# Factorized, Hierarchical Double-Agent Q-Learning: A Synergistic Approach for Complex Machine Learning Tasks

## 1. Introduction

The realm of machine learning increasingly encounters problems characterized by the presence of multiple interacting agents, demanding sophisticated strategies for effective decision-making. These complex multi-agent systems arise in diverse real-world scenarios, ranging from coordinating autonomous robots in dynamic environments to managing resources in large-scale networks [1]. Traditional single-agent reinforcement learning (RL) methodologies often prove inadequate in addressing the unique challenges posed by such systems. The simultaneous learning of multiple agents leads to non-stationary environments from an individual agent's perspective, and the joint action space grows exponentially with the number of agents, rendering conventional approaches computationally intractable [1]. Consequently, the development of advanced multi-agent reinforcement learning (MARL) techniques is crucial for achieving efficient coordination and learning in these intricate settings [1]. The transition from a single decision-maker to a collective of interacting agents fundamentally alters the learning landscape, necessitating specialized algorithms capable of handling the increased complexity.

To navigate the inherent difficulties of large-scale multi-agent problems, a promising direction lies in the integration of several advanced RL techniques. Factorized Q-learning offers a pathway to manage the expansive action spaces typical in MARL by approximating the global value function through a decomposition into more manageable components [1]. Hierarchical Reinforcement Learning (HRL) provides a framework for tackling the complexity of tasks that require long sequences of actions by breaking them down into sub-tasks operating at different levels of temporal abstraction [4]. Furthermore, Double Q-learning presents a solution to the overestimation bias that is often present in standard Q-learning, a problem that can be amplified in complex learning scenarios [28]. The convergence of these methodologies holds the potential to create a robust and efficient approach for addressing the challenges of scalability, complexity, and estimation accuracy in intricate multi-agent environments.

This paper aims to explore the synergistic potential of factorized, hierarchical double-agent Q-learning. We propose a conceptual framework for the integration of these three powerful techniques and discuss the anticipated benefits and potential challenges associated with such an approach. By examining how these methods can complement each other, we seek to highlight the advantages of a unified strategy for tackling complex machine learning tasks involving multiple interacting agents. Furthermore, we will explore potential applications of this integrated framework across various domains and suggest promising directions for future research in this evolving field.

# 2. Background and Foundational Concepts

## 2.1 Q-Learning: The Foundation

Q-learning stands as a cornerstone in the field of reinforcement learning, providing a model-free approach for an agent to learn an optimal policy by estimating the value of taking specific actions in particular states [44]. The core of Q-learning lies in the iterative update of Q-values, which represent the expected future reward when an agent performs an action in a given state, following an optimal policy thereafter. This update process is governed by the Bellman equation, a fundamental concept in dynamic programming that expresses the value of a state as the immediate reward plus the discounted value of the best next state [45].

Several key concepts underpin the Q-learning algorithm. States define the current situation of the agent within its environment, while actions represent the possible steps the agent can take in each state. Rewards are the feedback signals the agent receives after performing an action, indicating the desirability of that action in that state [44]. The discount factor (γ) determines the importance of future rewards relative to immediate rewards, influencing the agent's long-term planning horizon [45]. The learning rate (α) controls the extent to which newly acquired information overrides existing Q-values [45]. The learned Q-values are typically stored in a Q-table, a data structure where rows represent states and columns represent actions [44]. Initially, the Q-table is often initialized with arbitrary values (e.g., zeros), and these values are updated as the agent interacts with the environment.

A critical aspect of Q-learning is the balance between exploration and exploitation. To discover optimal actions, the agent must explore the environment by trying different actions, even those that may not seem optimal based on current knowledge. Simultaneously, the agent should exploit its learned knowledge by choosing actions that have historically yielded high rewards. A common strategy to manage this trade-off is the ε-greedy policy, where the agent chooses the action with the highest Q-value with a probability of (1-ε) and selects a random action with a probability of ε [44]. While Q-learning provides a powerful foundation for learning optimal policies, its traditional tabular form becomes a significant bottleneck when dealing with environments that have a large number of states and actions. The Q-table's size grows exponentially with the complexity of the state and action spaces, making it impractical for many real-world applications. To overcome this limitation, function approximation techniques, such as neural networks, are often employed to estimate the Q-values, allowing Q-learning to be applied to high-dimensional problems [45].

