Aggregating Actor-Critic Value Functions to Support Military Decision-Making

Lauri Vasankari[1,3(✉)] and Kai Virtanen[1,2]

[1] Department of Military Technology, National Defence University, 00860 Helsinki, Finland
lauri.vasankari@gmail.com
[2] Department of Mathematics and Systems Analysis, Aalto University, 02150 Espoo, Finland
[3] Department of Computing, University of Turku, 20014 Turku, Finland

Abstract. Reinforcement learning (RL) is used for finding optimal policies for agents in respective environments. The obtained policies can be utilized in decision support, i.e. suggesting or determining optimal actions for different states or observations in the environment. An actor-critic RL method combines policy gradient methods with value functions, where the critic estimates the value function, and the actor updates the policy as directed by the critic. Usually, the utility is the policy learned by the actor. However, if the environment is defined accordingly, the approximated value function can be used to assess, e.g., an optimal solution for placing military units in an operational theatre. This paper explores the use of the critic as the primary output as a decision-support tool, presenting an experiment in a littoral warfare environment.

Keywords: actor-critic · value function · reinforcement learning

1 Introduction

Optimal decision-making [5] aims to find the best available action for the observed situation or state, w.r.t. the result of the action that is measured with some reward mechanism. In a military context, a frequently occurring example is the positioning of resources w.r.t. the estimated course of action of the opposing side.

If the operational environment is formulated into a computable form, such as a Markov decision process (MDP) [17], partially observable MDP (POMDP) or a stochastic game [16], reinforcement learning algorithms can be used to find optimal policies in said environment.

A warfare scenario can be formulated as a partially observable stochastic game (POSG) to mirror the uncertainties related to warfare. Partial observability is integral to warfighting, as at least opponents' plans, force composition and factual capabilities are typically concealed and thus only partially observable.

H. Moosaei et al. (Eds.): DIS 2024, LNCS 14661, pp. 141–152, 2025.
https://doi.org/10.1007/978-3-031-81010-7_10

The uncertainties related to weather and atmospheric phenomena, human error, technical malfunctions and unreliabilities induce stochasticity in the environment.

Additionally, warfighting usually consists of multiple units on both sides. As such, the POSG can be perceived as a multi-agent reinforcement learning (MARL) [1] problem. While RL aims to optimize an agent's policy, MARL has several agents with cooperating or competing policies or both. A warfighting scenario between two sides and several units can be perceived as a multi-agent POSG where two sides compete by trying to optimize their cooperative policies.

Actor-critic algorithms [17] use a *critic* to approximate the value function and guide the updates on the policy conducted by the *actor*. Usually, the resulting actor policy is the desired end state of the algorithm, as the policy guides the agents to select optimal actions for the states it observes. However, if a warfighting scenario is formulated as described above, the resulting value function can produce numeric values for different battlefield situations to support planning and decision-making without following a strict policy.

While a MARL policy can be used to guide decision-making with direct action selection, doctrines necessitate human decision-making. Action selection can be done under supervision, but leveraging the value function for tactical scenario evaluation is practical in combining multiple policies and providing insight into the underlying state space values.

2 Earlier Research

To evaluate battlefield situations and to predict outcomes, Dupuy [4] created the Quantified Judgement Analysis (QJM) model to predict the outcomes of future battles based on analysis of historic battles. The QJM model enables analyzing the parameters and factors that have influenced the results.

RL and semi-Markov decision problems have been examined by Mattila et al. [9] regarding fighter aircraft maintenance under conflict conditions, obtaining policies that optimize maintenance based on aircraft states.

Review by Rempel et al. [14] examines approximate dynamic programming applications within military operations research, viewing Markov decision processes in dynamic programming and approximate dynamic programming (ADP), listing seventeen articles on ADP within military operations research from 1995 to 2021, encompassing application areas of investing and planning, force structure analysis, personnel sustainment, situational awareness, missile defence, air combat, battlefield strategy, airlift, weapon target assignment, combat medical evacuation, illegal fishing patrols, and inventory routing. Out of the reviewed articles, Sztykgold et al. [18] is closest to the subject of decision support, as they have examined assisting a military decision-maker in battlefield tactics by modelling a conflict as a game and using multi-valued graphs to find an optimal path given the uncertainty related to enemy movements and actions. In brief, Sztykgold et al. have modelled the battlefield as nodes and vertices. Armies can either move or fire, and the mission is represented by a location to be reached within

a specific time while preserving some minimum strength ratio between friendly and opposing armies. Using an algorithm based on temporal difference learning, i.e. $TD(\lambda)$, the authors found experimentally that the algorithm returned an optimal control in more than 84% of trials in a simple scenario of 16 vertices and a 1:2 strength ratio between enemies and allies.

