

УДК 004.8, 519.83

DOI: [10.26102/2310-6018/2020.28.1.040](https://doi.org/10.26102/2310-6018/2020.28.1.040)

## Алгоритмизация мультиагентного обучения с подкреплением в теоретико-игровых задачах поиска оптимальных стратегий

Е.С. Соколова, К.А. Разинкин

*Воронежский государственный технический университет,  
Воронеж, Российская Федерация*

**Аннотация:** Актуальность темы статьи обусловлена растущим интересом к мультиагентному имитационному моделированию динамических систем различной физической и социальной природы. В настоящее время на первый план выходит концепция интеллектуального агента как имитационной модели поведения активного элемента в сложных ситуациях и стратегиях взаимодействия с другими активными элементами и средой для достижения цели. В общей концепции интеллектуального агента и агентных технологий имитации взаимодействия динамических объектов в направлении достижения цели предлагается метод структурно-параметрического моделирования интеллектуальных агентов и мультиагентных систем с алгоритмами идентификации и прогнозирования состояния агентов, а также программная реализация мультиагентных имитационных моделей производственных, социальных и маркетинговых систем. В этой связи актуальность темы обусловливается необходимостью повышения эффективности мультиагентного обучения с подкреплением в теоретико-игровых задачах поиска оптимальных стратегий. В статье описываются алгоритмы мультиагентного обучения с подкреплением в теоретико-игровых задачах, такие как  $\text{minimax-Q}$ , когда реализуется минимизация возможных потерь из тех, которые агенту нельзя предотвратить при развитии событий по наихудшему для него сценарию и WoLF-PHC (Win or Learn Fast – Policy Hill Climbing), реализующему политику быстрого выигрыша или быстрого обучения. Показаны достоинства и недостатки данных подходов, принципы их модернизации и возможности реализации указанных подходов в средах имитационного моделирования.

**Ключевые слова:** мультиагентное обучение, обучение с подкреплением, стохастические игры, стратегии равновесия.

**Для цитирования:** Соколова Е.С., Разинкин К.А. Алгоритмизация мультиагентного обучения с подкреплением в теоретико-игровых задачах поиска оптимальных стратегий. *Моделирование, оптимизация и информационные технологии*. 2020;8(1). Доступно по: [https://moit.vivt.ru/wp-content/uploads/2020/02/SokolovaSoavtori\\_1\\_20\\_1.pdf](https://moit.vivt.ru/wp-content/uploads/2020/02/SokolovaSoavtori_1_20_1.pdf) DOI: 10.26102/2310-6018/2020.28.1.040

## Algorithmization of multi-agent learning with reinforcement in the game-theoretic problems of finding optimal strategies

E.S. Sokolova, K.A. Razinkin

*Voronezh State Technical University, Voronezh, Russian Federation*

**Abstract:** The relevance of the topic of the article is due to the growing interest in multi-agent simulation of dynamic systems of various physical and social nature. Currently, the concept of an intelligent agent as a simulation model of the behavior of the active element in complex situations and strategies for interaction with other active elements and the environment to achieve the goal is coming to the fore. In the general concept of an intelligent agent and agent technologies for simulating the interaction of dynamic objects in the direction of achieving a goal, a method of structural-parametric modeling of intelligent agents and multi-agent systems with algorithms for identifying and predicting the state of agents, as well as software for multi-agent simulation models of production, social and marketing

systems. In this regard, the relevance of the topic is determined by the need to increase the effectiveness of multi-agent training with reinforcement in the game-theoretic problems of finding optimal strategies. The article describes multi-agent learning algorithms with reinforcement in game-theoretic problems, such as minimax-Q, when minimizing possible losses from those that cannot be prevented by an agent when events develop according to his worst-case scenario and WoLF-PHC (Win or Learn Fast – Policy Hill Climbing), which implements a policy of quick gain or quick training. In this case, the WoLF-PHC algorithm, which is a modification of the PHC algorithm. The algorithm has different learning speeds when winning an agent and a pro-game. Agent training rates vary to maintain algorithm convergence. The main idea of this algorithm is to learn quickly, losing, and slowly, winning. The advantages and disadvantages of these approaches, the principles of their modernization and the possibility of implementing these approaches in simulation environments are shown.

