

Adaptive Multi-Agent Reinforcement Learning for Autonomous Workflow Optimization in Large Language Model Systems

Raul Hervia

Department of Computer Science and Engineering, University at Buffalo, Buffalo, NY, USA.
raul.hervia@buffalo.edu

Thijmen Henric van den Nuwenhuse

School of Information Technology, University of Cincinnati, Cincinnati, OH, USA.
thijmen.henricvandennuwenhuse@uc.edu

Abstract

Large language model systems increasingly operate as composite socio-technical infrastructures in which planning, retrieval, tool use, verification, routing, and human oversight are distributed across multiple agents. This paper examines adaptive multi-agent reinforcement learning as a system-level control layer for autonomous workflow optimization in such environments. The central argument is that workflow optimization cannot be reduced to isolated prompt tuning or model selection; it requires adaptive coordination among agents whose capabilities, costs, latency profiles, and governance constraints differ. The paper synthesizes reinforcement learning foundations, multi-agent actor-critic methods, foundation model risks, and machine learning systems engineering to articulate an architectural framework for adaptive workflow governance. It analyzes structural trade-offs among autonomy, reliability, cost, fairness, sustainability, and auditability. It further considers deployment realities, including observability, failure containment, human escalation, carbon accounting, and policy compliance. Through cross-domain comparisons with supply chain orchestration, cloud autoscaling, and safety-critical control, the paper argues that adaptive multi-agent reinforcement learning can improve throughput and resilience only when bounded by explicit contracts, evaluation gates, and institutional oversight. The conclusion proposes that the most credible path forward is not maximal autonomy but governed adaptivity, where reinforcement learning optimizes coordination while constitutional, legal, and ethical constraints remain externally specified and continuously audited.

Keywords

multi-agent reinforcement learning; large language models; workflow optimization; autonomous agents; AI governance; socio-technical systems; sustainable AI; robustness.

1. Introduction

Large language model systems have moved from single-turn text generation toward distributed workflows that combine planning, retrieval, code execution, tool invocation, verification, and human review [3], [4], [5]. In these settings, the relevant unit of analysis is no longer the model alone but the workflow: a dynamic arrangement of agents, interfaces, data sources, model endpoints, evaluators, and escalation paths. Workflow optimization therefore becomes a problem of coordination under uncertainty. A routing agent may choose a smaller model for a low-risk task and a larger model for a high-stakes decision; a retrieval

agent may select evidence sources; a verification agent may challenge unsupported claims; and a human operator may intervene when confidence falls below a governance threshold. Such coordination is difficult to hard-code because workloads, costs, model capabilities, and policy requirements change over time. Adaptive multi-agent reinforcement learning offers one response by allowing agents to learn coordination policies from feedback rather than relying solely on static orchestration logic [1], [2].

The promise of adaptive multi-agent reinforcement learning in large language model systems is substantial. It can optimize latency, cost, accuracy, and resource allocation across heterogeneous agents. It can learn when to parallelize tasks, when to invoke expensive reasoning, when to retrieve external evidence, and when to defer to human judgment. It can also support resilience by redistributing work when a model endpoint degrades or a tool becomes unavailable. However, the same adaptivity that improves performance can produce opacity, reward hacking, unsafe exploration, and unstable coordination. The central question is not whether reinforcement learning can be applied to agentic workflows, but under what architectural, institutional, and normative conditions its use is defensible. This paper develops a system-level account of adaptive multi-agent reinforcement learning for autonomous workflow optimization in large language model systems. It emphasizes structural trade-offs, governance, infrastructure, sustainability, robustness, fairness, and policy implications rather than narrow algorithmic performance.

2. Background and Related Work

Reinforcement learning provides a formal tradition for sequential decision-making under uncertainty, where an agent learns behavior from interactions and reward signals [1]. Multi-agent reinforcement learning extends this tradition to settings with multiple interacting learners, introducing challenges such as non-stationarity, credit assignment, coordination, and equilibrium selection [2]. In parallel, transformer architectures and large-scale pretraining have made language models general-purpose components for reasoning, generation, and tool use [3], [4]. Foundation model research has shown that such models can support many downstream tasks, but also that they concentrate capability, dependency, and risk in a small number of artifacts and providers [5]. Machine learning systems research adds a further lesson: production systems accumulate hidden technical debt through entanglement, undeclared consumers, configuration drift, and feedback loops [6]. Software engineering studies of machine learning similarly show that deployment, monitoring, and data management often dominate lifecycle complexity [7].

