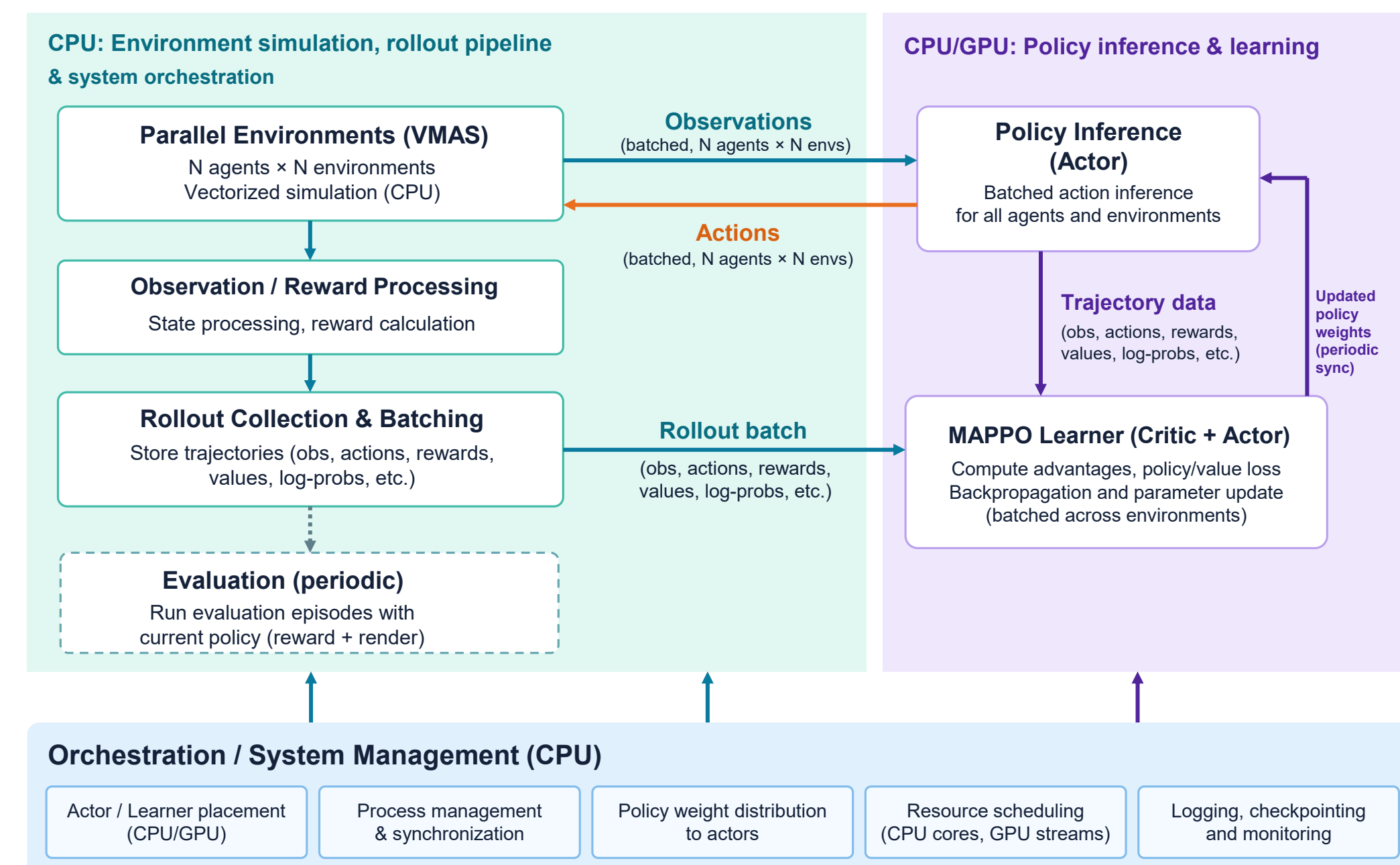


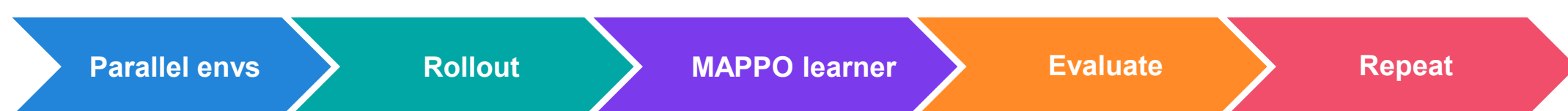
Takeaway: Arm Neoverse - A High-Performance CPU Platform for Reinforcement Learning Training and Simulation, from Physical AI to Agentic AI

01 Workload, architecture and pipeline

We study multi-agent RL with MAPPO and VMAS. CPU and GPU work together across the full RL loop.



