uchun $A(\langle \text{the} \rangle, d_1)$ va $A(\langle \text{cat} \rangle, d_1)$ toʻgʻri deb kuzatiladi. Hujjatda «neural» soʻzi yoʻqligi tifaqli $A(\langle \text{neural} \rangle, d_1)$ esa notoʻgʻri hisoblanadi. Boshqa soʻzlar uchun ham shunday boʻladi.

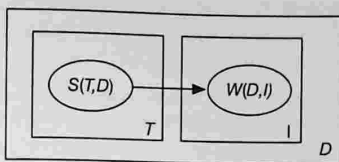
Bag-of-words (soʻzlar toʻplami) modelida hujjatdagi har bir pozitsiya uchun tasodifiy oʻzgaruvchi mavjud boʻlib, uning domeni barcha soʻzlar toʻplamidan iborat.

Ushbu modelda d_1 hujjat haqidagi kuzatuv quyidagicha koʻrinishga ega: $w(1, d_1) = \langle \text{the} \rangle$, $w(2, d_1) = \langle \text{cat} \rangle$ va hokazo, shuningdek, $w(7, d_1) = \perp$, bu yerda $\perp$ hujjatning oxirini ifodalovchi qiymatdir.

17.9-rasmda har bir hujjat uchun alohida mavzu ajratib koʻrsatilgan. Bu holat {hayvonlar, futbol, sunʼiy intellekt} domeniga ega boʻlgan $T(d_1)$ tasodifiy miqdor orqali ifodalangan.

Hujjat bir nechta mavzularni oʻz ichiga olishi mumkin boʻlgan boshqa ikkita modelda har bir mavzu uchun Boolean mantiqiy tasodifiy miqdor mavjud.

Bu holatda uchta tasodifiy oʻzgaruvchi mavjud boʻlib, ulardan biri S (hayvonlar, d_1) – d_1 hujjatning mavzusi hayvonlar boʻlganda toʻgʻri qiymat oladigan, boshqasi esa S (futbol, d_1) – d_1 hujjatning mavzusi futbol boʻlganda toʻgʻri qiymat oladigan oʻzgaruvchilardir.



17.11-rasm. Plastinkalardan foydalanilgan mavzu modeli: ko'p mavzular va so'zlar to'plamini o'z ichiga oladi

17.12-rasm **Dirixlening latent (yashirin) joylashuvi (LDA)** deb nomlanuvchi modelni ko'rsatadi. Ushbu model quyidagi ma'noni ifodalaydi:

- D hujjatni va I hujjatdagi so'zning indeksini anglatadi.
- T mavzuni bildiradi.
- $w(d, i)$ d hujjatdagi i -nchi so'zni anglatadi. w ning domeni barcha so'zlar to'plamidan iborat; shunday qilib, bu so'zlarning to'plami bag-of-words modelidir.

• $to(d, i)$ d hujjatdagi i -inchi so'zning mavzusini anglatadi. to ning domeni barcha mavzular to'plamini qamrab oladi. $P(w(d, i) | to(d, i))$ har bir mavzu bo'yicha so'zlarning taqsimlanishini ifodalaydi.

• $pr(d, t)$ d hujjatning t mavzuga oid qismining nisbatini ifodalaydi. Pr ning domeni aniqlanish sohasi haqiqiy sonlar to'plamidan iborat. Agar $pr(d, t)$ musbat (psevdo) hisob qiymat bo'lsa, u holda bu Dirixle taqsimotini ifodalaydi, bunda

$$P(to(D, I) = t | pr(D, T)) = \frac{pr(D, T = t)}{\sum_{t'} pr(D, T = t')}.$$

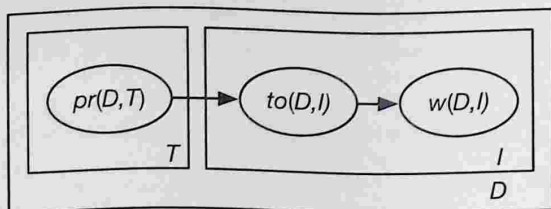
Boshqacha qilib aytganda, $pr(d, t)$ softmax sifatida ifodalanishi mumkin bo'lgan yerda

$$P(to(D, I) = t | pr(D, T)) = \frac{\exp(pr(D, T = t))}{\sum_{t'} \exp(pr(D, T = t'))}$$

bo'lib, bu diqqat mexanizmiga o'xshaydi. Bunda har bir so'z mavzularga e'tibor qaratadi.

So'zlar to'plami va so'zlar xaltasi (bag-of-words) tushunchalaridan tashqariga chiqish uchun, masalan, plastina modelida ko'rsatilganidan ko'ra murakkabroq tuzilma, n -gramm kabi so'zlar ketma-ketligini modellashtrish zarur bo'ladi. Xususan, bu jarayonlar avvalgi so'zlar ketma-ketligiga murojaat qilish qobiliyatini

talab etadi. Bu qobiliyat mantiqiy o'zgaruvchilardagi (plastinalardagi) tuzilma orqali, ya'ni mantiqdagi funksiya belgilari bilan ifodalanishi mumkin.



17.12-rasm. Plastinkalardan foydalangan holda mavzu modeli:
Dirixleneng latent (yashirin) joylashuvi

17.3.4. Ayrim o'ziga xos tasavvurlar

Ishonch tarmoqlarining har qanday ko'rinishidan relyatsion modellarni ifodalash uchun foydalanish mumkin. Ammo plastinkalar mantiqiy o'zgaruvchilarga mos kelgani tufayli, birinchi tartibli mantiq asosida belgilangan ifodalar bu modellarni tabiiy ravishda ifodalaydi. Modellar mantiqiy o'zgaruvchilarga ham struktura bera oladi. Neyron tarmoqlar sohasidagi adabiyotlarda, parametrlarning o'zaro bog'liqligi tufayli, plastinkali modellar **konvolyutsion modellar** deb yuritiladi. Bu xususiyat mantiqdagi universal kvantlashga ham mos keladi.

Tasavvurlar, asosan, agregatsiyani qanday qayta ishlashi va latent belgilar tasodifiy o'zgaruvchilar yoki sozlanadigan vaznli formulalar (weighted formulas) (neyron tarmoqlarida bo'lgani kabi) bo'lishi bilan farq qiladi.

Ba'zi e'tiborga molik tasavvurlar quyidagilarni o'z ichiga oladi.

Ehtimollikka asoslangan mantiqiy dasturda model parametrlangan shovqin o'zgaruvchilari bilan mantiqiy dastur ko'rinishida tavsiflanadi. Plastinkalar esa mantiqiy o'zgaruvchilarga mos keladi.

17.10-misol. 17.7-rasmning aniqlanish sohasini 17.8-misolda tasvirlangani-dek ifodalash uchun:

$otilgan(Y) \leftarrow \text{hech kim tomonidan otilmagan}(Y) - is_shot(Y) \leftarrow shot_by_no_one(Y)$

$otilgan(Y) \leftarrow X \text{ tomonidan otilgan}(X, Y) \wedge otish \text{ muvaffaqiyatli bo'lgan}(X, Y).$
 $is_shot(Y) \leftarrow shot(X, Y) \wedge shot_succeeds(X, Y).$

«Otishga hech kim yo'q» va «otishga muvaffaq bo'lindi» iboralarining har bir namunasi parametrlashtirilgan shovqin o'zgaruvchilari hisoblanadi.

$\underline{P}$ (*shot by no one*(Y)) – bu Y ga hech kim o‘q uzmagani bo‘lsa ham, uning otib o‘ldirilganligi ehtimolligidir.

P (*shot succeeds*(X, Y)) – bu X tomonidan Y ga o‘q uzilganda, Y ning otib o‘ldirilish ehtimolidir.

Ushbu ikki qoida har bir Y uchun X ning namunalarini noisy-or usulida amalga oshiradi. Parametrlashtirilgan tasodifiy miqdor «shot»ni quyidagicha qoidalar yordamida ifodalash mumkin:

$\text{shot}(X, Y) \leftarrow \text{has_motive}(X, Y) \wedge \text{has_gun}(X) \wedge \text{has_opportunity}(X, Y) \wedge \text{actually_shot}(X, Y).$

P (*actually shot* (X, Y)) – X da motiv, qurol va imkoniyat mavjud bo‘lganda, X ning Y ni o‘tish ehtimolini ifodalaydi. Boshqa qoidalar X hech qanday motivga ega bo‘lmagan holatlarini qamrab olishi mumkin. Ehtimollik mantiqiga asoslangan dasturlar oddiy model doirasidan tashqariga chiqadi, chunki ular yuqori darajadagi modelda sikllarga olib keladigan rekursiyani qayta ishlash imkoniyatiga ega.

Birinchi tartibli mantiqiy vazn formulalari

Muqobil usul – vazn formulalaridan foydalanish bo‘lib, bu yerda formulalar plastinalarga mos keladigan mantiqiy o‘zgaruvchilarni o‘z ichiga oladi. Formullalar yo **relyatsion logistik regressiya** deb ataladigan yo‘naltirilgan modelni, yoki **Markov mantiqiy tarmog‘i** (MLN) deb ataladigan yo‘naltirilgan anonni aniqlashni mumkin.

17.11-misol. 17.7-rasmda berilgan domen vazn formulalari yordamida ifodalaniishi mumkin.

$(\text{is_shot}(Y), w_0)$

$(\text{is_shot}(Y) \vee \neg \text{shot}(X, Y), w_1)$

$(\text{shot}(X, Y) \vee \neg \text{has_motive}(X, Y) \vee \neg \text{has_gun}(X) \vee \neg \text{has_opportunity}(X, Y) \vee \neg \text{actually_shot}(X, Y), w_2)$

bu yerda w_0 , w_1 va w_2 – tegishli formulaning barcha asosiy namunalari uchun umumiy bo‘lgan vaznlarni ifodalaydi. E’tibor bering, oxirgi vazn formulasi tanada konyunksiyalarga ega bo‘lgan implikasiya bilan bir xil shaklga ega; bu qoidaga istisnalar bo‘lmagan barcha hollarda to‘g‘ri hisoblanadi.

Markov mantiqiy tarmoqlari va relyatsion logistik regressiya modellaridagi xulosalash jarayoni, asoslashda kuzatilmagan o‘zgaruvchilarni marginalashtirishni (marginalizing) talab qiladi.

Ehtimoliy yumshoq mantiq. Markov mantiqiy tarmoqlaridagi kabi mantiqiy vazn formulalariga ega bo‘lib, unda yashirin o‘zgaruvchilar neyron tarmoqlaridagi yashirin xususiyatlarga o‘xshab ifodalanadi. Bu marginalizatsiyani chetlab o‘tish mumkin bo‘lgani uchun xulosa qilish ancha tezroq bo‘lishini anglatadi va o‘qitish gradiyent tushishning standart usullaridan foydalangan holda amalga oshirilishi mumkin. Bu yashirin o‘zgaruvchilarni

tasodifiy o'zgaruvchilar sifatida talqin qilib bo'lmashligini anglatadi, bu esa interpretatsiyani qiyinlashtirishi mumkin.

Ehtimoliy mantiq dasturlari kabi yo'naltirilgan modellar asoslanishdagi keraksiz o'zgaruvchilarni olib tashlash imkoniyatiga ega, misol uchun, haqiqatni aniqlash jarayonida ularning aksariyati olib tashlanadi hamda modulli tarzda o'rganilishi mumkin. Bu modellarning afzalliklari shundaki, ular zarur bo'lmagan ma'lumotlarni saralash va alohida qismlar bo'yicha o'rganish imkonini beradi. Ishonch tarmog'ining asoslanishida uchraydigan muammolardan biri uning asiklik bo'lishi kerak. Masalan, $do\ 'stlar\ (X, Y)$ $do\ 'stlar\ (Y, Z)$ ga bog'liq bo'lishi mumkin va asoslanishni asiklik qilish ixtiyoriy tartiblanishni talab qiladi. **Relyatsion bog'liqlik tarmoqlari** sikllarni o'z ichiga olishga imkon beradigan yo'naltirilgan modellardir. Modeldagi ehtimolliklar Markov zanjirida o'tishlarni belgilab beradi. Model tomonidan belgilangan taqsimot – bu hosil qilingan Markov zanjirining statsionar taqsimotidir.

Grafik neyron tarmoqlar

Grafik neyron tarmoqlar – bu grafik ma'lumotlari bilan ishlaydigan neyron tarmoqlar. Har bir tugun o'zining qo'shnilari va ularning qo'shnilarining ma'lum chuqurlikdagi parametrlashtirilgan chiziqli funksiyalari hamda faollashtirish funksiyalaridan kelib chiqadigan joylashuvga ega. **Relyatsion grafik konvolyutsion tarmog'i** (R-GCN) **bilimlar grafiglari** uchun joylashuvlarni o'rganish maqsadida qo'llaniladi, bunda tugunlar obyektlarni, yoylar esa munosabatlarni ifodalaydi. Bilimlar grafigidagi e obyekt uchun xususiyatlar $h_e^{(0)}$ kiritmasini hosil qilishda qo'llaniladi. Bu jarayon so'z kiritmalari uchun ishlatiladigan kodlovchiga o'xshash tarzda amalga oshiriladi. Ushbu vektorlar neyron tarmog'iga kiruvchi qiymatlarni taqdim etadi. Umumiy parametrlarga ega bo'lgan bir nechta qatlamlar mavjud bo'lib, ular har bir obyekt uchun chiquvchi qiymat kiritish uchun grafik tuzilishiga amal qiladi.

Obyekt uchun chiquvchi qiymat vektori obyektning xususiyatlarini yoki yoylarni bashorat qilish uchun poliadik dekompozitsiya kabi modellarda qo'llanilishi mumkin. U *konvolyutsion* deb ataladi, chunki boshqa relyatsion modellardagi kabi, har bir obyekt uchun qatlamlarda bir xil o'rganiladigan parametrlardan foydalaniladi. Ushbu qatlamlar har bir obyektning o'ziga xos xususiyatlari va uning atrofidagi muhitdan kelib chiqqan holda, alohida vektorlarni taqdim etadi.

Qatlamlar quyidagicha aniqlanadi. e obyekt uchun $l + 1$ qatlamning joylashuvi, oldingi qatlamdagi bilimlar grafigida e va uning qo'shnilarining joylashuvlari asosida aniqlangan $h_e^{(l+1)}$ vektori hisoblanadi:

$$h_e^{(l+1)} = \phi \left(W_0^{(l)} h_e^{(l)} + \sum_{r \in R} \sum_{\{n: (e,r,n) \in KG\}} \frac{1}{C_{e,r}} W_r^{(l)} h_n^{(l)} \right)$$

bu yerda φ faollashtirish funksiyasi bo'lib, u ReLU kabi $\varphi(x) = \max(x, 0)$ ko'rinishida aniqlanadi. R barcha munosabatlar to'plami va KG bilimlar grafigidagi uchliklar to'plamidir. $h_e^{(l)}$ qatlamdan olingan e obyektni vektorlashtirishdir. $W_r^{(l)} - l$ qatlam uchun r munosabatining matritsasi bo'lib, u har bir qo'shni n ning $h_n^{(l)}$ vektoriga ko'paytiriladi. $W_0^{(l)}$ – bu l qatlamdagi e obyekt uchun vektor keyingi qatlamdagi xuddi shu obyektga qanday ta'sir ko'rsatishini belgilaydigan matritsa hisoblanadi. $C_{e,r}$ – har bir munosabat uchun o'rtacha qiymat beradigan $| \{n: (e, r, n) \in KG \} |$ kabi normallashtirish doimiysidir. Boshlang'ich holat $h_e^{(0)}$ bo'lib, u hech qanday qo'shnilik ma'lumotlaridan foydalanmaydi.

Ushbu model har bir munosabat uchun alohida parametrlardan foydalanadi, bu esa bir nechta namunalar bilan munosabatlarga ortiqcha mos kelish holatiga olib kelishi mumkin. Bu muammoni yechishga ko'maklashish uchun $W_r^{(l)}$ munosabat matritsalarini munosabatlar orasida umumiy bo'lgan o'rganilgan asos matritsalarining chiziqli kombinatsiyasi sifatida ifodalash mumkin. Bunda chiziqli kombinatsiyadagi vazn koeffitsiyentlari munosabatga bog'liq bo'ladi.

17.2.2-mavzudagi o'rnatish asosidagi modellar reifikatsiyalangan obyektlar uchun samarali ishlamagan, chunki bunday obyektlar juda kam namunalardan iborat edi. Biroq, relyatsion grafik konvolyutsion tarmoqda ikki yoki undan ortiq qatlamlardan foydalanish tugunning qo'shni tugunlarning qo'shnilariga bog'liq bo'lishiga imkon yaratadi.

Tarmoqning chuqurligi, tugunning qo'llaniladigan qo'shnilar soni bilan to'g'ridan to'g'ri bog'liq. K o'lchamli tarmoq har bir tugundan k uzunlikdagi yo'l-lardan foydalanadi.

Ko'tarilgan grafik modellarga asoslangan usullar, oraliq o'zgaruvchilar kuzatilgan yoki kuzatilmaganidan qat'i nazar, bir xil tarzda ishlaydi. Misol uchun, otish misolidagi $Has_motiv(X, Y)$ boshqa parametrizatsiyalangan tasodifiy miqdorlarga bog'liq bo'lishi mumkin, masalan, kimdir X ni Y ni otish uchun yollaganlik holatiga (17.5-misolga qarang), bu holda $Has_motivi(X, Y)$ chetga chiqarilgan. Shu bilan birga, grafik neyron tarmoqlari bilish grafigida ko'rsatilgan obyektlar larni xulosa qilingan munosabatlardan farqli ravishda ko'rib chiqadigan. Ushbu uchun modelni ko'rsatilgan munosabatlarning funksiyasi sifatida o'rganish. Modullilikning yetishmasligi ularni kamroq moslashuvchan qiladi.

17.4. Mavjudlik va identifikatsiya noaniqligi

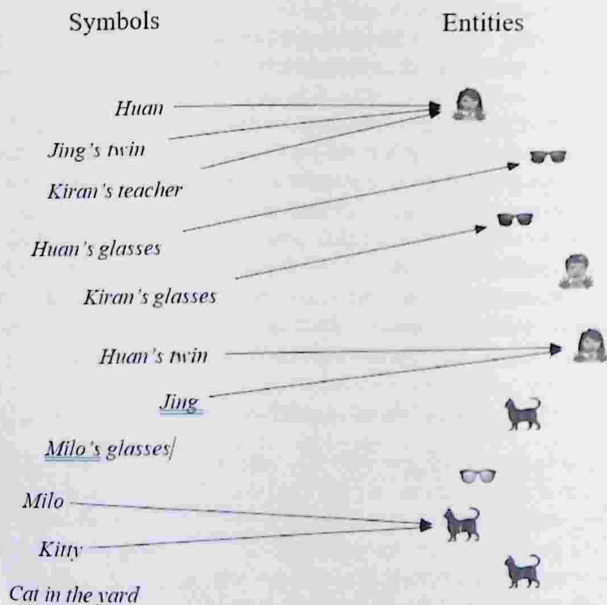
17.3-mavzuda taqdim etilgan modellar **relyatsion noaniqlikka**, ya'ni ba'zi obyektlar uchun munosabatning haqiqiy yoki noto'g'ri ekanligi haqidagi noaniqlikka bag'ishlangan. Masalan, $X \neq Y$ bo'lgan holda, turli odamlarning bir-birini yoqtirishi yoki yoqtirmasligini ko'rsatuvchi $(X, Y)likes$ ning ehtimoliy modeli X va Y ning xususiyatlariga hamda ular aloqador bo'lgan boshqa munosabatlarga bog'liq bo'lishi mumkin. $Likes(X, X)$ ning ehtimollik modeli – bu odamlarning

o'zlarini yoqtirishlari yoki yoqtirmasliklari haqidagi modeldir. Kim kimni yoqtirishi haqidagi modellar, masalan, agent-murabbiyga ikki kishi birgalikda ishlashga arziydimi yoki yo'qligini aniqlashda foydali bo'lishi mumkin.

Identifikatsiya noaniqligi muammosi ikkita belgi (atama yoki tavsiflarning) bir xil (bir narsani anglatishi) yoki yo'qligini aniqlashdagi noaniqlikka taalluqlidir.

17.12-misol. 17.13-rasmda ba'zi belgilarning ma'nolari ko'rsatilgan. Har bita strelka belgining qaysi obyektini ifodalashini ko'rsatadi. *Huan* = *Jing's twin* Xuan Jingning egizagi bo'lganligi sababli ular teng va bir xil obyektini ifodalaydi. *Huan's glasses* ≠ *Kiran's glasses*, ya'ni Xuanning ko'zoynagi Kiranning ko'zoynagiga teng emas, chunki ular turli juft ko'zoynaklarni anglatadi, garchi ko'zoynaklar bir xil bo'lishi mumkin bo'lsa ham; ular faqat bir xil juft ko'zoynakka ega bo'lgandagina ular teng bo'lar edi. *Milo's glasses*, ya'ni Miloning ko'zoynagi umuman bo'lmaganligi tufayli, u mavjud emas.

Identifikatsiya belgilarni ajratadi. Xuan (Jingning egizagi) va Kiranning o'qituvchisi bir xil bo'limda joylashgan, chunki ular aynan bir obyektini anglatadi. Ular Jingdan boshqa bo'limda joylashgan. Hech qanday obyektini anglatmaydigan belgilar, ya'ni Miloning ko'zoynagi va hovlidagi mushuk, alohida bo'limda bo'lishi mumkin.



Shaxsni aniqlashdagi noaniqlik tibbiyot tizimlari uchun muammo hisoblanadi, chunki hozirgi paytda tizim bilan muloqot qilayotgan shaxs kecha tashrif buyurgan shaxs bilan aynan bir kishi ekanligini aniqlash muhim ahamiyatga ega.

Agar bemor muloqotga kirishishni xohlamasa yoki tizimni aldashga harakat qilsa, masalan, giyohvand moddalar olish maqsadida, bu muammo ayniqsa mu-rakkablashadi.

Bu muammo «qayd qilish aloqasi» deb ham ataladi, chunki masala bir shaxs-ga tegishli bo'lgan (tibbiy) qaydlarni aniqlashdan iborat.

Ushbu muammo iqtibosni solishtirish jarayonida ham yuzaga keladi. Bunda turli maqolalardagi mualliflar bir xil shaxs ekanligini yoki ikki xil iqtibos aslida bir xil maqolaga tegishli ekanligini aniqlash zarur bo'ladi.

Shaxsning noaniqligini hisoblash imkonsiz, chunki bu belgilarni qismlarga ajratishga ekvivalentdir. Iqtiboslarni taqqoslashda bo'linish iqtiboslar bo'yicha amalga oshiriladi; bir xil ishni anglatuvchi iqtiboslar bir xil bo'linishda joylashadi. n ta elementning bo'linishlari soni **Bell soni** deb ataladi va u n ga nisbatan har qan-day eksponensial funksiyadan ham tezroq o'sadi. Bo'linmalar bo'yicha taqsimotni topishning eng keng tarqalgan usuli **Markov zanjiri Monte-Karlo** (MCMC) usuli-dan foydalanishdir. Bo'linma berilgan bo'lsa, obyektlar boshqa bo'linmalarga yoki yangi bo'linmalarga ko'chirilishi mumkin.

Avvalgi bo'limlarda tasvirlangan usullar obyektlar ma'lum ekanligini ko'rsata-di. Tavsifga mos keladigan obyektning mavjud yoki mavjud emasligini aniqlash muammosi – bu **mavjudlikning noaniqlik muammosidir**. Mavjudlikning noa-niqligi muammoli holat bo'lib, biror tavsifga mos keladigan subyektlar umuman bo'lmashligi yoki aksincha, bir nechta subyekt mavjud bo'lishi mumkin. Masalan, «mushuk hovlida» (the cat in the yard) tavsifi uchun hovlida hech qanday mushuk bo'lmashligi yoki bir nechta mushuk bo'lishi mumkin. Mavjud bo'lmagan narsa-larga xususiyatlar berib bo'lmaydi, chunki yo'q bo'lgan narsalarning xususiyatlari ham bo'lmaydi.

Agar mavjud bo'lgan biror obyektga nom bermoqchi bo'lsangiz, bir nech' obyektlar bor bo'lgan taqdirda, siz aynan qaysi narsani nazarda tutayotganingiz aniqlab olishimiz lozim.

Mavjudlik noaniqligining alohida bir turi – bu mavjud narsalarning **soni haq-dagi noaniqlikdir**. Masalan, xarid agenti sayohatga qiziqish bildirishi mumkin bo'lgan odamlar soni haqida aniq tasavvurga ega bo'lmashligi mumkin. Sayohatni taklif etish yoki etmaslik esa aynan qiziqish bildirishi mumkin bo'lgan odamlar soniga bog'liq bo'ladi.

Mavjudlikning noaniqligiga doir fikr yuritish murakkab rollar haqida gap ket-ganda qiyin bo'lishi mumkin. Bunda asosiy muammo shundan iboratki, ushbu rol-larni egallay oladigan obyektlarning bor-yo'qligini aniqlash lozim. Sem va uning farzandi Kris uchun kvartira topishi kerak bo'lgan xarid agentini tasavvur qiling.

Semning kvartiraga xohishi qisman uning xonasining kattaligiga va Krisning xonasi rangiga qarab ehtimollik bilan bog'liq bo'ladi.

Ammo, obyektlarning kvartiralarini «Semning xonasi» va «Krisning xonasi» deb belgilanmaydi va ularning har biri uchun alohida xona bo'lmazligi ham mumkin. Sem olmoqchi bo'lgan kvartira modelini hisobga olsak, kuzatuvlarni qanday shartlash kerakligi aniq emas.

17.5. Ijtimoiy ta'sir

Tavsiya beruvchi tizimlar sun'iy intellektning eng tijoriy tarmog'i hisoblanadi. Meta (Facebook) va Alphabet (Google) kabi ko'plab yirik kompaniyalar o'z daromadlarini reklama orqali topadi. Kompaniyalar bittagina tugmani bosish (klik) orqali daromad olganida, ularning foydasi to'g'ridan to'g'ri ikki narsaga: reklamalarni alohida foydalanuvchilarga qanchalik samarali yo'naltira olishlari va odamlarni o'z platformalarida qanchalik uzoq vaqt saqlab tura olishlariga bog'liq bo'ladi. Netflix kabi striming xizmatlari, agar sifatli tavsiyalar foydalanuvchilarni ularning saytlariga qayta-qayta jalb qila olsa, raqobatdoshlik ustunligiga ega bo'ladi.

Bu tizimlar o'z tavsiyalari uchun turli xil sun'iy intellekt vositalaridan foydalanadi. Steck va boshqalar [2021] Netflix tavsiya tizimida yuzaga keladigan muammolarni, ayniqsa chuqur o'rganishni o'zlarining tavsiya tizimlariga joriy etish masalasini tasvirlab bergan. Turli xil tavsiya beruvchi algoritmlar bilan o'tkazilgan tajribalarimiz shuni ko'rsatdiki, bu masalada «sehrli tayoqcha» (silver bullet) yo'q ekan. Eng samarali usul (u xoh chuqur o'rganish, xoh boshqa usul bo'lsin) hal qilinishi lozim bo'lgan aniq tavsiya vazifasiga hamda mavjud ma'lumotlarga bog'liq bo'ladi. Shu sababli, Netflix bosh sahifasining turli qismlari (masalan, qatorlari) uchun shaxsiylashtirilgan tavsiyalar yaratishda turli xil mashinali o'rganish modellaridan foydalaniladi [Steck va boshq., 2021].

Ijtimoiy tarmoqlarda faoliyat yurituvchi va jalb etilganlikni, demak, o'z foydasini maksimallashtirishga intiluvchi kompaniyalarning tavsiyalari qutblanishning kuchayishiga olib keladi [Sunstein, 2018; Levy, 2021]. Odamlar mo'tadil va vazmin mazmundagi kontentga qaraganda, ekstremistik qarashlarga ko'proq duch kelishadi. Bu «filtr pufaklari» va aks sado kameralarini hosil qiladi. Bunday joylarda foydalanuvchilar faqat o'z hamfikrlari bilan muloqot qilishadi [Acemoglu va boshq., 2021]

Kontent maqsadga yo'naltirilganligi sababli, jamoat arboblari, jumladan siyosatchilar, turli odamlarga turlicha, hatto bir-biriga zid bo'lgan xabarlarini yetkaza oladi. Xabarlar ommaviy bo'ladigan texnologiyalarda esa, ko'pchilik boshqalarni bezovta qilmaslik uchun o'z fikrlarini yumshatadi. Nisbiy maxfiylik va jalb etishni optimallashtirish uyg'unligi zaharli onlayn muhitni keltirib chiqaradi.

17.6. Takrorlash

Quyida ushbu bobdan o'rganishingiz kerak bo'lgan asosiy fikrlar keltirilgan:

- Munosabatli tasvirlar agent qaysi obyektlarga duch kelishini oldindan bil-magan holda modellarni berish yoki o'rganish talab qilinganda, yoki ma'lumotlar tarkibida detallar raqami va buyurtma raqami kabi identifikatorlar mavjud bo'lganda qo'llaniladi.

- Hamkorlikda filtrlash va o'rnatishga asoslangan boshqa usullar yashirin xususiyatlarni ixtiro qilish orqali boshqa namunalar yordamida munosabatlarning namunalari haqida bashorat qilish uchun ishlatilishi mumkin.

- Plastinkali modellar obyektlar ma'lum bo'lgunga qadar ehtimoliy modellar-ni aniqlashga imkon beradi.

- Oldingi boblarda tasvirlangan aksariyat ehtimoliy tasavvurlarni universal miqdoriy mantiqiy o'zgaruvchilar va parametrlarni ajratishni o'z ichiga olgan holda relyatsion qilish mumkin.

- Ko'pgina yirik korporatsiyalar asosida yotuvchi tavsiya tizimlari jalb qilish-ni optimallashtiradi, bu esa qutblanishning kuchayishiga olib keladi.

17.7. Adabiyotlar va qo'shimcha o'qish uchun materiallar

Jannach va boshqalar [2021] tavsiya qiluvchi tizimlarni ko'rib chiqadilar. Netflix mukofoti va eng yaxshi algoritmlar [2007] Bell va Koren, Jahrer va boshqalar tomonidan tasvirlangan. Hamkorlikda filtrlash algoritmi Koren va boshqalar algoritmlariga asoslangan [2009]; [2011] Koren and Belllarga murojaat qiling. MovieLens ma'lumotlar to'plamlari Harper va Konstan tomonidan tasvirlangan [2015] bo'lib, ushbu saytda mavjud: <http://grouplens.org/datasets/movielens/>. Jannach va Bauer [2020] tavsiyalarni optimallashtirish oson bo'lgan mezonlardan ko'proq narsa talab qilishning sabablarini tushuntiradilar.

Hitchcockning poliadik parchalanishi (dekompozitsiya) [1927] Trouillon va boshqalarning ishlarida bilimlar grafigini to'ldirish uchun ishlatilgan. Undan bilim grafigini bashorat qilish uchun inversiyalar bilan foydalanish Kazemi va Poole [2018] hamda Lakroix va boshqalar tomonidan mustaqil ravishda taklif qilingan [2018].

Fatemi va boshqalar [2020] ushbu metodni bir nechta argumentlari bo'lgan munosabatlar uchun kengaytiradilar.

SI statistik aloqadorligi De Raedt va boshqalar tomonidan ko'rib chiqilgan. [2016].

Plastinkali modellar Banting (1994) tomonidan taklif qilingan bo'lib, u ular-dan ta'lim jarayonini tavsiflash uchun foydalangan. Dirixlening yashirin taqsimoti va tilning plastinkali modellari Blei va boshqalarning [2003] ishlarida taqdim etil-gan.

Li va boshqalarning [2016] ishlarida haqiqatni kraudsorsing yordamida izlash muhokama qilinadi. Bu usul bu yerda keltirilgan boolean holatidan ko'ra umumiyroq bo'lishi mumkin.

Van den Broeck va boshqalar [2021] asoslarni o'rnatmasdan taxmin qilish imkonini beruvchi ko'tarilgan taxmin qilish (lifted inference) haqida kirish ma'lumotlarini taqdim etadilar.

Ehtimollik asosidagi mantiqiy dasturlashni Pul [1993] taklif etgan va u Problog [De Raedt va boshqalar, 2007]da amalga oshirilgan.

De Raedt va boshqalar [2008] hamda Getoor va Taskar [2007] ehtimoliy munosabatlar modellarini va ularni o'rganish usullarini ko'rib chiquvchi maqolalar to'plamini taqdim etadilar.

Domingos va Loud [2009] Markovning mantiqiy tarmoqlarini va (yo'naltirilmagan) relyatsion modellarning SI uchun umumiy maqsadli tasavvurni qanday ta'minlashi mumkinligini muhokama qiladilar.

Pujara va boshqalarning [2015] ishida statistik munosabatli SI usullari va ontologik cheklavlardan foydalanish orqali bilim grafiklarida bashoratlarni tuzish ko'rib chiqilgan. Ehtimoliy yumshoq mantiq Bach va boshqalar tomonidan tasvirlangan [2017]. Relatsion munosabatli bog'liqlik tarmoqlari Nevill va Jensen tomonidan ishlab chiqilgan [2007]. Grafik neyron tarmoqlari Xu va boshqalarning [2019], Hamilton [2020] hamda Chaudhri va boshqalarning asarlarida tasvirlangan. Schlichtkrull va boshqalar [2018] munosabatli (relyatsion) grafik konvolyutsion tarmoqlarini (R-GCNs) ta'riflaydilar.

Fellegi va Sunterlar [1969] tomonidan ishlab chiqilgan klassik mavjudlik va o'ziga xoslik noaniqligini hal qilish usuli qaydlarning bog'liqligi (record linkage) deb ataladi. Mavjudlik va o'ziga xoslik noaniqligini (relyatsion) munosabatli grafik modellarga integratsiyalashuvi Pasula va boshq [2003], Milch va boshq. [2005], Poole [2007]larning asarlarida ko'rib chiqilgan.

17.8. Mashqlar

17.1-mashq.

Kollaborativ filtrlash uchun gradiyent tushish algoritmining muqobilidan (17.2-rasm) $\{1, 2, 3, 4\}$ chegara qiymatlari uchun $P(\text{rating} > \text{threshold})$ ni bashorat qilishda foydalanish mumkin. Kodni shunday o'zgartiringki, u bunday ehtimollikni o'rganib olsin. [Maslahat: logistik regressiyada bo'lgani kabi chiziqli funksiya sigmoidini bashorat qiling]. Ushbu modifikatsiya eng yaxshi n filmlarini tavsiya qilish uchun yaxshiroq ishlaydimi, masalan, $n = 10$ uchun, agar maqsad eng yaxshi n ro'yxatida 5 reytingga ega filmlarning maksimal soni bo'lsa? Qaysi chegara qiymati yaxshi ishlaydi? Agar $\text{top-}n$ 4 yoki 5 ballga baholangan filmlar soniga qarab baholansa-chi?

