The Evolving Landscape of LLM- and VLM-Integrated Reinforcement Learning

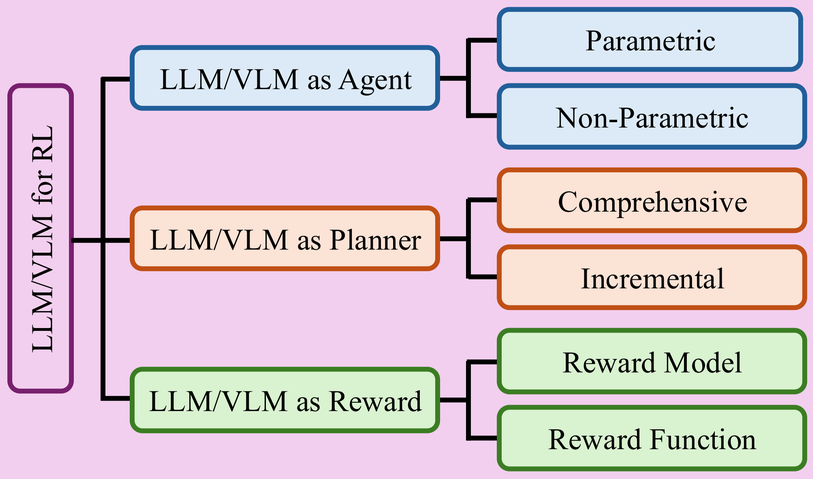
Sheila Schoepp[1], Masoud Jafaripour[1*], Yingyue Cao[1*], Tianpei Yang[2], Fatemeh Abdollahi[1], Shadan Golestan[3], Zahin Sufiyan[1], Osmar R. Zaiane[1,3], and Matthew E. Taylor[1,3]

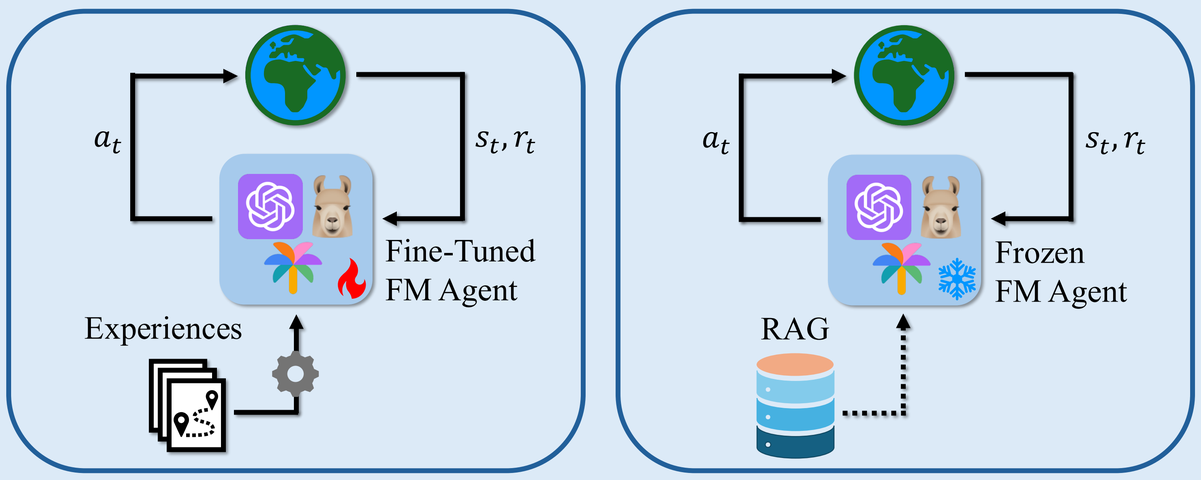
Motivation

- Reinforcement learning (RL) excels at trail-and-error, but struggles with prior knowledge, long horizon planning, reward design, sample efficiency, and interpretability.
- Large Language Models (LLMs) and vision-language models (VLMs) contribute world knowledge, reasoning, and perception that can address these gaps.

Inclusion Criteria

- Integrate a foundation model (LLM or VLM) into the RL framework.
- Frame tasks as Markov decision processes (MDPs).
- Use the RL reward signal to guide learning.
- Use LLMs/VLMs developed from GPT-3 (2020) onward.



