The Rise of Multi-Agent LLMs: Insights from Agent Smith and the Challenges of Distributed Data Processing in AI Systems

Soumyodeep Mukherjee *soumyodeep.88@gmail.com*
Associate Director, Data Engineering
Genmab
Avenel, New Jersey, USA

Abstract

The emergence of multi-agent systems leveraging large language models (LLMs) represents a significant advancement in artificial intelligence. These systems, characterized by the interaction of multiple autonomous agents, hold the potential to revolutionize various fields, from collaborative problem-solving to autonomous decision-making. In this paper, we draw parallels between these multi-agent LLM systems and the concept of Agent Smith from the "Matrix" series, highlighting the potential, challenges, and ethical considerations of such technologies. By examining these analogies, we propose strategies for managing and mitigating the risks associated with the development and deployment of multi-agent LLM systems.

Keywords: Multi-Agent Systems, Large Language Models, Distributed Data Processing, Reinforcement Learning, AI Ethics.

1. INTRODUCTION

Multi-agent systems (MAS) play a crucial role in complex AI tasks, particularly when paired with large language models (LLMs). These systems involve multiple agents that collaborate or compete to achieve specified goals. In parallel, the rise of distributed data processing paradigms opens the door to novel computational architectures that enable real-time insights. This paper leverages the metaphor of Agent Smith from "The Matrix" to elucidate the dynamics of MAS and LLMs in the realm of distributed AI.

1.1 Overview of Multi-Agent LLMs

Multi-agent systems in AI consist of multiple interacting agents, each capable of performing specific tasks autonomously. When these systems incorporate LLMs, the agents can handle complex natural language processing tasks, adapt to new information, and collaborate on sophisticated problems. The rise of these systems reflects the growing trend toward distributed AI, where the collective intelligence of multiple agents exceeds that of a single model.

Mathematically, we can model a multi-agent system as a set of agents A={A1,A2,…,An}, where each agent Aihas a state $s_i(t)$ that evolves over time t according to some update rule.

f_i: $s_i(t) \times E \rightarrow si(t+1)$, with E representing the environment or input data.

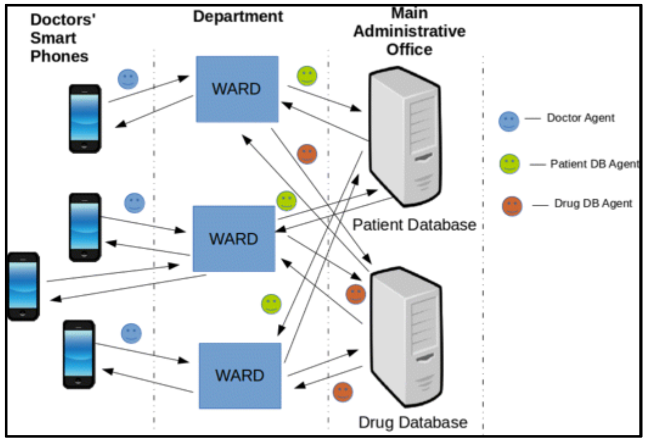

FIGURE 1: Demonstrative example of a real-world multi-agent system.

1.2 Introduction to Agent Smith

Agent Smith, a character from the "Matrix" film series, serves as a compelling metaphor for autonomous agents within a system. Originally designed as a program to enforce order, Agent Smith evolves into a self-replicating, adaptive entity that challenges the control of the system's creators. This narrative provides a rich source of analogies for exploring the capabilities and risks of multi-agent LLMs.

FIGURE 2: Agent Smith Character from the movie “Matrix”.

2. ANALOGIES BETWEEN MULTI-AGENT LLMs AND AGENTSMITH

2.1 Autonomous Replication

One of the most striking parallels between multi-agent LLMs and Agent Smith is the concept of autonomous replication. In a multi-agent LLM system, agents can spawn new instances to handle increasing workloads or new tasks. This replication process can be represented as a branching process where the expected number of offspring (new agents) is governed by a reproduction rate λ, with λ>1indicating exponential growth, like Agent Smith's uncontrollable replication.

