研究计划书

AgentInfraGraphWorld

面向长程 LLM Agent 的多图世界模型与执行结构共同演化学习

Multi-Graph World Modeling for Structural Co-evolution in Long-Horizon LLM Agent Infrastructure

学位	哲学博士（PhD）
研究方向	数据科学与工程
申请人	[申请人姓名]
邮箱	[邮箱地址]
入学学期	[入学学期]

导师

[导师姓名]
[导师职称 / 实验室]
计算学院
科学与工程学院

麦考瑞大学

2026 年 11 月 28 日

摘要

长程大语言模型（LLM）智能体以及建立在其上的多智能体系统正日益在基础设施规模上运行：它们共享 GPU、工具服务器、API 配额与缓存，其执行过程形成任务依赖、计算资源分配、工具调用与缓存复用等多个相互耦合的动态图。某一图层的局部结构变化可能跨图传播，并反馈至源图层，因此单独预测单个图既难以刻画执行系统的长期演化，静态运行时启发式也无法在动作执行前评估其影响。本研究拟提出 AgentInfraGraphWorld，一种面向长程智能体基础设施的多图世界模型。其核心是跨图结构转移机制，并由图层特定的图内演化与闭环反馈提供支撑。给定当前四图、演化历史与执行动作，模型直接预测未来四图及其跨图关系变化，不通过生成自然语言再解析为结构。多步联合推演支持基于预测的执行结构优化。实验不仅评估预测精度，还将检验一个科学假设：在匹配参数与计算预算的条件下，显式四图状态与跨图转移算子是否优于统一异构图世界模型？比较对象包括独立逐图预测、统一异构图世界模型、时序图神经网络与确定性运行时模型，评估维度覆盖多图结构预测、跨图传播、闭环反馈、长期推演、泛化能力与执行成本降低。

1 引言

随着 LLM 智能体处理任务的复杂程度不断提高，其运行过程逐渐形成由多个动态执行图组成的复杂系统。规划器生成与修改任务，调度器分配与迁移计算资源，执行器调用外部工具，缓存管理器存储、共享与驱逐中间结果。每个过程都具有独立的运行时语义，并在长程执行中持续改变图的拓扑。

当系统规模扩大时，运行时本身成为瓶颈。一次长程执行可能累积数千步，多智能体部署又会在共享 GPU、工具服务器、API 配额与缓存上复用多个执行实例。调度决策带来资源竞争；工具失败触发重试；重试导致缓存对象失效；缓存未命中迫使重新计算并改变任务图。模型能力的提升不会消除这些耦合效应，反而使执行系统更大、更动态。

四个图图层是该执行系统的核心，如表 1 所示。

表 1: 长程 LLM 智能体的四个动态执行图。

图层	结构语义
任务依赖图	任务的生成、依赖、分支与合并
计算资源图	计算资源的分配、共享、竞争与迁移
工具调用图	工具调用实例、执行顺序与数据依赖
缓存复用图	缓存的生成、共享、驱逐与复用关系

这些图相互耦合。任务并行化可能改变资源竞争与工具调用结构，进而改变缓存共享关系；缓存失效或资源不足也可能触发任务重新规划，从而改变任务图的后续拓扑。此类变化可以经过多个图层传播，并返回至源头，有时被放大，有时被抑制。

由于这些影响会在多轮执行中累积，单独预测某一个图的未来结构难以刻画整个执行系统的长期行为。一个有用的模型需要表示四个图的联合状态、联合状态之间的转移、局部结构变化在图层间的传播，以及返回源图层的反馈。

静态运行时启发式或单图预测不足以支持规划。动态图预测通常从观测结构学习 $P(G_{t+1} \mid G_{\leq t})$，它不以动作为条件，无法回答“若拆分任务、迁移作业或驱逐缓存对象会发生什么”。图世界模型学习 $P(\mathbf{G}_{t+1} \mid \mathbf{G}_t, a_t)$ 并可多步推演，因此能在动作执行前评估候选执行决策。二者的区别在于：预测一条已发生的轨迹，还是模拟一个动作的后果。

图世界模型为这一问题提供了起点。Graph World Model [6] 将图结构与动作纳入统一的

世界建模框架；WorldGraph [7] 将图演化视为世界动力学；近期研究进一步探索动态图世界建模 [9]、演化拓扑 [10] 以及带约束推演的结构动力学 [11]。这些方法建模单个图或单个结构系统。智能体世界模型 [12, 13] 则预测环境观测、任务进展或语言层面的环境行为。两类工作均未直接输出智能体四张执行图的未来结构，也未建模带回环反馈的跨图结构状态转移。

