Hierarchical World-Model Reinforcement Learning for Long-Horizon Reasoning in Large Language Model Agents

Roy West
Department of Computer Science and Engineering, University of Nevada, Reno, Reno, NV, USA.
rwest@unr.edu

Abstract

Large language model agents have demonstrated remarkable capabilities in language understanding and generation, yet they remain fundamentally limited in tasks requiring extended sequential reasoning and planning over long horizons. This paper proposes a framework that integrates hierarchical reinforcement learning with learned world models to address these limitations. By coupling a high-level abstract planner with a low-level world-model simulator, the agent can decompose complex long-horizon tasks into manageable subgoals, evaluate hypothetical action sequences in a learned internal model, and refine its reasoning through recursive credit assignment. The paper examines architectural trade-offs between abstraction granularity and model fidelity, discusses training stability and sample efficiency challenges, and explores the socio-technical implications of deploying such systems in critical infrastructure. Key considerations include robustness against distributional shift, fairness in reward design, computational sustainability, and the need for transparent governance mechanisms. The hierarchical world-model approach offers a principled path toward more deliberative and scalable LLM agents, but raises important questions about safety, accountability, and alignment in autonomous decision-making systems. The paper concludes with recommendations for future research and policy development.

Keywords

hierarchical reinforcement learning, world models, large language models, long-horizon reasoning, agent architecture, AI governance, fairness, robustness.

1. Introduction

Large language models have become central to modern artificial intelligence systems, powering applications ranging from conversational agents to automated code generation and scientific analysis. Despite their fluency, these models exhibit a fundamental weakness when tasked with problems that require sustained reasoning over many time steps, such as multi-step planning, goal-directed dialogue, or complex problem solving across extended contexts. The underlying autoregressive nature of typical LLMs encourages local coherence but fails to enforce global plan consistency, leading to hallucinations, forgetfulness, and suboptimal decision chains.

Reinforcement learning provides a natural framework for training agents to maximize cumulative reward through sequences of actions, but applying it to LLM-based agents introduces unique challenges. The action space is vast and symbolic, the reward signals are sparse, and the horizon over which credit must be assigned is often extremely long. World models, which learn an internal representation of the environment dynamics, offer a

mechanism for simulated planning and off-policy evaluation. When combined with hierarchical reinforcement learning, world models enable the agent to operate at multiple levels of temporal abstraction, thereby reducing the effective horizon and improving sample efficiency. This paper argues that a hierarchical world-model reinforcement learning approach is particularly well suited for extending the reasoning capabilities of LLM agents in long-horizon settings.

The paper proceeds as follows. Section 2 reviews relevant prior work in reinforcement learning, world models, and LLM reasoning. Section 3 describes the proposed hierarchical architecture, emphasizing the roles of the abstract planner and the world-model simulator. Section 4 addresses training and deployment considerations, including sample efficiency, computational cost, and scalability. Section 5 examines robustness and fairness issues inherent in such systems. Section 6 discusses governance and policy implications, focusing on accountability and alignment. The conclusion summarizes contributions and outlines future directions.

2. Background and Related Work

Reinforcement learning traditionally addresses sequential decision making through trial-and-error interaction with an environment. The seminal work of Sutton and Barto formalized the Markov decision process framework and introduced temporal-difference learning [1]. However, flat RL algorithms scale poorly to long horizons because the number of possible trajectories grows exponentially with length. Weekends and high-level abstraction were proposed by Dayan and Hinton through the options framework, which allows an agent to choose among temporally extended actions [2]. Dietterich later introduced hierarchical abstract machines and the MAXQ decomposition, providing a formal basis for hierarchical reinforcement learning [3].

World models, as articulated by LeCun, propose that intelligent agents should learn a predictive model of the environment to enable planning and reasoning in a compact latent space [4]. Ha and Schmidhuber demonstrated that a neural network can learn a world model for a latent-space controller, achieving impressive results in a simulated environment [5]. More recently, Hafner et al. developed Dreamer, a model-based RL agent that learns a latent dynamics model and uses it for imagined rollouts to learn effective policies [6]. These advances have shown that world models can dramatically improve sample efficiency and enable planning in high-dimensional observation spaces.

