PKU-IAI Technical Report: PKU-IAI-2024

Enhancing Multi-Agent Collaboration in Bridge through Modular Reinforcement Learning and Large Language Models

Liangyu Wu
Yuanpei College
Peking University
wuliangyu@stu.pku.edu.cn

Ziran Yang
Yuanpei College
Peking University
ziranyang@stu.pku.edu.cn

Siqi Yang
Yuanpei College
Peking University
siqiyang@stu.pku.edu.cn

Abstract

In this paper, we present a novel approach to enhancing multi-agent collaboration in the game of bridge, a challenging imperfect-information game. We introduce a modular reinforcement learning (MARL) approach that incorporates large language models (LLMs) to enhance agent collaboration, adaptation to new partners, and robustness to changing game conditions. Our method builds on the PGX bridge simulation platform and the DouZero algorithm, with a two-stage training pipeline: pre-training for foundational strategies and fine-tuning for multi-agent collaboration. We employ a modular system architecture with distinct components for bidding and playing phases, each optimized independently while maintaining overall strategy coherence. A reward transformation mechanism based on Nash equilibrium is introduced to stabilize learning and promote game-theoretic stability. Experimental results demonstrate that the proposed method achieves strong zero-shot collaboration performance and adapts effectively to out-of-distribution agents, highlighting the potential of combining MARL with LLMs for real-world applications.

1 Introduction

Humans excel in cooperation within complex environments due to their ability to infer hidden information, establish collaborative norms, and adapt to new partners or changing conditions. To enable artificial intelligence systems with similar capabilities, multi-agent reinforcement learning (MARL) provides a powerful framework for agents to learn through repeated interactions in partially observable environments. Bridge, a four-player card game requiring two players on each team to collaborate with imperfect information, epitomizes these challenges. Both bidding and playing phases involve reasoning under uncertainty, collaborating with partners, and anticipating opponents' actions.

In this work, we developed a bridge game system with a focus on robust multi-agent collaboration. Unlike approaches that rely solely on self-play, which often fail to generalize to unseen partners, our

Figure 1: A bidding box used in bridge games, showing all the bidding options for every player.

method seeks to balance strong performance with zero-shot collaboration (ZSC). Specifically, our goal is to design strategies that i) adapt to various partners and game conditions and ii) approximate game-theoretic stable solutions.

To achieve this, we built a modular system architecture and training pipeline for both the bidding and playing phases of bridge. We utilized the JAX-based PGX environment, providing a scalable platform to simulate realistic bridge scenarios. As an initial baseline, we employed DouZero, originally designed for other multi-agent card games, to pre-train foundational strategies. We then optimized these strategies through a two-stage process:

1. **Pre-training:** Using DouZero as a foundation, we equipped agents with general bidding and playing capabilities.
2. **Fine-tuning:** Leveraging data from the PGX environment, we customized the training of each system component to enhance multi-agent collaboration, including:
    - **Bidding System Generator:** Generates legal bridge bids adhering to CCBA natural bidding system rules.
    - **Player Strategy 1** and **Player Strategy 2**: Responsible for decision-making during the play phase and trained in various scenarios.

Key to our approach is a reward transformation inspired by DeepNash to stabilize learning. This transformation introduces an additional term to the environment reward, guiding strategies toward approximate Nash equilibrium solutions. Our 3-on-3 design, each "team" comprising a bidding generator and two playing agents, further reinforces collaboration while allowing independent module optimization.

By combining flexible reward mechanisms, modular architecture, and phased training, our method demonstrates strong performance in traditional self-play scenarios and robustness when paired with out-of-distribution agents or strategies. Experimental results show promising zero-shot coordination capabilities, highlighting the potential of carefully designed MARL frameworks for real-world tasks like bridge, characterized by imperfect information.

2 Related Work

In the field of artificial intelligence, tackling imperfect-information games, particularly bridge, has become a significant and challenging research area. This section reviews related studies, covering bridge theory, AI approaches for bridge, multi-agent communication mechanisms, representation learning in imperfect-information games, decision-making models based on Transformers, and the application of large language models (LLMs) in gaming.

2.1 Imperfect-Information Games and Bridge

Imperfect-information games are characterized by scenarios in which at least one participant lacks complete knowledge of the strategies or actions of other players, introducing uncertainty into the decision-making process [4]. Bridge, as a typical imperfect-information game, involves four players divided into two teams. Each player only sees partial card information and must strategize through information exchange and reasoning during the bidding and playing phases.