## 2.2 Multi-Agent Reinforcement Learning (MARL)

Multi-Agent Reinforcement Learning (MARL) extends the principles of RL to scenarios involving multiple agents that interact within a shared environment [1]. In contrast to single-agent settings where the environment's dynamics are often considered stationary, MARL introduces unique challenges arising from the simultaneous learning and adaptation of multiple agents. One of the primary difficulties is the non-stationarity of the environment from the perspective of each individual agent. As other agents learn and update their policies, the optimal strategy for a given agent can change over time, making it harder to achieve stable learning [1]. Furthermore, the joint action space, formed by the combination of actions of all agents, grows exponentially with the

number of agents, leading to a combinatorial explosion that complicates exploration and learning [1]. Effective coordination among agents is also crucial for achieving common goals in many multi-agent tasks, yet designing mechanisms that facilitate such coordination can be challenging.

Various paradigms have emerged within MARL to address these complexities. One prominent approach is centralized training with decentralized execution (CTDE) [7]. In this framework, agents are trained using global information, such as the states and actions of all other agents, to learn effective joint policies. However, during execution, each agent makes decisions based only on its local observations, allowing for scalability and autonomy. This paradigm attempts to leverage the benefits of centralized learning to learn coordinated behaviors while maintaining the practical advantages of decentralized action during deployment. The presence of multiple learning entities fundamentally alters the environment's dynamics for each agent. As every agent adapts its strategy based on its experiences, the landscape of optimal actions for any single agent is in constant flux. This inherent non-stationarity complicates the learning process, as agents must continuously adapt to the evolving behaviors of their counterparts.

## 2.3 Factorized Q-Learning

Factorized Q-learning is a class of MARL techniques designed to tackle the scalability challenges associated with large joint action spaces by decomposing the global Q-function into a more manageable form [1]. Instead of trying to learn a single Q-value for every possible combination of states and joint actions, which becomes computationally infeasible as the number of agents increases, factorized Q-learning aims to approximate the joint Q-function as a combination of individual or pairwise Q-functions [1]. This decomposition can take various forms, such as a simple sum of individual Q-functions or more complex interactions captured through pairwise terms.

One common approach involves using mixing networks to aggregate decentralized Q-functions learned by individual agents [8]. These networks learn to combine the individual value estimates in a way that reflects the overall value of the joint action. By focusing on learning these factorized components and their interactions, the model complexity can be significantly reduced, and the learning process can be accelerated, particularly in systems with a large number of agents [1]. This factorization strategy allows for a more efficient representation of the value function in multi-agent settings, making learning tractable even when the number of agents and the size of the joint action space are substantial. By breaking down the complex, high-dimensional joint Q-function into simpler, factorized elements, the number of parameters that need to be learned is drastically reduced, leading to improved scalability and faster convergence.

## 2.4 Hierarchical Reinforcement Learning (HRL)

Hierarchical Reinforcement Learning (HRL) offers a powerful approach for addressing the complexity of tasks that require extended sequences of decisions by learning policies at multiple levels of temporal abstraction [4]. The fundamental idea behind HRL is to decompose complex tasks into a hierarchy of sub-tasks, where higher-level policies select abstract actions, such as sub-goals or options, and lower-level policies are responsible for executing these abstract

actions through sequences of primitive steps [15]. This hierarchical structure provides several benefits, including improved exploration by allowing the agent to take "big steps" in the state space, faster learning by focusing on simpler sub-problems, and the ability to handle tasks with long time horizons and sparse rewards [6].

Several frameworks exist within HRL, with the Options framework being a particularly popular formalism [13]. An option is defined by three components: an initiation set (the states where the option can be started), a policy (the sequence of actions taken while the option is active), and a termination condition (when the option should end). By learning and utilizing options, an agent can effectively reason and plan at a higher level of abstraction, making it possible to solve tasks that would be intractable for flat RL agents that only consider primitive actions. The introduction of a hierarchical structure to the learning process allows agents to acquire complex behaviors by first mastering simpler, more focused sub-tasks. This decomposition makes the overall learning problem more manageable and efficient, as each level of the hierarchy can concentrate on a specific aspect of the task.

### 2.5 Double Q-Learning

Double Q-learning is a modification of the standard Q-learning algorithm designed to address the issue of overestimation bias [28]. Overestimation bias arises in Q-learning because the algorithm uses the maximum estimated Q-value for the next state as an approximation for the maximum expected Q-value. In noisy environments or when using function approximation, this can lead to a systematic overestimation of action values, potentially resulting in suboptimal policies [29].

Double Q-learning tackles this problem by using two separate Q-value estimators [28]. One estimator is used to determine the action that maximizes the Q-value in the next state, while the second estimator is used to evaluate the value of that chosen action [28]. By decoupling the action selection from the value evaluation, Double Q-learning reduces the tendency to select overestimated values, leading to more stable and accurate learning [28]. This double estimator approach provides a mechanism to mitigate the inherent overoptimistic bias in Q-learning, especially in complex scenarios where estimation errors are more likely to occur. The max operator in the standard Q-learning update rule can inadvertently lead to an inflation of action values, particularly when the estimates are noisy. Double Q-learning addresses this by employing two independent Q-value functions, one to select the maximizing action and the other to provide a less biased estimate of its value.