本计划的核心科学问题是：

如何构建多图世界模型，学习长程智能体多个动态执行图之间的图内演化、跨图结构传播与闭环反馈规律，并在给定执行动作的条件下预测未来四图结构？

为回答该问题，本计划提出 AgentInfraGraphWorld。该框架保持四个图的独立性，维护显式的跨图关系状态，并学习以跨图结构转移算子为核心、由图内演化与闭环反馈支撑的世界模型。给定执行动作，模型以结构化对象的形式直接输出未来四图及其关系变化。多步联合推演支持选择在保持任务正确性与质量的同时降低整体执行成本的执行结构变换。本计划的主要贡献如下：

- **跨图结构状态转移机制（核心贡献）：**学习一个图的变化如何传播至其他图并返回源图，直接输出未来图结构与关系变化，无需文本生成与解析。
- **多图世界模型架构：**在四个独立执行图上组合图内演化、跨图结构传播与闭环反馈。
- **基于预测的执行优化：**利用多步联合推演，在任务正确性约束下选择合法的结构变换。

2 研究现状

2.1 智能体执行基础设施

LLMCompiler [1] 将任务计划编译为函数调用的有向无环图，并行执行相互独立的调用。Parrot [2] 通过语义变量暴露 LLM 请求之间的依赖，并优化应用级服务。SAGA [3] 在 GPU 集群上以工作流为原子单位进行调度，ThunderAgent [4] 提供程序感知的智能体推理，GraphFlow [5] 以图的形式管理智能体工作流以提升服务效率。

这些系统表明，显式的执行结构能够降低延迟与成本。它们各自只处理四类运行时过程中的一部分——通常是任务依赖与工具调用，或调度与资源分配——并将相关结构视为运行时给定，而非需要学习与预测的对象。它们未提供四图联合演化的世界模型。

2.2 图世界模型

Graph World Model [6] 将图结构与动作纳入统一的世界建模框架。WorldGraph [7] 将动态图自身的演化视为世界动力学，研究节点级、边级与图级状态转移。近期综述 [8] 对图世界模型进行形式化与分类。DyG-WM [9] 从时间上下文中预测未来节点与关系表示；GDM [10] 处理随机、部分可观测环境中的演化拓扑；SD-GWM [11] 通过约束推演与校准建模结构动力学。

这些工作表明，图结构是世界建模的有效载体，拓扑也可以随时间变化。它们建模单个图或单个结构系统，未表示具有不同转移规则的多个执行图，也未学习一个图的结构编辑如何改变另一个图的转移分布并作为反馈返回。

2.3 智能体世界模型

智能体世界模型关注智能体与环境的交互。AEWM [12] 建模推理与动作如何影响未来任务进展，而非重构工具响应。Qwen-AgentWorld [13] 构建语言世界模型，用于模拟跨领域的智能体环境。这些模型作用于观测、任务进展或语言层面的状态。

它们未将智能体的运行时基础设施暴露为结构化多图状态，其输出也不是未来四张执行图及其跨图关系。通过文本预测执行结构还会引入不必要的生成与解析步骤；世界模型可以直接输出结构。

2.4 多图与跨图学习

多层网络分析 [14] 研究由多个关系层组成的系统，时序图网络 [15] 从带时间戳的事件中学习连续时间动力学。这些工作表明，多图层与动态拓扑可以被表示与学习。

仍未解决的是面向多个动态图的动作条件化联合结构预测器：它需要保持图层特定的转移规则，建模跨图传播，将影响闭环反馈至源图，并直接输出所有图的未来结构。现有方法尚未提供这一组合。

2.5 研究缺口与创新空间

一种自然的替代方案是将四张图展平为一张异构图，并在其上学习单一世界模型。这是一个强基线，而非稻草人：异构图可以表示所有节点与边类型，共享转移也可能利用分离图所忽略的跨类型规律。因此，独立图层是否带来收益是一个经验问题。我们将其表述为假设 H1：在匹配参数与计算预算的条件下，显式四图状态与跨图转移算子相比统一异构图世界模型，能够获得更低的长期预测误差与更好的执行成本降低。预期差异在于：统一模型将图层特定的转移规则压缩为同一种参数化转移，使任务拆分与缓存驱逐成为同一类结构事件；跨图关系退化为众多普通边中的一类，使传播与反馈难以归因和控制。实验设计要使 H1 被证伪的结果同样可见，而不是预设答案。