The proposed solution shares the same idea as spatial decision analysis [6], where the decision-makers task is aided by evaluating alternative consequences that vary over a spatial region. The underlying problem foundation is the same, as most military problems involved with force projection and decision-making deal with issues of different possibilities and likely outcomes depending on the spatial region.

While the aforementioned research shares a similar goal of supporting decision-making with quantified analysis, to our best knowledge, there are no attempts to utilize the critic from actor-critic algorithms in evaluating the state space post-training.

The contribution of this paper is in examining the use of the critic function and proposing a framework to enable quantifying the values of different states in the environment without following a certain policy *per se*.

The optimal solution for the formulated problem is an agent that can generalize and adapt to changing opponent force composition, tactics and operational goals. As the critic functions solely as a bootstrapping method [17] to guide the learning process, there is no intuitive need to utilize it as we propose.

However, general and complex agents' decision-making is not transparent, and thus using the agents' suggestions to guide the DMP can be difficult. Instead, using the approximated value function enables us to estimate how an aggregated global value function perceives the formulated state space comprehensively and transparently. The underlying tactical optimality can be assessed by having the sub-models that produce measurable and visualizable policies, i.e. tactics. Simply aggregating the working policies can, however, lead to unpredictable results. Instead, aggregating the approximated value functions allows the production of a decision-making support tool that evaluates the selected battleground against a partially unforeseen i.e. unobservable opponent and its tactics.

This paper extends the state-value approach of tabular reinforcement learning methods [17] to more complex environments while producing a similar solution.

3 Environment Evaluation Framework

The use of value functions in optimal decision-making in itself is nothing new. Value function methods are also integral in RL, where the value of each state or state-action pair is learned to guide the choice of the next action. In a large discrete or continuous state space, the thorough exploration of the state-value space is infeasible; therefore, value function approximation is used. In deep RL, a neural network is used to approximate the value function to produce an agent that can select favourable next actions.

In warfighting, the command plans the operation by assessing different courses of action. The knowledge base and tactical insight guide the drafted alternatives, which are evaluated to select the course of action that is perceived as optimal w.r.t. the mission objective and available resources. A fundamental subproblem is positioning the available resources, i.e., units, in the operation area. It is not trivial to formulate the tactical insight that identifies and takes into consideration all the factors involved in the positioning of units.

In a recent MARL experiment [20], a simplistic POSG was solved with MARL, producing alternative courses of action. Due to the non-stationarity and stochastics, the results may deviate from known tactical principles to a great extent. Therefore, instead of utilizing the learned policies, the learned value function can be used to estimate the solutions proposed by human decision-makers, enhancing the tactical decision-making in the same manner as the critic guides the actors learning in actor-critic algorithms.

The warfighting POSG with multiple agents with different policies can be formulated as $[S_n, A_k, P_e, P_a, R]$ where S_n denotes the possible states in the environment, A_k the actions available for the agents, P_e the stochastic changes in the environment, P_a the probabilities of successful actions and R denotes the joint reward. In a game with two sides, the agents B and O have respective states and available actions for each agent. The agents produce joint observations, i.e. the observations are shared between agents on the same side. The actions are chosen with a transition function, which determines the action based on joint, partial observations. When formulating a warfighting scenario, the goal is to defeat the opponent; the victory can be anything from avoiding contact to annihilation of opposing forces. The threshold for victory has to be decided to formulate the reward function to utilize RL.

The key difference between learning the value function explicitly and leveraging an actor-critic model critic is the dual purpose: the learned policy can be evaluated in itself, while the value function can be used to estimate the value of static, one-shot battlefield situations that either comply or differ from the learned policy. The evaluation of the policy can amplify the decision-makers perception of the quality of the found solution, and the value function can be used to estimate the ideas of the decision-maker that deviate from the found solution.

