Towards Low-Dimensional Search for Mastering Multi-Agent Planning

Sizhe Tang[1], **Yu Li**[1], **Mahdi Imani** [2], **Tian Lan** [1]

[1]The George Washington University
[2] Northeastern University
s.tang1@gwu.edu, yul@gwu.edu, m.imani@northeastern.edu, tlan@gwu.edu

Abstract

Monte Carlo Tree Search (MCTS) faces a severe scalability bottleneck in Multi-Agent Planning (MAP) due to the combinatorial explosion of joint action spaces. In this position paper, we argue that the key to tractable planning lies in leveraging low-dimensional representational structures to guide the tree search, rather than enumerating the raw action space. Specifically, under a linear approximation of joint-action returns, we demonstrate that the node expansion problem can be effectively cast and solved as a linear contextual bandit, providing theoretical regret guarantees. Empirical results on complex benchmarks confirm that this structure-aware search significantly outperforms state-of-the-art baselines, offering a scalable path for neuro-symbolic multi-agent planning. We further discuss promising extensions, including integrating dynamic agent grouping and coalition formation mechanisms to further reduce the effective branching factor.

Introduction

Monte Carlo Tree Search (MCTS) (Browne et al. 2012) has demonstrated remarkable performance in solving complex planning problems, ranging from competitive gaming to robotic control (Leisiazar et al. 2023), combinatorial optimization (Xiao, Liu, and Zhuo 2023) and network optimization (Li et al. 2026). By leveraging the Upper Confidence Bound for Trees (UCT) (Kocsis and Szepesvári 2006) to balance exploration and exploitation, MCTS achieves significantly higher data efficiency than standard model-free reinforcement learning (Schrittwieser et al. 2020; Ye et al. 2021). When integrated with deep neural networks, as exemplified by AlphaZero (Silver et al. 2017) and MuZero (Schrittwieser et al. 2020), it enables agents to perform deliberate look-ahead reasoning, achieving superhuman performance with minimal domain knowledge. However, the efficacy of standard MCTS relies heavily on the assumption that the action space is manageable enough for the tree search to visit promising branches with sufficient frequency (Browne et al. 2012).

This assumption breaks down fundamentally in cooperative multi-agent planning (MAP) (Hernandez-Leal, Kartal, and Taylor 2019; Mei et al. 2023). The central bottleneck is the combinatorial explosion of the joint-action space: for a system with n agents and d actions per agent, the size of the joint-action set grows exponentially as d^n (Lowe et al. 2017). Consequently, the branching factor during tree expansion becomes intractable. Naive MCTS approaches, which treat each joint action as an independent arm, fail to gather meaningful statistics within practical simulation budgets. The search tree becomes overwhelmingly sparse, and the planner inevitably degenerates into near-random exploration, unable to identify coordinated sequences of actions required for complex tasks (Kwak et al. 2024; Liu et al. 2024).

Addressing this scalability crisis requires revisiting the fundamental design of the search mechanism. Existing Multi-Agent Reinforcement Learning (MARL) methods have attempted to bypass this issue through value factorization (e.g., VDN (Sunehag et al. 2017), QMIX (Rashid et al. 2020)), imposing structural constraints like linearity or monotonicity to decompose the joint value function. While these methods successfully handle high-dimensional spaces during training, they typically yield reactive policies that lack the test-time reasoning capabilities of MCTS (Browne et al. 2012). Conversely, existing multi-agent planning attempts often limit search to rigid state abstractions or ignore the structural dependencies between agents entirely (Liu et al. 2024). A critical gap therefore remains: how can we retain the rigorous, uncertainty-aware planning of MCTS while incorporating the structural inductive biases necessary to navigate exponential spaces?

In this position paper, building on our prior work (Tang, Chen, and Lan 2025) which introduced a linear bandit-based planning framework for multi-agent MCTS, we argue that the curse of dimensionality in MAP can be overcome by shifting the search paradigm from exhaustive enumeration to structure-aware exploration over low-dimensional subspaces. Here, we consolidate this perspective, provide a self-contained presentation of the core methodology and its theoretical foundations, and outline promising future directions including dynamic agent grouping and coalition formation. We posit that joint-action returns in cooperative settings are not unstructured black boxes; rather, they exhibit inherent low-dimensional patterns—such as additivity or sparse interactions—that can be exploited to guide tree expansion. By approximating the joint return as a function of latent, low-dimensional components, we can transform the intractable

problem of selecting among d^n arms into a statistically efficient bandit problem. This approach allows the planner to perform precise credit assignment during the search, identifying which specific subsets of agents contribute to high-value outcomes without wasting computation on irrelevant combinations.