**Keywords:** multi-agent learning, reinforcement learning, stochastic games, equilibrium strategies

**For citation:** Sokolova E.S., Razinkin K.A. Algorithmization of multi-agent learning with reinforcement in the game-theoretic problems of finding optimal strategies. *Modeling, optimization and information technology*. 2020;8(1). Available by: [https://moit.vivt.ru/wp-content/uploads/2020/02/SokolovaSoavtori\\_1\\_20\\_1.pdf](https://moit.vivt.ru/wp-content/uploads/2020/02/SokolovaSoavtori_1_20_1.pdf) DOI: 10.26102/2310-6018/2020.28.1.040 (In Russ.).

## Введение

Поведенческие аспекты взаимодействия агентов в динамической системе могут быть реализованы программно в соответствии с определённым алгоритмическим решением. Однако, как правило, многоагентная система очень сложна, а предварительное программирование системы по практическим причинам невозможно. Кроме того, динамика агентов и окружающей среды не стационарна и, как следствие требуется обучение и адаптация.

В своей оригинальной работе Литман [1] сосредоточил внимание только на двух агентах, которые имели противоположные и противоположные цели, когда они могли бы использовать одни функции вознаграждения, которое один пытался максимизировать, а другой – пытался свести к минимуму. В рациональной многоагентной игре каждый агент должен каким-то образом отслеживать реализации других с тем, чтобы в результате получить согласованное поведение. Кроме того, мы имеем проблему масштабируемости, связанная с тем, что агенты должны отслеживать большое количество возможных состояний и совместных действий. Обучение в стохастических играх может быть формализовано как проблема мультиагентного обучения с подкреплением (MARL), когда агенты выбирают действия одновременно в текущем состоянии и получают награды в следующем. Как правило, в случае MARL агентам не известна переходная функция или функция вознаграждения, зависящие от состояния окружающей среды, при этом рациональность и сходимости являются двумя желательными свойствами для мультиагентных алгоритмов обучения в стохастических играх [2].

В алгоритмах стохастического обучения также присутствует идея конвергенции. Допустим, что все остальные игроки играют в стационарные стратегии, не учась и не меняя своих стратегий, что позволяет адаптироваться к такому поведению и реализовывать сходимости к рациональной стратегии (равновесие по Нэшу).

В [3] формализована дифференциальная игра по охране территории, где защитник пытается перехватить захватчика, прежде чем захватчик достигнет территории. Данная игра рассматривается как стохастическая с нулевой суммой где защитник играет против захватчика в сети. Дальнейшее изложение посвящено рассмотрению именно этой игры с использованием алгоритмов MARL, применительно к которой использовались два

алгоритма обучения с подкреплением в данной игре, и проверяется производительность этих алгоритмов обучения на предмет сходимости и рациональности.

### Методы исследования

В работе [1] предлагается алгоритм минимаксного Q-обучения, специально разработанный для стохастических игр с нулевой суммой для двух игроков. Алгоритм использует принцип минимакса, который нужно реализовать для стратегий равновесия Нэша и значения состояний для стохастических игр с нулевой суммой для двух игроков. Похожий на метод Q-learning, алгоритм *minimax-Q* представляет собой обобщение разностных методов обучения метод, который выполняет обратное распространение на значениях состояний или пар состояний-действий.

Второй алгоритм – политика восхождения (PHC) – это простой практический алгоритм, в который можно играть с использованием смешанных стратегий. Этот алгоритм был впервые предложен Боулингом и Велозо [4, 5]. PHC представляет собой простую модификацию алгоритма Q-обучения для одного агента, где поиск выполняется в PHC-алгоритме в пространстве смешанных стратегий и состоит из двух частей – первой – реализующей обучение с подкреплением, так как алгоритм Q-обучения поддерживает значения конкретных действий в состоянии и второй – теоретической части игры, в которой текущая стратегия сохраняется в каждом состоянии системы.