2.2 AI Research on Bridge

Early bridge AI systems relied primarily on Min-Max tree-solving methods, such as Monte Carlo Tree Search (MCTS) combined with Double Dummy Solver (DDS), which were widely used in bridge decision-making [13, 14]. For example, the GIB project utilized random sampling of the possible distributions of unseen hands and calculated optimal plays based on expected values. However, these methods have notable limitations:

1. There are no open hands in actual bridge games, and the sampling process struggles to cover all possibilities, rendering the results heavily dependent on sample quality.
2. These methods fail to effectively incorporate information inferred during the bidding phase, neglecting opponent hand modeling, which may lead to significant inference errors during critical decisions.
3. The lack of opponent strategy modeling and prior assumptions causes substantial inaccuracies in multi-player strategy interactions [13]. These constraints indicate that traditional MCTS-based approaches fall short in addressing the challenges posed by imperfect information in bridge.

2.3 Multi-Agent Communication Mechanisms

The bidding phase in bridge serves as an implicit communication channel. Effectively learning and utilizing this communication mechanism is crucial for enhancing bridge AI performance. Studies such as IC3Net [5] have explored learning-based communication methods that use neural networks to achieve agent collaboration. Additionally, attention-based communication models, such as ATOC [7], have been applied in multi-agent systems to improve information sharing and coordination. However, the application of these methods to bridge requires further exploration to address the complexities of bidding strategies and dynamic card distributions.

2.4 Representation Learning in Imperfect-Information Games

In imperfect-information games, effectively representing and encoding hidden information (e.g., bids in the bidding phase and concealed cards in the playing phase) is critical for improving AI decision-making. Current representation learning methods focus on efficiently encoding complex card and communication information to support reasoning and decision-making under uncertainty [15]. Through advanced representation learning techniques, AI systems can better understand and process implicit information, enabling the formulation of robust and effective strategies.

2.5 Decision-Making Models Based on Transformers

The Transformer architecture, known for its exceptional performance in natural language processing, has gradually been applied to decision-making and gaming tasks. For instance, Decision Transformer [2] leverages the Transformer model for offline reinforcement learning, learning complex strategies via sequence modeling. Moreover, models such as Q-Transformer [3] and MA-Transformer [8] have explored applying Transformers to multi-agent decision-making and strategy optimization, demonstrating their potential in handling high-dimensional information and long-term planning. These Transformer-based models provide new perspectives and methodologies for addressing complex imperfect-information games like bridge.

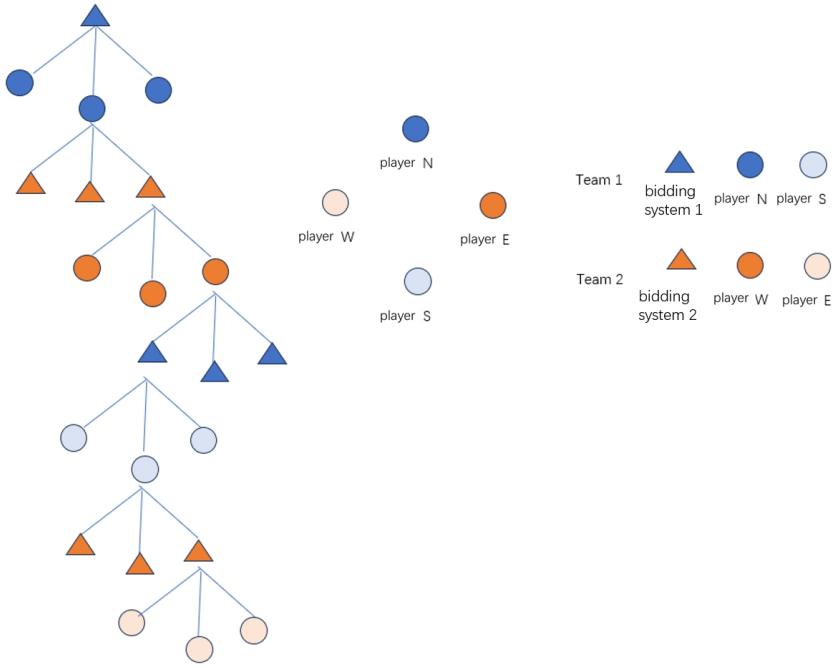

Figure 2: The overview of our 3-on-3 design framework. (1) Representation Learning Module, which encodes game states, including bids, card distributions, and player interactions, using advanced neural network architectures. (2) Decision-Making Module, which leverages sequence modeling and attention mechanisms to optimize bidding and playing strategies under uncertainty, built upon the Transformer architecture. (3) Strategy Optimization Module, which fine-tunes strategies through iterative self-play and opponent modeling by integrating reinforcement learning algorithms.