## 3. Integrating Factorized and Hierarchical Q-Learning in Multi-Agent Systems

### 3.1 Synergistic Potential

The integration of factorized Q-learning and hierarchical reinforcement learning holds significant promise for addressing the intertwined challenges of scalability and complexity in multi-agent systems [3]. By combining these two powerful paradigms, we can create a framework that is better equipped to handle the complexities of environments with many interacting agents and

tasks that require long-term planning and coordination. Factorization techniques can be applied at each level of the hierarchy to manage the action space for the sub-tasks that individual agents or groups of agents are responsible for. This allows for efficient learning even when the number of possible actions at each level is large. Conversely, a hierarchical structure can decompose a complex multi-agent problem into a series of smaller, more manageable multi-agent sub-problems. Within each of these sub-problems, factorized Q-learning can be effectively employed to learn coordinated behaviors among the relevant agents.

This multi-layered approach allows for tackling both the breadth and the depth of the challenges present in complex multi-agent learning scenarios. Factorization addresses the breadth by providing a scalable way to represent the value function in the face of a large number of agents and actions. Hierarchy manages the depth by enabling the agent to learn complex sequences of actions and long-term dependencies through the decomposition of the task into simpler, temporally abstracted steps. The synergy between these two techniques creates a more powerful and versatile framework than either could offer in isolation.

## 3.2 Potential Architectural Designs

Several architectural designs can be envisioned for integrating factorized and hierarchical Q-learning in multi-agent systems. One possibility is a **hierarchical factorized value decomposition** approach. In this design, a high-level policy operates at a more abstract level, selecting sub-goals or options for the agents to pursue. Each of these options could then be associated with a factorized Q-function that captures the value of joint actions taken by the agents involved in achieving that sub-goal. The factorization could occur at the level of individual agents or groups of agents, depending on the nature of the sub-task.

Another potential architecture is **factorized policies over hierarchical actions**. Here, agents learn factorized policies that operate over a hierarchical action space. Higher-level actions in this space could trigger sequences of lower-level actions, which might involve coordination with other agents. The factorization could be applied to the selection of actions at each level of the hierarchy, allowing agents to learn their contributions to the overall hierarchical plan in a distributed manner.

A **two-level hierarchy with factorization** represents a more specific design. In this structure, a meta-controller operates at the top level, selecting high-level goals for the multi-agent system [15]. Below this meta-controller, a lower level of agents utilizes factorized Q-learning to achieve these goals through coordinated actions. The meta-controller might receive feedback based on the success of the lower-level agents in reaching the assigned goals, allowing it to learn an effective strategy for goal selection. Within the lower level, the factorized Q-function would enable the agents to learn how to best coordinate their individual actions to fulfill the high-level directives. The choice of the most suitable architecture will likely depend on the specific characteristics of the problem being addressed, including the degree of required coordination, the natural hierarchical structure of the task, and the number of agents involved.

## 3.3 Algorithmic Considerations

The successful integration of factorized and hierarchical Q-learning in multi-agent systems

necessitates careful consideration of several algorithmic aspects. One crucial aspect is **defining sub-tasks and options in a multi-agent context** [15]. In single-agent HRL, sub-tasks and options often relate to the agent's interaction with the environment. In a multi-agent setting, sub-tasks might involve the coordinated actions of multiple agents to achieve a specific intermediate goal. Automated methods for discovering these meaningful multi-agent sub-tasks would be highly valuable.

Another important consideration is **learning factorized Q-functions for hierarchical actions**. Standard factorization techniques might need to be adapted to account for the temporal abstraction inherent in HRL. For instance, the Q-value of a high-level action (an option) should reflect the expected future reward obtained over the entire duration of that option's execution, considering the coordinated actions of multiple agents at the lower level.

**Credit assignment across hierarchy and agents** presents a significant challenge [17]. When a high-level goal is achieved, it is important to determine how to distribute the credit not only to the high-level policy that selected the goal but also to the individual agents and their coordinated actions at the lower level that contributed to its success. This becomes even more complex when the Q-function is factorized, as the contribution of each agent to the overall value needs to be accurately assessed.

Finally, **coordinating exploration in a hierarchical and factorized space** requires careful design [15]. Agents need to explore both the space of high-level abstract actions (e.g., which sub-goal to pursue) and the space of low-level coordinated actions within each sub-task. Exploration strategies that effectively balance the exploration of the hierarchy with the exploration of the factorized joint action space will be crucial for efficient learning.