Алгоритм WoLF-PHC (политика быстрого выигрыша или быстрого обучения) – это расширение алгоритма PHC [5]. Этот алгоритм использует механизм WoLF, так что алгоритм PHC сходится к равновесию Нэша в самостоятельной игре. Алгоритм имеет две разные скорости обучения:  $\delta_\omega$  когда алгоритм выигрывает, а  $\delta_l$  когда проигрывает. Разница между средней текущей стратегиями используется в качестве критерия для принятия решения, когда агент выигрывает или проигрывает. Скорость обучения  $\delta_l$  больше, чем скорость обучения  $\delta_\omega$ . Таким образом, когда агент проигрывает, он учится быстрее, чем, когда он побеждает, что заставляет агента быстро адаптироваться к изменениям в стратегиях других агентов и также дает другим агентам время для адаптации к стратегии агента.

Алгоритм WoLF-PHC обладает свойством сходимости, поскольку это заставляет агента сходиться к одному из равновесий Нэша. Алгоритм также является рациональным алгоритмом обучения, так как он гарантирует сходимость стратегии агента к оптимальной, когда у его противника стационарная игровая стратегия. Эти свойства позволяют широко использовать алгоритм WoLF-PHC к различным стохастическим играм [6-12].

### Результаты

Алгоритм минимакс-Q может быть реализован в виде следующей схемы:

В стохастической игре с нулевой суммой для двух игроков, учитывая текущее состояние  $s$ , мы определяем функцию значения состояния для игрока  $i$  как

$$V_i^*(s) = \max_{\pi_i(s, \bullet)} \min_{a_{-i} \in A_{-i}} \sum_{a_i \in A_i} Q_i^*(s, a_i, a_{-i}) \pi_i(s, a_i) \quad (i = 1, 2) \quad (1)$$

где  $-i$  обозначает противника  $i$ -го игрока,  $\pi_i(s, \bullet)$  обозначает все возможные стратегии игрока  $i$  в состоянии  $s$ , и  $Q_i^*(s, a_i, a_{-i})$  ожидаемая награда когда  $i$  – ый игрок и его

противник выбирают действие  $a_i \in A_i$  и  $a_{-i} \in A_{-i}$ , после этого осуществляется следование стратегиям равновесия по Нэшу. Если известно  $Q_i^*(s, a_i, a_{-i})$ , мы можем решить уравнение (1) и найти стратегию равновесия по Нэшу  $i$ -го игрока  $\pi^*(s)$ . Для проблем мультиагентного обучения с подкреплением (MARL),  $Q_i^*(s, a_i, a_{-i})$  неизвестна игрокам и минимакс-Q алгоритм может гарантировать сходимость к равновесию по Нэшу, если все возможные состояния и возможные действия игроков реализуются бесконечно часто. Недостатком алгоритма является то, что имеется необходимость использования линейного программирования для решения  $\pi_i(s)$  и  $V_i(s)$  на каждой итерации в алгоритме, что приводит к замедлению процесса обучения. Также для того, чтобы выполнить линейное программирование, игрок должен знать пространство действий противника.

Используя алгоритм  $\text{minimax-Q}$ , агент всегда будет реализовывать безопасную стратегию в случае худшего сценария, инициированного противником. Однако, если противник в настоящее время играет стационарную стратегию, не являющуюся стратегией равновесия, алгоритм минимакс-Q не может реализовать адаптацию поведения игрока к изменению стратегии противника. Причина в том, что алгоритм минимакс-Q является независимым от оппонента и будет сходиться к стратегии равновесия Нэша игрока, независимо от того, какую стратегию использует противник.

В целом, алгоритм минимакс-Q применим к стохастическим играм с нулевой суммой, не удовлетворяющих свойству рациональности, но удовлетворяющих свойству сходимости.

Рассмотрим алгоритм Policy Hill Climbing (PHC). Необходимо отметить, что вероятность выбора наиболее ценных действий увеличивается на малой скорости обучения  $\delta \in 0,1$ , так что политика улучшается. Алгоритм эквивалентен Q-Learning, когда реализуется «жадная» политика с вероятностью  $\delta = 1$  при выполнении самого ценного действия. Алгоритм рационален и сходится к оптимальному решению, когда за фиксированной (стационарной) стратегией следуют другие игроки. Однако алгоритм PHC может не сходиться к стационарной политике, если другие игроки обучаются, не смотря на то, что его средняя награда будет сходиться к награде равновесия Нэша. Алгоритм PHC реализуется в виде ряда шагов.