To validate this perspective, we propose a planning framework that leverages a linear approximation strategy: we model joint returns through a linear combination of latent per-agent rewards, formulating the tree expansion as a contextual linear bandit problem. This provides a rigorous foundation for exploration with provable regret bounds that scale linearly with the number of agents. Empirical evaluations on benchmarks demonstrate that this structure-aware approach significantly outperforms both model-free baselines and standard MCTS, offering a promising path toward scalable neuro-symbolic multi-agent intelligence. We further outline future directions, such as incorporating adaptive agent grouping and coalition formation strategies inspired by recent advances in multi-agent skill discovery.

Related Works and Background

Problem Formulation. We formulate the cooperative multi-agent planning task as a Decentralized Partially Observable Markov Decision Process (Dec-POMDP) (Oliehoek, Amato et al. 2016; Xu et al. 2023; Zhang et al. 2025), defined by the tuple $\mathcal{M} = \langle \mathcal{I}, \mathcal{S}, \mathcal{A}, P, R, \Omega, \mathcal{O}, \gamma \rangle$. Here, $\mathcal{I} = \{1, \dots, n\}$ denotes the set of n agents. At each step t, agents execute a joint action $\mathbf{a}_t = (a_t^1, \dots, a_t^n) \in \mathcal{A} \equiv \prod_{i=1}^n \mathcal{A}_i$, causing a state transition $s_{t+1} \sim P(\cdot|s_t, \mathbf{a}_t)$ and yielding a shared reward $r_t = R(s_t, \mathbf{a}_t)$ (Tang et al. 2025). The objective is to find a joint policy $\boldsymbol{\pi}$ maximizing the expected discounted return $J(\boldsymbol{\pi}) = \mathbb{E}[\sum_{t=0}^{\infty} \gamma^t r_t]$.

Factorized Representations in MARL. To tackle high-dimensional action spaces, MARL methods often employ value factorization (Li et al. 2024; Zhou, Lan, and Aggarwal 2022). Algorithms such as VDN (Sunehag et al. 2017) and QMIX (Rashid et al. 2020) approximate the joint action-value function Q_{tot} as a linear or monotonic combination of individual utilities. While these structural assumptions effectively reduce sample complexity during training, they produce reactive policies and do not natively support the uncertainty quantification required for the look-ahead search in MCTS. Our work seeks to bridge this gap by incorporating such low-dimensional representational structures directly into the MCTS exploration mechanism.

MCTS-Based Planning. MCTS approximates optimal policies via look-ahead search, balancing exploration and exploitation using the Upper Confidence Bound (UCB) (Auer, Cesa-Bianchi, and Fischer 2002). Recent model-based extensions like MuZero (Schrittwieser et al. 2020) replace the simulator with learned dynamics and value models, achieving success in single-agent domains. In the selection phase, MuZero typically uses a Probabilistic UCB (pUCT) rule:

$$a_t = \arg\max_{a \in \mathcal{A}} \left[\Phi(s,a) + c(s) P(s,a) \frac{\sqrt{\sum_b N(s,b)}}{1 + N(s,a)} \right], \tag{1}$$

where Φ estimates the value, P is a prior, $N(s,a)$ denotes the visiting count of a node and $c(s)$ is a constant. However, in multi-agent settings, the joint action space grows as $|\mathcal{A}| = d^n$, making the enumeration in the argmax operation computationally intractable. Existing solutions like Sampled MuZero (Hubert et al. 2021) address this by restricting the search to a randomly sampled subset $T(s) \subset \mathcal{A}$. While this allows the algorithm to run, reliance on unguided or heuristic sampling is highly sample-inefficient for discovering sparse, coordinated joint actions in combinatorial spaces.

Methodology

To overcome the curse of dimensionality in multi-agent planning, we propose to replace the exhaustive enumeration of joint actions with structure-aware exploration. We introduce a framework that leverages low-dimensional representational structures to approximate the joint-action value function $Q(s, \mathbf{a})$, thereby guiding the tree search efficiently.