End-to-end RL pipeline



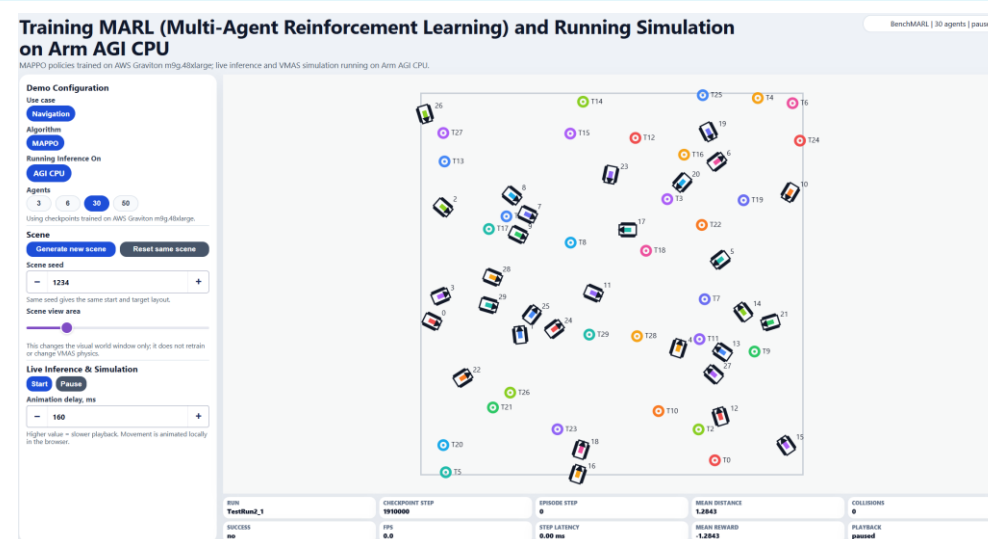
Experiment setup

Framework	PyTorch 2.11, BenchMARL, TorchRL
Algorithm	Multi Agent Proximal Policy Optimization (MAPPO) with shared actor and centralized critic
Environment	VMAS Navigation; 2D robotics-style multi-agent task
Scale	GH200: 3 & 30 agents / 64 environments Cloud CPU: 30 & 50 agents / 191 environments
Devices	NVIDIA GH200 - Arm Grace CPU + Hopper GPU; AWS Arm and AMD cloud CPU instances
Metrics	Iteration time, rollout time, learner time, FPS, reward, runtime

Live demo images + resources



MAPPO navigation with obstacle avoidance on real robots.
Robot Demo credit: Waheed Brown, Arm



