Multi-Agent Deep Reinforcement Learning: A Discussion on Methods, Experiments and Limitations

Thisarani Jayaweera
School of Computing
Informatics Institute of Technology
Colombo, Sri Lanka
thisarani.20232802@iit.ac.lk

Venuki Mudalige
School of Computing
Informatics Institute of Technology
Colombo, Sri Lanka
venuki.20232784@iit.ac.lk

Ehansa Gajanayaka
School of Computing
Informatics Institute of Technology
Colombo, Sri Lanka
ehansa.20232972@iit.ac.lk

***Abstract*—Multi-Agent Deep Reinforcement Learning (MADRL) has emerged as a powerful paradigm that integrates the representational capabilities of deep learning with the adaptive decision making of reinforcement learning to solve complex, dynamic, and cooperative multi-agent problems. Recent studies have explored MADRL across diverse domains such as wireless communication networks, energy management, and intelligent transportation systems, demonstrating its ability to achieve near-optimal performance in large-scale, uncertain environments. Centralized training with decentralized execution, communication based coordination, and cooperative exploration have proven to be essential techniques to overcome challenges like non-stationarity, scalability, and sparse rewards. Despite remarkable progress, existing MADRL methods still face limitations related to convergence instability, computational complexity, and generalization to unseen environments. This review provides a comprehensive synthesis of recent methodological advances, experimental outcomes, and benchmarking practices, while identifying critical gaps that continue to hinder the deployment of MADRL in real-world systems.**

Keywords—Multi-Agent Deep Reinforcement Learning (MADRL), Centralized Training and Decentralized Execution (CTDE)

I. Introduction

Multi-Agent Reinforcement Learning (MARL) extends single-agent reinforcement learning by enabling multiple agents to interact, cooperate, or compete within a shared environment to optimize their respective policies. The introduction of deep neural networks into reinforcement learning has led to the rise of Multi-Agent Deep Reinforcement Learning (MADRL), which combines deep representation learning with distributed decision-making, allowing agents to handle high-dimensional states and continuous action spaces. However, the extension from single-agent to multi-agent systems introduces new challenges such as the non-stationarity of the environment—since agents' actions influence each other—and the curse of dimensionality that arises with growing state-action combinations [1, 2]. Figure 1 shows how the Reinforcement Learning improved with the deep learning until it combines with the deep learning. These issues make learning stable and scalable policies significantly more complex than in traditional reinforcement learning settings.

Recent research demonstrates the versatility of MADRL across a wide range of real-world applications. For instance, in wireless networks, multi-agent deep Q-learning and actor-critic algorithms have been successfully applied to distributed power allocation and resource management, showing competitive or superior results compared to centralized optimization methods [3, 4]. Similarly, in rban mobility, MADRL-based traffic control systems, such as the Multi-Agent Recurrent Deep Deterministic Policy Gradient (MARDDPG) and advanced Reinforced AIM (adv.RAIM), have demonstrated improved traffic throughput, reduced congestion, and enhanced coordination among connected autonomous vehicles [5, 6]. Moreover, applications in energy management have leveraged cooperative learning strategies for multi-agent coordination in renewable energy optimization, achieving efficient energy distribution under fluctuating supply-demand conditions [7]. These successes highlight MADRL's capability to tackle complex distributed control problems where traditional optimization or single-agent approaches fail.

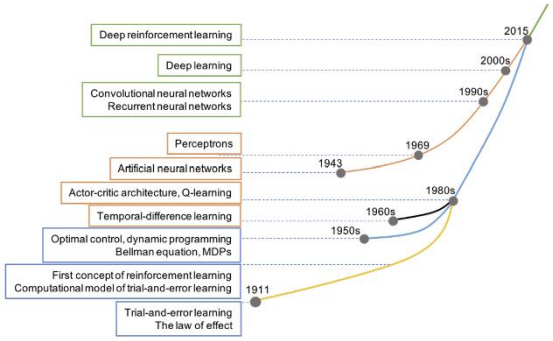

Figure 1: Turning Points of the improvement of Reinforcement Learning are presented, which coverage from the trial-and-error mechanism to deep reinforcement Learning.

Despite these advances, MADRL research continues to face key methodological and experimental challenges. Many approaches struggle with effective exploration in large joint action spaces, limited coordination among agents, and difficulties in generalizing to new or unseen environments [2, 8]. While recent efforts, such as Cooperative Multi-Agent Exploration (CMAE), centralized critics (e.g., MADDPG), and curriculum-based self-play, have improved training stability and scalability, the field still lacks standardized benchmarks and cross-domain evaluations to ensure fair comparison and reproducibility. Additionally, computational costs, communication overhead, and convergence issues remain significant obstacles. Addressing these limitations is critical for advancing MADRL from simulation-based research toward robust, real-world deployment in domains such as smart grids, autonomous driving, and large-scale communication systems.

II. Methodological Analysis

Multi agent deep reinforcement learning (MADRL) architecture emphasizes manipulation over discovery. Algorithms optimize for steadying unreliable dynamics, executing centralized governance, topological discounting, and formalized communication. Optimization concentrates on delimiting chaos instead of facilitating fundamental cooperation, transitioning the prioritizing from regarding agents can collaborate to regarding collaboration can sustain beneath imposed procedure and engineered regulation.

A. Fragile Equilibrium: Centralized Training and Decentralized Execution (CTDE)

The methodological core of the most analyzed strategies to deep reinforcement learning for multi-agent systems (MADRL) consisted

of the Centralized Training and Decentralized Execution (CTDE) framework. This composition is expected to harmonize the tension between global cooperation and local automation, preparing shared global learning while initiating limited local details. However, as examined across analysis including [18, 3, 12], CTDE continues to be a less combined solution and more a fragile equilibrium, constantly redefined to resist the uncertainty it itself implements.

$$L(\phi) = Es, a, r, s'[(Q\phi(s, a) - y)^2], \quad (1)$$

$$y = r + \gamma Q_{\phi^-}(s', a') \quad (2)$$

At this essence, CTDE methods distributes a frequent critic ($Q_\phi(s, a)$), trained applying the Bellman objective above (1), (2). Every local agent (i) studies an actor policy ($\pi_{\theta_i}(a_i|o_i)$) applying gradients obtained from this centralized critic. The edge of this arrangement lies in the critic's complete approach: it steadies value approximation by mitigating incomplete state throughout training. Yet, this equivalent global standpoint covers a core defect, data snooping. The supposition that each state-action pair is reachable in the course of centralized learning is infrequently reasonable beyond simulations.