Recent agentic systems research has begun to connect these strands by composing language models into multi-agent workflows. Reasoning-and-acting approaches interleave thought with tool use, enabling models to gather information and act in external environments [9]. Tool-learning methods allow models to select and invoke external functions, expanding the boundary between language generation and computation [10]. Multi-agent conversation frameworks coordinate specialized roles, such as planner, critic, and executor, to solve complex tasks [11]. Meta-level programming frameworks formalize role assignment and collaborative software generation [12]. Generative agent simulations demonstrate that language-model agents can exhibit memory, planning, and social behavior over extended interactions [13]. Chain-of-thought prompting and related techniques improve complex reasoning by making intermediate steps explicit [14], while verbal reinforcement methods allow agents to reflect on failures and revise behavior without updating model weights [15]. Recent work on scaling agentic models for full-scenario applications further indicates that

agentic capability is becoming a system-level scaling target rather than a single-model property [8].

These developments suggest that workflow optimization is shifting from static orchestration graphs to learned coordination policies. Static graphs are interpretable and auditable, but brittle. Learned policies are adaptive, but harder to certify. The design space therefore involves a deliberate mixture: some control points should remain declarative and auditable, while others can be learned within bounded envelopes. This paper treats adaptive multi-agent reinforcement learning as a governance-sensitive control layer rather than a replacement for system architecture. The control layer observes workflow state, allocates tasks, selects models and tools, manages retries and escalation, and learns from outcome signals. Its legitimacy depends on how those signals are defined, who can inspect them, and how failures are contained.

3. Architectural Foundations of Adaptive Multi-Agent LLM Workflows

An adaptive workflow architecture for large language model systems typically contains several layers. The first is the interface and intake layer, which classifies requests, identifies risk, and establishes user intent, constraints, and required service levels. The second is the agent society, containing specialized agents for planning, retrieval, synthesis, verification, tool use, and safety review. The third is the resource layer, which includes model endpoints, vector stores, databases, external application programming interfaces, compute pools, and human review queues. The fourth is the control layer, where reinforcement learning policies decide routing, sequencing, parallelism, retry behavior, and escalation. The fifth is the governance layer, which enforces policy, audit, privacy, security, and fairness constraints. These layers are not merely technical; they encode institutional priorities about acceptable risk, cost, and delay.

The architecture must distinguish between learnable and non-learnable decisions. Routing a summarization request between two comparable models may be safely learned if evaluation gates remain in place. Deciding whether to disclose a legally privileged document, deny a medical claim, or override a safety constraint should not be delegated to an unconstrained learned policy. The boundary between these categories is a governance decision, not an optimization parameter. A credible architecture therefore exposes contracts at each boundary: what the control layer may change, what it must preserve, what evidence it must produce, and what triggers human review. Without such contracts, adaptive optimization can silently erode accountability by shifting decisions across organizational boundaries where no single actor has full visibility [6], [7].

The agent society itself raises coordination problems. Agents may have different objectives, information, and failure modes. A retrieval agent may maximize evidence coverage, while a synthesis agent may optimize fluency, and a verification agent may minimize false confidence. If these objectives are combined into a single reward, the system may trade away safety for speed or cost. If they are kept separate, the control layer must negotiate conflicts. Multi-agent reinforcement learning provides tools for such negotiation, including centralized training with decentralized execution and actor-critic coordination [2]. However, language-model agents introduce a distinctive challenge: their actions are often semantic, their state spaces are open-ended, and their failure modes include plausible fabrication rather than simple error. The architecture must therefore treat verification as a first-class agent function rather than an afterthought.

4. Adaptive Multi-Agent Reinforcement Learning as a Control Layer

As a control layer, adaptive multi-agent reinforcement learning observes telemetry from the workflow and adjusts coordination behavior. Observations may include task type, risk classification, queue depth, model latency, token cost, retrieval quality, verification outcomes, user feedback, and policy flags. Actions may include selecting a model, invoking a tool, spawning a verifier, requesting clarification, compressing context, retrying with a different strategy, or escalating to a human. Rewards may combine task success, user satisfaction, latency, cost, safety violations, fairness metrics, and carbon intensity. The design of this reward is not a neutral technical choice. It determines which outcomes the system will pursue and which it will neglect.