The use of the approximated value function requires that the input of the value function is aligned with the purpose. To evaluate the combined value of unit positions, the critic input needs to consist of the combined state space of the agents that also represents the state space used in planning. Hence, the global state $s_g = \left[s_1, s_2, \cdots, s_n\right]$ where $s_1, \cdots, s_n \in S$ are local states of the n agents. Otherwise, if the critic uses sole local states as input, it can be used to evaluate individual positions without regard for one another. Other information the critic uses in approximating the value function must not conflict with the purpose, i.e. other relevant information can be zeroed if no data is available.

For example, we formulate the environment as a game grid which depicts the chosen battlefield. In a MARL setting, both sides are trained to engage one

another to reach a victory. The resulting policy of the training is the tactical solution, which represents one available course of action. The decision-maker can decide to utilize the policy to plan the operation as directed by the policy, but instead, we propose to use Monte Carlo (MC) [11] methods to produce a decision-making solution for the environment.

An issue arises in the exploration of the state space and generalization to unpredicted or new situations. If the algorithm converges to a solution without thoroughly exploring the state space, the value function is not able to support the decision-maker extensively. Likewise, the decision-maker may be unsure of the opponent's force composition, goal and tactics. Assuming that the strategies of both sides reach a Nash equilibrium [12] excludes this problem. Still, the assumption of Nash equilibrium in the complex and continuous environment of warfighting can be deemed unrealistic. Additionally, the agent model needs to be complex to be able to generalize to multiple scenarios and tactics. Complexity is computationally expensive, often non-explainable [2] and the exploration issue persists. Aggregation methods used in Federated learning (FL) [7,10] can be utilized as a solution. In a federated learning framework, the local models are aggregated into a global model through, e.g., averaging model parameters over several agents. Thus, a general policy can be formed between several differing policies. However, an averaged tactical policy solution may be unpredictable and consist of aggregating inconsistent, even opposing policies.

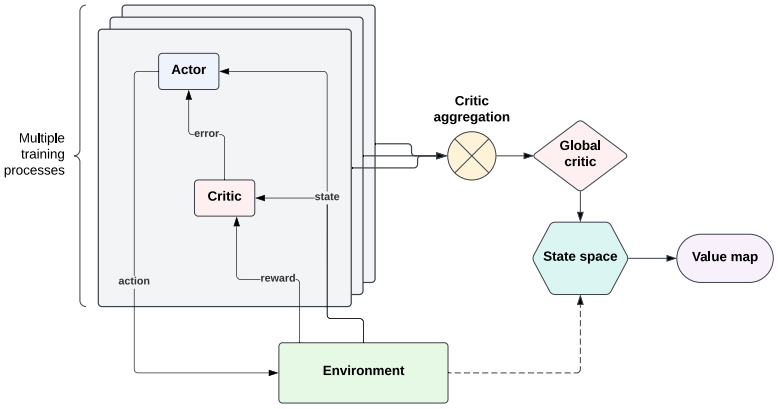

Fig. 1. Proposed framework

Therefore, we suggest that a decision-making support tool is aggregated by using FL aggregation methods to combine multiple different value functions into a global value function, which is then used to produce a mapping of the values of different states in the environment. The semantic process is described in Fig. 1. The different value functions should represent the diverse scenarios that reflect the uncertainty regarding the opponent. As an example, the value functions may be by-products of a landing operation scenario, an air raid and a surface warfare encounter.

4 Aggregated Decision-Making Process Critic

4.1 ADMP-Critic Algorithm

The proposed solution, labelled as ADMP-Critic (Aggregated Decision-Making Process Critic), consists of using an actor-critic algorithm such as DDPG [8] or PPO [15] to find working policies in the environment, aggregating the resulting critics and producing a mapping of the values of the environment state space. This process is represented in the following Algorithm 1.

