Decentralized Refinement Planning and Acting

Ruoxi Li, Sunandita Patra, Dana Nau
University of Maryland, College Park

Introduction

A decentralized group of agents needs to coordinate in an environment where unexpected things can occur, with limitations on sensors and communication. We're accomplishing this using hierarchical planning, coordination protocols, Monte Carlo sampling, and task allocation.

Key words:

- Decentralized multi-agent system
- Refinement/ hierarchical planning
- Operational models
- Task allocation

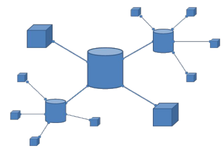
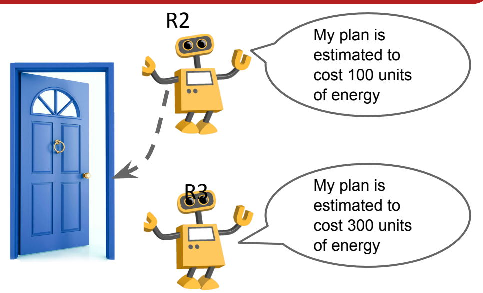


Fig1: Decentralized system

Motivating Scenario



The blue robot is given a task to deliver a box through the door. One of the yellow robots is chosen to come over and hold the door open while the blue robot is crossing the doorway (R2 is chosen because it has a lower-cost plan).

Fig1: Illustration of the motivation scenario

Decentralized Refinement Planning and Acting

- Generalize RAE and UPOM to do decentralized acting and planning
- Each agent has its own acting engine (Dec-RAE) and planner (D-UPOM)
- Agents don't explicitly know other agents' operational models
- Partial observability
- Non-deterministic action outcomes
- Decentralized planning and acting
- Types of communication
  - State: local state information
  - Goal: a desire that has been adopted for active pursuit by the agent
  - Task outsourcing:
    - a desired task that the agent needs other agents to accomplish
  - Plan: utility and the effects of the plan

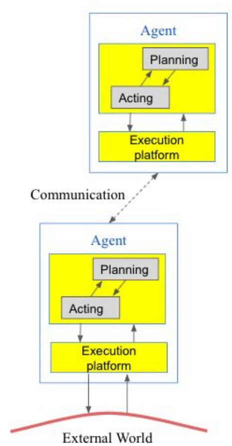


Fig2: System overview

D-UPOM: Decentralized UCT Planner with Operational Models

- Planner uses actor's operational models
- Repeated Monte Carlo rollouts (e.g., orange arrow) of the operational models in a simulated environment
- Statistical sample of possible outcomes (different method choices and action outcomes)
- Maximize the utility (total reward)
- Choose the method with greatest expected utility, choose the delegatee that has the most desirable plan (in terms of utility and effect)
- In the planning time, agent i selects m or m' according to their estimated utilities to act on task τ
- To estimate the utility of an outsourced task:
  - Agent i requests agent j and k to estimate how well they can solve task $\tau 2$
  - Agent j and k each plans for $\tau 2$
    - Each performs n rollouts locally
    - Returns estimated utility and effect to agent i

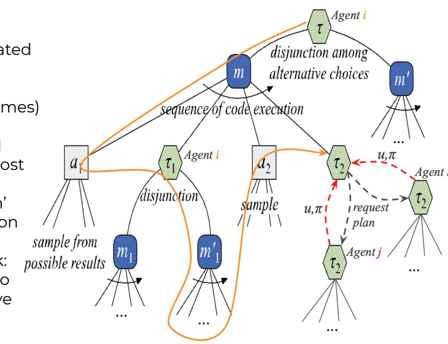
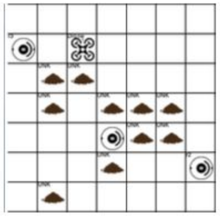


Fig3: The space of refinement trees

Experiments

We present experimental evaluations of Dec-RPAE in two domains and observe that:

- Additional Monte-Carlo rollouts in the planning component improve the performance of the acting component in both single-agent and multi-agent settings.
- Communication enables coordination between agents thereby improves their performance to a large extent.
- Agents can concurrently coordinate their actions.
- D-UPOM works in a setting where tasks need to be delegated among each other recursively.



Conclusion

We introduced a decentralized multi-agent planning and acting engine that uses operational models like the ones used in RAE. It consists of two components, Dec-RAE and D-UPOM

- Dec-RAE, the decentralized acting component, is a generalization of RAE. Multiple agents can run Dec-RAE concurrently in a decentralized fashion, and can use it to perform actions, communicate and delegate tasks among each other.
- D-UPOM, the decentralized planning component, uses a Monte-Carlo rollout technique based on the well- known UCT algorithm
- D-UPOM converges with monotonic utility functions.

Contact

Ruoxi Li (rli12314@umd.edu)
University of Maryland, College Park)