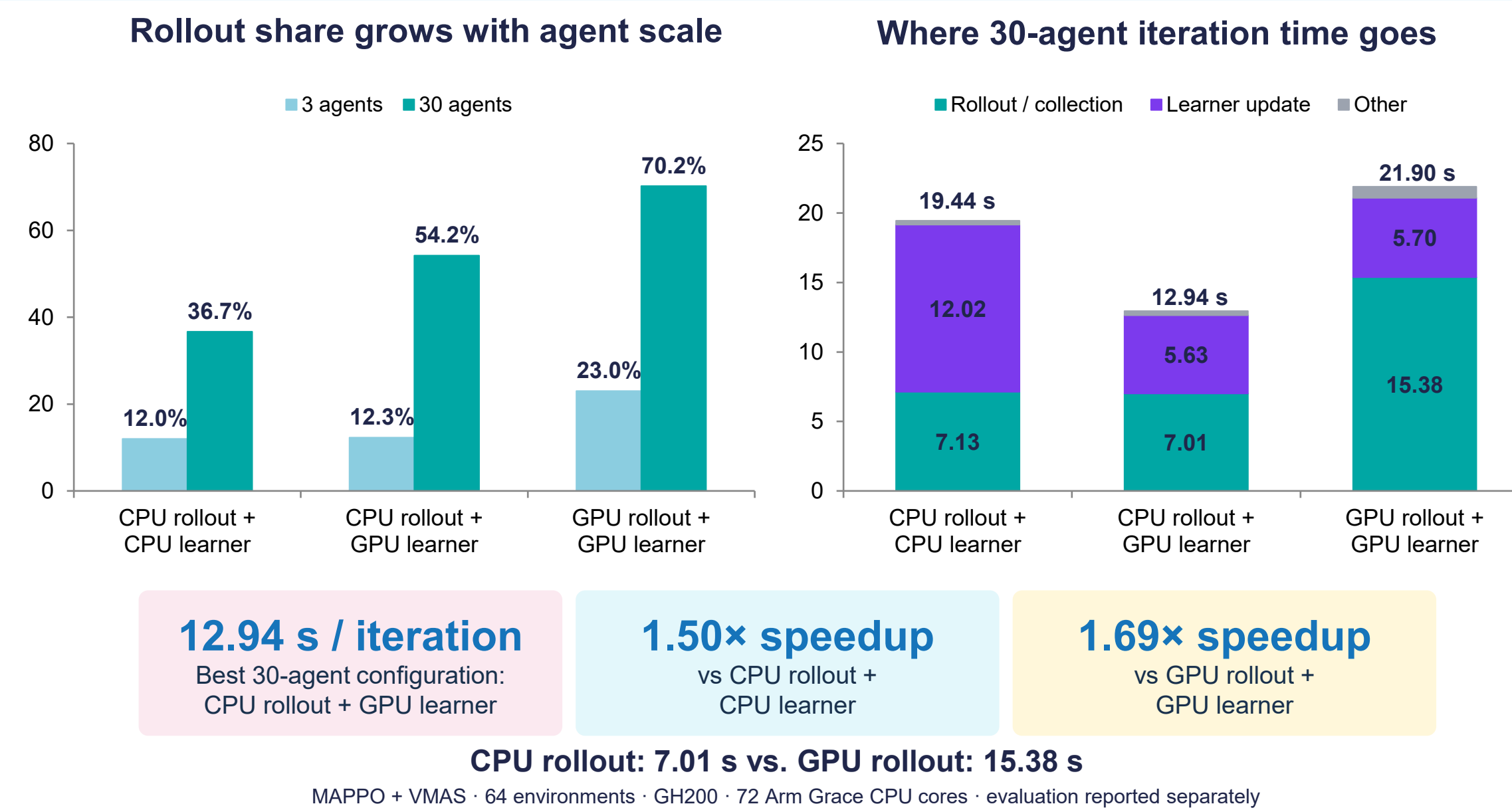
MAPPO navigation with obstacle avoidance simulation on Arm CPU.