In [18]'s power grid coordination, CTDE enhances temporary stability by centralized critic upgrades. However, when launched, agents mislay permission to global frequency and volage conditions, imposing them to forecast from partial facts. Therefore, the discovered policies hazard overfitting centralized correlations that disappear in the deployment of it. Similarly, [3] announce refined throughput in united power dispensing, yet their model rerun buffer gathers data beneath a non-stationary connection policy, weaking the transient steadiness CTDE is meant to defend.

These inconsistencies reveal a methodological contradiction: CTDE presumes constant global supervision all through learning, but decentralized processing assures its absence. Therefore, it operates not as an ultimate answer but as a balancing scaffold, reliant on the very terms it thereafter invalidates. Accordingly, CTDE must be constantly re-engineered, via recompense decomposition, attention-weighted critics, or communication regularization to rebuild the equilibrium it shatters.

Hence, concurrently CTDE remains the prevailing paradigm; its methodological correctness is tentative. It diminishes unsteadiness restrained centralization but misfires to generalize under surveillance decentralization. The domain's determined return to CTDE based models mirrors not consensus, but a current methodological approach on fractional centralization.

B. Revolving Non-Stationarity: Encapsulation, Instability

The uncertainty in MADRL appears from non- stationarity, as every agent's progressing policy amends environment's evaluation dynamic. This interrelation fractures the Markov property and negates convergence suppositions of classical RL. Methodologically, majority of reviewed papers; like [15, 5, 19], do not terminate this instability; they simply include it through estimations that localize its consequences.

$$y_i = r_i + \gamma \sum_j \alpha^{d_{ij}} Q_{\phi_i}(s', a') \quad (3)$$

In [15] reduces non-stationarity in allocated traffic control through spatial discounting. Here the Q-value upgrade for every agent is regulated by decomposing factor ($\alpha^{d_{ij}}$), where (d_{ij}) indicates the positional distance between agents (3). This weighted aggregation moderates the impact of remote agents, constructing learning feasible. However, this shortening trades away supremacy; by ranking locality, it impliedly presumes distant relations are insignificant; an assumption that fall apart in forcefully coupled fields like UAV swarms and energy grids.

$$h_t = f_{LSTM}(h_{t-1}, o_t, a_{t-1}) \quad (4)$$

In [5] integrate temporal memory utilizing recurrent networks, converting the learning procedure into a Recurrent Actor-Critic approach. The hidden state in (4).

Translates historical circumstances to estimated belief declares in partially visible environments. This generates temporal filtering but at the cost of computational steadiness; recurrent critics increase gradient noise and essential curriculum-based training agendas for approximation.

In [19]'s PowerNet embraces a third restraint strategy; dual timescale updates, where critics master on more slowly in temporal horizons than actors. This disconnects the fast-changing combined policy from the critic's learning sign, refining short-term firmness but initiating lag that misrepresents long-horizon responsibility.

These procedures distribute a methodological pattern: no steady MADRL dynamics by handling policy coupling: rather, they confine the score, memory or rate through what joint manifests. In other words, non-stationarity is not resolve but strained. Accordingly, the steadiness achieved in such techniques is not an element of the algorithm's confluence proof but a vestige of exclusive constraint. Therefore, existing MADRL architecture produces restricted steadiness, not accurate equilibrium; a critical conceptual difference often indistinct in performance analyzing.

C. Actor-Critic Alternatives: Synchronization by Regularization, Not Basics

The actor-critic clan dominates the methodological terrain of MADRL, significantly in constant action blanks where gradient-based upgrades are crucial. However, the escalation of stochastic, entropy regularized and deterministic versions through reviewed research studies e.g., [18, 19, 12, 11] uncovers a profound issue: these systems often unite not by theoretical validity but by ad-hoc regularization of unsteadiness.

Below is the agent's policy gradient in this clan:

$$\nabla_{\theta_i} J(\theta_i) = Es, a \sim \pi[\nabla\theta_i \log \pi_{\theta_i}(a_i|o_i) A_i(s, a)] \quad (5)$$

In above equation (5) ($A_i(s, a)$) is the approximated benefit under a global critic. However, since the global critic itself relies on all agents' movements, this anticipation becomes ambiguous under concurrent upgrades. To counter this, [18] adjust the critic's loss with clear restriction penalties:

$$L(\phi) = E\big[\big(r + \gamma Q_{\phi^-}(s', a') - Q_\phi(s, a)\big)^2 + \lambda C(a)\big] \quad (6)$$

In above (6) ($C(a)$) represents power grid practicability violations. This joining between policy gradients and physical requirements enforces steadiness, but only beneath the presumption of know dynamics: a supposition that breaks underneath stochastic disturbances.

In [19]'s *PowerNet* implements a benefit Actor-Critic (A2C) with derivable communications, where inter-agent signals ($m_{ij} = f_\psi(o_i, o_j)$) are implanted in the critic's entry. This guarantees

effortless gradient dissemination across agents but directs to credit ambiguity- display gains cannot be secluded to distinct contributions.

$$J(\pi) = E[Q(s_t, a_t) - \alpha \log \pi(a_t|s_t)] \quad (7)$$

Further, [12] and [11] combined entropy regularization, boosting (17). While entropy motivates exploration, it conflicts with MADRL's cooperative necessity for coordinated causality. The agents' stochastic guidelines initiate coupling changeability that can destroy stabilization.

Thus, while actor-critic structures display empirical achievement, their stabilization derives from architectural stabilizers (entropy damping, penalties, communication regularizes) rather than inherent theoretical ensures. Therefore, what emerges as algorithmic progress is often methodological reward – regularization replacing for stabilization proof.

D. Attention Techniques: Collaborative by Selective uncertainty

The establishment of attention in MADRL symbolizes an attempt to scope synchronization through exclusive communication. Rather than each agent simulating all other, attention permits contextual scheduling; agents concentrate on those peers whose declares most influence their own benefits. Figure 2 illustrates how each agent communicates with each agent. However, as demonstrated in [12] and [11], this strategy includes a methodological inconsistency: while it decreases communication intricacy, it restores centralized contingencies while training.