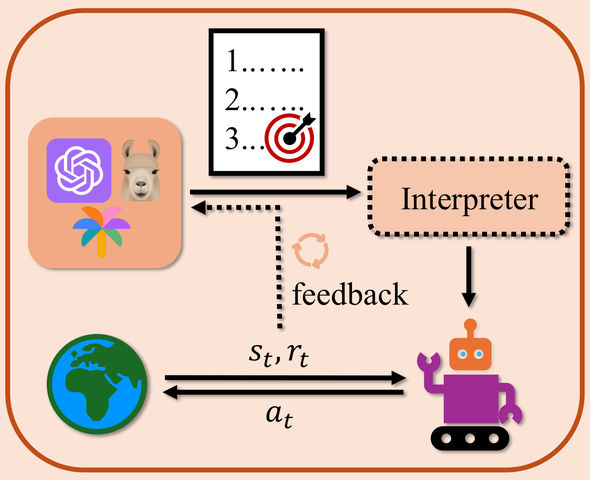


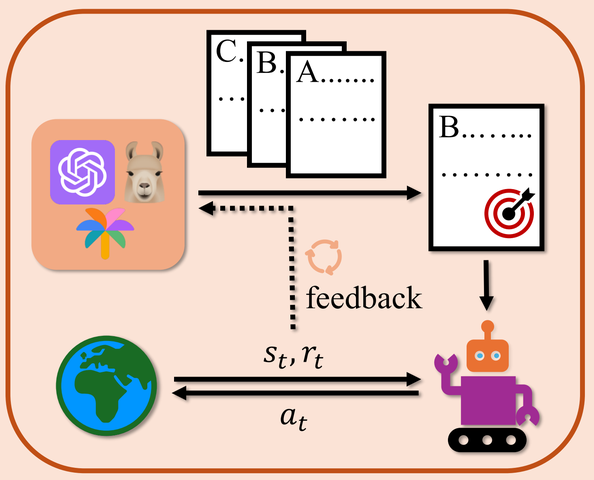
LLM/VLM as Agent

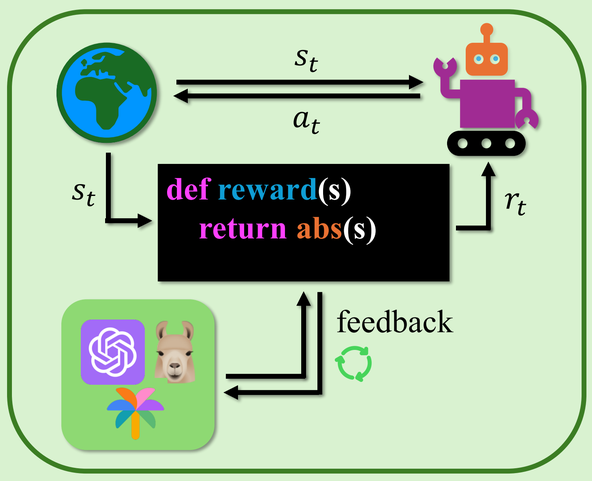
- LLM/VLM is the policy.
- Parametric agents – lightweight fine-tuning with RL improves adaptability and sample efficiency.
- Non-parametric agents – no fine-tuning; rely on in-context learning, memory, and self-reflection to scale and generalize.

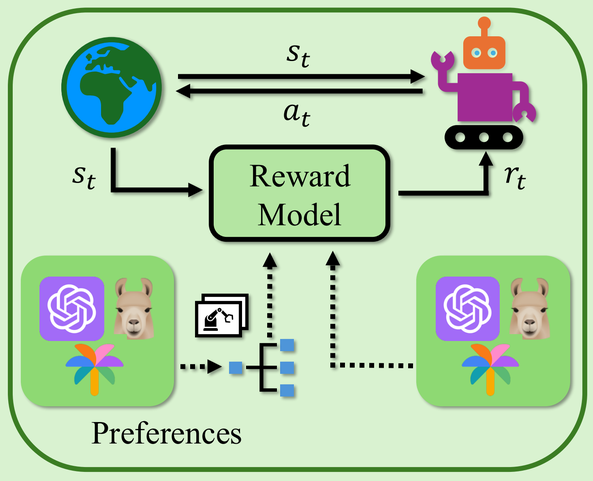
LLM/VLM as Planner

- LLM/VLM decomposes complex tasks into sub-goals that a low-level controller executes.
- Comprehensive planning – LLM/VLM proposes all sub-goals at once; efficient but brittle in dynamic settings.
- Incremental planning – LLM/VLM proposes sub-goals on the fly; robust to feedback but increases query cost









LLM/VLM as Reward

- LLM/VLM specifies reward code or trains/acts as a reward model.
- Reward function generation – LLM/VLM writes/refines reward code from environment abstractions and feedback, often matching/surpassing expert designed reward functions.
- Reward models – LLM/VLM maps trajectories to scalar rewards or provides preferences.

Open Problems

- Grounding – bridge natural language plans to low-level control without brittle interfaces.
- Inherent bias – debias decisions, plans, and rewards; analyze failure modes.
- Representation - fuse numeric sensors with language for precise control; explore multi-modal encoders.
- Action advice – use LLMs/VLMs as imperfect but helpful teachers to accelerate RL agent learning.

Read Our Survey

[1]Department of Computing Science, University of Alberta
[2]Nanjing University
[3]Alberta Machine Intelligence Institute (Amii)