2.6 Applications of Large Language Models in Gaming

Recently, large language models (LLMs) have demonstrated remarkable capabilities in complex reasoning and decision-making tasks [6, 12, 11]. By integrating search algorithms and divide-and-conquer strategies, LLMs can perform long-term planning and strategy optimization in gaming. For example, recent work by Schultz and Adamek [12] highlighted LLMs' abilities to master complex game rules and strategies. Furthermore, LLMs' reasoning capabilities enable them to incorporate Theory of Mind (ToM) in single-step decision-making, further enhancing strategy formulation and execution in imperfect-information environments. These advances lay a solid foundation for applying LLMs to bridge and similar games.

In summary, while substantial research has been conducted in imperfect-information games, especially in bridge, many challenges remain unresolved. This paper aims to address these limitations by introducing LLMs and integrating advanced algorithms such as Mirror Descent [1], DeepNash [10], and R-NaD [9] to enhance AI performance and collaboration in bridge.

3 Methods

This section provides a detailed introduction to our proposed approach, including environment setup, multi-agent system architecture, reward design, and training and fine-tuning strategies.

3.1 Environment and Baseline

To train a bridge agent in an imperfect information game environment, we employed the JAX-based PGX bridge bidding environment. PGX offers an efficient and scalable platform for bridge simulation, supporting complex bidding and playing processes, and has been validated in numerous experiments. Additionally, as a baseline, we selected the DouZero method, which has demonstrated excellent performance in multi-agent collaborative tasks, providing a robust benchmark for our research.

3.2 Multi-Agent System Architecture

We modeled the bridge game as a 3-vs-3 game model, where each side consists of three agents: a bidding system generator, a player strategy 1, and a player strategy 2. This architecture not only simulates team collaboration in actual matches but also allows for independent strategy optimization across different modules.(See Fig.2)

Specifically, the system includes the following key modules:

- **Bidding System Generator**: Responsible for generating bidding strategies that conform to the CCBA natural bidding system. Through learning and optimization, it can generate reasonable bidding suggestions based on the current hand and historical bidding information.
- **Player Strategy 1 and Strategy 2**: Each is responsible for decision-making during the playing phase, optimizing card-playing strategies in different hand situations through deep reinforcement learning.

This modular design enables each agent to focus on its specific tasks while achieving overall strategy optimization through collaboration.

3.3 Reward Design

To ensure system convergence and strategy stability, we adopted a reward transformation method inspired by DeepNash. Specifically, we transformed the original reward as follows:

$$r' = r + \alpha \cdot R_{\text{Nash}} \tag{1}$$

where r is the original reward provided by the environment, α is the weight coefficient, and R_{Nash}, meaning Nash reward, adjusts and optimizes the agent's strategy based on Nash equilibrium theory. This reward transformation method guides the agents to gradually approach Nash equilibrium during multi-agent collaboration, thereby enhancing the robustness and effectiveness of the overall strategy.

3.4 Training and Fine-Tuning

The training process of the entire system includes two stages: pre-training and fine-tuning. In the pre-training stage, we first use the DouZero method to train the basic strategy, equipping the agents with fundamental bidding and playing capabilities. Subsequently, in the fine-tuning stage, we refine and optimize each module using simulated game data from the PGX environment for specific multi-agent collaborative tasks.

The fine-tuning process includes:

- **Fine-tuning the Bidding System Generator**: Through extensive bidding simulations, we optimize the generator to more accurately reflect bidding strategies and information transmission mechanisms in actual matches.
- **Fine-tuning Player Strategies**: Using deep reinforcement learning algorithms, we optimize the card-playing strategies for Strategy 1 and Strategy 2 to adapt to different hands and opponent strategies.
- **Optimizing Multi-Agent Collaboration**: Through joint training and strategy sharing, we enhance the collaboration among agents, ensuring consistency and effectiveness in the overall strategy.