AgentInfraGraphWorld 保持四个图的独立性，维护显式的跨图关系状态，并将三个机制作为不同组件学习。这保留了图层身份，使传播与反馈可追踪，并允许解码器直接输出图层特定的结构编辑。因此，本研究的创新空间是多图结构状态转移的架构与学习机制，而非某一种神经网络基座。

3 研究目标与方法

3.1 多图执行系统表示

在时刻 t，智能体执行系统表示为

$$\mathbf{G}_t = \left(G_t^T, G_t^R, G_t^O, G_t^C, \mathcal{R}_t\right), \tag{1}$$

其中 G_t^T、G_t^R、G_t^O、G_t^C 分别表示任务图、资源图、工具图与缓存图，$\mathcal{R}_t$ 表示它们之间的动态关联关系。

每个图保留自身拓扑与运行时属性：

$$G_t^L = (V_t^L, E_t^L, X_t^L), \qquad L \in \{T, R, O, C\}, \tag{2}$$

其中 V_t^L 为节点集，E_t^L 为边集，X_t^L 为节点、边与图层属性。跨图关系集合 $\mathcal{R}_t$ 记录不同图元素之间的类型化依赖，例如任务需要资源、工具调用消费缓存对象、资源约束改变工具调用的可执行顺序。

该表示不限定单个图层是普通图、异构图还是超图。它确定两项要求：每个图层保留自身拓扑，跨图依赖被显式表示而非压缩进单一张图。

系统通过结构编辑演化：

$$\mathbf{G}_{t+1} = \mathbf{G}_t \oplus \Delta_t, \qquad \Delta_t = \left(\Delta G_t^T, \Delta G_t^R, \Delta G_t^O, \Delta G_t^C, \Delta \mathcal{R}_t\right), \tag{3}$$

其中每个 ΔG_t^L 包含节点与边的生成、删除与重组，$\Delta \mathcal{R}_t$ 记录跨图关系变化。图 1 展示了该表示：跨图关系是两个图层中具体实体之间的边，而不是中心枢纽；反馈关系可以从影响出现的图层返回至动作发生的图层。

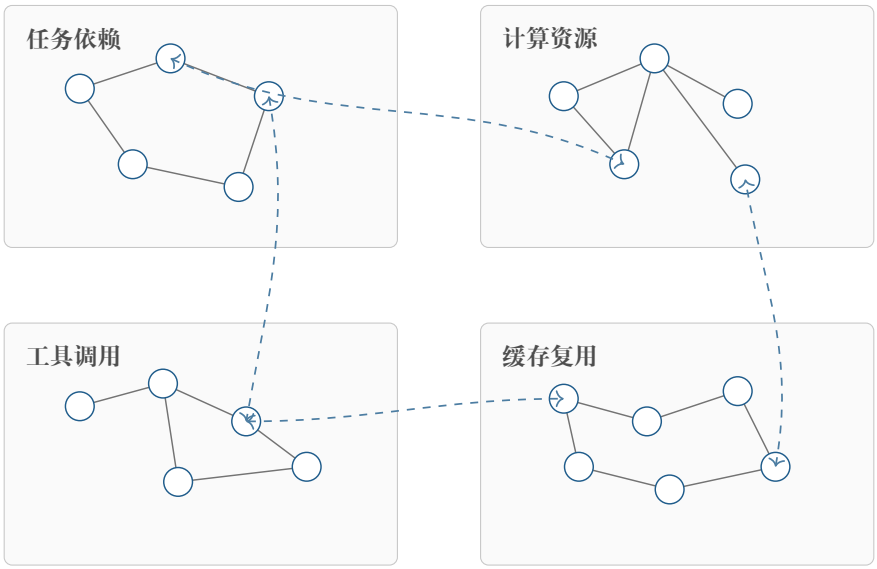

图 1: 四个动态执行图通过直接的跨图关系耦合。每个图层保留自身拓扑；虚线双向链接连接跨图层的具体实体，例如任务与其运行所在的资源、工具调用与其写入的缓存对象。反馈从影响出现的图层返回至动作发生的图层。

3.2 跨图结构转移

本研究的科学核心是跨图结构转移算子：它接收联合状态与执行动作，生成图层特定的结构增量，将增量跨图传播，并闭环反馈。图层特定的图内演化与闭环反馈是该算子的支撑组件，而非三个相互独立的预测器。该算子可以采用不同的神经网络基座；本研究的贡献是转移语义与架构，而非某一种编码器或解码器。