# 4. Addressing Overestimation Bias with Double Q-Learning

## 4.1 The Problem of Overestimation in Multi-Agent Hierarchical Factorized Q-Learning

The overestimation bias inherent in standard Q-learning can be further amplified in the context of complex MARL settings that incorporate hierarchical structures and factorized value functions [29]. The increased complexity of the function approximators used to represent the factorized Q-functions, combined with the potential for larger effective action spaces at higher levels of the hierarchy, can lead to greater estimation errors. The max operator in the Q-learning update rule, when applied over these potentially inaccurate estimates, can exacerbate the tendency to overestimate action values. This overestimation can be particularly problematic in multi-agent scenarios where agents' policies are interdependent, potentially leading to unstable learning dynamics and the convergence to suboptimal joint policies [29]. Therefore, it is crucial to address this bias when integrating factorized and hierarchical Q-learning in multi-agent systems.

## 4.2 Incorporating Double Q-Learning Principles

The core principle of Double Q-learning, which involves decoupling action selection and value evaluation, can be effectively applied within a hierarchical and factorized MARL framework to mitigate overestimation bias at multiple levels of decision-making [28]. For each factor in the factorized Q-function, or at each level of the hierarchy, two separate Q-value estimators can be maintained. One set of estimators can be used to determine the best action (or sub-goal) according to the current value estimates, while the other set is used to provide a more accurate estimate of the value of that chosen action, thereby reducing the overoptimistic bias that can arise from using a single estimator [28]. This approach can be implemented in various ways depending on the specific architecture of the integrated framework.

## 4.3 Potential Adaptations for the Integrated Framework

Several potential adaptations of Double Q-learning can be considered for an integrated factorized hierarchical multi-agent Q-learning framework. One possibility is **double factorized Q-learning**, where each agent or group of agents maintains two separate factorized Q-function estimators. During the update process, one set of factors is used to select the maximizing joint action, while the other set is used to evaluate the Q-value of that selected action. This can help to reduce overestimation within the factorized value representation.

Another approach is **hierarchical double Q-learning**, where the Double Q-learning update rule is applied at each level of the hierarchy. This could involve maintaining separate pairs of Q-estimators for each level of abstraction. For instance, the high-level policy might use two Q-functions to select and evaluate abstract actions (sub-goals), while the lower-level policies, which might also be factorized, could similarly employ two Q-functions for selecting and evaluating primitive or sub-task specific actions.

**Hybrid approaches** could also be explored. For example, Double Q-learning might be applied at the lower levels of the hierarchy, where individual agent actions are determined and where the risk of overestimation due to a large number of primitive actions might be higher. At higher levels of the hierarchy, where abstract actions are selected, a single factorized Q-function might suffice, or a different form of bias reduction technique could be employed. The optimal way to incorporate Double Q-learning will likely depend on the specific characteristics of the task, the chosen integration architecture, and the trade-offs between computational cost and the desired level of bias reduction.

# 5. Potential Applications and Use Cases

The integration of factorized, hierarchical double-agent Q-learning has the potential to revolutionize the way we approach complex machine learning tasks involving multiple interacting agents across a wide range of domains.

## 5.1 Robotics and Autonomous Systems

In robotics, coordinating teams of robots to perform complex tasks such as search and rescue operations, managing warehouses efficiently, or enabling fully autonomous driving requires sophisticated control strategies [4]. HRL can be used to break down high-level mission goals into a sequence of simpler, executable robot actions. Factorized Q-learning can then handle the

coordination of actions among multiple robots, allowing them to learn collaborative behaviors in a scalable manner. Furthermore, the incorporation of Double Q-learning can improve the reliability and stability of the learned policies, ensuring safer and more effective operation in real-world environments.

## 5.2 Multi-Agent Games and Simulations

Training artificial intelligence agents to play complex strategy games, such as StarCraft II or capture the flag, demands both long-term strategic planning and the ability to coordinate the actions of multiple units [3]. HRL can enable the learning of high-level game strategies by decomposing the game into a hierarchy of tactical objectives. Factorized Q-learning can manage the joint actions of numerous units, allowing for the development of complex coordinated maneuvers. The use of Double Q-learning can lead to more stable and robust training, preventing the agents from being misled by overoptimistic value estimates and resulting in more competitive and human-like gameplay.

## 5.3 Resource Allocation and Management

Optimizing the allocation of resources in large and complex systems, such as managing traffic flow in urban networks, allocating computational resources in cloud computing environments, or optimizing supply chain logistics, involves coordinated decision-making by multiple entities [2]. HRL can be used to manage different levels of resource allocation, from high-level strategic decisions down to low-level operational adjustments. Factorized Q-learning can handle the interactions and dependencies between different resource allocation units or agents, allowing for efficient and scalable solutions. Double Q-learning can improve the reliability of the decision-making process, preventing suboptimal allocations due to overestimation of potential gains.

