ISSN 0002-306X. ՀԳԱԱ և ՀԱՊՀ Տեղ. Տեխն. գիտ. սերիա. 2019. Հ. LXXII, N3.

ՀՏԴ 681.518 ԱԿՆԱՐԿԱՅԻՆ ՀՈԴՎԱԾ

Մ.Է. ՍԱՍՎԵԼՅԱՆ

ՀԱՄԱԳՈՐԾԱԿՑԱՅԻՆ ԽՈՐՔԱՅԻՆ ԲԱԶՄԱԳՈՐԾԱԿԱԼ ԱՄՐԱՊՆԴՄԱՄԲ ՈՒՍՈՒՑՈՒՄ. ԴԺՎԱՐՈՒԹՅՈՒՆՆԵՐ, ՄԿԶԲՈՒՆՔՆԵՐ ԵՎ ԲԱՑ ԽՆԴԻՐՆԵՐ

Վերջին տարիներին խորքային ամրապնդմամբ ուսուցման (ԱՈւ) ոլորտում նշանակալից հաջողություններ է արձանագրվել մի շարք դժվարին խնդիրներում, ինչպիսիք են համակարգչային խաղերը, սեղանի խաղերը և ռոբատաշինությունը: Չնայած այս հաջողությունների մեծ մասը տեղի է ունեցել միագործակալ ԱՈւ-ում, սակայն իրական աշխարհի բազում միջավայրեր բաղկացած են մեծ թվով գործակալներից, որոնք պետք է համագործակցեն՝ ընդհանուր խնդիրները լուծելու համար: Այնուամենայնիվ, միագործակալ ԱՈւ ոլորտում մեթոդները պիտանի չեն բազմագործակալ սցենարների համար՝ մի շարք դժվարությունների պատճառով: Սույն աշխատանքում, հետազոտելով ԱՈւ խնդիրն ու գոյություն ունեցող ԱՈւ մեթոդները, մատնանշվում են վերջիններիս օգտագործման հիմնախնդիրները բազմագործակալ միջավայրերում: Ակնարկ է արվում ընթացիկ բազմագործակալ ԱՈւ ալգորիթմների վերաբերյալ և մատնանշվում դրանց սահմանափակումները: Համառոտ նկարագրվում են մի քանի հեռանկարային ուղղություններ՝ հետագա աշխատանքների կատարման համար:

Առանցքային բառեր. ամրապնդմամբ ուսուցում, բազմագործակալ համակարգ, բազմագործակալ ուսուցում:

Ներածություն: Ամրապնդմամբ ուսուցումը (reinforcement learning) մեքենայական ուսուցման (ՄՈւ) (machine learning) ոլորտ է, որն առնչվում է այն հարցին, թե ինչպես պետք է որոշակի գործակալ գործի միջավայրում, որպեսզի առավելագույնի հասցնի միջավայրից տրված գումարային պարգևատրումը (reward) [1]: ԱՈւ-ն առանձնանում է ՄՈւ-ի այլ ճյուղերից և պլանավորումից (planning) նրանով, որ ԱՈւ-ում շեշտը դրվում է գործակալի վարքը շրջակա միջավայրի հետ անմիջական փոխազդեցության արդյունքում ուսուցանելու վրա՝ առանց հիմնվելու որևիցե վերահսկողության կամ շրջակա միջավայրի ամբողջական մոդելի վրա: Խորքային ուսուցման (deep learning) [2] վերջին տարիների զարգացումները պայմանավորեցին ԱՈւ մեթոդների առաջխաղացումը արհեստական բանականության մի շարք բարդ խնդիրների լուծման ոլորտում, ինչպիսիք են համակարգչային խաղերը [3], սեղանի խաղերը [4] և ռոբոտաշինությունը [5]:

Իրական աշխարհի բազում միջավայրեր բաղկացած են բազմաթիվ գործակալներից, որոնք հաճախ պետք է համագործակցեն՝ ընդհանուր նպատակին հասնելու համար: Օրինակ՝ անօդաչու թռչող սարքերի, ինքնագնաց մեքենաների և

արտադրական գործարաններում բազմաբնույթ ռոբոտային սարքերի համակարգումը բնական կերպով ներկայացվում է որպես համագործակցային բազմագործակալ խնդիրների շարք: Սակայն միագործակալ ԱՈւ-ի համար մշակված մեթոդները հաճախ պիտանի չեն բազմագործակալ համակարգերի խնդիրների լուծման համար՝ որոշակի դժվարությունների պատճառով, ինչպիսիք են գործողությունների մեծ տիրույթը, ապակենտրոնացումը (decentralisation), ոչ ստացիոնարությունը (non-stationarity), հաղորդակցությունը և այլն: Այս մարտահրավերներն էլ ավելի դժվարին են դարձնում բազմագործակալ ԱՈւ-ի (ԲԱՈւ) կիրառումը այն միջավայրերում, որտեղ առկա են մեծ թվով գործակալներ:

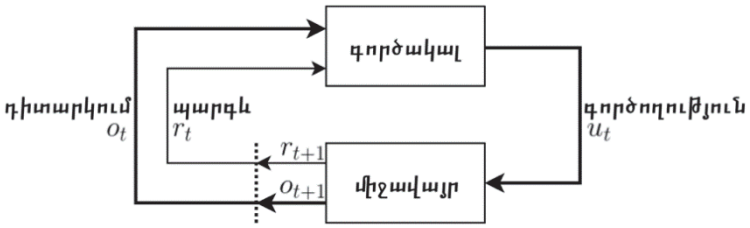
Սույն աշխատանքում, ուսումնասիրելով գոյություն ունեցող միագործակալ ԱՈւ-ի մեթոդները, նկարագրվում են բազմագործակալ համակարգերում վերջիններիս կիրառման դժվարությունները, ինչպես նաև գոյություն ունեցող լուծումները: Բացի այդ, մատնանշվում են ԲԱՈւ ալգորիթմների թերությունները, և առաջարկվում են մի քանի ուղղություններ՝ հետագա աշխատանքների կատարման համար:

Միագործակալ խորքային ամրապնդմամբ ուսուցում: Ներկայացվում են միագործակալ ԱՈւ-ն (այսուհետ ԱՈւ ասելով կհասկանանք միագործակալ ԱՈւ-ն), ԱՈւ-ի խնդրի դրվածքը, վերջինիս լուծման ժամանակակից մեթոդները:

ԱՈւ-ի նպատակն է սովորել՝ ինչպես պետք է իրավիճակներին համապատասխանեցնել այնպիսի գործողություններ, որ առավելագույնի հասցվի թվային պարգևատրումը [1]: Ուսումնառողին չի հաղորդվում, թե որ գործողությունները պետք է կատարվեն: Փոխարենը, փորձելով տարբեր գործողություններ, նա պետք է ինքնուրույն բացահայտի, թե որոշակի իրավիճակներում որ գործողություններն են առավելագույն չափով պարգևատրվում: Գործողությունները կարող են ազդեցություն ունենալ ոչ միայն անմիջապես պարգևատրման վրա, այլև հաջորդ իրավիճակի, ինչպես նաև բոլոր հետագա պարգևատրումների վրա: Փորձի և սխալի որոնումը և հետաձգված պարգևատրումը ԱՈւ-ի երկու ամենակարևոր առանձնահատկություններն են:

ԱՈւ-ի խնդիրը պարզագույն դեպքում ներառում է միայն մեկ ուսումնառող և որոշում կայացնող, որին սովորաբար անվանում են **գործակալ** (agent): Գործակալից դուրս ամեն ինչ կոչվում է **միջավայր** (environment): Միջավայրը և գործակալը գտնվում են շարունակական փոխազդեցության մեջ: Ժամանակի ամեն դիսկրետ t քայլին գործակալը ստանում է միջավայրից o_t **դիտարկում** (observation) և կատարում է u_t **գործողությունը** (action): Ժամանակի հաջորդ քայլին գործակալը ստանում է թվային r_t **պարգևատրումը** (reward)՝ որպես կատարած գործողությունների արդյունք: Մյուս կողմից՝ միջավայրը յուրաքանչյուր քայլում ստանում է u_t գործողությունը, անցում է կատարում նոր վիճակի և գործակալին է ուղարկում

o_{t+1} դիտարկումը և r_{t+1} պարգևատրումը: Նկարում ներկայացվում է գործակալի և միջավայրի միջև փոխազդեցությունը ԱՈւ-ում:



Նկ. Գործակալի և միջավայրի փոխազդեցությունը ԱՈւ-ում

Լիարժեքորեն դիտարկելի (fully observable) միջավայրերում գործակալը ուղղակիորեն կարող է դիտարկել միջավայրի առկա s_t վիճակը՝ $o_t = s_t$: *Մասնակիորեն դիտարկելի* (partially observable) միջավայրերում միջավայրի s_t վիճակը հասանելի չէ գործակալին, և փոխարենը նա ստանում է այնպիսի դիտարկում, որը սահմանափակ տեղեկություն է պարունակում վիճակի մասին:

Լիարժեքորեն դիտարկելի ԱՈւ-ի խնդիրը հաճախ կարելի է ձևակերպել որպես *Մարկովյան որոշումների գործընթաց* (ՄՈԳ) (Markov Decision Process) [1]: ՄՈԳ-ը ներկայացվում է $\langle S, U, \mathcal{P}, \mathcal{R}, \gamma \rangle$ հնգյակի միջոցով, որտեղ $\mathcal{S}$-ը վիճակների վերջավոր բազմությունն է (state space), իսկ $\mathcal{U}$-ն՝ գործակալի գործողությունների վերջավոր բազմությունը (action space): $\mathcal{P}: S \times \mathcal{U} \times S \to [0, 1]$ ներկայացնում է վիճակների անցման ֆունկցիան (transition function), որտեղ $\mathcal{P}(s, u, s') = \mathbb{P}[s_{t+1} = s' | s_t = s, u_t = u]$, իսկ $\mathcal{R}: S \times \mathcal{U} \to \mathbb{R}$-ն պարգևատրման ֆունկցիան է (reward function), որտեղ $\mathcal{R}(s, u) = \mathbb{E}[r_{t+1} | s_t = s, u_t = u]$: γ-ն ծառայում է որպես պարգևատրման զեղչման գործակից (discount factor):

Ժամանակի ամեն t քայլին (time step) գործակալի նպատակն է ընտրել այն գործողությունը, որն առավելագույնի կհասցնի հետագայում ստացվող պարգևատրումները: Մասնավորապես, գործակալի նպատակն է առավելագույնի հասցնել սպասվող **զեղչված շահույթը** (discounted return), որը տրվում է հետևյալ բանաձևով՝

$$R_t = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \cdots = \sum_{n=0}^{\infty} \gamma^n r_{t+n+1},$$

որտեղ զեղչման γ գործակիցը որոշում է հետագա պարգևատրումների առկա արժեքը:

Գործակալի վարքը նկարագրվում է π **ռազմավարությամբ** (policy), որն արտապատկերում է վիճակների բազմությունից դեպի գործակալի հնարավոր գործողություններն ընտրելու հավանականությունը: Ռազմավարությունը կարող է լինել դետերմինիստական, երբ $u = \pi(s)$, կամ ստոխաստիկական, երբ $\pi(u|s) =$

$= \mathbb{P}[u_t = u | s_t = s]$: Այսպիսով, ԱՈւ խնդրի նպատակն է մշակել մի ռազմավարություն, որն առավելագույնի է հասցնում սպասվող զեղչված շահույթը բոլոր $s \in S$ վիճակների դեպքում:

π ռազմավարության արժեքային ֆունկցիան (value function) սահմանվում է որպես սպասված շահույթ s վիճակում u գործողությունը կատարելիս, և այնուհետև, ըստ π ռազմավարության գործելիս՝ $Q^{\pi}(s,u) = \mathbb{E}[R_t | s_t = s, u_t = u]$: Բելմանի օպտիմալության հավասարումը ռեկուրսիվորեն նկարագրում է օպտիմալ Q-ֆունկցիան՝ $Q^*(s,u) = \max_{\pi} Q^{\pi}(s,u)$, որպես ֆունկցիա՝ կախված ՄՈԳ-ի պարգևատրման և վիճակների անցումային ֆունկցիաներից՝

$$Q^*(s,u) = \mathcal{R}(s,u) + \gamma \sum_{s'} \mathcal{P}(s,u,s') \max_{u'} Q^*(s',u'):$$

Ունենալով օպտիմալ Q-ֆունկցիան՝ կարող ենք մշակել օպտիմալ π^* ռազմավարությունը, որը միշտ «ագահաբար» է գործում (օպտիմալ ընտրություն է կատարում) Q^*-ի նկատմամբ՝ $\pi^*(u|s) = \arg\max_{u} Q^*(s,u)$: Այնուամենայնիվ, ՄՈԳ-ը, մասնավորապես $\mathcal{R}$ և $\mathcal{P}$ ֆունկցիաները, անհայտ է կամ շատ մեծ, ինչը գործնականում թույլ չի տալիս հաշվարկել օպտիմալ Q-ֆունկցիան դինամիկ ծրագրավորման միջոցով:

Ժամանակի ընթացքում առաջարկվել են բազում մեթոդներ ԱՈւ խնդիրը լուծելու համար՝ մոտարկելով Q-ֆունկցիան: ԱՈւ-ում առաջխաղացմանը նպաստած ալգորիթմներից են Q-ուսուցումն [6] ու TD-ուսուցումը [7], որոնք բոլոր $s \in S$ և $u \in U$ զույգերի համար պահում են $Q(s,u)$-ի արժեքները համապատասխան աղյուսակում: Ուսուցման ընթացքում, երբ գործակալը առկա ռազմավարությամբ գործողություններ է կատարում միջավայրում և ստանում գործողությունների համապատասխան իրական պարգևատրումները, աղյուսակի արժեքները թարմացվում են: Ապացուցված է, որ որոշակի պայմաններ բավարարելու դեպքում Q-ուսուցումն ու TD-ուսուցումը զուգամիտում են դեպի օպտիմալ Q^* -ն [7, 8]:

Աղյուսակային մեթոդները, այնուամենայնիվ, նպատակահարմար չեն այն ՄՈԳ-երում, որտեղ վիճակների S բազմությունը հսկայական է: Օրինակ՝ նարդի խաղում վիճակների բազմության հզորությունը 10^{20} կարգի է, «Գո» խաղում՝ 10^{170}, իսկ «Աթարի» համակարգչային խաղերում՝ $256^{84\times84\times4}$ [3]: Հնարավոր լուծումներից մեկն է Q-ֆունկցիայի մոտարկումը ֆունկցիաների մոտարկիչների (function approximator) միջոցով, ինչպիսիք են խորքային նեյրոնային ցանցերը [2]:

Խորքային ԱՈւ-ի առաջին հաջողված մոդելը խորքային Q-ցանցերն (ԽՔՑ) են [3], որոնցում Q-ֆունկցիան ներկայացված է որպես նեյրոնային ցանց՝ պարամետրացված θ պարամետրերով՝ $Q(s,u;\theta)$: Ուսման ժամանակ գործակալն ընտրում է գործողությունները՝ ելնելով ուսումնասիրության (exploration) որոշակի մեթո-

դից, ինչպիսին է ϵ-«ագահ» ուսումնասիրությունը, երբ $1-\epsilon$ հավանականությամբ ընտրվում է օպտիմալ գործողությունը՝ $\arg\max_u Q(s,u;\theta)$, իսկ ϵ հավանականությամբ՝ պատահական գործողություն։ Յուրաքանչյուր գործողությունը կատարելուց և պարգևատրումն ու հաջորդ վիճակը ստանալուց հետո $\langle s,u,r,s'\rangle$ քառյակը պահվում է հիշողության մեջ։ θ պարամետրերն ուսուցանվում են հիշողությունից պատահական կերպով b քառյակներ ընտրելու և հետևյալ ֆունկցիան նվազեցնելու արդյունքում՝

$$\mathcal{L}(\theta)=\sum_{i=1}^{b}\left[\left(r+\gamma\max_{u'}Q(s',u';\theta^{-})-Q(s,u;\theta)\right)^2\right], \qquad (1)$$

որտեղ θ^{-}-ն թիրախային Q-ցանցի պարամետրերի բազմությունն է։ Թիրախային Q-ցանցը (target network) հիմնական Q-ցանցի պատճենն է, որը «սառեցվում» է ուսուցանման ժամանակ և թարմացվում միայն որոշակի քանակի իտերացիաներից հետո։

Ժամանակի ընթացքում ԽՔՑ-ն բարելավվել է տարբեր եղանակներով։ Օրինակ՝ կրկնակի ԽՔՑ-ում օպտիմալացման $\mathcal{L}(\theta)$ ֆունկցիայի սահմանման մեջ հաջորդ քայլում ստացված պարգևատրումը գնահատելու համար ընտրվում լավագույն գործողությունը՝ ելնելով առկա, այլ ոչ թե թիրախային Q-ցանցից [9]։ Եվս մեկ բարելավում է ԽՔՑ-ում Q-ցանցում տեղի ունեցող կառուցվածքային փոփոխությունը, որի արդյունքում ցանցի ելքում գնահատվում են և՝ վիճակի արժեքային ֆունկցիան, և՝ վիճակում համապատասխան գործողություն ընտրելու առավելության (advantage) ֆունկցիան [10]։

