

Meta-Driven Test and Evaluation of LLM Agents

Nathalia Nascimento, Thao Min Le, and Nil Ergin

Presenter: Nil Ergin

AI4SE & SE4AI Workshop

September 23, 2026



PennState

The Research Problem

Integration of LLMs into Multiagent Systems

- Support communication, reasoning, decision making, interaction with humans

LLM Agent Behavior

- Nondeterministic, prompt sensitive, context-dependent, and emergent from agent interactions

Test & Evaluation Challenges

- Failures not attributed to a single software
- Sequence of LLM calls, messages, decisions, local actions, global interactions

Problem

Systematic Test and Evaluation method to evaluate LLM agents within MAS settings

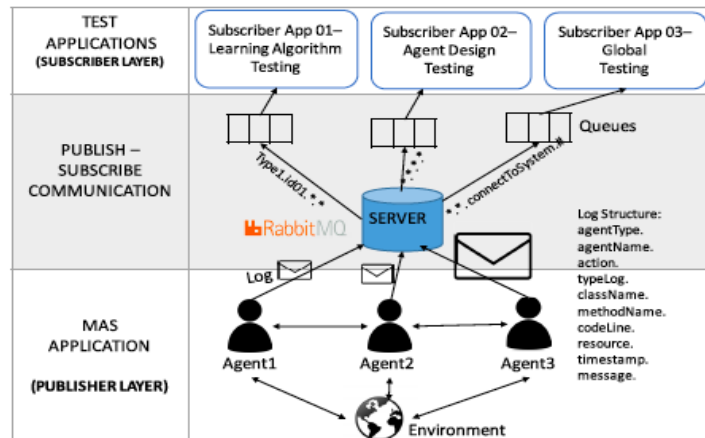


PennState

Test and Evaluation Model

Prior Work

- Metadata-driven T&E approach for MAS



- Metadata included: agent type, agent name, action, log type, resource, timestamp, message

Gap

This Research

- Extend Metadata T&E model for LLM-specific metadata.
- Testing LLM agent requires observing both external input-outputs as well as intermediate artifacts involved in LLM reasoning
- Trace global failure back to specific LLM invocations, prompts, contextual inputs, actions, and execution outcomes



PennState

Proposed Approach: Metadata T&E for LLM Agents



Extension of the Metadata Model

- LLM-specific metadata
 - ***llmCallId***: Unique identifier for each LLM invocation
 - ***promptId***: Identifier of the prompt template used
 - ***promptVersion***: Version of the prompt or instruction set
 - ***modelId***: Identifier of the LLM used
 - ***modelParameters***: temperature, maximum tokens, related parameters
 - ***contextInput***: summarized sensor data, received messages, memory items, feedback provided to the LLM
 - ***generatedArtifact***: generated controller, plan, rule, message, explanation, or adaptation strategy
 - ***constraintStatus***: whether generated artifact satisfies syntactic, semantic, safety, or scenario constraints
 - ***selectedAction***: executable action derived from the LLM output
 - ***executionOutcome***: result of executing the selected action or generated artifact in the MAS
- Local and global T&E levels
 - Local tests
 - Interaction tests
 - Adaptation tests
 - Constraint tests
 - Global tests