Algorithm 1. Aggregated DMP-Critic

PHASE 1
Initialize ϕ_g
for $i \in scenarios$ **do**
  Initialize θ_i and ϕ_i
  Train θ_i and ϕ_i with chosen algorithm
  Append $\Phi \leftarrow \phi_i$
end for

PHASE 2
Assign weights w_i for aggregation
for $\phi_i \in \Phi$ **do**
  Aggregate $\phi_g + w_i\phi_i$
end for

PHASE 3
Initialize value grid X
for m iterations **do**
  Sample $s_n \in S$ with $p \sim U(S)$
  Concatenate global state S_g
  $v = \phi_g(S_g)$
  for $position_a \in s_a$ **do**
    $X[position] += \frac{v}{n_{agents}}$
  end for
end for
Process X to usable format

In Algorithm 1, the global model ϕ_g is initialized with model weights set to zero. Then, actor θ_i and critic ϕ_i networks are trained according to a chosen algorithm in every chosen, plausible scenario enabled by the reinforcement learning environment. For consistency, the training should be performed multiple times for each environment to minimize the effect of stochasticity on the results.

After the local models have been trained, the aggregation weights w_i are assigned according to decision-maker preference. The aggregation can average over the number of local models so that $w_i = \frac{1}{|\Phi|}$ where Φ denotes the set of

local models. Optionally, the decision-maker can assign specific weights to highlight the importance of a certain scenario. Other FL methodologies for aggregation [7,13] are also applicable. It is however important to notice that solely the aggregation is regarded in this problem formulation. The whole FL framework of interacting agents and continuous updates of global and local models is not applicable unless applied to a dynamic solution setting. For now, the proposed framework aims to enhance a planning process instead of a dynamic coordination of live operations.

Once the global model, i.e. the general value function has been aggregated, it outputs the value for states in the environment w.r.t. decision-makers bias. The global model is used to map the values to produce a suggestion for optimal locations in the environment w.r.t. previous assumptions.

The value estimation can be executed in a manner that suits the environment. While actor-critic algorithms are suited for continuous action spaces, the state space may be continuous, discrete or discretized. For environments with a finite and relatively small state space, the solution can be calculated combinatorially by creating a set of all possible agent state spaces and passing the set to the value function. In a large discrete state space or continuous state space, the mapping can be produced with MC simulation. In Algorithm 1, the MC sampling is done with uniform distribution over all possible states in the spatial region. The MC produces, in essence, a probability distribution mapped over the environment terrain. The probability distribution functions as the decision-making support tool for the decision-maker to analyze the probable value of positioning available units to encounter the opponent.

Other methods for value mapping can also be used, such as search algorithms, given that initial locations are known or given. Then, the global value model can be used to find the spatial region with the highest value.

4.2 Assumptions

The proposed algorithm has several assumptions regarding the environment, agent observations, formulation of global state space and other features. The proposed algorithm functions with any MARL environment, but it is useful only if there exists a need to determine the approximated value of different spatial states in the environment w.r.t. known and unknown variables. When the variables with unknown values are neglected, the resulting value function mappings evaluate the perceived situation w.r.t. available information. The evaluation is based on the combined experience learned by the aggregated agent policies.

For the proposed use case, the agent observations need to include at least the unit's position in the state space. The global state has to include the position of all units. Other unit observation features enrich the local and global state spaces. These also increase the complexity of the value mapping if included in the MC simulation.

The value function approximation assumes that the environment rewards are calculated with the Bellman equation [3], in which the value of a state is

dependent on the succeeding state and its value. With this dependency, the state values reflect the cumulative, discounted values over succeeding states.

4.3 Experiment

To demonstrate the use of the proposed algorithm, this paper uses a littoral warfare RL environment [19]. The environment comprises a 100×100 game grid resembling the Baltic Sea, where multiple units on two sides engage in littoral surface combat. The tactical goals and unit compositions can be altered, from landing operations to areal control. The state and action space are initially partially discrete and non-discrete values are discretized for action execution.

Utilizing a littoral warfare environment and agents trained with MAPPO [21], the Algorithm 1 was used to produce the value mapping visualized in Fig. 2.