MAPPO workflow and resources
PyTorch · BenchMARL · TorchRL

Arm Learning Path
Follow the MAPPO Learning Path



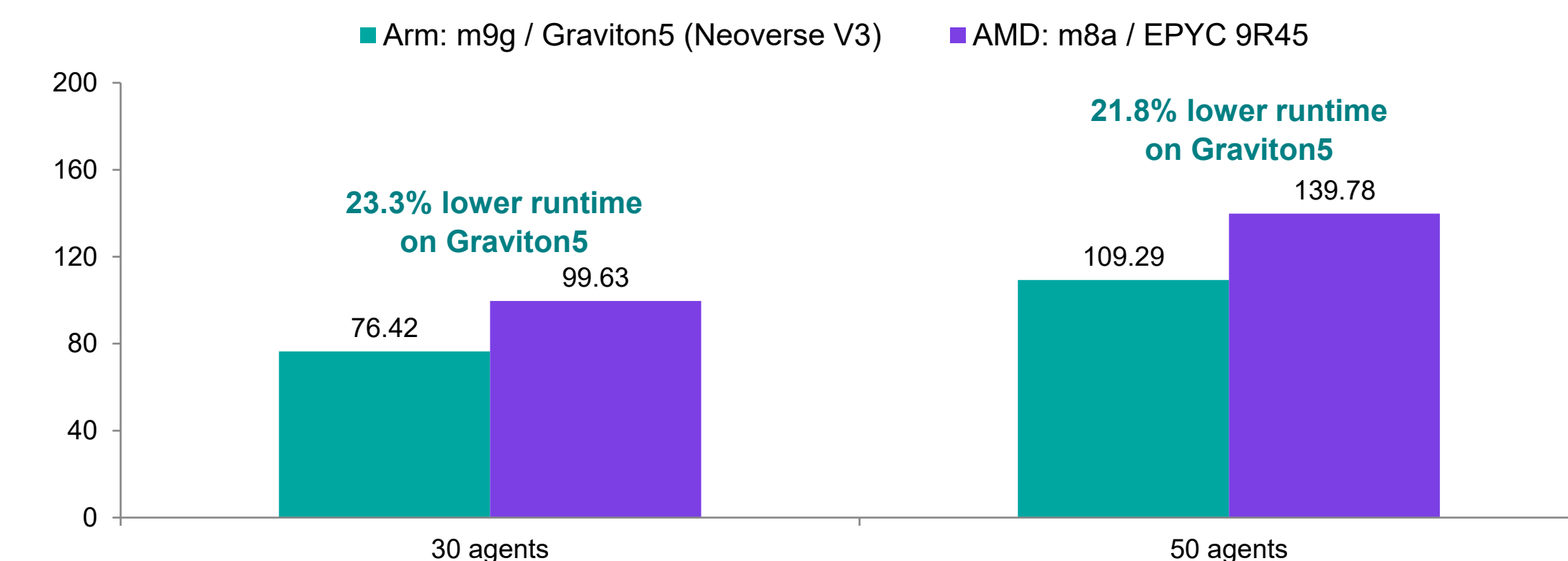
02 CPU is the better choice than GPU for RL Rollout



04 Graviton 5 delivers better performance than AMD Turin for RL training

AWS m9g.48xlarge vs m8a.48xlarge · 30 and 50 agents

Runtime (minutes) - lower is better



Test configuration · Train + periodic evaluation

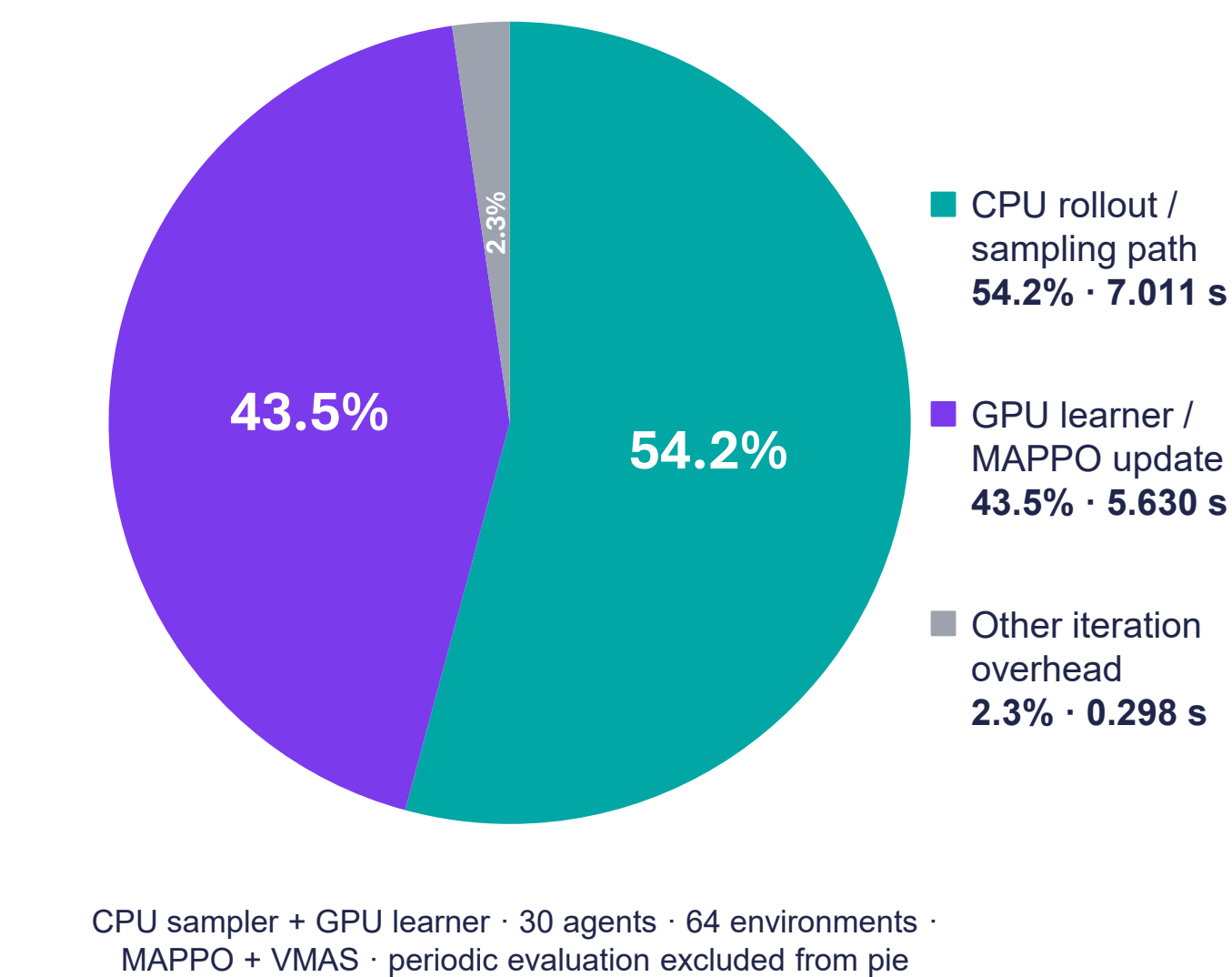
	Arm	AMD
Instance	AWS m9g.48xlarge	AWS m8a.48xlarge
CPU	Graviton5 (Neoverse V3)	EPYC 9R45
Compute	192 vCPUs / 192 visible cores	192 vCPUs / 192 visible cores
Memory	768 GiB	768 GiB
Workload	191 VMAS environments	191 VMAS environments

