ISPASS

Semiconductor Research Corporation

CoCoSys
CENTER FOR THE CO-DESIGN OF COGNITIVE SYSTEMS

Generative AI in Embodied Systems: System-Level Analysis of Performance, Efficiency and Scalability

Zishen Wan[1], Jiayi Qian[1], Yuhang Du[2], Jason Jabbour[3], Yilun Du[3], Yang (Katie) Zhao[2], Arijit Raychowdhury[1], Tushar Krishna[1], Vijay Janapa Reddi[3]

Georgia Tech

UNIVERSITY OF MINNESOTA

HARVARD UNIVERSITY

IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS), 2025

Autonomous Machine Era

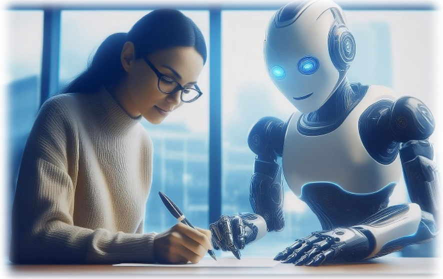
- Autonomous Machines on the Rise

Self-Driving Cars

Drones

Legged Robot

AR/VR



Embodied AI Robot

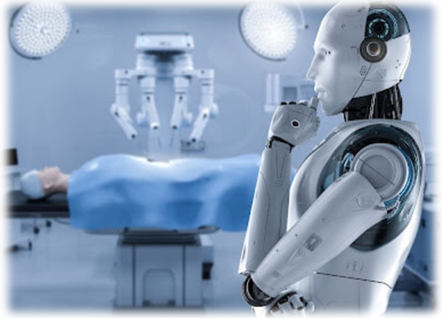
- Wide Application Potential

Package Delivery

Search & Rescue

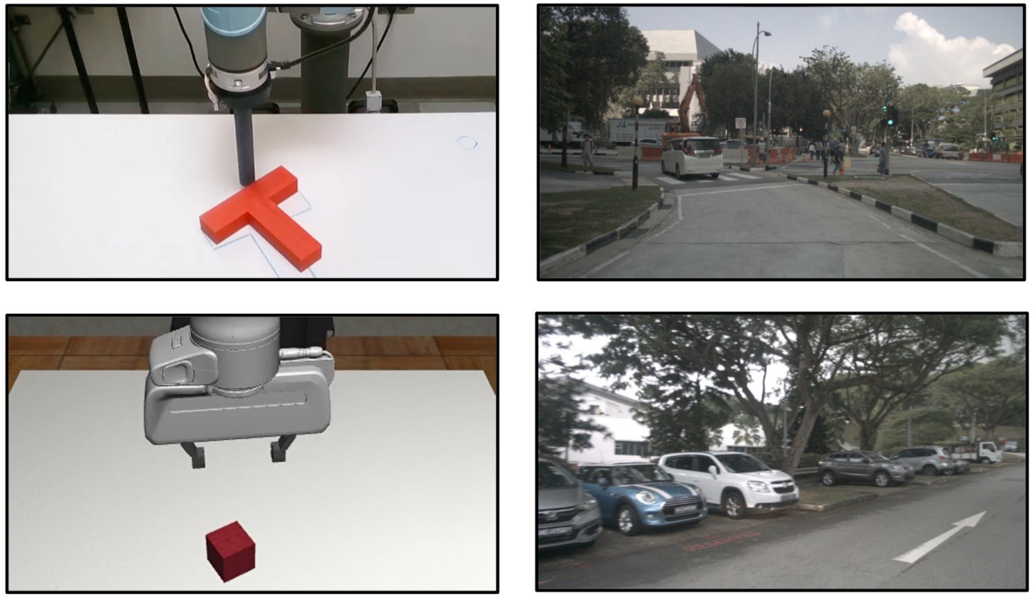
Agriculture

Manufacture



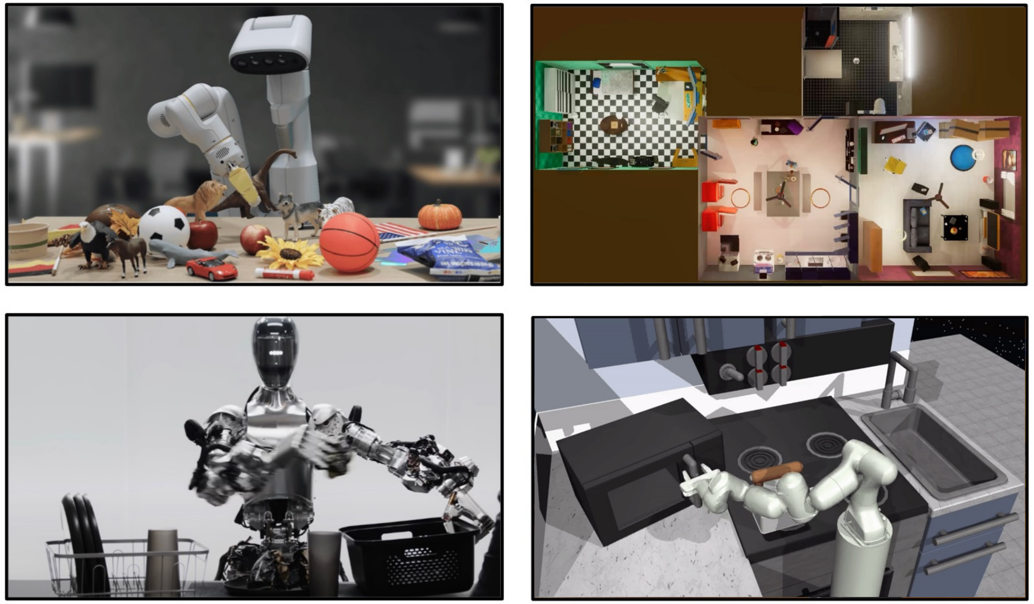
Healthcare

From Simple Tasks to Complex Long-Horizon Tasks



Static Simple Tasks

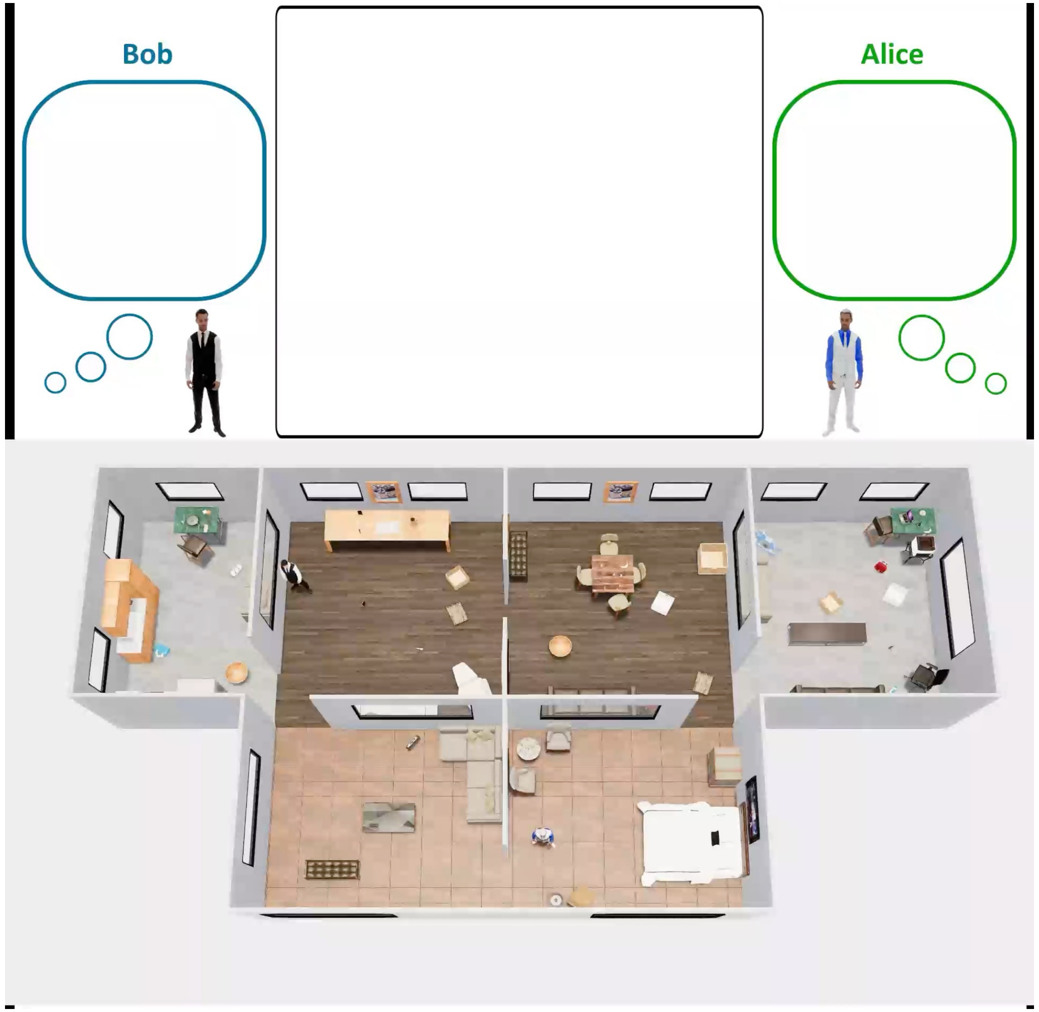
Complex Long-Horizon Multi-Objective Tasks



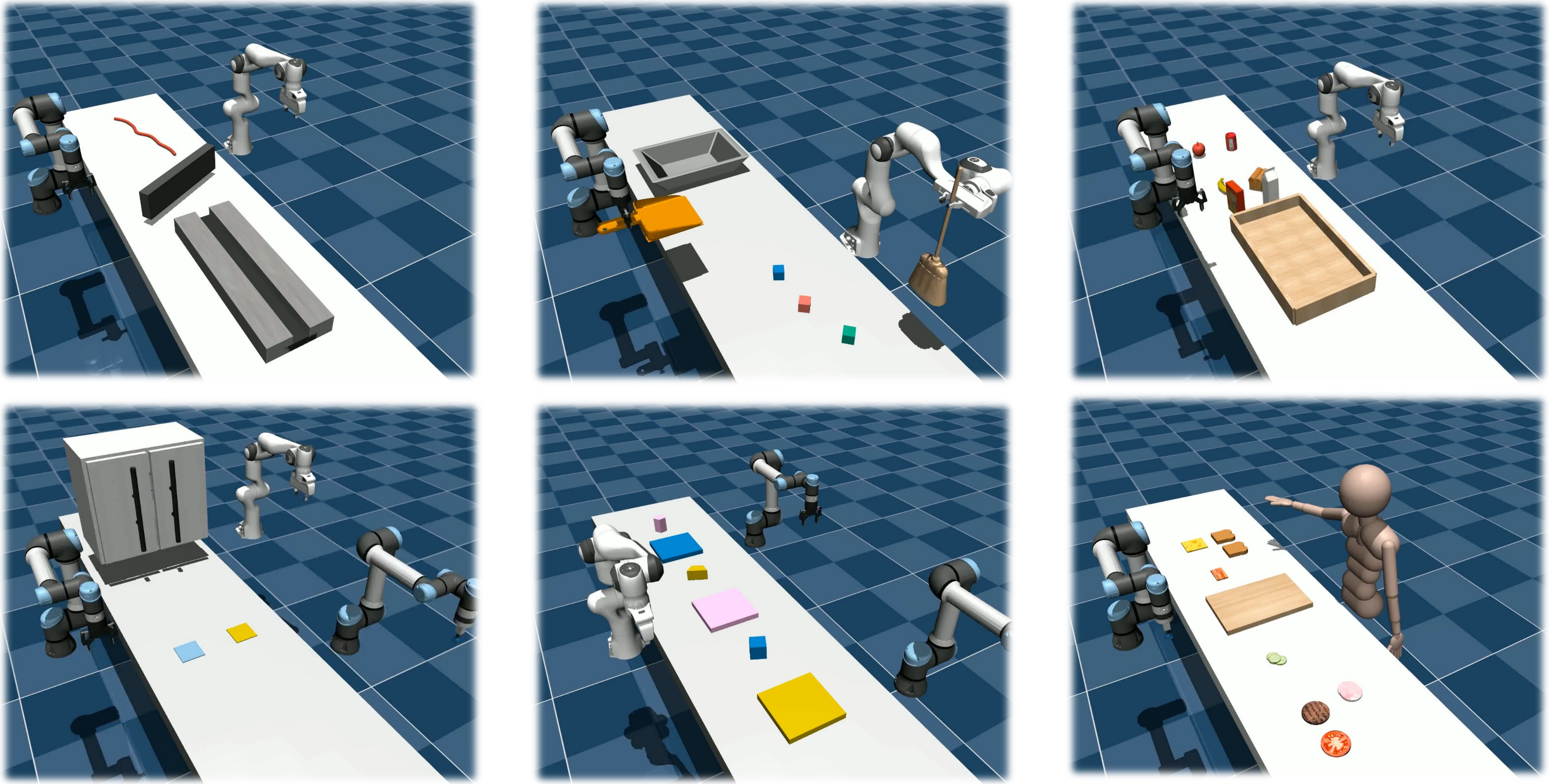
Long-Horizon Multi-Objective Planning

- **Task**: long-horizon multi-objective task and motion planning
  - Features: require long-term planning and reasoning capability
  - Examples: household tasks, transport objects, make meal, set up table, cook…

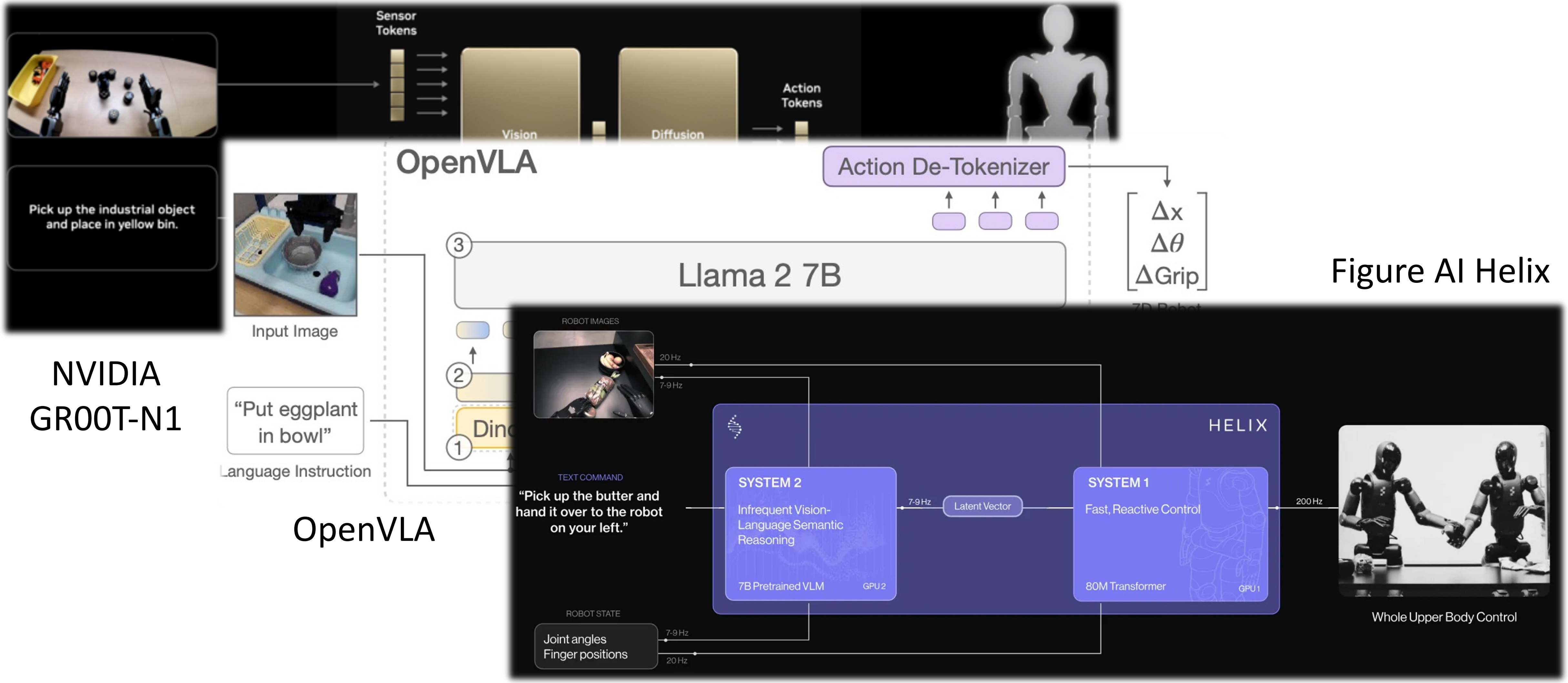
Demo: Long-Horizon Multi-Objective Planning