1. Инициализация  $Q_i(s, a_i) \leftarrow 0$  и  $\pi_i(s, a_i) \leftarrow \frac{1}{|A_i|}$ . Выбор скорости обучения

$\alpha, \beta$  и коэффициент дисконтирования  $\gamma$ .

2. Начало цикла for.

3. Выбор действия  $a_c$  из текущего состояния  $s$  на основе смешанной стратегии.

4. Выполнение действия  $a_c$ , получение награды  $r_i$  и переход в последующее состояние  $s'$

5. Обновление  $Q_i(s, a_c)$

$$Q_i(s, a_c) = Q_i(s, a_c) + \alpha \left[ r_i + \gamma \max_{a_i} Q_i(s', a_i) - Q_i(s, a_c) \right] \quad (2)$$

где  $a_i'$  действие  $i$ -го игрока в следующем состоянии  $s'$ ,  
 а  $a_c$  действие игрока в состоянии  $s$ .

6. Обновите  $\pi_i(s, a_i)$

$$\pi(s, a_c) = \pi_i(s, a_i) + \Delta_{sa_i} \quad (\forall a_i \in A_i) \quad (3)$$

где

$$\Delta_{sa_i} = \begin{cases} -\delta_{sa_i} & \text{если } a_c \neq \arg \max_{a_i \in A_i} Q_i(s, a_i) \\ \sum_{a_j \neq a_i} \delta_{sa_j} & \text{в противном} \end{cases},$$

$$\delta_{sa_j} = \min \left( \pi_i(s, a_i), \frac{\delta}{|A_i| - 1} \right).$$

7. Конец цикла

Рекурсивный Q-learning обучаемого агента  $j$  реализуется как

$$Q_{t+1}^j(s, a) = (1 - \alpha)Q_t^j(s, a) + \alpha(r^j + \gamma \max_a Q_t^j(s', a')) \quad (4)$$

Далее рассмотрим алгоритм WoLF-PHC для обучающего агента, и алгоритм обновления его стратегии на основе уравнения (5):

$$\pi_{t+1}^j(s, a) = \pi_t^j(s, a) + \Delta_{sa} \quad (5)$$

где

$$\Delta_{sa} = \begin{cases} -\delta_{sa} & \text{если } a \neq \arg \max_a Q_t^j(s, a') \\ \sum_{a' \neq a} \delta_{sa'} & \text{в противном случае} \end{cases}$$

$$\delta_{sa} = \min \left( \pi_t^j(s, a), \frac{\delta}{|A_j| - 1} \right)$$

$$\delta = \begin{cases} \delta_0, & \text{если } \sum_a \pi_t(s, a') Q_{t+1}^j(s, a') > \sum_a \bar{\pi}_{t+1}(s, a') Q_{t+1}^j(s, a') \\ \delta_1 & \text{в противном случае} \end{cases}$$

$$\bar{\pi}_{t+1}(s, a') = \bar{\pi}_{t+1}(s, a') + \frac{1}{C_{t+1}(s)} \left( \pi_t^j(s, a') - \bar{\pi}_t^j(s, a') \right) \quad \forall a' \in A_j$$

$$C_{t+1}(s) = C_t(s) + 1$$

Алгоритм WoLF-PHC представлен на Рисунке 1.

В отличие от ранее упомянутого алгоритма обучения, WoLFPHC алгоритм не учитывает стратегии других игроков и их действия. Следовательно, по сравнению с другими алгоритмами обучения, алгоритм WoLF-PHC требует меньше информации из окружающей среды.

Поскольку алгоритм WoLF-PHC основан на методе PHS, в отличие от последнего не требуется ни линейное ни квадратичное программирование. Результаты моделирования в иллюстрируют сближение игроков стратегии, тщательно выбирая скорость обучения в соответствии с различными примерами в матричных и стохастических играх.