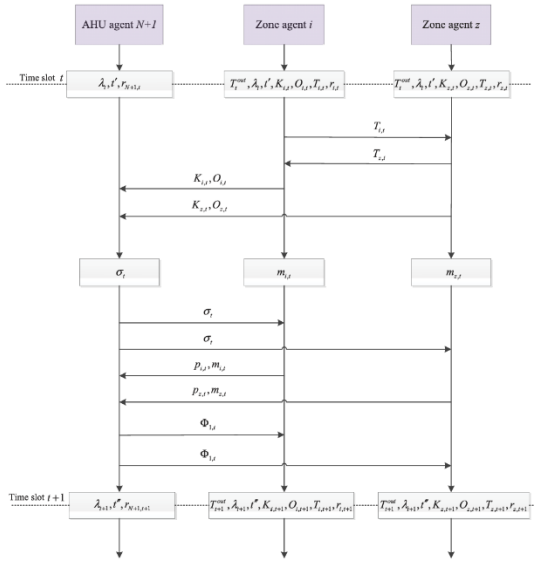

Figure 2: This shows how each agent communicates and exchange information with various agents with given time slots. Each agent collects and stores information and experiences.

$$Q_i(o, a) = f_{\psi_i}\left(o_i, a_i, \sum_{j \neq i} \omega_{ij} g_{\psi_j}(o_j, a_j)\right) \quad (8)$$

In attention-founded on critics, every agent (i) projection value as in (8):

$$\omega_{ij} = \frac{\exp(q_i^T k_j)}{\sum_k \exp(q_i^T k_k)} \quad (9)$$

With weights calculated as (9):

These erudite coefficients permit agents to deduce relational significance dynamically. However, because the calculation (9) of (ω_{ij}) needs permission to all agents' embeddings (k_j) while training, this apparatus is centralized in disguise. While processing, agents' absence of those embeddings, impelling them to depends on predicted or affixed attention weights, which speedily becomes outdated in non-stationary configurations. Moreover, attention mechanisms, while algorithmically elegant, present interpretability delusions. Elevated attention weights are often flawed for informal influence, although they only reflect figurative similarity training beneath the critic's hidden space. [11]'s UAV communication procedure, for instance, displays steady training underneath the attention critics, yet when latest UAVs introduce to the system, formerly learned attention arrangements misallocate precedence, degrading collaboration.

Hence, attention's expandability claim is contingent; it scopes not by decreasing true dependency but via graded moderating it. The outcome is a fragile equilibrium: collaboration emerges by discerning awareness, but this awareness is trained underneath global supervision that processing cannot duplicate. Therefore, attention provides a methodological concession; it restricts the extent of dependence without settling the fundamental joining between coordination and perception.

E. Graph-Structured Collaboration: From Architected Priors to Architected Bias

Graph-based procedures display an evaluation of attention, substituting soft relational conclusion with clear topological modelling. In [20]'s *DeepMAG,* the multi agent surrounding is converted as a graph $G = (V, E)$, where nodes match to edges and agents to their functional dependencies.

The global critic's Q-function is consequently calculated via graph convolution:

$$Q(s, a) = GNN(h_i(o_i, a_i)_{i \in V}, E) \quad (10)$$

According to the above equation (10) where ($h_i(o_i, a_i)$) are node embeddings. This procedure provides efficiency and interpretability, permitting agents to cause only regarding their graph neighbors. However, as DeepMAG hand operated defines edge categories (machine-to-job, job-to-machine), its universality is severely restricted. The studied policy cannot adjust if the topology modifies as an example when relations evolve or new nodes are added. From a methodological perspective, this shift to the direction of graph structured logic reflects cognitive retreat from model free learning to model imposed learning procedure. By converting interaction assumptions directly into the learning procedure, researchers guarantee sacrifices and steadiness adaptability. Moreover, since graphs convert to constant dependencies, they defeat in domains where connections are emergent or dynamic, like adaptive sensor networks or UAV swarms.

Therefore, graph structed learning gives robustness via rigidity. It steadies learning not because it universalizes but because it restricts, imposing procedural bias as a replacement for relational interface. While doing so, it emphasizes the boarder methodological theme via MADRL, alignment is accomplished not through expanded learning capability, but through cautiously engineered structural limitations.

F. Procedural Reinforcement Learning: Decomposition or Movement of Difficulty

The escalating complexity of multi-agent surroundings has driven hierarchical MADRL (HMARL) methods, where process sense is decayed into multi-level concepts. In [17] implemented this pattern in renewable aware cloud scheduling, unveiling local (worker) and global (manager) agents that function on discrete temporal scopes.

The inclusive objective can be stated below:

$$J = E\left[\sum_t \gamma^t \left(r_g(s_g, a_g) + \beta r_l(s_l, a_l)\right)\right] \quad (11)$$

In the above (11) where (β) steadiness local and global benefit priorities. The procedure promises expandability by restricting policy optimization inside diminished subspaces. However, the belief that these subspaces are isolated methodologically imperfect; local agents adjust to global policies, of which development, consequently, depends on combined local results. The outcome, often perceived in [17], is gradient interference: when advanced policies progress faster than lesser ones, the gradient supervision at the local level turns outdated before integration, causing vibrating learning. To reduce this, tiered systems or layered systems employ interchanging upgrades or synchronized timeframes, which ironically reintroduce centralized cooperation, the exact issue hierarchy desired to avoid.

Consequently, while modular decomposition emerges clarify control, it just displaces intricacy from the policy space to the arrangement procedure. The system transforms into scalable in state depiction but fragile in temporal uniformity. Hence, hierarchy in MADRL is not methodological compression but a rearrangement of unsteadiness across levels, arrangement is not eliminated, only delayed to synchronization patterns.

G. Memory- Augmented and Recurrent Cooperation: Temporal Understanding or Temporal Mirage

Although centralized critics and hierarchical concepts endeavor to tame dimensional complication, recurrent models target a more delicate challenge, temporal incomplete noticeability. Multi agent surroundings frequently disobey the Markov assumption, since every agent detects only particles of global condition through time. To counterbalance, several research works, particularly [5, 6], implant recurrent encoders like GRUs or LSTMs to restore concealed temporal situations.

Every agent sustains a hidden phase:

$$h_t = \mathrm{LSTM}(x_t, h_{t-1}) \quad (12)$$

$$x_t = [o_t, a_{t-1}] \quad (13)$$

According to the equation (12), (13) indirectly captures unnoticed variables through bygone summarization. This structure steadies the policy upgrades beneath delayed or unfinished observations. However, the presumption that (h_t) offers a satisfactory statistic of the past is hypothetically delicate: once compacted into a constant-length vector, enduring dependencies disappear exponentially.

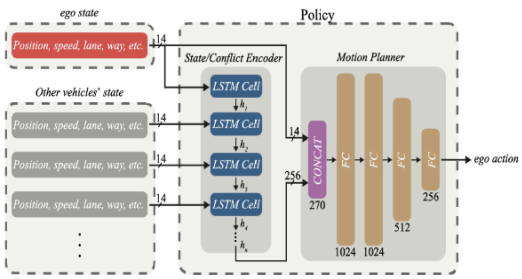

Figure 3: In the recent high-level RAIM (adv.RAIM) network, the policy evaluates what movement the ego state vehicle required to perform. The outcome is the scaled speed the ego sate vehicle required to prefer in the upcoming step. There is individual LSTM cell, which operates the variables of each vehicle separately, beginning with the ego state vehicle's state, after that remaining vehicles' states. The state/conflict encoder generates an outcome (hx) with 256 hidden components.