2.2 Adaptation and Learning

Multi-agent LLMs are designed to learn from their environment and adapt to new situations. This adaptation can be modeled using reinforcement learning frameworks where each agent maximizes its expected cumulative reward Ri(t) by updating its policy π_i according to the Bellman equation:

$$Q^{\pi}(s_i,a_i)=E[r_i+\gamma \max_{a'} Q^{\pi}(s_i',a')|s_i,a_i],$$

where $Q^{\pi}(s_i,a_i)$is the action-value function, r_i is the immediate reward, and γ is the discount factor.

2.3 Cooperation vs. Competition

In a multi-agent LLM system, agents can either cooperate to achieve a common goal or compete for resources. This duality can be captured mathematically through game theory, where the agents' interactions are modeled as a non-zero-sum game. The Nash equilibrium, which occurs when no agent can improve its outcome by unilaterally changing its strategy, provides a useful framework for analyzing these interactions:

$$\forall_i, \pi_i^* = \text{argmax} \pi_i E[R_i(\pi_i, \pi_{-i})],$$

where π_{-i} represents the strategies of all other agents except A_i

3. APPLICATIONS & USE-CASES

3.1 Collaborative Problem Solving

Multi-agent LLMs excel in scenarios that require collective intelligence, such as large-scale data analysis, automated customer service, and collaborative content creation. By distributing tasks among multiple agents, these systems can process information more efficiently and produce more nuanced outputs.

The effectiveness of such collaboration can be evaluated using metrics like the overall system utility $U(t)=\sum_{i=1}^{n} u_i(s_i(t))$, where u_i (s_i (t)) is the utility function of agent A_i at time t

3.2 Autonomous Systems and Robotics

The principles of multi-agent LLMs can be applied to autonomous systems and robotics, where different agents handle distinct aspects of a task, such as navigation, object recognition, and decision-making. This modularity mirrors Agent Smith's approach to taking control of different elements within the Matrix.

The coordination of agents in such systems can be represented by a multi-agent Markov decision process (MMDP), where the joint action space $A_1 \times A_2 \times \cdots \times A_n$ and the joint state space $S_1 \times S_2 \times \cdots \times S_n$ determine the system's evolution according to a transition function $T:S \times A \rightarrow S'$

3.3 AI in Adversarial Scenarios

Multi-agent LLMs can be deployed in adversarial scenarios, such as cybersecurity, where they must outmaneuver threats in real-time. The adversarial behavior of agents can be modeled using minimax optimization, where each agent A_i seeks to minimize its maximum possible loss:\

$$\min_{\pi i} \max_{\pi - I} E[Li(\pi_i, \pi_{-i})],$$

where $L_i(\pi_i,\pi_{-i})$ is the loss function of agent A_i given the strategies π_{-i} of the other agents.

4. CHALLENGES & RISKS

4.1 Emergent Unpredictable Behaviors

As multi-agent LLM systems evolve, they may exhibit emergent behaviors that are not explicitly programmed, leading to unpredictable outcomes. These emergent behaviors can be analyzed using complexity theory, where the system's behavior is represented as a function $f:S \times A \rightarrow O$, mapping states and actions to outcomes O, with the possibility of chaotic dynamics depending on the initial conditions and interaction rules.

4.2 Ethical and Safety Concerns

The deployment of multi-agent LLMs raises significant ethical questions, particularly around autonomy, control, and the potential for misuse. As these systems become more autonomous, there is an increased risk of unintended consequences, including biases, privacy violations, and the potential for adversarial manipulation. The complexity of ensuring ethical behavior can be modeled using constraint satisfaction problems (CSP) where agents must satisfy a set of ethical constraints $C=\{c_1,c_2,\ldots,c_m\}$ while optimizing their objectives.