## 5.4 Cybersecurity

Developing autonomous defense systems capable of detecting and responding to increasingly sophisticated cyber threats requires coordinated action across multiple security agents or systems [4]. HRL can be employed to decompose the complex task of network defense into a hierarchy of sub-tasks, such as network investigation, threat analysis, and host recovery. Factorized Q-learning can coordinate the actions of multiple defense agents, enabling them to work together to identify and neutralize threats effectively. Incorporating Double Q-learning can lead to more robust and reliable defense strategies, reducing vulnerabilities that might arise from overestimating the effectiveness of certain defensive actions.

## 5.5 Communication Networks

Optimizing routing protocols in dynamic and autonomous communication networks, including traditional networks and emerging technologies like segment routing, presents a complex multi-agent problem [51]. HRL can manage routing decisions at different levels of the network hierarchy, from high-level path selection down to low-level packet forwarding. Factorized Q-learning can handle the interactions between different network nodes or routing agents, allowing for the learning of efficient and adaptive routing strategies. Double Q-learning can

improve the stability and performance of the routing protocols by providing more accurate estimates of the value of different routing decisions.

**Table 1: Potential Applications of Factorized Hierarchical Double-Agent Q-Learning**

| Domain | Use Case | Benefits of the Integrated Approach |
|---|---|---|
| Robotics | Coordinated multi-robot tasks | Scalability for many robots, handling complex task sequences, robust policies. |
| Games & Simulations | Strategic multi-player games | Learning high-level strategies, managing numerous units, stable training. |
| Resource Management | Optimizing resource allocation in large systems | Handling complex allocation hierarchies, coordinating multiple allocation entities, reliable decision-making. |
| Cybersecurity | Autonomous cyber defense | Decomposing defense strategies, coordinating defense agents, reducing vulnerabilities due to overestimation. |
| Communication Networks | Intelligent routing in dynamic networks | Managing hierarchical network levels, coordinating routing nodes, improving network stability. |

# 6. Advantages, Limitations, and Discussion

## 6.1 Expected Advantages

The proposed integration of factorized, hierarchical double-agent Q-learning offers several compelling advantages for tackling complex machine learning tasks. The use of factorization

techniques directly addresses the scalability challenges inherent in multi-agent systems by providing a more compact and manageable representation of the value function, even as the number of agents grows [1]. The incorporation of a hierarchical structure allows the framework to effectively manage the complexity of long-horizon tasks by decomposing them into a series of more manageable sub-problems operating at different levels of temporal abstraction [4]. By integrating Double Q-learning principles, the framework benefits from more stable and accurate value estimates, mitigating the overestimation bias that can lead to suboptimal policies and hindering effective learning [28].

Furthermore, the hierarchical nature of the approach can facilitate both generalization and transfer learning. Learned sub-policies or options at lower levels of the hierarchy, which represent fundamental skills or behaviors, can potentially be reused or fine-tuned for new tasks or in different environments, leading to more efficient learning and adaptation [4]. In the context of multi-agent systems, the combination of factorization and hierarchy can lead to improved coordination among agents. Higher-level policies can guide the overall strategy, while factorized Q-functions at lower levels enable agents to learn coordinated actions to achieve sub-goals effectively [3]. The synergistic effect of these techniques offers a powerful toolkit for addressing the core challenges of learning in intricate multi-agent environments.

## 6.2 Potential Limitations and Challenges

Despite the promising advantages, the integration of factorized, hierarchical double-agent Q-learning also presents several potential limitations and challenges. Designing and implementing an effective framework that seamlessly combines these three advanced techniques can be inherently complex. Determining the optimal way to decompose tasks into a hierarchy of sub-tasks and to factorize the Q-functions for agents at different levels might require significant domain expertise or the development of automated methods for structure discovery [13]. The combination of these computationally intensive techniques could potentially lead to an increased computational overhead compared to using them individually [2].

Establishing theoretical convergence guarantees for such an integrated algorithm is likely to be a complex undertaking and remains an open research question [2]. Furthermore, devising effective exploration strategies that can navigate both the hierarchical action space and the factorized joint action space at each level presents a non-trivial challenge. Balancing the need to explore novel high-level strategies with the exploration of coordinated low-level actions requires careful consideration. While the potential benefits are significant, the successful realization of this integrated framework will require addressing these technical hurdles through rigorous design and analysis.