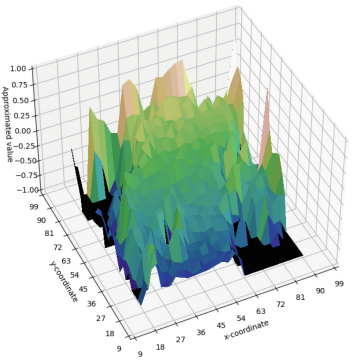

Fig. 2. A visualization of the global critic mapping of the state space

Figure 2 displays the value of each grid cell in the environment as the cumulative sum of value produced by the global critic, i.e. $v_{cell} = \sum_{i=1}^{n} \phi_g(s_{g_n}) \cdot \frac{1}{m}$, where m is the number of agents involved, n is the number of occurrences in MC iterations and $s_{g_n} \in U(states)$ meaning that the states are uniformly sampled from the whole state space. 5000 MC samples were drawn for the visualization in Fig. 2. The resulting value grid was then pooled with a maximum value over 4×4 kernel to make the result easier to interpret: max pooling is an operation that calculates the maximum value for a patch of data, in this case, the 4×4 grid area. Another solution is to use average pooling. Figure 2 shows that there are several locations with higher values, displayed in reddish and white, reaching values $\approx$0.4. The relevance of the results is improved in linear correlation with the amount of samples produced.

Figure 1 solely represents the value mapping, where infeasible locations in the environment map produce zero values and applicable areas differing, codependent values. The idea of the ADMP-Critic is to highlight that, in this case, there are at least 3 locations in the area that produce a higher value than the rest. For

tactical decision-making purposes, the regions with higher values are, according to the aggregated value function, most suitable for force projection to yield a positive outcome in an upcoming confrontation. In other words, the decision maker can use such a mapping to guide the planning process and iterate over the *high value regions* when contemplating the preferred tactical approach. The selection of high-ranking areas can be done with, e.g., top-k sorting to match the use case in the planning process.

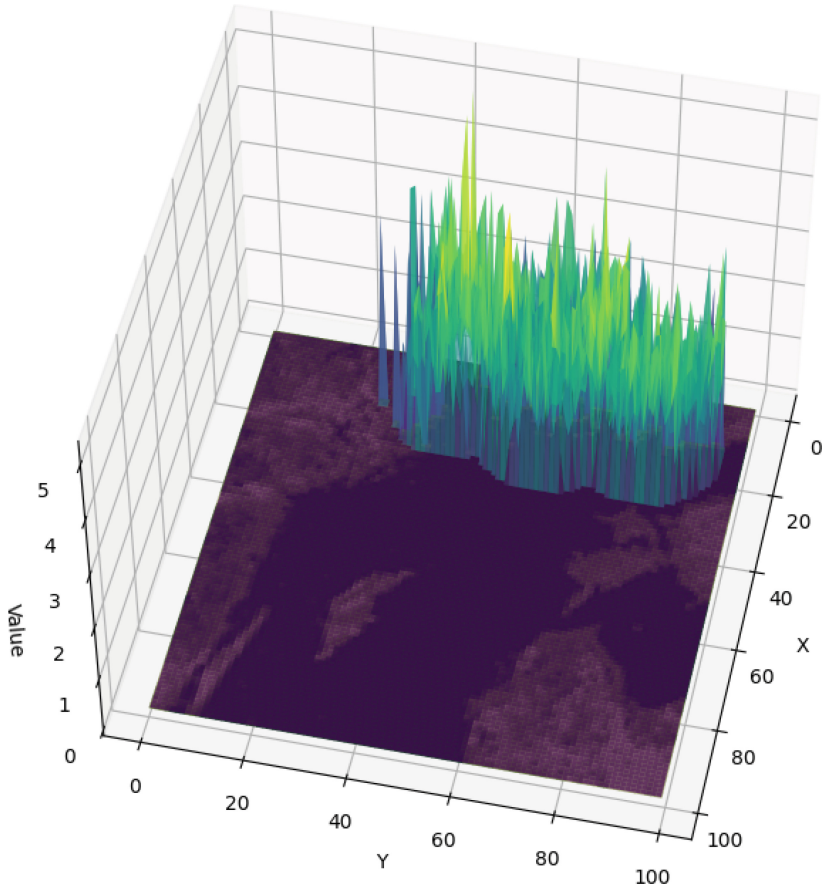

Fig. 3. A visualization of the value map with regard to designated unit areas and map features