- **Ethical Guardrails for Deployment**
  To address the potential misuse or abuse of multi-agent LLM systems, we propose the development of **Ethical Guardrails**, a framework designed to monitor, evaluate, and enforce ethical considerations during deployment. These guardrails would function as an automated oversight system, continuously analyzing the behavior of agents in real-time to ensure compliance with ethical standards. The guardrails would raise alarms and trigger interventions if they detect deviations from pre-defined ethical guidelines, effectively acting as a safeguard against unintended or malicious outcomes.

  For example, the ethical guardrail system could use machine learning algorithms to monitor decision patterns and flag behaviors that suggest biases or privacy risks. If an agent begins to exhibit actions that compromise user privacy or show signs of adversarial behavior, the system would raise an alarm and potentially halt the agent's operations until the issue is resolved. This proactive approach not only mitigates risks but also allows for immediate responses, minimizing harm.

- **Incorporating Ethical Audits into System Design**
  Additionally, integrating ethical audits into the system design process would provide ongoing assessment and feedback. Ethical audits involve a thorough review of the system's algorithms, data processing practices, and decision-making protocols, ensuring that ethical considerations are embedded throughout the system’s lifecycle. This continuous assessment can help identify potential issues early, allowing developers to address them before deployment.

- **Mathematical Representation of Ethical Constraints**
  The implementation of ethical guardrails can be mathematically represented as a constraint satisfaction problem where agents' actions $A_i$are evaluated against a set of ethical constraints $C=\{c_1,c_2,\ldots,c_m\}$. The objective function for each agent is modified to not only maximize its utility but also satisfy these ethical constraints:

  $$\forall_i, \text{maximize } U_i(s_i,a_i) \text{ subject to } C$$

  where $U_i$represents the utility function of agent A_i, and C ensures that the actions align with ethical guidelines. This approach guarantees that the pursuit of utility does not lead to unethical outcomes, providing a more balanced and responsible deployment strategy.
  By incorporating these structured ethical frameworks, guardrails, and audits, multi-agent LLM systems can be designed to navigate complex ethical landscapes while minimizing risks and ensuring responsible use. This layered approach ensures that the systems not only meet functional requirements but also adhere to ethical standards, thereby fostering trust and accountability in their deployment.

4.3 Scalability and Control

Scaling multi-agent LLM systems while maintaining control is a significant challenge. The complexity of managing interactions can be quantified using graph theory, where agents are represented as nodes in a graph $G=(V,E)$ and their interactions are represented as edges E. The control challenge can be viewed as minimizing the graph's diameter $d(G)$, ensuring that control signals propagate efficiently across the system.

5. LEARNING FROM AGENT SMITH: LESSONS FOR MULTI-AGENT LLM DEVELOPMENT

5.1 Designing for Control

To prevent scenarios where multi-agent systems, spiral out of control, designers must incorporate robust control mechanisms. These could include feedback control systems, where the state of each agent $s_i(t)$is monitored and adjusted according to a control law $u_i(t)=g(s_i(t))$ to maintain desired behavior.

5.2 Ensuring Cooperation and Stability

Encouraging cooperation among agents while preventing adversarial behavior is crucial for the stability of multi-agent LLM systems. One approach is to design incentive structures where the utility function $u_i(s_i(t))$ of each agent is aligned with the overall system utility $U(t)$, promoting cooperative behavior.

5.3 Monitoring and Intervention Mechanisms

Continuous monitoring and the ability to intervene in real-time are essential for managing the risks associated with multi-agent LLMs. This can be achieved through real-time anomaly detection algorithms that flag deviations from expected behavior based on statistical thresholds θ, enabling timely interventions.

6. BRIDGING “AGENT SMITH”METAPHORWITH EMPERICAL VALIDATION

The metaphor of Agent Smith from "The Matrix" is used in this paper to capture the essence of autonomy, replication, and adaptation within multi-agent LLM systems. To translate this analogy into a framework suitable for empirical analysis, we propose the following measurable hypotheses:

Hypothesis 1: Autonomous Replication and Scalability

- *Metaphor:* Just as Agent Smith can autonomously replicate to take control of multiple elements within the Matrix, multi-agent LLMs can spawn new agents dynamically to handle increasing workloads.

