研究计划书

AgentHyperEvo

面向长程 LLM 智能体的动态异构超图高阶结构共同演化学习

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

Amin Beheshti 教授
数据科学研究实验室主任
计算学院
科学与工程学院

麦考瑞大学

2026 年 11 月 27 日

摘要

大语言模型（LLM）智能体正日益承担需要动态任务规划、计算资源分配、外部工具调用与缓存复用的长程任务。上述过程各自形成具有不同运行时语义的动态图：任务依赖图、计算资源图、工具调用图与缓存复用图。这些图并非独立演化。某一图层的局部拓扑变化可能跨层传播、触发反馈，并在多轮执行中不断累积；资源不足、工具执行结果异常或缓存失效，同样可能迫使智能体重新规划，进而改变任务图拓扑。因此，仅对单个动态图建模，难以刻画整个执行系统的长期结构演化行为。

本计划提出 AgentHyperEvo：一种面向长程 LLM 智能体的动态异构超图框架，将执行状态表示为具有四个运行时语义层及显式跨层超边的动态异构超图。该框架将学习高阶结构共同演化模型，从当前超图及其演化历史预测下一时刻的联合结构；开展反事实结构预测，估计候选局部结构干预所诱导的未来多层轨迹；并利用这些预测选择满足执行约束的结构变换，在保持任务正确性与质量的前提下降低整体执行成本。实验将在数据分析、文档处理与数学推理任务上，将 AgentHyperEvo 与动态异构图、时序超图、关联二部图及确定性运行时模型进行比较，评估结构预测、高阶关系预测、反馈预测、反事实估计、泛化能力与执行优化六方面能力。

1 引言

LLM 智能体能够自主规划、推理并调用外部工具完成复杂任务。当任务周期从单次查询扩展到数据分析流水线、文档处理工作流与多步数学推理时，智能体执行本身已成为一个运行时系统：规划器分解目标，调度器分配计算与 API 容量，执行器调用工具，缓存管理器存储并复用中间结果。上述过程都会留下轨迹，而每条轨迹都具有持续变化的图结构。

该运行时包含四个核心图图层，如表 1 所示。它们并非同一数据的独立视图，而是语义不同、时间尺度不同、结构事件定义也不同的耦合过程。

表 1: 长程 LLM 智能体的四个运行时语义图图层。

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

层间耦合直接且具有实际后果。考虑将一个任务拆分为并行子任务：该拆分不仅改变任务图，也改变并发工具调用数量、资源需求模式以及可缓存中间结果的集合。若两个子任务竞争同一 GPU 或同一限流 API，调度器可能将其串行化，进而延迟依赖的工具调用并降低缓存命中机会。反之，工具超时、工具结果异常或缓存驱逐可能使规划假设失效，迫使智能体重新生成部分任务图。资源短缺还可能触发任务迁移，改变数据局部性，从而影响工具执行顺序与缓存复用。这些影响并非一次性发生：它们会在图层间传播、反馈至源头，并在长程运行的多轮执行中累积。

现有智能体基础设施研究主要优化该运行时的单个组件。任务依赖与工具调用结构被用于并行函数调用 [1]；应用级依赖被显式暴露以优化服务 [2]；工作流在 GPU 集群上以原子方式调度 [3]；智能体推理被赋予程序感知能力 [4]；工作流以图的形式进行管理以提升服务效率 [5]。这些系统表明，显式的执行结构能够带来显著的延迟与成本收益。然而，仍缺少对执行系统整体的建模：在已调研工作中，尚未发现将任务、资源、工具与缓存表示为同一异构超图的独立动态图层、学习图层间高阶结构共同演化、并预测局部结构干预将如何随时间重塑其他图层的方法。

本计划的核心科学问题是：

如何学习长程 LLM 智能体多层动态执行图之间的高阶结构共同演化规律，并预测局部结构变换对未来整体执行结构的影响？

为回答该问题，本计划提出 AgentHyperEvo。该框架将执行状态表示为四层动态异构超图，学习跨图层的联合共同演化模型，以候选结构干预为条件获得反事实未来轨迹，并利用预测轨迹选择在保持任务正确性与质量的同时降低整体成本的执行结构变换。本计划的主要贡献如下：

- **动态异构超图表示：**统一长程智能体的任务、资源、工具与缓存图层，将层内与跨层高阶关系作为一等对象。
- **高阶结构共同演化学习机制：**刻画异构图层之间的联合触发、跨层传播与递归反馈。
- **反事实结构预测方法：**建立局部结构干预与未来多层图轨迹之间的预测关系，为自适应执行结构优化提供基础。

2 研究现状

2.1 智能体执行图与推理基础设施

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，使任务依赖显式化。Parrot [2] 引入语义变量以暴露 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 LLM 智能体服务效率。