Large language models have been augmented with reasoning techniques such as chain-of-thought prompting, which encourages step-by-step thinking, and tree-of-thought search, which explores multiple reasoning paths [7][8]. The work of Dou et al. proposes a plan-then-action framework that provides high-level planning guidance for LLM reasoning, integrating symbolic planning with neural generation [8]. This approach aligns closely with hierarchical world-model concepts, as it separates high-level plan generation from low-level action execution. However, existing methods often lack a learned world model that can simulate the consequences of each step, limiting the agent's ability to backpropagate credit over extended sequences.

The intersection of hierarchical RL and world models has been explored in robotics and game playing, but its application to LLM-based agents remains nascent. Researchers have begun to use LLMs as components in hierarchical planners, where the language model generates subgoals or plans that are then executed by lower-level policies [9][10]. These systems benefit

from the vast prior knowledge encoded in pretrained language models but still struggle with grounding and long-term consistency.

3. Hierarchical World-Model Architecture

The proposed architecture consists of three interconnected modules: a high-level abstract planner, a learned world model, and a low-level action executor. The high-level planner operates over a temporally abstract action space, selecting subgoals or macro-actions that guide the agent toward the overall objective. These subgoals are defined in a symbolic or continuous embedding space that captures essential state features. The world model is a deep neural network trained to predict the next latent state and reward given the current latent state and a candidate action. It functions as a differentiable simulator that permits the planner to evaluate hypothetical sequences without interacting with the real environment.

The low-level executor translates abstract subgoals into primitive actions. In the context of an LLM agent, primitive actions might include generating a single token, calling an API, or retrieving a piece of information. The executor is trained to maximize subgoal completion under the guidance of the world model's predictions. The hierarchical structure naturally decomposes the long-horizon problem: the planner works at a coarse timescale, while the executor operates at a fine timescale.

One critical design trade-off is the choice of abstraction granularity. If the high-level actions are too coarse, the planner may miss important intermediate states that affect the overall reward. If they are too fine, the benefits of hierarchical decomposition diminish, and the planner effectively solves the original flat problem. The optimal granularity depends on the structure of the task domain. In domains with strong temporal abstractions, such as instruction-following or sequential reasoning, natural subgoal boundaries emerge from the task semantics. For example, in a multi-step math problem, each algebraic transformation constitutes a natural subgoal. In a dialogue system, each turn can be considered a macro-action.

The world model must balance accuracy and computational efficiency. A high-fidelity model that captures every detail of the environment dynamics is expensive to train and use for planning. Conversely, an overly abstract model may produce erroneous predictions, leading to suboptimal plans. Regularization techniques such as variational inference can be employed to learn a compact latent representation that preserves task-relevant information while discarding noise [11]. The world model can be pre-trained using unsupervised exploration data from the target environment and then fine-tuned with task-specific rewards.

The integration of a world model with hierarchical planning enables credit assignment across multiple time scales. The planner can evaluate the long-term consequences of a high-level action by simulating its future trajectory using the world model and accumulating the predicted rewards. This recursive evaluation approximates dynamic programming without requiring explicit models of the entire state space. Moreover, the planner can leverage the world model to perform counterfactual reasoning, asking "what if" questions to explore alternative strategies.

4. Training and Deployment Considerations

Training a hierarchical world-model system for LLM agents involves multiple stages. First, the world model is learned from interactions with the environment. This stage can be performed offline using logged data or online through exploration. The exploration policy

must balance coverage of the state space with task relevance. A common strategy is to use a random policy augmented with intrinsic motivation signals to encourage novel state visits [12]. Once the world model reaches satisfactory predictive accuracy, the high-level planner is trained using model-based reinforcement learning algorithms, such as Monte Carlo tree search or policy gradient methods operating on simulated trajectories.