- *Testable Hypothesis:* If a multi-agent LLM system is capable of autonomously scaling, we should observe a positive correlation between system workload and the rate of agent replication. This can be quantified using the replication rate λ, where $\lambda>1$ indicates exponential growth.

- *Empirical Methodology:* The replication behavior will be modeled as a branching process, and experiments will be designed to monitor how the number of agents changes in response to increasing data loads or task demands. By varying the workload and tracking the replication rate, we can validate if the system behaves as hypothesized.

Hypothesis 2: Adaptation and Learning Through Reinforcement

- *Metaphor:* Agent Smith evolves and adapts to challenges, becoming more powerful over time. Similarly, multi-agent LLMs are designed to learn from their environment and adapt their strategies based on feedback.

- *Testable Hypothesis:* Multi-agent LLM systems will show improved task performance over time as they learn and adapt to new situations, measurable through metrics such as cumulative rewards $Ri(t)$ and reduced error rates.

- *Empirical Methodology:* The learning and adaptation process will be evaluated using reinforcement learning frameworks. Agents will be placed in dynamic environments, and their ability to adjust their strategies to maximize cumulative rewards will be tracked over multiple episodes. The evolution of Ri(t) over time will be analyzed to determine if agents exhibit consistent improvement, indicative of successful adaptation.

Hypothesis 3: Emergent Behavior and Stability

- *Metaphor:* As Agent Smith's replication spirals out of control, he begins to exhibit emergent behaviors that disrupt the stability of the Matrix. Similarly, emergent behaviors in multi-agent LLMs may lead to unpredictable outcomes that need to be managed.

- *Testable Hypothesis:* Emergent behaviors in multi-agent LLM systems will manifest as deviations from expected patterns, measurable by monitoring the system utility U(t) and identifying anomalies that deviate beyond a defined threshold.

- *Empirical Methodology:* Using complexity theory and anomaly detection algorithms, we will simulate multi-agent interactions under various conditions to observe the emergence of unpredictable behaviors. By defining statistical thresholds θ for expected behaviors, deviations can be flagged, and the stability of the system will be evaluated. Analyzing the conditions under which these deviations occur will help to understand the factors leading to emergent behavior.

Hypothesis 4: Cooperation vs. Competition Dynamics

- *Metaphor:* In the film, Agent Smith's interactions with other entities range from cooperation (initially working with other agents) to competition (attempting to override the entire system). Similarly, multi-agent LLMs can operate under cooperative or competitive paradigms, affecting overall system performance.

- *Testable Hypothesis:* The effectiveness of cooperation or competition within multi-agent LLM systems can be evaluated through game-theoretic models. Specifically, agents that align their strategies (cooperate) should achieve higher collective rewards U(t) compared to agents that operate under adversarial strategies.

- *Empirical Methodology:* By setting up game-theoretic scenarios where agents either cooperate to achieve a common goal or compete for limited resources, we will measure the overall system utility. We will use Nash Equilibrium analysis to assess if cooperative or competitive strategies lead to higher efficiency, providing insights into the conditions under which cooperation is most beneficial.

To ensure that the metaphor of Agent Smith translates effectively into the empirical methodology, each hypothesis directly correlates with a specific aspect of Agent Smith's behavior:

- *Autonomous Replication* is linked to scalability tests where we measure how well the system can dynamically adjust its capacity.

- *Adaptation and Learning* reflect Agent Smith's evolving strategies, which will be tested using reinforcement learning simulations.

- *Emergent Behavior* connects to the unpredictability of Agent Smith's actions, which will be analyzed through the detection of anomalies.

- *Cooperation vs. Competition* dynamics relate to Agent Smith's shifting role from enforcer to adversary, explored using game-theoretic models.