Linear Approximation for Efficient Search

Our framework tackles the scalability bottleneck by hypothesizing that the joint return $r(\mathbf{a})$ can be effectively approximated through a low-dimensional linear structure. Instead of learning a monolithic value function over the exponentially large joint space $\mathcal{A}$, we decompose the problem into latent agent-wise components.

Low-Dimensional Linear Modeling. Let $\mathbf{a} = (a^1, \dots, a^n)$ denote a joint action. We introduce a feature mapping $\phi(\mathbf{a}) \in \mathbb{R}^D$ that is additively decomposable, defined as $\phi(\mathbf{a}) = \sum_{i=1}^n \psi_i(a^i)$, where $\psi_i : \mathcal{A}_i \to \mathbb{R}^d$ maps an individual agent's action to a d-dimensional embedding (resulting in a total dimension $D = nd$). We model the expected joint return as:

$$\mathbb{E}[r(\mathbf{a})] = \langle \phi(\mathbf{a}), \theta^* \rangle, \tag{2}$$

where $\theta^* \in \mathbb{R}^D$ is an unknown parameter vector. This formulation reduces the learning problem from estimating d^n values to estimating nd parameters, effectively casting the node expansion as a contextual linear bandit problem (Lattimore and Szepesvári 2020).

Generalized Loss and Optimization. To estimate θ^* robustly, we move beyond simple squared-error minimization. We introduce a generalized convex loss function $\mathcal{L} : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$, which measures the divergence between the predicted return $\hat{r} = \langle \phi(\mathbf{a}), \theta \rangle$ and the observed reward r. We assume L is strictly convex and μ-smooth, i.e., $|\nabla L(x) - \nabla L(y)| \leq \mu |x - y|$ for all x, y. This flexibility allows the framework to adapt to different reward distributions or to enforce asymmetric penalties (e.g., penalizing overestimation of sub-optimal branches).

At time step t, given the history $\mathcal{H}_t = \{(\mathbf{a}_\tau, r_\tau)\}_{\tau=1}^{t-1}$, we compute the estimator $\hat{\theta}_t$ by solving the regularized minimization problem:

$$\hat{\theta}_t = \arg\min_{\theta \in \mathbb{R}^D} \left(\sum_{\tau=1}^{t-1} \mathcal{L}(\langle \phi(\mathbf{a}_\tau), \theta \rangle, r_\tau) + \frac{\lambda}{2} \|\theta\|_2^2 \right), \quad (3)$$

where $\lambda > 0$ is a regularization parameter. For the specific case where $\mathcal{L}$ is the squared loss, this recovers the closed-form Ridge Regression solution.

Selection Rule. To balance exploration and exploitation, we derive a confidence ellipsoid centered at $\hat{\theta}_t$. The geometry of this ellipsoid is determined by the covariance matrix $\mathbf{V}_t$, defined as:

$$\mathbf{V}_t = \lambda \mathbf{I} + \sum_{\tau=1}^{t-1} \phi(\mathbf{a}_\tau) \phi(\mathbf{a}_\tau)^\top. \quad (4)$$

Leveraging the μ-smoothness of the loss and self-normalized martingale concentration inequalities, the joint action $\mathbf{a}_t$ is selected to maximize:

$$\mathbf{a}_t = \arg\max_{\mathbf{a} \in \mathcal{A}} \left[\langle \phi(\mathbf{a}), \hat{\theta}_t \rangle + \beta_t \sqrt{\phi(\mathbf{a})^\top \mathbf{V}_t^{-1} \phi(\mathbf{a})} \right], \quad (5)$$

where β_t is a scalar exploration parameter.

Crucially, because the features $\phi(\mathbf{a}) = \sum_{i=1}^{n} \psi_i(a^i)$ are additively decomposable, we can show that the selection objective is monotone submodular. Let $\mathcal{S}$ denote the set of selected actions and rewrite $V(\mathcal{S}) = \lambda I + \sum_{\mathbf{a} \in \mathcal{S}} \phi(\mathbf{a}) \phi(\mathbf{a})^\top$. Define the set function:

$$\Psi(\mathcal{S}) = \sum_{\mathbf{a} \in \mathcal{S}} \langle \phi(\mathbf{a}), \hat{\theta}_t \rangle + \sum_{\mathbf{a} \in \mathcal{S}} \beta_t \sqrt{\phi(\mathbf{a})^\top V(\mathcal{S})^{-1} \phi(\mathbf{a})}. \quad (6)$$