Մասնակիորեն դիտարկելի միջավայրերում նպատակահարմար է մշակել այնպիսի ռազմավարություն, որը հիմնվում է դիտարկումների ամբողջական պատմության վրա։ Խորքային ռեկուրենտ Q-ցանցերը [11] օգտագործում են ռեկուրենտ նեյրոնային ցանցեր (recurrent neural networks), ինչպիսիք են LSTM [12] և GRU [13] ցանցերը, որպեսզի մշակեն Q-ցանցեր, որոնք Q-արժեքի գնահատման ընթացքում օգտագործեն դիտարկումների հաջորդականություն մեկ դիտարկման փոխարեն։

Ռազմավարության գրադիենտային մեթոդներ։ Բազում խնդիրների համար (օրինակ՝ ռոբոտաշինության համար) գործողությունների տիրույթը ոչ թե դիսկրետ բազմություն է, այլ անընդհատ։ Այս բնույթի խնդիրների համար ԽՔՑ և դրա նման մեթոդները կիրառելի չեն։ Փոխարենը կիրառվում են ռազմավարության գրադիենտային մեթոդներ (policy gradient methods) [14], որոնք մշակում են գործակալի ռազմավարությունը ոչ թե Q-ուսուցման նման անուղղակի կերպով, այլ ուղղակիորեն մշակելով ստոխաստիկական $\pi(s,u;\theta)$ ռազմավարություն՝ հիմնված θ պարամետրերի վրա։ Գործնականում առանձնահատուկ արդյունք են դրսևո-

րում «*դերասան-քննադատ*» (actor-critic) բնույթի ալգորիթմները, որտեղ գործակալը գործողություններն ընտրում է՝ ելնելով $\pi(s,u;\theta)$ ստոխաստիկ ռազմավարությունից, սակայն ուսուցման ժամանակ պարամետրերը թարմացվում են՝ օգտագործելով գործակալի Q-ֆունկցիան: Այսպիսի խորքային ամրապնդմամբ ուսուցման ալգորիթմներից են DDPG-ն [15], A3C-ն [16], PPO-ն [17] և SAC-ը [18]:

Բազմագործակալ խորքային ամրապնդմամբ ուսուցում: Իրական աշխարհի տարաբնույթ խնդիրներ, որոնք կարող են լուծվել ԱՈւ-ի կիրառման միջոցով, կազմված են ոչ թե մեկ, այլ բազում գործակալներից: Օրինակ՝ ինքնագնաց մեքենաների, անօդաչու սարքերի և այլ բնույթի բազմառոբոտ համակարգերի ղեկավարումը դառնում է ավելի ու ավելի առաջնային: Արդյունքում անհրաժեշտություն է առաջանում զարգացնել ԲԱՈւ մեթոդներ, որոնք կարող են հաջողությամբ կիրառվել տարատեսակ բազմագործակալ համակարգերում:

Առանձնակի հետաքրքրություն են ներկայացնում մասնակիորեն դիտարկելի, համագործակցային (cooperative) բազմագործակալ ուսուցման խնդիրները: Համագործակցային խնդիրները կազմում են կարևոր խնդիրների հսկայական դաս, որտեղ մեկ օգտատեր, որը կառավարում է բաշխված համակարգը, կարող է որոշել ընդհանուր նպատակը, օրինակ՝ նվազագույնի հասցնել երթևեկության խցանումը կամ այլ իրադրություններ: Իրական աշխարհի խնդիրներից շատերը պայմանավորված են աղմկոտ կամ սահմանափակ սենսորային դիտարկումներով: Մասնակի դիտարկելի միջավայրերը նաև առաջացնում են հաղորդակցության սահմանափակումներ, որի արդյունքում գործակալներին անհրաժեշտ է լինում ուսանել ապակենտրոնացված (decentralised) ռազմավարություններ: Այնուամենայնիվ, մարզման ընթացքում հաճախակի առկա է լինում լրացուցիչ տեղեկություն, որը կարող է իրականացվել վերահսկվող պայմաններում կամ սիմուլյացիայի միջոցով:

Այսպիսի խնդիրները կարելի է ձևակերպել որպես ապակենտրոնացված մասնակի դիտարկելի ՄՈՊ (decentralised partially-observable MDP) [19]՝ կազմված հետևյալ յոթյակից՝ $\langle S, U, \mathcal{P}, \mathcal{R}, Z, O, n, \gamma\rangle$: $s \in S$ նկարագրում է միջավայրի վիճակը: n թիվը սահմանում է գործակալների քանակը: Ժամանակի ամեն քայլին յուրաքանչյուր $a \in A \equiv \{1, \dots, n\}$ գործակալ ընտրում է $u_a \in U$ գործողությունը, և նրանց գործողությունները միասին կազմում են համատեղ (joint) $\mathbf{u} \in \mathbf{U} \equiv U^n$ գործողությունը: Գործողությունը կատարելիս միջավայրի վիճակը փոփոխվում է՝ ելնելով վիճակների անցման ֆունկցիայից՝ $\mathcal{P}: S \times \mathbf{U} \times S \to [0,1]$: Բոլոր գործակալները կիսում են ընդհանուր պարգևատրումը ըստ $\mathcal{R}: S \times \mathbf{U} \to \mathbb{R}$ ֆունկցիայի: γ -ը պարգևատրման զեղչման գործակիցն է:

Դիտարկում ենք մասնակի դիտարկելի միջավայրերը, որտեղ յուրաքանչյուր գործակալ ստանում է $z \in Z$ դիտարկում՝ ելնելով դիտարկումների

$O(s,a): S \times A \to Z$ ֆունկցիայից: Յուրաքանչյուր գործակալ ունի գործողություն-դիտարկում զույգերի պատմություն $\tau \in T \equiv (Z \times U)^*$, ըստ որի կարելի է մշակել ստոխաստիկական $\pi^a(u^a|\tau^a): T \times U \to [0,1]$ ռազմավարություն: Համատեղ π ռազմավարության համար կարող ենք սահմանել համատեղ արժեքային ֆունկցիա՝ $Q^\pi(s_t, \mathbf{u}_t) = \mathbb{E}[R_t|s_t, \mathbf{u}_t]$:

ԲԱՈւ դժվարությունները: Վերոնշյալ խնդրների պարագայում ԱՈւ-ի հիմնական մեթոդները պիտանի չեն հետևյալ պատճառներով.