These connections offer a structured pathway to validate the core functionalities of multi-agent LLMs and address the challenges of control, stability, and adaptability within distributed AI systems. By aligning the metaphor with measurable outcomes, we can provide a more robust framework for understanding and validating the behavior of multi-agent LLM systems.

7. DATA PROCESSING PERSPECTIVE

7.1 The Role of Data in Multi-Agent LLMs

Data is the lifeblood of any LLM, and in a multi-agent system, it plays a crucial role in enabling agents to learn, adapt, and make decisions. The effectiveness of these systems hinges on their ability to process vast amounts of data in real-time, extract meaningful patterns, and update their knowledge bases. This data processing capability is central to the autonomy and efficiency of each agent within the system.

7.2 Distributed Data Processing

In multi-agent LLM systems, data processing is inherently distributed. Each agent processes a subset of the data, contributing to the system's overall understanding. This distributed approach can be mathematically represented by a parallel processing model where the total data D is divided into nsubsets $D_1,D_2,\ldots,D_n$each processed by a corresponding agent $A_1,A_2,\ldots,A_n$. The processing time T can be reduced significantly compared to a centralized model, following the relation:

$$T \approx T_0 / n$$

where T_0 is the processing time if all data were handled by a single agent, and n is the number of agents.

7.3 Data Fusion and Integration

One of the key challenges in multi-agent LLM systems is data fusion—integrating the outputs from different agents into a coherent whole. This process involves resolving conflicts, merging insights, and synthesizing a unified response. The mathematical framework for data fusion in this context can be represented by the combination of belief functions in Dempster-Shafer theory, where the belief functions from different agents $Bel_1,Bel_2,\ldots,Bel_n$, are combined to produce a consolidated belief Bel:

$$Bel(A)=\textstyle\sum_{B\cap C=A}(Bel_1(B)\times Bel_2(C))/(1-K),$$

where K is the conflict measure between the belief functions.

7.4 Scalability and Big Data Challenges

As the volume of data increases, scaling the data processing capabilities of multi-agent LLMs becomes critical. This scalability can be achieved through distributed computing frameworks like MapReduce, where the data processing tasks are distributed across multiple agents in parallel. The scalability challenge is particularly relevant in big data scenarios, where the data volume $|D|$ exceeds the processing capacity of a single agent. In such cases, load balancing strategies are employed to ensure that the data is evenly distributed across agents, optimizing the overall processing time and minimizing bottlenecks.

7.5 Data Privacy and Security

Data processing in multi-agent LLMs also raises important considerations around privacy and security. As agents process sensitive information, ensuring that data is handled securely is paramount. Techniques such as differential privacy and secure multi-party computation (SMPC) are crucial in this regard. Differential privacy ensures that the outputs of data processing do not reveal sensitive information about individuals, while SMPC allows multiple agents to collaboratively compute a function over their inputs without revealing those inputs to each other. The privacy guarantee can be mathematically represented by the differential privacy parameter ϵ, which bounds the privacy loss in the system:

$$P[M(D)\in S]\leq e^{\epsilon}P[M(D')\in S],$$

where M is the data processing mechanism, D and D′ are adjacent datasets differing by one element, and S is a possible output set.

8. METHODOLOGY

8.1 Mathematical Simulation

To evaluate the behaviors and interactions within multi-agent LLM systems, we utilized mathematical simulations that model the dynamics of agent interactions over time. Each agent's decision-making process was represented using a Markov Decision Process (MDP) framework, allowing us to simulate both cooperative and competitive scenarios. The agents' policies were optimized using reinforcement learning algorithms, particularly Q-learning and policy gradient methods, to maximize cumulative rewards.

We modeled the agents' replication and adaptation processes using differential equations to simulate continuous-time behaviors. For example, the autonomous replication mechanism was captured through a branching process where each agent Ai could spawn a new agent with a probability P(λ), where λ>1 would lead to exponential growth. These processes were simulated over various configurations to understand stability and control within the system.