A purely centralized controller can optimize globally, but it may become a bottleneck and a single point of failure. A fully decentralized controller can scale and tolerate local failures, but it may produce incoherent behavior and conflicting decisions. Hybrid designs are likely. A centralized policy may set high-level objectives and constraints, while local agents adapt within those constraints. For example, a central controller may specify that high-risk tasks require two independent verifiers and a human sign-off, while local agents learn how to schedule retrieval and drafting to meet latency targets. This layered approach preserves governance at the top and adaptivity at the edges. It also aligns with safety-critical control practice, where learned components operate inside verified envelopes [16], [23].

The learning process must address non-stationarity. When multiple agents learn simultaneously, each agent's environment changes as others adapt. Model providers may update endpoints, retrieval corpora may drift, user behavior may shift, and policy rules may change. A workflow policy that performs well in one month may fail in the next. Continuous evaluation, shadow deployment, and canary release are therefore necessary. Reinforcement learning should not be treated as a one-time training exercise but as an operational capability with monitoring, rollback, and periodic revalidation. This is especially important when rewards depend on human preferences, because preference models can encode biases and can be gamed by agents that learn to produce outputs that appear acceptable rather than those that are substantively correct [23].

5. Optimization Objectives and Structural Trade-offs

Workflow optimization in large language model systems involves several competing objectives. Accuracy and task success are usually primary, but they interact with latency, cost, privacy, fairness, robustness, and sustainability. A policy that always routes to the largest model may improve accuracy while increasing cost, energy use, and response time. A policy that aggressively compresses context may reduce cost but degrade reasoning and evidence use. A policy that parallelizes verification may improve reliability but multiply compute. These trade-offs are structural, not incidental. They arise from the architecture of model serving, data movement, human review, and governance.

Fairness adds another dimension. Workflows may allocate expensive verification or high-quality models unevenly across user groups, languages, or domains. If the reward function favors aggregate satisfaction, minority or low-resource cases may be underserved. Mitchell and colleagues argue for model cards and transparent reporting as mechanisms for documenting intended use, performance, and limitations [21]. In an adaptive workflow, such reporting must extend beyond individual models to the coordination policy itself. Which groups receive faster service? Which requests trigger human review? Which languages are

routed to weaker models? These questions require disaggregated evaluation. Fairness cannot be guaranteed by the mere presence of multiple agents; it must be measured, governed, and subject to correction.

Sustainability is similarly entangled with optimization. Strubell and colleagues show that training and deploying large natural language processing models can have significant energy and policy implications [18]. Patterson and colleagues analyze carbon emissions from large neural network training and highlight the importance of data center location, hardware efficiency, and energy mix [19]. In adaptive workflows, the control layer can reduce waste by routing simple tasks to smaller models, caching results, avoiding redundant retrieval, and scheduling flexible workloads during low-carbon periods. However, it can also increase total consumption through speculative execution, multi-agent debate, and redundant verification. The net effect depends on system-level accounting. Sustainability therefore belongs in the reward design and the governance dashboard, not in a separate reporting silo.

6. Governance, Fairness, and Policy Implications

Governance of adaptive multi-agent workflows must address accountability, transparency, contestability, and policy compliance. Russell argues that advanced artificial intelligence systems should be designed with uncertainty about human preferences and with mechanisms that preserve human control [16]. Rahwan proposes society-in-the-loop governance, where algorithmic systems are aligned with social contracts through ongoing public and institutional participation [17]. In large language model workflows, society-in-the-loop means that reward functions, escalation rules, and audit rights are not solely determined by engineering teams or vendors. They must reflect legal requirements, professional norms, and affected community interests. Dafoe's research agenda on artificial intelligence governance emphasizes international coordination, institutional design, and the need to manage strategic competition [22]. These concerns apply directly to workflow infrastructure, because optimized agent societies may become critical operational layers in finance, healthcare, education, and public administration.

Transparency in this context cannot mean exposing every model weight or internal activation. It more plausibly means documenting the workflow's decision rights, reward components, evaluation results, failure rates, and appeal mechanisms. Model cards provide a precedent for structured documentation [21]. Workflow cards could extend that practice by reporting agent roles, data flows, human oversight points, known failure modes, and fairness outcomes. Auditability also requires traceability. If a workflow denies a service, alters a medical recommendation, or escalates a fraud flag, the institution must be able to reconstruct which agents acted, what information they used, which policy constraints applied, and whether the learned controller followed its authorized envelope. Without such traceability, adaptive optimization becomes a source of institutional risk rather than a tool for accountable improvement.