In adv.RAIM [6] LSTM founded on agents synchronized at unsignalized convergences by predicting competitors' trajectories. Figure 3 illustrates how the adv.RAIM is used with the LSTM. Yet, their latent state concept performs well only beneath alike traffic compactness to those observed while training. When compact distributions transfer, memory impregnation guides to temporal aliasing, at the spot where discrete dynamic circumstances map to equivalent (h_t). Therefore, as recurrence refines robustness, it also produces a false of temporal comprehension, a methodological delusion of simplification obtained from memorization.

In addition, recurrent critics bring upon severe excerpt ineffectiveness. Gradients have to propagate over the course of time, exploding computational cost and magnifying variance. To counterbalance, [5] adjust curriculum learning, beginning from low-traffic reproductions and progressively expanding intricacy. This steadies convergence, nevertheless it reveals the intense methodological fragility: not architecture, training structures, enduring the burden of steadiness. Hence, recurrent cooperation emerges as a retrospective adjustment rather than a principled system for non-Markovian multi agent logic.

In summary, memory expansion is MADRL displays a temporal confinement approach equivalent to special disregarding, beneficial but partial. It converted the previous without simulating insecurity about it. As a result, recurrent agents do not genuinely reason with the time, they recall it. Differentiation is essential: memory integration reduces temporal uncertainty but cannot exchange formal cognitive state approximation.

H. Reward Modeling and Credit Assignment: Collaboration by Architecture, Not Insight

A determined methodological predicament in MADRL is the credit-assignment difficulty, deciding which agent's movements are assisted to combined reward. Approximately, every reviewed procedure evades via reward shaping, enforcing collaboration by way of engineered encouragement rather than appearing communication,

$$Q_{tot}(s, a) = \sum_i Q_i(o_i, a_i) \quad (14)$$

In collaboration assignments, the global Q-function is often decomposed as (14). The addictive supposition occupied by [7]'s *Value Decomposition Networks (VDN)*. This reduction allows decentralized training but disregards nonlinear inter-agent subordinations. When communications are cooperative or oppositional, simple total misrepresents accurate participation. For that reason, policy gradients refine the incorrect utility framework, firmness increases although perfection declines.

In [11] recommended attention-weighted credit approximation:

$$Q_{tot}(s, a) = \sum_i \sum_j \omega_{ij} Q_{ij}(o_i, o_j, a_i, a_j) \quad (15)$$

In (15), $((\omega_{ij})$ reveals connected importance. Although this unwinds the linearity presumption, it substitutes specific decomposition with data driven bias, mastered attention weights overtrain to dimensional regularities of the training configuration. When fresh agents join, misassignment occurs, demeaning justice and collaboration.

In the same way, [20] impose cooperation in DeepMAG throughout a uniform negative reward ($r_t = -1$) up until assignment completion, convincing agents to reduce global total execution time. This joint reward procedure ensures collaboration but removes individual distinction, an occurrence of cooperation by coercion. Agents learn to comply instead of being collaborative.

Accordingly, current MADRL achievement often originates from outside enforced alignment alternatively of inside learned cooperation. Credit assignment methods transform societal learning towards supervised optimization, encoding autonomy into compliance. The domain's methodological complication, therefore, is not to engineer more expand reward decompositions, but to generate differentiable causal estimators that feature outcomes without principal labels. Until then, multi agent collaboration will continue structurally simulated instead of behaviorally evolving.

I. Sample Productivity and Exploration: Organized Explore or Distributed Stagnation

Exploration, insignificant in single-agent RL; turns into abnormal in multi-agent circumstances because each agent's investigation alters others' occurrences. Arbitrary movements increase environmental uncertainty, undermining joint policy gradients. Established works, significantly [8], reinvent analysis as a coordinated collective process and not an individual venture.

Their cooperative multi agent exploration (CMAE) method outlines joint uncertainty over configuration space divisions:

$$H = -!\sum_{s\in\mathcal{S}} p(s)\log p\,(s) \tag{16}$$

And rewards agents in favor of collectively growing coverage of low-uncertainty regions. This (16) converts exploration to joint optimization objective. However, the procedure presupposes that configuration partitions ($(\mathcal{S}_{\#})$) are discoverable or recognized, which is infrequently realistic in constant domains. Moreover, agents' coordinated exploration restrains behavioral based diversity, a phenomenon [8] accept as 'exploratory homogenization.'

Other research works preserve independent exploration but restrict its variability through entropy regularization, as in [8]. Nevertheless, this generates contradictory gradients; high entropy policies may deviate in collaborative tasks, while low entropy ones are untimely intersected.

Therefore, MADRL exploration methods fluctuate between conformity and chaos. CMAE diminishes variance but threads slowdown in subspace, entropy driven procedures maintain diversity but unsteady collaboration. The methodological gridlock reveals an unexplained paradox, exploration and synchronization are conflicting objectives. Whichever method/algorithm optimizing one intrinsically compromises the other. Upcoming structures must therefore manage exploration not like a noise injection but as structured hypothesis testing inside the shared policy diverse.

J. Restraint Embedding and Physical Possibility: Protection at the Adaptation Overhead

In material domains like power grids, HVAC control and UAV directing, agents function beneath sacrosanct safety oy functional constraints. Including these into RL introduces however additional layer, practically preservation. from [18]'s integrates generator restrictions directly to the policy gradient upgrade by sanctioning movements within the critic loss:

$$L(\phi) = E!\left[\left(r + \gamma Q_{\phi^-}(s', a') - Q_{\phi}(s, a)\right)^2 + \lambda C(a)\right] \tag{17}$$

This (17) assures steadiness and compliance but biases learning heading for conservative actions. [7] standardize this far from Feasible Action Screening (FAS), changing the standard DQN update as:

$$Q(s, a)! \leftarrow !\,Q(s, a) + \eta\delta, I[a! \in !\,\mathcal{A}_{\text{fail}}] \tag{18}$$

From the above (18), ((δ)) is the time base- difference error. FAS implements safety predictably but deactivates exploration of boundary restrictions where ideal policies frequently reside.

Accordingly, constraint integrating replaces unpredictability with compliance. Whereas necessary for real-world execution, it transforms RL from an exploratory framework toward a rule-based optimizer. The tradeoff is clear contrast, as protection grows, flexibility reduces.

This tension reflects the wider methodological dualism of MADRL, steadiness accomplished through limitations. Agents stay bounded not just by environmental physics except by algorithmic over grading. Consequently, constraint aware MADRL systems unite protectively, but rarely optimally, mirroring a domain still emphasizing control over frequency.

K. Parameter Distributing and Expandability: Effectiveness through Standardization

To manage massive population of agents, multiple studies, including [5] and [11] accept parameter sharing, the place where often weights and identical network procedures are reclaimed through agents. This hierarchy dramatically diminishes stabilize gradients and sample complexity, hypothesizing functional consistency among agents.