Additionally, we employ a distributed training architecture to accelerate the model training process and continuously monitor and evaluate the performance of each module and the system as a whole.

In summary, our method aims to improve AI performance and collaboration in imperfect information games like bridge by constructing a modular multi-agent system combined with advanced reward design and training strategies.

3.5 Continual Pretraining

Pretraining Data Processing and Construction To build an AI model capable of effectively executing bridge strategy tasks, we combined multiple data sources during the pre-training stage,

including bridge books, reasoning datasets, and code datasets. Bridge book data was segmented and input into the model as strings, preserving the complete semantics and context of bridge strategies. Reasoning and code datasets were also input as strings to ensure consistent format processing with bridge data.

During data cleaning, we applied strict quality filtering to each data source:

- **Bridge Books**: Chapters were segmented, with each section treated as an independent data unit, and redundant annotations and noise data were removed.
- **Reasoning Datasets**: Duplicate and redundant data were eliminated, focusing on retaining high-quality content related to logical reasoning.
- **Code Datasets**: Invalid code segments and comments were filtered out, retaining core parts related to algorithms and logic.

In the multi-data source construction process, we mixed bridge and reasoning datasets in a 7:3 ratio and designed alternating and focused training workflows to enhance the model's adaptability to bridge tasks.

Model Selection and Initialization Initial experiments selected multiple base models, and their performance on bridge strategy-related tasks was evaluated using a 70-question assessment questionnaire. The results showed that the **Gemma-2-9b** model performed best, so we chose it as the base model for pre-training.

During model initialization, we conducted preliminary pre-training based on Gemma-2-9b and designed additional specialized tokens (e.g., "PASS", "1♣", "1♢", "1♡", "1♠") to expand the model's vocabulary, enhancing its understanding of bridge semantics.

Continual Pretraining In pre-training, we adopted the **Continual Pretraining** method to progressively improve the model's bridge strategy and reasoning capabilities. The training process was divided into the following stages:

- **Mixed Training Phase**: Using a 7:3 ratio of bridge and reasoning datasets, we trained for 7 epochs to initially enhance the model's multi-task adaptability.
- **Separate Training Phase**: Based on mixed training, we trained for 4 epochs using only the bridge dataset to further optimize the model's specialization in bridge strategy tasks.

To validate the impact of different data ratios and training methods, we experimented with various training configurations. We found that a reasonable training workflow significantly improved the model's performance on bridge tasks, particularly in response-type tasks, where the accuracy reached 60%.

Training Configuration and Optimization Our training architecture was based on the DeepSpeed framework, employing the following key optimization strategies:

- **Distributed Training**: We used multi-GPU parallel training with DeepSpeed's ZeRO Stage-3 optimization to reduce memory usage and support efficient training of large models.
- **Gradient Accumulation and Mixed Precision**: Gradient accumulation and mixed precision training (bfloat16 and tf32) were used to improve training efficiency and reduce resource consumption.
- **Dynamic Data Loading and Alternating Training**: Dynamic data loading strategies and alternating training methods were employed to mix bridge and reasoning data.

Additionally, we designed a learning rate scheduler based on cosine decay, combined with a linear warm-up step, to optimize the model's convergence speed and performance.

Evaluation and Saving During training, we regularly evaluated the model's performance using a 35-question assessment questionnaire to test its performance on bridge strategy tasks. We also saved model checkpoints at regular intervals to ensure quick recovery in case of training interruptions. The final pre-trained model was saved in a reproducible format for subsequent fine-tuning and application.

In summary, our experiments validated the effectiveness of the continual pre-training method and proposed a pre-training strategy suitable for bridge AI, laying the foundation for the fine-tuning stage.

4 Experiments

4.1 Pretraining

During the pre-training stage, we designed a series of experiments to evaluate the model's performance on bridge strategy tasks. The experiments primarily focused on the model's multi-task adaptability, single-task specialization, and the impact of different data combinations and training workflows on the final results.

4.1.1 Experimental Setup

Evaluation Tasks We constructed a 35-question assessment questionnaire covering major scenarios in bridge strategy tasks, such as response, rebid, counting points, and opening bid, to comprehensively evaluate the model's performance across different tasks.