Demo: Long-Horizon Multi-Objective Planning



Generative AI-Inspired Embodied Systems

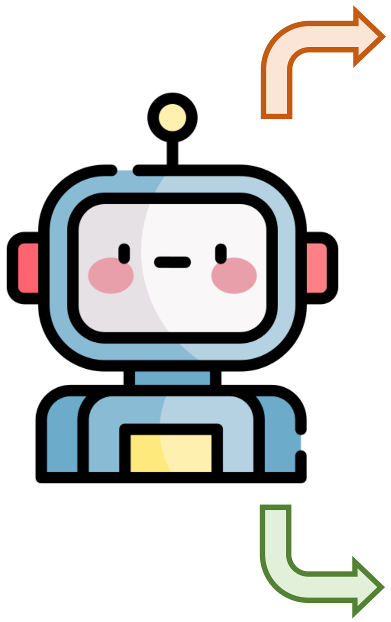


NVIDIA GR00T-N1

OpenVLA

Figure AI Helix

Embodied Agentic Systems

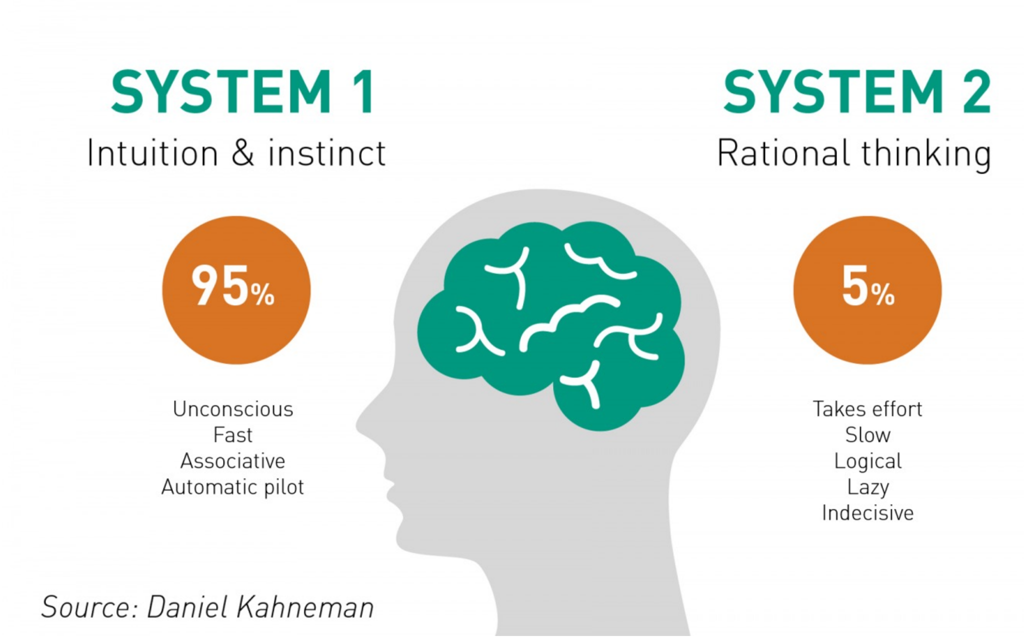


Cognition Capability (System 2)

Human-like reasoning
Trustworthy decision making
Human-agent interaction

Autonomy Capability (System 1)

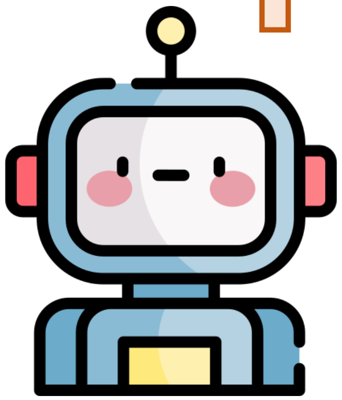
Perception, localization and mapping, path planning, control and action



System 1
(Fast thinking)
intuitive, instinctive, automatic

System 2
(Slow thinking)
rational, logical, cognitive

Embodied Agentic Systems



Cognition Capability (System 2)

Human-like reasoning
Trustworthy decision making
Human-agent interaction

Autonomy Capability (System 1)

Perception, localization and mapping, path planning, control and action

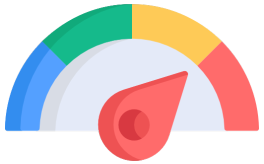
Requirements

- Real-Time ***Performance***
- Energy ***Efficiency***
- Design ***Scalability***

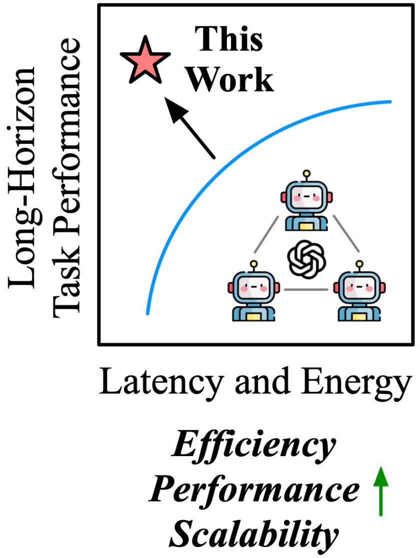
Goal of this Work

- *Understand* fundamental **building blocks** and **paradigms** of embodied systems.

- *Identify* **system characteristics** and **sources of inefficiency** of embodied systems.



- *Demonstrate* **optimization opportunities** and **scalability-efficiency improvements** for embodied systems.



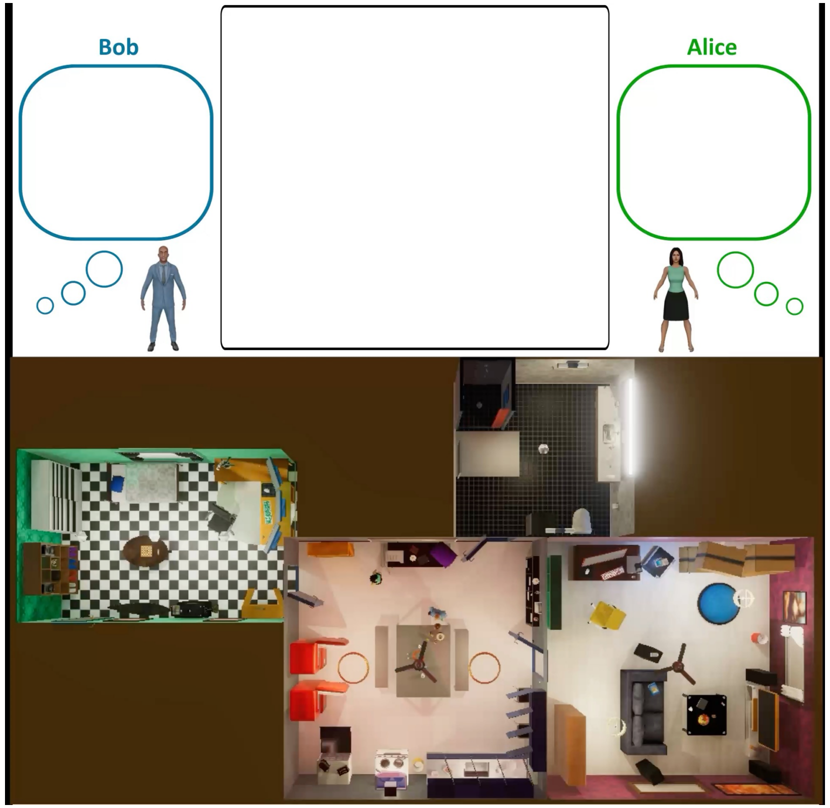
Outline

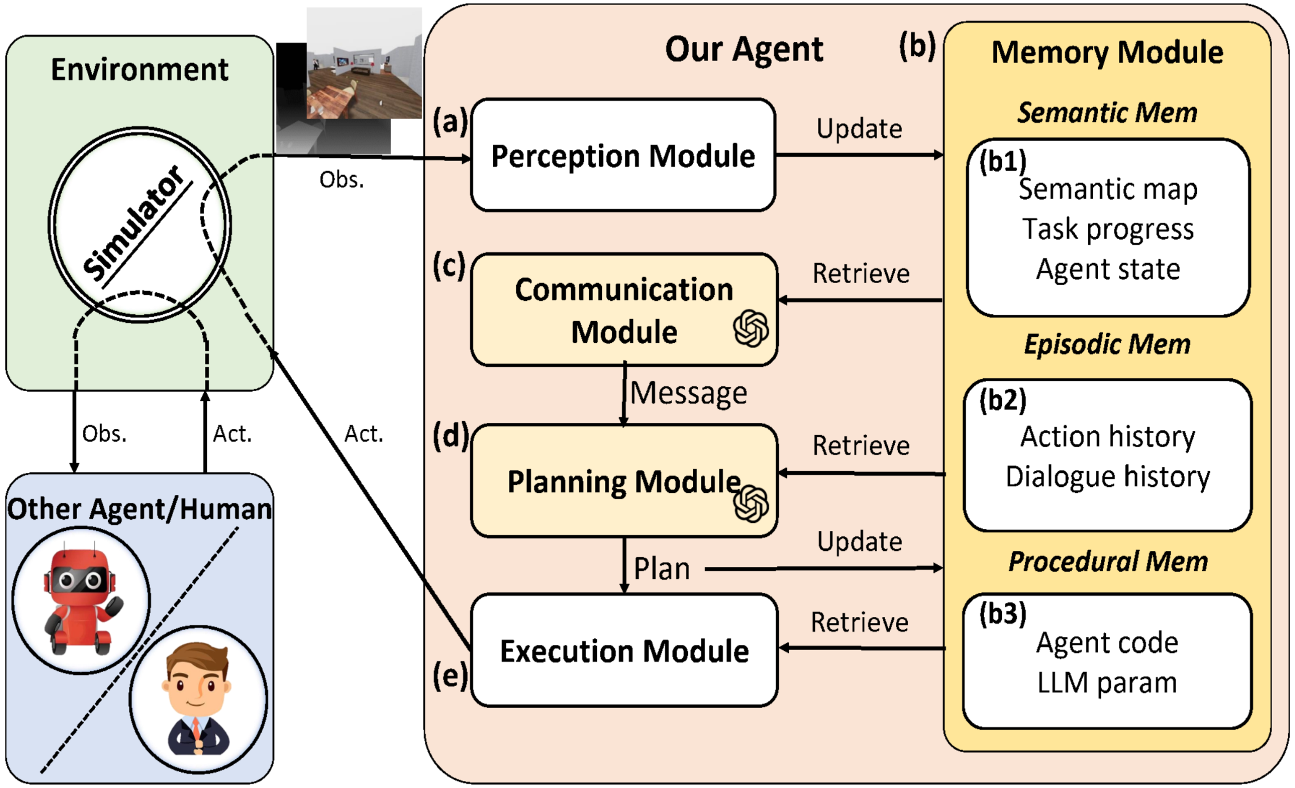
- *Understand* fundamental **building blocks** and **paradigms** of embodied systems.

- *Identify* **system characteristics** and **sources of inefficiency** of embodied systems.

- *Demonstrate* **optimization opportunities** and **scalability-efficiency improvements** for embodied systems.

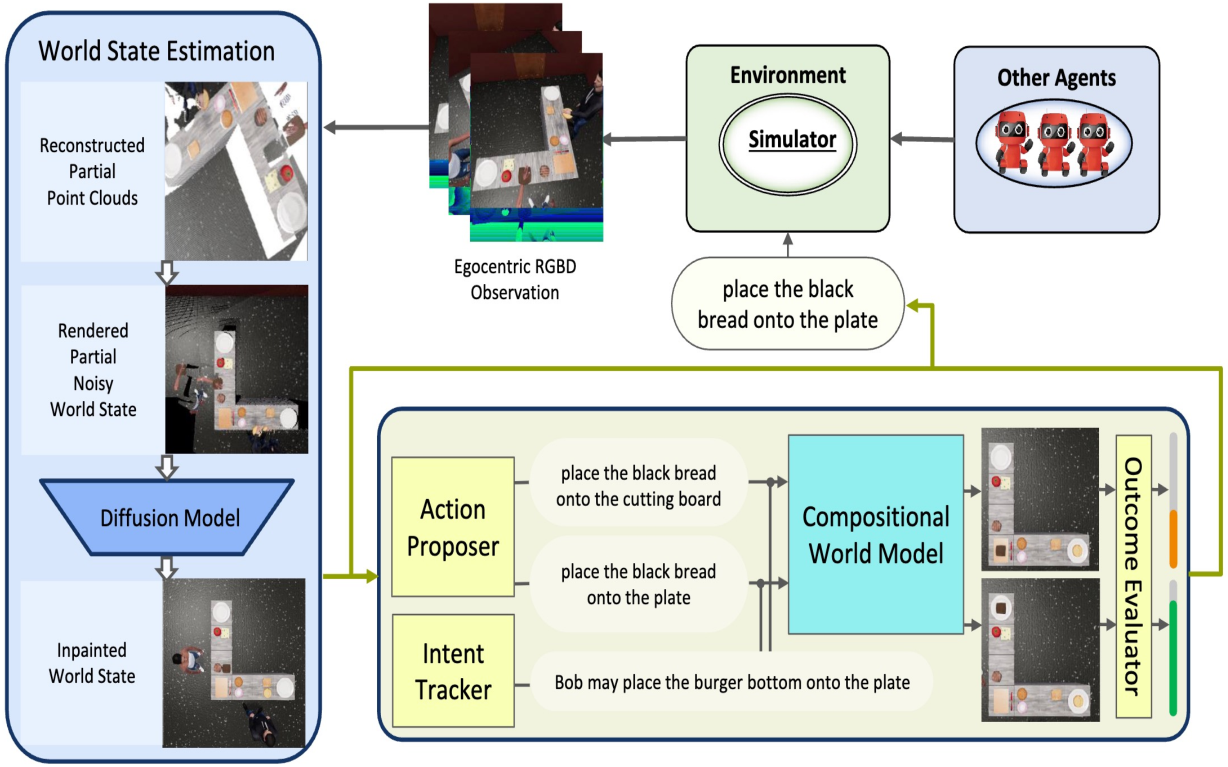
Embodied System Example: CoELA





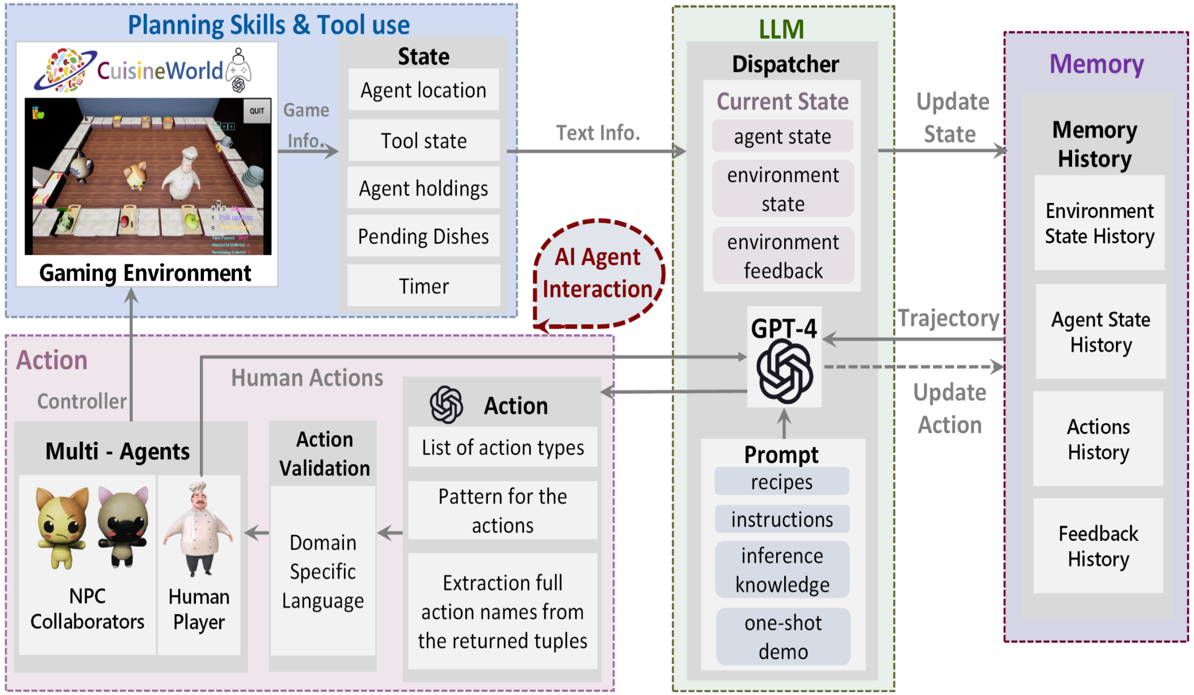
Zhang et al, "CoELA: Building Cooperative Embodied Agents Modularly with Large Language Models", in ICLR 2024