Theorem 1 (Submodularity of Ψ) *Ψ is a non-negative monotone submodular function over the ground set $\mathcal{A}$.*

The first term is modular (and thus submodular) over agent actions, while the second term is a concave composition over a modular function, which preserves submodularity. Since maximizing a monotone submodular function is NP-hard in general (Nemhauser, Wolsey, and Fisher 1978; Fisher, Nemhauser, and Wolsey 1978), we cannot solve it exactly. However, efficient approximation algorithms exist. Let $\mathcal{A} = \bigsqcup_{i=1}^{n} B_i$ be partitioned into n blocks so that any feasible joint action contains exactly one element from each B_i (i.e., an n-hot constraint).

Theorem 2 ($(1 - \frac{1}{e})$-Approximation) *There exists a $(1 - \frac{1}{e})$-approximation algorithm for the optimization of action selection.*

(a) Cardinality case $|\mathcal{S}| \leq T$. *The standard greedy algorithm that iteratively selects*

$$A_t = \arg\max_{a \in \mathcal{A} \setminus \mathcal{S}_{t-1}} \left[\Psi(\mathcal{S}_{t-1} \cup \{a\}) - \Psi(\mathcal{S}_{t-1}) \right],$$

$$\mathcal{S}_t = \mathcal{S}_{t-1} \cup \{A_t\},$$

returns $\mathcal{S}_T$ satisfying $\Psi(\mathcal{S}_T) \geq (1 - \frac{1}{e}) \Psi(\mathcal{S}^\star)$, where $\mathcal{S}^\star$ is an optimal subset of size at most T (Nemhauser, Wolsey, and Fisher 1978).

(b) n-Hot (partition-matroid) case. *One may apply the continuous greedy algorithm to the multilinear relaxation over the matroid polytope, followed by pipage rounding, to produce a feasible set $\hat{\mathcal{S}}$ with $\Psi(\hat{\mathcal{S}}) \geq (1 - \frac{1}{e}) \Psi(\mathcal{S}^\star)$ (Calinescu et al. 2011).*

Thus, under the n-hot partition-matroid constraint inherent to the joint action structure, there exists an efficient algorithm to compute action selection with a $(1 - \frac{1}{e})$-approximation guarantee.

Theoretical Analysis. We provide a theoretical guarantee for the performance of the proposed algorithm. The following theorem characterizes the cumulative realized regret $\hat{R}_T = \sum_{t=1}^{T} (r(\mathbf{a}_t^*) - r(\mathbf{a}_t))$, showing that it scales polynomially with the system size rather than exponentially.

Theorem 3 *[Regret Bound of Linear Approximation-Based Algorithm] With probability $1 - \delta$, the regret satisfies*

$$\begin{aligned} \hat{R}_t &\leq \sqrt{8 \mu t \beta_t \ln \left(\frac{\det(V_t)}{\det(\lambda I)} \right)} \\ &\leq \sqrt{8 \mu n d t \beta_t \ln \left(\frac{n d \lambda + \mu n t}{n d \lambda} \right)}. \end{aligned} \quad (7)$$

Corollary 4 (The Order of Regret Bound) *Under the above conditions, the cumulative regret bound with $\delta = 1/T$ satisfies*

$$\hat{R}_T = O\left(nd \cdot \sqrt{\mu T} \cdot \ln(T) \right). \quad (8)$$

Experiments

We evaluate the proposed framework (MALinZero) against state-of-the-art baselines including MARL methods (QMIX (Rashid et al. 2020), MAPPO (Yu et al. 2022)) and MCTS-based planners (MAZero (Liu et al. 2024), MAZero-NP, MA-AlphaZero) on MatGame and the StarCraft Multi-Agent Challenge (SMAC) (Samvelyan et al. 2019). MatGame serves as a fundamental testbed that generalizes the normal-form setting to n agents, where a shared reward is derived by querying a predefined payoff tensor for simultaneous discrete actions. The reported scores in Table 1 represent the cumulative reward averaged over 3 random seeds, with $\pm$ denoting standard deviation.

Main Results

As shown in Table 1, the proposed linear approximation-based algorithm outperforms all baselines across all MatGame configurations. The performance gap widens with problem complexity, reaching 11% in high-dimensional spaces (8^{10} joint actions), thereby validating the efficacy of our low-dimensional representation. Notably, the proposed algorithm excels even in non-linear reward structures by effectively avoiding local optima where baselines stagnate. Furthermore, it achieves faster convergence with computational costs comparable to MAZero (Liu et al. 2024), as the LinUCB formulation reduces sampling complexity from exponential $\mathcal{O}(d^n)$ to linear $\mathcal{O}(dn)$.