上述系统共同表明，显式执行结构能够显著降低延迟与成本。然而，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。据我们所知，尚无方法将任务、资源、工具与缓存显式建模为四个动态图图层，并系统学习其高阶结构共同演化。

2.2 超图表示学习

图描述成对关系，而超图允许单条超边连接任意数量的节点。由此，多实体共同参与的关系成为一等学习对象，而不再是成对链接的集合。超图神经网络（HGNN）[6] 通过超边级聚合将图卷积推广到超图；AllSet [7] 提出多重集函数框架，涵盖一大类超图神经网络。

这些工作为高阶结构学习提供了表示工具，但主要面向静态、同构超图。它们未提供图层特定的运行时语义、成员来自不同图层的异构跨层超边，也未提供超边及其成员关系在执行过程中动态生成、消失与重组的机制。本计划在这些形式化基础上，将其扩展至动态、多层、异构场景。

2.3 动态图学习

时序图网络（TGN）[8] 维护逐节点记忆，并根据带时间戳的事件流更新记忆，实现连续时间动态图学习。近期研究 [9] 表明，高阶结构能够提升时序图上的链接预测性能。这些方法建模单个图的演化，或其中固定的高阶模式。

它们尚未处理多个异构图层的联合演化——其中一个图层的事件会改变另一图层的事件分布——也不支持关于结构干预的反事实查询。因此，AgentHyperEvo 的共同演化与反事实部分超出了现有时序图模型的范围。

2.4 综合分析

由文献可得三点观察。第一，智能体基础设施研究已确认任务、资源、工具与缓存结构对性能至关重要，但分别处理这些结构。第二，超图学习为多实体关系提供了表示工具，但未面向动态异构图层。第三，时序图学习提供了事件驱动的演化机制，但未面向耦合的多层系统或干预。衔接这些研究方向需要四项能力：

(**R**1) 统一的四层动态异构超图，具有类型化节点、层内与跨层超边以及动态成员关系；

(**R**2) 可学习的共同演化模型，刻画跨层传播、联合触发与递归反馈；

(**R**3) 局部结构干预下未来多层轨迹的反事实预测；

(**R**4) 在任务正确性约束下利用上述预测的优化过程。

AgentHyperEvo 围绕这四项需求设计。

3 研究目标与方法

3.1 动态多层异构超图表示学习

第一项研究目标是建立执行状态的统一表示，在保留四个图层各自语义的同时显式刻画其交互。在时刻 t，整体执行超图为

$$\mathcal{H}_t = (\mathcal{V}_t, \mathcal{E}_t, \mathcal{X}_t), \tag{1}$$

其中 $\mathcal{V}_t$ 为异构节点集合，$\mathcal{E}_t$ 为层内与跨层超边集合，$\mathcal{X}_t$ 为相关运行时属性。

节点按图层赋予类型。令 $\mathcal{L} = \{\mathsf{T}, \mathsf{R}, \mathsf{C}, \mathsf{K}\}$ 分别表示任务、资源、工具与缓存图层，$\ell(v) \in \mathcal{L}$ 为节点 v 的图层。节点集合相应划分为

$$\mathcal{V}_t = \bigcup_{L \in \mathcal{L}} \mathcal{V}_t^L, \qquad \mathcal{V}_t^L \cap \mathcal{V}_t^{L'} = \emptyset \quad (L \neq L'). \tag{2}$$

一条超边为元组

$$e = (\tau_e, M_e, t_e, \mathbf{x}_e), \qquad M_e \subseteq \mathcal{V}_t, \qquad |M_e| \geq 2, \tag{3}$$

其中 τ_e 为类型，M_e 为成员集合，t_e 为时间戳，$\mathbf{x}_e$ 为运行时属性。层内超边的所有成员位于同一图层；跨层超边的成员来自两个或更多图层。类型 τ_e 记录该图层签名及关系语义，因此，连接任务、工具实例与资源的工具调用超边，与连接工具实例及若干缓存对象的缓存复用超边，是两类不同的对象。

定义 1 (动态多层异构执行超图)**.** *动态多层异构执行超图是时间索引序列 $\{\mathcal{H}_t\}_{t \geq 0}$，其节点与超边具有类型，且节点、超边及超边成员关系可在每一步生成、删除或重组。*

其动态性通过结构编辑表达。每一步

$$\mathcal{H}_{t+1} = \mathcal{H}_t \oplus \Delta_t, \qquad \Delta_t = \big(\Delta\mathcal{V}_t^+, \Delta\mathcal{V}_t^-, \Delta\mathcal{E}_t^+, \Delta\mathcal{E}_t^-, \Delta\mathcal{M}_t\big), \tag{4}$$