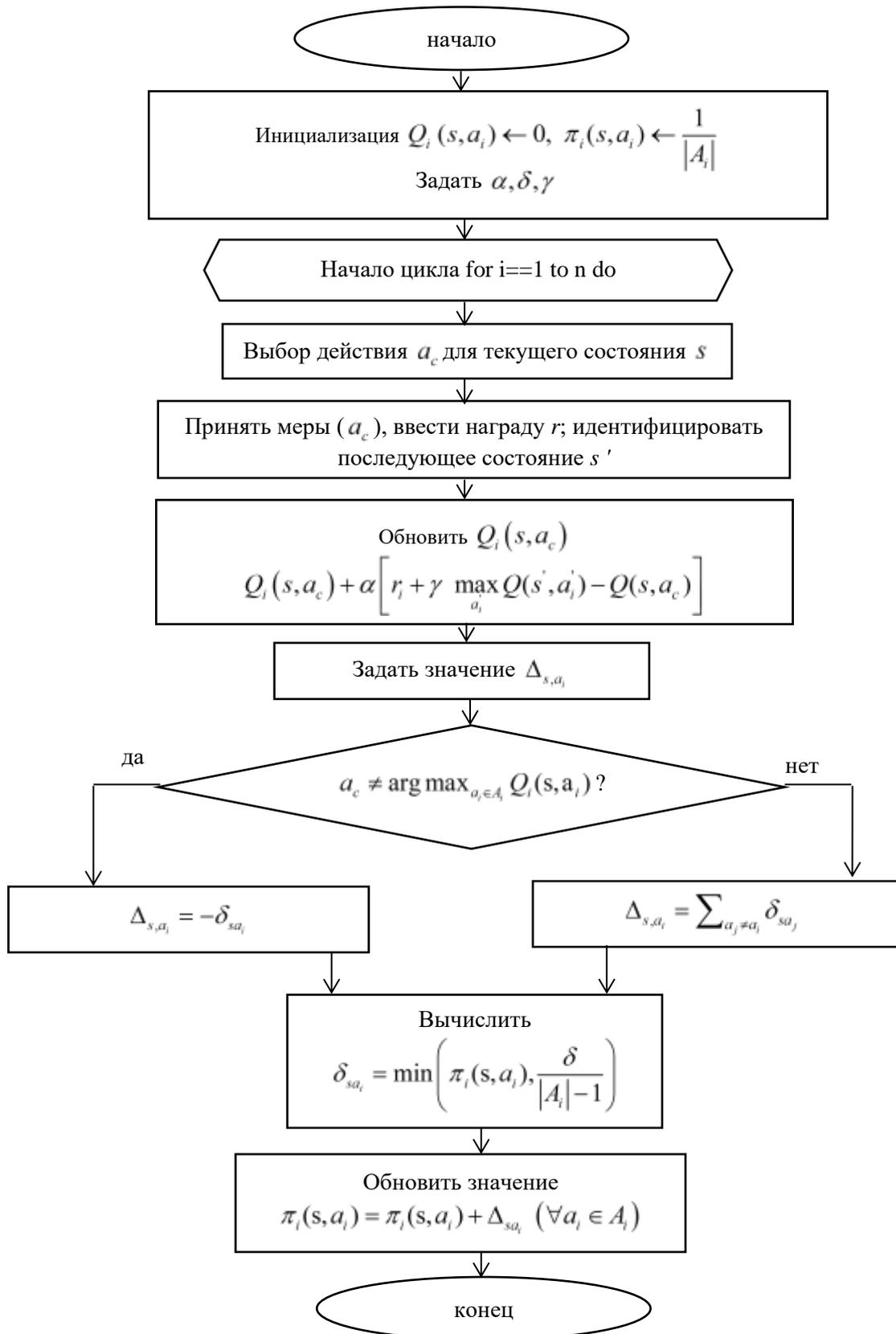


Рисунок 1 – Алгоритм WoLF-PHC  
 Figure 1 - WoLF-PHC Algorithm

### Обсуждение

Рассмотренные алгоритмы  $\text{minimax-Q}$  и WoLF-PHC, реализованы в рамках имитационного моделирования в среде Matlab при имитации сетевой игры по охране территории [11]. Реализована простая сетевая игра для исследования проблем смешанных стратегий, рациональности и конвергенции.