## 6.3 Discussion

The specific choices of factorization methods and hierarchical structures within the integrated framework will likely need to be tailored to the unique characteristics of the problem domain. For instance, in some applications, a simple two-level hierarchy with pairwise factorizations might be sufficient, while others might require deeper hierarchies or more complex factorization schemes. Empirical validation across a diverse range of challenging multi-agent tasks will be crucial to demonstrate the practical effectiveness of the proposed approach and to identify the conditions

under which it offers significant advantages over existing state-of-the-art MARL algorithms. Further research is essential to delve into the theoretical underpinnings of this integration and to develop robust and efficient algorithms that can overcome the inherent limitations and challenges.

# 7. Future Research Directions

The integration of factorized, hierarchical double-agent Q-learning opens up numerous exciting avenues for future research, spanning both theoretical investigations and practical applications.

## 7.1 Theoretical Analysis

Future work should focus on developing formal proofs of convergence for various integrated architectures. Analyzing the impact of overestimation bias within hierarchical factorized MARL and rigorously demonstrating the effectiveness of Double Q-learning in mitigating this bias is also crucial. Establishing theoretical bounds on the sample complexity and computational efficiency of the integrated approach would provide valuable insights into its scalability and applicability [2].

## 7.2 Empirical Validation

Extensive empirical evaluation of the proposed framework on challenging multi-agent benchmark environments, such as those found in robotics, games, and resource management, is necessary to validate its performance [1]. Comparisons with existing state-of-the-art MARL algorithms will help to quantify the benefits of this integrated approach. Furthermore, investigating the sensitivity of the framework's performance to different choices of factorization methods, hierarchical structures, and Double Q-learning adaptations will be important for guiding its practical application.

## 7.3 Automated Hierarchy and Factorization Discovery

Developing methods for automatically learning effective hierarchical decompositions of tasks in multi-agent settings would significantly enhance the applicability of this framework [4]. Similarly, exploring techniques for automatically identifying optimal factorization structures for multi-agent Q-functions, without relying on manual design or domain-specific knowledge, represents a promising direction for future research [1].

## 7.4 Exploration Strategies for Integrated Frameworks

Designing novel exploration techniques that are specifically tailored to the hierarchical and factorized nature of the action space is crucial for efficient learning. Exploration strategies that can effectively navigate both the abstract action space at higher levels of the hierarchy and the coordinated joint action spaces at lower levels are needed to ensure comprehensive and effective learning [15].

### 7.5 Extensions to Other MARL Paradigms

Investigating the integration of these concepts with other MARL algorithms, such as actor-critic methods, could lead to further advancements in the field [7]. Exploring the application of factorized hierarchical Double Q-learning in partially observable environments, which are common in many real-world scenarios, would also be a valuable extension.

## 8. Conclusion

This paper has explored the potential of integrating factorized Q-learning, hierarchical reinforcement learning, and double-agent Q-learning into a synergistic framework for tackling complex machine learning tasks involving multiple interacting agents. We have discussed the theoretical foundations of each technique and proposed potential architectural designs for their integration. Furthermore, we have highlighted the expected advantages of this combined approach, including improved scalability, enhanced complexity management, and mitigation of overestimation bias, while also acknowledging the potential limitations and challenges that need to be addressed. Finally, we have outlined several promising directions for future research, emphasizing the need for theoretical analysis, empirical validation, and the development of automated methods for hierarchy and factorization discovery. The integration of these advanced RL techniques represents a significant step towards developing more powerful and versatile machine learning solutions for the increasingly complex multi-agent systems encountered in various real-world applications.

#### Works cited


1. Factorized Q-Learning for Large-Scale Multi-Agent Systems - NASA ..., accessed March 16, 2025, https://ui.adsabs.harvard.edu/abs/2018arXiv180903738Z/abstract
2. arxiv.org, accessed March 16, 2025, https://arxiv.org/pdf/1809.03738
3. Factorized Q-Learning for Large-Scale Multi-Agent Systems - ResearchGate, accessed March 16, 2025, https://www.researchgate.net/publication/327592124_Factorized_Q-Learning_for_Large-Scale_Multi-Agent_Systems
4. arxiv.org, accessed March 16, 2025, https://arxiv.org/pdf/2410.17351
5. ojs.aaai.org, accessed March 16, 2025, https://ojs.aaai.org/index.php/AAAI/article/view/26387/26159
6. Hierarchical Multi-Agent Reinforcement Learning - Mohammad Ghavamzadeh, accessed March 16, 2025, https://mohammadghavamzadeh.github.io/PUBLICATIONS/agents01.pdf
7. Hierarchical Consensus-Based Multi-Agent Reinforcement Learning for Multi-Robot Cooperation Tasks - arXiv, accessed March 16, 2025, https://arxiv.org/html/2407.08164v1
8. Inverse Factorized Soft Q-Learning for Cooperative Multi-agent Imitation Learning - NeurIPS, accessed March 16, 2025, https://nips.cc/virtual/2024/poster/93057
9. Inverse Factorized Soft Q-Learning for Cooperative Multi-agent Imitation Learning | OpenReview, accessed March 16, 2025, https://openreview.net/forum?id=xrbgXJomJp&referrer=%5Bthe%20profile%20of%20Tien%20Anh%20Mai%5D(%2Fprofile%3Fid%3D~Tien_Anh_Mai1)
10. Inverse Factorized Soft Q-Learning for Cooperative Multi-agent Imitation Learning - NeurIPS, accessed March 16, 2025,