Sample efficiency is a primary advantage of this approach. Because the world model generates many simulated trajectories without requiring real environment interactions, the agent can learn effective policies with orders of magnitude fewer environment steps. This is particularly valuable in real-world domains where each interaction is costly or slow, such as robotics, healthcare, or autonomous driving. However, the world model itself may suffer from distributional shift when the agent's policy begins to explore states that were underrepresented in the training data. Techniques such as ensemble models or uncertainty-aware planning can mitigate this risk by discounting predictions with high uncertainty [13].

Deploying hierarchical world-model agents in production systems introduces additional challenges. Computational overhead must be managed, as planning with a world model can be expensive. One approach is to interleave planning and execution, where the planner re-evaluates only when uncertainty exceeds a threshold. Another is to compress the world model into a lightweight surrogate for fast inference. In edge deployment scenarios, the model may need to run on limited hardware, necessitating quantization or distillation.

Long-term sustainability is a growing concern. Training large world models and LLMs consumes substantial energy and computational resources. The hierarchical architecture can help by reducing the amount of exploration needed, but the upfront cost of training the world model remains significant. Researchers have begun to develop more efficient training methods, such as using pretrained embeddings or transfer learning from similar domains [14]. Moreover, the environmental cost of operating large-scale agents must be weighed against the benefits they provide.

5. Robustness and Fairness

Robustness in hierarchical world-model agents is a multi-faceted issue. The world model may produce inaccurate predictions when the environment changes (non-stationarity) or when the agent encounters out-of-distribution scenarios. A robust agent should detect such situations and fall back to conservative behaviors or request human intervention. Hierarchical architectures can improve robustness by allowing the high-level planner to re-plan when the low-level executor fails to achieve a subgoal. However, if the planner's assumptions about the world model are violated, the entire plan may become invalid.

Adversarial robustness is particularly relevant in safety-critical applications. An attacker could manipulate the environment to cause the world model to mispredict, leading the agent to take harmful actions. Defenses include training the world model with adversarial examples, using ensemble methods, and incorporating human-in-the-loop verification for high-stakes decisions.

Fairness considerations arise from the design of the reward function and the representation of states. If the reward function encodes biases present in historical data, the agent may learn to reinforce inequitable outcomes. For example, a credit-scoring agent trained on biased data might deny loans to certain demographic groups. Hierarchical reward design offers an opportunity to specify high-level fairness constraints that are then decomposed into subgoal

rewards. The world model can be used to simulate the long-term distributional impact of different policies, enabling fairness audits before deployment [15].

Another dimension of fairness is access to LLM-based agents. The computational and data requirements of hierarchical world-model systems may concentrate power in organizations with large resources, exacerbating digital divides. Policy interventions, such as open-source model releases and public compute subsidies, could promote more equitable access.

6. Governance and Policy Implications

The deployment of autonomous LLM agents that reason over long horizons raises profound governance questions. Who is responsible when an agent makes a harmful decision that was the result of a plan devised through hierarchical reasoning? Traditional notions of intent and causality become blurred when the agent's actions are mediated by learned world models and latent representations. Establishing clear lines of accountability requires that the system's planning process be auditable and that decisions can be traced back to specific reward signals or state representations.

Regulatory frameworks for AI are evolving, but most current regulations focus on simpler systems. The European Union's AI Act classifies high-risk AI systems and mandates transparency and human oversight. Hierarchical world-model agents, especially those used in critical infrastructure (e.g., power grid management, healthcare), would likely fall into the high-risk category. They would need to demonstrate robustness, fairness, and the ability to be overridden by human operators. The requirement for explainability poses a challenge because the internal decisions of a world model may not be easily interpretable. Researchers are developing methods to extract explanations from latent states and policy gradients, but these remain nascent.