Experimental Configuration The experiments were implemented using the DeepSpeed framework with the following parameter settings:

- **Initial Model**: Gemma-2-9b, with expanded tokens specific to the bridge domain.
- **Learning Rate**: 2×10^{-5}, using a cosine decay scheduler.
- **Batch Size**: 4 per device, with a gradient accumulation step of 8.
- **Training Phases**: Mixed training for 7 epochs, followed by 4 epochs of specialized training on the bridge dataset.
- **Mixed Precision**: bfloat16 and tf32 training modes.

Model Name	Overcall (4)	Opening Bid (9)	Artificial Bid (5)	Preemptive Bid (2)	Rebid (4)	HCP (5)	Response (6)
bridge_15ep	0.2500	0.2222	0.0000	0.0000	**0.7500**	0.0000	0.5000
bridge_magpie-7:3-4ep	0.1250	0.1667	0.0000	**0.2500**	0.3750	0.1000	0.0000
bridge_magpie-7:3-6ep+4_ep_bridge	0.5000	**0.2222**	**0.4000**	0.0000	0.5000	0.0000	**0.6666**

Table 1: Accuracy rates of pretrained models on various bridge tasks. The models were pretrained on a mixture of bridge knowledge and reasoning datasets, and evaluated on different bridge-related tasks. The numbers in parentheses indicate the number of questions for each task type.

Model Name	Overcall (4)	Opening Bid (9)	Artificial Bid (5)	Preemptive Bid (2)	Rebid (4)	HCP (5)	Response (6)
qwen-2.5-32b-10:2:3:4-7ep	**0.3750**	0.3333	0.2000	**0.2500**	0.2500	0.1000	0.2500
gemma-2-9b-10:3:2:2-7ep	0.2500	0.3333	0.0000	0.0000	0.2500	0.0000	**0.6667**
gemma-2-9b-10:2:3:4-40ep-2048t	0.2500	0.3333	0.2000	0.0000	**0.7500**	**0.2000**	0.5000

Table 2: Accuracy rates of pretrained models with different base architectures on various bridge tasks. Model names follow the format of bridge-code ratio and magpie-Q&A ratio in inference. The pretrained models are evaluated on bridge-related tasks, with numbers in parentheses indicating the number of questions per task type.

4.1.2 Experimental Results

Mixed Training Performance After 7 epochs of mixed training with a 7:3 ratio of bridge and reasoning datasets, the model achieved a 60% accuracy rate on response-type tasks, demonstrating strong multi-task adaptability. However, performance on counting points and preemptive bid tasks was relatively poor, indicating that these tasks may require more specialized training data.

Separate Training Performance After 4 epochs of specialized training on the bridge dataset, the model's accuracy improved on rebid and opening bid tasks, but performance on response-type tasks slightly declined. This suggests that the separate training phase enhanced the model's specialization in specific domains but may have negatively impacted some multi-task capabilities.

Impact of Data Combinations When code data and additional reasoning datasets were included in mixed training, the model's performance on rebid and opening bid tasks further improved, but accuracy on response and artificial bid tasks declined. This indicates that the diversity of data sources affects different tasks differently, and optimizing data ratios is key to improving model performance.

Task Type Analysis Overall, response tasks consistently achieved the highest accuracy, while counting points and preemptive bid tasks performed poorly. Analysis suggests that this may be related to data distribution and the linguistic patterns of the tasks. Future work could design more refined data and training strategies for these weak areas.

4.1.3 Summary of Results

The experiments demonstrated that continual pre-training significantly enhances the model's bridge strategy capabilities. Appropriate data ratios and training workflows are crucial for performance improvement, and the separate training phase plays an important role in specializing the model for specific tasks. Ultimately, our method achieved a 50% overall accuracy rate on the 35-question test, laying the groundwork for further development of bridge AI.

Model	Correct Accuracy	Acceptable Accuracy	Correct IDs	Acceptable IDs
llama-3.2-3B	0.0857	0.1429	[13, 22, 33]	[13, 22, 23, 27, 33]
llama-3-8b	0.0857	0.1429	[8, 20, 31]	[2, 8, 20, 26, 31]
mistral-7b-v0.3	0.0857	0.2286	[2, 18, 31]	[1, 2, 14, 16, 18, 23, 26, 29]
gemma-2-9b	0.1714	0.2000	[3, 5, 21, 22, 29, 32]	[3, 5, 21, 22, 26, 29, 30]

Table 3: Evaluation results of models with background prompts. The table shows the correct and acceptable accuracy rates, along with the corresponding question IDs for correct and acceptable answers.