Key Messages

- CPU is a strong choice for RL rollout** - CPU rollout outperformed GPU rollout in our MAPPO workload.
- CPU-only is a practical choice for full RL training** - rollout, learning, evaluation, and the demo can all run effectively on CPU.
- Arm delivers strong CPU-only RL performance** - Graviton5 / Neoverse V3 delivers better high-agent MAPPO performance than AMD EPYC Turin in matched cloud runs.

03 CPU matters in RL training

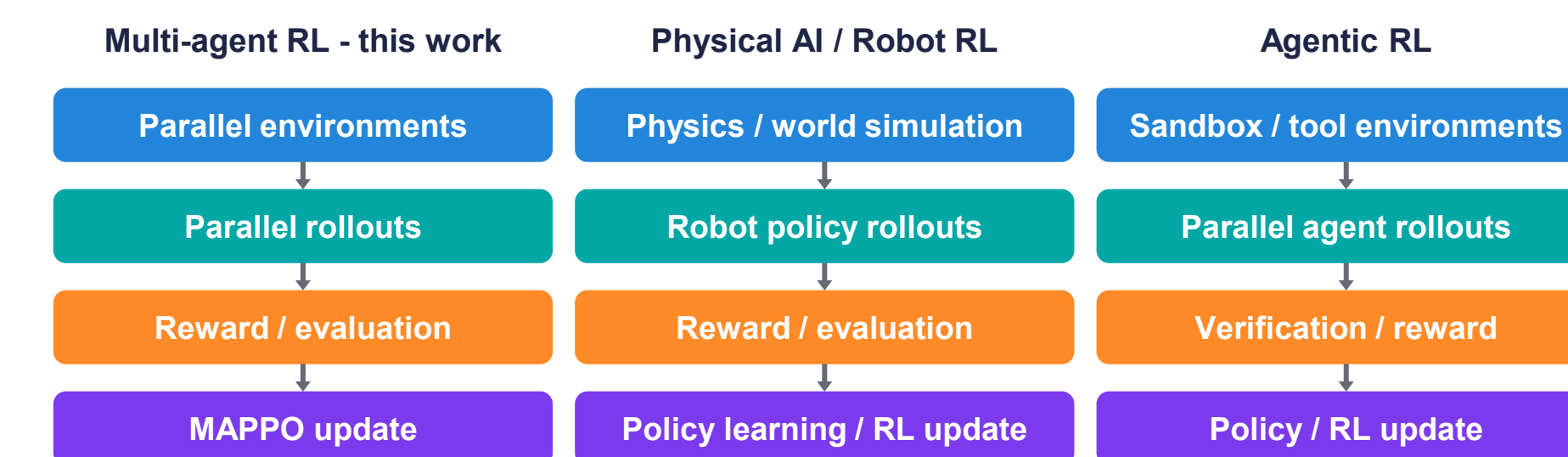
Training iteration breakdown (30 agents)