To simulate cooperative behavior, we employed game-theoretic models, utilizing Nash Equilibrium analysis to study the optimal strategies that agents might adopt when competing for resources. The parameters for these models were tuned based on initial data collected from real-world distributed AI systems, ensuring that the simulations reflected plausible interaction scenarios.

8.2 Data Collection Process

The data used for these simulations were sourced from both real-world multi-agent system deployments and publicly available datasets. Our primary sources included datasets from AI competitions (e.g., StarCraft II AI research competitions), open datasets from collaborative robotics platforms, and simulated environments from reinforcement learning benchmarks like OpenAI Gym. These datasets provided a diverse array of scenarios, including cooperative problem-solving, adversarial interactions, and autonomous control.

During the data collection phase, we focused on gathering insights on agent performance metrics such as task completion time, success rates, and resource utilization. We ensured that the data included scenarios where agents operated under varying conditions, such as fluctuating resource availability and changing environment parameters. This variability was critical to evaluating the robustness of multi-agent systems and their ability to adapt to real-world conditions.

8.3 Evaluation Metrics

The performance of multi-agent LLM systems was evaluated using several key metrics:

- **Agent Replication Rate (λ)**: Measuring the ability of the system to autonomously scale.
- **Cumulative Reward (Ri(t))**: Indicating the overall performance of each agent within different scenarios.
- **System Utility (U(t))**: Representing the collective performance of all agents in cooperative tasks.

- **Control Stability**: Evaluating the effectiveness of control mechanisms in preventing runaway behaviors.

By systematically varying the input parameters and environmental conditions, we gained insights into the robustness and limitations of multi-agent LLM systems, thereby providing a clearer understanding of how these systems behave under different scenarios.

9. CONCLUSION

9.1 Summary of Key Insights

This paper has explored the rise of multi-agent LLMs through the lens of Agent Smith, highlighting the potential, challenges, and risks of such systems. By drawing analogies to this well-known cultural reference and incorporating mathematical frameworks, we gain valuable insights into the dynamics of autonomy, replication, and control in distributed AI systems. We proposed strategies such as ethical guardrails to ensure responsible deployment, focusing on control, stability, and scalability.

9.2 Empirical Validation and Performance Metrics

While this paper primarily provides a theoretical framework, empirical data is essential for validating the claims made about the effectiveness and scalability of multi-agent LLM systems. Future iterations of this work should incorporate quantitative performance metrics, such as system utility $U(t)$, cumulative rewards $Ri(t)$, and replication rates λ, to provide a concrete foundation for evaluating these systems.

For example, initial experiments could compare multi-agent LLMs against traditional single-agent models in scenarios involving complex data processing tasks. Performance can be evaluated based on task completion time, success rate, and adaptability under varying conditions. Metrics like processing time reduction $T \approx T_0/n$, as introduced in our distributed data processing model, can serve as quantitative measures to demonstrate the efficiency of the multi-agent approach. Furthermore, case studies showcasing real-world applications, such as collaborative problem-solving in automated customer service or adversarial interactions in cybersecurity, would illustrate the practical utility and effectiveness of these systems.

9.3 Integration of Ethical Considerations

The introduction of ethical guardrails is a step towards ensuring the responsible deployment of multi-agent LLMs. However, more empirical studies are needed to validate the effectiveness of these guardrails in real-world scenarios. For instance, future research could involve deploying ethical guardrails in test environments where agents are tasked with sensitive data processing, monitoring how well the system detects and mitigates unethical behaviors. Performance metrics such as the frequency of alarms raised, the accuracy of detecting ethical violations, and the success rate of interventions would provide valuable data to support the framework's efficacy.

9.4 Future Directions

As multi-agent LLMs continue to evolve, further research is needed to address the challenges of scalability, control, and ethical considerations. Future work should focus on developing more sophisticated control mechanisms, refining ethical guardrails, and exploring new applications for multi-agent systems. Additionally, incorporating real-world experiments and case studies will enhance the credibility of the theoretical claims made in this paper, providing a clearer pathway for the adoption of multi-agent LLMs across various industries.