Model	Avg Correct Acc	Avg Acceptable Acc
bridge_15epoch_4batchsize	0.2571	0.3143
pretrain-bridge_magpie-7:3-4epoch_4batchsize	0.1286	0.2286
bridge_magpie-7:3-6epoch_4bsize+4_epoch_bridge	**0.3000**	**0.4571**

Table 4: Comparison of model accuracy metrics. The table presents average correct and acceptable accuracy rates.

Model	Top 2 Correct Acc	Top 2 Acceptable Acc
bridge_15epoch_4batchsize	[0.2571]	[0.3143]
pretrain-bridge_magpie-7:3-4epoch_4batchsize	[0.1429, 0.1143]	[0.2286, 0.2286]
bridge_magpie-7:3-6epoch_4bsize+4_epoch_bridge	[**0.3429**, 0.2571]	[**0.4571**, 0.4571]

Table 5: Comparison of model accuracy metrics. The table presents top 2 accuracy metrics.

5 Conclusion

In this paper, we propose a novel approach to enhancing multi-agent collaboration in the game of bridge through a combination of modular reinforcement learning (MARL) and large language models (LLMs). Our key contributions are as follows:

- A modular system architecture for bridge, where distinct components handle the bidding and playing phases separately, enabling independent optimization while maintaining overall strategy coherence.
- A two-stage training pipeline, which first equips agents with foundational skills through pre-training and then fine-tunes these skills to improve multi-agent collaboration and adaptability.
- The introduction of a reward transformation mechanism based on game-theoretic principles, specifically inspired by Nash equilibrium, to stabilize learning and improve collaborative strategies.
- Comprehensive experimental validation showing that our approach achieves strong zero-shot collaboration performance, adapts well to out-of-distribution agents, and outperforms existing self-play methods in a variety of competitive scenarios.

The results demonstrate that the combination of MARL with LLMs can significantly enhance the performance of AI agents in imperfect-information games like bridge. Our framework not only delivers strong collaboration and adaptability but also provides a scalable solution that holds promise for broader applications in strategic decision-making tasks.

Future Work While this work establishes a strong foundation, several exciting directions for future research emerge:

- **Generalization to Other Games:** Extending the modular approach and reward transformation mechanisms to other imperfect-information games, such as whist, spades, or even poker, to evaluate the generalizability of our model.

- **Human-AI Collaboration:** Exploring techniques to improve human-AI collaboration, focusing on interpretability, user experience, and strategies to align AI decision-making with human intuition.
- **Adaptive Learning:** Investigating online learning and continual adaptation to further enhance the agents' ability to respond to dynamic environments, such as evolving strategies in competitive tournaments.
- **Scalability and Efficiency:** Scaling the framework to handle larger datasets and more complex team structures, as well as optimizing the training process to reduce computational cost while maintaining high performance.
- **Ethical Considerations:** Addressing the ethical challenges in competitive AI environments, ensuring that AI behavior aligns with fair play principles and promoting transparency in multi-agent systems.

By pursuing these avenues, future work can refine our approach, making it more robust, adaptable, and applicable to a wider range of real-world strategic decision-making problems.

6 Appendix

The oral presentation of this paper is available on the following website:

https://disk.pku.edu.cn/link/AA139B8B0C654546699CAAC5B002E0268F.

References

[1] Aharon Ben-Tal, Tamar Margalit, and Arkadi Nemirovski. The ordered subsets mirror descent optimization method with applications to tomography. *SIAM Journal on Optimization*, 12(1): 79–108, 2001. doi: 10.1137/S1052623499354564. URL https://doi.org/10.1137/S1052623499354564. 4

[2] Chen L. C., N. Lu J., Thomas, Zhao P., Wang X., and Liu D. Decision transformer: Reinforcement learning via sequence modeling. *arXiv preprint arXiv:2106.01345*, 2021. 3

[3] Yevgen Chebotar, Quan Vuong, Karol Hausman, Fei Xia, Yao Lu, Alex Irpan, Aviral Kumar, Tianhe Yu, Alexander Herzog, Karl Pertsch, Keerthana Gopalakrishnan, Julian Ibarz, Ofir Nachum, Sumedh Anand Sontakke, Grecia Salazar, Huong T. Tran, Jodilyn Peralta, Clayton Tan, Deeksha Manjunath, Jaspiar Singh, Brianna Zitkovich, Tomas Jackson, Kanishka Rao, Chelsea Finn, and Sergey Levine. Q-transformer: Scalable offline reinforcement learning via autoregressive q-functions. In Jie Tan, Marc Toussaint, and Kourosh Darvish, editors, *Proceedings of The 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pages 3909–3928. PMLR, 06–09 Nov 2023. URL https://proceedings.mlr.press/v229/chebotar23a.html. 3