17.2-mashq. Kollaborativ (hamkorlikda) filtrlash uchun muqobil tartibga solish quyidagi ifodani minimalizatsiya qilishni o'z ichiga oladi:

$$\sum_{(u,i,r) \in D} ((\hat{r}((u,i) - r)^2 + \lambda(ib[i]^2 + ub[u]^2 + \sum(ip[i,p]^2 + up[u,p]^2)))$$

(a) 17.1 da berilgan formulani tartibga solishdan bu qanday farq qiladi? [Maslahat: reytingi past bo'lgan obyektlar yoki foydalanuvchilarning tartibga solishini reytingi baland bo'lganlarniki bilan taqqoslang.]

(b) Ushbu tartibga solishni amalga oshirish uchun 17.2-rasmda berilgan kodni qanday o'zgartirish kerak?

(c) Test sinov ma'lumotlarida qaysi biri yaxshiroq ishlaydi? [Maslahat: har bir usul o'zgacha bo'lishi uchun siz λ -qiymatini o'rnatishingiz kerak; har bir usul uchun o'zaro tekshirish yordamida λ qiymatini tanlang].

17.3-mashq. 17.2-rasmda ko'rsatilgan stokastik gradiyent tushish algoritmini o'zgartiring, u (17.1) formulani minimalizatsiya qiladi, shunda u parametrlarni, shu jumladan (regulyarizatsiyani) tartibga solishni, bir qator misollardan so'ng moslashtiradi. Bu algoritmnining murakkabligi 17.2-rasmda berilgan algoritimga nisbatan qanday farq qiladi? Qaysi biri amalda yaxshiroq ishlaydi? [Maslahat: barcha parametrlarni muntazamlashtirish kerakmi yoki faqat partiyada ishlatiladigan parametrlarnimi? O'ylab ko'ring.]

17.4-mashq. Tasavvur qiling, Boolean parametrli tasodifiy o'zgaruvchilar young (Person) va cool(Item) Boolean buys(Person, Item)ning ota-onalari hisoblanadi. Faraz qilaylik, 3000 nafar odam va 200 ta buyum mavjud.

(a) Buni plastinka belgilash tizimida tasvirlang.

(b) Ushbu modelning asoslanishida qancha tasodifiy o'zgaruvchi bor?

(c) Ushbu modelni jadval ko'rinishida tasvirlash uchun nechta sonni ko'rsatish zarur? (Boshqa ko'rsatilgan sonlarning funksiyasi bo'lgan raqamlarni kiritmang).

(d) Shaxs to'plami {sam, chris} va Item to'plami {iwatch, mortgage, spinach} ga teng deb faraz qilib, asosli ishonch tarmog'ini chizing.

(e) Kuzatuvlarni hisobga olgan holda, Cool (iwatch) va Cool (mortgage) ni bir-biriga bog'liq qilish uchun nimalarni kuzatish mumkin edi?

17.5-mashq. 17.8-misolini ko'rib chiqing. Tasavvur qiling, X ning Y ni o'ldirish motivi shundaki, unga boshqa bir Z , haq to'layapti, Z ning ham Y ni o'ldirish motivi bor. Tegishli plastinka modelini chizing. Agar plastinkalarni chizish qiyin bo'lsa, parametrli tasodifiy o'zgaruvchilarning argumentlaridan foydalaning. Tartiblangan tasodifiy o'zgaruvchilar qaysi mantiqiy o'zgaruvchilarga bog'liqligi aniq bo'lishi kerak.

17.6-mashq. Aytaylik, sizda kinoni bashorat qilish uchun relyatsion ehtimoliy model mavjud bo'lib, u quyidagilarni ifodalaydi:

P (yoqtirishlar $(P, M) | (P)$ yosh, (M) janr)

bu yerda yosh (P) va janr (M) avvaldan mustaqil.

17.14-rasmdagi likes uchun kuzatuvlar berilgan, yosh va janr kuzatilmagan holatda, age (Sem) so'rovi uchun asosiy ishonch tarmog'ining (keraksiz o'zgaruvchilar olib tashlangandan so'ng) daraxt kengligi qanday bo'ladi?

(b) Xuddi shu ehtimollik modeli uchun m ta film, n ta odam va r ta reyting mavjud bo'lganda, faqat reytinglar kuzatilgan holatda, tegishli grafikning eng yomon holat daraxt kengligi (keraksiz o'zgaruvchilar olib tashlangandan so'ng) qanday bo'ladi? [Maslahat: Daraxt kengligi kuzatishlarning tuzilishiga bog'liq; daraxt kengligini maksimal darajada oshirish uchun kuzatishlar qanday tuzilishi mumkinligini o'ylab ko'ring.]

(c) Aynan shu ehtimollik modeli uchun m ta film, n ta odam va r ta reyting bo'lgan holatda, faqat ayrim reytinglar, lekin barcha janrlar kuzatilganda, mos keladigan grafikning eng yomon holatdagi daraxt kengligi qanday bo'ladi?

17.7-mashq. Maktab o'quvchilari ko'p xonali sonlarni qo'shishda yo'l qo'yadigan xatolarni aniqlash va tahlil qilish ustida o'ylab ko'ring. Uchinchi ko'p xonali sonni hosil qilish uchun ikkita ko'p xonali sonni qo'shishni ko'rib chiqing, masalan, quyida berilgan ifoda kabi:

$$\begin{array}{r} A_1 \quad A_0 \\ + \quad B_1 \quad B_0 \\ \hline C_2 \quad C_1 \quad C_0 \end{array}$$

Bu yerda A_i , B_i va C_i barcha raqamlarni anglatadi.

Aytaylik, siz o'quvchilarning bir xonali sonlarni qo'shish va yodda saqlash ko'nikmalariga ega yoki ega emasligini modellashitmogchisiz. Agar o'quvchilar buni qanday bajarish usulini bilsalar, odatda to'g'ri javobni topadilar, lekin ba'zi xatolarga yo'l qo'yadilar. Bilmaydigan o'quvchilar esa shunchaki taxmin qiladilar. Bitta o'quvchi va ikkita ikki xonali sonni qo'shib uch xonali son hosil qilish masalasi uchun ishonch tarmog'ini chizing. [Maslahat: A_i , B_i va C_i larning har biri, shuningdek, yodda saqlanadigan sonlar ham o'zgaruvchilar bo'lib, o'quvchining har bir ko'nikmasi uchun ham alohida o'zgaruvchi mavjud.]

Bir nechta o'quvchilar va bir nechta masalalar bor deb faraz qilaylik. Plastinka tasvirini chizish yoki parametrlangan tasodifiy o'zgaruvchilarni belgilash orqali buni ifodalang. O'quvchilar hamda muammolar uchun alohida plitalar bo'lishi kerak. O'quvchiga, muammoga yoki har ikkalasiga ham qaysi o'zgaruvchilar bog'liq bo'lishini o'ylab ko'ring.

Person	Movie	Likes
Sam	Hugo	yes
Chris	Hugo	no

Sam	Avatar	no
Sam	HarryPotter 6	yes
Chris	HarryPotter 6	yes
Chris	AI	no
Chris	Avatar	no
David	AI	yes
David	Avatar	yes

17.14-rasm. 17.6-misol uchun taqdim etilgan ma'lumotlar

(c) Endi tasavvur qiling, siz bir nechta raqamni va ularning bir necha marta takrorlanishini ham ifodalashni xohlaysiz. O'quvchilarning bilimlari vaqt o'tishi bilan o'zgarishi mumkinligini nazarda tuting. Plastinka modelini chizing. Vaqt va raqamlar uchun plastinka modellarida o'quvchilar va masalalarda yuzaga kelmagan qanday muammo paydo bo'ladi?

17.8-mashq. Avvalgi boblarda ko'rib o'tilgan elektr domenini ehtimollik asosidagi mantiqiy dastur ko'rinishida ifodalang, shunda u AIPython (aipython.org) yoki Problog dasturlarida ishlasin. Ushbu ifoda 9.15-misoldagi ehtimollik bog'liqliklarini va 15.11-misoldagi munosabatlarni o'z ichiga olishi kerak.

VI QISM. UMUMIY MANZARA

Sining ijtimoiy oqibatlari, SI fan va texnologiyasining kelajakdagi ssenariylari qanday?

18-bob. SUN'IY INTELEKTNING IJTIMOIIY TA'SIRI

Insoniyat tarixida hech qachon biz mashinaga mustaqil ravishda kim yashashi va kim o'lishi kerakligini bir necha soniya ichida, real vaqt nazoratisiz belgilashga ruxsat bermaganmiz. Biz bu ko'priki har qanday vaqtda kesib o'tamiz va bu uzoqdagi harbiy operatsiyalar teatrida emas, balki hayotimizning eng oddiy jihatlaridan biri – har kuni transportda sodir bo'ladi. Avtomobillarimizga axloqiy qarorlar qabul qilishga ruxsat berishdan oldin, biz global mulot o'tkazishimiz kerak, bu orqali biz axloqiy algoritmlarni loyihalovchi kompaniyalarga va ularni tartibga soladigan siyosatchilarga o'z istaklarimizni bildiramiz.

– Avad va boshqalar [2018, 63-bet]

Sun'iy intellekt g'oyalar, algoritmlar va vositalarning o'zgaruvchan to'plamidir. Sun'iy intellekt tizimlari endi real dunyoda keng miqyosda joriy qilinmoqda [Littman va boshqalar, 2021; Zhang va boshqalar, 2022a]. Ular inson faoliyatining deyarli barcha shakllariga, jumladan iqtisodiy, ijtimoiy, psixologik, sog'liqni saqlash, huquqiy, siyosiy, hukumat, ilmiy, texnologik, ishlab chiqarish, harbiy, media, ta'lim, badiiy, transport, qishloq xo'jaligi, ekologik va falsafiy sohalariga sezilarli ta'sir ko'rsatmoqda. Ushbu ta'sirlar foydali bo'lishi, lekin zararli ham bo'lishi mumkin. Avad va boshqalar tomonidan ko'tarilgan axloqiy va ehtimol, tartibga solish muammolari nafaqat o'ziyurar mashinalarga, balki sun'iy intellektning barcha sohalarida qo'llaniladi.

Avtonom agentlar o'zlari sezib, qaror qabul qilishadi va harakat qilishadi. Ular **yarimavtonom agentlar** bilan birgalikda texnologiyadagi radikal, sifatli o'zgarishni va texnologiyaga bo'lgan qarashimizni ifodalaydi. Bunday agentlar inson nazoratidan tashqaridagi kutilmagan harakatlarni amalga oshirishi mumkin. Har qanday keskin o'zgarishlarga sabab bo'luvchi texnologiyalarda bo'lgani kabi, sezilarli foydali va zararli oqibatlar yuzaga kelishi mumkin – ularning ko'pchiligi baholash qiyin, ko'pchiligini esa odamlar oddiygina ko'rmaydi yoki ko'rishni istamaydi.

Foydalanuvchilar tomonidan yaratilgan katta ma'lumotlar to'plamida o'qitilgan chuqur o'rganish va ehtimollik modellariga o'xshash SI algoritmlariga taya-

nuvchi ijtimoiy media platformalarini ko'rib chiqing. Ushbu platformalar odamlarni bog'lanish, muloqot qilish va sayyora bo'ylab sog'lom tarzda ijtimoiy guruhlar tuzishga imkon beradi. Biroq, platformalar odatda foydalanuvchi lentasini reklama daromadlarini oshirishda jalb qilishni maksimal darajada oshirish uchun optimallashtiradi. Jalb qilishni maksimal darajada oshirish ko'pincha qarama-qarshi xatti-harakatlar va polarizatsiyaga olib keladi. Platformalar, shuningdek, demokratik saylovlarga salbiy ta'sir ko'rsatadigan bo'linishlarga olib keluvchi siyosiy bahs-munozaralarni kuchaytirish va «deep fake» (chuqur qalbaki) kabi boshqa zararli oqibatlarni keltirib chiqarish uchun ham manipulyatsiya qilinishi mumkin. Ular foydalanuvchilarning maxfiyligini buzishi ham mumkin. Avtomatlashtirilgan qaror qabul qilish tizimlari, ehtimol, tarfakash bo'lib, odamlarga kreditlar, ipoteka kreditlari va sug'urta polislarini tasdiqlash uchun ishlatiladi, hattoki potensial xodimlarni saralash uchun ham qo'llaniladi.

Odamlar ularga adolatli munosabatda bo'lish, qarorlar ustidan shikoyat qilish, javobgarlikka tortish va ishonchlilikni ta'minlash va shaxsiy hayot huquqiga ega bo'lishlarini kutishadi. Ushbu huquqlarning haqiqatan ham amalga oshirilishini ta'minlash mumkinmi?

18.1. Raqamli iqtisodiyot

Sun'iy intellekt fanlari, texnologiyalari va ularning qo'llanilishi keng tarqalgan raqamli aloqa va Internet davrida tez sur'atlar bilan rivojlandi. Bu rivojlanishlar dunyo iqtisodiyotini tubdan o'zgartirdi. Bozor kapitallashuvi bo'yicha o'lingan dunyoning o'nta yirik global korporatsiyasining aksariyati SI dasturlariga qattiq bog'liq. Ushbu kompaniyalar asosan moddiy mahsulotlar ishlab chiqarishdan ko'ra ko'proq axborotni foydalanishga yo'naltirilgan. Materiya o'rniga axborotdan foydalanishga o'tish **dematerializatsiya** sifatida tavsiflanadi.

Jismoniy pochta, elektron pochta, SMS va ijtimoiy media orqali o'zgartirildi. Shu bilan birga, elektron pochta spam bilan to'lib ketgan edi, lekin SI usullari yordamida u filtrlash uchun ishlatildi. Bosma darsliklar endi elektron darsliklar bilan to'ldiriladi. Analog fotografiya, kino va video raqamli media bilan almashtirildi. Samolyotlar va avtomobillarda sayohatning bir qismi raqamli aloqa bilan almashtirildi. CD disklar o'rnini audiooqimlar va gazetalar yangiliklar veb-saytlari egalladi. Atomlarni bitlarga aylantiruvchi bu jarayon tranzaksiyalarni kamroq qiyinchilik va katta tezlik bilan amalga oshirish imkonini beradi. CD sotib olish uchun do'konga borishdan ko'ra, musiqa oqimini uzatish osonroq, tezroq, arzonroq, materiallar va energiya jihatidan samaraliroqdir.

Raqamlashtirish, o'z navbatida, jamiyat va iqtisodiyotning umumiy tezlashuviga olib keladi. Shuningdek, telekommunikatsiya orqali masofalarni qisqartiradi. Hozir barchamiz Marshall McLuhan tomonidan 1962-yilda tavsiflangan global qishloqda yashaymiz. Bundan tashqari, raqamli inqilob ishlab chiqaruvchilar va

iste'molchilar o'rtasida vositachilar, masalan, chakana sotuvchilar, bank xodimlari va sayohat agentlariga bo'lgan ehtiyojni kamaytiradi yoki yo'q qiladi – bu jarayon **disintermediation** (vositachilikni yo'q qilish) deb nomlanadi.

Raqamli inqilob va sun'iy intellekt global iqtisodiyotni o'zgartirmoqda. Ushbu effektlar ba'zilar uchun foydali, ammo boshqalar uchun zararli. Yangi iqtisodiyotda foyda va zarar juda notekis taqsimlangan. **G'olib hamma narsani qo'lga kiritadi (winner-take-all)** tendensiyasi mavjud, chunki eng qudratli korporatsiyalar bozorda monopol yoki oligopolistik mavqega ega bo'lgunga qadar o'z hukmronligini kuchaytirish uchun o'z kuchlaridan foydalanadilar. Sun'iy intellekt va mashinaviy o'rganish algoritmlari, foydalanuvchilarni kuzatish va modellashirishga asoslangan, ushbu dinamikada markaziy rol o'ynaydi. Zuboff [2019] yangi **iqtisodiyotni nazorat kapitalizmi** deb tavsiflagan, bu shaxsiy ma'lumotlarni katta miqyosda yig'ish va savdo-sotiq hamda siyosiy maqsadlarda nishonli monitoring va reklama qilish uchun foydalanishni o'z ichiga oladi. Maxfiylik, qadr-qimmat, adolat, xilma-xillik va kiritilish kabi insoniy qadriyatlar buziladi.

Inson diqqatini, mavjud ma'lumotlarga tanlab diqqat qilishni, muhim va cheklangan resurs deb atash mumkin. Diqqat psixologik muammo bo'lib, shuningdek, iqtisodiy muammo sifatida ham qaraladi, bu haqda Simon [1971] tomonidan uzoq vaqt oldin diqqat iqtisodiyoti asosiy tushunchasini yaratganida ta'kidlangan.

Ma'lumotga boy dunyoda, ma'lumotning ko'pligi boshqa narsaning yetishmasligini anglatadi: ma'lumot iste'mol qiladigan narsaning tanqisligidir. Ma'lumot nimani iste'mol qilishi juda ravshan: u uning qabul qiluvchilarining diqqatini iste'mol qiladi. Shunday qilib, ma'lumotning ko'pligi diqqatning kamligini yaratadi va ushbu diqqatni uni iste'mol qilishi mumkin bo'lgan ko'plab ma'lumot manbalari orasida samarali taqsimlash zaruriyatini keltirib chiqaradi.

Ingliz tilida biror kishi muhim voqea uchun «diqqat qaratmoqda» deb aytiladi. Boshqacha qilib aytganda, diqqat bu iqtisodiyotda pul birligi kabi sarflanadigan valyutadir. Inson diqqatining bizning ekranlarga to'la raqamli davrimizdagi markaziy roli Richtel [2014] tomonidan tasvirlangan. Diqqatni tovar sifatida aylantirish foydalanuvchilarni kuzatishni talab qiladi, bu esa, o'z navbatida, maxfiylik bilan bog'liq xavotirlarni keltirib chiqaradi. Korporatsiyalar va boshqa ishtirokchilar nafaqat biz haqimizda ko'p narsalarni bilishni xohlashadi, balki ular ushbu bilimdan bizning diqqatimizni, fikrlarimizni va harakatlarimizni boshqarish uchun foydalanishadi.

18.2. Qadriyat va bias

Katta ma'lumotlar to'plamida o'qitilgan o'quv tanlanmalar o'quv to'plamlarida mavjud bo'lgan har qanday og'ishni aks ettiruvchi natijalarni beradi. Ma'lumotlar to'plamlari o'tmishda olinganligi sababli, kelajakda natijalarni bashorat qilish uchun ulardan foydalanish o'tmishdan kelajakka har qanday og'ishni ko'r-

satadi. Agar kelajak o'tmishga o'xshamasa yoki o'xshamasligi kerak bo'lsa, nima bo'ladi?

Mashinaviy o'rganishda bias(tarafkashlik/og'ish) neytral texnik ahamiyatga ega, «bir farazni boshqasidan ustun qo'yish tendensiyasi» hisoblanadi. **Bepul tushlik teoremasi (No-free-lunch theorem)** har qanday samarali o'rganish algoritmi shu ma'noda tarafkashlikka ega bo'lishi kerakligini anglatadi. Ammo umumiy til ishlatilishida inson tarafkashligi salbiy ma'noga ega bo'lib, «bir narsa, shaxs yoki guruhning boshqasiga nisbatan foydasiga yoki unga qarshi tarafkashlik, odatda adolatsiz deb hisoblanadi» degan ma'noni anglatadi [Stivenzon va Lindberg, 2010]. Yuzni tanish uchun trening to'plamlari, ko'pincha xabardor ruxsatsiz to'plangan, odatda odamlarni adolatli tarzda ifodalamaydi, bu esa noto'g'ri tasniflashga olib keladi, ko'pincha bu zararli ta'sirga ega bo'ladi (qarang, 7.7-mavzu). Katta matnli massivlarda oldindan ishlangan **katta til modellari** so'ralganda ko'pincha irqchi, seksist yoki boshqa yo'l bilan inson qadr-qimmatini kamsitadigan yangi matnni yaratadi.

Har qanday sun'iy intellektga asoslangan qaror qabul qilish tizimi tabiatan ma'lum bir yashirin qadriyatlar yoki afzalliklarni aks ettiradi. Asosiy savol shundaki: bu qaysi qadriyatlar? Odatda, sun'iy intellekt tizimiga kiritilgan qadriyatlar tizimning dizayneri yoki egasining qadriyatlari yoki chuqur o'rganish o'quv to'plamida yashirin bo'lgan qadriyatlar bo'ladi. Keyingi savollar tug'iladi: bu qadriyatlarini ochiq ko'rsatish mumkinmi? Ularni aniqlab belgilash mumkinmi? Ushbu qadriyatlar demokratik bo'lib, kamsitish va tarafkashlikdan qochish mumkinmi? Ular tizim foydalanuvchilari yoki maqsadlar uchun shaffof bo'lishi mumkinmi? Ular to'g'ridan to'g'ri yoki bilvosita ta'sir ko'rsatishi mumkin bo'lgan hamma odamlarning qadriyatlarini aks ettiradimi? Shaxsiy hayot, qadr-qimmat, adolat, xilma-xillik va kiritilishni hurmat qiladigan tizimlarni loyihalash mumkinmi?

O'quv ma'lumotlaridagi ijtimoiy qismlilikning roli 7.7-mavzuda tasvirlangan. Katta korpuslarda o'qitilgan zamonaviy chuqur o'rganish tizimlaridagi qismlilik va boshqa ijtimoiy ta'sirlar 8.7-mavzuda muhokama qilingan. Ushbu savollar, sun'iy intellekt tizimlari dizaynerlari uchun doimiy muammolar, O'Neil [2016], Eubanks [2018], Noble [2018], Broussard [2018], Benjamin [2019] Bender va boshqalar [2021] tomonidan tanqidiy o'rganilgan.

18.3. Insonga yo'naltirilgan sun'iy intellekt

Meri Uolstounkraft Shellining «Frankenshteyn yoki zamonaviy Prometey» [Shelley, 1818] birinchi haqiqiy ilmiy fantastika romanidir. Uni axloqiy hikoya sifatida o'qish mumkin, buni Shelleyning muqobil sarlavhasi «*Zamonaviy Prometey*» tasdiqlaydi. Qadimgi yunon mifologiyasiga ko'ra, Prometey ma'budlardan olovni o'g'irlab, insoniyatga bergan. Zevs Prometeyni texnologiya va bilimlarni o'g'irlagani uchun jazoladi va uni abadiy azobga mahkum qildi. Doktor Fran-

kenshteyn ijodi inson urf-odatlari va tilini o'rganish orqali insoniyat jamiyatiga singib ketishga harakat qildi, lekin insoniyat uni rad etdi va uning chinakam xayrli ishlarini noto'g'ri talqin qildi. Bu rad etish va uning yolg'izligi, o'rtog'i yo'qligini hisobga olib, qasos olishni tanlashiga olib keldi. Endi Frankenshteyn yirtqich hayvoni odamlarga qarshi o'girilib bo'lmaydigan, boshqarib bo'lmaydigan texnologiyaning ramziga aylandi.

Texnologiyaning nazorati to'g'risidagi xavotirlar sun'iy intellekt bizning dunyomizni o'zgartirgan sari tobora muhimlashmoqda. Shunday qilib, shunday ataluvchi **sun'iy umumiy intellekt (artificial general intelligence (AGI))** haqida muhokamalar insonlar tomonidan bajariladigan turli xil vazifalarda yuqori ko'rsatkichlarga ega tizimlarni tasavvur qiladi, bu esa ma'lum vazifalar uchun tizimlarni rivojlantiradigan va o'qitadigan «tor» sun'iy intellektidan farq qiladi. Ba'zilar AGI insonlarni dominatsiya qiladigan **superintellektga** o'tuvchi **singulyarlikka** olib kelishi mumkin deb ishonishadi [Good, 1965]. Yoki Bostrom [2014] taxmin qilganidek, AGI tizimi, agar maqsadi koinotdagi qog'oz qisqichlarining sonini maksimal darajada oshirish bo'lsa, insonlar uchun zarur bo'lgan resurslarni o'z ichiga olgan barcha mavjud resurslarni iste'mol qilishi mumkin. Ushbu bema'ni tuyulgan fikrlash tajribasi shuni ko'rsatadiki, agar uning maqsadlari noto'g'ri belgilangan yoki boshqacha tarzda odamlarning uzoq muddatli omon qolishi va tabiiy muhitga mos kelmasa, sog'lom aql-idroksiz, zararsiz AGI insoniyatga ekzistensial tahdid solishi mumkin. Ushbu xavfsizlik muammosi muvofiqlashtirish muammosi sifatida tanilgan [Christian, 2020].

Bevosita tahdid shundaki, sun'iy intellekt tizimlari, masalan, o'z-o'zini boshqaradigan avtomobillar va halokatli avtonom qurollar, insonlar nazorati bo'lmagan holda hayot va o'lim qarorlarini qabul qilishi mumkin. Dramatik jihatdan kamroq, sun'iy intellekt tizimlari inson farovonligiga ta'sir ko'rsatadigan zararli, hatto hayot uchun xavfli bo'lmasa ham, qadriyatga asoslangan qarorlar qabul qilishi mumkin, masalan, kimga ipoteka yoki ish taklifi berilishi kerakligini belgilashdir. Bu insoniy nazorat va avtonomlikka e'tiborni qaratishga olib keldi. **Loyihachilar insonga yo'naltirilgan sun'iy intellekt yoki insonga mos keluvchi sun'iy intellektni qanday yaratishlari mumkin?** Inson qadriyatlarini sun'iy intellekt tizimlariga singdirish mumkinmi? Ushbu savollar Russell [2019], Marcus, Davis [2019] va Shneiderman [2022] tomonidan o'rganilgan. Inson qadriyatlarini qo'shish uchun taklif qilingan bir usul – **Inson fikriga asoslangan mustahkamlash ta'limi (Reinforcement Learning from Human Feedback (RLHF))**dir [Knox va Stone, 2009]. RLHF u bema'ni tuyulgan fikrlash tajribasi, sog'lom fikrga ega bo'lmagan holda, zararsiz ko'rinadigan sun'iy intellekt, agar uning maqsadlari noto'g'ri aniqlangan bo'lsa yoki odamlar va atrof-muhitning uzoq muddatli omon qolishiga mos kelmasa, insoniyat uchun ekzistensial xavf tug'dirishi mumkinligini ko'rsatish uchun mo'ljallangan. RLHF ChatGPT kalit moduli uchun asosdir [Open AI, 2022].

Murakkab vaziyatlarda, inson qaror qabul qiluvchilariga **yarimavtonom agentlar** yordam berishi tobora ko'paymoqda. Bu kombinatsiya **insonning aralashuvi** deb ataladi. 2-bobda ko'rsatilganidek, aqlli tizimlar ko'pincha boshqaruvchilarning iyerarxiyasi sifatida tuziladi, pastki darajalarda tezkor harakat qilib, qisqa vaqt oralig'ida ishlaydi, yuqori darajalar esa uzoq vaqt oralig'ida, ko'proq simvolik ma'lumotlar asosida sekin ishlaydi. Insonlar iyerarxik tuzilgan tizimlar bilan o'zaro aloqalari odatda yuqori darajalarda sodir bo'ladi. Inson haydovchilari avtomobilning antitormoz tizimlarini real vaqtda mazmunli ravishda o'zgartira olmaydilar, ammo ular yuqori darajadagi navigatsiya afzalliklari yoki ko'rsatmalarini bera olishadi. Insonlar avariylardan qochish uchun burish yoki tormozlashlari mumkin, lekin bu faqat diqqatlarini jamlaganlarida; ammo transport vositalari yanada avtomatlashtirilganda, haydovchi diqqatini chalg'itishi yoki uxlashi mumkin va vaqtda e'tiborini qayta yo'naltira olmaydi.

Neyron tarmoqlardagi **diqqat** tushunchasi **insonning diqqati** tushunchasidan ilhomlanib yaratilgan. Inson diqqatiga bevosita bog'liq tushunchalarga **hushyorlik**, ya'ni mumkin bo'lgan xavf uchun ehtiyotkorlik bilan kuzatish holati va **ahamiyatlilik**, ya'ni ayniqsa sezilarli yoki muhim bo'lish sifati kiradi. Odamlarga ular bilan samarali muloqot qilish imkonini beruvchi sun'iy intellekt tizimlarini ishlab chiqish hushyorlik, ahamiyat va e'tiborning iqtisodiy, ijtimoiy, psixologik va axloqiy rolini tushunadigan dizaynerlarni talab qiladi. Inson e'tibori va hushyorligi bo'yicha dastlabki tadqiqotlar N. H. Makvort [1948] va J. F. Makkuort [1970]. Mole [2010] asarida diqqatning falsafiy nazariyasi keltirilgan. Archer [2022]da to'plangan maqolalar ahamiyatlilik, e'tibor va axloq bilan bog'liq masalalar qanday kesishishini ko'rsatadi.

Interaktiv sun'iy intellekt tizimlarining loyihachilari **inson-kompyuter o'zaro ta'siri** (HCI) [Rogers va boshq., 2023] va sun'iy intellekt tamoyillari va amaliyotlari bilan yaxshi tanish bo'lishlari kerak. Sun'iy intellekt uchun yaxshi dizaynlar ishonchli tizimlar yaratishda muhim ahamiyatga ega. Masalan, Amershi va boshqalarning [2019] «Inson–AI o'zaro ta'siri uchun yo'riqnomalari» tizim noaniq bo'lganida kamroq ish qilish strategiyalarini taqdim etadi, bu esa noto'g'ri taxminlarning xarajatlari va oqibatlarini kamaytirishga yordam beradi.

Nogironlar va keksa aholi uchun **yordamchi texnologiyalarni** ko'plab tadqiqotchilar va kompaniyalar rivojlantirmoqda. Xotira eslatmalari kabi **yordamchi kognitsiya** bir dastur hisoblanadi. **Yordamchi sezgi** va **yordamchi harakat**, aqlli aravachalar, qariyalar uchun hamrohlik va uzoq muddatli parvarish muassasalarida hamshiralarning yordamchilari shaklida foydali texnologiyalar hisoblanadi. Yordamchi texnologiya tizimlari Pollack [2005], Liu va boshq. [2006], Yang va Mackworth [2007] tomonidan tavsiflangan. **Yarim avtonom** aqlli aravachalar Mihailidis va boshq. [2007], Viswanathan va boshq. [2011] tomonidan muhokama qilingan. Biroq, Sharkey [2008] va Shneiderman [2022] qariyalar va juda yoshlar uchun hamroh sifatida robot yordamchilarga suyanishning ba'zi xavflari haqida

ogohlantiradi. Avtonom transport vositalari bilan bir xil, tadqiqotchilar o'z yaratuvchilari va foydalanish jarayoniga oid muhim savollarni berishlari kerak. Yordamchi texnologiya va boshqa sun'iy intellekt ilovalari tadqiqotchilari va ishlab chiquvchilari, Charlton [1998] tomonidan taqdim etilgan nogironlar huquqlari harakati diktumidan xabardor bo'lishlari kerak: «biz haqimizda hech narsa yo'q».

SI tizimlarini inson nuqtayi nazaridan baholash uchun ko'plab tushunchalar mavjud: **shaffoflik, izohlanish, tushunarli bo'lish, adolat, xavfsizlik, mas'uliyat va ishonchlilik**. Bu tushunchalar foydali, lekin ular ko'p, bir-biriga to'g'ri keladigan, o'zgaruvchan va bahsli ma'nolarga ega. **Shaffoflik** odatda SI dasturining atrofidagi to'liq ekosistemani, shu jumladan, trening ma'lumotlarini, dastur testini va sertifikatsiyasini hamda foydalanuvchi maxfiylik masalalarini tasvirlaydi. Biroq, shaffoflik, insonlar foydalanadigan modellarni va ma'lum bir qarorning ortidagi sabablarni tushunishi mumkin bo'lgan SI tizimlarini ham tasvirlash uchun ishlatiladi. Masalan, chuqur o'rganishga asoslangan qora quti SI tizimlari, bu nuqtayi nazardan, shaffof emas. Dunyo qanday ishlashini tushunadigan tizimlar, sababiy modellarni ishlatib, izoh berish imkoniyatiga ega bo'lishi mumkin. Masalan, Sreedharan va boshqalar [2022] tomonidan rejalashtirish nuqtayi nazaridan izohlanadigan inson-SI o'zaro ta'siri haqida taqdimotga qarang. Izohlanishdagi yaxshilanishlar, Russell [2019] taklif qilgandek, dastur ishonchligini oshirishi mumkin.

Yaxshilangan shaffoflik, tushunarlilik va adolat ham ishonchlilikni oshirishi mumkin. Tushunarlilik ishlab chiquvchilar uchun muammolarni baholash, tuzatish va kamaytirish uchun foydalidir. Biroq, uning oxirgi foydalanuvchilar uchun har doim foydali ekanligi to'g'risidagi dalillar kamroq ishontiruvchidir. Bashoratlar va harakatlarning sabablarini tushunish tushunarlilikni **izohlovchi sun'iy intellektning** mavzusi hisoblanadi. Agar tizim o'z xulosasini tushuntira oladigan bo'lsa, bu yaxshi bo'lishi va aniq ko'rinishi mumkin. Biroq, noto'g'ri xulosani tushuntira oladigan tizimga ega bo'lish, ayniqsa tushuntirish taxminiy bo'lsa, yaxshilikdan ko'ra ko'proq zarar etkazishi mumkin. Bansal va boshqalar [2021] «Tushuntirishlar insonlarning sun'iy intellektning tavsiyasini qabul qilish imkoniyatini oshirdi, bu uning to'g'riligi bilan bog'liq emas» deb ko'rsatadi.

7.7-mavzuda muhokama qilinganidek, hisoblash tizimlari tomonidan yaratilgan modellar odamlar qila olmaydigan usullarni tekshirish uchun ochiqdir. Tekshiruv va sinovlar barcha nodir hodisalarni yoki haqiqiy hududlar uchun **burchak holatlarini** qamrab olmasligi mumkin. Ularning xatti-harakatlari har doim aniq **xavfsizlik va maqsad cheklovlarini** o'z ichiga olgan rasmiy spetsifikatsiyaga mos kelishini ta'minlash uchun sinov tizimlari ularni yanada ishonchli qilishi mumkin [Mackworth va Zhang, 2003]. Odamlar bilan doimiy ravishda o'zaro aloqada bo'lib, hamkorlik qiluvchi **yarimavtonom tizimlar** ishonchli ekanliklarini isbotlasalar, ko'proq ishonchga ega bo'lishlari mumkin; ammo, bu ishonch ba'zi noaniq holatlar uchun noto'g'ri bo'ladi. 12.6-mavzuda ochiq va hisobdor

guruh qarorlarini qabul qilishda aniq foydalar rolini tasvirlab berilgan. 13.10-mavzuda kuchaytirish orqali o'rganishning real dunyo sharoitida qo'llanilishi borasidagi xavotirlar keltirilgan.