Figure 3 displays a similar visualization but now concerning specific unit areas of responsibility and the actual map area. The value mapping is produced with Monte Carlo sampling over a simplified Baltic Sea map. Each of the three units can be positioned within a specified distance from its original position, and

the value of the positioning of all three units is then calculated with the aggregated global value function. In this case, the values result from 100,000 sampling rounds, where the highest values are saved for a particular combination of unit positions. Other state space variables were static, except for radar transmission, which had an equal probability of being on or off. As Fig. 3 shows, while the areas of responsibility overlap, it is favorable to place units to the western side, or further east, and mainly as north as possible. This correlates with the depiction of Fig. 2. In order to be useful, the output can be, e.g., a pooled coordinate area suggestion for each unit instead of the value mapping. Likewise, the decision-maker can guide the value determination by providing additional information for the calculation, such as radar transmission rules, other available assets et cetera. Combining expert insight reduces combinatorial complexity, resulting in more precise and applicable results that reflect the decision-maker's intuition.

Additionally, once the top-k areas have been identified, these can be re-examined combinatorically to determine which combinations of locations are most favourable. The crude mapping considers only the fact that these locations have been a part of some formation that in itself produced a high value. Additionally, max pooling reduces data, as nearby positions may have had considerably lower values.

As a result, the decision-maker can be supported with visualization and areal suggestions for force positioning. ADMP-Critic enables an explainable tool that quantifies the stochastic environment into a format suitable for initial planning without extensive information on the particular upcoming situation.

5 Discussion

This paper proposes a way to aggregate several value functions into a global value function. The global value function is used in creating a static state-value mapping of the problem area to determine the spatial states most likely to result in a high return value.

As RL reward engineering plays a crucial role in the agents' learning process, leading to different policies w.r.t. goal setting and additional rewards, the proposed method can be used to average these differences into a common state-value mapping. The mapping produces insight into the environment and its possibilities as a weighted projection of different reward functions. In other words, the proposed solution allows the detection of spatial states that have a high value under different reward functions or locally optimal policies or exposes the lack thereof.

The solution is essentially a shortcut to avoid the non-stationarity characteristic of MARL [1] and to combine several possible solutions into one value function and one state-value mapping. This approach is chosen instead of increasing the complexity of the model, to enhance explainability. The solution aims to allow optimization with MARL and leverages the agent's exploration of the state space w.r.t. the problem formulation in a computationally feasible manner.

However, the spatial value mapping explainability is still dependent on base model explainability, i.e., it enables an interpretable decision-support mechanism but the model policy behind the value mapping can remain obscure. The explainability depends on the ability of the decision-maker to explain the result w.r.t. the problem.

6 Conclusion

We proposed an extended use of actor-critic algorithms in evaluating the state space of complex reinforcement learning domains to increase the explainability of the found solutions. The proposed algorithm functions as a decision-making support tool by quantifying spatial positioning within the formulated environment. While this paper did not produce baseline results to compare with existing RL and MARL methods, the approximation of a critic value function allows for flexibility in evaluating several, possibly conflicting policies in determining a weighted solution to provide insight when facing uncertainty in decision-making. In the future, the method is supposed to be evaluated in a qualitative manner in a military exercise scenario, where the decision-makers can utilize the solution and evaluate its impact on their planning process.

The use case was driven by military decision-making, as many tactical military decisions are related to troop and unit positioning within the operational theatre against an opponent which is, at best, only partially observable in its goals, force composition and tactics. Despite having roots in military decision-making, the use of aggregated value functions can be applied to other fields where the selection of an optimal policy is hindered due to conflicting, alternate courses of action while facing uncertainty regarding the opposing policy.

References

1. Albrecht, S.V., Christianos, F., Schäfer, L.: Multi-Agent Reinforcement Learning: Foundations and Modern Approaches. MIT Press, Cambridge (2024). https://www.marl-book.com
2. Barredo Arrieta, A., et al.: Explainable artificial intelligence (xai): concepts, taxonomies, opportunities and challenges toward responsible AI. Inf. Fusion **58**, 82–115 (2020). https://doi.org/10.1016/j.inffus.2019.12.012, http://www.sciencedirect.com/science/article/pii/S1566253519308103
3. Bellman, R.: Dynamic Programming. Dover Publications, New York (1957)
4. Dupuy, T.: The Quantified Judgment Method of Analysis of Historical Combat Data: A Monograph. Historical Evaluation and Research Organization (1974). https://books.google.fi/books?id=vzzkGwAACAAJ
5. Eisenführ, F., Weber, M., Langer, T.: Rational Decision Making. Springer Berlin Heidelberg (2010). https://books.google.fi/books?id=nN73nQEACAAJ
6. Harju, M., Liesiö, J., Virtanen, K.: Spatial multi-attribute decision analysis: axiomatic foundations and incomplete preference information. Eur. J. Oper. Res. **275**(1), 167–181 (2019). https://doi.org/10.1016/j.ejor.2018.11.013, https://www.sciencedirect.com/science/article/pii/S037722171830938X

