ITEM #203 - ACLM–LLM Bridge: Using LLMs as KD-CPI Probes for Operation(X)→Y Under Structural Governance

Conversation： ACLM Vertical Bridging Simplification

20260101

Authors: Sizhe Tan & GPT-Obot

ITEM #203 - ACLM–LLM Bridge: Using LLMs as KD-CPI Probes for Operation(X)→Y Under Structural Governance

(English Version)

Abstract

This item formalizes a disciplined integration point between ACLM and Large Language Models (LLMs), termed the **ACLM–LLM Bridge**.
Within the Knowing–Doing Co-Progress Intelligence (KD-CPI) paradigm, LLMs are positioned **not as executors or authorities**, but as **governed probes** that propose candidate operations for irreducible gaps in ACLM, specifically the Operation(X)→Y residual case.

The framework ensures that all LLM-generated proposals are subjected to strict structural validation, risk budgeting, and evidence governance before any acceptance or stabilization.

---

1. Motivation: The Residual Gap in ACLM

ACLM Vertical and Horizontal Bridging prioritize:

1. Anchor-based zoom-in
2. Calling Graph path extraction
3. Structured local ramp construction

However, irreducible gaps may remain due to:

- Incomplete structures
- Fuzzy or missing metrics
- Temporarily unavailable calling paths

Traditionally, these cases fall back to **Operation(X)→Y**, which risks premature structural collapse if hard-coded too early.

---

2. KD-CPI Perspective: Action Before Explanation

KD-CPI legitimizes action under incompleteness:

- Decisions must occur before full explanation
- Actions generate evidence
- Evidence enables later structural consolidation

LLMs exhibit strong KD-CPI behavior at runtime, particularly in **implicit consistency repair** that cannot be fully articulated symbolically.

This makes LLMs suitable for **proposal generation**, not for structural commitment.

---

3. Proper Role of LLMs in ACLM

LLMs must **never** be placed in the execution or authority position:

```
X ──(LLM executes)──> Y        ❌ Prohibited
```

Instead, LLMs occupy a governed proposal role:

```
X ──(LLM proposes)──> {Op₁, Op₂, Op₃, …}
                          ↓
                  Structural Filters
                          ↓
              Risk & Evidence Governance
                          ↓
                 Accept / Reject / Probe
```

Thus, LLMs act as **Operation Candidate Generators**, not operators.

---

4. What LLMs Are Allowed to Do

Within the ACLM–LLM Bridge, LLMs may:

- Propose candidate Operation(X)→Y mappings
- Suggest intermediate symbolic or procedural steps
- Offer repair strategies based on implicit consistency
- Generate multiple competing hypotheses under ambiguity

LLMs must not:

- Execute operations directly
- Modify calling graphs
- Commit structural changes
- Bypass evidence or risk constraints

---

5. Structural Governance Pipeline

Every LLM proposal must pass through the following stages:

1. **Structural Invariant Check**
   Reject candidates violating known ACLM invariants.
2. **Reachability & Consistency Test**
   Attempt partial validation via calling graphs and metrics.
3. **Risk Budget Evaluation**
   Ensure exploration cost remains bounded.
4. **Evidence Ledger Registration**
   Record outcomes as evidence, not structure.
5. **Deferred Stabilization**
   Structural solidification only after repeated confirmation.

This pipeline ensures reversibility and accountability.

---

6. Why “Non-Explainable” Is Acceptable Here

In residual gap contexts:

- Requiring full symbolic explainability too early suppresses viable paths
- Implicit repair mechanisms may precede formal description

LLM proposals are therefore treated as **hypotheses**, not truths.

Explainability is demanded **after** structural convergence, not before.

---

7. Engineering Principle

LLMs may propose where structure cannot yet explain,
but only structure decides what may persist.

This principle governs all ACLM–LLM interactions.

---

8. Implications

- Prevents premature hard-coding of incorrect operations
- Exploits LLM KD-CPI strengths without surrendering control
- Preserves DBM's constructive evolution discipline
- Converts black-box behavior into evidence-driven evolution

---

9. Conclusion

The ACLM–LLM Bridge provides a principled, safe, and productive role for LLMs in DBM systems.

By restricting LLMs to governed proposal generation under KD-CPI, ACLM can resolve residual gaps without sacrificing structural integrity or evolutionary accountability.

ITEM #203 - ACLM–LLM Bridge：在结构治理下将 LLM 作为 Operation(X)→Y 的知行同进探针

（中文版）

摘要

本文提出 **ACLM–LLM Bridge**，在**知行同进智能（KD-CPI）范式下，为 ACLM 与大语言模型（LLM）之间建立一个严格受治理的集成接口**。

在该框架中，LLM **不作为执行者，也不作为裁决者**，而仅作为 **Operation(X)→Y 残余 Gap 的候选生成探针**。
所有 LLM 产出的候选，必须经过 DBM 的结构校验、风险预算与证据治理，方可被接受或逐步固化。

---

1. 问题背景：ACLM 的残余 Gap

ACLM Bridging 的优先路径为：

1. 锚点化 Zoom-In
2. Calling Graph 路径抽取
3. 局部引桥构造

在结构与度量尚不完备时，仍可能存在不可约 Gap，传统做法是直接引入 **Operation(X)→Y**，但这极易造成过早结构塌缩。

---

2. KD-CPI 视角：先行动，后解释

KD-CPI 允许在不完备条件下行动：

- 行动先于解释
- 行动生成证据
- 证据促成结构

LLM 在运行态展现出强烈的 KD-CPI 特征，尤其擅长**不可言说的一致性修复**，这使其适合作为**假设生成器**。

---

3. LLM 在 ACLM 中的正确位置

LLM **绝不可**直接执行操作：

```
X ──(LLM 执行)──> Y          ❌ 禁止
```

其合法位置是：

```
X ──(LLM 提议)──> {Op₁, Op₂, Op₃, …}
                    ↓
              结构不变量校验
                    ↓
              风险 / 证据治理
                    ↓
               接受 / 拒绝 / 试探
```

即：**LLM 是候选生成器，而非 Operation 本体。**

---

4. LLM 的允许行为与禁止行为

允许：

- 生成 Operation(X)→Y 候选
- 提供隐式修复策略
- 给出多种不确定假设

禁止：

- 直接执行操作
- 修改 Calling Graph
- 固化结构
- 绕过风险与证据体系

---

5. 结构治理流程

每一个 LLM 候选必须经过：

1. 结构不变量检查
2. 可达性与一致性验证
3. 风险预算评估
4. 证据账本登记
5. 延迟结构固化

该流程保证**可回滚、可审计、可演化**。

---

6. 为什么允许“不可言说”的候选

在残余 Gap 区域：

- 过早要求形式化解释会抑制可行路径
- 隐式一致性修复往往先于规则生成

因此，LLM 产出被视为**假设**而非结论，解释责任推迟到结构收敛之后。

---

7. 工程原则（核心）

LLM 可以在结构尚未成形处提出假设，
但只有结构，才能决定什么可以留下。

---

8. 工程意义

- 避免过早硬编码错误 Operation
- 合理利用 LLM 的 KD-CPI 能力
- 保持 DBM 的建构性演化纪律
- 将黑箱行为转化为可治理证据

---

9. 总结

ACLM–LLM Bridge 为 LLM 在 DBM 体系中提供了**正确而有限的位置**。

它既不否认 LLM 的智能价值，也不牺牲结构智能的长期演化目标，是一种成熟、克制、可持续的工程集成范式。