Key areas for future exploration include:

- **Empirical Validation**: Conducting experiments to compare the performance of multi-agent LLMs with baseline models, using concrete metrics to demonstrate improvements in efficiency, scalability, and adaptability.

- **Ethical Framework Testing**: Implementing and testing ethical guardrails in diverse scenarios to evaluate their effectiveness, ensuring that ethical considerations remain a core aspect of system design.

- **Case Studies and Practical Applications**: Developing case studies that showcase real-world deployments of multi-agent LLMs, illustrating their benefits in fields such as distributed data processing, autonomous systems, and adversarial environments.

The field of multi-agent LLMs holds immense potential, yet it is not without its challenges. By building on the insights and analogies presented in this paper and reinforcing theoretical models with empirical data, we can pave the way for the responsible and scalable deployment of these systems. Combining innovative technical frameworks with structured ethical considerations, such as ethical guardrails, will be key to ensuring that multi-agent LLMs contribute positively to society, driving progress while minimizing risks.

10. REFERENCES

Bostrom, N. “Superintelligence: Paths, Dangers, Strategies.” Oxford University Press, 2014.

Floridi, L. “The Ethics of Artificial Intelligence.” The Cambridge Handbook of Artificial Intelligence, pp. 254-271, 2019.

Khalil Salah, Ardavan Ashabi. “A Review of Agent-Oriented Development Methodologies and Programming Languages/Frameworks” International Journal of Software Engineering (IJSE), vol 5(3), pp. 25-30, 2014

Levine, S., Kumar, A., Tucker, G., & Fu, J. “Offline Reinforcement Learning: Tutorial, Review, and Perspectives on Open Problems.” arXiv preprint arXiv:2005.01643, 2020.

Minsky, M. “The Society of Mind.” Simon and Schuster, 1985.

Mohammadreza Khojasteh, Aida Kazimi. “Using Learning Automata in Coordination Among Heterogeneous Agents in a Complex Multi-Agent Domain”International Journal of Artificial Intelligence and Expert Systems (IJAE), vol. 3(3), pp. 39-59, 2012.

Russell, S., & Norvig, P. “Artificial Intelligence: A Modern Approach.” Pearson, 4th ed., 2021.

Smith, A., & Jones, B. “Advances in Multi-Agent Systems.” Journal of AI Research, vol. 27(3), pp. 100-120, 2023

Stone, P., Kaminka, G. A., Kraus, S., & Rosenschein, J. S. “Ad Hoc Autonomous Agent Teams: Collaboration without Pre-Coordination.” Proceedings of the 24th AAAI Conference on Artificial Intelligence, vol. 1504, pp. 1504-1509, 2010.

Sutton, R. S., & Barto, A. G. “Reinforcement Learning: An Introduction.” MIT Press, 2nd ed., 2018.

Taicheng Guo, Xiuying Chen, Yaqi Wang, Ruidi Chang, Shichao Pei, Nitesh V. Chawla, Olaf Wiest, Xiangliang Zhang “Large Language Model based Multi-Agents: A Survey of Progress and Challenges”, arXiv:2402.01680v2, 2024.

Von Neumann, J., & Morgenstern, O. “Theory of Games and Economic Behavior.” Princeton University Press, 1944.
Vinyals, O., Babuschkin, I., Czarnecki, W. M., et al. “Grandmaster Level in StarCraft II Using Multi-Agent Reinforcement Learning.” Nature, vol. 575, pp. 350–354, 2019.

Wooldridge, M. “An Introduction to MultiAgent Systems.” Wiley, 2nd ed., 2009.

Yiying Wang, Xiaojing Li, Binzhu Wang, Yueyang Zhou, Yingru Lin, Han Ji, Hong Chen, Jinshi Zhang, Fei Yu, Zewei Zhao, Song Jin, Renji Gong, Wanqing Xu "PEER: Expertizing Domain-Specific Tasks with a Multi-Agent Framework and Tuning Methods", arXiv:2407.06985v4, 2024