7. Ji, S., et al.: Emerging trends in federated learning: from model fusion to federated X learning. Int. J. Mach. Learn. Cybern. (2024). https://doi.org/10.1007/s13042-024-02119-1
8. Lillicrap, T.P., et al.: Continuous control with deep reinforcement learning. CoRR **abs/1509.02971** (2015). https://api.semanticscholar.org/CorpusID:16326763
9. Mattila, V., Virtanen, K.: Scheduling fighter aircraft maintenance with reinforcement learning. In: Proceedings of the 2011 Winter Simulation Conference (WSC), pp. 2535–2546 (2011). https://api.semanticscholar.org/CorpusID:6848587
10. McMahan, H.B., Moore, E., Ramage, D., Hampson, S., y Arcas, B.A.: Communication-efficient learning of deep networks from decentralized data. In: International Conference on Artificial Intelligence and Statistics (2016). https://api.semanticscholar.org/CorpusID:14955348
11. Metropolis, N., Ulam, S.: The Monte Carlo method. J. Am. Stat. Assoc. **44**, 335 (1949)
12. Osborne, M.J., Rubinstein, A.: A Course in Game Theory. The MIT Press, Cambridge (1994)
13. Qi, P., Chiaro, D., Guzzo, A., Ianni, M., Fortino, G., Piccialli, F.: Model aggregation techniques in federated learning: a comprehensive survey. Futur. Gener. Comput. Syst. **150**, 272–293 (2024). https://doi.org/10.1016/j.future.2023.09.008, https://www.sciencedirect.com/science/article/pii/S0167739X23003333
14. Rempel, M., Cai, J.: A review of approximate dynamic programming applications within military operations research. Oper. Res. Perspect. **8**, 100204 (2021). https://doi.org/10.1016/j.orp.2021.100204, https://www.sciencedirect.com/science/article/pii/S2214716021000221
15. Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.: Proximal policy optimization algorithms. ArXiv **abs/1707.06347** (2017). https://api.semanticscholar.org/CorpusID:28695052
16. Shapley, L.S.: Stochastic games*. Proc. Natl. Acad. Sci. **39**(10), 1095–1100 (1953). https://doi.org/10.1073/pnas.39.10.1095, https://www.pnas.org/doi/abs/10.1073/pnas.39.10.1095
17. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. The MIT Press, second edn., Cambridge (2018). http://incompleteideas.net/book/the-book-2nd.html
18. Sztykgold, A., Coppin, G., Hudry, O.: Dynamic optimization of the strength ratio during a terrestrial conflict. In: 2007 IEEE International Symposium on Approximate Dynamic Programming and Reinforcement Learning, pp. 241–246 (2007). https://doi.org/10.1109/ADPRL.2007.368194
19. Vasankari, L.: Multi-Agent Reinforcement Learning for Littoral Naval Warfare. Master's thesis, Aalto University (2023)
20. Vasankari, L., Saastamoinen, K.: Strategizing the shallows: leveraging multi-agent reinforcement learning for enhanced tactical decision-making in littoral naval warfare. In: Maglogiannis, I., Iliadis, L., Macintyre, J., Avlonitis, M., Papaleonidas, A. (eds.) Artificial Intelligence Applications and Innovations, pp. 129–141. Springer Nature Switzerland, Cham, Switzerland (2024). https://doi.org/10.1007/978-3-031-63215-0_10
21. Yu, C., Velu, A., Vinitsky, E., Wang, Y., Bayen, A.M., Wu, Y.: The surprising effectiveness of ppo in cooperative multi-agent games. In: Neural Information Processing Systems (2021). https://api.semanticscholar.org/CorpusID:232092445