https://proceedings.neurips.cc/paper_files/paper/2024/hash/2fbeed1dd7162f91804e7b9246e0c1a8-Abstract-Conference.html
11. Asynchronous Factorization for Multi-Agent Reinforcement Learning - OpenReview, accessed March 16, 2025, https://openreview.net/forum?id=5pd46nlxc6
12. Multi-Agent Determinantal Q-Learning, accessed March 16, 2025, http://proceedings.mlr.press/v119/yang20i/yang20i.pdf
13. Hierarchical Reinforcement Learning: A Survey and Open Research ..., accessed March 16, 2025, https://www.mdpi.com/2504-4990/4/1/9
14. Data-Efficient Hierarchical Reinforcement Learning - NeurIPS, accessed March 16, 2025, http://papers.neurips.cc/paper/7591-data-efficient-hierarchical-reinforcement-learning.pdf
15. Hierarchical Reinforcement Learning (HRL) in AI - GeeksforGeeks, accessed March 16, 2025, https://www.geeksforgeeks.org/hierarchical-reinforcement-learning-hrl-in-ai/
16. Hierarchical Reinforcement Learning - Towards Data Science, accessed March 16, 2025, https://towardsdatascience.com/hierarchical-reinforcement-learning-56add31a21ab/
17. The Promise of Hierarchical Reinforcement Learning - The Gradient, accessed March 16, 2025, https://thegradient.pub/the-promise-of-hierarchical-reinforcement-learning/
18. Hierarchical Reinforcement Learning, Sequential Behavior, and the Dorsal Frontostriatal System - PMC, accessed March 16, 2025, https://pmc.ncbi.nlm.nih.gov/articles/PMC9274316/
19. A Neural Signature of Hierarchical Reinforcement Learning - PMC - PubMed Central, accessed March 16, 2025, https://pmc.ncbi.nlm.nih.gov/articles/PMC3145918/
20. On Efficiency in Hierarchical Reinforcement Learning - NeurIPS, accessed March 16, 2025, https://proceedings.neurips.cc/paper/2020/file/4a5cfa9281924139db466a8a19291aff-Paper.pdf
21. Accelerating Task Generalisation with Multi-Level Hierarchical Options - arXiv, accessed March 16, 2025, https://arxiv.org/html/2411.02998v1
22. Hierarchical Multi-Agent Skill Discovery - NeurIPS, accessed March 16, 2025, https://proceedings.neurips.cc/paper_files/paper/2023/file/c276c3303c0723c83a43b95a44a1fcbf-Paper-Conference.pdf
23. Multi-Agent Reinforcement Learning with a Hierarchy of Reward Machines - arXiv, accessed March 16, 2025, https://arxiv.org/html/2403.07005v1
24. Target-Oriented Multi-Agent Coordination with Hierarchical Reinforcement Learning - MDPI, accessed March 16, 2025, https://www.mdpi.com/2076-3417/14/16/7084
25. (PDF) Hybrid MDP based integrated hierarchical Q-learning - ResearchGate, accessed March 16, 2025, https://www.researchgate.net/publication/220362932_Hybrid_MDP_based_integrated_hierarchical_Q-learning
26. Hierarchical correlated Q-learning for multi-layer optimal generation command dispatch, accessed March 16, 2025, https://ui.adsabs.harvard.edu/abs/2016IJEPE..78....1Y/abstract
27. The MAXQ Method for Hierarchical Reinforcement Learning, accessed March 16, 2025, http://matt.colorado.edu/teaching/RL/readings/dietterich%201998%20ICML%20maxQ.pdf
28. Double Q-learning Explained | Papers With Code, accessed March 16, 2025, https://paperswithcode.com/method/double-q-learning
29. Deep Reinforcement Learning with Double Q-Learning - AAAI, accessed March 16, 2025, https://cdn.aaai.org/ojs/10295/10295-13-13823-1-2-20201228.pdf
30. An Introduction to Double Deep Q-Learning - Built In, accessed March 16, 2025, https://builtin.com/artificial-intelligence/double-deep-q-learning
31. Double Deep Q Networks. Tackling maximization bias in Deep… | by Chris Yoon | TDS Archive | Medium, accessed March 16, 2025, https://medium.com/towards-data-science/double-deep-q-networks-905dd8325412