18.4. Mehnat va avtomatlashtirish

Umuman avtomatlashtirishning va xususan, SIning ish va bandlik tabiatiga ta'siri iqtisodchilar va ijtimoiy olimlar tomonidan keng o'rganilgan; ammo, avtomatlashtirish yoki sun'iy intellektning ta'siri haqida hali konsensus yo'q. Ba'zilarining ta'kidlashicha, SI, shu jumladan robototexnika ommaviy ishsizlikka olib keladi; boshqalar esa, sun'iy intellektga asoslangan ishlanmalar asosida ko'plab yangi ish toifalari paydo bo'lishini ta'kidlaydilar. Ushbu hodisaga qarashning eng yaxshi usuli – har qanday ish bir qator ko'nikmalarni talab qilishini tushunishdir. Ushbu ko'nikmalarning ba'zilari haqiqatan ham keraksiz bo'lishi mumkin. Boshqa ko'nikmalar zarur bo'lib qolishi va yangilashni ham talab qilishi mumkin. Yuqorida muhokama qilinganidek, **vositachilikni** yo'qotish ko'plab ish toifalarini yo'q qiladi, ammo boshqa ish toifalarining «malakasini oshirish»ni ham talab qiladi. Bu savollar Brynjolfsson va McAfee [2014], Agrawal va boshqalar [2019] hamda Ford [2021] tomonidan o'rganilgan. Agrawal va boshqalar [2022] tomonidan ishlab chiqilgan mavzulardan biri shundaki, biznes qarorlari prognoz va mulohazani talab qiladi. Mashinani o'rganish endi bashorat qilishni avtomatlashtirishga imkon beradi, shuning uchun insoniy fikrlash va qarorlarni tahlil qilish ko'nikmalari nisbatan qimmatroq bo'ladi. Danaher [2021] avtomatlashtirish etikasi va ish kelajagini ko'rib chiqadi.

Sun'iy intellekt va unga aloqador texnologiyalar ko'plab yangi ish toifalarini yaratmoqda; ammo, yangi post-industrial yuqori texnologiyali korporatsiyalar odatda eski sanoat iqtisodiga asoslangan korporatsiyalar bilan bir xil bozor o'lchamida ancha kam odamni ishga oladilar. Sun'iy intellekt hozirda butun iqtisodiyotga kirib bormoqda, eski sanoat korporatsiyalarida, masalan, avtotransport va boshqa ishlab chiqarish sektorlari, shuningdek, sog'liqni saqlash, huquq, ta'lim, ko'ngilochar va moliya sohalarida sun'iy intellektga oid ish o'rinlari yaratilmoqda. O'yin sanoatidagi sun'iy intellektning roli haqida 6.6-mavzuda muhokama qilingan. Ehtimol, jamiyatning tovar va xizmatlarini ishlab chiqarish uchun qo'shimcha odam talab qilinadi. Bundan tashqari, sun'iy intellekt shunchalik qimmatki, yangi boylik imkoniyatlarini yaratishi mumkinki, har bir kishi uchun shartsharoitlari kafolatlangan universal kerakli natijani taqdim etish mumkin va zarur, shunday qilib, bu boylikni adolatli tarzda qayta taqsimlash mumkin [Ford, 2021]. UBI uchun asoslangan argument shundaki, sun'iy intellekt ko'p qo'l mehnati va aqliy mehnatga bo'lgan ehtiyojni kamaytiradi, shuning uchun inson huquqlari uy-joy va oziq-ovqatga to'liq ish haqi bilan bog'liq bo'lmasligi kerak. Bu ko'proq ijodiy dam olish va norasmiy parvarish imkoniyatini yaratishi mumkin.

Ish bilan bandlik manzarasi sezilarli darajada o'zgarmoqda, sun'iy intellekt tomonidan buzilmoqda. Ko'pgina ishchilar endi qisqa muddatli maxsus shartnomalar asosida ishlaydigan ish joylari portfeliga ega. Gig iqtisodiyoti deb ataladigan texnologiya iste'mol tovarlari va xizmatlarini, jumladan, transportda tashish va oziq-ovqat mahsulotlarini o'z vaqtida buyurtma qilish va yetkazib berish uchun zarur bo'lgan resurslarni sun'iy intellekt yordamida rejalashtirish va tashkil etish imkonini beradi. Boshqa tomondan, ishchilar kasaba uyushma tashkiloti bilan bog'liq imtiyozlar va himoyani yo'qotadilar, shu jumladan ish xavfsizligi, ish haqi va nafaqalar bo'yicha muzokaralar olib borish imkoniyati, masalan, ta'til, pullik kasallik ta'tillari, pensiyalar va tibbiy sug'urta (agar u universal bo'lmasa). Ushbu muammolarni hal qilish uchun davlat qonunchiligi, me'yoriy hujjatlar va huquqni qo'llash amaliyotini takomillashtirish taklif etiladi.

18.5. Transport

Odamlar va yuklarni tashish iqtisodiyotning turli xil ijtimoiy ehtiyojlarini qondiradigan asosiy sohasidir. Bu sun'iy intellektning ijtimoiy va iqtisodiy ta'sirini o'rganish uchun foydali misol bo'lib xizmat qiladi. **Avtonom transport vositalari** ishlab chiqilmoqda va joriy etilmoqda. O'z-o'zini boshqaradigan transport vositalarida aniq joylashishni aniqlash uchun ishlatiladigan texnologiyalar 9.8-mavzuda tasvirlangan. Bilan bog'liq ba'zi axloqiy qarorlar o'z-o'zini boshqarish vositalari 2.4-mavzuda ko'rib chiqiladi. Avtomatlashtirilgan marshrutni rejalashtirishda afzalliklarning roli 3.9-mavzuda muhokama qilinadi. Avtotransport parkidan foydalangan holda yetkazib berishni rejalashtirish uchun cheklovlardan foydalanish 4.9-mavzuda tasvirlangan. Aqlli avtomobillar va yuk mashinalaridan foydalanishning ijobiy ta'siri sezilarli bo'lishi mumkin [Thrun, 2006]. Yo'llarda yillik qirg'inini kamaytirishning xavfsizlik jihatidan mavjud. Har yili dunyo bo'ylab yo'l-transport hodisalarida taxminan 1,2 million kishi halok bo'ladi va 50 milliondan ortiq kishi jarohatlanadi [Peden va boshq., 2004]. Avtomobillar chorrahalarida muloqot qilishlari va muzokaralar olib borishlari mumkin edi. Natijada baxtsiz hodisalarning kamayishi bilan bir qatorda, transport hajmi uch baravar oshishi mumkin [Dresner va Stoun, 2008].

Yo'l samaradorligini oshirishga chorrahalarini oqilona boshqarish va avtomatlashtirilgan transport vositalarining bir-birini xavfsiz kuzatib borishini ta'minlaydigan «vzvod» effekti orqali erishiladi, chunki ular harakat qilishdan oldin o'z niyatlari haqida xabar berishlari va inson haydovchilariga qaraganda tezroq javob berishlari mumkin. Ushbu transport vositalari o'z harakatlaridan oldin niyatlarini aloqa qilib, inson haydovchilardan ancha tezroq harakat qilishadi. Yo'ldan yanada samarali foydalanish, nafaqat avtomagistrallarning kapital va texnik xizmat ko'rsatish xarajatlarini kamaytiradi, balki qishloq xo'jaligi, o'rmonlar yoki yovvoyi

tabiatni asfaltlash o'rniga, avtomagistrallardan yanada samarali foydalanish orqali ekologik tejash imkoniyatini yaratadi.

To'liq mustaqil holda, keksalar va nogiron odamlar o'zlari haydovchisiz harakatlana oladilar. Odamlar transport vositalarini avtonom ravishda to'xtash joyiga yuborishi va keyin ularni qaytarishi mumkin. Avtonom taksi safariga chiqish, shaxsiy transport vositasidan arzon va qulay bo'lib qolishi bilan shaxsiy avtomobilga egalik qilish asosan eskirishi mumkin. Shaxsiy transport vositalarining ko'pchiligidan atigi 5% vaqt davomida foydalaniladi. Transport vositalari flotini yaxshiroq foydalanish, transport vositalarini ishlab chiqarish va saqlashga bo'lgan talabni sezilarli darajada kamaytiradi. SI tizimlari tomonidan qo'llab-quvvatlanayotgan odamlar, safar uchun eng mos keladigan mavjud transport vositasini oddiygina buyurtma qilishlari mumkin. Avtomatlashtirilgan robot omborlari transport vositalarini yer yuzasida saqlashga qaraganda samaraliroq saqlashi mumkin. Juda zich shaharlarda shaxsiy avtomobillar egaligi allaqachon eskirgan bo'lib qolmoqda. Ushbu tendensiya avtonom transport vositalari bilan tezlashadi. Shahar hududlaridagi ko'plab asfaltlangan joylar turarjoylar yoki ekologik jihatdan yaxshilangan foydalanishlar, masalan, bog'lar, o'yin maydonchalari yoki shahar fermalari uchun ishlatilishi mumkin. Shaxsiy transport vositalari va jamoat transporti o'rtasidagi qat'iy farq yo'qolishi mumkin.

Bu taxminlar, hozircha, asosan ilmiy fantastikadir. To'liq avtonomiyaning ko'plab dastlabki va'dalari amalga oshmadi. Inson haydovchilari, avtonom transport vositalari, jamoat transporti, piyodalar, velosipedchilar va boshqalardan iborat aralash transport tizimiga o'tish qiyin.

To'liq avtomobil avtonomiyasiga ega bo'lmasdan, avtoturargoh, yo'lakni saqlash, yo'lakni o'zgartirish, moslashuvchan kruiz nazorati, favqulodda tormozlash va piyodalar va velosipedchilardan avtomatik qochish kabi ko'plab aqlli haydash funksiyalari endi haydovchiga yordam beradi va xavfsizlikni oshiradi. Avtomobillar va avtomobillardan tashqari turli xil transport vositalari yuk mashinalari, shu jumladan mikroavtomobillar, elektron velosipedlar elektr skuterlar va bir g'ildirakli velosipedlar endi «mikromobillik» sarlavhasi ostida mavjud bo'lib, ko'pincha yarimavtonomiya, marshrutlash va transport vositalarini almashish uchun sun'iy intellekt yaxshilanishlari mavjud. Aqlli ekipaj va transport vositalarini boshqarish va ba'zi avtonomiyalarga ega jamoat transporti ham yaxshilanmoqda. Ko'pchilik tomonidan eksperimental avtonom transport vositalari robot tanklari, harbiy yuk ko'chiruvchilar va avtomatlashtirilgan urushning oldingi belgisi sifatida ko'rilmogda. Avtomatlashtirilgan urushning ma'lum ma'noda ahamiyatli foydalari bo'lishi mumkin bo'lsa-da, ayni paytda juda real xavf-xatarlar mavjud. O'tmishda bular faqat ilmiy-fantastik tushlar edi. Endi avtomatlashtirilgan urush haqiqatga aylanganda, bu xavf-xatarlarga qarshi turish zarur. Sharkey [2008], Singer [2009a, b], Russell [2019] va Robillard [2021] avtonom qurol tizimlari va robot urushining xavflari va etikasi haqida muhokama qiladilar.

18.6. Barqarorlik

Hozirda insoniyat duch kelayotgan barqarorlik inqirozi ko'plab jihatlariga ega. Jumladan, iqlim favqulodda holati, global tengsizlik, biologik xilma xillikning yo'qolishi, suv va oziq-ovqat resurslarining tanqisligidir. **Barqarorlik** – bu tizimdagi jarayonning uzoq muddatda muvozanatini saqlash qobiliyati. **Ekologik barqarorlik** – bu ekotizimning kelajakda ekologik jarayonlar, funksiyalar, biologik xilma-xillik va mahsuldorlikni saqlab qolish qobiliyatidir. Ekotizimning **chidamliligi** – bu ekotizimning bezovtalikka bardosh bera olish qobiliyati bo'lib, u sifat jihatdan boshqa holatga tushib qolmasdan o'zini saqlab qoladi. Ijtimoiy tizimlarda chidamlilik odamlarning kelajakni oldindan ko'ra bilish va rejalashtirish qobiliyati orqali kuchayadi [Holling, 1973].

Barqaror rivojlanish hozirgi ehtiyojlarni kelajak avlodlarning o'z ehtiyojlari qondirish qobiliyatiga putur yetkazmasdan aniqlash va qondirish qobiliyatidir. Birlashgan Millatlar Tashkilotining Brundtland hisobotida «Bizning umumiy kelajagimiz» [Brundtland va boshq., 1987] barqaror rivojlanishga urg'u berilgan. Barqaror rivojlanish ekologik, ijtimoiy va iqtisodiy **cheklov**larni qondirishni talab qiladi [Rockstrom va boshq., 2009; Birlashgan Millatlar Tashkiloti, 2015b]. Atrof-muhit, ijtimoiy va iqtisodiy masalalar o'zaro chambarchas bog'liq.

Malthus [1798] o'zining «Aholi tamoyili haqida insho» asarida, asosan, aholi o'sishi va oziq-ovqat ta'minoti o'rtasidagi nomutanosiblikdan xavotir bildirgan edi. Aholi soni eksponensial ravishda o'sgan bo'lsa-da, oziq-ovqat resurslari cheklanar. U shunday yozgan:

«Aholi va yerning mahsuldorlik kuchlarining bu tabiiy nomutanosibligi va ularning ta'sirlarini doimiy ravishda tenglashtiruvchi tabiatimizning buyuk qonuni katta qiyinchilikni tashkil etadi».

Boshqacha aytganda, global sayyora tizimi o'sib borayotgan aholining iste'moli Yerning oziq-ovqat ishlab chiqarishi bilan cheklanishi haqidagi cheklovni qondirishi kerak. Bu bizning sayyora tizimimiz barqaror va chidamli bo'lishi uchun qondirilishi kerak bo'lgan ko'plab cheklovlardan biridir.

Umuman olganda, barqarorlik va hisoblash texnologiyasi, xususan, sun'iy intellekt o'rtasidagi munosabat qanday? Hisoblash texnologiyasi barqarorlik nuqtayi nazaridan ikkitomonlama qilichdir. Raqamli asr paydo bo'lgani sayin, hisoblash va aloqa tarmoqlari kuchining ajoyib o'sishi barqarorlik uchun sezilarli darajada foyda keltirdi. Hisoblash texnologiyasi jamiyat va iqtisodiyotda inqilob qilmoqda. 18.1-mavzuda muhokama qilinganidek, hisoblash texnologiyasida asosiy barqaror dinamik jarayon, ya'ni **dematerializatsiya**, atomlarni bitlar bilan almashtirish mavjud. Dematerializatsiya tabiatan ko'plab resurslarni tejaydi.

Boshqa tomondan, raqamli inqilobda ko'plab resurslar sarflanadi va isrof qilinadi. Kompyuterlar, qurilmalar va batareyalarni ishlab chiqarish uchun zarur bo'lgan materiallarni ishlab chiqarish uchun qazib olish atrof-muhitga jiddiy

ta'sir ko'rsatishi mumkin. Mahsulotning qisqa umrining oxirida har yili millionlab tonna elektron chiqindilar ishlab chiqariladi, bu esa ayniqsa Janubiy yarimsharda halokatli ekologik oqibatlariga olib keladi. Katta bulutli serverlar tomonidan ishlatiladigan quvvat yana bir asosiy iste'mol qilinadigan resursdir. Xususan, 8.5.5-mavzuda muhokama qilingan katta modellarning o'qitilishi ulkan hisoblash resurslarini talab qiladi. Sun'iy intellekt «ekstraksiya texnologiyasi» sifatida tavsiflanadi [Crawford, 2021]. Xuddi shunday, ayrim **kriptovalyuta** tanga, masalan, Bitkoin qazib olish va kriptovalyuta tranzaksiyalarini tasdiqlash ham katta resurslarni talab qiladi.

Ushbu tendensiyalarga qarshi kurashish uchun **yashil axborot texnologiyalari** harakati deb ataladigan harakat mavjud bo'lib, u kompyuterlar, serverlar va boshqa qurilmalarni minimal energiya sarfi va atrof-muhitga ta'sir ko'rsatadigan tarzda loyihalash, ishlab chiqarish, foydalanish, ta'mirlash va utilizatsiya qilishni maqsad qilgan.

Yangi fan sohasi – **hisoblash barqarorligi** paydo bo'lmoqda [Gomes va boshq., 2019]. U atrof-muhit, ijtimoiy va iqtisodiy ehtiyojlarni barqaror rivojlantirish uchun muvozanatlashtirishga qaratilgan sun'iy intellekt, kompyuter ilmlari, axborot ilmlari, operatsion tadqiqotlar, amaliy matematika va statistika texnikalarini qo'llaydi. Hisoblash barqarorligining ikkita asosiy yo'nalishi mavjud:

- Ekotizim resurslarini boshqarish va taqsimlash uchun **offlayn** qaror qabul qilishga mo'ljallangan hisoblash modellari va usullarini ishlab chiqish.

- **Onlayn** real vaqt rejimida ekotizimni kuzatish, boshqarish va nazorat qilish uchun to'g'ridan to'g'ri kiritilgan hisoblash modullarini ishlab chiqish.

Sun'iy intellekt ikkala yo'nalishda ham muhim rol o'ynaydi. «*Sayyora chegaralari: Insoniyat uchun xavfsiz ishlash makonini o'rganish*» asarida, Rockström va boshqalar [2009] Yerning biofizik jarayonlarining barqarorligini ta'minlash uchun to'qqizta muhim chegarani aniqladilar. Ushbu chegaralar sayyoramiz barqarorligini ta'minlash uchun qo'yilgan maqsad cheklovlaridir:

- iqlim o'zgarishi
- biologik xilma xillikning yo'qolish darajasi (quruqlik va dengizda)
- azot va fosfor sikllariga aralashish
- stratosferada ozon qatlamining yemirilishi
- okeanlarning kislotalanishi
- global chuchuk suvdan foydalanish
- yer foydalanishidagi o'zgarishlar
- kimyoviy ifloslanish
- atmosfera aerosol yuklanishi

Masalan, antropogen **iqlim o'zgarishiga** qo'yilgan cheklov atmosferadagi karbonat anhidrid (CO_2) konsentratsiyasini hajm bo'yicha millionda 350 qismdan (ppmv) kam bo'lishini talab qiladi. Sanoatgacha bo'lgan davrda bu qiymat 280 ppmv edi; 2009-yilda u 387 ppmv, 2023-yilda esa 412 ppmv ga yetdi. Biologik

xilma-xillikning yo'qolish darajasi yo'qolgan turlar soni (yiliga million turlar boshiga) bilan aniqlanadi. Uning chegaraviy qiymati 10 ga belgilangan, ammo 2023-yilda u 100 dan oshgan. **Cheklovlarni qondirish**, 4-bob va 6.4-mavzuda muhokama qilinganidek, hisoblash barqarorligining asosida yotadi.

2015-yilda Birlashgan Millatlar Tashkiloti «2030-yil Barqaror rivojlanish kun tartibi» [Birlashgan Millatlar Tashkiloti, 2015a]ni qabul qildi, bu 17 ta **Barqaror rivojlanish maqsadlarini (SDG)** belgilaydi [Birlashgan Millatlar Tashkiloti, 2015b]. SDGlar to'qqizta biofizik sayyoraviy chegara cheklovlarini qamrab oladi va ularni insonning ijtimoiy va iqtisodiy maqsadlarini, masalan, qashshoqlik, ochlik va tengsizlikni kamaytirish, shuningdek, sog'liqni saqlash, ta'lim va odil sudlovga kirishni yaxshilash kabi maqsadlarni qamrab olish uchun kengaytiradi. Sun'iy intellekt usullarining to'liq spektridan, shu jumladan chuqur o'rganish, mustahkamlash o'rganish, cheklovlarni qondirish, rejalashtirish, ko'rish, robototexnika va tilni tushunishdan foydalangan holda ko'plab tizimlar SDG'larga erishishda yordam berish uchun ishlab chiqilmoqda. Masalan, avvalgida ta'riflanganidek, Perrault va boshq. [2020] ko'p agentli texnikalar **Stackelberg xavfsizlik o'yinlariga** asoslangan holda jamoat salomatligi, xavfsizligi va ijtimoiy adolatni yaxshilashga qanday yordam berishi mumkinligini ko'rsatadi. Ko'p agentli usullar, shuningdek, barqaror rivojlanish muammolarining markazida bo'lgan **umumiy fojia** deb ataladigan narsani hal qilishga qaratilgan [Hardin, 1968]. Ostrom [1990] jamoaviy harakat institutlari umumiy mulkni boshqarish uchun rivojlanishi mumkinligini ko'rsatdi.

Sun'iy intellekt tadqiqotchilari va rivojlanish muhandislari global isish, qashshoqlik, oziq-ovqat ishlab chiqarish, qurol-yarog nazorati, sog'liqni saqlash, ta'lim, qarish kabi muammolarni hal qilish uchun zarur bo'lgan ko'nikmalarning ayrimlariga ega. Ular soha mutaxassislari bilan ishlashlari kerak bo'ladi va soha mutaxassislarini sun'iy intellekt yechimlari faqat yangi «ilon moyi» yemasligiga ishonitirishlari kerak bo'ladi. Oddiy misol sifatida, ushbu darslik va AISpace [Knoll va boshq., 2008] kabi sun'iy intellekt haqida o'rganish vositalariga ochiq kirish imkoniyati odamlarni sun'iy intellekt texnikalarini tushunishga va ularni o'z muammolariga tatbiq etishga imkon beradi, shuning o'rniga shaffof bo'lmagan, qora quti sifatidagi tijorat tizimlariga tayanmaslik mumkin bo'ladi. Sun'iy intellekt tizimlariga asoslangan o'yinlar va tanlovlar **RoboCupning** robot futboli bo'yicha muvaffaqiyati ko'rsatganidek, juda samarali o'rganish, o'qitish va tadqiqot muhiti bo'lishi mumkin [Visser va Burkhard, 2007]. Aqlli transport vositalari va aqlli yo'l harakatini boshqarishning ijobiy ekologik ta'sirlari 18.5-mavzuda muhokama qilingan edi. Bakker va boshq. [2020] dinamik atrof-muhitni boshqarish uchun raqamli texnologiya ilovalari haqida umumiy ma'lumot beradi.

Atrof-muhitga oid qarorlar qabul qilish ko'pincha murakkab tizimning bir qismi sifatida birgalikda ishlaydigan komponentlar to'plamini tanlashni talab qiladi. **Combinatorial auction (kombinatsion kimoshdi savdosi)** – bu agentlar paketlar,

ya'ni diskret elementlar kombinatsiyalariga takliflar beradigan auksiondir [Shoham va Leyton-Braun, 2008]. G'olibni aniqlash qiyin, chunki odatda afzalliklar qo'shilmaydi, lekin elementlar odatda o'zaro to'ldiruvchi yoki o'rmini bosuvchi bo'ladi. Spekrtni ajratish (televizor yoki mobil telefonlar uchun kompaniyalarga radiochastotalarni ajratish) [Leyton-Braun va boshq., 2017], logistika (tovarlarni tashishni rejalashtirish) va ta'minot zanjiri konfiguratsiyasiga [Sandholm, 2007] allaqachon qo'llanilgan kombinatorik auksionlar uglerod bozorlarini qo'llab-quvvatlash, energiya ta'minoti va talabini optimallashtirish va iqlim o'zgarishini yumshatish uchun qo'llanilishi mumkin. Binoenergiya sarfini yaxshilaydigan tarqatilgan datchiklar va aktuatorlardan foydalanadigan aqlli energiya boshqaruvchilari bo'yicha ko'plab ishlar mavjud.

18.7. Etika

Sun'iy intellektning ta'sirini hisobga olgan holda ko'plab **axloq** va **tarbiya** masalalari paydo bo'ladi. Axloq shaxsning to'g'ri va noto'g'ri haqidagi tushunchalariga nisbatan qo'llaniladigan ko'rsatmalardir. Etik tamoyillar esa jamoa, tashkilot yoki kasb darajasida qo'llaniladi. Axloq va etika, albatta, bir-biri bilan yaqin bog'liq: shaxsning shaxsiy axloqi turli manbalardan, shu jumladan, ular a'zo bo'lgan guruhlarining etik tamoyillaridan kelib chiqishi mumkin. Normativ etika kodekslari faylasuflar tomonidan fazilatga asoslangan, oqibatga asoslangan yoki deontologik sifatida tasniflanadi [Hursthouse va Pettigrove, 2018]:

- **Fazilat** etikasi odobli agentda mavjud bo'lgan qadriyatlar va xarakter xususiyatlariga urg'u beradi [Vallor, 2021].

- **Oqibatga asoslangan (yoki utilitar)** etika agent olishi mumkin bo'lgan harakatlarning natijalariga e'tibor qaratadi va har bir natijaning global foydasini o'lchaydi.

- **Deontologik (yoki Kantian)** etika kodekslari agent amal qilishi kerak bo'lgan qoidalar to'plamiga asoslanadi.

Sun'iy intellekt etikasiga e'tibor qaratishning sababi, qisman, sun'iy intellekt tizimlari to'g'ri harakat qilishi mumkinmi degan xavotirlar bilan bog'liq. Avtonom aqlli agentlarga tayanish savol tug'diradi: sun'iy intellekt tizimlariga **ishonishimiz** mumkinmi? Ular hozirgi shaklda qurilgan bo'lgani uchun to'liq ishonchli va ishonchli emas. Shunday ekan, ular to'g'ri ish qila oladimi? Ammo ishonch tizimning to'g'ri ish qilishiga bog'liq emas. Inson faqat tizim to'g'ri ish qiladiganiga ishonch hosil qilsa, uni ishonchli deb biladi. Mashhur filmlar va darsliklar bilan isbotlanganidek, bizning jamiyat ongimizda robotlar va boshqa SI tizimlari ishonchli emasligi haqida qo'rquv mavjud. Ular to'liq avtonom, erkin iroda, aql va onga ega bo'lishi mumkin. Ular bizga qarshi isyon ko'tarib, Frankenshteyn kabi yovuz kuchlarga aylanishi mumkin.

Ishonch masalalari axloqiy savollarni ko'taradi. Agar SI tizimlarini loyihalashtiruvchilar, amalga oshiruvchilar, ishlab chiquvchilar va foydalanuvchilar aniq axloqiy qoidalarga amal qilsa, bu tizimlar ishonchli bo'lish ehtimoli ko'proq. Bundan tashqari, agar ushbu tizimlar aslida aniq axloqiy qoidalarga rioya qilsa, ular ham ishonchli bo'lishi ehtimoli ko'proq. SI axloqi fani quyidagi savollarga javob berish bilan shug'ullanadi:

- SI olimlari nazariyalar, algoritmlar va vositalarni ishlab chiqishda axloqiy tamoyillarga rioya qilishlari kerakmi?

- SI tizimlarini loyihalashtiruvchilar va ishlab chiquvchilar uchun axloqiy faoliyatlar qanday?

- SI tizimlarini joriy qiluvchilar uchun qaysi ilovalar ko'rib chiqilmasligi kerak?

- Insonlar SI tizimlari bilan o'zaro munosabatda bo'lganida axloqiy tamoyillarga rioya qilishlari kerakmi?

- SI tizimlari o'zaro munosabatlarda insonlarga, boshqa agentlarga yoki dunyoga nisbatan axloqiy tamoyillarga amal qilishi kerakmi?

- SI tizimlarini o'qitish uchun qanday ma'lumotlardan foydalanish kerak?

- Ushbu muammolarning har biri uchun qaysi tashkilotlar axloqiy qoidalarni belgilaydi?

SI axloqi, rivojlanayotgan va kengayib borayotgan soha sifatida, ikkita o'zaro bog'liq, ammo farqli masalalarni ko'rib chiqadi:

A. Odamlar uchun SI axloqi: tadqiqotchilar, dizaynerlar, ishlab chiquvchilar, tizimlarni joriy etuvchilar va foydalanuvchilar uchun.

B. Tizimlar uchun SI axloqi: dasturiy agentlar va jismoniy robotlar uchun.

Har biri insonlar yoki tizimlar uchun axloqiy kodekslarni ishlab chiqish va tekshirish bilan shug'ullanadi.

Odamlar uchun SI etikasi haqida gap ketganda, ko'pchilik SI dizaynerlari va muhandislari uchun kuchli professional axloqiy kodekslarga ehtiyoj borligini his qiladi. Boshqalar bunga qo'shilmaydi. Yuridik, tibbiyot va kompyuter sohalari-ning barchasida amaliyotchilar rioya qilishi kutiladigan yoki talab qilinadigan aniq **deontologik** axloq kodekslari mavjud. Kompyuter sohasida, ACM Professional Ethics [2018], AAAI [2019] va IEEE [2020] o'z a'zolariga nisbatan axloqiy kodekslarni belgilagan.

SI tizimlarini loyihalash, qurish va joriy etishda qanday axloqiy ishlar bajarilishi kerakligi haqida bir qancha masalalar mavjud. Biz, insonlar, ular bilan o'zaro aloqada bo'lganimizda qanday axloqiy muammolar yuzaga keladi? Ularga qandaydir huquqlar berishimiz kerakmi? Inson huquqlari kodekslari mavjud; SI tizimlari huquqlari kodekslari ham bo'ladimi?

Faylasuflar axloqiy agentlar, axloqiy bemorlar va boshqa agentlarni ajratib ko'rsatadilar. Axloqiy agentlar to'g'ri va noto'g'ri narsani ajrata oladilar va o'z harakatlari uchun javobgar bo'lishlari mumkin. Axloqiy bemor – ax-

loqiy agent tomonidan axloqiy tamoyillar bilan munosabatda bo'lishi kerak bo'lgan agentdir. Masalan, odatdagi kattalar axloqiy agent va axloqiy bemor bo'lishi mumkin; chaqaloq axloqiy bemor, lekin axloqiy agent emas, an'anaviy avtomobil esa ikkalasi ham emas. SI agentlari hech qachon axloqiy agent bo'lishi (yoki bo'lishi kerakmi) degan savol ustida davom etayotgan munozaralar mavjud. Bundan tashqari, hozirgi SI tizimlari axloqiy bemorlar sifatida ko'rib chiqilishi kerakmi, insonlar tomonidan ehtiyotkorlik bilan axloqiy munosabatda bo'lish zarurmi? Ehtimol, yo'q, lekin kelajakdagi SI tizimlari, shu jumladan robotlar, axloqiy bemorlar sifatida ko'rilishi yoki shunday munosabatda bo'lishi mumkinmi yoki kerakmi? Ushbu masalalar Bryson [2011], Mackworth [2011], Bryson [2018], Gunkel [2018] va Nyholm [2021] tomonidan qisman yoritilgan. Ushbu masalalarning bir qismi IEEE Global Initiative on Ethics of Autonomous and Intelligent Systems [2019], OECD [2019] va UNESCO [2022] tomonidan ishlab chiqilgan ko'plab taklif etilgan SI axloqi kodekslarida qisman ko'rib chiqilgan.

SI tizimlarini loyihalash, qurish va joriy etishda qanday axloqiy ishlar bajarilishi kerakligi haqida bir qancha masalalar mavjud. Biz, insonlar, ular bilan o'zaro aloqada bo'lganimizda qanday axloqiy muammolar yuzaga keladi? Ularga qandaydir huquqlar berishimiz kerakmi? Inson huquqlari kodekslari mavjud. SI tizimlari huquqlari kodekslari ham bo'ladimi?

Falsafachilar **axloqiy agentlar, axloqiy bemorlar** va boshqa agentlar o'rtasida farq qiladi. Axloqiy agentlar to'g'ri noto'g'ridan ajrata oladigan va o'z harakatlari uchun javobgar bo'lgan agentlardir. Axloqiy bemor – bu axloqiy agent tomonidan axloqiy tamoyillar asosida muomala qilinishi kerak bo'lgan agent. Masalan, odatdagi kattalar axloqiy agent va axloqiy bemor bo'lishi mumkin; chaqaloq axloqiy bemor, lekin axloqiy agent emas, an'anaviy avtomobil esa ikkalasi ham emas. SI agentlari hech qachon axloqiy agent bo'lishi (yoki bo'lishi kerakmi) degan savol ustida davom etayotgan munozaralar mavjud. Bundan tashqari, hozirgi SI tizimlari axloqiy bemorlar sifatida ko'rib chiqilishi kerakmi, insonlar tomonidan ehtiyotkorlik bilan axloqiy munosabatda bo'lish kerakmi? Ehtimol, yo'q, lekin kelajakdagi SI tizimlari, shu jumladan robotlar, axloqiy bemorlar sifatida ko'rilishi yoki shunday munosabatda bo'lishi mumkinmi yoki kerakmi? Ushbu masalalar Bryson [2011], Mackworth [2011], Bryson [2018], Gunkel [2018] va Nyholm [2021] tomonidan qisman yoritilgan. Ushbu masalalarning bir qismi IEEE Global Initiative on Ethics of Autonomous and Intelligent Systems [2019], OECD [2019] va UNESCO [2022] tomonidan ishlab chiqilgan ko'plab taklif etilgan SI axloqi kodekslarida qisman ko'rib chiqilgan.

SI tizimlarining axloqi nuqtayi nazaridan, ular avtonomlik darajasini oshirib borayotganda qanday qarorlar qabul qilishi kerak? Bu masalada dastlabki fikr yuritganlardan biri bo'lgan ilmiy fantastika yozuvchisi Isaak Asimov [1950]

tomonidan ilgari surilgan, ehtimol sodda bo'lsa-da, qiziqarli takliflarni ko'rib chiqaylik. Asimovning **Robotlar uchun qonunlari** mantiqiy va qisqa ko'rinishli sababli, ular yaxshi boshlang'ich nuqta sifatida qabul qilinishi mumkin.

Asimovning asl uchta qonuni quyidagicha:

I. Robot insonga zarar yetkazmasligi yoki loqaydlik tufayli insonga zarar yetishiga yo'l qo'ymasligi kerak.

II. Robot insonga zarar yetkazishi mumkin bo'lgan buyruqlar birinchi qonunga zid kelmaganda, robot insonlar tomonidan berilgan buyruqlarga bo'ysunishi kerak.