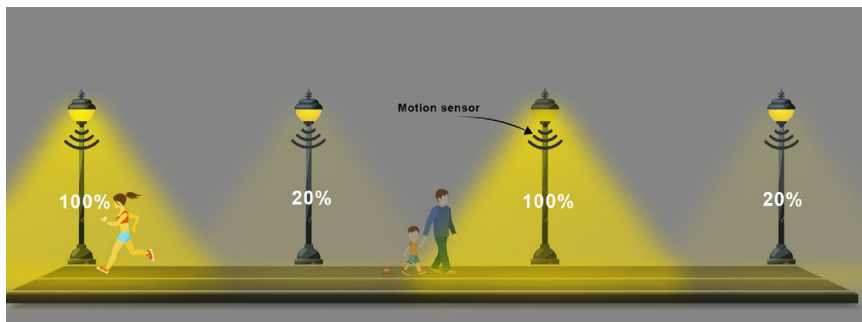


Illustrative Scenario: Smart Streetlights

LLM-Enabled Smart Streetlight

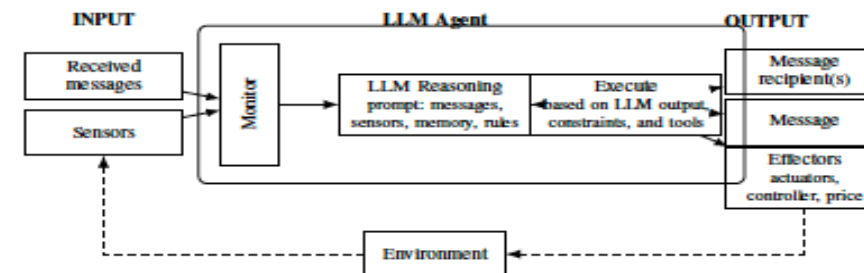
Autonomous streetlights agents

- Ambient light sensors
- Motion sensors
- Wireless communication
- Light actuators
- Objective: reduce energy consumption while maintaining sufficient illumination



Implementation

- Original self-organizing version
 - NN controllers evolve using GA
- LLM-enabled version
 - If-else controllers based on problem statement and feedback from previous simulation results
- Controller evaluated in MAS simulation
- Resulting feedback used to request an improved controller



Preliminary LLM Enabled Smart Street Light

Solution	Energy	People	Trip	Fitness
LLM Iteration 1	4.03	66.66	59.25	29.49
LLM Iteration 2	15.02	100.00	54.62	61.20
LLM Iteration 3	11.92	100.00	54.62	62.44
Best Neuroevolution	8.10	100.00	62.03	59.53
Best Human Solution	9.46	100.00	55.55	62.88

- Need for systematic test of LLM agents
 - Traceability among LLM prompt, generated controller, local agent actions, and global outcomes
 - Final fitness value not sufficient

Meta-data Driven Tests for LLM Agents

Level	Function	Procedure	Expected Result	Metadata Evidence
Local	Prompt execution	Agent sends contextual inputs to the LLM.	LLM call is identified, versioned, and associated with model parameters.	llmCallId, promptId, promptVersion, modelId, modelParameters, timestamp.
Local	Artifact generation	LLM generates a controller, plan, rule, message, or explanation.	Generated artifact is recorded and linked to the LLM call.	llmCallId, contextInput, generatedArtifact, timestamp.
Adaptation	Feedback loop	The system evaluates a controller and provides performance feedback as context.	New artifact version is generated using prior execution results.	llmCallId, contextInput, promptVersion, generatedArtifact, executionOutcome.
Interaction	Agent communication	Agents exchange wireless or message-based signals.	Messages are consistent with the selected action and scenario assumptions.	agentName, action, resource, message, selectedAction, timestamp.
Global	Mission evaluation	MAS executes the selected controller in the streetlight scenario.	Energy target is met and all people complete routes.	selectedAction, executionOutcome, energyValue, peopleCompleted, tripDuration, fitnessValue.
Constraint	Constraint checking	Test application verifies whether the generated artifact satisfies scenario constraints.	Invalid or unsafe artifacts are rejected before execution.	llmCallId, generatedArtifact, constraintStatus, selectedAction, executionOutcome.



Implications for Practice



Systematic T&E approach

Reusable

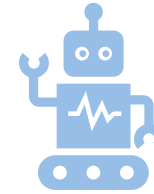


Support

Evaluate LLM before
deployment

Monitor LLM during
operation

Fault diagnosis



AI-enabled Applications

Mission-oriented cyber-
physical systems

LLM agents that negotiate,
allocate resources,
coordinate tasks



PennState

Future Work

- Implement the extended metadata T&E approach
- Evaluate the approach across multiple LLM-agent scenarios
- Investigate how resulting traces can support assurance cases for AI-enabled systems



Q&A Thank you.

Nil Ergin: nhe2@psu.edu



PennState