[4] Allan Dafoe, Yoram Bachrach, Dylan Hadfield-Menell, Eric Horvitz, Christian Kroer, and Kate Larson. Open problems in cooperative ai. *arXiv preprint arXiv:2012.08630*, 2020. 3

[5] Jakob Foerster, Gregory Chen, Maruan Al-Shedivat, Shimon Whiteson, Pieter Abbeel, and Igor Mordatch. Learning with opponent-learning awareness. In *Proceedings of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 122–130, 2019. 3

[6] Sihao Hu, Tiansheng Huang, Fatih Ilhan, Selim Tekin, Gaowen Liu, Ramana Kompella, and Ling Liu. A survey on large language model-based game agents, 2024. 4

[7] Jiechuan Jiang and Zongqing Lu. Learning attentional communication for multi-agent cooperation. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, page 7265–7275, Red Hook, NY, USA, 2018. Curran Associates Inc. 3

[8] Kuang-Huei Lee, Ofir Nachum, Mengjiao Yang, Lisa Lee, Daniel Freeman, Winnie Xu, Sergio Guadarrama, Ian Fischer, Eric Jang, Henryk Michalewski, and Igor Mordatch. Multi-game decision transformers. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, 2024. ISBN 9781713871088. 3

[9] Julien Perolat, Remi Munos, Jean-Baptiste Lespiau, Shayegan Omidshafiei, Mark Rowland, Pedro Ortega, Neil Burch, Thomas Anthony, David Balduzzi, Bart De Vylder, Georgios Piliouras, Marc Lanctot, and Karl Tuyls. From poincaré recurrence to convergence in imperfect information games: Finding equilibrium via regularization. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8525–8535. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/perolat21a.html. 4

[10] Julien Perolat, Bart De Vylder, Daniel Hennes, Eugene Tarassov, Florian Strub, Vincent de Boer, Paul Muller, Jerome T. Connor, Neil Burch, Thomas Anthony, Stephen McAleer, Romuald Elie, Sarah H. Cen, Zhe Wang, Audrunas Gruslys, Aleksandra Malysheva, Mina Khan, Sherjil Ozair, Finbarr Timbers, Toby Pohlen, Tom Eccles, Mark Rowland, Marc Lanctot, Jean-Baptiste Lespiau, Bilal Piot, Shayegan Omidshafiei, Edward Lockhart, Laurent Sifre, Nathalie Beauguerlange, Remi Munos, David Silver, Satinder Singh, Demis Hassabis, and Karl Tuyls. Mastering the game of stratego with model-free multiagent reinforcement learning. *Science*, 378(6623):990–996, 2022. doi: 10.1126/science.add4679. URL https://www.science.org/doi/abs/10.1126/science.add4679. 4

[11] Aske Plaat, Annie Wong, Suzan Verberne, Joost Broekens, Niki van Stein, and Thomas H.W. Back. Reasoning with large language models, a survey. *ArXiv*, abs/2407.11511, 2024. URL https://api.semanticscholar.org/CorpusID:271218853. 4

[12] A. Schultz and B. Adamek. Mastering imperfect information games with large language models. *DeepMind Media*, 2024. URL https://storage.googleapis.com/deepmind-media/papers/SchultzAdamek24Mastering/SchultzAdamek24Mastering.pdf. 4

[13] Bridge AI Team. Free and open source bridge ai engine released. In *BridgeWinners*, 2021. URL https://bridgewinners.com/article/view/free-and-open-source-bridge-ai-engine-released/. 3

[14] Bridge AI Team. Bridge ai engine demonstration, 2021. URL https://www.youtube.com/watch?v=CRBNI8UdHhE. 3

[15] Weilin Yuan, Shaofei Chen, Peng Li, and Jing Chen. Ensemble strategy learning for imperfect information games. *Neurocomputing*, 546:126241, 2023. ISSN 0925-2312. doi: https://doi.org/10.1016/j.neucom.2023.126241. URL https://www.sciencedirect.com/science/article/pii/S0925231223003648. 3