III. Robot o'z mavjudligini himoya qilishi kerak, agar bunday himoya Birinchi yoki Ikkinchi qonunlarga zid kelmasa.

Asimov bu qonunlarni barcha robotlar uchun amal qilishini taklif qilgan va ishlab chiqaruvchilar bu qonunlarni ta'minlashlari kerak edi. Ushbu qonunlar robotlar uchun deontologik axloqiy kodeksni tashkil qiladi va qabul qilinishi mumkin bo'lgan robotik xatti-harakatlarga cheklovlar qo'yadi. Asimovning asarlaridagi syujetlar asosan insonlar robotga nima qilishni niyat qilgani bilan uning aslida nima qilgani o'rtasidagi yoki qonunlarning so'zma-so'z va mantiqiy talqinlari o'rtasidagi ziddiyatlardan kelib chiqadi, chunki ular hech qanday rasmiy tilga kiritilmagan. Asimovning fantastik hikoyalari qonunlarning yashirin ichki qarama-qarshiliklari va ularning oqibatlarini o'rganadi.

SI tizimlari uchun axloqiy me'yorlar bo'yicha davom etayotgan munozaralar mavjud, ammo bu munozaralar ko'pincha hali mavjud bo'lmagan texnik qobiliyatlarga, ya'ni SI tizimlarining xavfsizlik talablarini o'rnatish va tekshirish qobiliyatlariga asoslanadi.

Selinger va Leong [2021] tomonidan o'rganilgan **yuzni tanib olish** texnologiyasining axloqiy masalalari to'rtta shaklga bo'linadi, ularning har biri o'ziga xos xavf va foydalarga ega:

– **Yuzni aniqlash:** Rasmda yuzlarning joylashuvini topadi, bu telefonlarda keng tarqalgan bo'lib, yuzlar ustiga kvadrat chiziqlar qo'yadi.

– **Yuzni xarakterlash:** Yuzning individual xususiyatlarini aniqlaydi, masalan, taxminiy yosh, his-tuyg'ular (masalan, tabassum yoki qayg'u) va odamning nima-ga qarayotganini aniqlaydi.

– **Yuzni tasdiqlash:** Odamning bitta andozaga mos kelishini aniqlaydi, bu telefon foydalanuvchisini tasdiqlash va aeroport xavfsizligida qo'llaniladi.

– **Yuzni identifikatsiya qilish:** Rasmdagi har bir odamni yuzlar ma'lumotlar bazasidan aniqlaydi; bu do'stlar va oila a'zolarini aniqlash uchun umumiy foto kutubxona dasturiy ta'minotida qo'llaniladi.

Bu shakllarning har biri bilan bog'liq axloqiy masalalar yuzni tanib olish texnologiyasining to'g'ri qo'llanilishi va undan foydalanish bilan bog'liq xavflar va foydalarni o'z ichiga oladi.

Yuzni identifikatsiya qilish, odatda eng muammoli hisoblanadi, chunki bu jayroning ikkita jihati – mukammal bo'lganda va xatolar yuzaga kelganda muammolarni keltirib chiqaradi.

Agar yuzni identifikatsiya qilish mukammal va keng tarqalgan bo'lsa, odamlar doimiy ravishda kuzatuv ostida ekanliklarini bilishadi. Bu, ularni noqonuniy yoki tor ijtimoiy normalardan chetga chiqadigan harakatlarni qilishdan juda ehtiyot bo'lishga olib keladi. Odamlarning xatti-harakatlari o'z-o'zini senzura qilish bilan bog'liq, hatto ular hech qanday yomonlik qilish niyatida bo'lmasa ham. Hukmron tartibga nisbatan har qanday tanqid yoki normal xatti-harakatdan og'ish noqonuniy bo'lganda, noqonuniy faoliyatni oldini olish muammoli bo'ladi. Kuzatuv o'z-o'zini ifoda etishni, ijodkorlikni va o'sishni cheklaydigan «sovuq ta'sir»ga ega bo'ladi va g'oyalar bozorini mahrum qiladi.

Yuzlar ma'lumotlar bazasi berilganda, yuz identifikatsiyasi yuzni aniqlash va yuzni tekshirish kombinatsiyasiga aylanadi. iPhone yuz tekshiruvini 30 000 ballga asoslangan yuzning uch o'lehovli modelini yaratish uchun bir nechta sensorlardan foydalanadi. U **maxfiylikka asoslangan dizaynga** ega; ma'lumotlar telefonning o'zida saqlanadi, serverda emas. Bu xato ijobiylik darajasi 10 millionda 1 ni tashkil etadi, bu esa odatdagi foydalanishda xato ijobiylik bo'lishi ehtimoli past ekanligini anglatadi. Agar shu xato darajasi har bir insonning bazasida qo'llanilsa, o'rtaacha 800 odam Yerda ma'lum bir yuzga mos keladi. Faqat ko'rish texnikalari ancha yuqori xato darajasiga ega bo'lib, bu esa noto'g'ri aniqlash juda keng tarqalgan bo'lishini anglatadi. Odamlar yuzni tanib olishning har qanday shaklini normal hayotga kiritish, muammoli holatlarga osongina o'tishga imkon beruvchi sirli bir yo'l ekanligini ta'kidlaganlar. Masalan, agar biror jamoada jinoyatni aniqlash va oldini olish uchun kuzatuv kameralar mavjud bo'lsa, yuzni tanib olishni qo'shib, kameralar orqali qo'shimcha qiymat olish juda arzon bo'lishi mumkin.

5.10-mavzuda rasmiy apparat va dasturiy ta'minotni tekshirish bo'yicha ba'zi yutuqlar tasvirlangan. Joy [2000] yangi texnologiyalarning xavf-xatarlarini nazorat qila olmasligimizdan shunchalik tashvishda ediki, u robototexnika (va sun'iy intellekt), nanotexnologiya va genetik muhandislikning rivojlanishini to'xtatib turishni muvaffaqiyatsiz tarzda talab qildi.

Ehtimol, bu darslikda ishlab chiqilgan aqlli agent dizayn makoni va agent dizayn tamoyillari aqlli agentlar uchun ijtimoiy, axloqiy va huquqiy kodlarni ishlab chiqishda texnik jihatdan yaxshiroq asosni taqdim qilishi mumkin.

Biroq, skeptik qarama-qarshilikda Munn [2022] «SI axloqiy tamoyillari foydasiz, sun'iy intellekt texnologiyalarining irqiy, ijtimoiy va ekologik zararlarini hech qanday ma'noda kamaytirmaydi» deb ta'kidlaydi. Lundgrenning «Axloq qoidalarini himoya qilishda» asariga qarang [2023].

18.8. Boshqaruv va tartibga solish

SI texnologiyalarining keng tarqalgan ishlatilishi tomonidan keltirib chiqarilgan ayrim haqiqiy va potensial zararlarni hal qilish uchun axloqiy kodlar zarur, ammo yetarli emas. Hozirda SIning javobgarlik va sug'urta masalalari mavjud. SI masalalariga qaratilgan qonunlar butun dunyo bo'ylab kuchga kirmoqda. Ko'plab mamlakatlar hozirda SI qoidalarini va qonunlarini ishlab chiqmoqda. 50 mamlakatda milliy SI siyosatlarini va amaliyotlari bo'yicha o'tkazilgan so'rovni SI va raqamli siyosat markazi [2023] taqdim etgan. Robotlarni tartibga solish va robot qonunchiligi masalalari Calo [2014] va boshqalar [2016]da muhokama qilingan.

Zuboff [2019] **«kuzatuv kapitalizmi»** atamasini SI asosidagi foydalanuvchi izlash, ijtimoiy media va zamonaviy savdo o'rtasidagi bog'liqlikni xarakterlash uchun ishlatgan. Ushbu masalani faqat milliy darajada hal qilish qiyin, chunki bu global muammo hisoblanadi. 2016-yilda Yevropa Ittifoqi (YI) umumiy ma'lumotlarni himoya qilish va maxfiylik bo'yicha regulatsiya sifatida Umumiy ma'lumotlarni himoya qilish regulatsiyasini (General Data Protection Regulation (GDPR)) qabul qildi [Yevropa Komissiyasi, 2021]. GDPR Yilda maxfiylik rejimi doirasida inson huquqlari bilan bog'liq regulatsiya sifatida belgilangan. GDPR ko'plab YI dan tashqaridagi davlatlarda shunga o'xshash qonunlarga ta'sir ko'rsatdi. Yevropa bozorining o'lchovi sababli ko'plab korporatsiyalar GDPRni ma'lumotlarni himoya qilishda yagona yo'nalishni beruvchi sifatida kutib oldilar; ammo, GDPR kuzatuv kapitalizmini to'xtatishga olib kelmadi. YI, shuningdek, 2022-yilda **Raqamli xizmatlar qonunini** (Digital Services Act (DSA)) qabul qildi [Yevropa Komissiyasi, 2022c]. DSA raqamli xizmatni iste'molchilarni mazmun, tovar yoki boshqa xizmatlar bilan bog'laydigan har qanday vositachi sifatida ta'riflaydi, jumladan ijtimoiy median ham. Bu bolalar va boshqa foydalanuvchilarning huquqlarini himoya qilish va iste'molchilarni firibgarlik, noto'g'ri ma'lumot, misoginiya va saylov manipulyatsiyasidan himoya qilish uchun mo'ljallangan. Bu qonunlarni buzganlik uchun jiddiy jazolar mavjud.

OECD SI tamoyillari [OECD, 2019] Sun'iy intellekt siyosati va boshqaruvi uchun birinchi global asosni taqdim etdi. 2022-yilda Yevropa Ittifoqi Sun'iy intellekt to'g'risidagi qonun loyihasini (Artificial Intelligence Act (AI Act)) [Yevropa Komissiyasi, 2022b] muhokama qilayotgan edi, bu butun dunyo bo'ylab barcha sohalarda sun'iy intellektni tartibga solishni maqsad qilgan birinchi qonunchilikdir. Bu asosan sun'iy intellekt tizimlaridan kelib chiqadigan zararlarni bartaraf etishga qaratilgan, deb tushuntiradi Algorithm Watch [2022]. AI Actning asosiy tamoyili shundan iboratki, zarar qanchalik jiddiy bo'lsa, tizimlarga shunchalik ko'p cheklovlar qo'yiladi. Qabul qilinishi mumkin bo'lmagan xavflarga ega tizimlar taqiqlanadi. Yuqori xavfli tizimlar ma'lum cheklovlarga javob berishi kerak. Kam xavfli tizimlar tartibga solinmaydi. Masalan, agar hukumat tomonidan amalga oshirilsa, lekin xususiy sektor tomonidan amalga oshirilmasa, shaxslarning ishonchligini

baholaydigan ijtimoiy baholash taqiqlanadi. Profilaktik politsiya faoliyati taqiqlangan bo'lar edi. Huquqni muhofaza qilish organlari tomonidan jamoat joylarida **yuzni aniqlash** cheklanadi. Keyinchalik, Yevropa Ittifoqi sun'iy intellekt uchun javobgarlik to'g'risidagi direktivani (Yevropa komissiyasi, 2022d,a) qabul qildi. Agar u qabul qilinsa, odamlar va kompaniyalarga sun'iy intellekt tizimidan zarar ko'rgan taqdirda zararni qoplash uchun sudga murojaat qilishlariga imkon beradi. AQSH ilmiy-texnik siyosat idorasi [2022] avtomatlashtirilgan tizimlarni ishlab chiqish, ulardan foydalanish va joylashtirishga yordam beradigan beshta tamoyil va tegishli amaliyotlar to'plami bo'lgan «sun'iy intellekt huquqlari to'g'risidagi qonun loyihasi»ni ishlab chiqdi.

Boshqaruv davlat qonunchiligi va qoidalarini, **tashqi boshqaruvni** qamrab oladi, lekin u sun'iy intellekt mahsulotlari va xizmatlarini ishlab chiqadigan va amalga oshiradigan korporatsiyalar, davlat idoralari va boshqa subyektlarning **ichki boshqaruvini** ham anglatadi. Ushbu ishtirokchilarning aksariyati sun'iy intellektidan mas'uliyatli foydalanish tamoyillarini amalga oshirish uchun ichki boshqaruv choralari, shu jumladan axloq qoidalarini amalga oshiradilar [Amershi va boshq., 2019; Jahon iqtisodiy forumi, 2021]. Mas'uliyatli sun'iy intellekt tizimlarini yaratish uchun hal qilinishi kerak bo'lgan madaniy va tashkiliy muammolar Rakova va boshqalar tomonidan tasvirlangan [2021]. Ogohlantirish sifatida Grin [2022] shunday taklif qiladi: «Hukumatning algoritmik qarorlarini qabul qilish natijasida yuzaga kelishi mumkin bo'lgan zararlardan himoya qilish o'rniga, inson nazorati siyosati algoritmiklarni amalga oshirishda noto'g'ri xavfsizlik hissi yaratib, sotuvchilar va agentliklarga algoritmik zarar uchun javobgarlikdan qochish imkonini beradi». Professional standartlar, mahsulotlarning sifatiga mustaqil nazorat – bu sun'iy intellekt **xavfsizligini** ta'minlaydigan muhim vositalardir. Boshqaruvdan tashqari boshqa vositalardir (Falco va boshq., 2022).

Hukumatning tartibga solish doirasi qizg'in muhokama mavzusi bo'lib, li lobbichilik harakatlariga duch kelmoqda. Ko'p millatli korporatsiyalar **me'yorlarga** tayanib, ichki axloqiy kodekslarni joriy qilish zararlarning olinishiga uchun yetarli deb hisoblash orqali qo'shimcha tartibga solishdan qochishga harakat qilishayotganlikda ayblanmoqda. Bundan tashqari, qonunchilar va **tartibga soluvchilar** o'zlari tartibga solishi kerak bo'lgan korporatsiyalar ta'siri ostida qolib, o'ziga moslashib ketishi, ya'ni tartibga solishning qo'lga olinishi keng tarqalgan holatdir. Bu sun'iy intellektni boshqarish uchun jiddiy va muhim tashvish hisoblanadi.

18.9. Xulosa

Ushbu bobdan o'rganishingiz kerak bo'lgan asosiy fikrlar quyidagilar:

- Raqamli iqtisodiyot moddiy narsalardan ko'ra axborotga e'tibor qaratadi.
- **Atomlarni bitlarga aylantirish, dematerializatsiya** va sun'iy intellekt iqtisodiy operatsiyalarda ishqalanishni kamaytiradi va ularni tezlashtiradi.

– Sun’iy intellekt **vositachilik** funksiyalarini yo‘q qilish jarayonini ta’minlaydi. Bu raqamli iqtisodiyotda bandlik tuzilishini buzadi

– Katta ma’lumotlar to‘plamlarida o‘qitilgan mashinani o‘rganish tizimlari irqchilik, jinsiy kamsitish va inson qadr-qimmatini kamsituvchi boshqa munosabatlarni o‘z ichiga olishi mumkin.

– Qisqa muddatli va uzoq muddatli istiqbolda inson qadriyatlari va sun’iy intellekt tizimlari o‘rtasidagi muvofiqlik borasida tashvishlar mavjud.

– Sun’iy intellekt ilovalari iqtisodiyotga kirib kelmoqda, ko‘plab ko‘nikmalarni talabdan chiqarib tashlamoqda va boshqa ko‘nikmalarga bo‘lgan talabni oshirmoqda.

– Transport va barqarorlik sun’iy intellektning foydali qo‘llanilishi mumkin bo‘lgan ikki yo‘nalishdir.

– Sun’iy intellektning zararli ilovalarini cheklash uchun axloqiy kodekslar, qonunchilik, tartibga solish va sertifikatlash ishlab chiqilmoqda.

18.10. Mashqlar

18.1-mashq. Halokatli avtonom qurol tizimlaridan (Lethal autonomous weapons systems (LAWS)) foydalanishni global taqiqlash bo‘yicha takliflar berildi. Qonunlardan foydalanishni taqiqlash bo‘yicha harakatning hozirgi holatini tekshiring va tavsiflang. Bunday taqiqning foydasiga yoki qarshi qisqacha dalil keltiring.

18.2-mashq. Qariyalar, nogironlar yoki go‘daklar uchun robotlar va sheriklar ishlatilishini ko‘rib chiqing. Ushbu guruhlardan biri uchun sheriklarning zamonaviy holatini o‘rganing va qisqacha tasvirlab bering. Ularning foydalanishini qo‘llab-quvvatlash uchun uchta sabab va unga qarshi uchta sabab keltiring.

18.3-mashq. Inson huquqlari va hayvonlar huquqlari yaxshi tan olingan. Robot huquqlari uchun yoki unga qarshi dalil keltiring. Siz muhokama qilayotgan robot huquqlari haqida aniq bo‘ling.

18.4-mashq. Dunyo miqyosida chuqur o‘rganish tizimlari va Bitkoin qazib oluvchilarning energiya sarfini baholang va solishtiring. Natijalaringizni odatdagi shahar energiya sarfi bilan taqqoslang.

18.5-mashq. Normativ axloqiy tizimlardan biri (fazilatli axloq, konsekvensializm yoki deontologiya) avtonom avtomobil uchun dizayn ko‘rsatmalarida qanday qo‘llanilishi mumkinligini muhokama qiling.

18.6-mashq. Sun’iy intellekt dizaynerlari va muhandislari uchun kuchli kasbiy axloq kodekslarini talab qilish taklifini ko‘rib chiqing. Bunday kodekslarni talab qilishni qo‘llab-quvvatlash uchun uchta va unga qarshi uchta sabab keltiring. Umuman olganda, qaysi birini qo‘llab-quvvatlaysiz?

18.7-mashq. Hukumat tomonidan sun’iy intellektidan foydalanishga qo‘yilgan cheklavlarni tartibga solish, qonunchilik va ijro etish zaruriyatini ko‘rib chiqing. Cheklavlarni qo‘llab-quvvatlash uchun uchta va unga qarshi uchta sabab keltiring. Umuman olganda, qaysi birini qo‘llab-quvvatlaysiz?

19-bob. O'TMISHGA NAZAR VA ISTIQBOLLAR

Hisoblash zamonaviy makonlarimiz mash'ali hisoblanadi. 2056-yilga kelib, hisoblash inqilobi sanoat inqilobiga teng keladigan muhim o'zgarish sifatida tan olinadi. Hisoblashning rivojlanishi va uning tahliliy natijalarining keng tarqalishi ijtimoiy-iqtisodiy hayotga, fan va madaniyatga ulkan ta'sir ko'rsatadi.

– Erik Horvitz [2006]

Ushbu bob ilovalar uchun sun'iy intellektni joriy etishning zamonaviy holati bilan boshlanib, keyin agent dizayn maydoni nuqtayi nazaridan keng qamrovli manzarani ko'rib chiqadi va sun'iy intellektning kelajagi haqida mulohaza yuritadi. Agent dizayn maydonida yoritilgan ko'plab tasavvurlarni joylashtirganimizda, ular o'rtasidagi munosabatlar yanada aniqroq ko'zga tashlanadi. Bu bizga sun'iy intellekt tadqiqotlari sohasining chegaralari qayerda ekanligini ko'rish imkonini beradi va ushbu sohaning rivojlanish jarayonini tushunishga yordam beradi. Horvitz ta'kidlaganidek, hisoblash jarayonlari dunyoni o'zgartirmoqda; uning foydali va zararli ta'sirlari hammamizga ta'sir qiladi.

19.1. Sun'iy intellektni joriy etish

Sun'iy intellektni ilovada qo'llash jarayonida, ma'lumotlar real dunyodan olinadi va bashoratlarni baholash uchun sinov to'plamidan foydalanish o'rniga, amaliy harakatlar bevosita dunyoda amalga oshiriladi. 19.1-rasmda qaysi sun'iy intellekt texnologiyalaridan foydalanishni tanlashda yordam beradigan qaror daraxti keltirilgan.

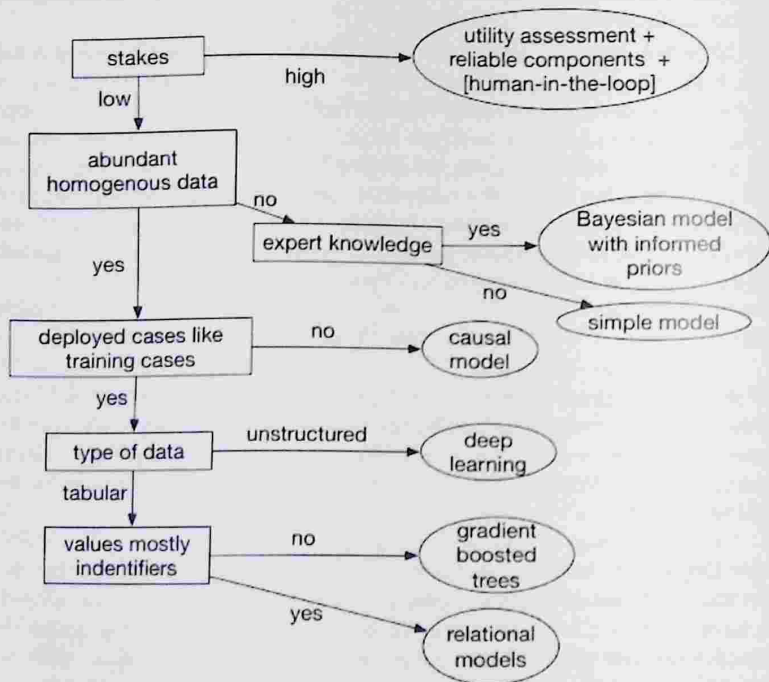
Ushbu daraxtdagi shartlar quyidagilardir:

- Eng yuqori darajada, **xavflarning** yuqori yoki past ekanligini tanlash kerak. Xatolar oqibatlar jiddiy bo'lmagan holatlarda, masalan, o'yinda, qanday videolarni tomosha qilishni tavsiya qilishda yoki oqsillarni qanday yig'ishni taklif qilishda, garovlar past bo'ladi, chunki har qanday dori-darmon to'liq sinovdan o'tkaziladi.

Agar tibbiy diagnostika agenti shifokorlarga unutgan bo'lishi mumkin bo'lgan imkoniyatlarni eslatib turayotgan bo'lsa, bu past xavf tug'dirishi mumkin, hatto agar unga tayanilsa, xavf yuqori bo'ladi. Past xavf holatlarida raqobatchilardan yaxshiroq bo'lish kifoya qiladi. Xavflar **yuqori** bo'lganda, odamlar yoki atrof-muhitga zarar yetkazilishi mumkin yoki har bir qaror uchun ko'p miqdorda resurslar sarflanadi. Takrorlanadigan qarorlar uchun odatda ko'plab mumkin bo'lgan

gan natijalar mavjud, ularning har biri past ehtimollikka ega, lekin ulardan birining yuzaga kelish ehtimoli juda yuqori bo'ladi. Masalan, o'ziyurar avtomobilda yo'l-da ko'plab noodatiy narsalar bo'lishi mumkin, ularning har biri ehtimoliy emas, lekin millionlab avtomobillar bilan, ulardan biriga kimdir duch kelishi ehtimoli juda yuqori. Agar xatoliklarning narxi yuqori bo'lsa, past xatolik darajasiga ega bo'lish dastur hali ham yuqori xavfli ekanligini anglatadi.

Onlayn tasvirlar, matnlar, yuqori tarmoqli kengligi bo'yicha ilmiy tajribalar yoki o'yin kabi ma'lumotlarni yaratish uchun simulyator kabi **bir turdagi ma'lumotlar** mavjud bo'lganda, katta hajmdagi ma'lumotlarni talab qiladigan mashinani o'rganish vositalaridan foydalanish mumkin.



19.1-rasm. Sun'iy intellekt texnologiyasini joriy etish tanlovi

Ba'zan ko'p miqdorda ma'lumot mavjud bo'ladi, lekin ular turli xildir. Masalan, hukumat ifloslanish to'g'risidagi ma'lumotlarni, aytaylik, ko'p joylarda bir nechta ifloslantiruvchi moddalarni har oy yoki muntazam ravishda sinovdan

o'tkazish, garchi har qanday ma'lum bir joy va ifloslantiruvchi modda uchun o'nlab yoki yuzlab ma'lumotlar mavjud bo'lsa ham, ular juda katta bo'ladi. Ushbu ma'lumotlar katta hajmga ega bo'lishi mumkin, garchi har bir joy va ifloslantiruvchi moddalar uchun faqat o'nlab yoki yuzlab ma'lumot nuqtalari mavjud bo'lsa ham. Yana bir misol, tibbiyotda ko'plab kam uchraydigan kasalliklar mavjud bo'lib, ularning faqat ozgina qayd etilgan holatlari bo'ladi. Garchi har bir kasallik kam uchrasa ham, juda ko'p kasalliklar mavjudligi sababli, bir jamiyatda ularning biriga chalingan odamni topish odatiy holdir. Juda ko'p kasalliklar mavjud bo'lganligi sababli, kasalliklarning aksariyat juftliklari uchun dunyoda hech kim ikkala kasallikka ham chalinmagan, garchi ko'p odamlar bir nechta kasallikka chalingan bo'lsa ham.

- Agar ko'p ma'lumot bo'lmasa, ko'pincha qo'llash mumkin bo'lgan **mutaxassis bilimlari** mavjud bo'ladi. Mutaxassislarning afzalligi shundaki, ular nafaqat misollardan o'rganadilar, balki maktabda yoki darsliklar o'qish orqali o'rgangan tasdiqlangan nazariyalarga tayanishlari va turli bilimlarni qo'llashlari mumkin.

- Ko'pgina mashinani o'rganish algoritmlari, **joriy etilgan holatlar o'quv holatlariga o'xshaydi**, deb taxmin qiladi: joriy etishda foydalaniladigan ma'lumotlar o'quv ma'lumotlari bilan bir xil taqsimotdan kelib chiqadi. Ushbu taxminni tasdiqlash kerak va u ko'pincha mos kelmaydi. O'yinlarga qarama-qarshi bo'lgan real dunyo sohalarida kelajak odatda o'tmishga o'xshamaydi, masalan, texnologiya yutuqlari yoki global isish tufayli. Siz kelajakning o'tmishga o'xshash bo'lishini istamasligingiz mumkin.

Olinishi mumkin bo'lgan bir nechta **ma'lumotlar turlari** mavjud. Ba'zi usullar matn, nutq, tasvirlar, videolar va oqsillar ketma-ketligi kabi **tuzilmagan** va seziladigan ma'lumotlarni qayta ishlash uchun ishlatiladi, bu yerda oson kirish mumkin bo'lgan funksiyalar mavjud emas. **Jadval ma'lumotlari** ba'zida obyektlarni shakllantirish uchun ishlatilishi mumkin bo'lgan toifalar, mantiqiy qiymatlar va raqamlar kabi qiymatlarni o'z ichiga oladi. Masalan, talabalar raqamlari, mahsulot identifikatorlari yoki tranzaksiya raqamlari, o'z-o'zidan ular nimaga tegishli ekanligi haqida hech qanday ma'lumotga ega emas.

Agar xavflar past bo'lsa, ko'p miqdorda bir xil turdagi ma'lumotlar mavjud bo'lsa va joriy etilgan holatlar o'quv holatlariga o'xshaydi deb kutilsa, sof mashinani o'rganish usullaridan foydalanish mumkin. Oldindan belgilangan xususiyatlar mavjud bo'lmagan tuzilmagan ma'lumotlar, masalan, tasvirlar, nutq, matn va oqsil ketma-ketliklari uchun chuqur o'rganish eng yaxshi tanlov bo'lib chiqdi. Jadval ma'lumotlarida qiymatlar xususiyatlarni qurish uchun ishlatilishi mumkin bo'lgan holatlarda, ushbu qiymatlar haqida farazlar birlashmalarining chiziqli funksiyalaridan foydalanadigan **gradiyent bilan mustahkamlangan daraxtlar** yaxshi ishlaydi. Identifikatorlar bilan jadval ma'lumotlari uchun munosabatli modellar mo'ljallangan.

Agar joriy etish o'rganishga o'xshaydi degan taxmin noto'g'ri bo'lsa, mumkin bo'lgan o'zgarishlar yoki yo'qolgan ma'lumot larni hisobga oladigan sababiy model qurilishi mumkin. Kuzatuv ma'lumotlari faqatgina harakatlarning ta'sirini bashorat qilish uchun yetarli emasligi, masalan, Simpson paradoksi kabi holatlarda isbotlangan. Sababiy modelning shartli ehtimolliklari 7- yoki 8-boblarda keltirilgan usullar yordamida o'rganilishi yoki butun model o'rganilishi mumkin, bunda sababiylik hisobga olinadi.

Agar ma'lumotlar ko'p bo'lmasa, lekin mutaxassislarning bilimlari mavjud bo'lsa, sababiy modelni ma'lumotli dastlabki ehtimolliklar bilan (masalan, Dirichlet taqsimotidan foydalanib) qurish, mutaxassislik va ma'lumotlarni birlashtirishning usuli bo'lib, bu ma'lumotlarning yo'qligi holatidan, ma'lumotlar juda ko'p bo'lgan holatga silliq o'tishni ta'minlaydi. Agar ma'lumotlar oz bo'lsa va hech qanday mutaxassislik bo'lmasa, odatda qaror daraxti yoki chiziqli model kabi oddiy model eng yaxshi natija beradi, bu yerda soddalik ma'lumotlar miqdoriga bog'liq bo'ladi.

Agar xavf yuqori bo'lsa, qaror ta'sir qilishi mumkin bo'lgan barcha manfaatdor tomonlarning foydasi asosida murakkab xarajat-foyda tahlili o'tkazish kerak. Harakatlar uchun katta resurslar talab etilganda yoki yomon natijalar yuzaga kelishi mumkin bo'lganda, qarorlar tushuntirilishi kerak. Xatolar yuzaga kelganda, tizimni tuzatish imkoniyatiga ega bo'lishi zarur. Yuqori xavf-xatarlar holatlari odatda har bir komponent yaxshi sinovdan o'tgan va ishonchli bo'lgan turli xil texnikalarning kombinatsiyasidan foydalanadi.

19.2. Agent dizayn makonini qayta ko'rib chiqish

Agent dizayn maydoni SI haqidagi bilimlar chegarasini tushunishning usulini beradi. Darslikda keltirilgan tasvirlarning ushbu maydonda qanday joylashishini ko'rish o'rganish uchun foydalidir.

19.2-rasm murakkablik o'lchovlarini ko'rib chiqadi va darslikda yoritilgan ba'zi agent modellarini har bir o'lchov bo'yicha qiymatlar nuqtayi nazaridan tasniflaydi. Ushbu agent modellar turli o'lchovlarda turli qiymatlarga ega bo'lgani uchun tanlangan.

Agent modellari

Quyida 19.2-rasmning ustunlarini tashkil etuvchi agent modellari tasvirlangan.

– **Iyerarxik boshqaruv** bir nechta abstraksiya darajalarida mulohaza yuritishni anglatadi. Taqdim etilganidek, bu rejalashtirishsiz bo'lgan va u maqsadlar yoki foydani hisobga olmagan.

– **Holat va makon qidiruvi**, 3-bobda taqdim etilganidek, cheksiz ufqqa imkon beradi, ammo boshqa barcha o'lchovlar bo'yicha eng oddiy qiymatni beradi. Deterministik rejalashtirish (6-bob), regressiya rejalashtirish, oldinga rejalashtirish yoki CSP rejalashtirish orqali, holat fazosini qidirishni xususiyatlar nuqtayi nazaridan mantiq yuritishga kengaytiradi.

– Qarorlar tarmoqlari ishonch tarmoqlarini kengaytirib, qaror va foyda tugunlarini qoʻshadi va xususiyatlarni, stoxastik taʼsirlarni, qisman kuzatiladigan holatlarni va murakkab afzalliklarni foyda nuqtayi nazaridan ifoda etishi mumkin. Biroq, bu tarmoqlar faqat cheklangan bosqichlarga ega rejalashtirish ufqini modellaydi.

Markov qaror jarayonlari (Markov Decision Processes (MDP)) stoxastik harakatlar va murakkab afzalliklar bilan cheksiz va cheklangan bosqichli muam-molarni hal qilishga ruxsat beradi; ammo ular holatga asoslangan ifodalardir, bu esa holatning toʻliq kuzatilishini nazarda tutadi. Dinamik qaror tarmoqlari (dina-mik DN) MDPlarni kengaytirib, holatlarning xususiyatlarga asoslangan ifodasini taʼminlaydi. Qisman kuzatiladigan MDPlar (POMDPlar) qisman kuzatiladigan holatlarni ruxsat beradi, lekin ularni yechish ancha qiyin.

	Hier. Control	Search	Det. Planning	Decision Net	MDP	POMDP	Dynamic DN	Extensive game	Q-Learning	Deep RL	Stochastic PI
Modularity											
flat	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
modular	✓	✓									
hierarchical	✓										
Planning Horizon											
non-planning	✓										
finite			✓	✓				✓			✓
indefinite			✓	✓		✓	✓	✓	✓	✓	
infinite						✓	✓		✓		
Representation											
states	✓	✓			✓				✓		✓
features			✓	✓	✓	✓	✓			✓	
relational											
Computational Limits											
perfect	✓	✓	✓	✓				✓			
bounded						✓	✓	✓	✓	✓	✓
Learning											
given	✓	✓	✓	✓	✓	✓	✓	✓			
learned									✓	✓	✓
Sensing Uncertainty											
fully obs.	✓	✓	✓		✓	✓	✓		✓	✓	✓
partial obs.				✓			✓	✓			
Effect Uncertainty											
deterministic	✓	✓	✓								
stochastic				✓	✓	✓	✓	✓	✓	✓	✓
Preference											
goals	✓	✓									
utility				✓	✓	✓	✓	✓	✓	✓	✓
Number of Agents											
single	✓	✓	✓	✓	✓	✓	✓				
adversary									✓	✓	✓
multiple	✓							✓			✓
Interactivity											
offline		✓	✓	✓	✓	✓	✓	✓		✓	✓
online	✓										

19.2-rasm. Murakkablik oʻlchovlari boʻyicha baholangan baʼzi agent modellari

– **O'yin daraxti qidiruvi** o'yinning kengaytirilgan shakli uchun davlatlar fazosini qidirishni bir nechta agentlar va foydalilikni o'z ichiga olgan holda kengaytiradi. U ma'lumot to'plamlari yordamida qisman kuzatiladigan domenlarni boshqarishi mumkin.

– **Q-ta'lim (Q-learning)** MDPlarni kengaytirib, onlayn o'rganishni ta'minlaydi, lekin faqat holatlar bilan ishlaydi. **Chuqur mustahkamlashni o'rganish (Deep RL)** va mustahkamlashni o'rganishdagi funktsiya yaqinlashtirishdan foydalanган boshqa usullar xususiyatlar bilan mustahkamlash o'rganishini amalga oshiradi. Ular bitta agentlar yoki qarama-qarshi agentlar bilan ishlaydi, ammo har qanday ko'p agentli sohalar uchun mos kelmaydi.