Agent	Action	Type	Steps	MAZero	MAPPO	QMIX	Ours
6	8	Linear	1000	393.7 ± 9.9	390.6 ± 9.2	386.1 ± 10.4	$\mathbf{396.6 \pm 8.4}$
6	8	Linear	2000	434.2 ± 7.2	431.8 ± 8.4	430.1 ± 9.5	$\mathbf{439.8 \pm 6.8}$
6	8	Non-Linear	1000	399.8 ± 13.7	388.8 ± 13.1	390.5 ± 12.2	$\mathbf{410.6 \pm 8.9}$
6	8	Non-Linear	2000	443.9 ± 12.1	430.1 ± 8.5	431.7 ± 7.6	$\mathbf{451.1 \pm 12.8}$
8	10	Linear	1000	618.8 ± 16.9	617.1 ± 11.1	612.7 ± 15.4	$\mathbf{637.1 \pm 15.8}$
8	10	Linear	2000	692.7 ± 14.5	681.8 ± 12.5	679.4 ± 12.7	$\mathbf{705.2 \pm 15.7}$
8	10	Non-Linear	1000	615.2 ± 18.7	561.4 ± 20.9	558.7 ± 19.1	$\mathbf{630.1 \pm 16.3}$
8	10	Non-Linear	2000	672.3 ± 16.1	657.1 ± 17.3	648.2 ± 18.7	$\mathbf{693.4 \pm 15.6}$

Table 1: Evaluation in MatGame with different numbers of agents and actions. We consider both linear and non-linear reward structures. The proposed linear approximation-based algorithm is shown to outperform both MCTS and MARL baselines.

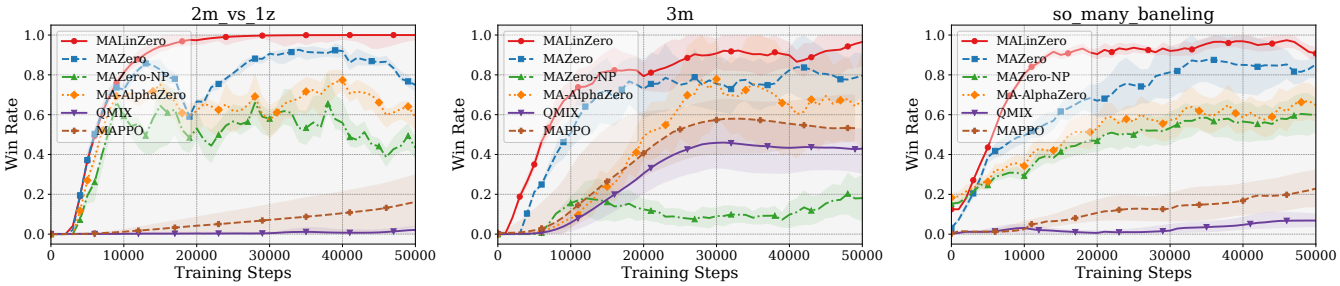

Figure 1: Evaluations on 3 SMAC tasks/maps. Y-axis denotes the win rate and X-axis denotes training steps. Each algorithm is executed with 3 random seeds. The proposed MALinZero achieves over 96% win rate on all 3 maps.

We further evaluate on the high-dimensional SMAC benchmarks. As shown in Figure 1, MALinZero achieves over 96% win rate across all three tasks (2m_vs_1z, 3m, and so_many_baneling), outperforming all baselines including MAZero, MAZero-NP, MA-AlphaZero, QMIX, and MAPPO. Crucially, MALinZero exhibits superior sample efficiency, converging to high win rates significantly faster than its closest competitors. For instance, on the 3m map, MALinZero reaches above 90% win rate within approximately 10,000 training steps, while MAZero and other baselines require considerably more steps to achieve comparable performance. This gain stems from the linear bandit formulation, which enables statistically efficient exploration by decomposing the joint-action space into per-agent components, allowing the planner to perform effective credit assignment without exhaustive enumeration of the combinatorial action space.