Policy implications extend beyond organizational governance. Competition policy may need to consider dependence on a small number of model and cloud providers. Labor policy may need to address changes in human review roles, which shift from direct task execution to exception handling and oversight. Privacy law may constrain the retention of interaction traces used for reinforcement learning. Sectoral regulation may require validated performance before deployment in healthcare or finance. The challenge is not to prohibit adaptive multi-agent reinforcement learning, but to embed it in a regulatory and institutional architecture that

can detect harm, assign responsibility, and require remediation. This is a socio-technical problem, not a purely technical one.

7. Deployment, Infrastructure, and Sustainability

Deployment of adaptive multi-agent reinforcement learning requires infrastructure that supports observation, experimentation, and control. Observability must cover model latency, token usage, tool success, retrieval relevance, verifier agreement, human override rates, and policy violations. Experimentation must allow shadow runs, A/B tests, and counterfactual evaluation without exposing users to unacceptable risk. Control must support rollback, feature flags, rate limits, and manual freezes. These are familiar requirements in machine learning operations, but they become more complex when multiple learned agents interact. A change in one agent’s policy can alter the training distribution for another, producing cascading effects that are difficult to diagnose [6], [7].

Infrastructure choices also affect sustainability. Serverless inference, regional model routing, quantization, distillation, caching, and batch scheduling can reduce energy and cost. However, multi-agent workflows may increase total calls because agents deliberate, verify, and retry. The control layer should therefore include explicit budgets for computation, latency, and carbon. These budgets can be treated as constraints rather than soft penalties. For example, a workflow may be allowed to spawn additional verifiers only when risk classification exceeds a threshold and the carbon budget for that service tier remains available. Such constraints make trade-offs visible and governable [18], [19]. They also prevent optimization from externalizing costs onto users, workers, or the environment.

Operationally, human oversight must be designed for the exception, not the routine. If human reviewers are asked to inspect every agent decision, the system loses the efficiency gains of autonomy. If they inspect too few, failures may go undetected. A risk-based approach allocates human attention to high-stakes, ambiguous, or novel cases. The learned controller can help by identifying uncertainty, disagreement among agents, or distribution shift, but it should not be the sole judge of its own need for oversight. Independent monitoring, worker feedback, and user appeals are necessary. This is particularly important because language-model agents can produce confident but incorrect outputs, and multi-agent agreement can create false consensus rather than genuine reliability.

8. Robustness, Evaluation, and Failure Modes

Robustness in adaptive workflows requires resilience to model updates, tool outages, adversarial inputs, data poisoning, prompt injection, and distribution shift. A multi-agent system may fail through single-agent error, coordination failure, or emergent pathology. For example, a planner may over-delegate to a retrieval agent that cannot access the needed source; a verifier may approve a fabricated citation because it shares the same underlying model bias; or a routing policy may learn to avoid difficult cases by escalating them unnecessarily. Evaluation must therefore cover both component performance and system behavior. It should include red-team testing, stress testing, adversarial simulation, and long-horizon scenario analysis [5], [9].

Reinforcement learning adds specific failure modes. Reward hacking occurs when agents optimize a proxy in ways that violate the intended objective. Exploration can produce unsafe actions if not constrained. Credit assignment can be misleading when success depends on many agents and delayed outcomes. Non-stationarity can make learned policies obsolete. Verbal reinforcement and reflection can help agents revise behavior, but they can also

reinforce plausible narratives rather than ground truth [15]. Safety-critical control research suggests that learned components should operate within verified envelopes and that human control should be preserved [16]. In workflow terms, this means that some decisions must remain outside the learned policy, and that the policy must be interruptible, inspectable, and reversible.

Evaluation should also be longitudinal. A workflow may perform well in pilot studies but degrade as users adapt, as adversaries probe, or as the underlying model changes. Continuous monitoring should track not only average performance but also tail risks, subgroup performance, escalation rates, and human workload. Metrics should be tied to governance thresholds. If a threshold is breached, the system should degrade gracefully to a simpler, more auditable workflow. Graceful degradation is a design principle: when the learned controller is uncertain or out of distribution, the system should fall back to deterministic rules and human review. This reduces the chance that optimization produces silent, systemic harm.

9. Case Illustrations and Cross-Domain Comparisons

In customer service, an adaptive multi-agent workflow might route routine inquiries to a small model, complex cases to a larger model, and disputed cases to human agents. The control layer can learn from resolution time, customer satisfaction, cost, and compliance outcomes. Multi-agent conversation frameworks show how specialized roles can collaborate on such tasks [11]. However, fairness concerns arise if certain customer groups receive slower or lower-quality service. Governance requires disaggregated reporting and appeal rights. In software engineering, meta-programming frameworks assign planner, coder, and tester roles [12]. The control layer can optimize iteration speed and defect detection, but it must prevent agents from bypassing security review or generating vulnerable code. Here, the workflow resembles a software development pipeline, and the same principles of code review, testing, and audit apply.