Использовались две симуляции для каждой игры сетки. В первом моделировании игроки в игре используют один и тот же алгоритм обучения для игры друг против друга. При этом исследовались свойства сходимости.

Во второй симуляции фиксировались стратегии одного игрока, а второй игрок определял оптимальную стратегию против первого. Использовались алгоритмы  $\text{minimax-Q}$  и WoLF-PHC для обучения и сравнивались показатели обученности игрока с использованием алгоритма минимакс-Q игрок и WoLF-PHC. Согласно свойству рациональности показано, что при обучении с использованием WoLF-PHC защитник имеет лучшую производительность, чем с использованием  $\text{minimax-Q}$  обучения защитника во второй симуляции. В качестве примера на Рисунках 2 и 3 представлены стратегии с использованием двух алгоритмов на примере первой симуляции.

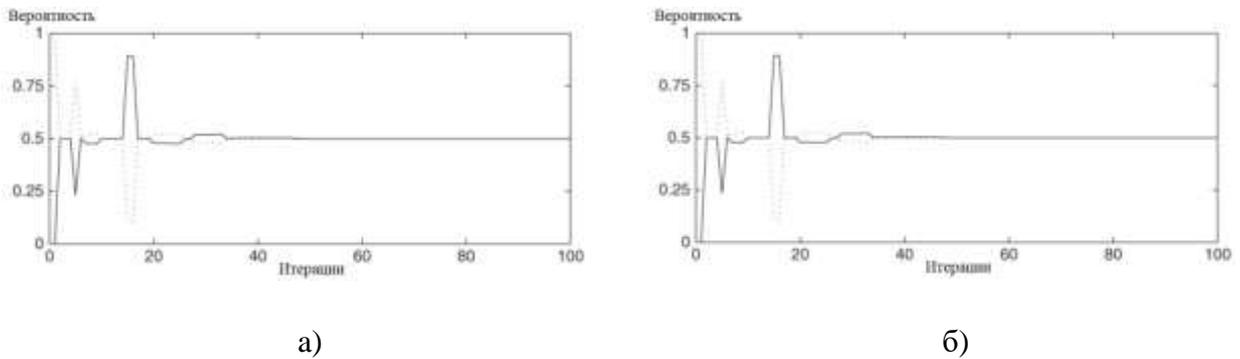


Рисунок 2 – Стратегии игроков в состоянии с использованием алгоритма минимакс-Q в первой симуляции для сетки игры  $2 \times 2$

- а) Стратегия защитника  $\pi_D(s_1, a_{left})$ , (сплошная линия) и  $\pi_D(s_1, a_{up})$  (пунктирная линия);  
 б) стратегия захватчиков  $\pi_I(s_1, o_{down})$  (сплошная линия) и  $\pi_I(s_1, o_{right})$  (пунктирная линия) [8]

Figure 2 – Strategies of players in a state using the minimax-Q algorithm in the first simulation for a  $2 \times 2$  game grid

- a) Defender's strategy  $\pi_D(s_1, a_{left})$ , (solid line) and  $\pi_D(s_1, a_{up})$  (dashed line);  
 b) the invaders' strategy  $\pi_I(s_1, o_{down})$  (solid line) and  $\pi_I(s_1, o_{right})$  (dashed line) [8]

Далее рассматривалась реализация алгоритма WoLF-PHC применительно к сетевой игре [12]. В соответствии с настройками параметров в [2], мы устанавливаем обучение оценки  $\alpha$  как  $1/(10 + t/10000)$ ,  $\delta_\omega$  как  $1/(10 + t/2)$  и где  $\delta_l$  как  $3/(10 + t/2)$ , где  $t$  – это номер текущей итерации. Количество итераций обозначает сколько раз шаг повторяется в алгоритме.