其中 $\Delta\mathcal{V}_t^{\pm}$ 为节点生成与消失，$\Delta\mathcal{E}_t^{\pm}$ 为超边生成与消失，$\Delta\mathcal{M}_t$ 记录成员关系替换。要求模型对结构编辑进行打分，而非预测固定规模的邻接矩阵，这一点至关重要：在长程运行中节点与超边数量持续变化，同一节点也可能在不被删除的情况下改变结构角色。

图 1 展示了该表示。阴影区域为层内超边；短虚线箭头表示相邻图层间的跨层依赖；外侧两条曲线分别表示跨越非相邻图层的跨层超边与递归反馈。

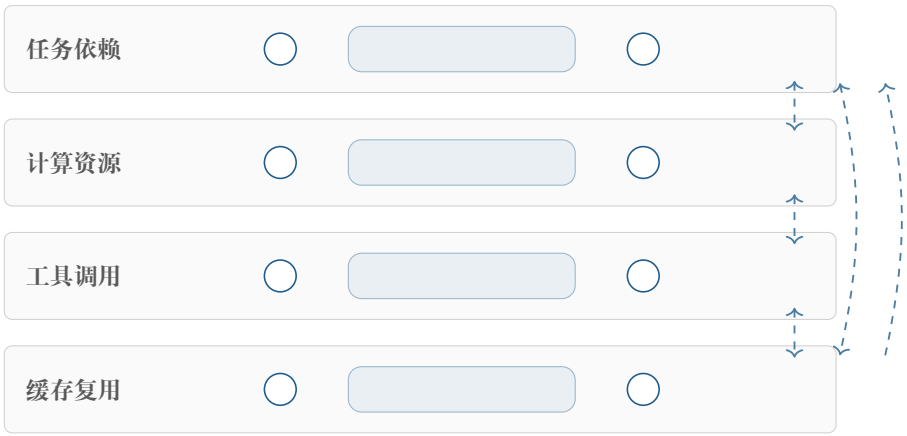


图 1: 四层执行超图示意。阴影框为层内超边，短虚线箭头为相邻图层间的跨层依赖，外侧两条曲线分别表示跨越非相邻图层的跨层超边与递归反馈。

3.2 高阶结构共同演化学习

第二项研究目标是学习图层之间如何相互影响。给定当前超图及其演化历史摘要，模型预测联合的下一时刻结构：

$$\hat{\mathcal{H}}_{t+1} = F_\theta(\mathcal{H}_t, \mathcal{S}_t), \tag{5}$$

其中 $\mathcal{S}_t$ 表示历史结构演化信息，F_θ 为待学习的共同演化模型。预测对象包括：

- 未来节点与超边的生成、消失与重组；
- 超边成员关系及节点结构角色的变化；
- 跨层高阶关系的形成与消失；
- 多层结构变化的传播路径与反馈关系。

我们将以结构事件为对象构造预测器。结构增量由以所有图层为条件的分布生成：

$$\Delta_t \sim p_\theta\big(\Delta_t \mid \mathcal{H}_t, \mathcal{S}_t\big), \qquad \hat{\mathcal{H}}_{t+1} = \mathcal{H}_t \oplus \hat{\Delta}_t, \tag{6}$$

多步预测通过自回归方式获得：

$$\hat{\mathcal{H}}_{t+1:t+K} = F_\theta^{(K)}(\mathcal{H}_t, \mathcal{S}_t). \tag{7}$$

模型将组合四个组件。*类型化超边编码器*将节点、超边及其成员关系映射为图层特定嵌入，并将跨层超边编码为异构成员上的高阶关系。*时序记忆*维护逐节点、逐超边与逐图层的状态，用于概括 $\mathcal{S}_t$，并由观测到的结构事件更新。*共同演化预测器*以所有图层及跨层上下文为条件，对下一事件分布进行建模，使任务事件能够改变资源、工具与缓存图层的事件得分，反之亦然。*结构化解码器*输出合法的结构编辑——生成、消失或成员关系变化——而非相互独立的成对链接，从而保证预测超图的良构性。

训练将结合观测轨迹上的下一事件似然，以及惩罚累积结构误差的多步 rollout 损失，并辅以成员关系预测与跨层关系预测的辅助目标。rollout 损失十分重要，因为模型的下游用途是 K 步预测，而非单步重构。

3.3 反事实结构预测

第三项研究目标是预测结构干预下的结果。给定候选干预 a，模型预测未来多层轨迹如下：

$$\hat{\mathcal{H}}^{(a)}_{t+1:t+K} = F_\theta(\mathcal{H}_t, \mathcal{S}_t, a). \tag{8}$$