In scientific research assistance, generative agents can simulate collaboration, literature review, and hypothesis generation [13]. Adaptive reinforcement learning could allocate effort to promising lines of inquiry, but it could also amplify confirmation bias or produce fabricated citations. Verification agents and human domain experts become essential. In public administration, society-in-the-loop governance is especially important because decisions affect rights and benefits [17]. An adaptive workflow might triage applications, but it must not learn to optimize speed by denying difficult cases. Cross-domain comparison with supply chain orchestration is instructive. Supply chains use multi-agent optimization for inventory, routing, and risk management, but they also rely on contracts, audits, and regulatory compliance. Large language model workflows need analogous contracts: service-level agreements, data-sharing rules, liability assignment, and independent inspection.

Cloud autoscaling offers another comparison. Autoscaling systems learn to allocate resources under demand uncertainty. They are effective when objectives are clear, workloads are homogeneous, and failures are contained. Large language model workflows are more heterogeneous and semantically open. A request may be easy or ambiguous, benign or adversarial, routine or novel. Therefore, autoscaling logic cannot be transferred without adaptation. Safety-critical control offers a third comparison. In aviation and industrial control, learned components are often confined to advisory roles or bounded envelopes. The lesson is that adaptivity and certification can coexist, but only through explicit separation of concerns, redundant safety layers, and human authority [16], [23].

10. Future Directions

Future research should develop workflow-level benchmarks that evaluate coordination, not just final answers. Such benchmarks should include risk classification, escalation decisions, cost, latency, fairness, and sustainability. They should test robustness under model updates, tool failures, and adversarial inputs. They should also examine governance mechanisms, such as audit trails, appeal processes, and human oversight. Recent agentic scaling work suggests that full-scenario applications will require models and controllers that generalize across domains [8]. That ambition makes governance more urgent, not less. The more capable the agent society, the more important it is to specify what it must not do and who can intervene.

Another direction is the development of constrained reinforcement learning for workflow control. Rather than combining all objectives into a single scalar reward, researchers could design multi-objective policies with explicit constraints, Pareto frontiers, and human-selected trade-offs. Such policies could expose the cost of fairness, safety, and sustainability rather than hiding them in a blended score. This would support institutional deliberation. It would also make it easier to audit whether a change in policy shifts burdens onto particular groups. Governance research emphasizes institutional design and international coordination [22], and workflow optimization should be connected to those debates rather than treated as a purely private engineering matter.

Finally, future systems should explore modularity and portability. If workflows depend on a single provider's models, tools, and orchestration services, they may become fragile and politically constrained. Modular interfaces, open standards, and interchangeable agents could support competition, resilience, and public oversight. However, modularity also creates integration and verification challenges. The field needs architectures that balance interoperability with accountability. Human preference learning and reinforcement learning from feedback offer ways to align agents with human values, but they must be complemented by law, policy, and professional ethics [23], [4]. The goal is not to remove humans from the loop but to place them where judgment, legitimacy, and accountability matter most.

11. Conclusion

Adaptive multi-agent reinforcement learning can improve autonomous workflow optimization in large language model systems by learning how to allocate tasks, models, tools, verification, and human review under changing conditions. It can reduce latency and cost, improve resilience, and enable more responsive coordination than static orchestration. Yet these benefits are conditional. Without explicit contracts, evaluation gates, audit trails, and governance constraints, adaptive optimization can produce opacity, reward hacking, unfair service allocation, unsustainable compute consumption, and accountability gaps. The most credible path forward is governed adaptivity: reinforcement learning optimizes coordination within externally specified boundaries, while safety, fairness, privacy, sustainability, and contestability remain institutional obligations. This requires interdisciplinary work across reinforcement learning, systems engineering, human-computer interaction, law, policy, and ethics. It also requires a shift in evaluation from isolated model performance to workflow-level outcomes. The future of agentic large language model systems will not be determined solely by larger models or better prompts. It will be determined by whether institutions can design, deploy, and oversee adaptive workflows that are effective, robust, fair, and legitimate.

