Agent Primitives: Reusable Latent Building Blocks for Multi-Agent Systems

ICML International Conference On Machine Learning

*Haibo Jin, Peng Kuang, Ye Yu, Xiaopeng Yuan, Haohan Wang**

UNIVERSITY OF ILLINOIS URBANA-CHAMPAIGN

Contact Information:
haibo@illinois.edu

Website

Code

TL;DR

We propose Agent Primitives, a set of reusable latent building blocks for LLM-based multi-agent systems. Communicating via KV cache, primitives improve accuracy by 12.0 – 16.5% over single-agent baselines, reduce token and latency by 3× – 4× compared to text-based MAS, with only 1.3× – 1.6× overhead relative to single-agent inference.

+12.0 – 16.5%
Avg. Accuracy Gain

3× – 4×
Lower Tokens & Latency vs. Text MAS

1.3× – 1.6×
Overhead vs. Single-Agent

More Stable
Across Model Backbones

1. Motivation

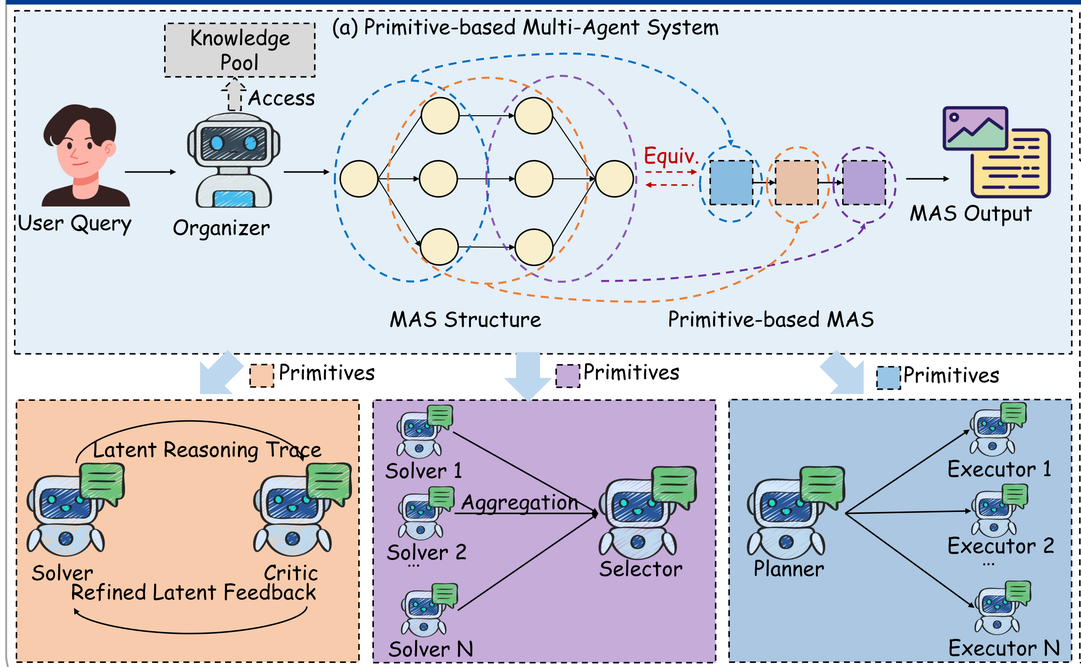
Existing MAS rely on manually designed roles and natural-language integration, leading to:

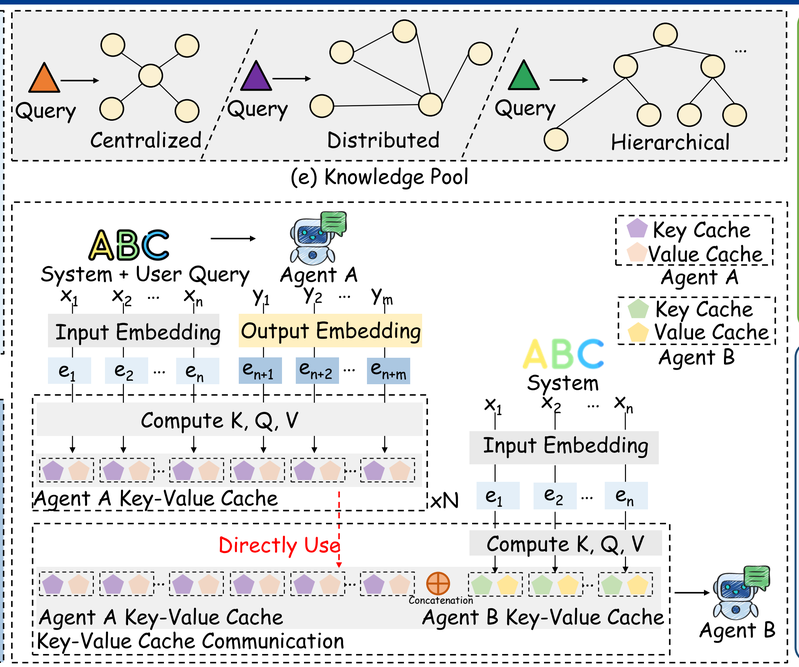
- **High task-specific design cost**
- **Limited reusability across tasks**
- **Degradation from long-context language communication**

Our Insight

Many MAS can be decomposed into a small number of recurring computation patterns. Like neural networks, we can build complex MAS from reusable building blocks

2. Overview of Agent Primitives





Key Benefit

- **Reusable across tasks**
- **Light-weight knowledge pool guided composition**
- **Plug-and-play construction for any query**

Why KV Cache

- **Avoids language degradation in long context**
- **Preserve rich latent information**
- **Faster and more stable interaction**

3. Performance of Agent Primitives Across Tasks

Models	Methods	Math Problem Solving				Code Generation		Q&A		Avg.
		AIME25	AIME24	MATH	GSM8K	HumanEval+	MBPP+	MedQA	GPQA-Diamond	
Qwen3-8B	Single	46.7%	50.0%	60.8%	81.1%	74.4%	64.8%	53.0%	39.9%	58.8%
	TextMAS	53.3% (+6.6%↑)	53.3% (+3.3%↑)	61.4% (+0.6%↑)	92.3% (+11.2%↑)	80.5% (+6.1%↑)	69.5% (+4.7%↑)	75.0% (+22.0%↑)	43.4% (+3.5%↑)	66.1% (+7.3%↑)
	LatentMAS	53.3% (+6.6%↑)	56.7% (+6.7%↑)	62.6% (+1.8%↑)	93.8% (+12.7%↑)	80.5% (+6.1%↑)	74.6% (+9.8%↑)	75.3% (+22.3%↑)	45.5% (+5.6%↑)	67.8% (+8.9%↑)
	Review	60.0% (+13.3%↑)	63.3% (+13.3%↑)	61.0% (+0.2%↑)	93.2% (+12.1%↑)	78.6% (+4.2%↑)	70.6% (+5.8%↑)	64.2% (+11.2%↑)	48.9% (+9.0%↑)	67.5% (+8.6%↑)
	Voting	66.7% (+20.0%↑)	70.0% (+20.0%↑)	61.4% (+0.6%↑)	91.8% (+10.7%↑)	81.0% (+6.6%↑)	74.3% (+9.5%↑)	70.3% (+17.3%↑)	55.0% (+15.1%↑)	71.3% (+12.5%↑)
	Planning	66.7% (+20.0%↑)	63.3% (+13.3%↑)	60.8% (+0.0%↑)	93.2% (+12.1%↑)	78.6% (+4.2%↑)	75.9% (+11.1%↑)	67.0% (+14.0%↑)	51.0% (+11.1%↑)	69.6% (+10.7%↑)
	Primitives-based MAS	**73.3%** (**+26.6%**↑)	**76.7%** (**+26.7%**↑)	**63.7%** (**+2.9%**↑)	**94.2%** (**+13.1%**↑)	**82.3%** (**+7.9%**↑)	**75.9%** (**+11.1%**↑)	**76.7%** (**+23.7%**↑)	**59.6%** (**+19.7%**↑)	**75.3%** (**+16.5%**↑)