32. Improving the DQN algorithm using Double Q-Learning | Stochastic Expatriate Descent, accessed March 16, 2025, https://davidrpugh.github.io/stochastic-expatriate-descent/pytorch/deep-reinforcement-learning/deep-q-networks/2020/04/11/double-dqn.html
33. CONTROLLING THE ESTIMATION BIAS OF Q-LEARNING - OpenReview, accessed March 16, 2025, https://openreview.net/pdf?id=Bkg0u3Etwr
34. Why does Q-learning overestimate action values? - Cross Validated - Stack Exchange, accessed March 16, 2025, https://stats.stackexchange.com/questions/277442/why-does-q-learning-overestimate-action-values
35. DDQN: Tackling Overestimation Bias in Deep Reinforcement Learning - Medium, accessed March 16, 2025, https://medium.com/@kdk199604/ddqn-tackling-overestimation-bias-in-deep-reinforcement-learning-b1b0d6fa72a4
36. Double Q-Learning and Value overestimation in Q-Learning | by ..., accessed March 16, 2025, https://justin-l.medium.com/double-q-learning-and-value-overestimation-in-q-learning-8d186eb5df9c
37. Fixing overestimation bias in continuous reinforcement learning - Samsung Research, accessed March 16, 2025, https://research.samsung.com/blog/Fixing-overestimation-bias-in-continuous-reinforcement-learning
38. Double Q-learning - NIPS papers, accessed March 16, 2025, https://proceedings.neurips.cc/paper/3964-double-q-learning.pdf
39. (PDF) Deep Reinforcement Learning with Double Q-Learning - ResearchGate, accessed March 16, 2025, https://www.researchgate.net/publication/282182152_Deep_Reinforcement_Learning_with_Double_Q-Learning
40. Deep Double Q-Learning — Why you should use it | by Ameet ..., accessed March 16, 2025, https://medium.com/@ameetsd97/deep-double-q-learning-why-you-should-use-it-bedf660d5295
41. (PDF) Double Q-learning. - ResearchGate, accessed March 16, 2025, https://www.researchgate.net/publication/221619239_Double_Q-learning
42. [2502.02018] Dual Ensembled Multiagent Q-Learning with Hypernet Regularizer - arXiv, accessed March 16, 2025, https://arxiv.org/abs/2502.02018
43. Multi-agent Reinforcement Learning with Deep Networks for Diverse Q-Vectors - arXiv, accessed March 16, 2025, https://arxiv.org/html/2406.07848v1
44. Q-Learning in Reinforcement Learning - GeeksforGeeks, accessed March 16, 2025, https://www.geeksforgeeks.org/q-learning-in-python/
45. Q-learning - Wikipedia, accessed March 16, 2025, https://en.wikipedia.org/wiki/Q-learning
46. A Beginner's Guide to Q Learning - KDnuggets, accessed March 16, 2025, https://www.kdnuggets.com/2022/06/beginner-guide-q-learning.html
47. Understanding Q-Learning in Reinforcement Learning | by Aminu Hamza Nababa (Al'amin), accessed March 16, 2025, https://medium.com/@alaminhnab4/understanding-q-learning-in-reinforcement-learning-3b0e10223ae5
48. An Introduction to Q-Learning: A Tutorial For Beginners - DataCamp, accessed March 16, 2025, https://www.datacamp.com/tutorial/introduction-q-learning-beginner-tutorial
49. Q-Learning Explained: Learn Reinforcement Learning Basics - Simplilearn.com, accessed

March 16, 2025, https://www.simplilearn.com/tutorials/machine-learning-tutorial/what-is-q-learning
50. An introduction to Q-Learning: Reinforcement Learning - FloydHub Blog, accessed March 16, 2025, https://floydhub.ghost.io/an-introduction-to-q-learning-reinforcement-learning/
51. Q Learning in Machine Learning [Explained by Experts] - Applied AI Course, accessed March 16, 2025, https://www.appliedaicourse.com/blog/q-learning-in-machine-learning/
52. NeurIPS Poster Disentangled Unsupervised Skill Discovery for Efficient Hierarchical Reinforcement Learning, accessed March 16, 2025, https://neurips.cc/virtual/2024/poster/94271
53. What is Q-Learning? - Wandb, accessed March 16, 2025, https://wandb.ai/cosmo3769/Q-Learning/reports/What-is-Q-Learning---Vmlldzo1NTI1NzE0
54. arxiv.org, accessed March 16, 2025, https://arxiv.org/abs/1910.04041