其中，a 可作用于任意图层，表示局部任务子图替换、资源分配关系变化、工具调用调整或缓存复用关系变化。在实现上，将干预作用于当前状态的副本，$\mathcal{H}^{(a)}_t = \mathrm{do}(\mathcal{H}_t, a)$，并将预测器向前滚动 K 步。显式地以 a 为条件，而不仅以干预后的状态为条件，正是该查询具有反事实性质的原因：同一观测状态可与不同候选干预配对。

训练数据将通过在受控初始状态下进行配对执行获得。干预的直接确定性后果由构造已知——例如拆分任务会创建其子任务——而后续跨层响应则需要学习。因此，训练目标将区分这两个部分：由规则确定的增量部分被精确应用，模型学习剩余的演化部分。配对反事实损失将奖励模型按下游结构影响正确排序不同干预，同时拟合每条观测轨迹。

在推理阶段，比较不同干预的预测轨迹分布，即可估计其长期结构影响。该信号将用于第 3.4 节的结构优化过程。

3.4 整体执行结构优化

第四项研究目标是在反事实预测的基础上改进执行结构本身。给定一组合法候选变换，智能体选择

$$a^* = \arg\min_{a \in \mathcal{A}_{\mathrm{valid}}} C\left(\hat{\mathcal{H}}^{(a)}_{t+1:t+K}\right), \tag{9}$$

其中 $\mathcal{A}_{\mathrm{valid}}$ 为满足执行约束的结构变换集合，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；若某项变换虽能降低预测成本但违反上述要求，则被排除。成本 C 汇总四个图层所暴露的运行时量，包括延迟、token 与计算消耗、失败与重试惩罚以及缓存缺失。由于预测轨迹具有不确定性且干预空间具有组合性，候选干预将由模型提出的高概率结构事件生成，并通过 rollout 评估，选择时考虑预测不确定性。该过程采用滚动时域方式：执行所选变换后，利用观测事件更新超图并重复优化。

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，并由这些事件重建四层超图及其演化轨迹。长程任务将取自数据分析、文档处理与数学推理，这些任务均会涉及四个图层，但结构模式各不相同。

基线与消融。将 AgentHyperEvo 与以下方法比较：(i) 不含超边的动态异构图；(ii) 时序超图模型；(iii) 关联二部图表示；(iv) 确定性运行时模型。消融实验将分别移除跨层超边、演化历史与反事实训练，以分离各组件的贡献。

评估维度。表 2 总结了评估方案。结构预测采用节点与超边的生成—消失 F1 及成员关系准确率；高阶关系预测采用跨层超边 F1；反馈预测采用跨层影响的方向与幅度准确率；反事实预测采用预测干预影响与观测影响之间的误差，以及干预之间的排序相关性；泛化能力采用训练中未见的结构组合与任务族上的性能；执行优化采用在任务成功率与质量匹配条件下的成本降低。所有比较均在任务正确性匹配的条件下进行，以确保成本降低不以输出质量下降为代价。

表 2: 评估维度及其研究目标。

评估维度	研究目标	代表性指标
结构预测	准确预测未来多层图结构	生成/消失 F1、成员关系准确率
高阶关系预测	识别多个图层之间的联合交互	跨层超边 F1
反馈预测	捕捉跨层双向结构演化	影响方向与幅度准确率
反事实预测	估计不同结构干预的影响	干预影响误差、排序相关性
泛化能力	预测训练中未见的结构组合	留出任务与结构上的性能
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. **动态异构超图表示**。面向长程智能体执行，建立统一的多层高阶表示，将任务、资源、工具与缓存结构建模为相互区别又相互耦合的动态图层。
2. **高阶共同演化学习**。提出一种新的图学习机制，学习异构图层之间的联合交互、跨层传播与递归反馈，超越现有时序图与超图模型所假设的单图演化。
3. **反事实结构预测**。建立局部结构干预与未来整体图结构之间的预测关系，为自适应执行结构优化提供基础。

AgentHyperEvo 旨在从图机器学习视角揭示长程智能体复杂运行时系统的结构演化规律，推动动态异构图学习与智能体执行基础设施的交叉研究。该框架将进行端到端评估：将预测结构与观测轨迹比较，将预测干预影响与配对执行比较，并在任务质量相同的条件下，将所选变换与未修改运行的执行成本进行比较。

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] Y. Feng et al., "Hypergraph Neural Networks," in *Proc. AAAI Conference on Artificial Intelligence (AAAI)*, 2019.

[7] E. Chien et al., "You Are AllSet: A Multiset Function Framework for Hypergraph Neural Networks," in *Proc. International Conference on Learning Representations (ICLR)*, 2022.

[8] E. Rossi et al., "Temporal Graph Networks for Deep Learning on Dynamic Graphs," 2020.

[9] J. Liu et al., "Higher-order Structure Boosts Link Prediction on Temporal Graphs," 2025.