Methodologically, if each policy is conveyed as:

$$\pi_{\theta_i}(a_i|o_i) = \pi_{\theta_{\text{shared}}+\Delta_i}(a_i|o_i) \tag{19}$$

with tiny tasks-define deviations in (19) (Δ_i), optimization clarifies to changing (θ_{shared}). However, this shared portrayal blurs conduct based concentration. When surroundings contain varied dynamics, as an example UAVs with varying payload capabilities, collective parameters contribute to policy aliasing, where various contexts correspond to indistinguishable moves.

Parameter sharing accordingly presents a scalability, variant trade-off. It levels efficiency beneath homogeneity but crumbles when agent roles deviate. Moreover, mutual networks inspire coordinated exploration, accumulating the non-stationarity issue they were meant to soften.

Therefore, even though parameter sharing is visible as a pragmatic effectiveness measure, methodologically it stands for a recede from individuality, enhancing computation instead of adaptability. The recurring embracing of this mechanism via fields underscores the domain's bias approaching engineering viability over behavioral abundance.

L. Synthesis of Methodological Insights: Constraint as the Core Design Basis

Across all the methods analyzed, firmness in MADRL appears not from self-governance but from restriction. Whether through knowledge regularization in CTDE [18, 3], feasibility screening in safety critical control [7], or correspondence controlling in attention critics [12, 11], each procedure embeds specific limits on contact, action or awareness. These techniques are not secondary. They are architectural steady that substitute the infinite adjustability of agents with bounded reliability.

Analytically, most MADRL deliverables can be redefined as constrained policy optimization:

$$min_{\theta_i} E_{\pi}\left[\sum_i \left(\ell(Q_i, \pi_i) + \lambda^1 R_{comm} + \lambda^2 R_{entropy} + \lambda^3 C_{feasibility}\right)\right] \quad (20)$$

In (20) the $\mathcal{R}_{\text{comm}}$ sanctions extreme inter-agent messaging [12], $\mathcal{R}_{\text{entropy}}$ reduces stochastic variation in collaborative settings [11], and $\mathcal{C}_{\text{feasibility}}$ assures physical requirement satisfaction [18, 7].

Every regularizer minimizes the solution space by not including inconsistency, preferring resolving it.

Graph based literary critics like DeepMAG [20] steady by stabilizing relational network structure E, structured schedulers [17] contain tuning to hierarchical sub policies, recurrent encoders [5, 6] minimize temporal variability into constrained hidden configurations. Cumulatively, these express a combined methodological production rule:

The empirical achievements documented, optimized convergence, speedy collaboration, safer exploration, originate precisely because the studying and learning space has been systematic. MADRL's improvements, therefore, are convergent, each fresh architecture narrows self-management to protect an order.

The contention that arises is provocative but predetermined, multi agent intelligence present is not self-organizing collaboration, but engineered standardization.

III. Experimental Results

Research on different domains, mainly including power systems, energy systems, traffic control systems, wireless communications and other related areas, that have applied MADRL has shown significant improvements in achieving better performance and higher quality of service rather than using traditional methods. The experiments and their results were gathered from several studies to assess the performance of using MADRL algorithms to achieve the desired goal.

In a study conducted in 2019, the authors have explored how to use multiple agents as transmitters (learning from their neighbors) in a wireless network using deep reinforcement learning to dynamically adjust the power levels to maximize the throughput of data (sum-rate) across the network while handling signal condition changes and delays. The average sum-rate performance per link in bps/Hz is measured to check the efficiency of data transmission in each link by varying several parameters (for example, the distance between transmitters). The results show that the DQN (Deep Q-learning) algorithm achieves the highest sum rates when it is trained with matched conditions compared to the WMMSE (Weighted Minimum Mean Square Error) and FP (Fractional Programming) algorithms and central, random and full-power allocations. However, the performance of the DQN drops slightly when it is tested with unmatched conditions but still managed to outperform the benchmarks [3]. The work in [4] focuses on managing resources and reducing interference in wireless networks by using a deep RL agent in each transmitter (Access Point, AP). During training, DQN and Advantage Actor-Critic (A2C) favored the sum-rate performance (average user rate), but they learned to balance the performance with that of the weaker users (5th user percentile rate). The results show that, while the A2C achieved a better sum-rate, DQN has had faster convergence due to its experience buffer reaching a higher Rscore after only 12 epochs of training outperforming A2C and centralized Information-Theoretic Link Scheduling (ITLinQ) algorithms. In the final tests with similar train and test configurations, DQN shows excellent performance in the 5th user percentile rate (0.35 bps/Hz with 40 UEs), similar to ITLinQ, while A2C surpassed ITLinQ by achieving a higher sum-rate (21 bps/Hz with 8 APs), where both RL methods outperformed the Full Reuse (0.12-0.14 bps/Hz 5th percentile) and Time Division Multiplexing (TDM) (0.10-0.12 bps/Hz) due to stronger interference handling. Although DQN showed less performance with increasing APs, A2C showed better performance proving more stability and adaptability. Further tests with discrepant configurations highlighted less than 3% performance variation which confirmed the robustness of both the models. In addition to the model-free algorithms such as DQN and A2C, Model-Based Reinforcement Learning (MBRL) show considerable advantages with the ability to plan ahead, making them more sample efficient and more robust to changes in the environment with the ability to learn and have a model of the environment being the reason [14]. However, modelling a complex wireless network environment accurately seems challenging, therefore, model-free algorithms are preferred due to that reason.