4. Comparison with MAS

Methods	MATH	GSM8K	HumanEval+	GPQA
Single	50.6%	92.4%	75.8%	36.7%
Chain-of-Thought	53.2% (+2.6↑)	92.8% (+0.4↑)	77.0% (+1.2↑)	35.3% (-1.4↓)
Self-Consistency	61.6% (+11.0↑)	95.0% (+2.6↑)	75.8% (+0.0)	37.2% (+0.5↑)
LLM-Debate	61.4% (+10.8↑)	91.6% (-0.8↓)	74.5% (-1.3↓)	34.4% (-2.3↓)
Self-Refine	58.5% (+7.9↑)	90.8% (-1.6↓)	62.7% (-13.1↓)	38.3% (+1.6↑)
Quality-Diversity	60.5% (+9.9↑)	93.0% (+0.6↑)	70.2% (-5.6↓)	33.6% (-3.1↓)
SPP	51.7% (+1.1↑)	92.8% (+0.4↑)	73.3% (-2.5↓)	35.1% (-1.6↓)
AgentVerse	55.6% (+5.0↑)	93.4% (+1.0↑)	73.9% (-1.9↓)	40.2% (+3.5↑)
GPTSwarm	55.4% (+4.8↑)	93.2% (+0.8↑)	73.9% (-1.9↓)	36.5% (-0.2↓)
DyLAN	59.6% (+9.0↑)	91.2% (-1.2↓)	75.8% (+0.0)	36.0% (-0.7↓)
MAS-GPT	68.7% (+18.1↑)	93.4% (+1.0↑)	78.9% (+3.1↑)	37.6% (+0.9↑)
Primitives-based MAS	**72.4%** (**+21.8↑**)	**93.8%** (**+1.4↑**)	**82.3%** (**+6.5↑**)	**53.2%** (**+16.5↑**)

5. Efficiency

Method	MATH Cost	MATH Acc.	MATH Eff.	GSM8K Cost	GSM8K Acc.	GSM8K Eff.	HumanEval+ Cost	HumanEval+ Acc.	HumanEval+ Eff.	GPQA Cost	GPQA Acc.	GPQA Eff.
Single	.0024	50.6%	210.8	.0016	92.4%	573.9	.0037	75.8%	204.9	.0117	36.7%	31.4
CoT	.0033	53.2%	161.2	.0025	92.8%	369.4	.0041	77.0%	186.6	.0130	35.3%	27.2
Self-Consistency	.0204	61.6%	30.2	.0153	95.0%	62.1	.0352	75.8%	21.5	.0729	37.2%	5.1
LLM-Debate	.0081	61.4%	75.8	.0101	91.6%	90.9	.0151	74.5%	49.3	.0250	34.4%	13.8
Self-Refine	.0055	58.5%	106.1	.0038	90.8%	240.5	.0086	62.7%	73.1	.0268	38.3%	14.3
Quality-Diversity	.0132	60.5%	45.9	.0090	93.0%	103.5	.0205	70.2%	34.2	.0352	33.6%	9.5
SPP	.0052	51.7%	99.8	.0034	92.8%	273.7	.0067	73.3%	109.3	.0198	35.1%	17.7
AgentVerse	.0080	55.6%	69.6	.0061	93.4%	152.3	.0161	73.9%	45.9	.0248	40.2%	16.2
GPTSwarm	.0074	55.4%	74.9	.0061	93.2%	153.6	.0152	73.9%	48.6	.0232	36.5%	15.7
DyLAN	.0085	59.6%	70.1	.0062	91.2%	147.1	.0165	75.8%	45.9	.0283	36.0%	12.7
MAS-GPT	.0082	68.7%	83.5	.0065	93.4%	143.7	.0174	78.9%	45.4	.0257	37.6%	14.6
Ours (GPT-5.2)	.0056	72.4%	129.3	.0048	93.8%	195.4	.0118	82.3%	69.7	.0171	53.2%	31.1
Ours (Llama-3-70B)	.0028	72.4%	258.6	.0020	93.8%	469.0	.0063	82.3%	130.6	.0122	53.2%	43.6

Key Papers

[1] Zou, J., Yang, X., Qiu, R., Li, G., Tieu, K., Lu, P., ... & Yang, L. (2025). Latent collaboration in multi-agent systems. arXiv preprint arXiv:2511.20639.

[2] Hong, S., Zhuge, M., Chen, J., Zheng, X., Cheng, Y., Wang, J., ... & Schmidhuber, J. (2024, May). MetaGPT: Meta programming for a multi-agent collaborative framework. In International Conference on Learning Representations (Vol. 2024, pp. 23247-23275).