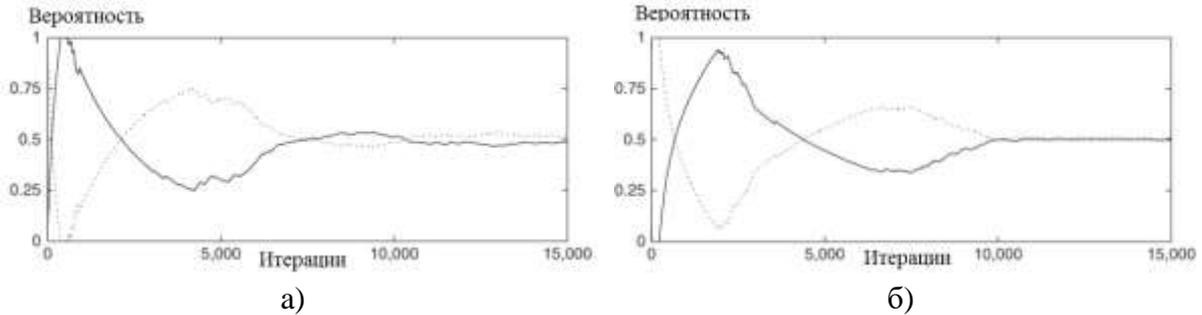


Рисунок 3 – Стратегии игроков в состоянии  $s_1$ , используя WoLF-PHC Алгоритм в первой симуляции для сетки игры

а) Стратегия защитников  $\pi_D(s_1, a_{left})$  (сплошная линия) и  $\pi_D(s_1, a_{up})$  (пунктирная линия); б) Стратегия захватчиков  $\pi_D(s_1, o_{down})$  (сплошная линия) и  $\pi_D(s_1, o_{right})$  (пунктирная линия)

Figure 3 – Strategies of players in state  $s_1$  using WoLF-PHC Algorithm in the first simulation for the game grid

a) Defenders Strategy  $\pi_D(s_1, a_{left})$  (solid line) and  $\pi_D(s_1, a_{up})$  (dashed line);  
 b) Invader strategy  $\pi_D(s_1, o_{down})$  (solid line) and  $\pi_D(s_1, o_{right})$  (dashed line)

Результат Рисунка 3 показывает, что стратегии игроков сходятся близко к равновесию по Нэшу после 15 000 итераций.

### Заключение

В работе были рассмотрены алгоритмы мультиагентного обучения с подкреплением в теоретико-игровых задачах, такие как  $\minimax$ -Q, и WoLF-PHC (Win or Learn Fast – Policy Hill Climbing), реализующего политику быстрого выигрыша или быстрого обучения. Предложен обобщённый алгоритм WoLF-PHC с учётом нивелирования недостатка метода, связанного с варьированием скорости обучения агентов.

Экспериментальные исследования, проведённые в режиме имитационного моделирования применительно к сетевым стохастическим играм, показали достаточную степень сходимости.

Перспективами для дальнейших исследований является расширение и модификация данных методов к сетевым играм на большее число агентов, для исследования возможности идентификации и прогнозирования состояния и поведения социальной сети как активной системы, являющейся результатом многошаговых взаимодействий элементов системы и окружающей среды.

### ЛИТЕРАТУРА

1. Littman M.L. Markov games as a framework for multi-agent reinforcement learning, in 11<sup>th</sup> International Conference on Machine Learning (New Brunswick, United States), July 1994:157-163.
2. Bowling M. and Veloso M. Multiagent learning using a variable learning rate. *Artificial Intelligence*. 2002;136(2):215-250.
3. Isaacs. Differential Games: A Mathematical Theory with Applications to Warfare and Pursuit, Control and Optimization. New York, New York: John Wiley and Sons, Inc. 1965.