1) Գործողությունների բազմություն էքսպոնենցիալ աճ: Գործակալների համատեղ գործողությունների բազմությունն աճում է էքսպոնենցիալ՝ արագ գործակալների քանակի աճի հետ: Խնդիրը լուծելի դարձնելու համար հաճախ անհրաժեշտ է օգտագործել ապակենտրոնացված ռազմավարություն, երբ յուրաքանչյուր գործակալ ընտրում է իր գործողությունը՝ հիմնվելով բացառապես իր սեփական դիտարկումների ու գործողությունների պատմության վրա:

2) Ոչ ստացիոնարություն: Մարզման ընթացքում յուրաքանչյուր գործակալի ռազմավարությունը փոփոխվում է՝ առանձին գործակալների տեսանկյունից միջավայրը դարձնելով ոչ ստացիոնար:

3) Ներդրման վերագրում (credit assignment): Այն դեպքում, երբ գործակալները ստանում են ընդհանուր թիմային պարգևատրում, անհատ գործակալները պետք է որոշեն իրենց ներդրումը թիմի հաջողության մեջ:

4) Ապակենտրոնացում: Հաղորդակցման սահմանափակումները և մասնակի դիտարկելիությունը կարող են ստիպել ուսանել ապակենտրոնացված ռազմավարություններ նույնիսկ այն դեպքում, երբ համատեղ գործողությունների տիրույթը շատ մեծ չէ:

5) Հաղորդակցություն: Բազմագործակալ համակարգերում հաճախ առկա են հաղորդակցման ուղիներ, որոնցով գործակալները կարող են տեղեկություններ փոխանակել: Այնուամենայնիվ, առայժմ լուծված չէ այն խնդիրը, թե ինչպես հաղորդակցվել այլ գործակալների կամ մարդկանց հետ՝ ընդհանուր նպատակին հասնելու համար:

Վերոհիշյալ հանգամանքները դժվարացնում են ԱՈւ մեթոդների կիրառումը բազմագործակալ համակարգերում, որտեղ առկա են մեծ թվով գործակալներ: Այնուամենայնիվ, ԲԱՈւ մեթոդների բարելավումը շատ կարևոր է արհեստական բանականությամբ օժտված համակարգեր կառուցելու համար, որոնք արդյունավետ կերպով կարող են հաղորդակցվել ինչպես միմյանց, այնպես էլ մարդկանց հետ:

ԲԱՈՒ գոյություն ունեցող մեթոդներ: Համագործակցային ԲԱՈւ-ում թերևս ամենատարածված մեթոդն անկախ Q-ուսուցումն է (independent Q-learning) [20], որը ԲԱՈւ խնդիրը լուծում է՝ այն վերածելով միևնույն միջավայրը կիսող բազում

միագործակալ ԱՈւ խնդիրների հավաքածուի: Գործակալներից յուրաքանչյուրն ուսանում է իր Q-ֆունկցիան, որը կարող է հիմք ծառայել ապակենտրոնացված «ագահ» ռազմավարություն մշակելու համար: Կարելի է նաև ձևակերպել անկախ ուսուցման «դերասան-քննադատ» տարբերակը [21]: Այս մոտեցումները, այնուամենայնիվ, անտեսում են ԲԱՈւ խնդրիրներում գործակալների փոփոխվող ռազմավարությունների արդյունքում առաջացած ոչ ստացիոնարությունը:

Ապակենտրոնացված ռազմավարությունները հաճախ կարելի է մարզել կենտրոնացված ձևով, օրինակ՝ սիմուլացված միջավայրում կամ լաբորատոր պայմաններում: Սա կարող է ապահովել միջավայրի $s \in S$ վիճակի հասանելիությունը մարզման ժամանակ, որն այլապես հասանելի չէ անհատ գործակալներին, ինչպես նաև հեռացնել հաղորդակցման սահմանափակումները գործակալների միջև: Ապակենտրոնացված ռազմավարությունների կենտրոնացված մարզումը կարող է օգնել՝ հաղթահարելու էքսպոնեցիալ արագ աճող գործողությունների բազմության խնդիրը՝ ուսուցանելով համատեղ Q-ֆունկցիա, որը կարող է ֆակտորացվել առանձին գործակալների Q-ֆունկցիաների: VDN մեթոդը ներկայացնում է համատեղ Q -ֆունկցիան որպես անհատ գործակալների Q -ֆունկցիաների գումար՝ $Q_{tot}(\boldsymbol{\tau}, \mathbf{u}) = \sum_{i=1}^{n} Q_i(\tau, u^i; \theta^i)$ [22], ինչը թույլ է տալիս մշակել ապակենտրոնացված ռազմավարություն: QMIX մեթոդը ներկայացնում է համատեղ Q-ֆունկցիան որպես անհատ Q-ֆունկցիաների ոչ գծային համադրություն [23]: Մասնավորապես, յուրաքանչյուր գործակալի Q-ֆունկցիա տրվում է որպես հատուկ խառնիչ (mixing) նեյրոնային ցանցի մուտք, որը արտածում է համատեղ Q-ֆունկցիան: Մարզման ընթացքում խառնիչ ցանցի գործակիցները մշակվում են հիպերցանցի միջոցով, որը որպես մուտք ստանում է միջավայրի վիճակը և արտածում է խառնիչ ցանցի գործակիցների բազմությունը: Հիպերցանցի ելքային շերտում կիրառվում է բացարձակ արժեք ֆունկցիան, որի արդյունքում խառնիչ ցանցի գործակիցները լինում են դրական թվեր: Արդյունքում, ստեղծվում է մոնոտոն կապ համատեղ Q-ֆունկցիայի և գործակալների Q-արժեքների միջև՝ $\frac{\partial Q_{tot}}{\partial Q_a} \geq 0$ կամայական a գործակալի համար, որի շնորհիվ $\arg\max_{\mathbf{u}} Q_{tot}(\boldsymbol{\tau}, \mathbf{u}) = \{\arg\max_{u^1} Q_1(\tau^1, u^1), \dots, \arg\max_{u^n} Q_n(\tau^n, u^n)\}$, այսինքն՝ առանձին գործակալները, «ագահաբար» վարվելով, կատարում են «ագահ» համատեղ գործողությունը:

Հնարավոր է նաև ուսանել ամբողջովին կենտրոնացված համատեղ Q-ֆունկցիան, որն ուղղորդում է ռազմավարության օպտիմալացումը «դերասան-քննադատ» բնույթի ալգորիթմներում՝ մշակելով ապակենտրոնացված ռազմավարություններ: MADDPG ալգորիթմը մարզում է կենտրոնացված քննադատ յուրաքանչյուր գործակալի համար, որը կարող է կիրառվել և՛ համագործակցային, և՛ մրցակցային միջավայրերում [24]: COMA մեթոդը, ի տարբերություն MADDPG-ի, ուղղա-

կիորեն անդրադառնում է թիմի համատեղ պարգևատրմանը՝ առանձին գործակալների ներդրումների վերագրման խնդրին, օգտագործելով կենտրոնացված քննադատ և ապակենտրոնացված առավելության ֆունկցիաներ, բայց սահմանափակվում է միայն այն միջավայրերով, որտեղ գործողությունների բազմությունը դիսկրետ է [21]: Գոյություն ունեն նաև այլ «դերասան-քննադատ» տիպի մեթոդներ, որոնք կիրառում են ապակենտրոնացված քննադատ Q-ֆունկցիաներ յուրաքանչյուր գործակալի համար՝ առանց օգտագործելու կենտրոնացված մարզման հնարավորությունը [25]: Այնուամենայնիվ, վերոգրյալ բոլոր ալգորիթմները մարզման ընթացքում պահանջում են հսկայական ծավալի օրինակներ և հաճախ զուգամիտում են ոչ օպտիմալ լոկալ մինիմումի կետերի: Բացի այդ, կենտրոնացված քննադատների մարզումը հաճախ անիրագործելի է, եթե գործակալների թիվը շատ մեծ է:

ԲԱՈՒ բաց խնդիրներ: Ներկայացվում են գոյություն ունեցող խորքային ԲԱՈւ մեթոդներում առկա որոշ խնդիրներ, և համառոտ նկարագրվում են մի քանի հեռանկարային ուղղություններ հետագա հետազոտությունների համար:

Միջավայրի համակարգված ուսումնասիրություն: ԲԱՈւ-ում համատեղ գործողությունների մեծ բազմությանն առնչվող հիմնական խնդիրը վիճակների բազմության ուսումնասիրությունն է: Ներկայիս ԲԱՈւ մեթոդների գերակշռող մասն օգտագործում է անհատ գործակալների կողմից իրականացվող պարզ, չուղղորդված հետազոտություններ: Այնուամենայնիվ, օպտիմալ համատեղ ռազմավարությունը հաճախ պահանջում է բոլոր գործակալներից կամ գործակալների ինչ-որ ենթաբազմությունից համատեղ ուսումնասիրել նոր ռազմավարությունը՝ համաձայնեցված ընտրելով պատահական գործողություններ:

ԲԱՈւ-ում միջավայրի ուսումնասիրության հետագա առաջընթացի համար հեռանկարային ուղղություն է պատահական պրոցեսների վրա հիմնված ուսումնասիրության մեթոդների մշակումը, որոնք փոխկապակցված են գործակալների միջև և կատարվում են կենտրոնացված մարզման ընթացքում: Մասնավորապես, ուսումնասիրությունը կարող է իրականացվել գործակիցների տիրույթում, գործողությունների տիրույթի փոխարեն, քանի որ ուսումասիրվող ռազմավարությունը կարող է պահանջել բազմաթիվ գործողություններ կամ ամբողջական էպիզոդներ:

Հիերարխիկ ԲԱՈւ: ԲԱՈւ-ում հավանական առաջխաղացում կարող է հանդիսանալ որոշումների կայացման բարդության նվազեցումը ռազմավարության հիերարխիկ ներկայացման միջոցով: Չնայած նորագույն մեթոդները, ինչպիսին է, օրինակ, QMIX-ը, կարող են լուծել 30 գործակալներ ներառող բարդ բազմագործակալ խնդիրներ [26], այնուամենայնիվ, քիչ հավանական է, որ դրանք մշտապես կարող են մշակել օպտիմալ ռազմավարություններ այնպիսի միջավայրերում, որոնք կազմված են շատ ավելի մեծ թվով գործակալներից (օրինակ՝ 100-ից ավելի):

Գործակալների թվի աճի հետ մեկտեղ բարդանում է նաև ներդրման վերագրման խնդիրը:

Նման զանգվածային բազմագործակալ միջավայրերի համար կարող է արդյունավետ լինել համատեղ Q-ֆունկցիան ֆակտորացնելը հիերարխիկ շերտերի, որոնք ներկայացնում են գործակալների ֆիքսված խմբեր: Յուրաքանչյուր խմբի ներսում գործող գործակալները հնարավորություն կունենան շփվելու խմբի անդամների հետ՝ կանխորոշված կամ ուսուցանված արձանագրությունների միջոցով: Այնուհետև յուրաքանչյուր խումբ կստեղծի համատեղ խմբի Q-արժեքը, որը, ի վերջո, կխառնվի համընդհանուր Q-արժեքի հետ: Ուշադրության արժանի է այն հարցը, թե ինչպես լավագույնս օգտագործել և կենտրոնացված, և տեղային դիտարկումները համատեղ Q-արժեքների հիերարխիկ հաշվարկի մեջ, և թե ինչպես կարգավորել գործակալների միջև կապը հիերարխիայի տարբեր մակարդակներում:

Մարդակենտրոն ԲԱՈւ: Քննարկման արժանի է այն խնդիրը, թե արդյոք հնարավո՞ր է մի խումբ գործակալների ուսուցանել բնական լեզվով հաղորդակցվել այլ գործակալների և մարդկանց հետ: Նման բարդ նպատակի հասնելու համար անհրաժեշտ է համատեղել հետազոտությունների առաձին ուղղություններ, ինչպիսիք են միջգործակալ հաղորդակցությունը (inter-agent communication) [27], նպատակաուղղված երկխոսության համակարգերը (goal-oriented dialogue systems) [28], հրահանգներին հետևելը (instruction following) [29], ինչպես նաև խոսելն ու գործելը (speaking and acting) [30]:

Նախ և առաջ անհրաժեշտ է ստեղծել այնպիսի հատուկ ԱՈւ միջավայր, որը կպահանջի մեկից ավելի գործակալների բնական լեզվով հրահանգներ ստանալը, անհրաժեշտության դեպքում հարցեր տալը, այնուհետև միացյալ ուժերով ընդհանուր խնդիրը լուծելը: Կարևոր է ուսանել ընդհանրացնել նախկինում չհանդիպած ցուցումները և կարողանալ համագործակցել տարանջատման պայմաններում մարզված գործակալների հետ: Գործակալները պետք է նաև կարողանան կարգավորել անսպասելի իրադարձությունները: Օրինակ՝ գործակալները կարող են ստանալ նոր հրահանգներ, որից հետո գործակալների մի խումբ պետք է ընդհատի իր ընթացիկ առաջադրանքը, կատարի նոր հրահանգը և դրանից հետո շարունակի ընդհատված առաջադրանքը: Նման միջավայր կարելի է կառուցել LIGHT համակարգի հիման վրա [30], որը կլրացվի խաղային վիճակի վրա ազդեցություն ունեցող գործողություններով և կներառի մի քանի գործակալ:

Եզրակացություն: Խորքային ամրապնդմամբ ուսուցման ոլորտում վերջին տարիներին արձանագրվել են էական առաջխաղացում և նշանակալից հաջողություններ տարաբնույթ խնդիրների լուծման հարցում [3-5]: Այնուամենայնիվ, նշված խնդիրները պարունակում էին միայն մեկ ուսումնառող գործակալ, մինչ-

դեռ իրական կյանքի բազում համակարգեր բաղկացած են մեծ թվով գործակալներից, որոնք միասնական ուժերով փորձում են լուծել ընդհանուր խնդիրը։

Չնակերպելով միագործակալ և համագործակցային բազմագործակալ ԱՈւ խնդիրները՝ ներկայացվել են այն մարտահրավերները, որոնք դժվարին են դարձնում հաջողության հասած միագործակալ ԱՈւ մեթոդների կիրառումը համագործակցային բազմագործակալ համակարգերում։ Նկարագրելով գոյություն ունեցող ԲԱՈւ մեթոդները՝ նշվել են դրանց որոշ թերությունները, և առաջարկվել մի քանի հեռանկարային ուղղություններ՝ ապագա հետազոտությունների կատարման համար։

ԳՐԱԿԱՆՈՒԹՅԱՆ ՑԱՆԿ

1. **Sutton, R.S. and Barto, A.G.** Reinforcement Learning: An Introduction. Second edition. - MIT press, Cambridge, MA, 2018.- 526 p.
2. **Lecun, Y., Bengio Y., and Hinton G.** Deep learning // Nature. -2015.- 521(7553). -P. 436–444.
3. Human-level control through deep reinforcement learning / **V. Mnih, K. Kavukcuoglu, D. Silver, A.A. Rusu, et al** // Nature. -2015.- 518(7540). -P. 529–533.
4. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play / **D. Silver, T. Hubert, J. Schrittwieser, et al** // Science. -2018.- 362(6419). -P. 1140–1144.
5. **Levine, S., Finn, C., Darrell, T., and Abbeel, P.** End-to-end training of deep visuomotor policies // Journal of Machine Learning Research. -2016.- 17(39). - P. 1–40.
6. **Watkins, C.** Learning from delayed rewards: PhD thesis. - University of Cambridge, England, 1989.
7. **Sutton, R.** Learning to predict by the methods of temporal differences // Machine learning. -1988.- 3(1). -P. 9–34.
8. **Watkins, C. and Dayan, P.** Q-learning // Machine Learning. -1992.- P. 279–292.
9. **Van Hasselt, H. Guez, H., Silver, D.** Deep reinforcement learning with double Q-learning // Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence. -2016.- P. 2094-2100.
10. Dueling network architectures for deep reinforcement learning / **Z. Wang, T. Schaul, M. Hessel, et al** // Proceedings of the 33rd International Conference on Machine Learning. -2016.- P. 1995-2003.
11. **Hausknecht, M., Stone, P.** Deep Recurrent Q-Learning for Partially Observable MDPs // AAAI Fall Symposium Series. -2015.
12. **Hochreiter, S. and Schmidhuber, J.** Long short-term memory // Neural computation. -1997.- 9(8). -P. 1735–1780.
13. **Chung, J., Gulcehre, C., Cho, K., and Bengio, Y.** Empirical evaluation of gated recurrent neural networks on sequence modeling // NIPS Workshop on Deep Learning. -2014.