International cooperation is essential to avoid a race to the bottom where safety standards are compromised for speed of deployment. The development of shared benchmarks for long-horizon reasoning and hierarchical agents can facilitate regulatory harmonization. Additionally, the use of world models could itself serve as a governance tool: by simulating the outcomes of different policies, regulators can anticipate unintended consequences before real-world deployment.

7. Conclusion

Hierarchical world-model reinforcement learning offers a promising architecture for enabling large language model agents to perform long-horizon reasoning. By decomposing tasks into temporally abstract subgoals and using a learned internal simulator for planning, these agents can overcome the limitations of flat autoregressive generation. The paper has discussed key architectural trade-offs, training challenges, and deployment considerations. It has also highlighted the importance of robustness, fairness, and governance in ensuring that such systems are safe and equitable.

Future work should focus on developing more sample-efficient and uncertainty-aware world models, integrating human feedback into the hierarchical planning loop, and creating standardized evaluation suites for long-horizon tasks. As LLM agents become more capable, the hierarchical world-model framework may become an essential component of trustworthy autonomous systems.

References

1. Sutton, R. S., & Barto, A. G. (2018). Reinforcement learning: An introduction (2nd ed.). MIT Press.

2. Dayan, P., & Hinton, G. E. (1993). Feudal reinforcement learning. In Advances in Neural Information Processing Systems 5 (pp. 271–278). Morgan Kaufmann.

3. Dietterich, T. G. (2000). Hierarchical reinforcement learning with the MAXQ value function decomposition. Journal of Artificial Intelligence Research, 13, 227–303.

4. LeCun, Y. (2022). A path towards autonomous machine intelligence. Open Review. https://openreview.net/forum?id=BZ5a1r-kVsf

5. Ha, D., & Schmidhuber, J. (2018). World models. arXiv preprint arXiv:1803.10122.

6. Hafner, D., Lillicrap, T., Ba, J., & Norouzi, M. (2020). Dream to control: Learning behaviors by latent imagination. In International Conference on Learning Representations.

7. Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., ... & Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. In Advances in Neural Information Processing Systems 35.

8. Dou, Z., Zhao, Q., Wan, Z., Zhang, D., Wang, W., Raiyan, T., ... & Biswas, S. (2025). Plan Then Action: High-Level Planning Guidance Reinforcement Learning for LLM Reasoning. arXiv preprint arXiv:2510.01833.

9. Yao, S., Yu, D., Zhao, J., Shafran, I., Griffiths, T. L., Cao, Y., & Narasimhan, K. (2023). Tree of thoughts: Deliberate problem solving with large language models. arXiv preprint arXiv:2305.10601.

10. Ahn, M., Brohan, A., Brown, N., Chebotar, Y., Cortes, O., David, B., ... & Zhang, M. (2022). Do as I can, not as I say: Grounding language in robotic affordances. In Conference on Robot Learning (pp. 287–318). PMLR.

11. Hafner, D., Lillicrap, T., Norris, M., Ba, J., & Norouzi, M. (2021). Learning latent dynamics for planning from pixels. In International Conference on Machine Learning (pp. 3939–3949). PMLR.

12. Pathak, D., Agrawal, P., Efros, A. A., & Darrell, T. (2017). Curiosity-driven exploration by self-supervised prediction. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (pp. 16–17).

13. Kahn, G., Villaflor, A., Pong, V., Abbeel, P., & Levine, S. (2017). Uncertainty-aware reinforcement learning for collision avoidance. arXiv preprint arXiv:1702.01182.

14. 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).

15. 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).

16. Doshi-Velez, F., & Kim, B. (2017). Towards a rigorous science of interpretable machine learning. arXiv preprint arXiv:1702.08608.

17. Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., & Mané, D. (2016). Concrete problems in AI safety. arXiv preprint arXiv:1606.06565.

18. Russell, S., Dewey, D., & Tegmark, M. (2015). Research priorities for robust and beneficial artificial intelligence. AI Magazine, 36(4), 105–114.

19. Floridi, L., & Cowls, J. (2019). A unified framework of five principles for AI in society. Harvard Data Science Review, 1(1).