Conclusion and Future Work

In this work, we addressed the fundamental scalability bottleneck of Multi-Agent MCTS by exploiting the low-dimensional structure of joint-action returns. We proposed a linear approximation-based framework to navigate the exponential action space without exhaustive enumeration, projecting returns into a linear space and formulating the node expansion as a contextual linear bandit problem optimized via submodular maximization. Theoretically, we proved that the framework achieves sublinear regret with sample complexity scaling linearly in the number of agents rather than exponentially in the joint action space (d^n).

Several promising directions remain for future investigation. First, we plan to develop fully learnable decomposable representations that can automatically discover the most informative low-dimensional structures from data, thereby further generalizing the linear approximation proposed in this work. Second, an important avenue is to integrate dynamic agent grouping and coalition formation mechanisms into the planning framework. Recent advances in multi-agent skill discovery (Chen, Lan, and Aggarwal 2024) have demonstrated that agents in cooperative tasks often form latent subgroups whose compositions shift over time, and that explicitly modeling these subgroup coordination patterns can substantially improve learning efficiency and transferability. Incorporating such adaptive grouping strategies into our tree search could allow the planner to first identify which subsets of agents require tight coordination at a given state, and then restrict the combinatorial search to within and across these dynamically formed coalitions, further reducing the effective branching factor. Third, extending the proposed frameworks to partially observable and communication-constrained settings, where agents must reason about both coordination structure and information asymmetry, presents a natural and challenging next step toward fully scalable multi-agent planning.

References

Auer, P.; Cesa-Bianchi, N.; and Fischer, P. 2002. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47: 235–256.

Browne, C. B.; Powley, E.; Whitehouse, D.; Lucas, S. M.;

Cowling, P. I.; Rohlfshagen, P.; Tavener, S.; Perez, D.; Samothrakis, S.; and Colton, S. 2012. A survey of monte carlo tree search methods. *IEEE Transactions on Computational Intelligence and AI in games*, 4(1): 1–43.

Calinescu, G.; Chekuri, C.; Pal, M.; and Vondrák, J. 2011. Maximizing a monotone submodular function subject to a matroid constraint. *SIAM Journal on Computing*, 40(6): 1740–1766.

Chen, J.; Lan, T.; and Aggarwal, V. 2024. Variational offline multi-agent skill discovery. *arXiv preprint arXiv:2405.16386*.

Fisher, M. L.; Nemhauser, G. L.; and Wolsey, L. A. 1978. *An analysis of approximations for maximizing submodular set functions—II*. Springer.

Hernandez-Leal, P.; Kartal, B.; and Taylor, M. E. 2019. A survey and critique of multiagent deep reinforcement learning. *Autonomous Agents and Multi-Agent Systems*, 33(6): 750–797.

Hubert, T.; Schrittwieser, J.; Antonoglou, I.; Barekatain, M.; Schmitt, S.; and Silver, D. 2021. Learning and planning in complex action spaces. In *International Conference on Machine Learning*, 4476–4486. PMLR.

Kocsis, L.; and Szepesvári, C. 2006. Bandit based monte-carlo planning. In *European conference on machine learning*, 282–293. Springer.

Kwak, Y.; Hwang, I.; Kim, D.; Lee, S.; and Zhang, B.-T. 2024. Efficient Monte Carlo tree search via on-the-fly state-conditioned action abstraction. *arXiv preprint arXiv:2406.00614*.

Lattimore, T.; and Szepesvári, C. 2020. *Bandit algorithms*. Cambridge University Press.

Leisiazar, S.; Park, E. J.; Lim, A.; and Chen, M. 2023. An MCTS-DRL based obstacle and occlusion avoidance methodology in robotic follow-ahead applications. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 221–228. IEEE.

Li, Y.; Tang, S.; Chen, R.; Yu, F. X.; Jiang, G.; Imani, M.; Bastian, N. D.; and Lan, T. 2026. ACDZero: Graph-Embedding-Based Tree Search for Mastering Automated Cyber Defense. *arXiv preprint arXiv:2601.02196*.

Li, Z.; Tang, S.; Tian, H.; Xiang, H.; Xu, X.; and Dou, W. 2024. A Crowdsensing Service Pricing Method in Vehicular Edge Computing. In *2024 IEEE International Symposium on Parallel and Distributed Processing with Applications (ISPA)*, 82–89. IEEE.