MADRL techniques have also been used to control the traffic lights in big cities where there is an RL agent available in every traffic light to reduce the complexity of using a single central system to make decisions. The study [15] introduces, multi-Agent A2C (MA2C) comparing it to Independent A2C (IA2C), Independent Q-Learning with Linear Regression (IQL-LR), Independent Q-Learning with Deep Neural Networks (IQL-DNN), and a Greedy policy by testing on a large fake traffic grid and a real-world network of Monaco city. Average reward ($\bar{R}$), average queue length (veh), average intersection delay (s/veh), average vehicle speed (m/s), trip completion flow (veh/s), and trip delay (s) were used as key metrics. The results of the synthetic traffic grid show MA2C achieving the highest $\bar{R}$ (-414) outperforming IA2C (-845), Greedy (-972), and IQL-LR (-1409) excluding IQL-DNN as its policy was meaningless. MA2C showed shorter queue lengths and delays at peak congestion times compared to the other methods. Once again, MA2C showed excellent performance when tested with the Monaco city grid achieving scores of -78.7 $\bar{R}$, 0.75 veh, 104 s/veh, 14.26 m/s, 2.40 veh/s and 1701s trip delay outperforming Greedy (1.08 veh, 272 s/veh), even though it achieved a higher vehicle speed (14.96 m/s), and IA2C (-117.9 $\bar{R}$, 1.16 veh). However, the IQL variants have shown poor performance with $\bar{R}$ values less than -200 and trip delays greater than 2600 s. Another MARL framework that has included methods such as parameter sharing, local rewards, action masking, and a priority-based safety supervisor was used to teach autonomous vehicles (AVs) to safely merge onto highways with human-driven vehicles (HDVs) using curriculum learning aiming to reduce collisions [16]. Easy, medium and hard traffic modes were used to perform experiments compared to MAA2C, Multi-Agent Proximal Policy Optimization (MAPPO), Multi-Agent Actor-Critic using Kronecker-Factored Trust Region (MAACKTR) and Model Predictive Control (MPC). The proposed method shows very low average collision rates, mostly zero and maintained the average speed above 20m/s across all three modes compared to other algorithms. While MAPPO and MPC maintained similar speeds, they reached higher collision rates (0.34,0.40) across medium and hard traffic modes. In the case of multiple-through lane, the proposed approach achieved best performance with zero collision rates through all modes with the other algorithms also showing reduced collision rates compared to the single lane case. However, all the algorithms showed lower average speeds due to more AVs and HDVs in the multiple-through lane case. These results proved that using the safety supervisor and curriculum learning improved the performance making it more scalable and reliable. Further research shows that adv.RAIM, a proposed MADRL-based system with self-play curriculum learning, to allow CAVs to control signal-free intersections without traffic lights, was tested using a 4-way, 3-lane with different levels of traffic where adv.RAIM showed better performance reducing time loss by 95%, travel time by 59% while minimizing emissions and fuel consumption compared to Fixed-Time, improved Random Early Detection for Vehicles Dynamic (iREDVD) and other AIM algorithms. Furthermore, adv.RAIM

adapted well to varying conditions such as accidents proving to be fast and adaptive [6].

To control the HVAC systems in buildings to reduce the Total Energy Cost (TEC), a model-free multi-agent actor critic RL method-based control algorithm was proposed by [12] which outperformed Rule-Based and Heuristic Schemes (RS & HS) by achieving the lowest TEC proving the flexibility in coordinating air supply rates reducing and increasing them according to the electricity prices which showed a 56.5-72.25% energy cost reduction compared to RS and HS while maintaining appropriate temperature and CO2 levels. A recent study [17] proposed a local neighborhood based multi-agent actor-critic algorithm to optimize the use of renewable energy and workflow scheduling on distributed cloud data centers, which proved to have a faster learning rate than a Common-Actor MADRL algorithm showing faster convergence in under 100 episodes making it 5 times faster, due to the focus on local neighborhoods, than Common-Actor which required 500 episodes to converge, and other traditional methods such as Greedy (Green-Opt) and Random.

In multi-area power systems, for cooperative load frequency control (LFC), a multi-agent deep deterministic policy gradient (MA-DDPG) was proposed in [18]. It showed better performance on the 3-area system by reducing frequency overshoot, mean ACE and max variation by 57.5%, 37.8% and 17.1% compared to PID and outperformed single-agent DDPG with 13.9-17.9% improvements proving the benefit of cooperation, and with 76.3-87.7% improvements against DQN.

MADRL techniques have proven to outperform single-agent algorithms and traditional methods across several domains addressed in this section showing the efficiency of multi-agent cooperativeness and the adaptability to dynamic environments. Agents are found to be able to coordinate efficiently without needing constant communication due to its centralized training with decentralized execution. However, having enough agents and secure data sharing to unlock its full potential can be a challenge. Nevertheless, MADRL is an emerging key technology for intelligent, cooperative control in areas such as power, energy, traffic systems and other related areas.

IV. BENCHMARKING

In this section, the performance of MADRL algorithms and techniques over single-agent and traditional methods are summarized in Table 1, comparing key MADRL algorithms such as DQN, A2C, MA-DDPG, adv.RAIM against WMMSE, FP, single-agent DDPG and other methods to highlight MADRL's efficiency in coordination and adaptability to dynamic conditions.

TABLE 1: BENCHMARKING

Citation	Algorithms Used	Task	Environment	Key Metrics	Performance
[18]	Proposed MA-DDPG	Cooperative Load Frequency Control (LFC) in Multi-Area Power Systems	3-Area Power System	Action Value Q Mean Absolute ACE (Area Control Error) [%] Largest Variation of ACE [p.u]	Action Value Q: -0.0105 Mean Absolute ACE: 0.023 Largest variation of ACE: 0.029
	Fine-tune PID				Action value Q: -0.0247 Mean absolute ACE: 0.037 Largest variation of ACE: 0.035
	Deep Q-learning				Action value Q: -0.0851 Mean absolute ACE: 0.093 Largest variation of ACE: 0.048
	Single-agent DDPG				Action value Q: -0.0122 Mean absolute ACE: 0.028 Largest variation of ACE: 0.031
	Proposed MA-DDPG (with Generation Rate Constraint, GRC and Governor Deadband, GDB)				Action value Q: -1.2e-3 Mean absolute ACE: 0.029 Largest variation of ACE: 0.048
	Fine-tune PID (with GRC and GDB)				Action value Q: -1.8e-3 Mean absolute ACE: 0.042 Largest variation of ACE: 0.049
	Deep Q-learning (with GRC and GDB)				Action value Q: -3.2e-3 Mean absolute ACE: 0.061 Largest variation of ACE: 0.049
	Single-agent DDPG (with GRC and GDB)				Action value Q: -1.5e-3 Mean absolute ACE: 0.038 Largest variation of ACE: 0.048
	MA2C (proposed algorithm)			Reward $\bar{R}$ Avg. Queue Length [veh]	Reward $\bar{R}$: -78.7 Avg. queue length: 0.75 Avg. intersection delay: 104 Avg. vehicle speed: 14.26 Trip completion flow: 2.40 Trip delay: 1701
	Greedy				Reward $\bar{R}$: -86.4 Avg. queue length: 1.08 Avg. intersection delay: 272