3.2.1 图内演化

每个图在图层特定的转移规则下演化。对于图层 L，结构增量由下式生成：

$$\Delta_t^L \sim p_\theta\left(\Delta_t^L \mid G_t^L, \mathcal{S}_t^L, a_t\right), \qquad L \in \{T, R, O, C\}, \tag{4}$$

其中 $\mathcal{S}_t^L$ 为该图层的演化历史，a_t 为执行动作。该机制刻画任务分解、资源分配、工具调用顺序与缓存复用，而不强制它们共享同一转移。

3.2.2 跨图结构传播

一个图的结构变化可以通过关系状态改变另一个图的转移分布。传播算子将图内增量集合与关系映射为其他图上的增量：

$$\Delta_t^{\text{prop}} = \Pi_\theta\left(\{\Delta_t^L\}_{L\in\{T,R,O,C\}}, \mathcal{R}_t, a_t\right). \tag{5}$$

它学习哪些关系承载变化、变化传播多远以及在何处停止。传播不是固定的消息传递规则，而是从观测到的多图转移中学习。

3.2.3 闭环反馈

传播后的变化可以返回源图。反馈算子将传播增量与关系映射回源图层：

$$\Delta_t^{\text{fb}} = \Phi_\theta\left(\{\Delta_t^L\}_{L\in\{T,R,O,C\}}, \Delta_t^{\text{prop}}, \mathcal{R}_t\right). \tag{6}$$

反馈可以是即时的或延迟的。模型学习返回路径，以及局部变化在何种条件下被放大、抑制或反转。该机制使多图世界模型区别于前馈式多图预测器。

3.2.4 组合与架构优势

联合转移组合三个机制：

$$\mathbf{G}_{t+1} = \mathbf{G}_t \oplus \left(\{\Delta_t^L\}_{L\in\{T,R,O,C\}} \cup \Delta_t^{\text{prop}} \cup \Delta_t^{\text{fb}} \right). \tag{7}$$

图内演化保留图层身份，传播建模跨图影响，反馈闭合回路。核心对象是这些机制的组合——作用于联合多图状态的转移算子，而非任何单一机制。统一的异构图世界模型会将第一项压缩为共享转移，并将后两项退化为普通边。表 2 总结了这一差异。

表 2: 多图世界模型与统一异构图世界模型的比较。

方面	统一异构图	AgentInfraGraphWorld
图层身份	压缩为节点与边类型	四个独立图，保留图层特定状态
状态转移	所有边共享一个转移	图内演化加跨图传播
反馈	隐含于消息传递	显式闭环反馈算子
输出	单张混合图中的边	直接输出四图结构与关系增量
归因	跨类型纠缠	逐图层传播与反馈轨迹

3.3 动作条件化多步联合推演

给定动作序列，世界模型推演未来联合结构：

$$\hat{\mathbf{G}}_{t+1:t+K} = F_\theta^{(K)}\left(\mathbf{G}_t, \mathcal{S}_t, a_{t:t+K-1}\right). \tag{8}$$

输出是结构化状态，而非文本。对于每个图，解码器输出节点与边的生成、删除与属性变化；对于关系状态，输出关系的生成、删除与更新。模型不生成自然语言描述，也不将其解析回结构。

训练结合三个目标：观测多图转移上的下一状态似然；预测状态与观测状态在 $2,\ldots,K$ 步上的多步推演一致性；跨图传播与反馈的辅助损失。多步项至关重要，因为任务图中的一步误差可能改变后续的资源竞争、工具顺序与缓存复用。

3.4 基于预测的执行优化

世界模型用于规划。给定一组合法动作序列，智能体选择

$$a^*_{t:t+K-1} = \arg\min_{a_{t:t+K-1}\in\mathcal{A}_{\text{valid}}} \mathbb{E}\left[C\left(\hat{\mathbf{G}}_{t+1:t+K}\right)\right], \tag{9}$$

其中 $\mathcal{A}_{\text{valid}}$ 为合法结构变换集合，C 为整体执行成本。

合法性由任务正确性与质量要求，以及资源与工具可用性限制共同定义。成本汇总延迟、token 与计算消耗、失败与重试惩罚以及缓存缺失。候选动作序列由模型提出的高概率结构变化生成，并通过推演评估。该过程采用滚动时域方式：执行动作后，更新实际图并重新预测未来结构。

3.5 实验设计与评估

数据采集。我们将对智能体运行时进行埋点，使规划器、调度器、工具执行器与缓存管理器发出带时间戳的事件，并由这些事件重建多图执行轨迹。长程任务将取自数据分析、文档处理与数学推理。