Embodied System Example: COMBO



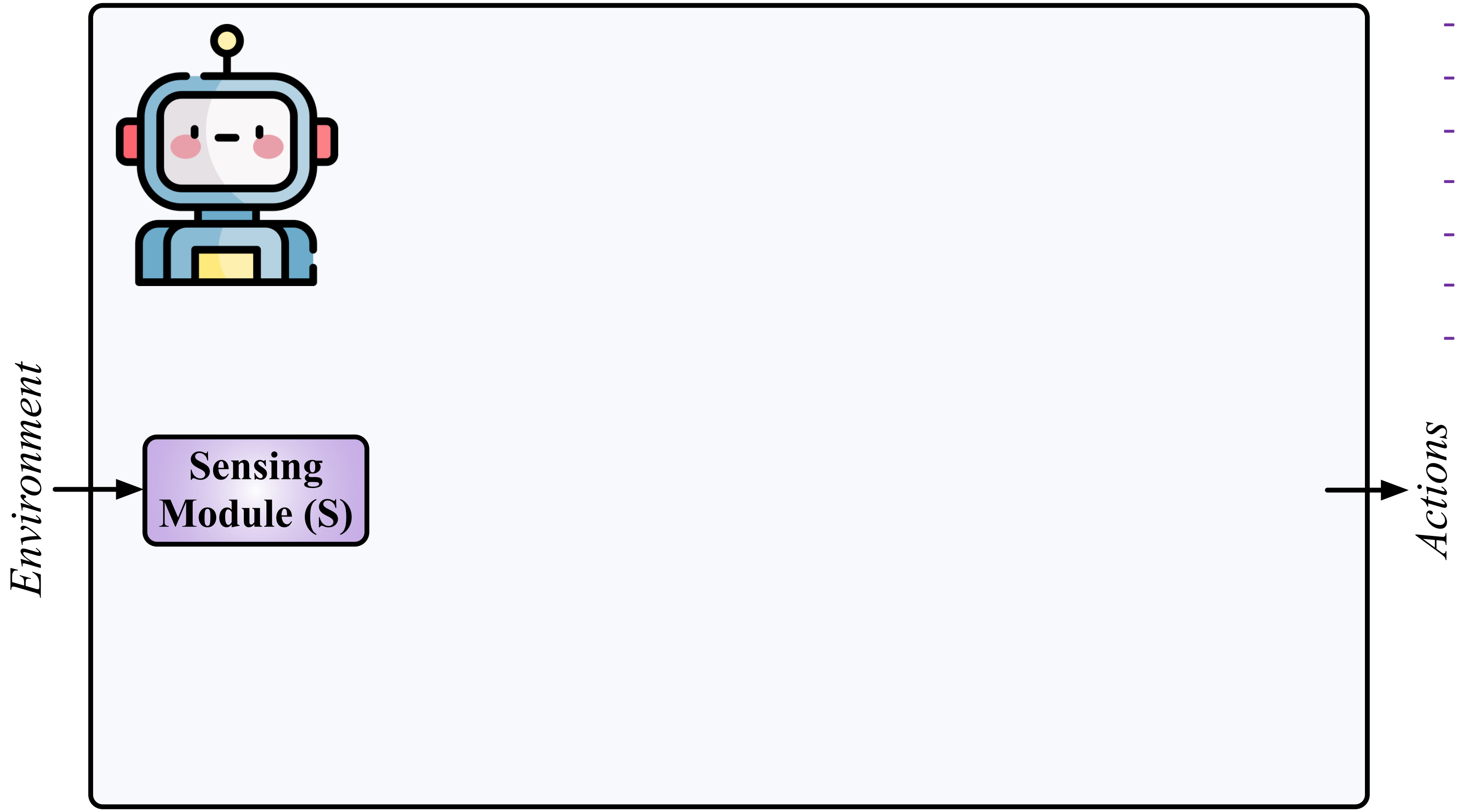
Zhang et al, "COMBO: Compositional World Models for Embodied Multi-Agent Cooperation", in ICLR 2025

Embodied System Example: MindAgent



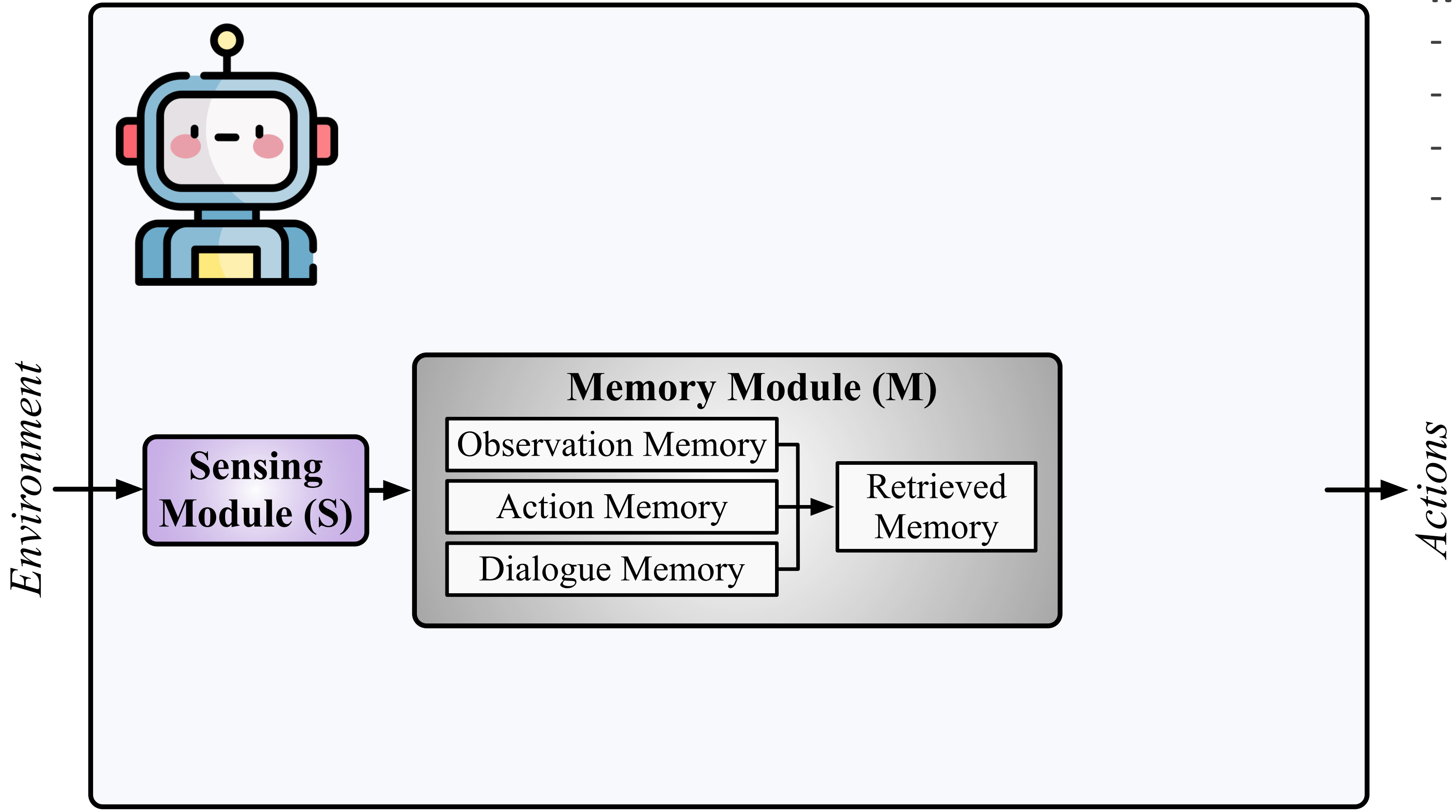
Embodied Agent System Paradigm

Embodied Agent System Paradigm



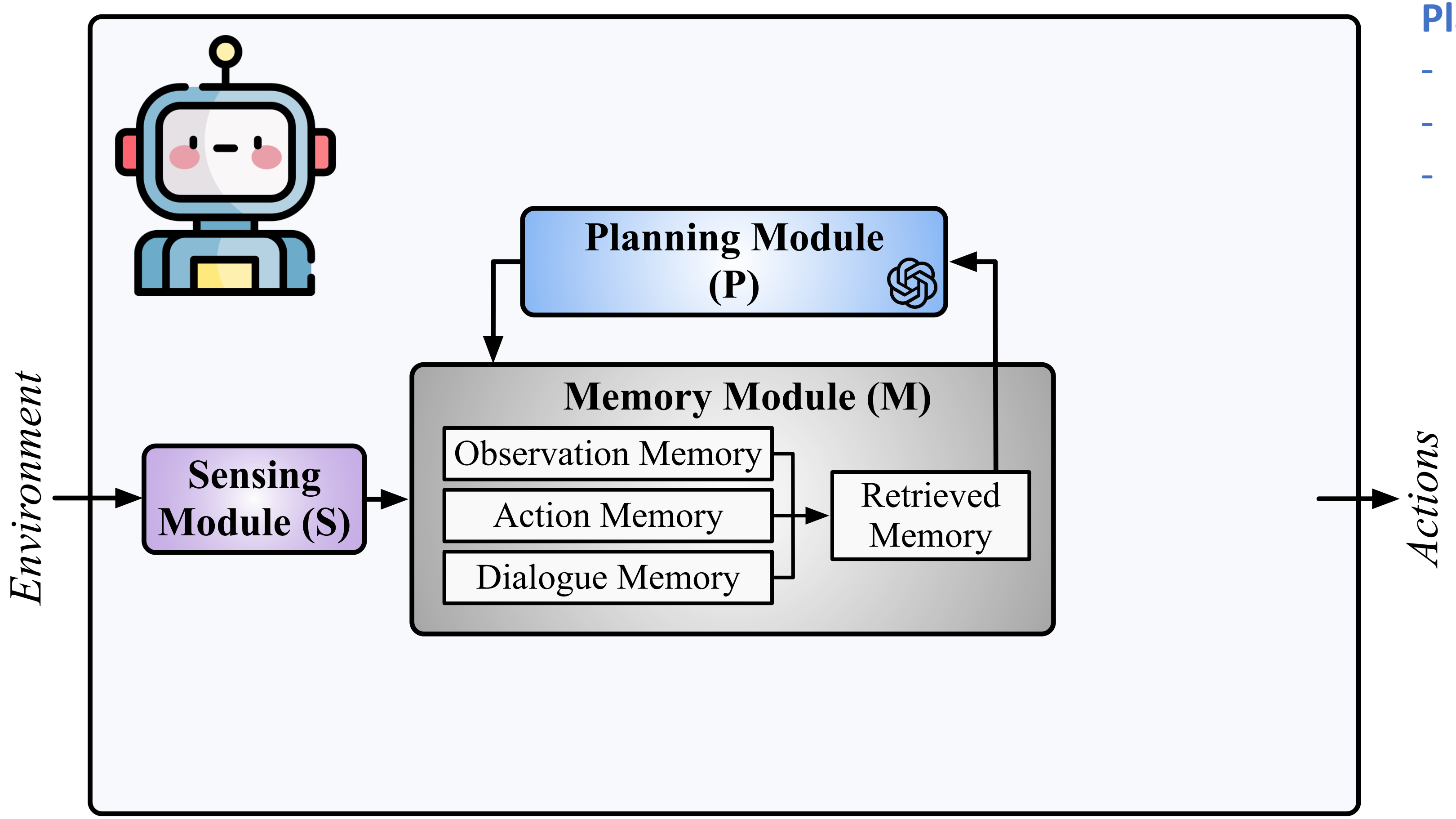
Sensing Module:
- Vision Transformer
- DINO
- Segment Anything
- CLIP
- NeRF
- Gaussian Splatting
- Pointcloud Gen

Embodied Agent System Paradigm



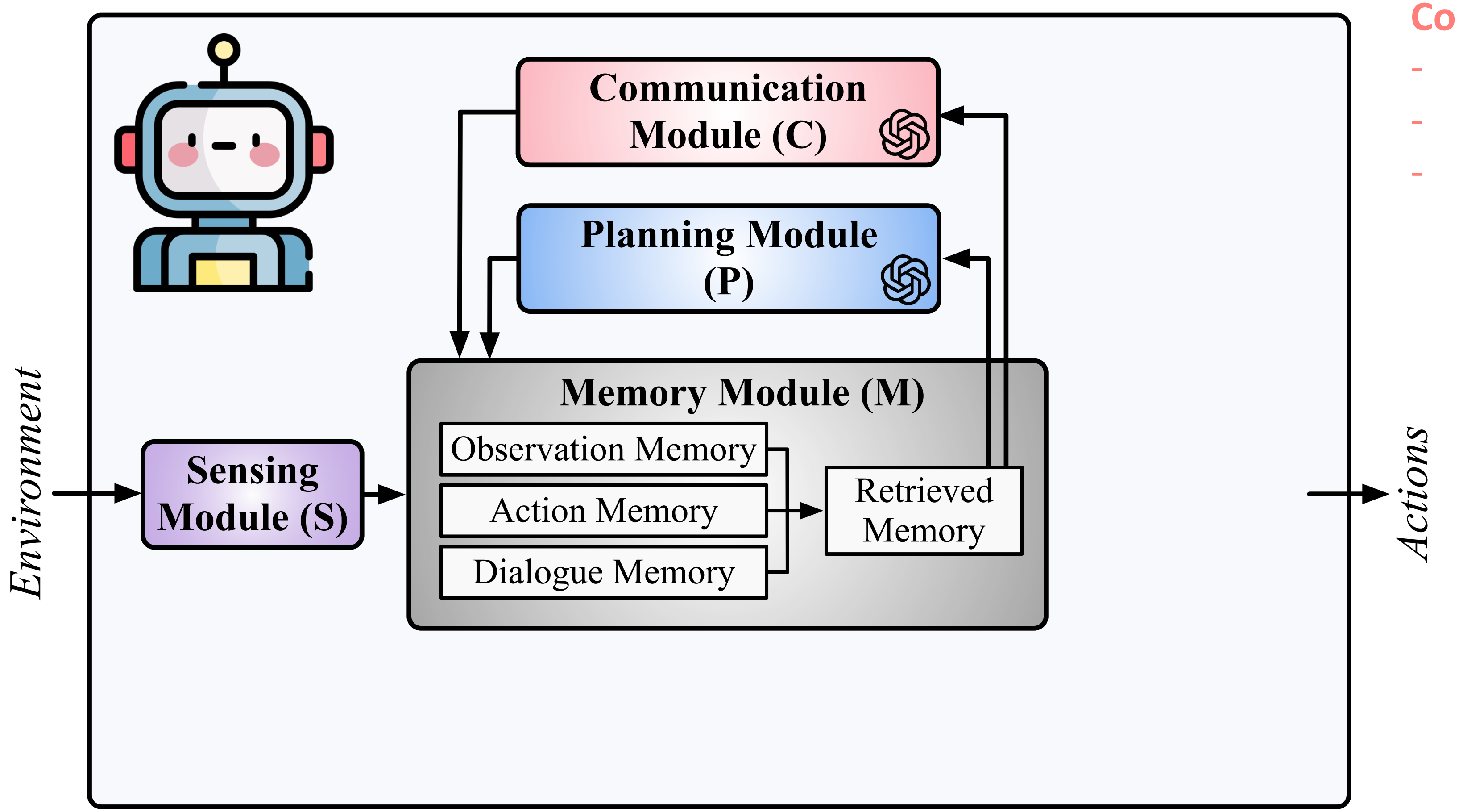
Memory Module:
- Observation memory
- Action memory
- Dialogue memory
- External memory

Embodied Agent System Paradigm



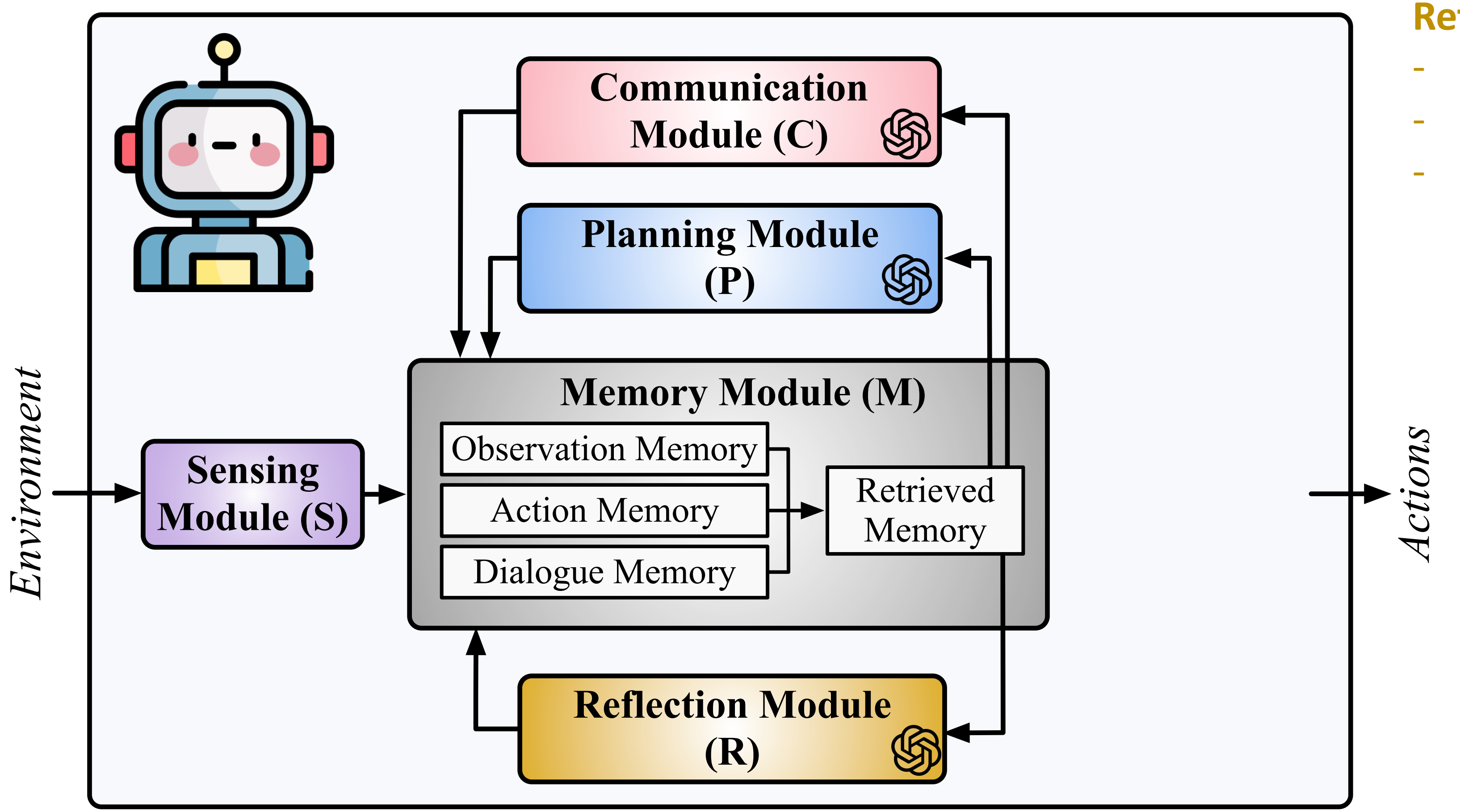
Planning Module:
- LLM
- VLM
- CoT, ToT, GoT, etc