– **Stoxastik SI**, stoxastik siyosat iteratsiyasi, bir nechta agentlar bilan o'rganishga imkon beradi, lekin koordinatsiya qilish uchun bir xil agentlar bilan bir xil o'yinni takroriy o'ynashni talab qiladi.

Qayta ko'rilgan o'lchovlar

19.2-rasmda taqdim etilgan hech qanday rejalashtirish vakillari **iyerarxik nazoratni** boshqarmaydi. Iyerarxik nazoratda, past darajadagi nazoratlar yuqori darajadagi mulohazalardan tezroq harakat qiladi. Chuqur mustahkamlashni o'rganish holatining taqdimotida iyerarxiyalardan foydalanadi, lekin faqat bitta abstraksiya darajasida rejalashtiradi. Iyerarxik rejalashtirish va iyerarxik mustahkamlash o'rganish bu darslikda taqdim etilmagan, garchi ko'p tadqiqotlar mavjud. Iyerarxik fikrlash birlik sifatida ishlashi shart emas; quyi darajalarda yuqori darajadan farqli usullardan foydalanish mumkin. Odamlar past darajadagi idrok va reaksiyalarga qaraganda yuqori darajada o'ylash uchun mutlaqo boshqacha tizimlarga egaligi haqida dalillar mavjud.

Ba'zi vakillar (masalan, qaror tarmoqlari) har bir qarorni alohida modellash-tiradi va faqat cheklangan qarorlar ketma-ketligi uchun qo'llaniladi. Ba'zilar cheksiz qarorlar olishga imkon beradi, lekin shunda siyosatlar odatda stasionar (vaqtdan bog'liq emas, agar vaqt holat maydoni bir qismi bo'lmasa). (Chegirmali yoki o'rtacha) mukofotlarga asoslangan rejalashtirish tizimlari cheksiz bosqichli rejalashtirishni boshqarishi mumkin, bu yerda agentlar doimiy ravishda mukofotlar to'playdilar. Maqsadga yo'naltirilgan tizimlar doimiy ravishda davom etishi ma'qul emas.

Barcha tasvirlar holatlarni yagona xususiyatga ega bo'lgan degenerativ holat sifatida ko'rib chiqishi mumkin. Xususiyatlar nuqtayi nazaridan fikrlash ko'plab tasavvurlar uchun asosiy dizayn tanlovi bo'lib, muhandislik xususiyatlari yoki o'rganilgan xususiyatlardir. Xususiyatlar bo'yicha mulohaza yuritish holatlar bo'yicha mulohaza yuritishdan ancha samaraliroq bo'lishi mumkin, chunki holatlar soni xususiyatlar sonidan eksponensial ravishda ko'p. 19.2-rasmdagi tasvirlarning hech biri relyatsion modellarga ruxsat bermaydi, garchi ko'plab algoritmlarni relyatsion qilish mumkin.

Cheklangan ratsionallik ilovalar uchun ishlatiladigan ko'plab taxminiy usullarning asosini tashkil etadi; ammo, agent darhol harakat qilishi yoki ko'proq

o'ylashi kerakligi haqida mulohaza yuritadigan fikrlash va aktyorlik o'rtasidagi aniq kelishuvni amalga oshirish hali ham nisbatan kam uchraydi. 19.2-rasmda faqat uchta o'rganish algoritmi ko'rsatilgan, ammo boshqalar uchun modellarni o'rganish mumkin, masalan, shartli ehtimolliklarni o'rganish yoki ehtimollik modellarining tuzilishi, masalan, modelga asoslangan mustahkamlashni o'rganishda o'tish va MDP uchun mukofot olish ehtimoli o'rganiladi. Agent yaratishda eng katta qiyinchilik tug'diradigan jihat noaniqlikni sezishdir. Qisman kuzatish sharoitida dunyo ko'plab turli holatlarda bo'lishi mumkin. Agentning harakati natijasi dunyoning haqiqiy holatiga bog'liq bo'ladi. Agent faqatgina o'zining o'tmishdagi idroklar va harakatlar tarixiga ega bo'ladi. Agentning tarixidan uning harakatlari-gacha bo'lgan funksiyani ifodalashning turli usullari mavjud. Noaniqlik va cheksiz ufq muammolarini hisobga olgan holda rejalashtirishni kengaytirish usullari POMDPs kontekstida muhokama qilinadi. Sensorli ma'lumotlarni barcha shakllarda qanday boshqarish kerakligi haqidagi savol zamonaviy sun'iy intellekt tadqiqotlarining eng faol yo'nalishlaridan biridir.

Stoxastik harakatlardan foydalanishi mumkin bo'lgan modellar deterministik harakatlarni ham boshqarishi mumkin (chunki deterministik stoxastiklarning alohida holatidir). MDP va mustahkamlashni o'rganish algoritmlari kabi ba'zilar deterministik sohalarda yaxshi ishlaydi. Stoxastik harakatlardan foydalanishi mumkin bo'lgan modellar deterministik harakatlarni ham boshqarishi mumkin (chunki deterministik stoxastikning alohida holati). Ulardan ba'zilar, masalan, MDP va mustahkamlashni o'rganish algoritmlari, deterministik sohalarda yaxshi ishlaydi.

Afzalliklar maqsadlar yoki foydalilik nuqtayi nazaridan beriladi; 12.3-mavzuda juda yumshoq taxminlar bo'yicha murakkab imtiyozlar foydalilik nuqtayi nazaridan ifodalanishi mumkinligi isbotlangan. Murakkab kardinal imtiyozlarni bajara oladigan modellar, shuningdek, maqsadga erishish uchun mukofot berish orqali maqsadlarni boshqarishi mumkin. Maqsadga erishish uchun eng qisqa yo'lni afzal ko'rish maqsadga erishishga olib kelmaydigan harakatlar uchun salbiy mukofot yoki diskontlash orqali amalga oshirilishi mumkin. Umuman olganda, kommunal xizmatlar maqsadlarga qaraganda ko'proq ifodali.

Bir nechta agentlar bilan ishlash bitta agent bilan ishlashni rejalashtirishdan ko'ra ancha qiyin. Agentning yagona raqibga duch kelishi odatiy holatlarga qaraganda osonroq. Bir nechta agentlar hamkorlik qilishi yoki raqobatlashishi mumkin, yoki ko'pincha ular o'rtasida biron-bir joyda bo'lishi mumkin, bu yerda ular ba'zi jihatlarni taqlid qilishlari va boshqalarida hamkorlik qilishlari mumkin. Agentlar o'rtasidagi aloqa jamiyatda hamkorlikka erishishning standart usuli hisoblanadi. O'z manfaatlariga qiziqqan subyektlar o'rtasida hamkorlikka erishishning yana bir usuli bu mexanizmlarni ishlab chiqishdir, jamiyatning ishlashiga, shu jumladan pul mablag'lari va qonunchilikka imkon beradi. Ushbu darslikda biz nima qilish mumkinligi haqida faqat yuzaki to'xtalib o'tdik.

Interaktivlik bitta o'lov bo'lsa-da, haqiqiy agentlar tezkor onlayn qarorlar qabul qilishlari, shuningdek, uzoq muddatli qarorlar qabul qilishlari kerak. Agentlar bir nechta vaqt o'lovlarini hisobga olishlari kerak va bir soniya ichida qabul qilinishi kerak bo'lgan qarorga nisbatan oflayn bo'lib ko'rinadigan narsa kunlar miqyosida onlayn qaror sifatida ko'rilishi mumkin. Cheksiz oflayn hisoblash agentning hech qachon harakat qilmaslik xavfini tug'diradi.

Ushbu darslikda sun'iy intellektni loyihalash sohasining kichik bir qismi batafsil bayon etilgan. Zamonaviy tadqiqotlar ushbu darslikda tasvirlanganidan tashqariga chiqadi.

Sun'iy intellektning barcha sohalarida faol tadqiqotlar olib borilmoqda. Rejalashtirish, o'rganish, idrok etish, tabiiy tilni tushunish, robototexnika va sun'iy intellektning boshqa sohalarida ta'sirchan yutuqlar bo'lgan va shunday bo'lib qolmoqda. Ushbu ishlarining aksariyati turli o'lovlar va ularning o'zaro ta'siriga bag'ishlangan bir vaqtning o'zida barcha jihatlarini va ko'plab vazifalarni (masalan, umumiy sun'iy intellekt doirasida) ko'rib chiqishga qiziqish ortib bormoqda, ammo hamma narsani yaxshi bajarish qiyin.

Sning kichik sohalariga bo'linishi ajablanarli emas. Loyihalash maydoni bir vaqtning o'zida to'liq o'rganish uchun juda katta. Agar tadqiqotchi, masalan, obyektlarning mavjudligi haqida munosabatli domenlar va mantiqiy mulohazalarni hal qilishga qaror qilsa, sezgilar noaniqligini qo'shish qiyin bo'ladi. Agar tadqiqotchi cheksiz ufqlar bilan o'rganishdan boshlasa, iyerarxik mulohazalarni qo'shish qiyin, cheksiz ufqlar va munosabatlar bilan birga iyerarxiyalarni o'rganishni qo'shish esa yanada murakkabroq. Sun'iy intellekt mutaxassislari sifatida, biz hali ham cheksiz bosqichlarda, qisman kuzatiladigan domenlarda, shaxslar va munosabatlardan iborat bo'lgan, bir necha agentlar mustaqil ravishda harakat qiladigan agentni qanday qilib ratsional tarzda qurishni bilmaymiz. Insonlar buni amalga oshirishi mumkin, ehtimol, iyerarxik va taxminiy fikrlash orqali. Biz hali inson darajasidagi intellektual sun'iy agentni yarata olmasak-da, buni rivojlantirish uchun asosiy bloklarga ega bo'lishimiz mumkin. Asosiy muammo real dunyo murakkabligini boshqarishdir. Ammo hozirgacha inson darajasidagi yoki undan yuqori darajadagi ishlashga qodir kompyuterlashtirilgan agentlarni qurishga hech qanday ichki to'siqlar mavjud emasga o'xshaydi.

19.3. Kelajakka nazar

Dengiz floti bugun o'zini anglay oladigan, yuradigan, gapiradigan, ko'radigan, yozadigan, o'zini ko'paytira oladigan elektron kompyuterning embrionini taqdim etdi. Xizmat ... birinchi Perseptron deb ataladigan fikrlovchi mashinasini yaratishini aytdi, bu mashina o'qish va yozishni biladi. Uni taxminan bir yil ichida 100 000 dollar sarf-xarajat bilan tugatish rejalashtirilmoqda.

— New York Times [1958]

Kelajakni bashorat qilish har doim xavflidir. Bruks [2018] yangi, asosli bashoratlarni taqdim etadi va har yili taraqqiyot ma'lumotlarini yangilaydi. So'nggi o'n yil ichida katta ma'lumotlar to'plamlariga va ulkan hisoblash quvvatiga tayanadigan dasturlar juda ko'paydi. Bu o'sish asosan onlayn ma'lumotlar to'plamlari va vektor qayta ishlash birliklari (jumladan GPUlar)dan foydalanish orqali, ayniqsa, katta hisoblash markazlariga ega bo'lish imkoniyatiga ega yirik korporatsiyalar tomonidan ta'minlandi. Biroq, sun'iy intellekt ilmidagi ko'proq o'lichamlarni integratsiya qilish jarayoni ancha sekinroq sodir bo'lmoqda. Texnologiyaga kelsak, 19.1-rasmda ko'rsatilgan texnikaning hozirgi holatini hisobga olgan holda xavfsiz ko'rinadigan bir nechta bashoratlar mavjud. Vizual ma'lumotlar, matn, video va katta ma'lumotlar ilmi kabi bir xil ma'lumotlar mavjud bo'lganda, past stavkali qarorlar qabul qilish, ehtimol ko'proq ma'lumot va ko'proq ishlov berish quvvatini talab qiladi va hozirda ishlab chiqilayotgan takomillashtirilgan algoritmlar bilan birgalikda bunday holatlarda yaxshiroq bashorat qilish imkonini beradi. Neyron tarmoqlar kabi cheklangan hududlardagi har qanday funksiyani taxmin qila oladigan universal funksiya yaqinlashtiruvchilari juda ko'p ma'lumotlar va ishlov berish quvvatiga ega bo'lganda juda samarali ekanligi isbotlangan.

Generativ SI deb ataladigan dorilar va boshqa kimyoviy moddalar uchun rasmlar, videolar, matnlar, kodlar va yangi ishlanmalarni yaratish tobora murakkablashadi. Muhim qarorlar qabul qilish uchun foydalanilganda, agar ular to'g'ri bo'lish ehtimoli yuqori bo'lmasa, bashoratlarni muammoni hal qilish uchun ishonchli komponentlar sifatida foydalanish mumkinligiga ishonch hosil qilish uchun tanqidiy baholash kerak bo'ladi.

Muhim qarorlarni qabul qilish uchun foydalanilganda, agar ular aniq aniqlik bilan yaratilmasa, bashoratlarni muammoni hal qilish uchun ishonchli komponentlar sifatida foydalanish mumkinligiga ishonch hosil qilish uchun tanqidiy baholash kerak bo'ladi.

Bashorat texnologiyasidagi yaxshilanishlar foydali natijalar beradi, chunki yaxshiroq bashoratlar yaxshiroq qarorlar qabul qilishga olib keladi. Biroq, qarorlarga kiritilgan qadriyatlar insonlar farovonligiga mos kelmaganda, bu texnologiyalar odamlarga zarar yetkazishi mumkin. Masalan, generativ sun'iy intellektidan matn, tasvirlar, video va musiqa yaratish uchun foydalanish keskin o'sishi kutilmoqda. Ushbu texnikalar ish jarayonida san'at yaratishda, inson ijodkorligini oshirishda yoki chalgitish maqsadida ishlab chiqarilgan «deep fake»larni yaratishda qo'llanilishi mumkin. Ushbu soxta mahsulotlarni yaratish va aniqlash bo'yicha raqobat mavjud, ammo shuni yodda tutish kerakki, raqobatlashadigan tarmoqlar (shu jumladan GANlar) bu soxtalarni aniqlashga qaratilgan harakatlarga qarshi modellarning qurilishini ochiqdan ochiq ta'minlaydi.

O'rta muddatli rivojlanishlardan biri, amalda foydalanilayotgan dunyo o'quv ma'lumotlari kelgan dunyodan farq qilganda yoki transfer o'rganishda, bir domenning ma'lumotlarini boshqa bir domen uchun ishlatganda yuz beradigan ho-

latlar bo'ladi. Faqatgina kuzatuv ma'lumotlari harakatlarning ta'sirini bashorat qilish uchun yetarli emas; sababiylik va ekspert bilimlari hisobga olinishi kerak. Biz kichik sohalar o'rtasidagi ko'proq o'zaro ta'sirlarni kutamiz.

Ko'p hollarda yetarli ma'lumotlar mavjud emas. Masalan, SNOMED CT tibbiy ontologiyasi kasalliklar va insonlarda uchraydigan boshqa asosiy sabablarga tegishli taxminan 100,000 atamani o'z ichiga oladi. Kasalliklar katta ehtimol bilan shuni anglatadiki, ko'pchilik kasalliklar uchun juda kam odam kasal bo'ladi. Dunyoda deyarli hech kimda bir vaqtning o'zida ikkita juft kasallik mavjud emas. Ushbu kasalliklarning o'zaro ta'sirini faqat ma'lumotlardan o'rganish imkoni yo'q, chunki barcha o'zaro ta'sirlarni qamrab oladigan yetarli odamlar yo'q. Bunday holatlar uchun yanada murakkab modellar ko'rib chiqilishi kerak va mutaxassis bilimlarini e'tiborsiz qoldirish mumkin emas. Ma'lumotlar va oldindan kutishlarga mos keladigan farazlarni yaratish uchun boy metadata va mutaxassis bilimlariga ega bo'lgan ma'lumotlarning ko'proq integratsiyasini ko'rishni kutmoqdamiz.

Dunyoda qaror qabul qilish uchun agent passiv kuzatuvdan, masalan, matn yoki videodan saboq ololmaydi. Matn faqat kimgadir qiziq tuyulgan narsani o'z ichiga oladi va hamma biladigan oddiy tafsilotlarni o'z ichiga olmaydi. Videoda agentning xatti-harakatlari ko'rsatilmagan va agar mavjud bo'lsa, u boshqa holatlarda nima qilishini ko'rsatmaydi, bu haqiqatga ziddir. Sun'iy domenlardan tashqari, agent hech qachon ikki marta bir xil holatda bo'lmaydi. **Mujassamlangan** agent atrof-muhit bilan o'zaro ta'sir qiladi. Haqiqiy dunyoda mujassamlangan agent **ongli fikrga** muhtoj [Brahman va Levesk, 2022b]. Bunga siz ilgari hech qachon duch kelmagan kutilmagan narsalarga javob berishga tayyor bo'lish kiradi.

Katta ahamiyatga ega bo'lgan qarorlar uchun ta'sir ko'rsatuvchi manfaatdor tomonlardan **imtiyozlarni olish** ko'rib chiqilishi kerak. Umuman olganda, imtiyozlarni ma'lumotlardan o'rganish orqali olish mumkin emas. Biror kishining xatti-harakatlariga asoslanib, ularning xohish-istaklarini taxmin qilish mumkin bo'lsa-da, **teskari mustahkamlashni o'rganish** deb ataladigan narsada, ular turli vaziyatlarda nima qilganini bilish qiyin yoki ular o'z harakatlaridan pushaymon bo'lishi, ularning afzalliklari o'zgarishi mumkin. Turli manfaatdor tomonlar juda boshqacha imtiyozlarga ega bo'lishi mumkin va ular qandaydir tarzda bir nechta manfaatdor tomonlarga ta'sir ko'rsatadigan agent uchun adolatli va shaffof tarzda birlashtirilishi kerak.

Ba'zilar kompyuterlar odamlarning intellektidan oshib ketib, bir o'ziga xoslikka erishishini taklif qilishgan, bu holda odamlar kerak bo'lmasligi mumkin. Kompyuterlar odamlar bilan solishtirganda qachon ko'proq aqlli bo'lishi haqida ko'p taxminlar mavjud. Buni aniqlash qiyin, chunki «odamlar» yoki «kompyuterlar» nima degani aniq emas. Odamlar tasodifiy madaniyat haqidagi savollarga javob beradigan tasodifiy odamni, o'z madaniyati haqida savollarga javob beradigan o'qimishli odamni, dunyo ekspertini yoki Internet va boshqalarga ulanish imkoniyatiga ega bo'lgan odamni anglatishi mumkin. Kompyuter esa mustaqil ishlaydigan telefon yoki noutbukni, Internetga ulangan telefon yoki noutbukni, insonlar ishtirokisiz butun In-

ternetni yoki insonlar ishtirokida ishlaydigan Internetni anglatishi mumkin. Ushbu tushunchalarning eng umumiy darajasida ularning o'rtasidagi farqni aniqlash qiyin.

Ba'zilar sun'iy intellekt mashinalar hukmronlik qiladigan distopik kelajakka olib kelishini bashorat qilishgan, ammo boshqalar buni nirvana – ideal yoki xayoliy joy sifatida ko'radi, bu yerda yaxshi qarorlar qabul qilinadi va odamlar g'amxo'rlik qilinadi. Qaysi biri paydo bo'lishi sun'iy intellektni yaratadigan va uni tartibga soladigan odamlarga bog'liq bo'ladi.

Turing [1950] o'zining asosiy maqolasini «Biz faqat qisqa masofani ko'rishimiz mumkin, lekin u yerda amalga oshirilishi kerak bo'lgan ko'p narsalarni ko'rishimiz zarur» degan so'zlar bilan yakunlagan edi. Garchi o'sha vaqtdan beri sezilarli yutuqlar bo'lgan bo'lsa ham, hozirda ham bu haqiqatdir.

19.4. Manbalar va qo'shimcha adabiyotlar

Ushbu o'lchovlarning boshqa ba'zi kombinatsiyalari quyidagilarni o'z ichiga oladi. Iyerarxik rejalashtirish Nau [2007] tomonidan muhokama qilingan. Iyerarxik mustahkamlash o'rganishi Dietterich [2000b] tomonidan yoritilgan. Ko'p agentli mustahkamlash o'rganishi Stone [2007] tomonidan ko'rib chiqilgan. Teskari mustahkamlash o'rganishi Ng va Russell [2000] tomonidan kiritilgan va Russell [2019] tomonidan muhokama qilingan. Dzeroski va boshqalar [2001] munosabatli mustahkamlash o'rganishini ko'rib chiqadi. Khardon va Sanner [2021] munosabatli qaror-nazariy rejalashtirishni muhokama qiladilar.

SI kelajagi uchun muammolar Russell [2019], Marcus va Davis [2019], Littman va boshqalar [2021] hamda Brachman va Levesque [2022a] tomonidan tasvirlangan.

19.5. Mashqlar

19.1-mashq. 19.1-rasm qaror daraxtidagi har bir barg uchun (qaysi biri dalalanish» yoki «bajarish» bilan boshlanadi), ushbu bargda tugaydigan xarakteristikaga ega bo'lgan ilovaning misolini keltiring. Masalan, birinchi barg uchun stavkalar past bo'lgan, ko'p miqdorda bir xil ma'lumot mavjud bo'lgan va hokazo bo'lgan dasturiy misolni keltiring. 19.1-rasmdagi tavsiyalarni bajarish kerak bo'lmagan dasturiy misolni taklif qiling. Nima uchunligini tushuntiring.

19.2-mashq. Quyidagi ssenariylardan birini ko'rib chiqing.

(i) Siz bir kompaniyada ishlaysiz va ularga kelgusi saylov natijalarini bashorat qilish uchun vosita yaratish imkoniyati haqida so'rashgan. Sizda oldingi saylovlar natijalari, saylovchilar haqida aholini ro'yxatga olish ma'lumotlari, partiyalar va nomzodlar haqidagi ma'lumotlarga ega bo'lasiz. Maqsad qaysi partiya yoki partiyalar hukumatni tuzish ehtimolini bashorat qilishdir. Raqobatchi kompaniya bu muammoni yashirin Markov modellarini va gradiyent kuchaytirilgan daraxtlarni birlashtirish orqali hal qilishni taklif qilmoqda.

(ii) Yirik milliy qahva va ponchik do'koni o'zlarining ponchik ishlab chiqarishni modernizatsiya qilishni xohlaydi. Hozirda ular turli xil ponchiklar va qandolat mahsulotlarini pishirib bezaydigan va ularni bitta liniyaga tashlaydigan yagona ishlab chiqarish mashinasiga ega. Ushbu chiziqdagi ponchiklar keyin inson operatorlari tomonidan ko'rgazma va yetkazib berish uchun qutilarga solinadi. Ular ikkita turdagi qutilarni to'ldirishi kerak: hamma ponchiklar bir xil turdagi bo'lgan qutilar va 12 ta noyob turdagi ponchiklarni o'z ichiga olgan qutilar (ponchiklar turi 12 tadan ko'p). Kompaniya ushbu saralovchi kasbida ishlovchilarni robotlar bilan almashtirishni taklif qilmoqda. Sizning ishingiz ularga maslahat berishdir. Raqobatchi kompaniya chuqur o'rganish va deterministik rejalashtirishdan foydalanishni taklif qilgan.

(iii) Bir dori-darmonlar firmasi sizning kompaniyanizga turli dori-darmonlarning bemorlarga ta'sirini tushunish maqsadida shifokorlar uchun dori-darmonlar o'zaro ta'siri bo'yicha maslahat vositasi ishlab chiqish uchun murojaat qiladi. Ular turli dorilarning ta'siri bo'yicha yetakchi tadqiqotlar haqidagi bir qator katta ma'lumotlar bazalariga ega. Biroq, ma'lumotlar bazalari ba'zi dorilar va kasalliklar haqida qisman mos keladi, lekin boshqalari haqida emas. Ma'lumotlar bazalari dorilar, kasalliklar va ularning ta'sirini ifodalash uchun turli belgilarni ishlatadi. Har bir ma'lumotlar bazasi dorilarning ma'lum kasalliklarni davolash, salbiy ta'sirlarni keltirib chiqarish yoki hech qanday ta'sir ko'rsatmaslik ehtimolini bashorat qiladi. Raqobatchi kompaniya bu muammoni ontologiyalar va ehtimoliy relyatsion modellar aralashmasi yordamida hal qilishni taklif qilgan.

(a) Muammo qanday qilib agentning abstraksiyasiga mos kelishini tushuntiring.

(b) Raqobatchi kompaniyaning yechimi qanday ishlashi mumkinligini tushuntiring va ular taklif qilgan texnologiyalarni tanlash sabablari haqida izoh bering.

(c) Ushbu muammoni hal qilishning eng muammoli jihati nima? Buni hal qilish uchun qanday tavsiya berasiz? Har qanday tavsiyalarni asoslang.

19.3-mashq. Hozirgi SI ilovalarini toping va ushbu ilovalar uchun zamonaviy holatini o'lgan bo'yicha tasniflang. Ushbu ilova Kahneman [2011] tomonidan 1-tizim yoki 2-tizim deb atalgan narsalarni yoxud ikkisini ham avtomatlashtiradimi (yoki hech birini)?

19.4-mashq. 19.4-mashq Kompyuterlar odamlardan ko'ra aqlliyoq bo'lishi mumkin bo'lgan o'ziga xoslik ehtimoli to'g'risida ushbu bayonotlarning har biri uchun va unga qarshi dalillar keltiring.

(a) Singulyarlik hech qachon yuz bermaydi, chunki insonlar kompyuterlar bilan hech qachon taqqoslanmaydigan umumiy aqlga ega.

(b) Singulyarlik allaqachon yuz berdi. Google tomonidan berilgan javobga tasodifiy insonning javobidan ko'ra ko'proq ishonaman.

(c) Singulyarlik ma'nosiz, chunki insonlar kompyuterlar bilan shunchalik yaqin bog'langanlarki, o'zaro ta'sir har doim ularning har biriga qaraganda aqlliyoq bo'ladi.

(d) Singulyarlik bir necha o'nyillik ichida yuz beradi va insonlarni kompyuterlarga bo'ysundiradi.

- AAAI [2019]. AAAI code of professional ethics and conduct. <https://www.aaai.org/Conferences/code-of-ethics-and-conduct.php>.
- Abelson, H. and DiSessa, A. [1981]. *Turtle Geometry: The Computer as a Medium for Exploring Mathematics*. MIT Press.
- Acemoglu, D., Ozdaglar, A., and Siderius, J. [2021]. A model of online misinformation. Working Paper 28884, National Bureau of Economic Research. <http://dx.doi.org/10.3386/w28884>.
- ACM Committee on Professional Ethics [2018]. ACM code of ethics and professional conduct. <https://ethics.acm.org>.
- Agrawal, A., Gans, J., and Goldfarb, A. [2019]. *The Economics of Artificial Intelligence: An Agenda*. National Bureau of Economic Research Conference Report. University of Chicago Press.
- Agrawal, A., Gans, J., and Goldfarb, A. [2022]. *Prediction Machines, Updated and Expanded: The Simple Economics of Artificial Intelligence*. Harvard Business Review Press.
- Agre, P. E. [1995]. Computational research on interaction and agency. *Artificial Intelligence*, 72:1–52.
- Ajunwa, I. [2020]. The paradox of automation as anti-bias intervention. *Cardozo, L. Rev.*, 167.
- Alammar, J. [2018]. The illustrated transformer. <https://jalammar.github.io/illustrated-transformer/>.
- Albus, J. S. [1981]. *Brains, Behavior and Robotics*. BYTE Publications.
- Algorithm Watch [2022]. A guide to the AI act. <https://algorithmwatch.org/en/ai-act-explained/>.
- Allais, M. and Hagen, O. (eds.) [1979]. *Expected Utility Hypothesis and the Allais Paradox*. Reidel.
- Allemang, D., Hendler, J., and Gandon, F. [2020]. *Semantic Web for the Working Ontologist: Effective Modeling for Linked Data, RDFS and OWL*. ACM Books, 3rd edition.
- Amershi, S., et al. [2019]. Guidelines for human–AI interaction. In *CHI 2019*. ACM. <https://www.microsoft.com/en-us/research/publication/guidelines-for-human-ai-interaction/>. CHI 2019 Honorable Mention Award.
- Amodei, D., et al. [2016]. Concrete Problems in AI Safety. *ArXiv e-prints*, arXiv:1606.06565.
- Andreac, J. H. [1963]. STELLA: A scheme for a learning machine. In *2nd IFAC Congress*, pp. 497–502.
- Andrieu, C., de Freitas, N., Doucet, A., and Jordan, M. I. [2003]. An introduction to MCMC for machine learning. *Machine Learning*, 50(1–2):5–43.
- Antoniou, G. and van Harmelen, F. [2008]. *A Semantic Web Primer*. MIT Press, 2nd edition.
- Apt, K. and Bol, R. [1994]. Logic programming and negation: A survey. *Journal of Logic Programming*, 19/20:9–71.

- Archer, S. [2022]. *Salience: A Philosophical Inquiry*. Routledge.
- Aristotle [350 BCE]. *Categories*. Translated by E. M. Edghill. <http://classics.mit.edu/Aristotle/categories.html>.
- Arrp, R., Smith, B., and Spear, A. [2015]. *Building Ontologies with Basic Formal Ontology*. MIT Press.
- Arrow, K. [1963]. *Social Choice and Individual Values*. Wiley, 2nd edition.
- Asimov, I. [1950]. *I, Robot*. Doubleday.
- Auer, P., Cesa-Bianchi, N., and Fischer, P. [2002]. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47(2):235–256. <http://dx.doi.org/10.1023/A:1013689704352>.
- Auer, S., et al. [2007]. DBpedia: A nucleus for a web of open data. In *6th International Semantic Web Conference (ISWC)*.
- Awad, E., et al. [2018]. The moral machine experiment. *Nature*, 563(7729):59–64. <http://dx.doi.org/10.1038/s41586-018-0637-6>.
- Awad, E., et al. [2020]. Crowdsourcing moral machines. *Communications of the ACM*, 63(3):Pages 48–55.
- Baader, F., et al. (eds.) [2007]. *The Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press, 2nd edition. <http://dx.doi.org/10.1017/CBO9780511711787>.
- Bacchus, F. and Grove, A. [1995]. Graphical models for preference and utility. In *Uncertainty in Artificial Intelligence (UAI-95)*, pp. 3–10.
- Bacchus, F. and Kabanza, F. [1996]. Using temporal logic to control search in a forward chaining planner. In Ghallab, M. and Milani, A. (eds.), *New Directions in AI Planning*, pp. 141–153. ISO Press.
- Bach, S. H., Broecheler, M., Huang, B., and Getoor, L. [2017]. Hinge-loss Markov random fields and probabilistic soft logic. *Journal of Machine Learning Research (JMLR)*, 18:1–67.
- Bäck, T. [1996]. *Evolutionary Algorithms in Theory and Practice*. Oxford University Press.
- Back, M., et al. [2021]. Accurate prediction of protein structures and interactions using a three-track neural network. *Science*, 373(6557):871–876. <http://dx.doi.org/10.1126/science.abj8754>.
- Bahdanau, D., Cho, K., and Bengio, Y. [2015]. Neural machine translation by jointly learning to align and translate. In *International Conference on Learning Representations (ICLR)*. <http://dx.doi.org/10.48550/arXiv.1409.0473>.
- Bakhtin, A., et al. [2022]. Human-level play in the game of Diplomacy by combining language models with strategic reasoning. *Science*, 378(6624):1067–1074. <http://dx.doi.org/10.1126/science.ade9097>.
- Bakker, K., et al. [2020]. Digital technologies and dynamic resource management. *2020 IEEE International Conference on Smart Computing (SMARTCOMP)*, pp. 368–373.

Ballard, B. W. [1983]. The α -minimax search procedure for trees containing chance nodes. *Artificial Intelligence*, 21(3):327–350.

Bansal, G., et al. [2021]. Does the whole exceed its parts? The effect of AI explanations on complementary team performance. In *2021 CHI Conference on Human Factors in Computing Systems*. <http://dx.doi.org/10.1145/3411764.3445717>.

Bartlett, F. C. [1932]. *Remembering: A Study in Experimental and Social Psychology*. Cambridge University Press. <http://dx.doi.org/10.1017/CBO9780511759185>.

Baum, E. B. [2004]. *What is Thought?* MIT Press.

Bayes, T. [1763]. An essay towards solving a problem in the doctrine of chances. *Philosophical Transactions of the Royal Society of London*, 53:370–418. <https://doi.org/10.1098/rstl.1763.0053>.

Bell, R. M. and Koren, Y. [2007]. Lessons from the netflix prize challenge. *SIGKDD Explor. Newsl.*, 9(2):75–79. <http://dx.doi.org/10.1145/1345448.1345465>.

Bellman, R. [1957]. *Dynamic Programming*. Princeton University Press.

Bender, E. M., Gebru, T., McMillan-Major, A., and Shmitchell, S. [2021]. On the dangers of stochastic parrots: Can language models be too big? In *2021 ACM Conference on Fairness, Accountability, and Transparency*, pp. 610–623. <http://dx.doi.org/10.1145/3442188.3445922>.

Bender, E. M. and Koller, A. [2020]. Climbing towards NLU: On meaning, form, and understanding in the age of data. In *58th Annual Meeting of the Association for Computational Linguistics*, pp. 5185–5198. <http://dx.doi.org/10.18653/v1/2020.acl-main.463>.

Bengio, Y., Ducharme, R., Vincent, P., and Janvin, C. [2003]. A neural probabilistic language model. *J. Mach. Learn. Res. (JMLR)*, 3:1137–1155.

Benjamin, R. [2019]. *Race After Technology : Abolitionist Tools for the New Jim Code*.

Polity, Bent, R. and Van Hentenryck, P. [2004]. A two-stage hybrid local search for the vehicle routing problem with time windows. *Transportation Science*, 38(4):515–530.

Berners-Lee, T., Hendler, J., and Lassila, O. [2001]. The semantic web: A new form of web content that is meaningful to computers will unleash a revolution of new possibilities. *Scientific American*, May:28–37.

Bertel'e, U. and Brioschi, F. [1972]. *Nonserial Dynamic Programming*. Academic Press.

Bertsekas, D. P. [2017]. *Dynamic Programming and Optimal Control*. Athena Scientific, 4th edition.

Besnard, P. and Hunter, A. [2008]. *Elements of Argumentation*. MIT Press.

Bickel, P. J., Hammel, E. A., and O'Connell, J. W. [1975]. Sex bias in graduate admissions: Data from Berkeley. *Science*, 187(4175):398–404.

Biere, A., Heule, M., van Maaren, H., and Walsh, T. (eds.) [2021]. *Handbook of Satisfiability*. IOS Press, 2nd edition.