[15]		Large-Scale Adaptive Traffic Signal Control (ATSC) in Urban Networks	Real-world traffic network of Monaco city (at peak congestion times)	Avg. Intersection Delay [s/veh] Avg. Vehicle Speed [m/s] Trip Completion Flow [veh/s] Trip Delay [s]	Avg. vehicle speed: 14.96 Trip completion flow: 2.10 Trip delay: 2077
	IA2C				Reward $\bar{R}$: -117.9 Avg. queue length: 1.16 Avg. intersection delay: 316 Avg. vehicle speed: 14.26 Trip completion flow: 2.10 Trip delay: 2418
	IQL-LR				Reward $\bar{R}$: -202.1 Avg. queue length: 2.21 Avg. intersection delay: 202 Avg. vehicle speed: 14.26 Trip completion flow: 1.60 Trip delay: 2755
	IQL-DNN				Reward $\bar{R}$: -256.2 Avg. queue length: 2.69 Avg. intersection delay: 238 Avg. vehicle speed: 13.98 Trip completion flow: 1.20 Trip delay: 3283
[16]	Proposed MADRL with parameter sharing, local rewards, action masking, and a priority-based safety supervisor	Highway On-Ramp Merging in Mixed Traffic for Autonomous Vehicles	Custom Open-Source Highway Simulator with Easy, Medium and Hard Traffic Modes Single Through-Lane Case	Collision Rate Avg. Speed [m/s]	Collision Rate: 0 (all 3 modes) Avg. Speed: 25.72 (Easy), 24.08 (Medium), 22.73 (Hard)
	MPC				Collision Rate: 0.03, 0.03, 0.40 Avg. Speed: 22.05, 19.67, 21.02
	MAA2C				Collision Rate: 0.02, 0.08, 0.52 Avg. Speed: 21.00, 19.33, 19.68
	MAACKTR				Collision Rate: 0.08, 0.12, 0.18 Avg. Speed: 24.71, 21.94, 18.19
	MAPPO				Collision Rate: 0, 0.02, 0.34 Avg. Speed: 25.70, 24.00, 22.41
	Proposed MADRL algorithm		Custom Open-Source Highway Simulator (Gym-like) Multiple Through-Lane Case		Collision Rate: 0 (all 3 modes) Avg. Speed: 23.53, 21.05, 20.95
	MAA2C				Collision Rate: 0, 0.03, 0.40 Avg. Speed: 19.78, 15.16, 22.04
	MAACKTR				Collision Rate: 0.03, 0.07, 0.27 Avg. Speed: 22.71, 20.07, 21.60
	MAPPO				Collision Rate: 0, 0.03, 0.10 Avg. Speed: 23.07, 19.59, 19.65

V. GAPS

While Multi-Agent Deep Reinforcement Learning (MADRL) has achieved considerable advancements across domains such as traffic control, UAV coordination, and energy management, the field continues to face critical gaps that hinder its broader adoption and scalability. These gaps emerge across technological, research, and application domains, affecting both the efficiency and generalization of MADRL frameworks. The following subsections summarize the key limitations and underexplored areas identified through recent literature.

A. Technological Gaps

Despite architectural advances such as centralized training and decentralized execution (CTDE) and communication-enhanced coordination, MADRL systems continue to face significant technological constraints. One of the most critical challenges is limited scalability. As the number of agents grows, computational requirements increase exponentially, resulting in high communication overhead and instability during joint policy optimization. The complexity of multi-agent environments makes maintaining stable learning dynamics difficult, especially when agents must operate with partial observability and shared state representations [1, 9].

Another major technological limitation is the **c**ommunication bottleneck during centralized training. In high-dimensional or continuous-action environments, agents must exchange large volumes of information, which creates latency and bandwidth constraints that degrade coordination efficiency. This challenge becomes more pronounced when scaling to dozens or hundreds of agents, such as in smart cities or swarm robotics scenarios [1, 10]. Studies on traffic light control confirm that unstable data exchange among agents leads to degraded performance and convergence delays, especially under dynamic network conditions [5].

A further issue is computational resource dependency. Many state-of-the-art MADRL models require high end GPUs and extensive training iterations to achieve convergence. This limits deployment in edge computing and embedded systems, where

lightweight models and energy efficiency are necessary. For example, UAV-assisted mobile edge computing and HVAC control systems require low-latency decision-making under limited computational power, yet most existing MADRL algorithms cannot meet these hardware constraints [11, 12].

B. Research Gaps

At the algorithmic and theoretical level, MADRL research faces persistent challenges in learning stability, generalization, and exploration efficiency. One of the most pressing issues is training instability and non-convergence. Since each agent's policy continually changes during learning, the environment becomes non-stationary, violating the assumptions of traditional reinforcement learning. This causes oscillations and prevents convergence to stable equilibria [1, 2].

In addition, generalization to unseen environments remains a key research gap. Most MADRL algorithms are highly task specific and fail to transfer knowledge across different domains or agent configurations. The absence of adaptive policy transfer and meta-learning techniques results in poor generalization when environmental conditions change. This limitation is emphasized across several survey papers, which identify the lack of domain-independent policy structures as a critical weakness of current MADRL research [9, 10].

Another unresolved challenge lies in cooperative exploration. Many existing algorithms still rely on noise-based exploration strategies derived from single-agent reinforcement learning. These approaches are inefficient in multi-agent systems, especially in sparse reward environments where coordinated behavior among agents is essential. Cooperative Multi-Agent Exploration (CMAE) was proposed as a partial solution to enhance exploration through shared goals, but even this approach faces limitations when scaling to heterogeneous or competitive agents [8].

Finally, benchmarking inconsistencies persist across MADRL studies. Due to the lack of standardized environments and evaluation criteria, comparing results between different frameworks remains difficult. Many papers introduce custom environments or hyperparameter settings, leading to biased performance assessments and limited reproducibility [1, 13].

C. Domain Gaps

From an application perspective, MADRL research has been concentrated primarily in network optimization, traffic management, and energy control systems, leaving several other domains largely unexplored. There is a noticeable underrepresentation of MADRL in fields such as healthcare, cybersecurity, industrial automation, and social robotics, where multi-agent coordination could be highly impactful. This uneven distribution of research attention highlights a need for broader domain adaptation and cross-disciplinary integration [10, 13].

In existing application domains, environmental assumptions often oversimplify real-world dynamics. For example, UAV coordination models typically assume perfect synchronization and full observability of states, ignoring noise, communication failures, or human factors present in actual deployments [11]. Similarly, traffic control applications often disregard uncertainties arising from unpredictable driver or pedestrian behaviors, leading to optimistic but unrealistic simulation outcomes [5].

Another key domain-level limitation is the lack of cross-domain transferability. Current models are designed for narrow, task-specific optimization and cannot be reused across domains without significant retraining. Early work has suggested the use of transfer learning to accelerate multi-agent adaptation, but few studies have implemented these techniques effectively on large-scale systems [2].

VI. Conclusion

In summary, Multi-Agent Deep Reinforcement Learning (MADRL) has advanced rapidly through architectures such as CTDE, actor critic frameworks, attention mechanisms, and hierarchical control, yet its progress remains bounded by engineered constraints rather than emergent cooperation. While current methods achieve stability through structured dependence and regularization, they struggle with scalability, convergence, and generalization in dynamic real-world environments. Bridging these limitations will require shifting from constraint-based designs to adaptive, self-organizing systems capable of learning stability autonomously transforming MADRL from a framework of controlled coordination into one of genuine collective intelligence.