14. **Sutton, R, McAllester, D., Singh, S., and Mansour, Y.** Policy Gradient Methods for Reinforcement Learning with Function Approximation // Advances in Neural Information Processing Systems. -2000.- P. 1057-1063.
15. Continuous control with deep reinforcement learning / **T. Lillicrap, J. Hunt, A. Pritzel, et al** // International Conference on Learning Representations. -2016.
16. Asynchronous methods for deep reinforcement learning / **V. Mnih, A. Badia, M. Mirza, et al** // Proceedings of the 33rd International Conference on Machine Learning. -2016.- P. 1928– 1937.
17. Proximal Policy Optimization Algorithms / **J. Schulman, F. Wolski, P. Dhariwal, et al** // ArXiv -2017.- abs/1707.06347.
18. **Haarnoja, T., Zhou, A., Abbeel, P., Levine, S.** Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor // Proceedings of the 35th International Conference on Machine Learning. -2018.- P. 1861-1870.
19. **Oliehoek, F.A. and Amato, C.A.** Concise Introduction to Decentralized POMDPs. - SpringerBriefs in Intelligent Systems.- Springer, 2016. – 114 p.
20. **Tan, M.** Multi-agent reinforcement learning: Independent vs. cooperative agents // Proceedings of the 10th International Conference on Machine Learning. -1993.- P. 330–337.
21. Counterfactual multiagent policy gradients / **J. Foerster, G. Farquhar, T. Afouras, et al** // Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence. -2018.- P. 2974–2982.
22. Value-Decomposition Networks For Cooperative Multi-Agent Learning / **P. Sunehag, G. Lever, A. Gruslys, et al** // Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems. -2017.- P. 2085-2087.
23. QMIX: Monotonic Value Function Factorisation for Deep Multi-Agent Reinforcement Learning / **T. Rashid, M. Samvelyan, C. Schroeder, et al** // Proceedings of the 35th International Conference on Machine Learning. -2018.- P. 4295– 4304.
24. Multi-agent actorcritic for mixed cooperative-competitive environments / **R. Lowe, Y. Wu, A. Tamar, et al** // Advances in Neural Information Processing Systems. -2017.- P. 6382–6393.
25. **Gupta, J.K., Egorov, M., and Kochenderfer, M.** Cooperative Multi-agent Control Using Deep Reinforcement Learning // Autonomous Agents and Multiagent Systems. - 2017.- P. 66–83.
26. The StarCraft Multi-Agent Challenge / **M. Samvelyan, T. Rashid, C. Schroeder, et al** // Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems. -2019.- P. 2186-2188.
27. **Mordatch, I. and Abbeel, P.** Emergence of Grounded Compositional Language in Multi-Agent Populations // Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence. -2018.- P. 1495 - 1502.
28. **Bordes, A. and Weston, J.** Learning end-to-end goal-oriented dialog // International Conference on Learning Representations. - 2017.

29. Grounded Language Learning in a Simulated 3d World / **K.M. Hermann, F. Hill, S. Greenet, et al** // ArXiv. -2017.- abs/1706.06551.
30. Learning to speak and act in a fantasy text adventure game / **J. Urbanek, A. Fan, S. Karamcheti, et al** // Proceedings of the Conference on Empirical Methods in Natural Language Processing. -2019.

Հայ-Ռուսական համալսարան: Նյութը ներկայացվել է խմբագրություն 25.09.2019:

М.Э. САМВЕЛЯН

КООПЕРАТИВНОЕ ГЛУБОКОЕ МНОГОАГЕНТНОЕ ОБУЧЕНИЕ С ПОДКРЕПЛЕНИЕМ: СЛОЖНОСТИ, ПРИНЦИПЫ И ОТКРЫТЫЕ ЗАДАЧИ

В последние годы область глубокого обучения с подкреплением (ОП) достигла значительных успехов в ряде сложных задач, таких как компьютерные игры, настольные игры и робототехника. Хотя большая часть этого успеха состояла в одноагентном ОП, многие реальные среды состоят из нескольких агентов, которые должны взаимодействовать для решения общих задач. Тем не менее, методы для ОП с одним агентом плохо подходят для сценариев с несколькими агентами из-за некоторых сложностей. В работе подробно рассмотрены проблема ОП и существующие методы ОП, выявлены проблемы использования этих методов в многоагентных средах. Дан обзор текущих алгоритмов многоагентного ОП и тщательно выявлены их ограничения. Описаны некоторые перспективные направления дальнейших работ.

Ключевые слова: обучение с подкреплением, многоагентная система, многоагентное обучение.

M.E. SAMVELYAN

COOPERATIVE DEEP MULTI-AGENT REINFORCEMENT LEARNING: DIFFICULTIES, PRINCIPLES AND OPEN PROBLEMS

In recent years, the field of deep reinforcement learning (RL) has achieved remarkable success in a number of challenging problems, such as computer games, board games and robotics. Although most of this success has been in single-agent RL, many real-world environments consist of multiple agents that need to cooperate in order to solve common tasks. Nonetheless, methods for single-agent RL are poorly suited for multi-agent scenarios due to several difficulties. In this work, the RL problem and the existing RL methods are considered and the difficulties of using these methods in multi-agent environments are revealed. An overview of current multi-agent RL algorithms is then presented and their limitations are carefully identified. Several promising directions for future work are described.

Keywords: reinforcement learning, multi-agent system, multi-agent learning.