Bishop, C. M. [2008]. *Pattern Recognition and Machine Learning*. Springer-Verlag.
Bisk, Y., et al. [2020]. Experience grounds language. *CoRR*, abs/2004.10151. <https://arxiv.org/abs/2004.10151>.

Blei, D. M., Ng, A. Y., and Jordan, M. I. [2003]. Latent Dirichlet allocation. *Journal of Machine Learning Research (JMLR)*, 3:993–1022.

Bleichrodt, H., Rohde, K. I., and Wakker, P. P. [2008]. Koopmans' constant discounting for intertemporal choice: A simplification and a generalization.

Journal of Mathematical Psychology, 52(6):341–347. <http://dx.doi.org/https://doi.org/10.1016/j.jmp.2008.05.003>.

Blodgett, S. L., Barocas, S., Daumé III, H., and Wallach, H. [2020]. Language (technology) is power: A critical survey of “bias” in NLP. In *58th Annual Meeting of the Association for Computational Linguistics*. <http://dx.doi.org/10.18653/v1/2020.acl-main.485>.

Blum, A. and Furst, M. [1997]. Fast planning through planning graph analysis. *Artificial Intelligence*, 90:281–300.

Bobrow, D. G. [1967]. Natural language input for a computer problem solving system. In Minsky, M. (ed.), *Semantic Information Processing*, pp. 133–215. MIT Press.

Bobrow, D. G. [1993]. Artificial intelligence in perspective: a retrospective on fifty volumes of *Artificial Intelligence*. *Artificial Intelligence*, 59:5–20.

Boddy, M. and Dean, T. L. [1994]. Deliberation scheduling for problem solving in time-constrained environments. *Artificial Intelligence*, 67(2):245–285.

Bodlaender, H. L. [1993]. A tourist guide through treewidth. *Acta Cybernetica*, 11(1–2):1–21.

Bommasani, R. et al. [2021]. On the opportunities and risks of foundation models. *CoRR*, abs/2108.07258. <https://arxiv.org/abs/2108.07258>.

Bonnefon, J.-F. [2021]. *The Car That Knew Too Much Can a Machine Be Moral?* MIT Press.

Bostrom, N. [2014]. *Superintelligence: Paths, Dangers, Strategies*. Oxford University Press.

Boutilier, C., Dean, T., and Hanks, S. [1999]. Decision-theoretic planning: Structural assumptions and computational leverage. *Journal of Artificial Intelligence Research*, 11:1–94.

Boutilier, C., et al. [2004]. CP-nets: A tool for representing and reasoning with conditional ceteris paribus preference statements. *Journal of Artificial Intelligence Research*, 21:135–191.

Brachman, R. J. and Levesque, H. J. (eds.) [1985]. *Readings in Knowledge Representation*.

Morgan Kaufmann. Brachman, R. J. and Levesque, H. J. [2004]. *Knowledge Representation and Reasoning*.

Morgan Kaufmann. Brachman, R. J. and Levesque, H. J. [2022a]. *Machines like Us: Toward AI with Common Sense*. MIT Press.

- Brachman, R. J. and Levesque, H. J. [2022b]. Toward a new science of common sense. In *Thirty-Sixth AAAI Conference on Artificial Intelligence (AAAI-22)*.
- Breiman, L. [2001]. Random forests. *Machine Learning*, 45(1):5–32.
- Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. [1984]. *Classification and Regression Trees*. Wadsworth & Brooks.
- Brémaud, P. [1999]. *Markov Chains: Gibbs Fields, Monte Carlo Simulation and Queues*. Springer.
- Brin, S. and Page, L. [1998]. The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems*, 30(1–7):107–117. <http://www.sciencedirect.com/science/article/pii/S016975529800110X>.
- Brooks, R. A. [1986]. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, 2(1):14–23.
- Brooks, R. A. [1990]. Elephants don't play chess. *Robotics and Autonomous Systems*, 6:3–15.
- Brooks, R. A. [1991]. Intelligence without representation. *Artificial Intelligence*, 47:139–159.
- Brooks, R. A. [2018]. My dated predictions. <https://rodneymbrooks.com/my-dated-predictions/>.
- Broussard, M. [2018]. *Artificial Unintelligence: How Computers Misunderstand the World*. MIT Press. <http://dx.doi.org/10.7551/mitpress/11022.001.0001>.
- Brown, N. and Sandholm, T. [2019]. Superhuman AI for multiplayer poker. *Science*, 365(6456):885–890. <http://dx.doi.org/10.1126/science.aay2400>.
- Brown, T., et al. [2020]. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901. <https://arxiv.org/abs/2005.14165>.
- Brundtland, G. H. et al. [1987]. *Our Common Future*. United Nations, Report of the World Commission on Environment and Development. <https://sustainabledevelopment.un.org/content/documents/5987our-common-future.pdf>.
- Bryce, D. and Kambhampati, S. [2007]. A tutorial on planning graph-based reachability heuristics. *AI Magazine*, 28(1):47–83.
- Brynjolfsson, E. and McAfee, A. [2014]. *The Second Machine Age: Work, Progress, and Prosperity in a Time of Brilliant Technologies*. W. W. Norton & Co.
- Bryson, J. J. [2011]. AI robots should not be considered moral agents. In Berlatsky, N. (ed.), *Artificial Intelligence, Opposing Viewpoints*, pp. 155–168. Greenhaven Press.
- Bryson, J. J. [2018]. Patience is not a virtue: the design of intelligent systems and systems of ethics. *Ethics and Information Technology*, 20(1):15–26. <http://dx.doi.org/10.1007/s10676-018-9448-6>.
- Buchanan, B. G. [2005]. A (very) brief history of artificial intelligence. *AI Magazine*, 26(4):53–60.

Buchanan, B. G. and Feigenbaum, E. A. [1978]. Dendral and Meta-Dendral: Their applications dimension. *Artificial Intelligence*, 11:5–24.

Buchanan, B. G. and Shortliffe, E. (eds.) [1984]. *Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project*. Addison-Wesley.

Buntine, W. [1992]. Learning classification trees. *Statistics and Computing*, 2:63–73.

Buntine, W. L. [1994]. Operations for learning with graphical models. *Journal of Artificial Intelligence Research*, 2:159–225.

Buolamwini, J. and Gebru, T. [2018]. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *1st Conference on Fairness, Accountability and Transparency*. <https://proceedings.mlr.press/v81/buolamwini18a.html>.

Burch, R. [2022]. Charles Sanders Peirce. *The Stanford Encyclopedia of Philosophy*. <http://plato.stanford.edu/archives/sum2022/entries/peirce/>.

Busoniu, L., Babuska, R., and Schutter, B. D. [2008]. A comprehensive survey of multiagent reinforcement learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 38(2):156–172.

Calo, R. [2014]. The case for a federal robotics commission. *Brookings Institution Center for Technology Innovation*.

Calo, R., Froomkin, A. M., and Kerr, I. [2016]. *Robot Law*. Edward Elgar.

Campbell, M., Hoane Jr., A. J., and Hse, F.-h. [2002]. Deep Blue. *Artificial Intelligence*, 134(1–2):57–83.

Caswell, I. and Liang, B. [2020]. Recent advances in Google translate. <https://ai.googleblog.com/2020/06/recent-advances-in-google-translate.html>.

Cauchy, A. [1847]. Méthode g'én'érale pour la r'esolution des syst'emes d'equations simultan'ees. *C. R. Acad. Sci. Paris*, 25:536–538.

Center for AI and Digital Policy [2023]. Artificial intelligence and democratic values. <https://www.caidp.org/reports/aidv-2021/>.

Chapman, D. [1987]. Planning for conjunctive goals. *Artificial Intelligence*, 32(3):333–377.

Charlton, J. I. [1998]. *Nothing About Us Without Us: Disability Oppression and Empowerment*. University of California Press, 1st edition. <http://www.jstor.org/stable/10.1525/j.ctt1pnqn9>.

Chaudhri, V. K., et al. [2022]. Knowledge graphs: Introduction, history and, perspectives. *AI Magazine*, 43(1):17–29.

Cheeseman, P., et al. [1988]. Autoclass: A Bayesian classification system. In *Fifth International Conference on Machine Learning*, pp. 54–64. Reprinted in Shavlik and Dietterich [1990].

Chen, J., Holte, R. C., Zilles, S., and Sturtevant, N. R. [2017]. Front-to-end bidirectional heuristic search with near-optimal node expansions. In *IJCAI-2017*.

- Chen, T. and Guestrin, C. [2016]. Xgboost: A scalable tree boosting system. In *KDD '16: 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794. <https://doi.org/10.1145/2939672.2939785>.
- Cheng, J. and Druzdzel, M. [2000]. AIS-BN: An adaptive importance sampling algorithm for evidential reasoning in large Bayesian networks. *Journal of Artificial Intelligence Research*, 13:155–188. <http://www.jair.org/papers/paper764.html>.
- Chesnevar, C., Maguitman, A., and Loui, R. [2000]. Logical models of argument. *ACM Computer Surveys*, 32(4):337–383.
- Choi, Y., Vergari, A., and Van den Broeck, G. [2020]. Probabilistic circuits: A unifying framework for tractable probabilistic models. Technical report, UCLA StarAI Lab. <http://starai.cs.ucla.edu/papers/ProbCirc20.pdf>.
- Chollet, F. [2021]. *Deep Learning with Python*. Manning.
- Chomsky, N. [1957]. *Syntactic Structures*. Mouton & Co.
- Chowdhery, A., et al. [2022]. PaLM: Scaling language modeling with pathways. <http://dx.doi.org/10.48550/arXiv.2204.02311>.
- Chrisley, R. and Begeer, S. [2000]. *Artificial intelligence: Critical Concepts in Cognitive Science*. Routledge.
- Christian, B. [2020]. *The Alignment Problem: Machine Learning and Human Values*. W. W. Norton & Co.
- Clark, K. L. [1978]. Negation as failure. In Gallaire, H. and Minker, J. (eds.), *Logic and Databases*, pp. 293–322. Plenum Press.
- Cohen, P. R. [2005]. If not Turing's test, then what? *AI Magazine*, 26(4):61–67.
- Colledanchise, M. and O'gren, P. [2018]. *Behavior Trees in Robotics and AI: An Introduction*. CRC Press.
- Colmerauer, A., Kanoui, H., Roussel, P., and Pasero, R. [1973]. Un syst'eme de communication homme-machine en francais. Technical report, Groupe de Recherche en Intelligence Artificielle, Universit'e d'Aix-Marseille.
- Colmerauer, A. and Roussel, P. [1996]. The birth of Prolog. In Bergin, T. J. and Gibson, R. G. (eds.), *History of Programming Languages-II*, pp. 331–367. ACM Press/Addison-Wesley.
- Conati, C., Gertner, A. S., and VanLehn, K. [2002]. Using Bayesian networks to manage uncertainty in student modeling. *User Modeling and User-Adapted Interaction*, 12(4):371–417. <http://dx.doi.org/10.1023/A:1021258506583>.
- Confucius [500 BCE]. *Confucian Analects*. translated by James Legge [1893]. <https://www.sacred-texts.com/cfu/conf1.htm>.
- Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C. [2022]. *Introduction to Algorithms*. MIT Press, 4th edition.
- Cover, T. M. and Thomas, J. A. [2006]. *Elements of Information Theory*. Wiley, 2nd edition.

Cramer, J. [2002]. The origins of logistic regression. Working Paper 2002-119/4, Tinbergen Institute. <http://dx.doi.org/10.2139/ssrn.360300>.

Crawford, K. [2021]. *The Atlas of AI: Power, Politics, and the Planetary Costs of Artificial Intelligence*. Yale University Press.

Culberson, J. and Schaeffer, J. [1998]. Pattern databases. *Computational Intelligence*, 14(3):318–334.

Dadich, S. [2016]. Barack Obama, neural nets, self-driving cars, and the future of the world. *Wired*.

Dahl, V. [1994]. Natural language processing and logic programming. *Journal of Logic Programming*, 19/20:681–714.

Danaher, J. [2021]. Automation and the future of work. In *The Oxford Handbook of Digital Ethics*. Oxford University Press. <http://dx.doi.org/10.1093/oxfordhb/9780198857815.013.37>.

Darwiche, A. [2001]. Recursive conditioning. *Artificial Intelligence*, 126(1-2):5–41.

Darwiche, A. [2009]. *Modeling and Reasoning with Bayesian Networks*. Cambridge University Press.

Darwiche, A. [2018]. Human-level intelligence or animal-like abilities? *Communication of the ACM*, 61(10):56–67. <http://dx.doi.org/10.1145/3271625>.

Davis, E. [1990]. *Representations of Commonsense Knowledge*. Morgan Kaufmann.

Davis, E. [2015]. A collection of Winograd schemas. <http://www.cs.nyu.edu/faculty/davise/papers/WinogradSchemas/WSCollection.html>.

Davis, J. and Goadrich, M. [2006]. The relationship between precision-recall and ROC curves. In *23rd International Conference on Machine Learning (ICML)*, pp. 233–240.

Davis, M., Logemann, G., and Loveland, D. [1962]. A machine program for theorem proving. *Communications of the ACM*, 5(7):394–397.

Davis, M. and Putnam, H. [1960]. A computing procedure for quantification theory. *Journal of the ACM*, 7(3):201–215.

De Jong, K. A. [2006]. *Evolutionary Computation: A Unified Approach*. MIT Press.

de Kleer, J. [1986]. An assumption-based TMS. *Artificial Intelligence*, 28(2):127–162. de Kleer, J., Mackworth, A. K., and Reiter, R. [1992]. Characterizing diagnoses and systems. *Artificial Intelligence*, 56:197–222.

De Raedt, L., Frasconi, P., Kersting, K., and Muggleton, S. H. (eds.) [2008]. *Probabilistic Inductive Logic Programming*. Springer.

De Raedt, L., Kersting, K., Natarajan, S., and Poole, D. [2016]. *Statistical Relational Artificial Intelligence: Logic, Probability, and Computation*. Morgan & Claypool. <http://dx.doi.org/10.2200/S00692ED1V01Y201601AIM032>.

De Raedt, L., Kimmig, A., and Toivonen, H. [2007]. ProbLog: A probabilistic Prolog and its application in link discovery. In *20th International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 2462–2467.

- Dean, T. and Kanazawa, K. [1989]. A model for reasoning about persistence and causation. *Computational Intelligence*, 5(3):142–150.
- Dean, T. L. and Wellman, M. P. [1991]. *Planning and Control*. Morgan Kaufmann.
- Dechter, R. [1996]. Bucket elimination: A unifying framework for probabilistic inference. In *Twelfth Conference on Uncertainty in Artificial Intelligence (UAI-96)*, pp. 211–219.
- Dechter, R. [2003]. *Constraint Processing*. Morgan Kaufmann.
- Dechter, R. [2019]. *Reasoning with Probabilistic and Deterministic Graphical Models*. Morgan & Claypool, 2nd edition.
- Dechter, R. and Pearl, J. [1985]. Generalized best-first search strategies and the optimality of A*. *Journal of the Association for Computing Machinery*, 32(3):505–536.
- Dellaert, F., Fox, D., Burgard, W., and Thrun, S. [1999]. Monte Carlo localization for mobile robots. In *IEEE International Conference on Robotics and Automation (ICRA)*.
- Delling, D., Goldberg, A. V., Pajor, T., and Werneck, R. F. [2015]. Customizable route planning in road networks. *Transportation Science*, 51(2):566–591. <https://doi.org/10.1287/trsc.2014.0579>.
- Dempster, A., Laird, N., and Rubin, D. [1977]. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39:1–38. With discussion.
- Deng, J., et al. [2009]. ImageNet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition Conference (CVPR)*.
- Denil, M., Matheson, D., and de Freitas, N. [2014]. Narrowing the gap: Random forests in theory and in practice. In *International Conference on Machine Learning (ICML)*.
- Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. [2019]. BERT: Pre-training of deep bidirectional transformers for language understanding. In *2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 4171–4186. <http://dx.doi.org/10.18653/v1/N19-1423>.
- Dietterich, T. G. [2000a]. An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. *Machine Learning*, 40(2):139–158.
- Dietterich, T. G. [2000b]. Hierarchical reinforcement learning with the MAXQ value function decomposition. *Journal of Artificial Intelligence Research*, 13:227–303.
- Dietterich, T. G. [2002]. Ensemble learning. In Arbib, M. (ed.), *The Handbook of Brain Theory and Neural Networks*, pp. 405–408. MIT Press, 2nd edition.
- Dijkstra, E. W. [1959]. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271. <https://doi.org/10.1007/BF01386390>.
- Dijkstra, E. W. [1976]. *A Discipline of Programming*. Prentice-Hall.

Dolgov, D., Thrun, S., Montemerlo, M., and Diebel, J. [2010]. Path planning for autonomous vehicles in unknown semi-structured environments. *The International Journal of Robotics Research*, 29(5):485–501. <http://dx.doi.org/10.1177/0278364909359210>.

Domingos, P. and Lowd, D. [2009]. *Markov Logic: An Interface Layer for Artificial Intelligence*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool. <http://dx.doi.org/doi:10.2200/S00206ED1V01Y200907AIM007>.

Doucet, A., de Freitas, N., and Gordon, N. (eds.) [2001]. *Sequential Monte Carlo in Practice*. Springer-Verlag.

Doyle, J. [1979]. A truth maintenance system. AI Memo 521, MIT AI Laboratory.

Dresner, K. and Stone, P. [2008]. A multiagent approach to autonomous intersection management. *Journal of Artificial Intelligence Research*, 31:591–656.

du Boulay, B., Mitrovic, T., and Yacef, K. (eds.) [2023]. *Handbook of Artificial Intelligence in Education*. Edward Elgar.

Dua, D. and Graff, C. [2017]. UCI machine learning repository. <http://archive.ics.uci.edu/ml>.

Duda, R. O., Hart, P. E., and Stork, D. G. [2001]. *Pattern Classification*. Wiley-Interscience, 2nd edition.

Dung, P. [1995]. On the acceptability of arguments and its fundamental role in non-monotonic reasoning, logic programming and *n*-person games. *Artificial Intelligence*, 77(2):321–357.

Dzeroski, S., De Raedt, L., and Driessens, K. [2001]. Relational reinforcement learning. *Machine Learning*, 43:7–52.

Einstein, A. [1934]. On the method of theoretical physics. *Philosophy of Science*, 1(2):163–169.

Eubanks, V. [2018]. *Automating Inequality: How High-Tech Tools Profile, Police, and Punish the Poor*. St. Martin's Publishing Group.

European Commission [2021]. The general data protection regulation. https://commission.europa.eu/law/law-topic/data-protection/data-protection-eu_en.

European Commission [2022a]. AI liability directive. [https://www.europarl.europa.eu/RegData/etudes/BRIE/2023/739342/EPRS_BRI\(2023\)739342_EN.pdf](https://www.europarl.europa.eu/RegData/etudes/BRIE/2023/739342/EPRS_BRI(2023)739342_EN.pdf).

European Commission [2022b]. The artificial intelligence act. <https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:52021PC0206>.

European Commission [2022c]. The digital services act package. <https://digital-strategy.ec.europa.eu/en/policies/digital-services-act-package>.

European Commission [2022d]. New liability rules on products and AI to protect consumers. https://ec.europa.eu/commission/presscorner/detail/en/ip_22_5807.

Falco, G., et al. [2021]. Governing AI safety through independent audits. *Nature Machine Intelligence*, 3(7):566–571. <http://dx.doi.org/10.1038/>

- Fanshel, S. and Bush, J. [1970]. Health-status index and its application to health-services outcomes. *Operations Research*, 18.
- Fatemi, B., Taslakian, P., Vazquez, D., and Poole, D. [2020]. Knowledge hypergraphs: Prediction beyond binary relations. In *29th International Joint Conference on Artificial Intelligence (IJCAI)*.
- Fedus, W., Zoph, B., and Shazeer, N. [2021]. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. <http://dx.doi.org/10.48550/arXiv.2101.03961>.
- Fellegi, I. and Sunter, A. [1969]. A theory for record linkage. *Journal of the American Statistical Association*, 64(328):1183–1280.
- Felner, A., Korf, R. E., and Hanan, S. [2004]. Additive pattern database heuristics. *Journal of Artificial Intelligence Research*, 22:279–318.
- Feurer, M. and Hutter, F. [2019]. Hyperparameter optimization. In *Automated Machine Learning*. Springer. http://dx.doi.org/https://doi.org/10.1007/978-3-030-05318-5_1.
- Fikes, R. E. and Nilsson, N. J. [1971]. STRIPS: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, 2(3–4):189–208.
- Foot, P. [1967]. The problem of abortion and the doctrine of the double effect. *Oxford Review*, 5:5–15. <https://philpapers.org/archive/FOOTPO-2.pdf>.
- Forbus, K. [2019]. *Qualitative Representations: How People Reason and Learn about the Continuous World*. MIT Press.
- Forbus, K. D. and Hinrich, T. [2017]. Analogy and relational representations in the companion cognitive architecture. *AI Magazine*, 38(4):34–42. <http://dx.doi.org/10.1609/aimag.v38i4.2743>.
- Ford, M. [2021]. *Rule of the Robots: How Artificial Intelligence Will Transform Everything*. John Murray Press.
- François-Lavet, V., et al. [2018]. An introduction to deep reinforcement learning. *CoRR*, abs/1811.12560. <http://arxiv.org/abs/1811.12560>.
- Freuder, E. C. and Mackworth, A. K. [2006]. Constraint satisfaction: An emerging paradigm. In Rossi, F., Van Beek, P., and Walsh, T. (eds.), *Handbook of Constraint Programming*, pp. 13–28. Elsevier.
- Friedman, J. H. [2001]. Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5):1189–1232. <http://www.jstor.org/stable/2699986>.
- Friedman, N., Greiger, D., and Goldszmidt, M. [1997]. Bayesian network classifiers. *Machine Learning*, 29:103–130.
- Gabrilovich, E., et al. [2014]. Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *20th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.

Gal, K. and Grosz, B. J. [2022]. Multi-agent systems: Technical & ethical challenges of functioning in a mixed group. *Daedalus*.

Galton, F. [1886]. Regression towards mediocrity in hereditary stature. *Journal of the Anthropological Institute*, 15:246–263. <http://galton.org/essays/1880-1889/galton-1886-jaigi-regression-stature.pdf>.

Gangemi, A., Guarino, N., Masolo, C., and Oltramari, A. [2003]. Sweetening WordNet with DOLCE. *AI Magazine*, 24(3):13–24.

Garcia-Molina, H., Ullman, J. D., and Widom, J. [2009]. *Database Systems: The Complete Book*. Prentice Hall, 2nd edition.

Gardner, H. [1985]. *The Mind's New Science*. Basic Books.

Gebru, T., et al. [2021]. Datasheets for datasets. *Communication of the ACM*, 64(12):86–92. <http://dx.doi.org/10.1145/3458723>.

Geffner, H. and Bonet, B. [2013]. *A Concise Introduction to Models and Methods for Automated Planning*. Springer. <http://dx.doi.org/doi:10.2200/S00513ED1V-01Y201306AIM022>.

Geffner, H., Dechter, R., and Halpern, J. Y. (eds.) [2022]. *Probabilistic and Causal Inference: The Works of Judea Pearl*. ACM Books.

Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. B. [2013]. *Bayesian Data Analysis*. Chapman & Hall/CRC, 3rd edition. <http://www.stat.columbia.edu/~gelman/book/>.

Gelman, A., Hill, J., and Vehtari, A. [2020]. *Regression and Other Stories*. Cambridge University Press.

Genesereth, M. and Thielscher, M. [2014]. *General Game Playing*. Morgan & Claypool. <http://dx.doi.org/doi:10.2200/S00564ED1V01Y201311AIM024>.

Gers, F. A., Schmidhuber, J., and Cummins, F. [2000]. Learning to forget: Continual prediction with LSTM. *Neural Computation*, 12(10):2451–2471. <http://dx.doi.org/https://doi.org/10.1162/089976600300015015>.

Getoor, L. and Taskar, B. (eds.) [2007]. *Introduction to Statistical Relational Learning*. MIT Press.

Ghahramani, Z. [2015]. Probabilistic machine learning and artificial intelligence. *Nature*, 521(7553):452–459. <http://dx.doi.org/10.1038/nature14541>.

Ghallab, M., Nau, D., and Traverso, P. [2004]. *Automated Planning: Theory and Practice*. Elsevier.

Gibbard, A. [1973]. Manipulation of voting schemes: A general result. *Econometrica*, 41:587–601.

Gil, Y., et al. [2017]. Towards continuous scientific data analysis and hypothesis evolution. In *Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17)*. <http://www.isi.edu/~gil/papers/gil-et-al-aaai17.pdf>.

Gil, Y., et al. [2019]. Intelligent systems for geosciences: An essential research agenda. *Communications of the ACM*, 62. <http://dx.doi.org/10.1145/3192335>.

Glorot, X. and Bengio, Y. [2010]. Understanding the difficulty of training deep feed-forward neural networks. In *Thirteenth International Conference on Artificial Intelligence and Statistics*, pp. 249–256. <https://proceedings.mlr.press/v9/glorot10a.html>.

Glorot, X., Bordes, A., and Bengio, Y. [2011]. Deep sparse rectifier neural networks. In *14th International Conference on Artificial Intelligence and Statistics*, pp. 315–323.

Goble, C., et al. [2020]. FAIR Computational Workflows. *Data Intelligence*, 2(1-2):108–121. <http://dx.doi.org/10.1162/dint.a.00033>.

Goldberg, D. E. [2002]. *The Design of Innovation: Lessons from and for Competent Genetic Algorithms*. Addison-Wesley.

Goldberg, Y. [2016]. A primer on neural network models for natural language processing. *Journal of Artificial Intelligence Research*, 57:345–420. <http://dx.doi.org/doi:10.1613/jair.4992>.

Gomes, C., et al. [2019]. Computational sustainability: Computing for a better world and a sustainable future. *Communication of the ACM*, 62(9):56–65. <http://dx.doi.org/10.1145/3339399>.

Good, I. J. [1965]. Speculations concerning the first ultraintelligent machine. In Alt, F. and Ruminoff, M. (eds.), *Advances in Computers*, volume 6. Academic Press.

Goodfellow, I., Bengio, Y., and Courville, A. [2016]. *Deep Learning*. MIT Press. <http://www.deeplearningbook.org>.

Goodfellow, I. J., et al. [2014]. Generative adversarial networks. In *Advances in Neural Information Processing Systems 27 (NIPS 2014)*. <http://dx.doi.org/10.48550/arXiv.1406.2661>.

Gordon, M. L., et al. [2021]. The disagreement deconvolution: Bringing machine learning performance metrics in line with reality. In *2021 CHI Conference on Human Factors in Computing Systems*. <http://dx.doi.org/10.1145/3411764.3445423>.

Green, B. [2022]. The flaws of policies requiring human oversight of government algorithms. *Computer Law and Security Review*, 45:105681. <http://dx.doi.org/https://doi.org/10.1016/j.clsr.2022.105681>.

Green, C. [1969]. Application of theorem proving to problem solving. In *1st International Joint Conference on Artificial Intelligence*, pp. 219–237.

Grosz, B. [2012]. What question would Turing pose today? *AI Magazine*, 33(4):73. <http://dx.doi.org/10.1609/aimag.v33i4.2441>.

Grosz, B. J. [2018]. Smart enough to talk with us? Foundations and challenges for dialogue capable AI systems. *Computational Linguistics*, 44(1):1–15. <http://dx.doi.org/10.1162/COLI.a.00313>.

Grunwald, P. D. [2007]. *The Minimum Description Length Principle*. MIT Press.

Gunkel, D. [2018]. *Robot Rights*. MIT Press.

Halevy, A., Norvig, P., and Pereira, F. [2009]. The unreasonable effectiveness of data. *IEEE Intelligent Systems*, 24(2):8–12.

Halpern, J. Y. [2003]. *Reasoning about Uncertainty*. MIT Press.

Hamilton, W. L. [2020]. *Graph Representation Learning*. Morgan & Claypool.

Hardin, G. [1968]. The tragedy of the commons: The population problem has no technical solution; it requires a fundamental extension in morality. *Science*, 162(3859):1243–1248.

Harper, F. M. and Konstan, J. A. [2015]. The MovieLens datasets: History and context. *ACM Transactions on Interactive Intelligent Systems*, 5(4). <http://dx.doi.org/10.1145/2827872>.

Hart, P. E., Nilsson, N. J., and Raphael, B. [1968]. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2):100–107.

Hart, T. P. and Edwards, D. J. [1961]. The tree prune (TP) algorithm. Memo 30, MIT Artificial Intelligence Project.

Haslum, P., Lipovetzky, N., Magazzeni, D., and Muise, C. [2019]. *An Introduction to the Planning Domain Definition Language*. Morgan & Claypool.

<https://doi.org/10.2200/S00900ED2V01Y201902AIM042>.

Hastie, T., Tibshirani, R., and Friedman, J. [2009]. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2nd edition.

Haugeland, J. [1985]. *Artificial Intelligence: The Very Idea*. MIT Press.

Haugeland, J. (ed.) [1997]. *Mind Design II: Philosophy, Psychology, Artificial Intelligence*. MIT Press, revised and enlarged edition.

Hayes, P. J. [1973]. Computation and deduction. In *2nd Symposium on Mathematical Foundations of Computer Science*, pp. 105–118. Czechoslovak Academy of Sciences.

He, K., Zhang, X., Ren, S., and Sun, J. [2015]. Deep residual learning for image recognition. *CoRR*, abs/1512.03385. <http://arxiv.org/abs/1512.03385>.

Heath, T. and Bizer, C. [2011]. *Linked Data: Evolving the Web into a Global Data Space*. Springer.

Heckerman, D. [1999]. A tutorial on learning with Bayesian networks. In Jordan, M. (ed.), *Learning in Graphical Models*. MIT Press.

Hendler, J., Berners-Lee, T., and Miller, E. [2002]. Integrating applications on the semantic web. *Journal of the Institute of Electrical Engineers of Japan*, 122(10):676–680. <http://www.w3.org/2002/07/swint>.

Henrion, M. [1988]. Propagating uncertainty in Bayesian networks by probabilistic logic sampling. In Lemmer, J. F. and Kanal, L. N. (eds.), *Uncertainty in Artificial Intelligence 2*, pp. 149–163. Elsevier Science.

Hewitt, C. [1969]. Planner: A language for proving theorems in robots. In *1st International Joint Conference on Artificial Intelligence*, pp. 295–301.

Hinton, G., et al. [2012a]. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *Signal Processing Magazine, IEEE*, 29(6):82–97. <http://dx.doi.org/10.1109/MSP.2012.2205597>.

- Hinton, G. E., et al. [2012b]. Improving neural networks by preventing coadaptation of feature detectors. *CoRR*, abs/1207.0580. <http://arxiv.org/abs/1207.0580>.
- Hitchcock, F. L. [1927]. The expression of a tensor or a polyadic as a sum of products. *Studies in Applied Mathematics*, 6(1-4):164-189.
- Hitzler, P., et al. (eds.) [2012]. *OWL 2 Web Ontology Language Primer (Second Edition)*. W3C Recommendation 11 December 2012. <http://www.w3.org/TR/owl2-primer/>.
- Ho, J., Jain, A., and Abbeel, P. [2020]. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pp. 6840-6851. https://proceedings.neurips.cc/paper_files/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca-4b-Paper.pdf.
- Hochreiter, S. and Schmidhuber, J. [1997]. Long short-term memory. *Neural Computation*, 9:1735-1780.
- Hoffart, J., Suchanek, F., Berberich, K., and Weikum, G. [2013]. YAGO2: A spatially and temporally enhanced knowledge base from Wikipedia. *Artificial Intelligence*, 194:28-61.
- Hofstadter, D. [2022]. Artificial neural networks today are not conscious, according to Douglas Hofstadter. *The Economist*, June 11th 2022.
- Hogan, A. et al. [2021]. Knowledge graphs. *ACM Computing Surveys*, 54(4). <https://doi.org/10.1145/3447772>.
- Holland, J. H. [1975]. *Adaption in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. University of Michigan Press.
- Holling, C. S. [1973]. Resilience and stability of ecological systems. *Annual Review of Ecology and Systematics*, 4(1):1-23. <http://dx.doi.org/10.1146/annurev.es.04.110173.000245>.
- Hoos, H. H. and Stutzle, T. [2004]. *Stochastic Local Search: Foundations and Applications*. Morgan Kaufmann.
- Horvitz, E. J. [1989]. Reasoning about beliefs and actions under computational resource constraints. In Kanal, L., Levitt, T., and Lemmer, J. (eds.), *Uncertainty in Artificial Intelligence 3*, pp. 301-324. Elsevier.
- Horvitz, E. J. [2006]. Eric Horvitz forecasts the future. *New Scientist*, 2578:72.
- Howard, R. A. and Matheson, J. E. [1984]. Influence diagrams. In Howard, R. A. and Matheson, J. E. (eds.), *The Principles and Applications of Decision Analysis*. Strategic Decisions Group.
- Howson, C. and Urbach, P. [2006]. *Scientific Reasoning: The Bayesian Approach*. Open Court, 3rd edition.
- Huang, Y. and Valtorta, M. [2006]. Pearl's calculus of intervention is complete. In *Conference on Uncertainty in Artificial Intelligence*, pp. 217-224.