Liu, Q.; Ye, J.; Ma, X.; Yang, J.; Liang, B.; and Zhang, C. 2024. Efficient Multi-agent Reinforcement Learning by Planning. *arXiv preprint arXiv:2405.11778*.

Lowe, R.; Wu, Y. I.; Tamar, A.; Harb, J.; Pieter Abbeel, O.; and Mordatch, I. 2017. Multi-agent actor-critic for mixed cooperative-competitive environments. *Advances in neural information processing systems*, 30.

Mei, Y.; Zhou, H.; Lan, T.; Venkataramani, G.; and Wei, P. 2023. Mac-po: Multi-agent experience replay via collective priority optimization. *arXiv preprint arXiv:2302.10418*.

Nemhauser, G. L.; Wolsey, L. A.; and Fisher, M. L. 1978. An analysis of approximations for maximizing submodular set functions—I. *Mathematical programming*, 14: 265–294.

Oliehoek, F. A.; Amato, C.; et al. 2016. *A concise introduction to decentralized POMDPs*, volume 1. Springer.

Rashid, T.; Samvelyan, M.; De Witt, C. S.; Farquhar, G.; Foerster, J.; and Whiteson, S. 2020. Monotonic value function factorisation for deep multi-agent reinforcement learning. *Journal of Machine Learning Research*, 21(178): 1–51.

Samvelyan, M.; Rashid, T.; de Witt, C. S.; Farquhar, G.; Nardelli, N.; Rudner, T. G. J.; Hung, C.-M.; Torr, P. H. S.; Foerster, J.; and Whiteson, S. 2019. The StarCraft Multi-Agent Challenge. *CoRR*, abs/1902.04043.

Schrittwieser, J.; Antonoglou, I.; Hubert, T.; Simonyan, K.; Sifre, L.; Schmitt, S.; Guez, A.; Lockhart, E.; Hassabis, D.; Graepel, T.; et al. 2020. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588(7839): 604–609.

Silver, D.; Schrittwieser, J.; Simonyan, K.; Antonoglou, I.; Huang, A.; Guez, A.; Hubert, T.; Baker, L.; Lai, M.; Bolton, A.; et al. 2017. Mastering the game of go without human knowledge. *nature*, 550(7676): 354–359.

Sunehag, P.; Lever, G.; Gruslys, A.; Czarnecki, W. M.; Zambaldi, V.; Jaderberg, M.; Lanctot, M.; Sonnerat, N.; Leibo, J. Z.; Tuyls, K.; et al. 2017. Value-decomposition networks for cooperative multi-agent learning. *arXiv preprint arXiv:1706.05296*.

Tang, S.; Chen, J.; and Lan, T. 2025. MALinZero: Efficient Low-Dimensional Search for Mastering Complex Multi-Agent Planning. *arXiv preprint arXiv:2511.06142*.

Tang, S.; Xia, X.; Bilal, M.; Dou, W.; and Xu, X. 2025. Human-centric service offloading with cnn partitioning in cloud-edge computing-empowered metaverse networks. *IEEE Transactions on Consumer Electronics*.

Xiao, Y.; Liu, J.; and Zhuo, H. H. 2023. BalMCTS: Balancing Objective Function and Search Nodes in MCTS for Constraint Optimization Problems. *arXiv preprint arXiv:2312.15864*.

Xu, X.; Tang, S.; Qi, L.; Zhou, X.; Dai, F.; and Dou, W. 2023. Cnn partitioning and offloading for vehicular edge networks in web3. *IEEE Communications Magazine*, 61(8): 36–42.

Ye, W.; Liu, S.; Kurutach, T.; Abbeel, P.; and Gao, Y. 2021. Mastering atari games with limited data. *Advances in neural information processing systems*, 34: 25476–25488.

Yu, C.; Velu, A.; Vinitsky, E.; Gao, J.; Wang, Y.; Bayen, A.; and Wu, Y. 2022. The surprising effectiveness of ppo in cooperative multi-agent games. *Advances in neural information processing systems*, 35: 24611–24624.

Zhang, Z.; Zhou, H.; Imani, M.; Lee, T.; and Lan, T. 2025. Learning to collaborate with unknown agents in the absence of reward. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 14502–14511.

Zhou, H.; Lan, T.; and Aggarwal, V. 2022. Pac: Assisted value factorization with counterfactual predictions in multi-agent reinforcement learning. *Advances in Neural Information Processing Systems*, 35: 15757–15769.