Embodied Agent System Paradigm



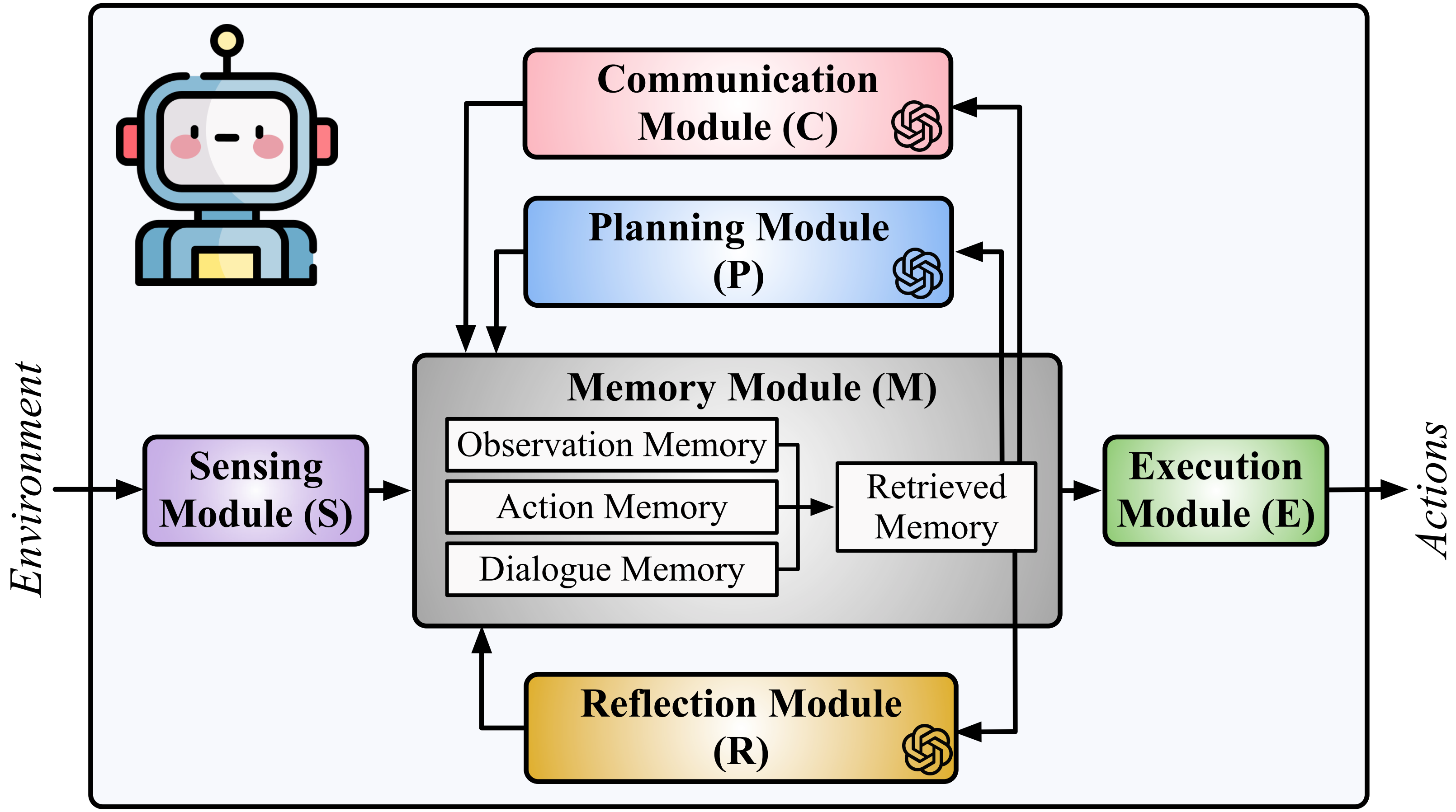
Communication Module:
- LLM
- VLM
- CoT, ToT, GoT, etc

Embodied Agent System Paradigm



Reflection Module:
- LLM
- VLM
- CoT, ToT, GoT, etc

Embodied Agent System Paradigm

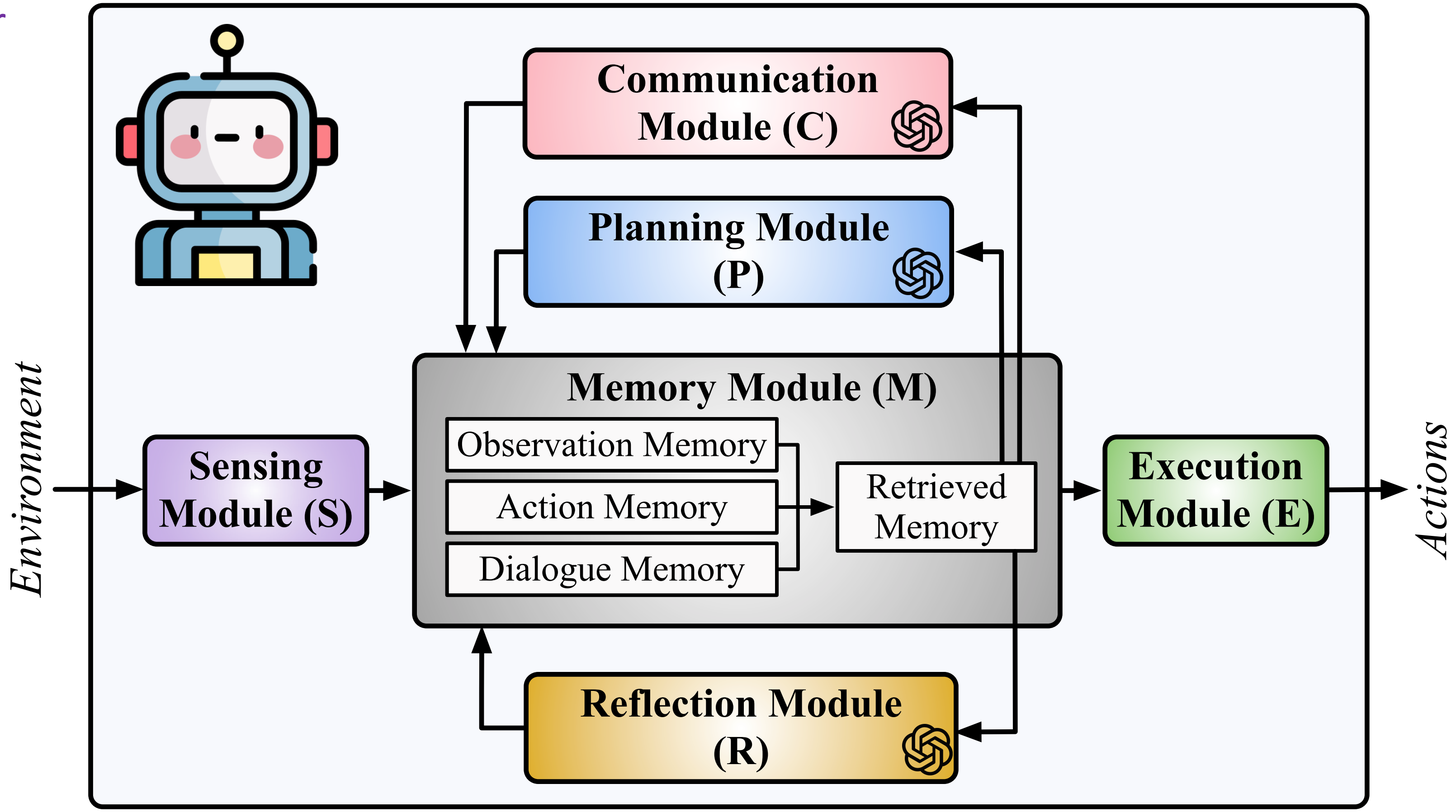


Execution Module:
- A-star planning
- RRT planning
- Factor graph optimization
- Model predictive control (MPC)
- Inverse dynamics control
- QP-based control

Embodied Agent System Paradigm

Sensing Module:
- Vision Transformer
- DINO
- Segment Anything
- CLIP
- NeRF
- Gaussian Splatting
- Pointcloud Gen

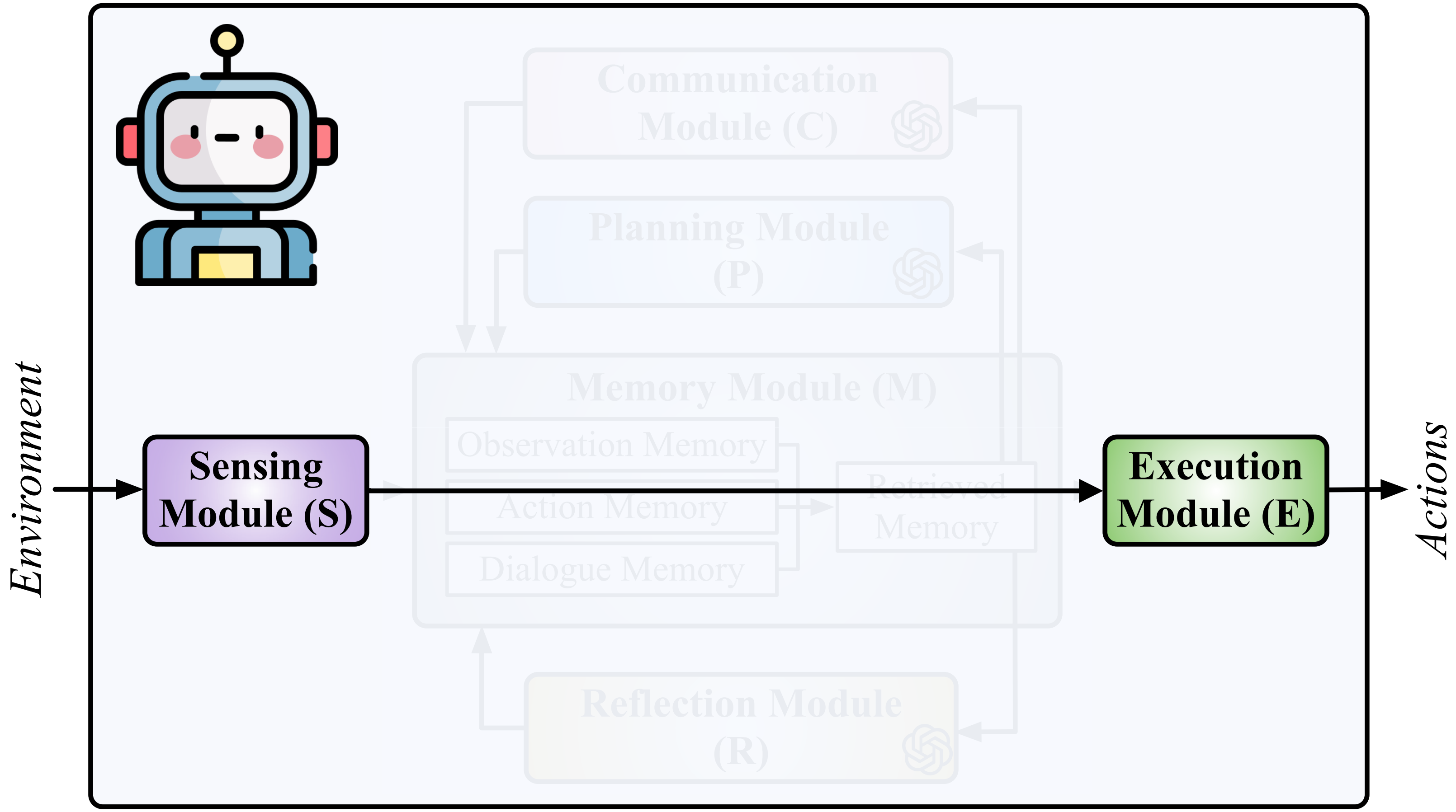
Memory Module:
- Observation mem
- Action mem
- Dialogue mem
- External mem



Planning/Communication/Reflect Module:
- LLM
- VLM
- CoT, ToT, GoT, etc

Execution Module:
- A-star planning
- RRT planning
- Factor graph optimization
- Model predictive control (MPC)
- Inverse dynamics control
- QP-based control

Embodied Agent System Paradigm



System 1

(Autonomy Capability)

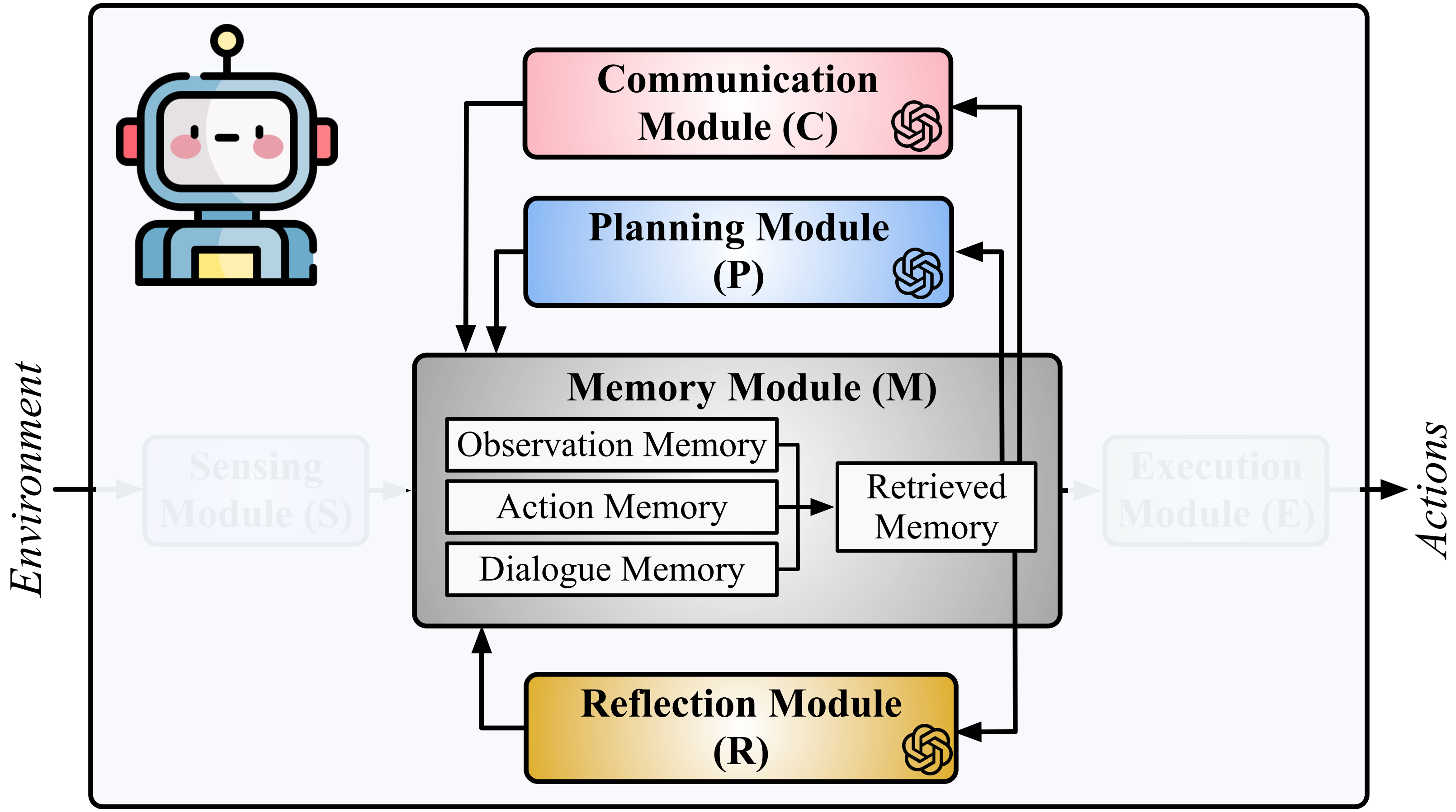
Fast & Intuitive Thinking

Embodied Agent System Paradigm

System 2

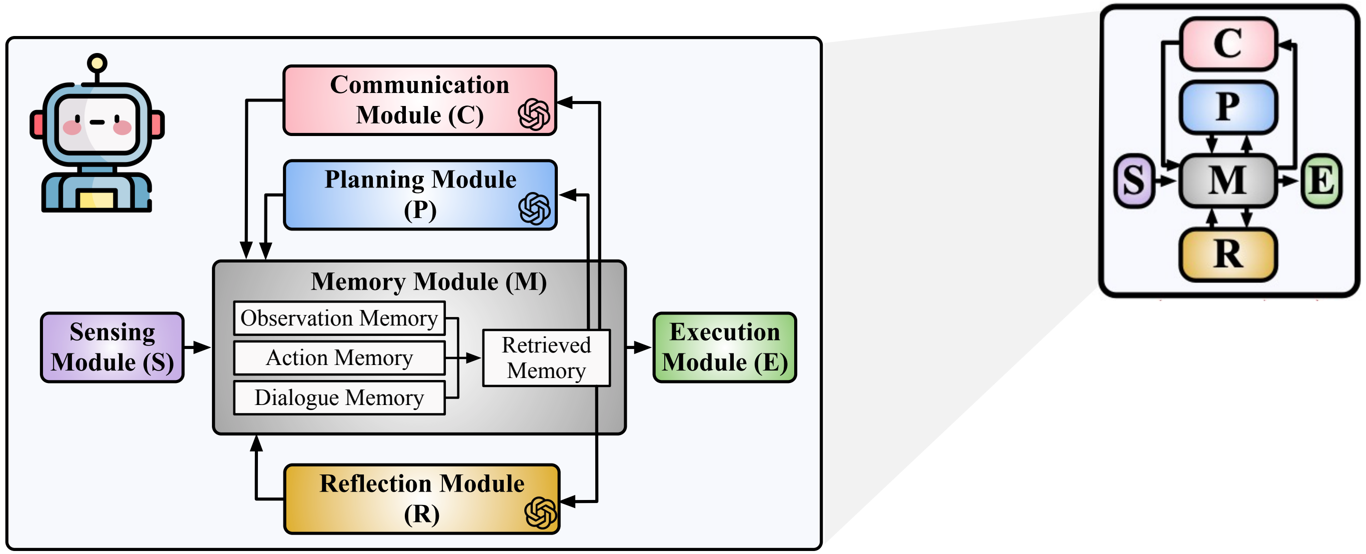
(Cognition Capability)