- Hume, D. [1739–40]. *A Treatise of Human Nature: Being an Attempt to Introduce the Experimental Method of Reasoning into Moral Subjects*. <https://gutenberg.org/files/4705/4705-h/4705-h.htm>.
- Hursthouse, R. and Pettigrove, G. [2018]. Virtue ethics. In Zalta, E. N. (ed.), *The Stanford Encyclopedia of Philosophy*. Winter 2018 edition. <https://plato.stanford.edu/archives/win2018/entries/ethics-virtue/>.
- Hutter, F., Kotthoff, L., and Vanschoren, J. (eds.) [2019]. *Automated Machine Learning Methods, Systems, Challenges*. Springer.
- IEEE [2020]. IEEE code of ethics. <https://www.ieee.org/about/corporate/governance/p7-8.html>.
- IEEE Global Initiative on Ethics of Autonomous and Intelligent Systems [2019]. Ethically aligned design: A vision for prioritizing human well-being with autonomous and intelligent systems. <https://standards.ieee.org/content/ieee-standards/en/industry-connections/ec/autonomous-systems.html>.
- IHTSDO [2016]. *SNOMED CT Starter Guide*. International Health Terminology Standards Development Organisation. <http://snomed.org>.
- Jackson, M. O. [2011]. *A Brief Introduction to the Basics of Game Theory*. SSRN. <http://dx.doi.org/10.2139/ssrn.1968579>.
- Jacobs, A. Z. and Wallach, H. [2021]. Measurement and fairness. In *2021 ACM Conference on Fairness, Accountability, and Transparency, FAccT '21*, pp. 375–385. <http://dx.doi.org/10.1145/3442188.3445901>.
- Jahrer, M., Töschner, A., and Legenstein, R. [2010]. Combining predictions for accurate recommender systems. In *16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 693–702. <http://dx.doi.org/10.1145/1835804.1835893>.
- Jannach, D. and Bauer, C. [2020]. Escaping the McNamara fallacy: Towards more impactful recommender systems research. *AI Magazine*, 41(4):79–95. <http://dx.doi.org/10.1609/aimag.v41i4.5312>.
- Jannach, D., Pu, P., Ricci, F., and Zanker, M. [2021]. Recommender systems: Past, present, future. *AI Magazine*, 42(3):3–6. <http://dx.doi.org/10.1609/aimag.v42i3.18139>.
- Janowicz, K., van Harmelen, F., Hendler, J. A., and Hitzler, P. [2015]. Why the data train needs semantic rails. *AI Magazine*, 36(1):5–14.
- Jarrett, K., Kavukcuoglu, K., Ranzato, M., and LeCun, Y. [2009]. What is the best multi-stage architecture for object recognition? In *2009 IEEE 12th International Conference on Computer Vision*. <http://dx.doi.org/10.1109/ICCV.2009.5459469>.
- Jaynes, E. T. [2003]. *Probability Theory: The Logic of Science*. Cambridge University Press. <https://bayes.wustl.edu/etj/prob/book.pdf>.
- Jordan, M. I. [2019]. Artificial intelligence – the revolution hasn't happened yet. *Harvard Data Science Review*, 1(1). <https://hdsr.mitpress.mit.edu/pub/wot7mkc1>.

- Joy, B. [2000]. Why the future doesn't need us. *Wired*. <http://www.wired.com/wired/archive/8.04/joy.html>.
- Jozefowicz, R., Zaremba, W., and Sutskever, I. [2015]. An empirical exploration of recurrent network architectures. In *32nd International Conference on Machine Learning, ICML'15*, pp. 2342–2350.
- Jumper, J., et al. [2021]. Highly accurate protein structure prediction with AlphaFold. *Nature*, 596(7873):583–589. <http://dx.doi.org/10.1038/s41586-021-03819-2>.
- Jurafsky, D. and Martin, J. H. [2023]. *Speech and Language Processing*. Unpublished, 3rd edition. <https://web.stanford.edu/~jurafsky/slp3/>.
- Kahneman, D. [2011]. *Thinking, Fast and Slow*. Allen Lane.
- Kakas, A. and Denecker, M. [2002]. Abduction in logic programming. In Kakas, A. and Sadri, F. (eds.), *Computational Logic: Logic Programming and Beyond*, pp. 402–436. Springer-Verlag.
- Kambhampati, S., Knoblock, C. A., and Yang, Q. [1995]. Planning as refinement search: A unified framework for evaluating design tradeoffs in partial order planning. *Artificial Intelligence*, 76:167–238.
- Kant, I. [1787]. *The Critique of Pure Reason*. <https://gutenberg.org/ebooks/4280>.
- Karimi, H., Nutini, J., and Schmidt, M. [2016]. Linear convergence of gradient and proximal-gradient methods under the Polyak–Łojasiewicz condition. In *European Conference on Machine Learning (ECML)*.
- Karpathy, A. [2015]. The unreasonable effectiveness of recurrent neural networks. <http://karpathy.github.io/2015/05/21/rnn-effectiveness/>.
- Katoch, S., Chauhan, S. S., and Kumar, V. [2021]. A review on genetic algorithm: past, present, and future. *Multimedia Tools and Applications*, 80(5):8091–8126. <http://dx.doi.org/10.1007/s11042-020-10139-6>.
- Kautz, H. and Selman, B. [1996]. Pushing the envelope: Planning, propositional logic and stochastic search. In *13th National Conference on Artificial Intelligence*, pp. 1194–1201.
- Kazemi, S. M. and Poole, D. [2018]. Simple embedding for link prediction in knowledge graphs. In *32nd Conference on Neural Information Processing Systems*.
- Kc, G., et al. [2017]. LightGBM: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems* 30.
- Kearns, M., Mansour, Y., and Ng, A. [2002]. A sparse sampling algorithm for near-optimal planning in large Markovian decision processes. *Machine Learning*, 49:193–208.
- Keeney, R. L. and Raiffa, H. [1976]. *Decisions with Multiple Objectives*. Wiley.
- Kendall, E. F. and McGuinness, D. L. [2019]. *Ontology Engineering*. Springer. <http://dx.doi.org/10.1007/978-3-031-79486-5>.
- Kharon, R. and Sanner, S. [2021]. Stochastic planning and lifted inference. In Van den Broeck, G., Kersting, K., Natarajan, S., and Poole, D. (eds.), *Introduction to Lifted Inference*. MIT Press.

- King, G. [2007]. An introduction to the dataverse network as an infrastructure for data sharing. *Sociological Methods and Research*, 36(2):173–199.
- King, R., et al. [2004]. Functional genomic hypothesis generation and experimentation by a robot scientist. *Nature*, 427:247–252. <http://www.doc.ic.ac.uk/~shm/Papers/OliverJan15hi.pdf>.
- King, R. D., et al. [2009a]. The automation of science. *Science*, 324(5923):85–89. <http://dx.doi.org/10.1126/science.1165620>.
- King, R. D., et al. [2009b]. The robot scientist Adam. *Computer*, 42(8):46–54. <http://dx.doi.org/10.1109/MC.2009.270>.
- Kirkpatrick, S., Gelatt, C. D., and Vecchi, M. P. [1983]. Optimization by simulated annealing. *Science*, 220:671–680.
- Kirsh, D. [1991a]. Foundations of AI: The big issues. *Artificial Intelligence*, 47:3–30.
- Kirsh, D. [1991b]. Today the earwig, tomorrow man? *Artificial Intelligence*, 47:161–184.
- Kleinberg, J., Ludwig, J., Mullainathan, S., and Sunstein, C. R. [2020]. Algorithms as discrimination detectors. In *National Academy of Sciences*.
- Knoll, B., et al. [2008]. AIspace: Interactive tools for learning artificial intelligence. In *AAAI 2008 AI Education Workshop*, p. 3.
- Knox, W. B. and Stone, P. [2009]. Interactively shaping agents via human reinforcement: The TAMER framework. In *Fifth International Conference on Knowledge Capture*, pp. 9–16. <http://dx.doi.org/10.1145/1597735.1597738>.
- Knublauch, H., Oberle, D., Tetlow, P., and Wallace, E. [2006]. A semantic web primer for object-oriented software developers. Working Group Note 9 March 2006, W3C. <http://www.w3.org/TR/sw-oosd-primer/>.
- Knuth, D. E. and Moore, R.W. [1975]. An analysis of alpha-beta pruning. *Artificial Intelligence*, 6(4):293–326.
- Kochenderfer, M. J. [2015]. *Decision Making Under Uncertainty*. MIT Press.
- Kochenderfer, M. J., Wheeler, T. A., and Wray, K. H. [2022]. *Algorithms for Decision Making*. MIT Press. <https://algorithmsbook.com>.
- Kocsis, L. and Szepesvári, C. [2006]. Bandit based Monte-Carlo planning. In *17th European Conference on Machine Learning (ECML)*, pp. 282–293.
- Koller, D. and Friedman, N. [2009]. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press.
- Koller, D. and Milch, B. [2003]. Multi-agent influence diagrams for representing and solving games. *Games and Economic Behavior*, 45(1):181–221. <http://people.csail.mit.edu/milch/papers/geb-maid.pdf>.
- Koopmans, T. [1972]. Representations of preference orderings over time. In McGuire, C. and Radner, R. (eds.), *Decisions and Organization*. North-Holland.
- Koren, Y. and Bell, R. [2011]. Advances in collaborative filtering. In Ricci, F., Rokach, L., Shapira, B., and Kantor, P. B. (eds.), *Recommender Systems Handbook*, pp. 145–186. Springer. http://dx.doi.org/10.1007/978-0-387-85820-3_5.

- Koren, Y., Bell, R., and Volinsky, C. [2009]. Matrix factorization techniques for recommender systems. *IEEE Computer*, 42(8):30–37.
- Korf, K. E. [1985]. Depth-first iterative deepening: An optimal admissible tree search. *Artificial Intelligence*, 27(1):97–109.
- Kowalski, R. [1979]. Algorithm = logic + control. *Communications of the ACM*, 22:424–431.
- Kowalski, R. A. [1974]. Predicate logic as a programming language. In *Information Processing 74*, pp. 569–574. North-Holland.
- Kowalski, R. A. [1988]. The early history of logic programming. *Communications of the ACM*, 31(1):38–43.
- Kowalski, R. A. [2014]. *Logic for Problem Solving, Revisited*. Books on Demand.
- Kramár, J., et al. [2022]. Negotiation and honesty in artificial intelligence methods for the board game of diplomacy. *Nature Communications*, 13(1):7214. <http://dx.doi.org/10.1038/s41467-022-34473-5>.
- Krizhevsky, A., Sutskever, I., and Hinton, G. [2012]. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25*, pp. 1090–1098.
- Krötzsch, M. [2012]. OWL 2 Profiles: An introduction to lightweight ontology languages. In Eiter, T. and Krennwallner, T. (eds.), *8th ReasoningWeb Summer School, Vienna, Austria*, pp. 112–183. Springer. http://korrekt.org/page/OWL_2_Profiles.
- Kuppe, M. A., Lamport, L., and Ricketts, D. [2019]. The TLA+ toolbox. *Electronic Proceedings in Theoretical Computer Science*, 310:50–62. <http://dx.doi.org/10.4204/ep-tcs.310.6>.
- Lacroix, T., Usunier, N., and Obozinski, G. [2018]. Canonical tensor decomposition for knowledge base completion. In *35th International Conference on Machine Learning (ICML)*.
- Lakshmanan, V., Gørner, M., and Gillard, R. [2021]. *Practical Machine Learning for Computer Vision: End-to-End Machine Learning for Images*. O'Reilly.
- Lally, A., et al. [2012]. Question analysis: How Watson reads a clue. *IBM Journal of Research and Development*, 56(3/4).
- Lamport, L. [2002]. *Specifying Systems: The TLA+ Language and Tools for Hardware and Software Engineers*. Addison-Wesley Longman.
- Langley, P., Iba, W., and Thompson, K. [1992]. An analysis of Bayesian classifiers. In *10th National Conference on Artificial Intelligence*, pp. 223–228.
- Langton, C. G. [1997]. *Artificial Life: An Overview*. MIT Press.
- Laplace, P. [1812]. *Théorie Analytique de Probabilités*. Courcier.
- Latombe, J.-C. [1991]. *Robot Motion Planning*. Kluwer Academic.
- Lawler, E. L. and Wood, D. E. [1966]. Branch-and-bound methods: A survey. *Operations Research*, 14(4):699–719.
- LeCun, Y., Bengio, Y., and Hinton, G. [2015]. Deep learning. *Nature*, 521(7553):436–444.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. [1998a]. Gradient-based learning applied to document recognition. *IEEE*, 86(11):2278–2324. <http://dx.doi.org/10.1109/5.726791>.

LeCun, Y., Bottou, L., Orr, G., and Muller, K. [1998b]. Efficient backprop. In Orr, G. and Muller, K.-R. (eds.), *Neural Networks: Tricks of the Trade*. Springer.<http://yann.lecun.com/exdb/publis/pdf/lecun-98b.pdf>.

Lehman, J., Clune, J., Misevic, D., et al. [2018]. The surprising creativity of digital evolution: A collection of anecdotes from the evolutionary computation and artificial life research communities. *CoRR*. <http://arxiv.org/abs/1803.03453>.

Leibniz, G. W. [1677]. *The Method of Mathematics: Preface to the General Science*. Selections reprinted by Chrisley and Begeer [2000].

Leibniz, G. W. [1705]. *New Essays on Human Understanding*, Book 3. www.early-moderntexts.com.

Lenat, D. B. and Feigenbaum, E. A. [1991]. On the thresholds of knowledge. *Artificial Intelligence*, 47:185–250.

Lepikhin, D., et al. [2021]. GShard: Scaling giant models with conditional computation and automatic sharding. In *International Conference on Learning Representations*. <https://openreview.net/pdf?id=qrwe7XHTmYb>.

Lertvittayakumjorn, P. and Toni, F. [2021]. Explanation-based human debugging of NLP models: A survey. *Transactions of the Association for Computational Linguistics*, 9:1508–1528. <http://dx.doi.org/10.1162/tacl.2020.00440>.

Levesque, H. J. [1984]. Foundations of a functional approach to knowledge representation. *Artificial Intelligence*, 23(2):155–212.

Levesque, H. J. [2012]. *Thinking as Computation*. MIT Press.

Levesque, H. J. [2014]. On our best behaviour. *Artificial Intelligence*, 212:27–35.

Levy, R. [2021]. Social media, news consumption, and polarization: Evidence from a field experiment. *American Economic Review*, 111(3):831–70. <http://dx.doi.org/10.1257/aer.20191777>.

Leyton-Brown, K., Milgrom, P. R., and Segal, I. [2017]. Economics and computer science of a radio spectrum reallocation. *National Academy of Sciences*, 114:7202–7209.

Leyton-Brown, K. and Shoham, Y. [2008]. *Essentials of Game Theory*. Morgan & Claypool.

Li, Y. [2018]. Deep reinforcement learning. *CoRR*, abs/1810.06339. <http://arxiv.org/abs/1810.06339>.

Li, Y., et al. [2016]. A survey on truth discovery. *SIGKDD Explorations Newsletter*, 17(2):1–16. <http://dx.doi.org/10.1145/2897350.2897352>.

Liao, T., Taori, R., Raji, D., and Schmidt, L. [2021]. Are we learning yet? A meta review of evaluation failures across machine learning. In *Neural*

Information Processing Systems (NeurIPS) Track on Datasets and Benchmarks. <https://datasets-benchmarks-proceedings.neurips.cc/paper/2021/hash/>

757b505cfd34c64c85ca5b5690ee5293-Abstract-round2.html.

Lin, A. Y., Kuehl, K., Schöning, J., and Hecht, B. [2017]. Understanding “death by GPS”: A systematic study of catastrophic incidents associated with personal navigation technologies. In *CHI Conference on Human Factors in Computing Systems*. <http://dx.doi.org/10.1145/3025453.3025737>.

Lindholm, T., et al. [2022]. *The Java Virtual Machine Specification: Java SE 19 Edition*.

Oracle America, Inc. <https://docs.oracle.com/javase/specs/jvms/se19/jvms19.pdf>.

Little, R. J. A. and Rubin, D. B. [1987]. *Statistical Analysis with Missing Data*. Wiley.

Littman, M. L., et al. [2021]. *Gathering Strength, Gathering Storms: The One Hundred Year Study on Artificial Intelligence (AI100) 2021 Study Panel Report*. Stanford University. <http://ai100.stanford.edu/2021-report>.

Liu, A. L., et al. [2006]. Indoor wayfinding: Developing a functional interface for individuals with cognitive impairments. In *8th International ACM SIGACCESS Conference on Computers and Accessibility*.

Lloyd, J. W. [1987]. *Foundations of Logic Programming*. Symbolic Computation Series. Springer-Verlag, 2nd edition.

Lloyd, S. [1982]. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–137. <http://dx.doi.org/10.1109/TIT.1982.1056489>.

Lopez, A. and Bacchus, F. [2003]. Generalizing GraphPlan by formulating planning as a CSP. In *18th International Joint Conference Artificial Intelligence (IJCAI)*, pp. 954–960.

Luenberger, D. G. [1979]. *Introduction to Dynamic Systems: Theory, Models and Applications*. Wiley.

Lum, K. and Isaac, W. [2016]. To predict and serve? *Significance*, 13(5).

Lundgren, B. [2023]. In defense of ethical guidelines. *AI and Ethics*. <http://dx.doi.org/10.1007/s43681-022-00244-7>.

Ma, Y., et al. [2022]. Identification of antimicrobial peptides from the human gut microbiome using deep learning. *Nature Biotechnology*, 40(6):921–931. <http://dx.doi.org/10.1038/s41587-022-01226-0>.

MacKay, D. [2003]. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press.

Mackworth, A. K. [1977a]. Consistency in networks of relations. *Artificial Intelligence*, 8:99–118.

Mackworth, A. K. [1977b]. On reading sketch maps. In *Fifth International Joint Conference on Artificial Intelligence*, pp. 598–606.

Mackworth, A. K. [1993]. On seeing robots. In Basu, A. and Li, X. (eds.), *Computer Vision: Systems, Theory, and Applications*, pp. 1–13. World Scientific Press.

Mackworth, A. K. [2009]. Agents, bodies, constraints, dynamics and evolution. *AI Magazine*.

Mackworth, A. K. [2011]. Architectures and ethics for robots: Constraint satisfaction as a unitary design framework. In Anderson, M. and Anderson, S. L. (eds.), *Machine Ethics*, pp. 335–360. Cambridge University Press. <http://dx.doi.org/10.1017/CBO9780511978036.024>.

Mackworth, A. K. and Zhang, Y. [2003]. A formal approach to agent design: An overview of constraint-based agents. *Constraints*, 8(3):229–242.

Mackworth, J. F. [1970]. *Vigilance and Attention: A Signal Detection Approach*. Penguin.

Mackworth, N. H. [1948]. The breakdown of vigilance during prolonged visual search. *Quarterly Journal of Experimental Psychology*, 1(1):6–21. <http://dx.doi.org/10.1080/17470214808416738>.

Mahdisoltani, F., Biega, J., and Suchanek, F. M. [2015]. YAGO3: A knowledge base from multilingual wikipedias. In *Conference on Innovative Data Systems Research (CIDR 2015)*. <http://suchanek.name/work/publications/cidr2015.pdf>.

Malthus, T. R. [1798]. *An Essay on the Principle of Population: As it Affects the Future Improvement of Society*. J. Johnson.

Manning, C. and Schütze, H. [1999]. *Foundations of Statistical Natural Language Processing*. MIT Press.

Marcus, G. and Davis, E. [2019]. *Rebooting AI: Building Artificial Intelligence We Can Trust*. Pantheon.

Marlin, B. M., Zemel, R. S., Roweis, S. T., and Slaney, M. [2011]. Recommender systems, missing data and statistical model estimation. In *22nd International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 2686–2691.

Matheson, J. E. [1990]. Using influence diagrams to value information and control. In Oliver, R. M. and Smith, J. Q. (eds.), *Influence Diagrams, Belief Nets and Decision Analysis*, chapter 1, pp. 25–48. Wiley.

Mausam and Kolobov, A. [2012]. *Planning with Markov Decision Processes: An AI Perspective*. Morgan & Claypool. <http://dx.doi.org/doi:10.2200/S00426ED1V-01Y201206AIM017>.

McAllester, D. and Rosenblitt, D. [1991]. Systematic nonlinear planning. In *9th National Conference on Artificial Intelligence*, pp. 634–639.

McCarthy, J. [1958]. Programs with common sense. In *Teddington Conference on the Mechanization of Thought Processes*. <http://jmc.stanford.edu/articles/mcc59/mcc59.pdf>.

McCarthy, J. [1986]. Applications of circumscription to formalizing commonsense knowledge. *Artificial Intelligence*, 28(1):89–116.

McCarthy, J. and Hayes, P. J. [1969]. Some philosophical problems from the standpoint of artificial intelligence. In Meltzer, M. and Michie, D. (eds.), *Machine Intelligence 4*, pp. 463–502. Edinburgh University Press.

McCulloch, W. and Pitts, W. [1943]. A logical calculus of ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5:115–133.

McDermott, D. and Hendler, J. [1995]. Planning: What it is, what it could be, an introduction to the special issue on planning and scheduling. *Artificial Intelligence*, 76:1–16.

McElreath, R. [2020]. *Statistical Rethinking: A Bayesian Course with Examples in R and STAN*. Chapman & Hall. <https://xcelab.net/rm/statistical-rethinking/>.

McFadden, D. L. [2000]. Prize lecture. *The Sveriges Riksbank Prize in Economic Sciences in Memory of Alfred Nobel 2000*. <https://www.nobelprize.org/prizes/economic-sciences/2000/mcfadden/lecture/>.

McGuffie, K. and Newhouse, A. [2020]. The radicalization risks of GPT-3 and advanced neural language model. Technical report, Center on Terrorism, Extremism, and Counterterrorism, Middlebury Institute of International Studies at Monterrey. <https://www.middlebury.edu/institute/sites/www.middlebury.edu.institute/files/2020-09/gpt3-article.pdf>.

McLuhan, M. [1962]. *The Gutenberg Galaxy: The Making of Typographic Man*. University of Toronto Press.

Meir, R. and R"atsch, G. [2003]. An introduction to boosting and leveraging. In *Advanced Lectures on Machine Learning*, pp. 119–184. Springer.

Michie, D. [1963]. Experiments on the mechanisation of game learning. 1. Characterization of the model and its parameters. *Computer Journal*, 1:232–263.

Michie, D., Spiegelhalter, D. J., and Taylor, C. C. (eds.) [1994]. *Machine Learning, Neural and Statistical Classification*. Series in Artificial Intelligence. Ellis Horwood.

Mihailidis, A., Boger, J., Candido, M., and Hoey, J. [2007]. The use of an intelligent prompting system for people with dementia. *ACM Interactions*, 14(4):34–37.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. [2013]. Efficient estimation of word representations in vector space. <http://dx.doi.org/10.48550/arXiv.1301.3781>.

Milch, B., et al. [2005]. BLOG: Probabilistic models with unknown objects. In *19th International Joint Conference Artificial Intelligence (IJCAI-05)*.

Minaee, S., et al. [2021]. Deep learning-based text classification: A comprehensive review. *ACM Computing Surveys*, 54(3). <http://dx.doi.org/10.1145/3439726>.

Minsky, M. L. [1952]. A neural-analogue calculator based upon a probability model of reinforcement. Technical report, Harvard University Psychological Laboratories.

Minsky, M. L. [1961]. Steps towards artificial intelligence. *IEEE*, 49:8–30. <http://web.media.mit.edu/~minsky/papers/steps.html>.

Minsky, M. L. [1975]. A framework for representing knowledge. In Winston, P. (ed.), *The Psychology of Computer Vision*, pp. 211–277. McGraw-Hill. Alternative version is in Haugeland [1997].

Minsky, M. L. [1986]. *The Society of Mind*. Simon & Schuster.

Minsky, M. L. and Papert, S. [1988]. *Perceptrons: An Introduction to Computational Geometry*. MIT Press, expanded edition.

- Minton, S., Johnston, M. D., Philips, A. B., and Laird, P. [1992]. Minimizing conflicts: A heuristic repair method for constraint satisfaction and scheduling problems. *Artificial Intelligence*, 58(1–3):161–205.
- Mitchell, M. [1996]. *An Introduction to Genetic Algorithms*. MIT Press.
- Mitchell, T. [1997]. *Machine Learning*. McGraw-Hill. <http://www.cs.cmu.edu/afs/cs.cmu.edu/user/mitchell/ftp/mlbook.html>.
- Mnih, V. et al. [2015]. Human-level control through deep reinforcement learning. *Nature*, 518:529–533. <http://www.nature.com/nature/journal/v518/n7540/abs/nature14236.html>.
- Mohan, K. [2022]. Causal graphs for missing data: A gentle introduction. In *Probabilistic and Causal Inference: The Works of Judea Pearl*. ACM Books.
- Mohan, K., Pearl, J., and Tian, J. [2013]. Graphical models for inference with missing data. In *Advances in Neural Information Processing Systems*, volume 26, pp. 1277–1285.
- Mole, C. [2010]. *Attention Is Cognitive Unison: An Essay in Philosophical Psychology*. Oxford University Press. <http://dx.doi.org/10.1093/acprof:oso/9780195384529.001.0001>.
- Moore, E. F. [1959]. The shortest path through a maze. In *International Symposium on the Theory of Switching*, pp. 285–292. Harvard University Press.
- Moore, T. J., Heyward, J., Anderson, G., and Alexander, G. C. [2020]. Variation in the estimated costs of pivotal clinical benefit trials supporting the US approval of new therapeutic agents, 2015–2017: A cross-sectional study. *BMJ Open*, 10(6). <http://dx.doi.org/10.1136/bmjopen-2020-038863>.
- Morin, F. and Bengio, Y. [2005]. Hierarchical probabilistic neural network language model. In *international Workshop on Artificial Intelligence and Statistics*, pp. 246–252.
- Motik, B., Patel-Schneider, P. F., and Grau, B. C. (eds.) [2012]. *OWL 2 Web Ontology Language: Direct Semantics*. W3C Recommendation 11 December 2012, 2nd edition. <http://www.w3.org/TR/owl2-direct-semantics/>.
- Munn, L. [2022]. The uselessness of AI ethics. *AI and Ethics*. <https://doi.org/10.1007/s43681-022-00209-w>.
- Murphy, K. P. [2022]. *Probabilistic Machine Learning: An Introduction*. MIT Press. <https://probml.github.io/pml-book/book1.html>.
- Murphy, K. P. [2023]. *Probabilistic Machine Learning: Advanced Topics*. MIT Press. <https://probml.github.io/book2>.
- Muscettola, N., Nayak, P., Pell, B., and Williams, B. [1998]. Remote agent: To boldly go where no AI system has gone before. *Artificial Intelligence*, 103:5–47.
- NASA [2022]. EarthData: Open access for open science. <https://www.earthdata.nasa.gov>.
- Nash, J. F. [1950]. Equilibrium points in N-person games. *National Academy of Sciences of the United States of America*, 36:48–49.
- Nau, D. S. [2007]. Current trends in automated planning. *AI Magazine*, 28(4):43–58.

Neufeld, X., Mostaghim, S., Sancho-Pradel, D. L., and Brand, S. [2019]. Building a planner: A survey of planning systems used in commercial video games. *IEEE Transactions on Games*, 11(2):91–108. <http://dx.doi.org/10.1109/TG.2017.2782846>.

Neumann, J. V. and Morgenstern, O. [1953]. *Theory of Games and Economic Behavior*. Princeton University Press, 3rd edition.

Neville, J. and Jensen, D. [2007]. Relational dependency networks. *Journal of Machine Learning Research (JMLR)*, 8:653–692.

New York Times [1958]. New Navy device learns by doing: Psychologist shows embryo of computer designed to read and grow wiser. <https://timesmachine.nytimes.com/timesmachine/1958/07/08/83417341.html?pageNumber=25>.

Newell, A. and Simon, H. A. [1956]. The logic theory machine: A complex information processing system. Technical Report P-868, The Rand Corporation. <http://shelf1.library.cmu.edu/IMLS/MindModels/logictheorymachine.pdf>.

Newell, A. and Simon, H. A. [1976]. Computer science as empirical enquiry: Symbols and search. *Communications of the ACM*, 19:113–126.

Ng, A. [2018]. *Machine Learning Yearning*. deeplearning.ai. <https://www.deeplearning.ai/resources/>.

Ng, A. Y. [2004]. Feature selection, L1 vs. L2 regularization, and rotational invariance. In *Twenty-First International Conference on Machine Learning*.

Ng, A. Y. and Russell, S. J. [2000]. Algorithms for inverse reinforcement learning. In *International Conference on Machine Learning (ICML)*, pp. 663–670.

Niles, I. and Pease, A. [2001]. Towards a standard upper ontology. In Welty, C. and Smith, B. (eds.), *2nd International Conference on Formal Ontology in Information Systems (FOIS-2001)*.

Nilsson, N. J. [2007]. The physical symbol system hypothesis: Status and prospects. In Lungarella, M. et al. (eds.), *50 Years of AI, Festschrift*, pp. 9–17.

Springer. <http://ai.stanford.edu/~nilsson/OnlinePubs-Nils/PublishedPapers/pssh.pdf>.

Nilsson, N. J. [2010]. *The Quest for Artificial Intelligence: A History of Ideas and Achievements*. Cambridge University Press.

Nisan, N. [2007]. Introduction to mechanism design (for computer scientists).

In Nisan, N. et al. (eds.), *Algorithmic Game Theory*, chapter 9, pp. 209–242. Cambridge University Press.

Noble, S. U. [2018]. *Algorithms of Oppression: How Search Engines Reinforce Racism*. NYU Press.

Nocedal, J. and Wright, S. [2006]. *Numerical Optimization*. Springer-Verlag.

Nyholm, S. [2021]. The ethics of human-robot interaction and traditional moral theories. In *The Oxford Handbook of Digital Ethics*. Oxford University Press. <http://dx.doi.org/10.1093/oxfordhb/9780198857815.013.3>.

Obermeyer, Z., Powers, B., Vogeli, C., and Mullainathan, S. [2019]. Dissecting racial bias in an algorithm used to manage the health of populations. *Science*, 366(6464):447–453.

OECD [2019]. OECD AI principles. <https://oecd.ai/en/ai-principles>.

Office of Science and Technology Policy [2022]. The blueprint for an AI bill of rights. <https://www.whitehouse.gov/ostp/ai-bill-of-rights/>.

Olah, C. [2015]. Understanding LSTM networks. <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>.

O’Neil, C. [2016]. *Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy*. Crown.

OpenAI [2022]. ChatGPT: Optimizing language models for dialogue. <https://openai.com/blog/chatgpt/>.

OpenAI [2023]. GPT-4 technical report. *ArXiv e-prints*, arXiv:2303.08774.

Ordeshook, P. C. [1986]. *Game Theory and Political Theory: An Introduction*. Cambridge University Press.

Orkin, J. [2006]. Three states and a plan: The AI of F.E.A.R. In *Game Developers Conference*.

Ostrom, E. [1990]. *Governing the Commons: The Evolution of Institutions for Collective Action*. Cambridge University Press.

Page, L., Brin, S., Motwani, R., and Winograd, T. [1999]. The PageRank citation ranking: Bringing order to the Web. Technical Report SIDL-WP-1999-0120, Stanford InfoLab.

Panton, K., et al. [2006]. Common sense reasoning – from Cyc to intelligent assistant. In Cai, Y. and Abascal, J. (eds.), *Ambient Intelligence in Everyday Life*, pp. 1–31, Springer.

Pasula, H., et al. [2003]. Identity uncertainty and citation matching. In *Advances in Neural Information Processing Systems*, volume 15.

Pearl, J. [1984]. *Heuristics*. Addison-Wesley.

Pearl, J. [1988]. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann.

Pearl, J. [2009]. *Causality: Models, Reasoning and Inference*. Cambridge University Press, 2nd edition.

Pearl, J. and Mackenzie, D. [2018]. *The Book of Why: The New Science of Cause and Effect*. Basic Books.

Pease, A. [2011]. *Ontology: A Practical Guide*. Articulate Software Press.

Peden, M. et al. (eds.) [2004]. *World Report on Road Traffic Injury Prevention*. World Health Organization.

Pereira, F. C. N. and Shieber, S. M. [2002]. *Prolog and Natural-Language Analysis*. Microtome Publishing.

Perrault, A., Fang, F., Sinha, A., and Tambe, M. [2020]. Artificial intelligence for social impact: Learning and planning in the data-to-deployment pipeline. *AI Magazine*, 41(4):3–16. <http://dx.doi.org/10.1609/aimag.v41i4.5296>.

Peters, M. E., et al. [2018]. Deep contextualized word representations. In *2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. <http://dx.doi.org/10.18653/v1/N18-1202>.

Phuong, M. and Hutter, M. [2022]. Formal algorithms for transformers. <http://dx.doi.org/10.48550/arXiv.2207.09238>.

Piaget, J. [1953]. *The Origin of Intelligence in the Child*. Routledge & Kegan Paul.

Pinker, S. [1997]. *How the Mind Works*. Norton.

Pohl, I. [1971]. Bi-directional search. *Machine Intelligence*, 6(127–140).

Pollack, M. E. [2005]. Intelligent technology for an aging population: The use of AI to assist elders with cognitive impairment. *AI Magazine*, 26(2):9–24.

Poole, D. [1993]. Probabilistic Horn abduction and Bayesian networks. *Artificial Intelligence*, 64(1):81–129.

Poole, D. [2007]. Logical generative models for probabilistic reasoning about existence, roles and identity. In *22nd AAAI Conference on AI (AAAI-07)*. <http://cs.ubc.ca/~poole/papers/AAAI07-Poole.pdf>.

Poole, D., Goebel, R., and Aleliunas, R. [1987]. Theorist: A logical reasoning system for defaults and diagnosis. In Cercone, N. and McCalla, G. (eds.),

The Knowledge Frontier: Essays in the Representation of Knowledge, pp. 331–352. Springer-Verlag.

Posner, M. I. (ed.) [1989]. *Foundations of Cognitive Science*. MIT Press.

Powell, W. B. [2022]. *Reinforcement Learning and Stochastic Optimization: A Unified Framework for Sequential Decisions*. Wiley. <https://castlelab.princeton.edu/RLSO/>.

Prabhu, V. U. and Birhane, A. [2020]. Large image datasets: A pyrrhic win for computer vision? <http://dx.doi.org/10.48550/arXiv.2006.16923>.

Pujara, J., Miao, H., Getoor, L., and Cohen, W. W. [2015]. Using semantics and statistics to turn data into knowledge. *AI Magazine*, 36(1):65–74. <http://dx.doi.org/10.1609/aimag.v36i1.2568>.

Puterman, M. [1994]. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley.

Qian, K., et al. [2021]. XNLP: A living survey for XAI research in natural language processing. In *26th International Conference on Intelligent User Interfaces*, pp. 78–80. <http://dx.doi.org/10.1145/3397482.3450728>.

Qiu, X., et al. [2020]. Pre-trained models for natural language processing: A survey. *CoRR*, abs/2003.08271. <https://arxiv.org/abs/2003.08271>.

Quinlan, J. R. [1993]. *C4.5 Programs for Machine Learning*. Morgan Kaufmann.

Rabiner, L. [1989]. A tutorial on hidden Markov models and selected applications in speech recognition. *IEEE*, 77(2):257–286.

Rae, J. W., et al. [2021]. Scaling language models: Methods, analysis & insights from training gopher. *CoRR*, abs/2112.11446. <https://arxiv.org/abs/2112.11446>.

Rakova, B., Yang, J., Cramer, H., and Chowdhury, R. [2021]. Where responsible AI meets reality: Practitioner perspectives on enablers for shifting organizational practices. *Proceedings of the ACM on Human-Computer Interaction*, 5. <http://dx.doi.org/10.1145/3449081>.

Randell, B. [1982]. From analytical engine to electronic digital computer: The contributions of ludgate, torres, and bush. *Annals of the History of Computing*, 4(4).