ACKNOWLEDGMENT

The authors would like to express their sincere gratitude to Mr. Prasan Yapa and Mr. Dilanka Perera for their invaluable guidance, constructive feedback, and continuous support throughout the preparation of this review paper. Their insights were instrumental in shaping the quality and depth of the analysis. The authors also extend their appreciation to Robert Gordon University for providing access to unique resources, research facilities, and essential tools that enabled a comprehensive and thorough review of the literature on Multi-Agent Deep Reinforcement Learning.

REFERENCES

[1] *A. Wong, T. Bäck, A. V. Kononova, and A. Plaat, "Deep multiagent reinforcement learning: challenges and directions," Artificial Intelligence Review, Oct. 14, 2022. doi: https://link.springer.com/article/10.1007/s10462-022-10299-x*

[2] *M. Egorov, "Multi-Agent Deep Reinforcement Learning." http://vision.stanford.edu/teaching/cs231n/reports/2016/pdfs/122_Report.pdf*

[3] *Y. S. Nasir and D. Guo, "Multi-Agent Deep Reinforcement Learning for Dynamic Power Allocation in Wireless Networks," IEEE Journal on Selected Areas in Communications, vol. 37, no. 10, pp. 2239–2250, Oct. 2019. https://ieeexplore.ieee.org/abstract/document/8792117*

[4] *N. Naderializadeh, J. Sydir, M. Simsek, and H. Nikopour, "Resource Management in Wireless Networks via Multi-Agent Deep Reinforcement Learning," IEEE Transactions on Wireless Communications, pp. 1–1, Jun. 2021. https://ieeexplore.ieee.org/document/9329087*

[5] *T. Wu et al., "Multi-Agent Deep Reinforcement Learning for Urban Traffic Light Control in Vehicular Networks," IEEE Transactions on Vehicular Technology, vol. 69, no. 8, pp. 8243–8256, Aug. 2020. https://ieeexplore.ieee.org/abstract/document/9103316*

[6] *A. Guillen-Perez and M.-D. Cano, "Multi-Agent Deep Reinforcement Learning to Manage Connected Autonomous Vehicles at Tomorrows Intersections," IEEE Transactions on Vehicular Technology, pp. 1–1, Jul. 2022. https://ieeexplore.ieee.org/abstract/document/9762548*

[7] *R. Shen et al., "Multi-agent deep reinforcement learning optimization framework for building energy system with renewable energy," Applied Energy, vol. 312, p. 118724, Apr. 15, 2022. https://www.sciencedirect.com/science/article/abs/pii/S0306261922001829*

[8] *I.-J. Liu, U. Jain, R. A. Yeh, and A. Schwing, "Cooperative Exploration for Multi-Agent Deep Reinforcement Learning," PMLR, pp. 6826–6836, Jul. 2021. https://proceedings.mlr.press/v139/liu21j.html*

[9] *S. Gronauer and K. Diepold, "Multi-agent deep reinforcement learning: a survey," Artificial Intelligence Review, Apr. 15, 2021. https://link.springer.com/article/10.1007/s10462-021-09996-w*

[10] *T. T. Nguyen, N. D. Nguyen, and S. Nahavandi, "Deep Reinforcement Learning for Multiagent Systems: A Review of Challenges, Solutions, and Applications," IEEE Transactions on Cybernetics, pp. 1–14, Mar. 2020. https://ieeexplore.ieee.org/abstract/document/9043893*

[11] *L. Wang, K. Wang, C. Pan, W. Xu, N. Aslam, and L. Hanzo, "Multi-Agent Deep Reinforcement Learning-Based Trajectory Planning for Multi-UAV Assisted Mobile Edge Computing," IEEE Transactions on*

Cognitive Communications and Networking, vol. 7, no. 1, pp. 73–84, Mar. 2021. https://ieeexplore.ieee.org/abstract/document/9209079

[12] *L. Yu et al., "Multi-Agent Deep Reinforcement Learning for HVAC Control in Commercial Buildings," IEEE Transactions on Smart Grid, pp. 1–1, Jan. 2021. https://ieeexplore.ieee.org/abstract/document/9146920*

[13] *W. Du and S. Ding, "A survey on multi-agent deep reinforcement learning: from the perspective of challenges and applications," Artificial Intelligence Review, vol. 54, no. 5, pp. 3215–3238, Nov. 24, 2020. https://link.springer.com/article/10.1007/s10462-020-09938-y*

[14] *A. Feriani and E. Hossain, "Single and Multi-Agent Deep Reinforcement Learning for AI-Enabled Wireless Networks: A Tutorial," IEEE Communications Surveys & Tutorials, vol. 23, no. 2, pp. 1226–1252, 2021, doi: https://doi.org/10.1109/comst.2021.3063822.*

[15] *T. Chu, J. Wang, L. Codeca, and Z. Li, "Multi-Agent Deep Reinforcement Learning for Large-Scale Traffic Signal Control," IEEE Transactions on Intelligent Transportation Systems, pp. 1–10, 2019, doi: https://doi.org/10.1109/tits.2019.2901791.*

[16] *D. Chen et al., "Deep Multi-Agent Reinforcement Learning for Highway On-Ramp Merging in Mixed Traffic," IEEE transactions on intelligent transportation systems, vol. 24, no. 11, pp. 11623–11638, Nov. 2023, doi: https://doi.org/10.1109/tits.2023.3285442.*

[17] *A. Jayanetti, Saman Halgamuge, and Rajkumar Buyya, "Multi-Agent Deep Reinforcement Learning Framework for Renewable Energy-Aware Workflow Scheduling on Distributed Cloud Data Centers," IEEE transactions on parallel and distributed systems, pp. 1–12, Jan. 2024, doi: https://doi.org/10.1109/tpds.2024.3360448.*

[18] *Z. Yan and Y. Xu, "A Multi-Agent Deep Reinforcement Learning Method for Cooperative Load Frequency Control of Multi-Area Power Systems," IEEE Transactions on Power Systems, pp. 1–1, 2020, doi: https://doi.org/10.1109/tpwrs.2020.2999890.*

[19] *D. Chen et al., "PowerNet: Multi-agent Deep Reinforcement Learning for Scalable Powergrid Control," IEEE Transactions on Power Systems, pp. 1–1, 2021, doi: https://doi.org/10.1109/tpwrs.2021.3100898.*

[20] *J.-D. Zhang, Z. He, W.-H. Chan, and C.-Y. Chow, "DeepMAG: Deep reinforcement learning with multi-agent graphs for flexible job shop scheduling," Knowledge-Based Systems, vol. 259, pp. 110083–110083, Jan. 2023, doi: https://doi.org/10.1016/j.knosys.2022.110083.*