Key observations

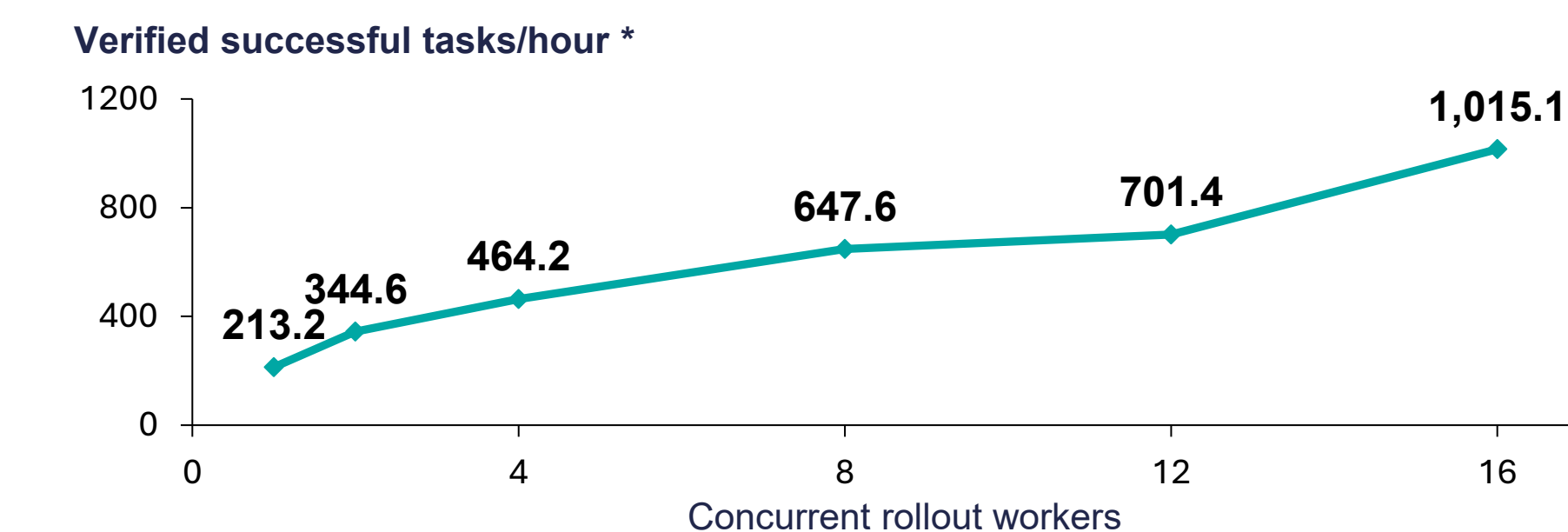
- CPU rollout remains first-class**
Even with GPU learning, 54.2% of the 30-agent iteration is spent in CPU rollout/sampling.
- Rollout share increases with agent scale**
In the CPU-sampler + GPU-learner configuration, rollout grows from about 12.3% of iteration time at 3 agents to 54.2% at 30 agents.
- System balance determines end-to-end performance**
At 30 agents, CPU rollout + GPU learning produces the lowest measured iteration time.

05 RL extends to Physical AI and Agentic AI



06 CPU activity rises with rollout concurrency for Agentic AI

Separate AWM tool-use experiment · mixed tasks 2, 4 and 8 · 1-second worker-launch stagger



4.8×

Useful throughput
213 → 1,015 successful tasks/hour

3.1×

CPU activity
3.38 → 10.34 system busy cores

+38.9

Percentage points
GPU utilization: 45.5% → 84.4%

CPU-side environment, tool, verification and orchestration capacity becomes part of scaling Agentic RL infrastructure.

* Successful tasks/hour is the directly measured outcome here. For RL infrastructure, valid completed rollouts/hour is a useful complementary metric because both successful and unsuccessful trajectories can provide learning signal.