Real, E., Liang, C., So, D. R., and Le, Q. V. [2020]. AutoML-Zero: Evolving machine learning algorithms from scratch. In *37th International Conference on Machine Learning*. <http://dx.doi.org/10.48550/arXiv.2003.03384>.

Richtel, M. [2014]. *A DeadlyWandering: A Mystery, a Landmark Investigation, and the Astonishing Science of Attention in the Digital Age*. HarperCollins.

Roberts, L. [1965]. *Machine Perception of 3-D Solids*. MIT Press.

Robillard, M. [2021]. The ethics of weaponized AI. In *The Oxford Handbook of Digital Ethics*. Oxford University Press. <http://dx.doi.org/10.1093/oxfordhb/9780198857815.013.29>.

Robinson, J. A. [1965]. A machine-oriented logic based on the resolution principle. *Journal ACM*, 12(1):23–41.

Rockström, J., et al. [2009]. A safe operating space for humanity. *Nature*, 461(7263):472–475.

Rogers, Y., Sharp, H., and Preece, J. [2023]. *Interaction Design: Beyond Human-Computer Interaction*. Wiley, 6th edition.

Rosenblatt, F. [1958]. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408.

Rosenschein, S. J. and Kaelbling, L. P. [1995]. A situated view of representation and control. *Artificial Intelligence*, 73:149–173.

Rossi, F., Venable, K. B., and Walsh, T. [2011]. *A Short Introduction to Preferences: Between Artificial Intelligence and Social Choice*. Morgan & Claypool. <http://dx.doi.org/doi:10.2200/S00372ED1V01Y201107AIM014>.

Rubin, D. B. [1976]. Inference and missing data. *Biometrika*, 63(3):581–592.

Rubinstein, R. Y. [1981]. *Simulation and the Monte Carlo Method*. Wiley.

Ruder, S. [2016]. An overview of gradient descent optimization algorithms. *CoRR*, abs/1609.04747. <http://arxiv.org/abs/1609.04747>.

Rudin, C., et al. [2022]. Interpretable machine learning: Fundamental principles and 10 grand challenges. *Statistics Surveys*, 16:1–85. <http://dx.doi.org/10.1214/21-SS133>.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. [1986]. Learning internal representations by error propagation. In Rumelhart, D. E. and McClelland,

J. L. (eds.), *Parallel Distributed Processing*, chapter 8, pp. 318–362. MIT Press.

- Russakovsky, O., et al. [2014]. Imagenet large scale visual recognition challenge. *CoRR*, abs/1409.0575. <http://arxiv.org/abs/1409.0575>.
- Russell, S. [1997]. Rationality and intelligence. *Artificial Intelligence*, 94:57–77.
- Russell, S. [2019]. *Human Compatible: AI and the Problem of Control*. Penguin Books Limited.
- Russell, S. and Norvig, P. [2020]. *Artificial Intelligence: A Modern Approach (4th Edition)*. Pearson. <http://aima.cs.berkeley.edu/>.
- Russo, D., et al. [2018]. A tutorial on Thompson sampling. *Foundations and Trends in Machine Learning*, 11(1):1–96. <http://dx.doi.org/10.48550/arXiv.1707.02038>.
- Sacerdoti, E. D. [1975]. The nonlinear nature of plans. In *4th International Joint Conference on Artificial Intelligence*, pp. 206–214.
- Salimans, T., et al. [2017]. Evolution strategies as a scalable alternative to reinforcement learning. <http://dx.doi.org/10.48550/arXiv.1703.03864>.
- Samuel, A. L. [1959]. Some studies in machine learning using the game of checkers. *IBM Journal on Research and Development*, 3(3):210–229.
- Sandholm, T. [2007]. Expressive commerce and its application to sourcing: How we conducted \$35 billion of generalized combinatorial auctions. *AI Magazine*, 28(3):45–58.
- Satterthwaite, M. [1975]. Strategy-proofness and Arrow's conditions: Existence and correspondence theorems for voting procedures and social welfare functions. *Journal of Economic Theory*, 10:187–217.
- Savage, L. J. [1972]. *The Foundation of Statistics*. Dover, 2nd edition.
- Schank, R. C. [1990]. What is AI, anyway? In Partridge, D. and Wilks, Y. (eds.), *The Foundations of Artificial Intelligence*, pp. 3–13. Cambridge University Press.
- Schapire, R. E. [2002]. The boosting approach to machine learning: An overview. In *MSRI Workshop on Nonlinear Estimation and Classification*. Springer-Verlag.
- Schlichtkrull, M., et al. [2018]. Modeling relational data with graph convolutional networks. In *European Semantic Web Conference (ESWC 2018)*, pp. 593–607. Springer. <https://arxiv.org/abs/1703.06103>.
- Schmidhuber, J. [1990]. Making the world differentiable: On using fully recurrent self-supervised neural networks for dynamic reinforcement learning and planning in non-stationary environments. Technical Report FKI-126-90, T.U. Munich.
- Schmidhuber, J. [2015]. Deep learning in neural networks: An overview. *Neural Networks*, 61:85–117. <http://dx.doi.org/10.1016/j.neunet.2014.09.003>.
- Schubert, E. [2022]. Stop using the elbow criterion for k-means and how to choose the number of clusters instead. <http://dx.doi.org/10.48550/arXiv.2212.12189>.
- Schwarz, G. [1978]. Estimating the dimension of a model. *Annals of Statistics*, 6(2):461–464. <https://projecteuclid.org/euclid.aos/1176344136>.
- Seger, C.-J. H. [2021]. Formal verification of complex data paths: An industrial experience. In *FM*.

Selinger, E. and Leong, B. [2021]. The ethics of facial recognition technology. In *The Oxford Handbook of Digital Ethics*. Oxford University Press. <http://dx.doi.org/10.1093/oxfordhb/9780198857815.013.32>.

Senior, A. W., et al. [2020]. Improved protein structure prediction using potentials from deep learning. *Nature*, 577(7792):706–710. <http://dx.doi.org/10.1038/s41586-019-1923-7>.

Settles, B. [2012]. *Active Learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool. <http://dx.doi.org/doi:10.2200/S00429ED1V-01Y201207AIM018>.

Shachter, R. and Peot, M. A. [1992]. Decision making using probabilistic inference methods. In *Eighth Conference on Uncertainty in Artificial Intelligence (UAI-92)*, pp. 276–283.

Shahriari, B., et al. [2016]. Taking the human out of the loop: A review of Bayesian optimization. *IEEE*, 104(1):148–175. <http://dx.doi.org/10.1109/JPROC.2015.2494218>.

Shanahan, M. [2022]. Talking about large language models. <http://dx.doi.org/10.48550/arXiv.2212.03551>.

Shannon, C. E. and Weaver, W. [1949]. *The Mathematical Theory of Communication*.

Sharkey, N. [2008]. The ethical frontiers of robotics. *Science*, 322(5909):1800–1801. DOI:10.1126/science.1164582.

Shavlik, J. W. and Dietterich, T. G. (eds.) [1990]. *Readings in Machine Learning*. Morgan Kaufmann.

Shelley, M. W. [1818]. *Frankenstein; or, The Modern Prometheus*. Lackington, Hughes, Harding, Mavor & Jones.

Shneiderman, B. [2022]. *Human-Centered AI*. Oxford University Press.

Shoeybi, M., et al. [2019]. Megatron-LM: Training multi-billion parameter language models using model parallelism. *CoRR*, abs/1909.08053. <http://arxiv.org/abs/1909.08053>.

Shoham, Y. [2016]. Why knowledge representation matters. *Communications of the ACM*, 59(1):47–49.

Shoham, Y. and Leyton-Brown, K. [2008]. *Multiagent Systems: Algorithmic, Game Theoretic, and Logical Foundations*. Cambridge University Press.

Shpitser, I. and Pearl, J. [2008]. Complete identification methods for the causal hierarchy. *Journal of Machine Learning Research*, 9:1941–1979.

Sikos, L., Seneviratne, O., and McGuinness, D. L. [2021]. *Provenance in Data Science: From Data Models to Context-Aware Knowledge Graphs*. Springer. <https://www.springer.com/gp/book/9783030676803>.

Silver, D., Singh, S., Precup, D., and Sutton, R. S. [2021]. Reward is enough. *Artificial Intelligence*, 299. <http://dx.doi.org/https://doi.org/10.1016/j.artint.2021.103535>.

Silver, D., et al. [2016]. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587):484–489.

Silver, D., et al. [2017]. Mastering chess and shogi by self-play with a general reinforcement learning algorithm. *CoRR*, abs/1712.01815. <http://arxiv.org/abs/1712.01815>.

Simon, H. A. [1971]. Designing organizations for an information rich world. In Greenberger, M. (ed.), *Computers, Communications, and the Public Interest*, pp. 37–72. Johns Hopkins Press.

Simon, H. A. [1995]. Artificial intelligence: An empirical science. *Artificial Intelligence*, 77(1):95–127.

Simon, H. A. [1996]. *The Sciences of the Artificial*. MIT Press, 3rd edition.

Singer, P. W. [2009a]. Robots at war: The new battlefield. *The Wilson Quarterly*.

Singer, P. W. [2009b]. *Wired for War: The Robotics Revolution and Conflict in the 21st Century*. Penguin.

Sinz, C., Kaiser, A., and Kuchlin, W. [2003]. Formal methods for the validation of automotive product configuration data. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 17:75 – 97.

Siu, H. C., et al. [2021]. Evaluation of human-AI teams for learned and rulebased agents in Hanabi. In *Neural Information Processing Systems (NeurIPS)*. <http://dx.doi.org/10.48550/arXiv.2107.07630>.

Smith, B. [2003]. Ontology. In Floridi, L. (ed.), *Blackwell Guide to the Philosophy of Computing and Information*, pp. 155–166. Blackwell. <http://ontology.buffalo.edu/smith/articles/ontologies.htm>.

Smith, B. [2015]. Basic formal ontology 2.0: Specification and user's guide. Technical report, Institute for Formal Ontology and Medical Information Science (IFOMIS). <https://github.com/bfo-ontology/BFO/wiki>.

Smith, B. C. [1996]. *On the Origin of Objects*. MIT Press.

Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., and Ganguli, S. [2015].

Deep unsupervised learning using nonequilibrium thermodynamics. In *32nd International Conference on Machine Learning*, pp. 2256–2265. <https://proceedings.mlr.press/v37/sohl-dickstein15.html>.

Sowa, J. F. [2000]. *Knowledge Representation: Logical, Philosophical, and Computational Foundations*. Brooks Cole.

Sowa, J. F. [2011]. Future directions for semantic systems. In Tolk, A. and Jain, L. C. (eds.), *Intelligence-Based Software Engineering*, pp. 23–47. Springer–Verlag. <http://www.jfsowa.com/pubs/futures.pdf>.

Spall, J. C. [2003]. *Introduction to Stochastic Search and Optimization: Estimation, Simulation*. Wiley.

Sparkes, A., et al. [2010]. Towards robot scientists for autonomous scientific discovery. *Automated Experimentation*, 2(1):1. <http://dx.doi.org/10.1186/1759-4499-2-1>.

Spencer, A., et al. [2022]. The QALY at 50: One story many voices. *Social Science and Medicine*, 296:114653. <http://dx.doi.org/https://doi.org/10.1016/j.socscimed.2021.114653>.

Spiegelhalter, D. J., Franklin, R. C. G., and Bull, K. [1990]. Assessment, criticism and improvement of imprecise subjective probabilities for a medical expert system. In Henrion, M., Shachter, R. D., Kanal, L., and Lemmer, J. (eds.), *Uncertainty in Artificial Intelligence 5*, pp. 285–294. North-Holland.

Spirtes, P., Glymour, C., and Scheines, R. [2001]. *Causation, Prediction, and Search*. MIT Press, 2nd edition.

Springer Nature [2022]. SN SciGraph: A linked open data platform for the scholarly domain. <https://www.springernature.com/gp/researchers/scigraph>.

Sreedharan, S., Kulkarni, A., and Kambhampati, S. [2022]. *Explainable Human-AI Interaction: A Planning Perspective*. Morgan & Claypool. <https://doi.org/10.2200/S01152ED1V01Y202111AIM050>.

Srivastava, A. et al. [2022]. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. <http://dx.doi.org/10.48550/arXiv.2206.04615>.

Stanley, K. O., Clune, J., Lehman, J., and Miikkulainen, R. [2019]. Designing neural networks through neuroevolution. *Nature Machine Intelligence*, 1(1):24–35. <http://dx.doi.org/10.1038/s42256-018-0006-z>.

Steck, H., et al. [2021]. Deep learning for recommender systems: A netflix case study. *AI Magazine*, 42(3):7–18. <http://dx.doi.org/10.1609/aimag.v42i3.18140>.

Sterling, L. S. and Shapiro, E. Y. [1994]. *The Art of Prolog: Advanced Programming Techniques*. MIT Press, 2nd edition.

Stevenson, A. and Lindberg, C. A. (eds.) [2010]. *The New Oxford American Dictionary*. Oxford University Press.

Stillings, N. A., et al. [1987]. *Cognitive Science: An Introduction*. MIT Press.

Stodden, V., et al. [2016]. Enhancing reproducibility for computational methods. *Science*, 354. <http://dx.doi.org/10.1126/science.aah6168>.

Stone, P. [2007]. Learning and multiagent reasoning for autonomous agents. In *The 20th International Joint Conference on Artificial Intelligence (IJCAI-07)*, pp. 13–30. <http://www.cs.utexas.edu/~pstone/Papers/bib2html-links/IJCAI07-award.pdf>.

Stone, P. and Veloso, M. [2000]. Multiagent systems: A survey from a machine learning perspective. *Autonomous Robots*, 8:345–383.

Such, F. P., et al. [2017]. Deep neuroevolution: Genetic algorithms are a competitive alternative for training deep neural networks for reinforcement learning. *CoRR*, abs/1712.06567. <http://arxiv.org/abs/1712.06567>.

Suchanek, F. M., Kasneci, G., and Weikum, G. [2007]. YAGO: A core of semantic knowledge – unifying WordNet and Wikipedia. In *16th International World Wide Web Conference (WWW 2007)*.

Sundermann, C., et al. [2021]. Applications of #SAT solvers on feature models. *15th International Working Conference on Variability Modelling of Software-Intensive Systems*.

- Sunstein, C. R. [2018]. *#Republic: Divided Democracy in the Age of Social Media*. Princeton University Press. <http://www.jstor.org/stable/j.ctv8xnhtd>.
- Sutton, R. S. [1988]. Learning to predict by the methods of temporal differences. *Machine Learning*, 3(1):9–44. <http://dx.doi.org/10.1007/BF00115009>.
- Sutton, R. S. and Barto, A. G. [2018]. *Reinforcement Learning: An Introduction*. MIT Press, 2nd edition.
- Szepesvári, C. [2010]. *Algorithms for Reinforcement Learning*. Morgan & Claypool. <http://dx.doi.org/doi:10.2200/S00268ED1V01Y201005AIM009>.
- Tarski, A. [1956]. *Logic, Semantics, Metamathematics*. Clarendon Press. Papers from 1923 to 1938 collected and translated by J. H. Woodger.
- Tate, A. [1977]. Generating project networks. In *5th International Joint Conference on Artificial Intelligence*, pp. 888–893.
- Tay, Y., Dehghani, M., Bahri, D., and Metzler, D. [2022]. Efficient transformers: A survey. *ACM Computing Surveys*. <http://dx.doi.org/10.1145/3530811>.
- Thompson, W. [1933]. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples”. *Biometrika*, 25(3/4):285–294.
- Thrun, S. [2006]. Winning the DARPA grand challenge. In *Innovative Applications of Artificial Intelligence Conference (IAAI-06)*, pp. 16–20.
- Thrun, S., Burgard, W., and Fox, D. [2005]. *Probabilistic Robotics*. MIT Press.
- Torrance, G. [1970]. A generalized cost-effectiveness model for the evaluation of health programs. Technical report, Faculty of Business, McMaster University. <http://hdl.handle.net/11375/5559>.
- Trouillon, T., et al. [2016]. Complex embeddings for simple link prediction. In *ICML*, volume abs/1606.06357. <http://arxiv.org/abs/1606.06357>.
- Turing, A. [1950]. Computing machinery and intelligence. *Mind*, 59:433–460. <https://doi.org/10.1093/mind/LIX.236.433>.
- Tversky, A. and Kahneman, D. [1974]. Judgment under uncertainty: Heuristics and biases. *Science*, 185:1124–1131.
- UNESCO [2022]. Recommendation on the ethics of artificial intelligence. <https://unesdoc.unesco.org/ark:/48223/pf0000381137>.
- United Nations [2015a]. Transforming our world: The UN 2030 agenda for sustainable development. <https://sdgs.un.org/2030agenda>.
- United Nations [2015b]. The UN sustainable development goals. <https://sdgs.un.org/goals>.
- U.S. Government [2022]. GPS accuracy. <https://www.gps.gov/systems/gps/performance/accuracy/>.
- Vallor, S. [2021]. Virtues in the digital age. In *The Oxford Handbook of Digital Ethics*. Oxford University Press. <http://dx.doi.org/10.1093/oxfordhb/9780198857815.013.2>.
- van Beek, P. and Chen, X. [1999]. Cplan: A constraint programming approach to planning. In *AAAI-99*, pp. 585–590.

van de Meent, J.-W., Paige, B., Yang, H., and Wood, F. [2018]. An introduction to probabilistic programming. <https://arxiv.org/abs/1809.10756>.

Van den Broeck, G., Kersting, K., Natarajan, S., and Poole, D. (eds.) [2021]. *Introduction to Lifted Inference*. MIT Press.

van Diggelen, F. and Enge, P. [2015]. The world's first GPS MOOC and worldwide laboratory using smartphones. In *28th International Technical Meeting of the Satellite Division of The Institute of Navigation*.

van Emden, M. H. and Kowalski, R. A. [1976]. The semantics of predicate logic as a programming language. *Journal ACM*, 23(4):733–742.

Vaswani, A., et al. [2017]. Attention is all you need. In *31st Conference on Neural Information Processing Systems*. <https://arxiv.org/abs/1706.03762>.

Veitch, V. and D'Amour, A. [2023]. Causality. In *Murphy [2023]*, chapter 36. MIT Press.

Visser, U. and Burkhard, H.-D. [2007]. Robocup: 10 years of achievements and challenges. *AI Magazine*, 28(2):115–130.

Viswanathan, P., Little, J., Mackworth, A. K., and Mihailidis, A. [2011]. Navigation and obstacle avoidance help (NOAH) for older adults with cognitive impairment: A pilot study. In *International ACM SIGACCESS Conference on Computers and Accessibility*.

Vlassis, N. [2007]. *A Concise Introduction to Multiagent Systems and Distributed Artificial Intelligence*. Morgan & Claypool. <http://dx.doi.org/doi:10.2200/S00091ED1V-01Y200705AIM002>.

Vrandećić, D. and Krötzsch, M. [2014]. Wikidata: A free collaborative knowledge-base. *Communications of the ACM*, 57(10):78–85.

W3C OWL Working Group (ed.) [2012]. *OWL 2 Web Ontology Language Document Overview*. W3C Recommendation 11 December 2012, 2nd edition. <http://www.w3.org/TR/owl2-overview/>.

Wakker, P. P. [2010]. *Prospect Theory: For Risk and Ambiguity*. Cambridge University Press.

Waldinger, R. [1977]. Achieving several goals simultaneously. In Elcock, E. and Michie, D. (eds.), *Machine Intelligence 8: Machine Representations of Knowledge*, pp. 94–136. Ellis Horwood.

Walsh, T. [2007]. Representing and reasoning with preferences. *AI Magazine*, 28(4):59–69.

Walter, W. G. [1950]. An imitation of life. *Scientific American*, 182(5):42–45.

Walter, W. G. [1951]. A machine that learns. *Scientific American*, 185(2):60–63.

Wang, H. [1960]. Toward mechanical mathematics. *IBM Journal of Research and Development*, 4(1):2–22. <http://dx.doi.org/doi:10.1147/rd.41.0002>.

Warren, D. H. D. and Pereira, F. C. N. [1982]. An efficient easily adaptable system for interpreting natural language queries. *Computational Linguistics*, 8(3–4):110–122. <http://portal.acm.org/citation.cfm?id=972944>.

Watkins, C. J. C. H. and Dayan, P. [1992]. Q-learning. *Machine Learning*, 8(3):279–292. <http://dx.doi.org/10.1007/BF00992698>.

Weidinger, L., et al. [2021]. Ethical and social risks of harm from language models. <http://dx.doi.org/10.48550/arXiv.2112.04359>.

Weizenbaum, J. [1976]. *Computer Power and Human Reason: From Judgment to Calculation*. Freeman.

Weld, D. S. [1994]. An introduction to least commitment planning. *AI Magazine*, 15(4):27–61.

Weld, D. S. [1999]. Recent advances in AI planning. *AI Magazine*, 20(2).

Wellman, M. P. [2011]. *Trading Agents*. Morgan & Claypool. <http://dx.doi.org/doi:10.2200/S00370ED1V01Y201107AIM012>.

Whitehead, A. N. and Russell, B. [1925, 1927]. *Principia Mathematica*. Cambridge University Press, 2nd edition.

Wikidata [2021]. Q262802 – wikidata. <https://www.wikidata.org/wiki/Q262802>.

Wilkins, D. E. [1988]. *Practical Planning: Extending the Classical AI Planning Paradigm*. Morgan Kaufmann.

Wilkinson, M. D., et al. [2016]. The fair guiding principles for scientific data management and stewardship. *Scientific Data*, 3(1):160018. <http://dx.doi.org/10.1038/sdata.2016.18>.

Winograd, T. [1972]. *Understanding Natural Language*. Academic Press.

Winograd, T. [1990]. Thinking machines: Can there be? Are we? In Partridge, D. and Wilks, Y. (eds.), *The Foundations of Artificial Intelligence: A Sourcebook*, pp. 167–189. Cambridge University Press.

Wolpert, D. H. [1996]. The lack of a priori distinctions between learning algorithms. *Neural Computation*, 8(7):1341–1390. <http://dx.doi.org/10.1162/neco.1996.8.7.1341>.

Woods, W. A. [2007]. Meaning and links. *AI Magazine*, 28(4):71–92.

Wooldridge, M. [2009]. *An Introduction to MultiAgent Systems*. Wiley.

World Economic Forum [2021]. Responsible use of technology: The IBM case study. <https://www.weforum.org/whitepapers/responsible-use-of-technology-the-ibm-case-study/>.

Xu, K., Hu, W., Leskovec, J., and Jegelka, S. [2019]. How powerful are graph neural networks? In *ICLR*.

Xu, L., Hutter, F., Hoos, H. H., and Leyton-Brown, K. [2008]. SATzilla: Portfolio-based algorithm selection for SAT. *Journal of Artificial Intelligence Research*, 32:565–606. <https://www.aaai.org/Papers/JAIR/Vol32/JAIR-3214.pdf>.

Yang, Q. [1997]. *Intelligent Planning: A Decomposition and Abstraction-Based Approach*. Springer-Verlag.

Yang, S. and Mackworth, A. K. [2007]. Hierarchical shortest pathfinding applied to route-planning for wheelchair users. In *Canadian Conference on Artificial Intelligence, AI-2007*.

Yannakakis, G. N. and Togelius, J. [2018]. *Artificial Intelligence and Games*. Springer. <http://gameaibook.org>.

Zador, A., et al. [2023]. Catalyzing next-generation artificial intelligence through NeuroAI. *Nature Communications*, 14(1):1597. <http://dx.doi.org/10.1038/s41467-023-37180-x>.

Zhang, B. H., Lemoine, B., and Mitchell, M. [2018]. Mitigating unwanted biases with adversarial learning. In *2018 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 335–340. <http://dx.doi.org/10.1145/3278721.3278779>.

Zhang, D., et al. [2022a]. *The AI Index 2022 Annual Report*. AI Index Steering Committee, Stanford Institute for Human-Centered AI, Stanford University. <https://aiindex.stanford.edu/wp-content/uploads/2022/03/2022-AI-Index-Report-Master.pdf>.

Zhang, H., et al. [2022b]. On the paradox of learning to reason from data. <http://starai.cs.ucla.edu/papers/ZhangArxiv22.pdf>.

Zhang, N. L. [2004]. Hierarchical latent class models for cluster analysis. *Journal of Machine Learning Research*, 5(6):697–723.

Zhang, N. L. and Poole, D. [1994]. A simple approach to Bayesian network computations. In *10th Canadian Conference on Artificial Intelligence*, pp. 171–178.

Zhang, Y. and Mackworth, A. K. [1995]. Constraint nets: A semantic model for hybrid dynamic systems. *Theoretical Computer Science*, 138:211–239.

Zilberstein, S. [1996]. Using anytime algorithms in intelligent systems. *AI Magazine*, 17(3):73–83.

Zimmer, M. [2022]. A celebrated AI has learned a new trick: How to do chemistry. *The Conversation*. <https://theconversation.com/a-celebrated-ai-has-learned-a-new-trick-how-to-do-chemistry-182031>.

Zuboff, S. [2019]. *The Age of Surveillance Capitalism : The Fight for a Human Future at the New Frontier of Power*. Profile Books.

MUNDARIJA

IV QISM. NOANIQLIK SHAROITIDA REJALASHTIRISH VA HARAKAT QILISH

12-BOB. NOANIQLIK SHAROITIDA REJALASHTIRISH

12.1. Afzalliklar va foyda.....	4
12.1.1. Ratsionallik aksiomalari.....	4
12.1.2. Faktorli foydalilik.....	12
12.1.3. Istiqbollar nazariyasi.....	15
12.2. Yagona qarorlar.....	17
12.2.1. Bir bosqichli qaror tarmoqlari.....	22
12.3. Ketma-ket qarorlar.....	24
12.3.1. Qaror tarmoqlari.....	25
12.3.2. Tamoyillar.....	30
12.3.3. Qidiruv yordamida qaror tarmoqlarini optimallashtirish.....	32
12.3.4. Qaror tarmoqlari uchun o'zgaruvchini istisno qilish.....	33
12.4. Ma'lumot va boshqaruvning qiymati.....	39
12.5. Qaror qabul qilish jarayonlari.....	41
12.5.1. Tamoyillar.....	47
12.5.2. Qiymatni takrorlash.....	50
12.5.3. Tamoyilni takrorlash.....	53
12.6. Ijtimoiy ta'sir.....	62
12.7. Takrorlash.....	64
12.8. Qo'shimcha o'qish uchun manbalar.....	64
12.9. Mashqlar.....	65

13-BOB. KUCHAYTIRILGAN O'QITISH (REINFORCEMENT LEARNING)

13.1. Kuchaytirilgan o'qitish muammosi.....	76
13.2. Evolyutsion algoritmlar.....	80
13.3. Vaqtinchalik farqlar.....	81
13.4. Tajribalardan o'rganish.....	82
13.4.1. Q-o'rganish.....	82
13.5. Tadqiq qilish va foydalanish.....	84
13.6. RL algoritmlarini baholash.....	87
13.7. On-policy (Tamoyilga asoslangan o'rganish).....	89
13.8. Modelga asoslangan RL.....	91
13.9. Umumlashtirilgan kuchaytirilgan o'qitish.....	93
13.9.1. SARSA algoritmlarining chiziqli funksiya bilan yaqinlashuvi.....	94
13.9.2. Lokal optimallashtirishdan qochish.....	97
13.10. Ijtimoiy ta'sir.....	98
13.11. Takrorlash.....	99
13.12. Qo'shimcha o'qish uchun manbalar.....	99
13.13. Mashqlar.....	100

14-BOB. KO'P AGENTLI TIZIMLAR

14.1. Ko'p agentli tizimlar.....	103
14.2. O'yinlarni ifodalash.....	105
14.2.1. Normal shakldagi o'yinlar.....	105
14.2.2. O'yinning keng ko'lamli shakli.....	106
14.2.3. Ko'p agentli qaror tarmoqlari.....	109

14.3. Mukammal ma'lumotli o'yinlarni hal etish	110
14.3.1. Raqiblar o'yinlari	111
14.4. Mukammal bo'lmagan ma'lumotlar bilan mulohaza qilish	115
14.4.1. Nash muvozanatlarini hisoblash	122
14.5. Guruh bo'lib qaror qabul qilish	124
14.6. Mexanizmlarni loyihalash	126
14.7. Ko'p agentli mustahkamlashni o'rganish	129
14.7.1. Mukammal ma'lumotli o'yinlar	129
14.7.2. Stoxastik tamoyillar bilan mustahkamlashni o'rganish	129
14.7.3. Eng zamonaviy o'yinchilar	133
14.8. Ijtimoiy ta'sir	134
14.9. Takrorlash	135
14.10. Qo'shimcha o'qish uchun manbalar	136
14.11. Mashqlar	137

V BO'LIM. SHAXSLAR VA MUNOSABATLARNI IFODALASH

15-BOB. SHAXSLAR VA MUNOSABATLAR

15.1. Munosabatlar tuzilmasidan foydalanish	141
15.2. Belgilar va semantika	143
15.3. Predikatlar hisobi	144
15.3.1. Asosiy mantiqiy formulalarning semantikasi	146
15.3.2. O'zgaruvchilarni talqin qilish	148
15.4. Datalog: relyatsion qoidalar tili	151
15.4.1. O'zgaruvchilar bilan so'rovlar	153
15.5. Tasdiqlash va almashtirish	155
15.5.1. Isbotlash va almashtirish	155
15.5.2. Ma'lumotlarni yozib olishning pastdan yuqoriga tartibi	157
15.5.3. Birlashtirish	159
15.5.4. O'zgaruvchilar bilan aniq o'lcham	160
15.6. Funksional belgilar va ma'lumot tuzilmalari	162
15.6.1. Funksional belgilar bilan isbot protseduralari	166
15.7. Tabiiy til ilovalari	170
15.7.1. Kontekstsiz grammatika uchun aniq gaplardan foydalanish	172
15.7.2. Grammatikani kengaytirish	176
15.7.3. Ma'lumotlar bazasi uchun tabiiy til interfeysini qurish	176
15.7.4. Katta til modellari bilan taqqoslash	180
15.8. Tenglik	181
15.8.1. Tenglikni tasdiqlashga ruxsat berish	182
15.8.2. Yagona nomlar taxmini	183
15.9. To'liq bilimlar taxmini	186
15.9.1. To'liq bilim taxminining isbot jarayonlari	188
15.10. Ijtimoiy ta'sir	189
15.11. Takrorlash	190
15.12. Qo'shimcha o'qish uchun manbalar	190
15.13. Mashqlar	190

16-BOB. BILIM GRAFIKLARI VA ONTOLOGIYALAR

16.1. Bilim grafiklari	196
16.1.1. Tripletlar (uchliklar)	196

16.1.2. Shaxslar va identifikatorlar	199
16.1.3. Graf taqdimotlar	200
16.2. Sinflar va xususiyatlar	202
16.2.1. Sinf va xususiyatlar iyerarxiyalari	203
16.2.2. Sinflarni loyihalash	206
16.3. Ontologiyalar va bilim almashish	209
16.3.1. Tavsif mantiqi	213
16.3.2. Yuqori darajadagi ontologiyalar	218
16.4. Ijtimoiy ta'sir	221
16.5. Sharhlar	222
16.6. Manbalar va qo'shimcha adabiyotlar bilan ishlash	223
16.7. Mashqlar	223

17-BOB. NISBIY TA'LIMNI O'RGANISH VA EHTIMOLIY FIKRLASH

17.1. Relyatsion xususiyatlar va tasodifiy o'zgaruvchilar	227
17.2. Embeddingga asoslangan modellar	229
17.2.1. Ikkilik munosabatni o'rganish	229
17.2.2. Bilimlar grafini o'rganish	237
17.3. Relyatsion o'zaro bog'liqlikni o'rganish	240
17.3.1. Relyatsion ehtimoliy modellar	240
17.3.2. Jamoaviy tasniflash va Crowd sourcing	247
17.3.3. Til va mavzu modellari	248
17.3.4. Ayrim o'ziga xos tasavvurlar	252
17.4. Mavjudlik va identifikatsiya noaniqligi	255
17.5. Ijtimoiy ta'sir	259
17.6. Takrorlash	259
17.7. Adabiyotlar va qo'shimcha o'qish uchun materiallar	260
17.8. Mashqlar	260

VI QISM. UMUMIY MANZARA

18-BOB. SUN'IY INTELEKTNING IJTIMOY TA'SIRI

18.1. Raqamli iqtisodiyot	265
18.2. Qadriyat va bias	266
18.3. Insonga yo'naltirilgan sun'iy intellekt	267
18.4. Mehnat va avtomatlashtirish	271
18.5. Transport	272
18.6. Barqarorlik	274
18.7. Etika	277
18.8. Boshqaruv va tartibga solish	282
18.9. Xulosa	283
18.10. Mashqlar	284

19-BOB. O'TMISHGA NAZAR VA ISTIQBOLLAR

19.1. Sun'iy intellektni joriy etish	285
19.2. Agent dizayn makonini qayta ko'rib chiqish	288
19.3. Kelajakka nazar	292
19.4. Manbalar va qo'shimcha adabiyotlar	295
19.5. Mashqlar	295
Foydalanilgan adabiyotlar	297

Ushbu xorijiy o'quv adabiyoti o'zbek tiliga ilk marotaba tarjima qilinganligi munosabati bilan nashriyot va tarjimonlar tomonidan ayrim kamchilik va xatoliklarga yo'l qo'yilgan bo'lishi mumkin. Kamchiliklarni bartaraf etish maqsadida fikr-mulohazalaringizni quyidagi manzil va te-lefon raqamlari orqali qabul qilamiz.

Manzil: Toshkent shahar, Navoiy ko'chasi, 30-uy. Telefon raqami 71-244-15-53.

DEVID L. PUL, ALAN K. MAKVORT

SUN'IY INTELLEKT

HISOBLASH AGENTLARINING ASOSLARI

II jild

Muharrir

Badiiy muharrir

Texnik muharrir

Sahifalovchi

T. Mirzayev

D. Mulla-Axunov

R. Ahmedov

B. Dushanova

Noshirlik faoliyati to'g'risidagi xabarnoma
6.11.2023-yil qabul qilingan, tasdiqnoma № 156149.
Bosishga 2025-yil 25-iyunda ruxsat etildi.
Bichimi 70x100¹/₁₆. Bosma tabog'i 21.
Adadi 50. Buyurtma № 039/O-25.

«Donishmand ziyosi» MChJ nashriyotida nashrga tayyorlandi.
Toshkent shahar, Navoiy ko'chasi, 30-uy.

«Kamalak-PRESS» MChJ bosmaxonasida chop etildi.
Toshkent shahar, Navoiy ko'chasi, 30-uy.



«DONISHMAND ZIYOSI»

ISBN 978-9910-715-91-4



9 789910 715914