4. Sutton R.S. and Barto A.G. Reinforcement learning: An Introduction. Cambridge, Massachusetts: The MIT Press, 1998.
5. Bowling M. Multiagent Learning in the Presence of Agents with Limitations. PhD thesis, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, May 2003.
6. Соколова Е.С. Мультиагентный подход к моделированию межмодульных взаимодействий в стохастических сетевых распределённых системах. *Системы управления и информационные технологии*. 2020;1(79):67-71.
7. Ивашкин Ю.А. Мультиагентное моделирование в имитационной системе Simplex3. Учебное пособие. М.: Лаборатория знаний: Лаборатория Базовых Знаний. 2016:361.
8. Lu X. On Multi-Agent Reinforcement Learning in Games. Ph.D. Thesis Carleton University, Ottawa, ON, Canada. 2012.
9. Littman M.L., Szepesvári C. A generalized reinforcement learning model: Convergence and applications. *Proceedings of the 13th International Conference on Machine Learning, (Bari, Italy)*. July 1996:310-318.
10. Hu J., Wellman M.P. Multiagent reinforcement learning: theoretical framework and an algorithm. *Proceedings of the Fifteenth International Conference on Machine Learning (ICML 1998), Madison, Wisconsin, USA, July 24-27*. 1998:242-250.
11. Hu J., M. P. Wellman M.P. Nash q-learning for general-sum stochastic games. *Journal of Machine Learning Research*. 2003;4:1039-1069.
12. Schwartz H.M. Multi-agent machine learning: a reinforcement approach. By John Wiley & Sons, Inc. 2014:315.

## REFERENCES

1. Littman M.L. Markov games as a framework for multi-agent reinforcement learning, in 11<sup>th</sup> International Conference on Machine Learning (New Brunswick, United States), July 1994:157-163.
2. Bowling M. and Veloso M. Multiagent learning using a variable learning rate. *Artificial Intelligence*. 2002;136(2):215-250.
3. Isaacs. Differential Games: A Mathematical Theory with Applications to Warfare and Pursuit, Control and Optimization. New York, New York: John Wiley and Sons, Inc. 1965.
4. Sutton R.S. and Barto A.G. Reinforcement learning: An Introduction. Cambridge, Massachusetts: The MIT Press, 1998.
5. Bowling M. Multiagent Learning in the Presence of Agents with Limitations. PhD thesis, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, May 2003.
6. Sokolova E.S. A multi-agent approach to modeling intermodular interactions in stochastic distributed network systems. *Management Systems and Information Technology*. 2020;1(79):67-71. (In Russ.).
7. Ivashkin Yu.A. Multi-agent modeling in the Simplex3 simulation system. Tutorial. М.: Laboratory of knowledge: Laboratory of Basic Knowledge. 2016:361. (In Russ.).
8. Lu X. On Multi-Agent Reinforcement Learning in Games. Ph.D. Thesis Carleton University, Ottawa, ON, Canada. 2012.
9. Littman M.L., Szepesvári C. A generalized reinforcement learning model: Convergence and applications. *Proceedings of the 13th International Conference on Machine Learning, (Bari, Italy)*. July 1996:310-318.
10. Hu J., Wellman M.P. Multiagent reinforcement learning: theoretical framework and an algorithm. *Proceedings of the Fifteenth International Conference on Machine Learning (ICML 1998), Madison, Wisconsin, USA, July 24-27*. 1998:242-250.

11. Hu J., M. P. Wellman M.P. Nash q-learning for general-sum stochastic games. *Journal of Machine Learning Research*. 2003;4:1039-1069.
12. Schwartz H.M. Multi-agent machine learning: a reinforcement approach. By John Wiley & Sons, Inc. 2014:315.

#### ИНФОРМАЦИЯ ОБ АВТОРАХ / INFORMATIONS ABOUT AUTHORS

**Соколова Елена Сергеевна**, старший преподаватель кафедры систем информационной безопасности ФГБОУ ВО «Воронежский государственный технический университет», Воронеж, Российская Федерация.  
*email:* [lenoks.sokolova@mail.ru](mailto:lenoks.sokolova@mail.ru)

**Elena S. Sokolova**, Senior lecturer, Department of information security systems, Voronezh state technical University, Voronezh, Russian Federation.

**Разинкин Константин Александрович**, доктор технических наук, профессор, профессор кафедры систем информационной безопасности ФГБОУ ВО «Воронежский государственный технический университет», Воронеж, Российская Федерация.  
*email:* [kostyr@mail.ru](mailto:kostyr@mail.ru)

**Konstantin A. Razinkin**, Doctor of Technical Sciences, Professor, Professor of the Department of information security systems, Voronezh state technical University», doctor of technical Sciences, associate professor, Voronezh, Russian Federation.