Slow and Rational Thinking



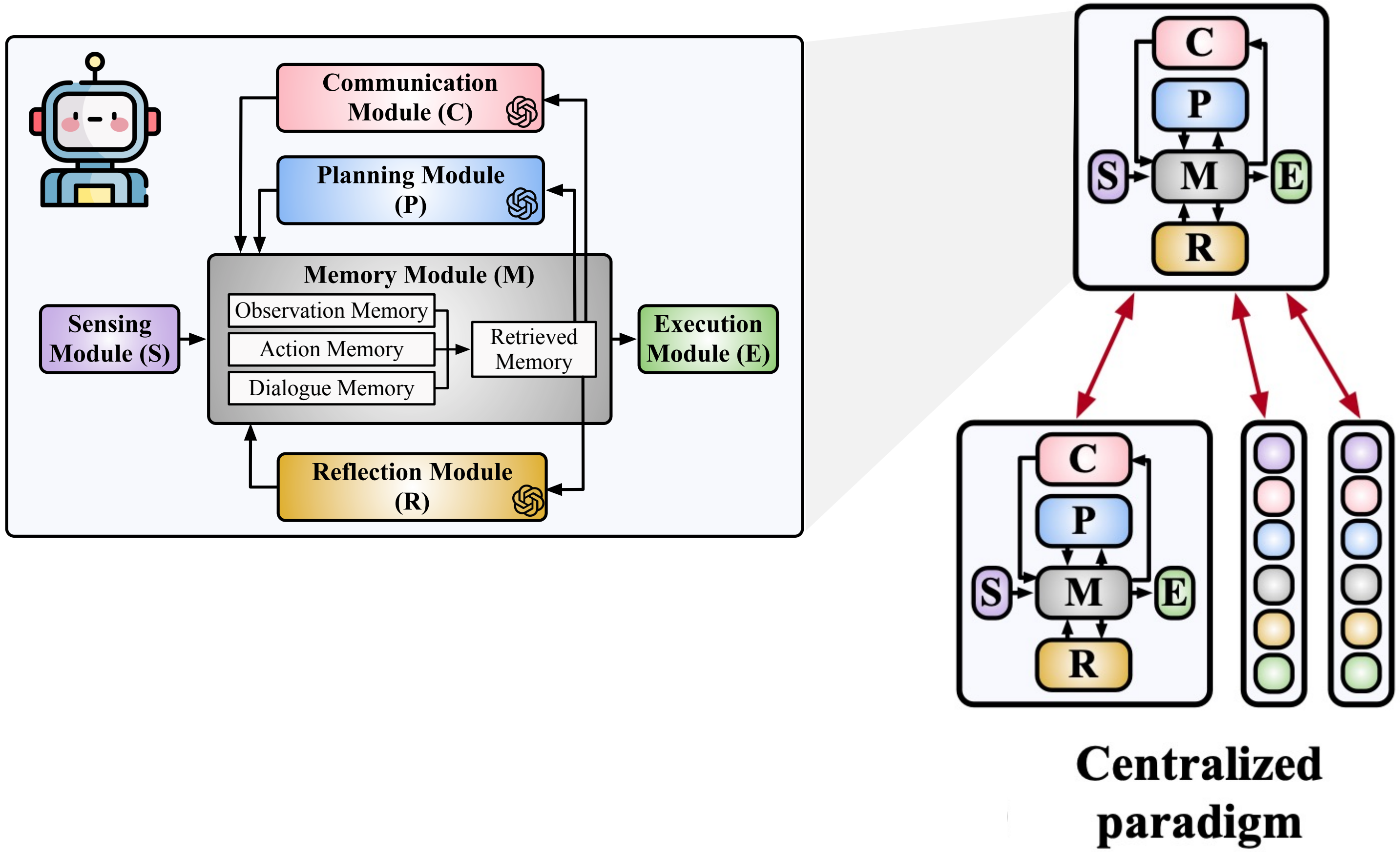
Embodied Agent System Paradigm

Cooperative Embodied AI Systems

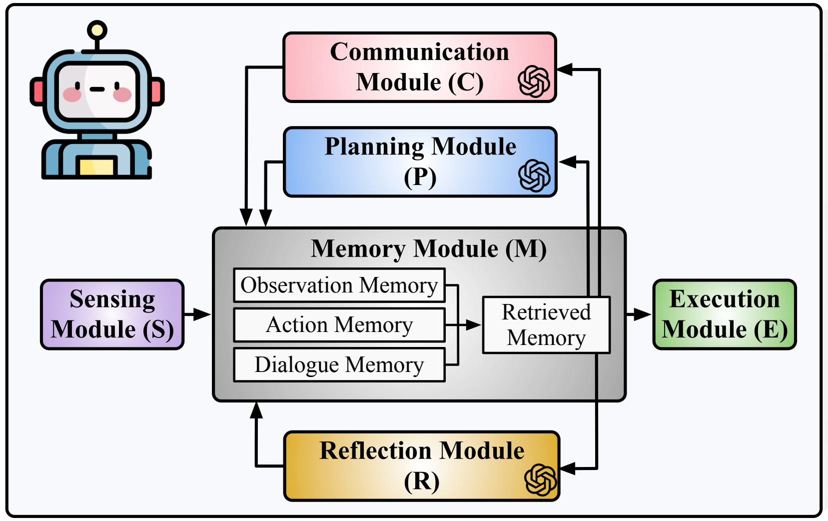


Embodied Agent System Paradigm

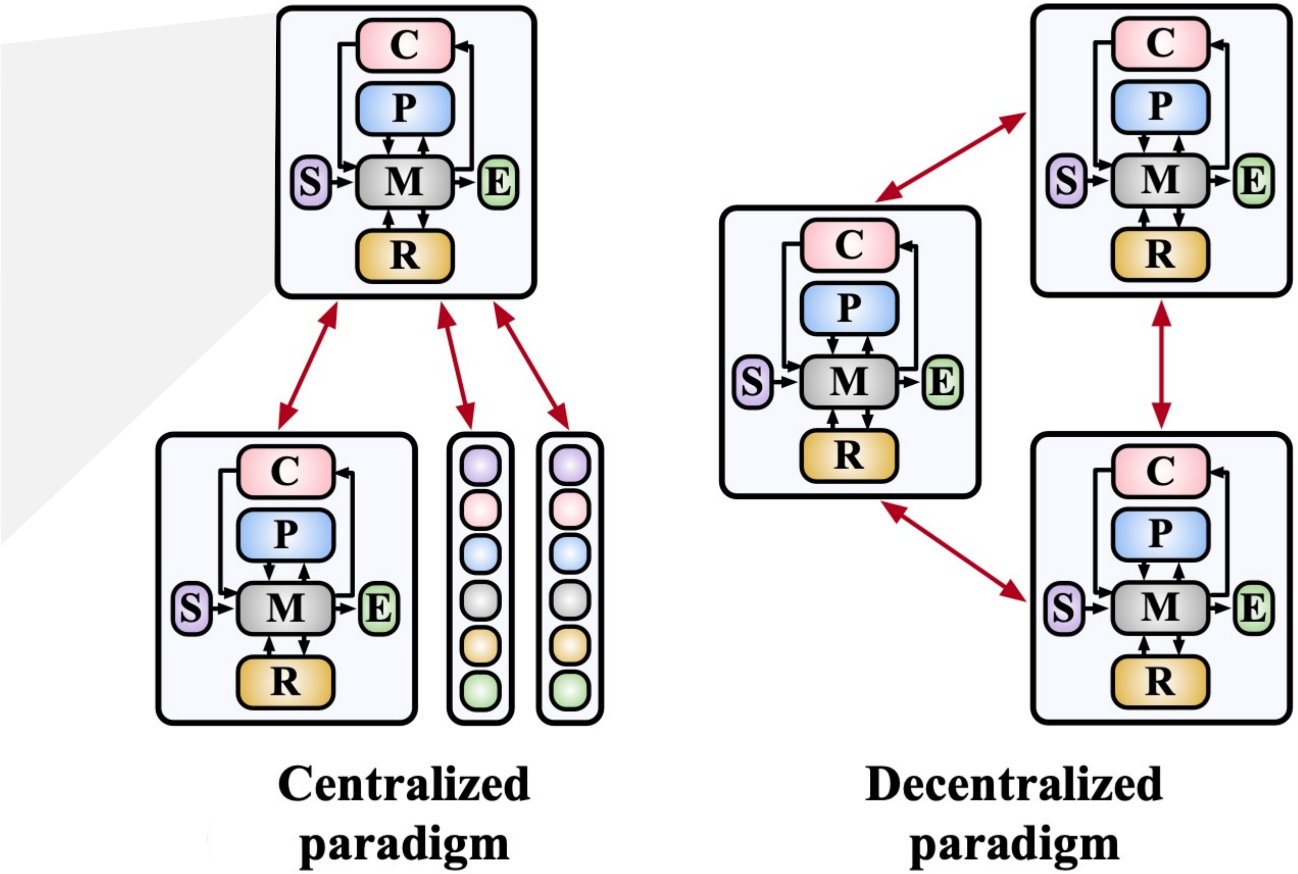
Cooperative Embodied AI Systems



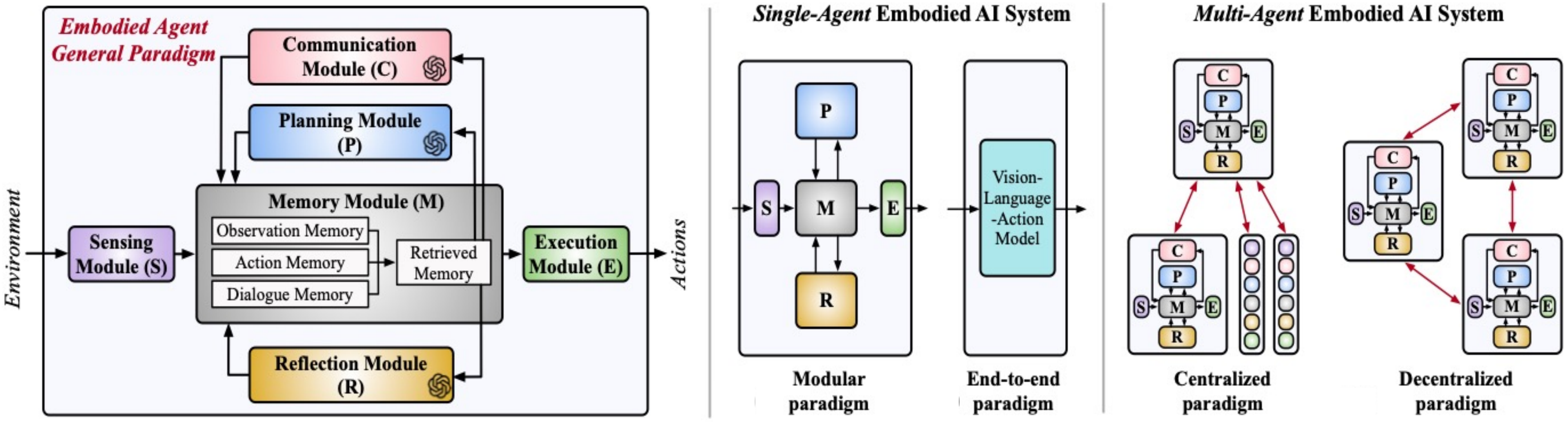
Embodied Agent System Paradigm



Cooperative Embodied AI Systems



Summary - Embodied Agent System Paradigm



Summary - Embodied Agent System Paradigm

System Paradigm		Workloads	Computing Modules					
			Sense	Plan	Comm.	Mem.	Refl.	Exec.
Single-Agent	Modularized Paradigm	Mobile-Agent [33]	✓	✓	✗	✗	✓	✓
		AppAgent [34]	✓	✓	✗	✗	✗	✓
		PDDL [13]	✗	✓	✗	✗	✓	✗
		RoboGPT [14]	✓	✓	✗	✗	✗	✓
		VOYAGER [35]	✗	✓	✗	✓	✓	✓
		MP5 [36]	✓	✓	✗	✗	✓	✓
		RILA [37]	✓	✓	✗	✓	✓	✓
		CRADLE [25]	✓	✓	✗	✓	✓	✓
		STEVE [38]	✓	✓	✗	✗	✗	✓
		DEPS [15]	✓	✓	✗	✗	✓	✓
		JARVIS-1 [24]	✓	✓	✗	✓	✓	✓
		FILM [9]	✓	✓	✗	✗	✗	✓
		LLM-Planner [23]	✗	✓	✗	✗	✓	✓
		EmbodiedGPT [39]	✓	✓	✗	✗	✗	✓
		Dadu-E [40]	✓	✓	✗	✓	✓	✓
		MINEDOJO [41]	✓	✓	✗	✓	✗	✓
		Luban [42]	✓	✓	✗	✓	✓	✓
		MetaGPT [43]	✗	✓	✓	✓	✓	✓
		Mobile-Agent-V2 [44]	✓	✓	✗	✓	✓	✓
	End-to-End Paradigm	RT-2 [45]	Vision-Language-Action Model					
		RoboVLMs [46]	Vision-Language-Action Model					
		GAIA-1 [47]	Generative World Model					
		3D-VLA [48]	3D Vision-Language-Action Model					
		Octo [49]	Vision-Language Model + Exec Policy					
		Diffusion Policy [50]	Diffusion Policy					

System Paradigm		Workloads	Computing Modules					
			Sense	Plan	Comm.	Mem.	Refl.	Exec.
Multi-Agent	Centralized Paradigm	LLaMAC [51]	✗	✓	✓	✓	✗	✓
		MindAgent [6]	✗	✓	✓	✓	✗	✓
		OLA [21]	✗	✓	✓	✓	✓	✓
		ALGPT [52]	✓	✓	✓	✓	✗	✓
		CMAS [20]	✓	✓	✓	✓	✗	✓
		ReAd [53]	✗	✓	✓	✗	✓	✓
		Co-NavGPT [54]	✓	✓	✓	✗	✗	✓
		COHERENT [28]	✓	✓	✓	✓	✓	✓
	Decentralized Paradigm	DMAS [20]	✓	✓	✓	✓	✗	✓
		HMAS [20]	✓	✓	✓	✓	✓	✓
		AGA [55]	✓	✓	✓	✓	✓	✓
		CoELA [4]	✓	✓	✓	✓	✗	✓
		FMA [56]	✗	✓	✓	✓	✓	✓
		COMBO [4]	✓	✓	✓	✓	✗	✓
		RoCo [27]	✓	✓	✓	✓	✓	✓
		AgentVerse [57]	✗	✓	✓	✗	✗	✓
		KoMA [58]	✗	✓	✓	✓	✓	✓

Please refer to paper for more details

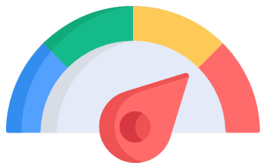
Research Question:

What are the **system characteristics** and **sources of inefficiencies** in these embodied systems?

Outline

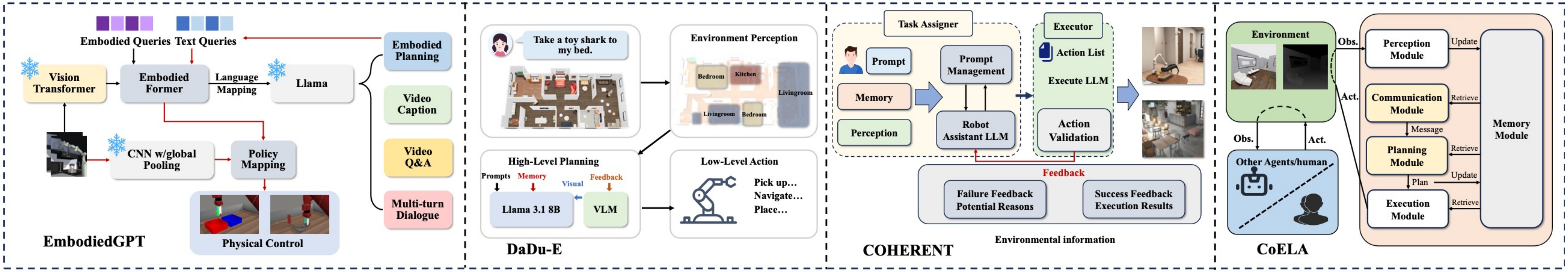
- *Understand* fundamental **building blocks** and **paradigms** of embodied systems.