基线与消融。将 AgentInfraGraphWorld 与独立逐图预测、统一异构图世界模型、时序图神经网络及确定性运行时模型进行比较。消融实验将分别移除跨图传播、闭环反馈与动作条件化，以分离各组件的贡献。

评估维度。表 3 总结了评估方案。

表 3: 评估维度及其研究目标。

评估维度	研究目标	代表性指标
动作条件化预测	预测候选动作的影响，而非仅预测下一观测状态	反事实推演误差、动作排序准确率
H1：多图因子化	在匹配容量下检验显式四图状态是否优于统一异构图	匹配参数/计算下的预测误差与成本
多图结构预测	预测各图未来的拓扑结构	生成/消失 F1、关系准确率
跨图传播	学习局部变化如何到达其他图	传播路径 F1、目标变化准确率
闭环反馈	捕捉变化如何返回源图	反馈路径准确率、回传影响误差
长期推演	减少 K 步推演中的误差累积	K 步误差增长
泛化能力	预测未见的跨图结构组合	留出结构上的性能
执行优化	降低整体执行成本	成功率匹配下的成本降低

4 预期贡献与结论

本计划预期取得三项主要科学贡献。

1. **跨图结构状态转移学习（核心）**。学习一个图的结构变化如何传播至其他图并返回源图，直接输出未来图结构与关系变化，而非通过文本生成与解析。
2. **多图世界模型架构**。在四个独立执行图上组合图内演化、跨图结构传播与闭环反馈，保留图层身份与显式跨图关系。
3. **基于预测的执行优化**。利用多步联合推演，在任务正确性约束下选择合法的结构变换。

本研究将长程智能体的运行时基础设施视为一个由多个相互作用图构成的动态世界。通过将图内演化、跨图传播与闭环反馈作为同一转移算子的组件学习，为多图世界建模提供新的架构，并为智能体执行基础设施优化提供基础。核心经验问题是显式多图因子化是否值得其结构先验；H1 对比旨在回答该问题，而非预设答案。

参考文献

[1] S. Kim et al., "An LLM Compiler for Parallel Function Calling," in *Proc. International Conference on Machine Learning (ICML)*, 2024.

[2] C. Lin et al., "Parrot: Efficient Serving of LLM-based Applications with Semantic Variable," in *Proc. USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, 2024.

[3] D. Guo et al., "SAGA: Workflow-Atomic Scheduling for AI Agent Inference on GPU Clusters," 2026.

[4] H. Kang et al., "ThunderAgent: A Simple, Fast and Program-Aware Agentic Inference System," 2026.

[5] A. Li et al., "GraphFlow: A Graph-Based Workflow Management for Efficient LLM-Agent Serving," in *Proc. International Conference on Machine Learning (ICML)*, 2026.

[6] T. Feng, Y. Wu, G. Lin, and J. You, "Graph World Model," in *Proc. International Conference on Machine Learning (ICML)*, 2025.

[7] Z. Ding, Y. Li, and X. Xie, "WorldGraph: Graph-Native World Modeling," arXiv:2609.34159, 2026.

[8] J. Liu, S. Yang, M. Wang, Y. Wang, and B. Yu, "Graph World Models: Concepts, Taxonomy, and Future Directions," arXiv:2604.27895, 2026.

[9] B. Li et al., "Dynamic Graph World Modelling: Learning Future Dynamics from Temporal Context," arXiv:2610.19364, 2026.

[10] A. Schutz, N. Hawes, and V.-A. Darvariu, "Beyond Static Graph World Models: Learning Stochastic Latent Dynamics over Evolving Topologies," arXiv:2609.28670, 2026.

[11] W. Wang et al., "A Structural Dynamics Graph World Model: Unified Modeling, Constrained Rollout, and Interpretable Calibration," arXiv:2608.08689, 2026.

[12] S. Sun et al., "Agent-Editing World Model: Rethinking World Modeling for LLM Agents," arXiv:2609.28416, 2026.

[13] Y. Zuo et al., "Qwen-AgentWorld: Language World Models for General Agents," arXiv:2606.24597, 2026.

[14] M. Kivelä et al., "Multilayer Networks," *Journal of Complex Networks*, vol. 2, no. 3, pp. 203–271, 2014.

[15] E. Rossi et al., "Temporal Graph Networks for Deep Learning on Dynamic Graphs," arXiv:2006.10637, 2020.