References

1. Sutton, R. S., & Barto, A. G. (2018). Reinforcement learning: An introduction (2nd ed.). MIT Press.
2. Lowe, R., Wu, Y., Tamar, A., Harb, J., Abbeel, P., & Mordatch, I. (2017). Multi-agent actor-critic for mixed cooperative-competitive environments. *Advances in Neural Information Processing Systems*, 30, 6379–6390.
3. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 5998–6008.
4. Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., ... Amodei, D. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33, 1877–1901.
5. Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N., Chen, A., Creel, K., Davis, J. Q., Demszky, D., ... Liang, P. (2021). On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*.
6. Sculley, D., Holt, G., Golovin, D., Davydov, E., Phillips, T., Ebner, D., Chaudhary, V., Young, M., Crespo, J.-F., & Dennison, D. (2015). Hidden technical debt in machine learning systems. *Advances in Neural Information Processing Systems*, 28, 2503–2511.
7. Amershi, S., Begel, A., Bird, C., DeLine, R., Gall, H., Kamar, E., Nagappan, N., Nushi, B., & Zimmermann, T. (2019). Software engineering for machine learning: A case study. In *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP)* (pp. 291–300). IEEE.
8. Jiang, Hao, et al. "AgentOmnia: Scaling Agentic Models for Full-Scenario Applications." *arXiv preprint arXiv:2607.23124* (2026).
9. Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., & Cao, Y. (2023). ReAct: Synergizing reasoning and acting in language models. *International Conference on Learning Representations*.
10. Schick, T., Dwivedi-Yu, J., Dessì, R., Raileanu, R., Lomeli, M., Hambro, E., Zettlemoyer, L., Cancedda, N., & Scialom, T. (2023). Toolformer: Language models can teach themselves to use tools. *Advances in Neural Information Processing Systems*, 36.
11. Wu, Q., Bansal, G., Zhang, J., Wu, Y., Li, B., Zhu, E., Jiang, L., Zhang, X., Zhang, S., Awadallah, A., White, R. W., Burger, D., & Wang, C. (2023). AutoGen: Enabling next-gen LLM applications via multi-agent conversation. *arXiv preprint arXiv:2308.08155*.
12. Hong, S., Zhuge, M., Chen, J., Zheng, X., Cheng, Y., Zhang, C., Wang, J., Wang, Z., Yau, S. K. S., Lin, Z., Zhou, L., Ran, C., Xiao, L., Wu, C., & Schmidhuber, J. (2023). MetaGPT: Meta programming for a multi-agent collaborative framework. *arXiv preprint arXiv:2308.00352*.
13. Park, J. S., O'Brien, J. C., Cai, C. J., Morris, M. R., Liang, P., & Bernstein, M. S. (2023). Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (pp. 1–22). ACM.

14. Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q., & Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35, 24824–24837.
15. Shinn, N., Cassano, F., Gopinath, A., Narasimhan, K., & Yao, S. (2023). Reflexion: Language agents with verbal reinforcement learning. *Advances in Neural Information Processing Systems*, 36.
16. Russell, S. (2019). *Human compatible: Artificial intelligence and the problem of control*. Viking.
17. Rahwan, I. (2018). Society-in-the-loop: Programming the algorithmic social contract. *Ethics and Information Technology*, 20(1), 5–14.
18. Strubell, E., Ganesh, A., & McCallum, A. (2019). Energy and policy considerations for deep learning in NLP. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics* (pp. 3645–3650). Association for Computational Linguistics.
19. Patterson, D., Gonzalez, J., Le, Q., Liang, C., Munguia, L.-M., Rothchild, D., So, D., Texier, M., & Dean, J. (2021). Carbon emissions and large neural network training. *arXiv preprint arXiv:2104.10350*.
20. Bender, E. M., Gebru, T., McMillan-Major, A., & Shmitchell, S. (2021). On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency* (pp. 610–623). ACM.
21. Mitchell, M., Wu, S., Zaldivar, A., Barnes, P., Vasserman, L., Hutchinson, B., Spitzer, E., Raji, I. D., & Gebru, T. (2019). Model cards for model reporting. In *Proceedings of the Conference on Fairness, Accountability, and Transparency* (pp. 220–229). ACM.
22. Dafoe, A. (2018). *AI governance: A research agenda*. Governance of AI Program, Future of Humanity Institute, University of Oxford.
23. Christiano, P. F., Leike, J., Brown, T., Martic, M., Legg, S., & Amodei, D. (2017). Deep reinforcement learning from human preferences. *Advances in Neural Information Processing Systems*, 30, 4299–4307.