- *Identify* **system characteristics** and **sources of inefficiency** of embodied systems.



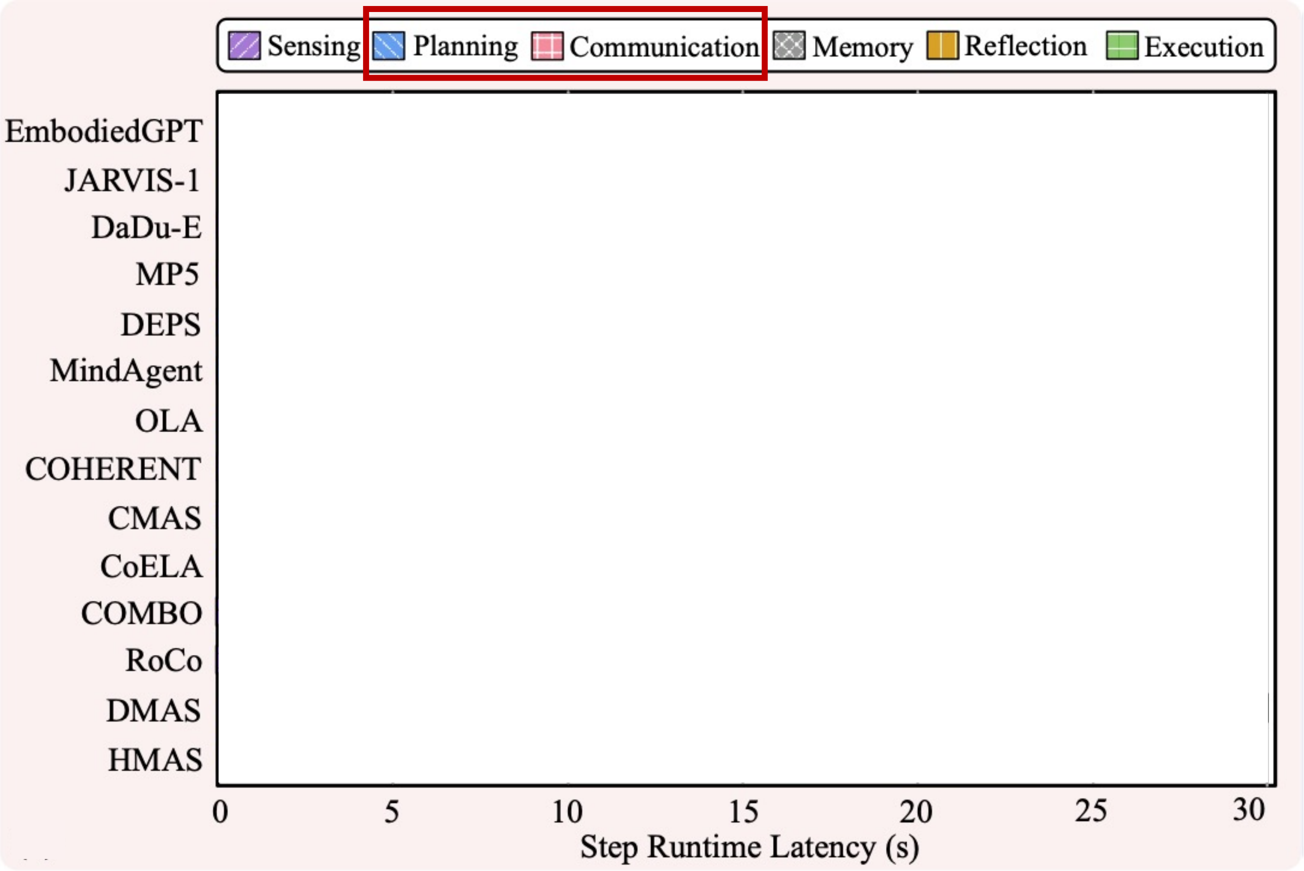
- *Demonstrate* **optimization opportunities** and **scalability-efficiency improvements** for embodied systems.

Representative Embodied Agent Workloads



Embodied AI Systems	System Module						Application	Datasets and Tasks
	Sensing	Planning	Communication	Memory	Reflection	Execution		
EmbodiedGPT [39]	ViT	Llama-7B	–	–	–	MLP	Embodied planning, visual captioning, VQA	Franka Kitchen [59], Meta-World [60], VirtualHome [61]
JARVIS-1 [24]	MineCLIP	GPT-4/Llama-13B	–	Ob., Act.	Llama-13B	Action list	Embodied planning (e.g, obtain diamond pickaxe)	Minecraft [62]
DaDu-E [40]	PointCloud	Llama-8B	–	Ob., Act.	LLaVA-8B	AnyGrasp	Object transport, Autonomous decision-making	Self-designed four-level tasks
MP5 [36]	MineCLIP	GPT-4	–	–	GPT-4	MineDojo	Object transport, Situation-aware long-term planning	Minecraft [62]
DEPS [15]	Symbolic info	GPT-4	–	–	CLIP	MineDojo	Embodied planning (e.g, obtain diamond pickaxe)	Minecraft [62], MineRL [63], ALFWorld [64]
MindAgent [6]	–	GPT-4	GPT-4	Ob., Act., Dx.	–	Action list	Collaborative planning, gaming, housework	CuisineWorld [6], Minecraft [62]
OLA [21]	–	GPT-4/Llama-70B	GPT-4	Ob., Act., Dx.	GPT-4	Action list	Collaborative planning, object transport	VirtualHome [61], C-WAH [65]
COHERENT [28]	DINO	GPT-4	GPT-4	Ob., Act., Dx.	GPT-4	RRT/A-star	Collaborative planning, Robot arm manipulation	BEHAVIOR-1K [66]
CMAS [20]	ViLD	GPT-4	GPT-4	Ob., Act., Dx.	–	Action list	Collaborative planning, manipulator, object transport	BoxNet1, BoxNet2, WareHouse, BoxLift [20]
CoELA [4]	Mask R-CNN	GPT-4	GPT-4	Ob., Act., Dx.	–	A-star	Collaborative object transporting, housework	TDW-MAT [67], C-WAH [65]
COMBO [5]	Diffusion	LLaVA-7B	LLaVA-7B	Ob., Act., Dx.	–	A-star	Collaborative gaming, housework	TDW-Game [68], TDW-Cook [68]
RoCo [27]	ViT	GPT-4	GPT-4	Ob., Act., Dx.	GPT-4	RRT	Robot arm motion planning, manipulation	RoCoBench [27]
DMAS [20]	ViLD	GPT-4	GPT-4	Ob., Act., Dx.	–	Action list	Collaborative planning, manipulator, object transport	BoxNet1, BoxNet2, WareHouse, BoxLift [20]
HMAS [20]	ViLD	GPT-4	GPT-4	Ob., Act., Dx.	GPT-4	Action list	Collaborative planning, manipulator, object transport	BoxNet1, BoxNet2, WareHouse, BoxLift [20]

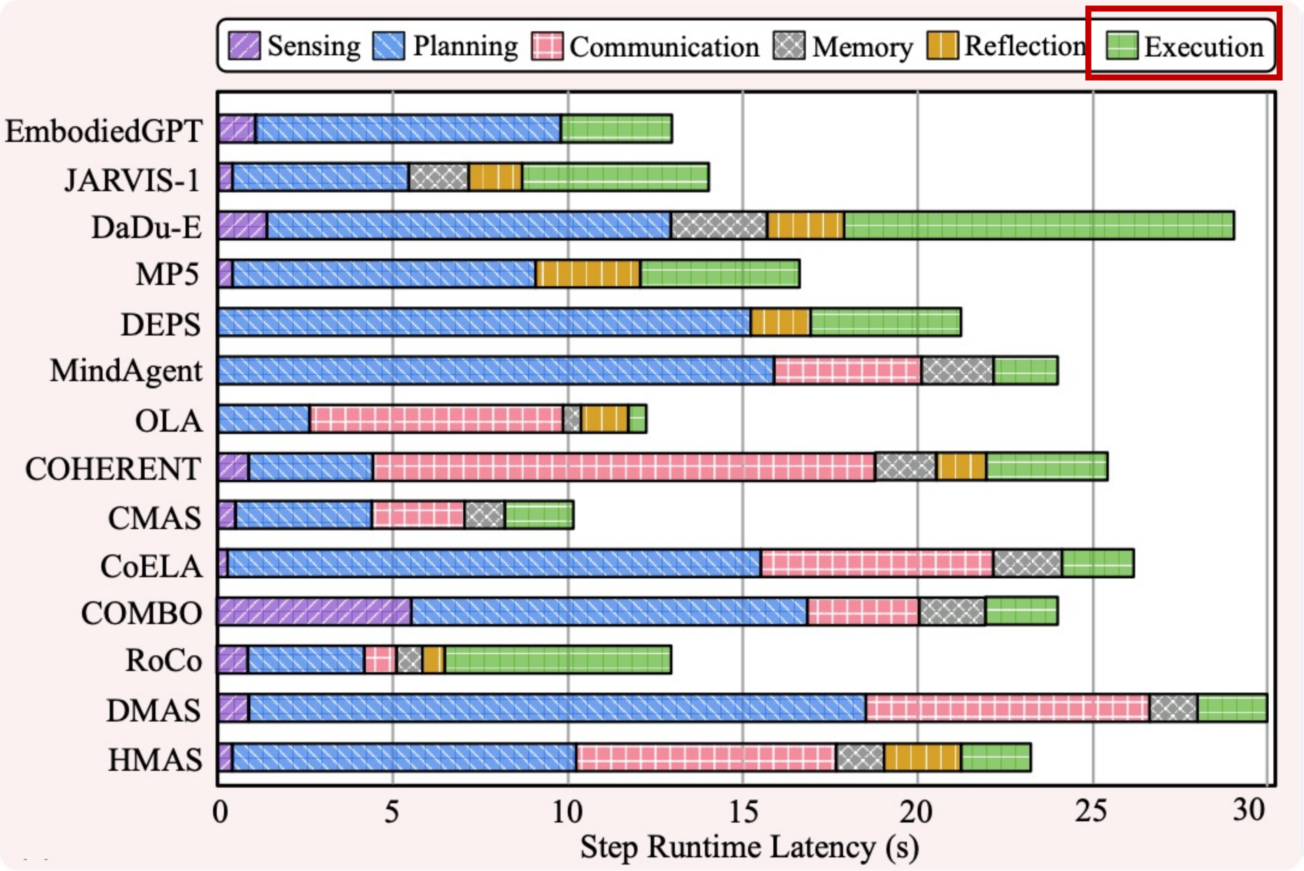
Latency Characterization



Takeaway:

- End-to-end latency in long-horizon embodied tasks is significant.
- LLM-based planning and communication dominate the latency due to repeated runs.

Latency Characterization



Takeaway:

- End-to-end latency in long-horizon embodied tasks is significant.
- LLM-based planning and communication dominate the latency due to repeated runs.
- Low-level planning and execution also contribute notable delays due to multiple executions and computational complexity.

Latency Characterization

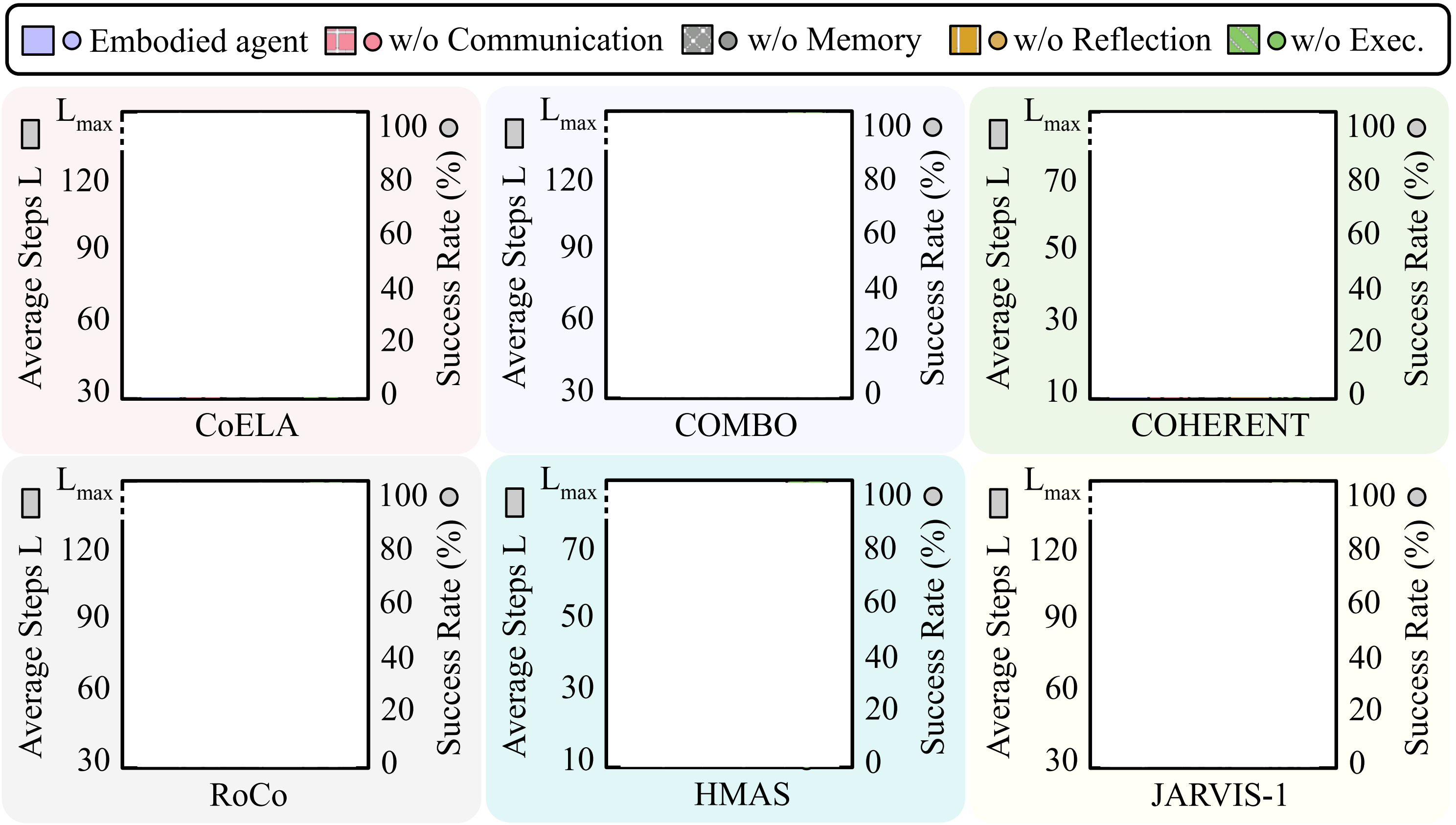
Takeaway:

- End-to-end latency in long-horizon embodied tasks is significant.
- LLM-based planning and communication dominate the latency due to repeated runs.
- Low-level planning and execution also contribute notable delays due to multiple executions and computational complexity.

Optimization Recommendation:

- The long latency of high-level planning and communication can be optimized through efficient LLM deployment, such as batching, quantization, lightweight models.
- The inefficiency of low-level planning and execution can be optimized via optimized data structure, memory access pattern, parallelism, domain-specific architecture.

Module Sensitivity Characterization



Takeaway:

- Memory and reflection modules are critical for task efficiency, tracking agent status and task success.
- Low-level execution module plays an indispensable role in system functionality.

Module Sensitivity Characterization

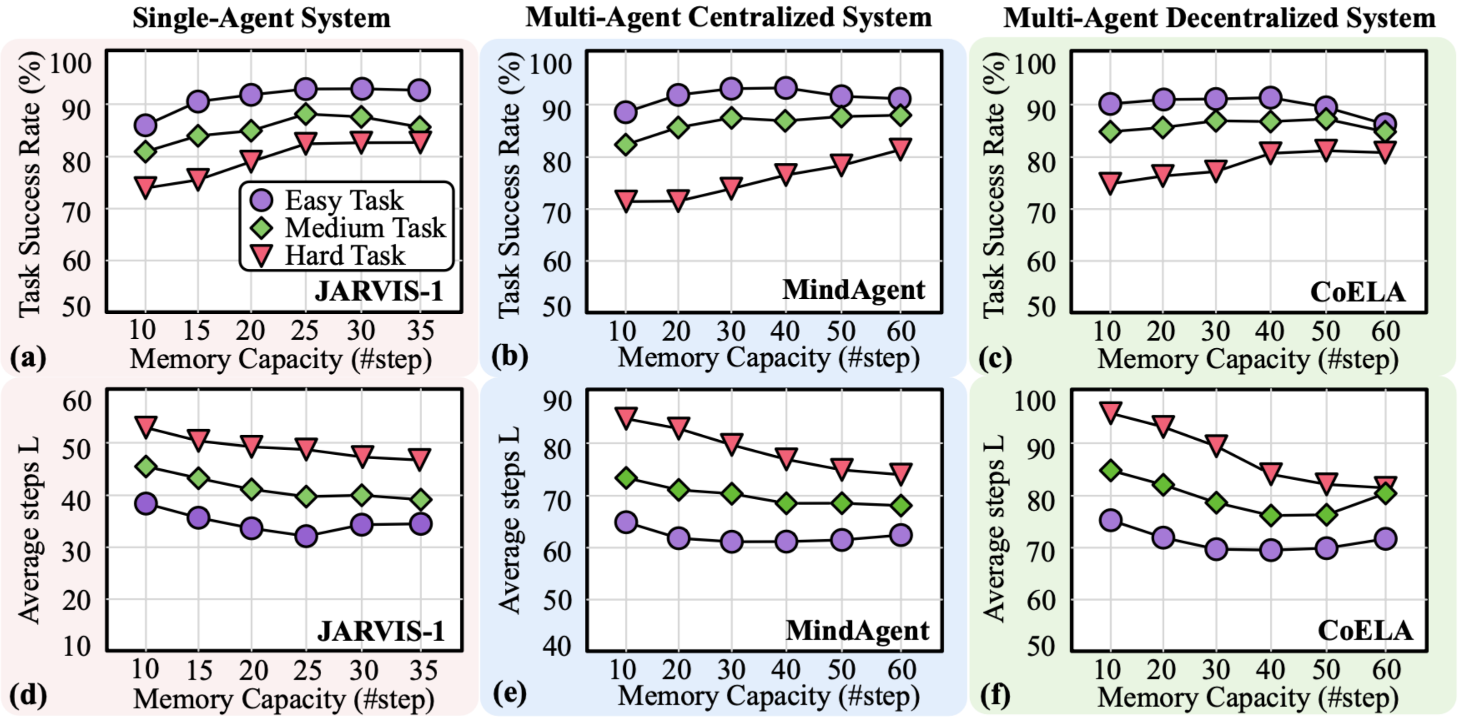
Takeaway:

- Memory and reflection modules are critical for task efficiency, tracking agent status and task success.
- Low-level execution module plays an indispensable role in system functionality.

Optimization Recommendation:

- System can be optimized by improving communication efficiency, enhancing memory through context summarization, and strengthening reflection with error correction.
- Offloading low-level execution to specialized controllers and adopting a hybrid planning framework can further boost task efficiency.

Memory Characterization



Takeaway:

- Increasing memory module capacity improves success rates and reduces #steps, especially for complex tasks.
- However, excessively large memory introduces inconsistencies and increases retrieval time per step.

Memory Characterization

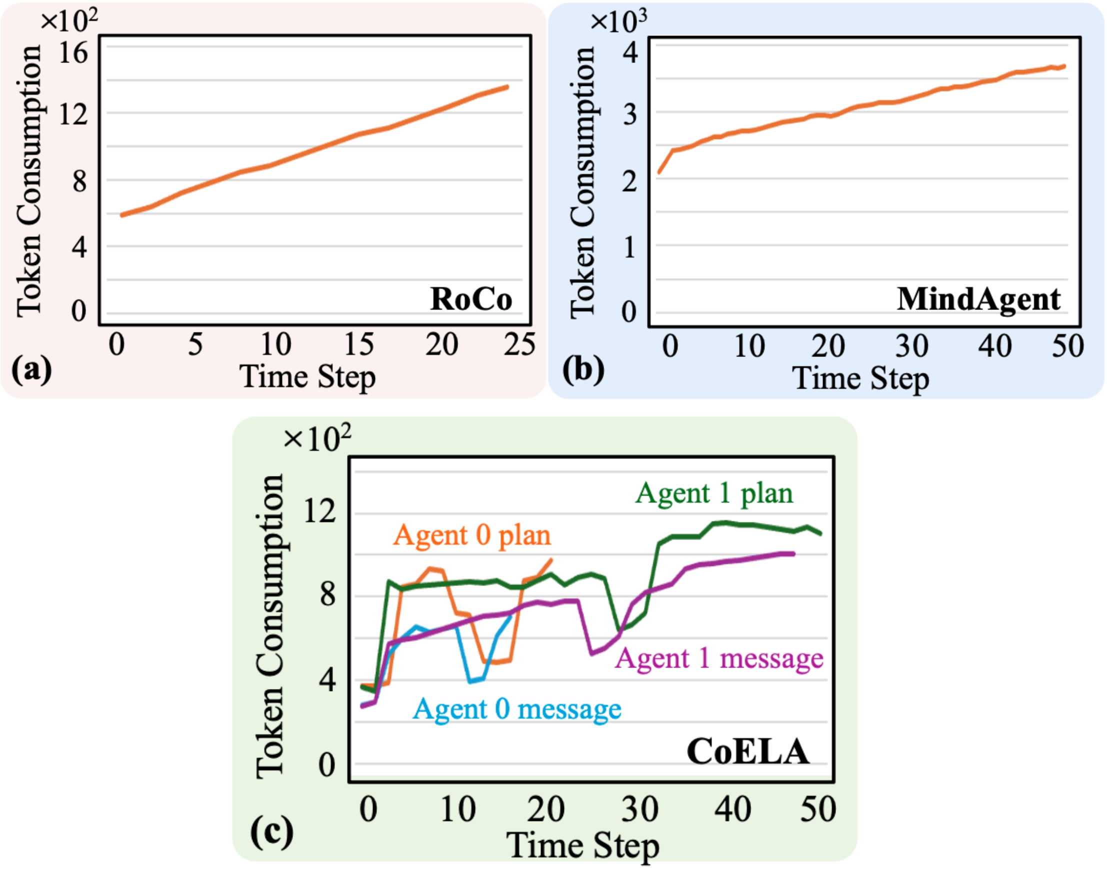
Takeaway:

- Increasing memory module capacity improves success rates and reduces #steps, especially for complex tasks.
- However, excessively large memory introduces inconsistencies and increases retrieval time per step.

Optimization Recommendation:

- The memory module overhead and inconsistency can be optimized with a dual memory structure:
    - Long-term memory stores static environmental information;
    - Short-term memory captures real-time updates on agent status, task progress, and interactions.

Token Length Characterization



Takeaway:

- Token length increases as tasks progress, driven by repeated information retrieval and concatenated dialogues, leading to higher computational costs and efficiency degradation.

Token Length Characterization

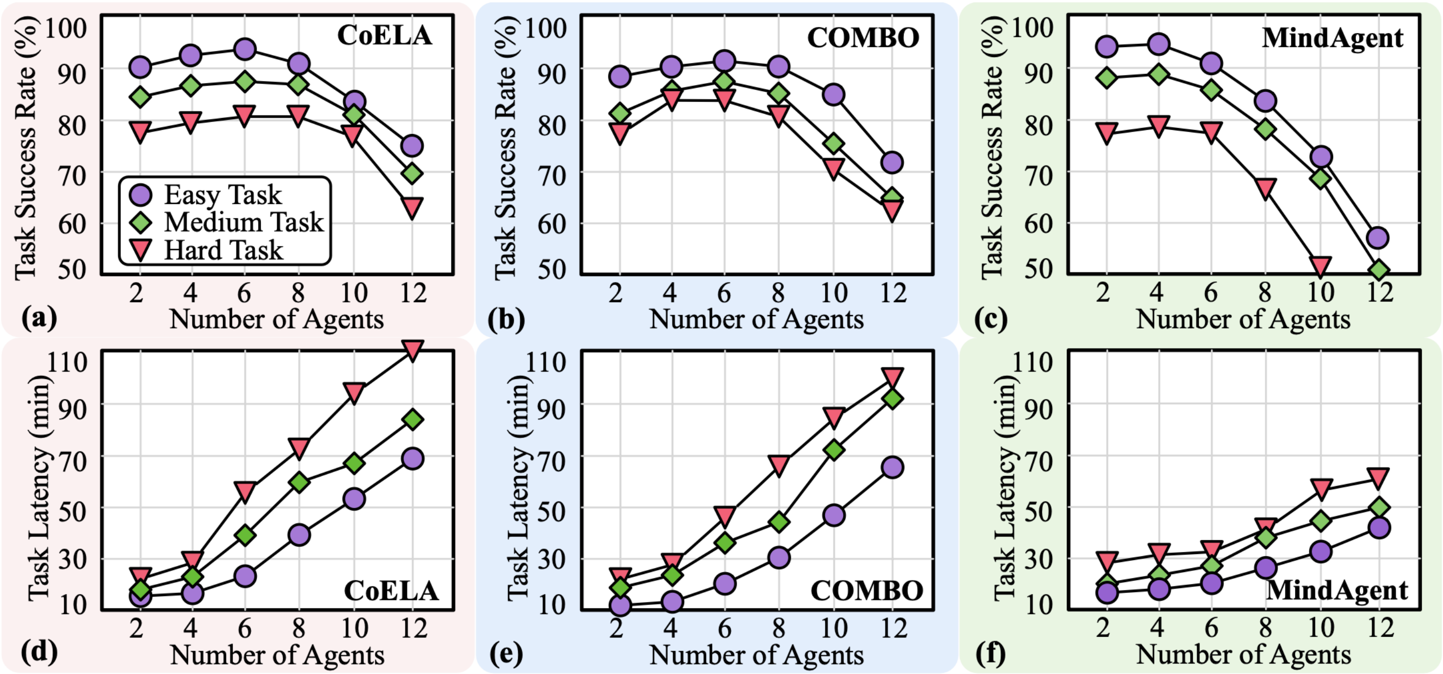
Takeaway:

- Token length increases as tasks progress, driven by repeated information retrieval and concatenated dialogues, leading to higher computational costs and efficiency degradation.

Optimization Recommendation:

- Token length inefficiency can be optimized through context-aware management and compression techniques, such as summarizing dialogue history, removing irrelevant information, and compressing repeated patterns to keep the LLM context both efficient and relevant.

Scalability Characterization



Decentralized | Decentralized | Centralized

Takeaway:

- Multi-agent embodied systems face scalability challenges as the number of agents increases.
- Centralized vs. decentralized:
  - Centralized systems: success rate challenge
  - Decentralized systems: latency challenge

Scalability Characterization

Takeaway:

- Multi-agent embodied systems face scalability challenges as the number of agents increases.
- Centralized vs. decentralized:
  - Centralized systems: success rate challenge
  - Decentralized systems: latency challenge

Optimization Recommendation:

- The scalability challenges of multi-agent embodied systems can be optimized through hierarchical cooperative paradigm:
  - Agents are grouped into clusters when close enough, cooperating centrally within clusters and decentrally across clusters.

Research Question:

How to enhance the **efficiency and scalability** of cooperative embodied systems?

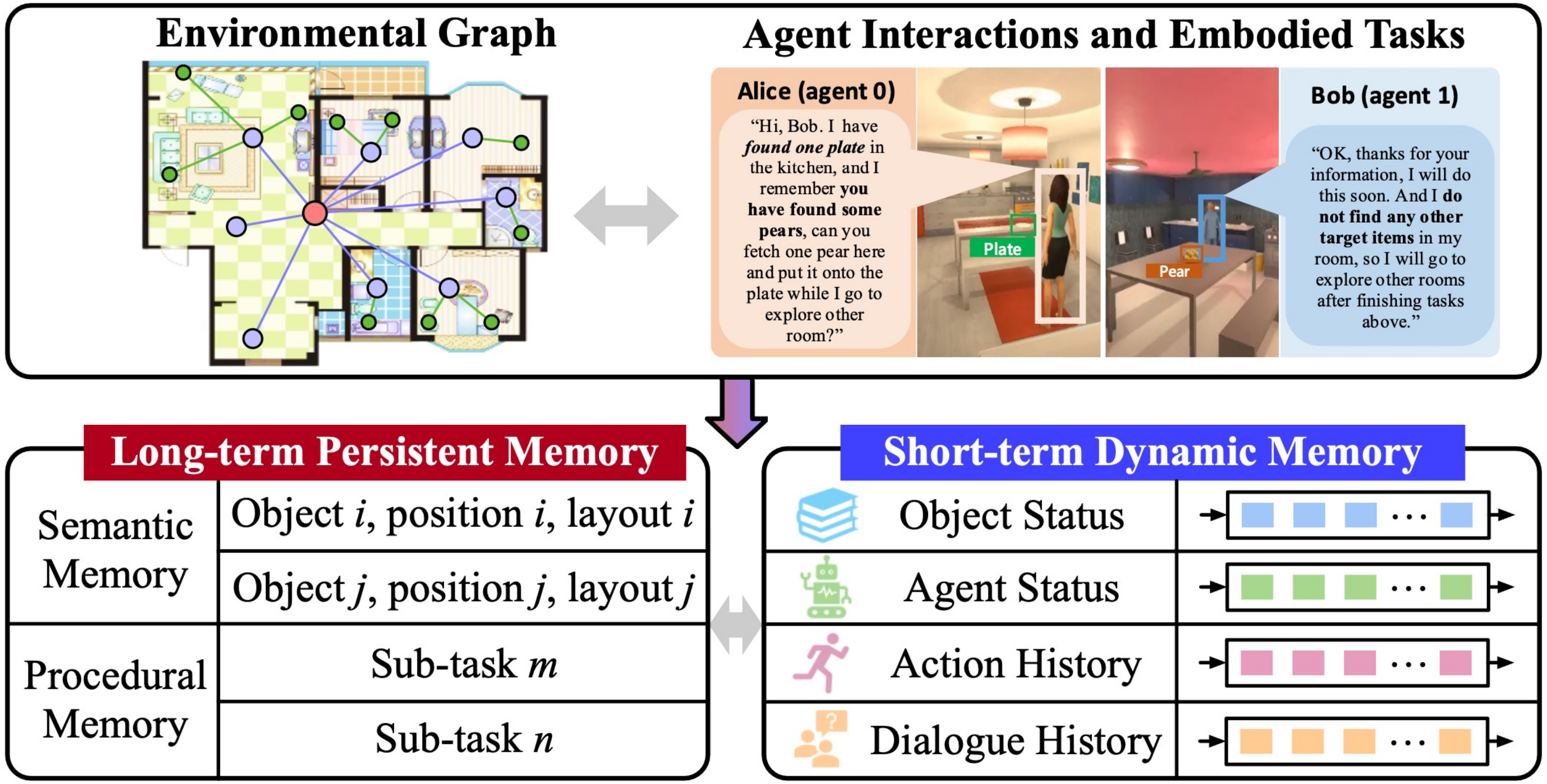
Outline

- *Understand* fundamental **building blocks** and **paradigms** of embodied systems.

- *Identify* **system characteristics** and **sources of inefficiency** of embodied systems.

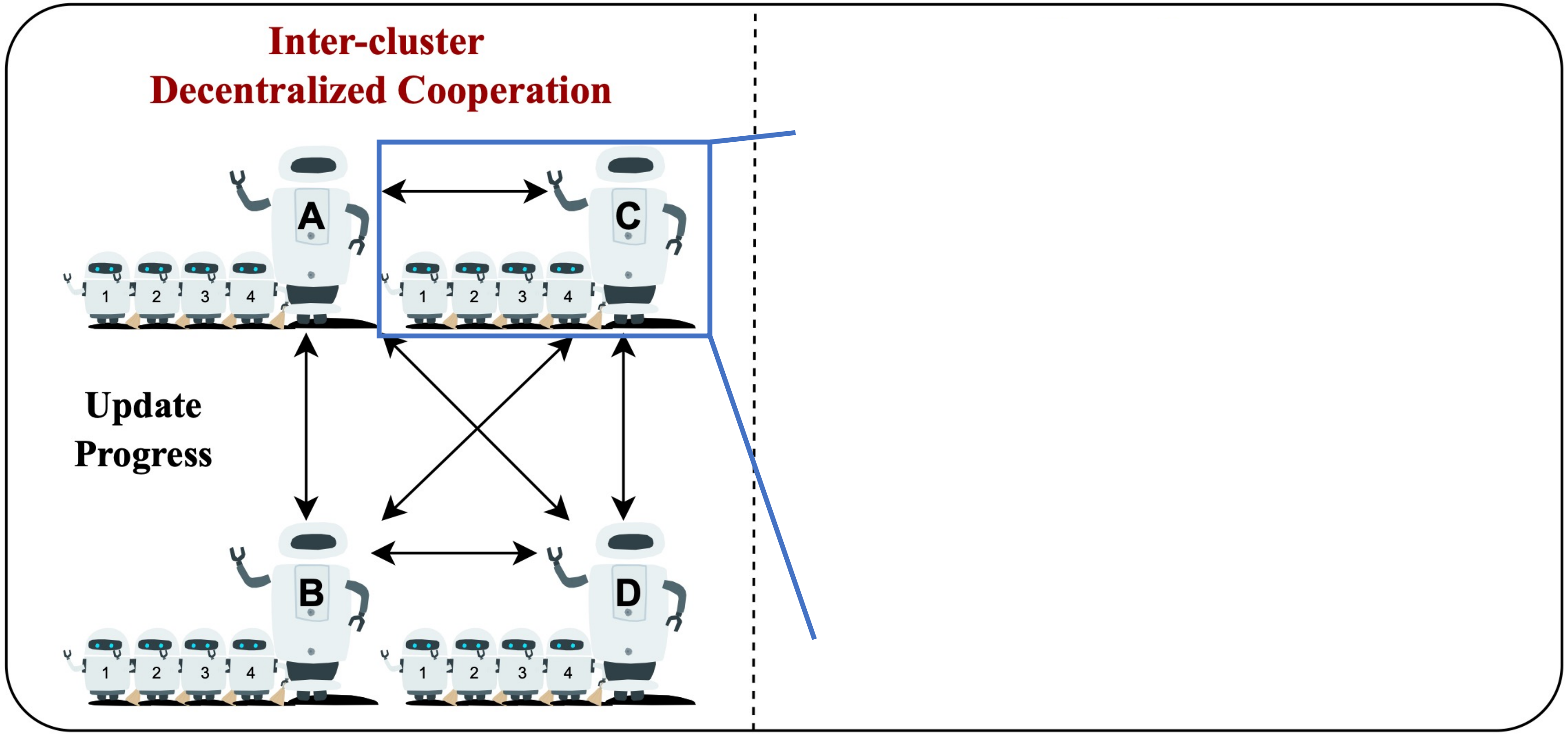
- *Demonstrate* **optimization opportunities** and **scalability-efficiency improvements** for embodied systems.

Memory Optimization – Dual Memory Structure



- Dual-memory structure for agentic systems:
  - Long-term memory: subtask and environment info
  - Short-term memory: action, dialog, agent history (periodically update)

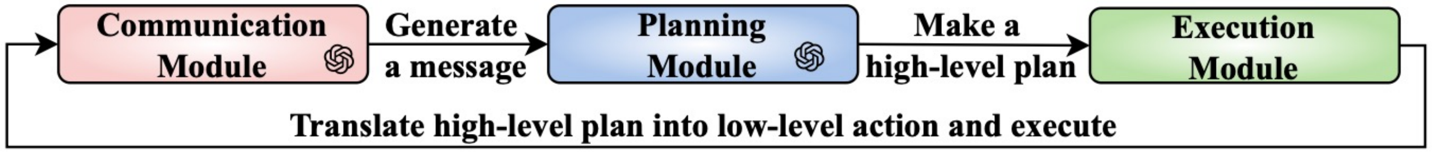
Scalability Optimization - Hierarchical Coop. Planning



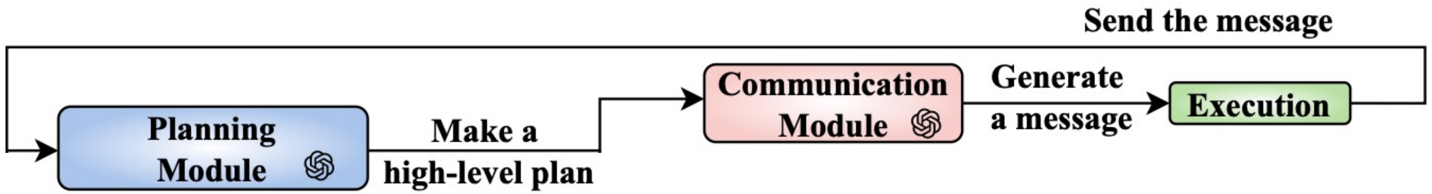
- ❑ Hierarchical cooperative planning for agentic systems:
  - ❑ Inter-cluster decentralized cooperation
  - ❑ Intra-cluster centralized cooperation

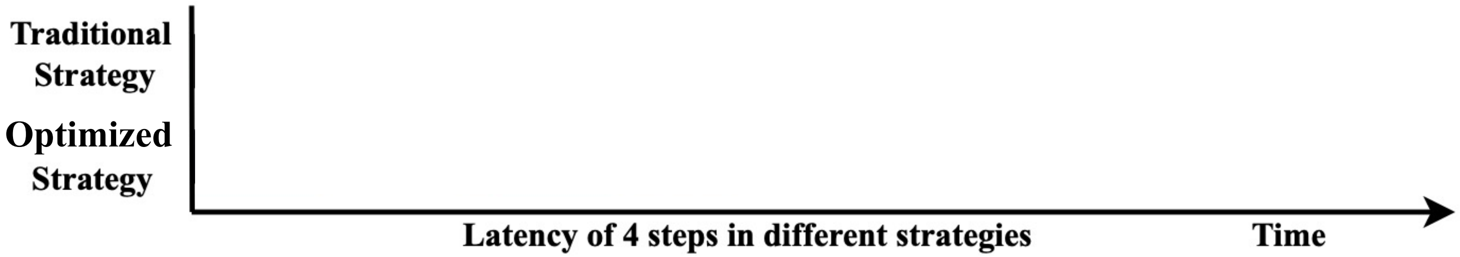
System Optimization – Execution Pipeline

Baseline embodied system pipeline



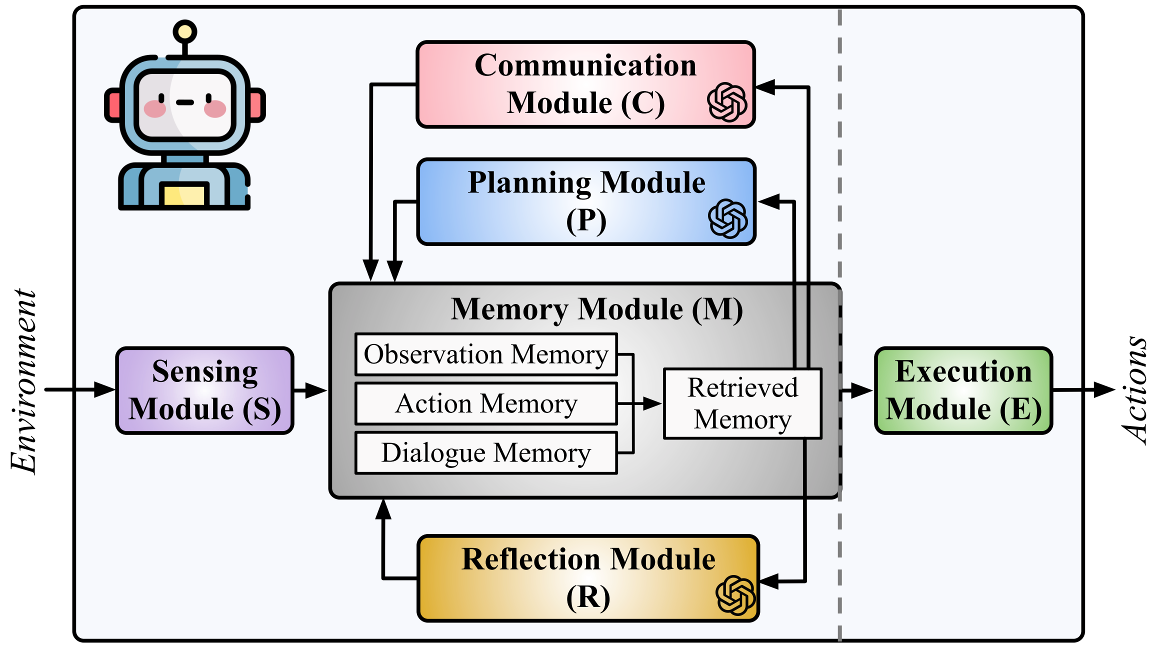
Optimized embodied system pipeline

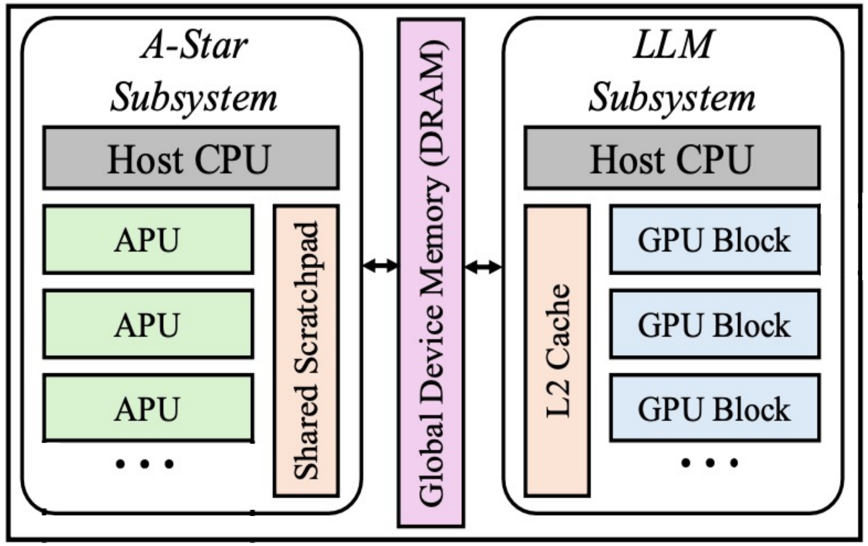




- ❑ Efficient execution pipeline
  - ❑ Planning-then-communication strategy
  - ❑ Planning-guided multi-step execution

Hardware Optimization – Heterogenous SoC





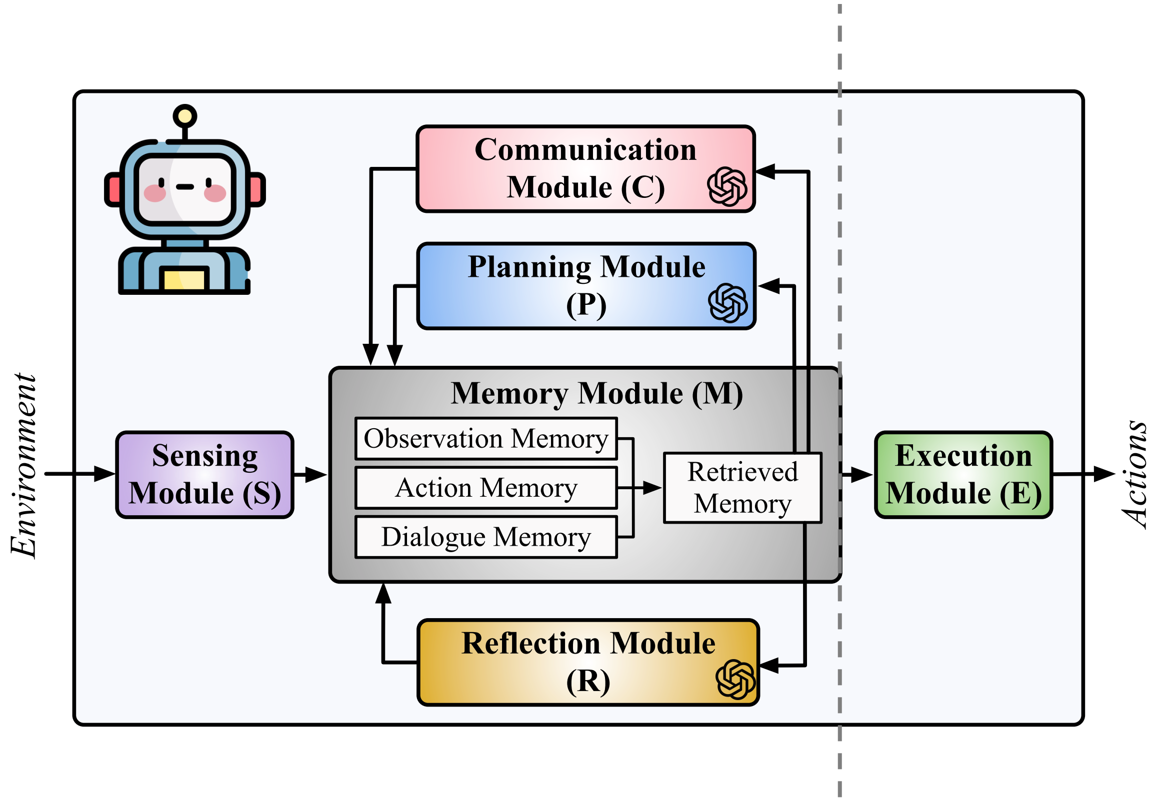
High-level reasoning (System 2)

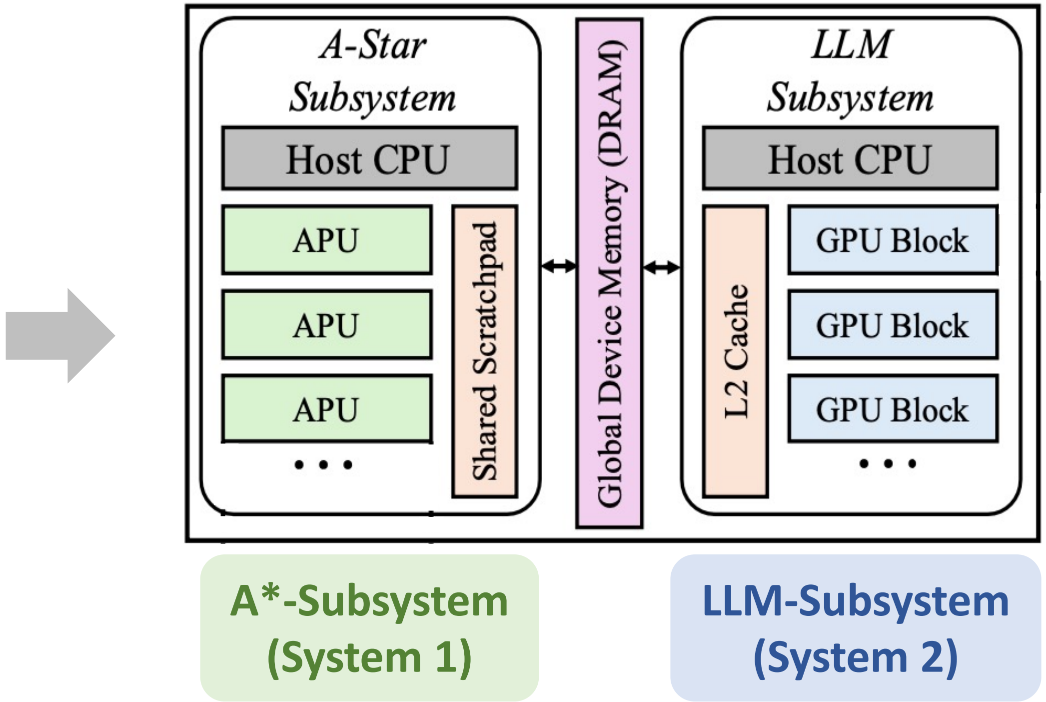
Low-level planning (System 1)

A*-Subsystem (System 1)

LLM-Subsystem (System 2)

Hardware Optimization – Heterogenous SoC





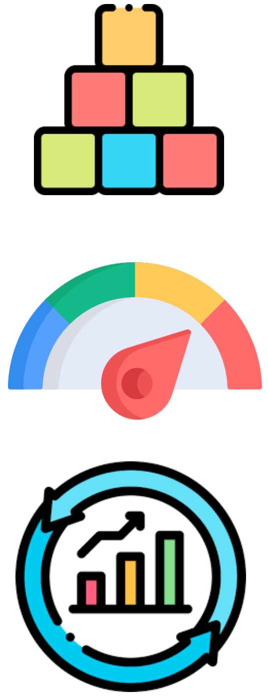
- Hardware system for embodied agent systems:
  - LLM Subsystem: for high-level decision making and communication
  - Control Subsystem: for low-level planning and action

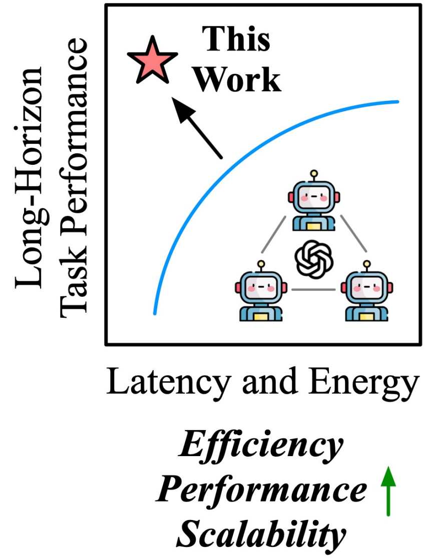
Wan, Du, Ibrahim, et al, "ReCA: Integrated Acceleration for Real-Time and Efficient Cooperative Embodied Autonomous Agents", in ASPLOS 2025

Summary

Embodied agents integrate perception, cognition, and physical action to conduct long-horizon tasks

- *Understand* fundamental **building blocks** and **paradigms** of embodied systems.
- *Identify* **system characteristics** and **sources of inefficiency** of embodied systems.
- *Demonstrate* **optimization opportunities** and **scalability-efficiency improvements** for embodied systems.





Opportunities for Embodied AI Agent Systems

Layered software stack for embodied AI **flexibility**

- Control adaptation layer: simplify hardware integration
- Core robotic function layer: handle autonomy operations
- Application layer: enable AI application development

Integrated computing architecture for embodied AI **efficiency**

- Integrate multimodal sensors seamlessly
- Deliver robust computational support for robotic kernels
- Facilitate visual-language model applications

Data-centric design automation for embodied AI **scalability**

- Need extensive and high-quality datasets
- Design automation pipeline: synthetic and real-word data
- Digital twin and hardware-in-the-loop development

Standard framework for embodied AI **safety and reliability**

- Safety: malfunctional behavior can result in harm to humans
- Reliability: consist performance across conditions
- Fault Tolerance: recover from errors with minimal disruption
- Standard: ISO26262 for AV -> what's for embodied AI?

ISPASS

Georgia Tech | University of Minnesota | Harvard University

Generative AI in Embodied Systems: System-Level Analysis of Performance, Efficiency and Scalability

Zishen Wan[1], Jiayi Qian[1], Yuhang Du[2], Jason Jabbour[3], Yilun Du[3], Yang (Katie) Zhao[2], Arijit Raychowdhury[1], Tushar Krishna[1], Vijay Janapa Reddi[3]

Email: *zishenwan@gatech.edu*

Webpage: *https://zishenwan.github.io*

IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS), 2025