Interleaving Acting and Planning Using Operational Models

Sunandita Patra[1] (patras@umd.edu), Malik Ghallab[2], Dana Nau[1], Paolo Traverso[3]

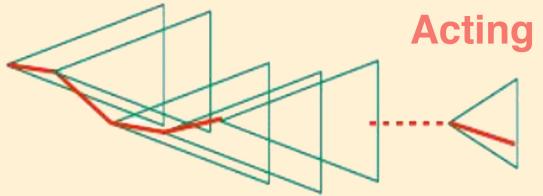[1]University of Maryland, USA, [2]LAAS-CNRS, France, [3]FBK-ICT, Italy

Planning

- Prediction + search
- To reach a goal or accomplish a task

Acting

- Performing tasks and actions in the real world
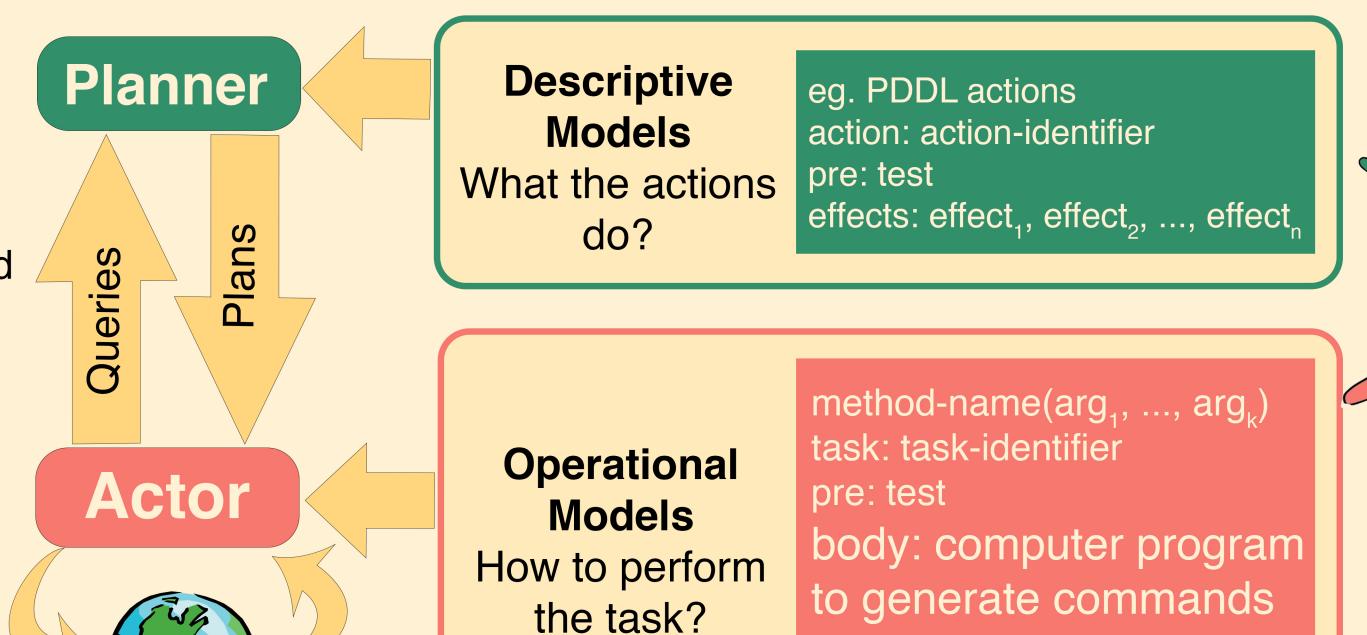- Adapt to context, react to events
- Dynamic environment

Planning Stage
Acting Stage

Planner

Queries | Plans

Actor

Descriptive Models
What the actions do?

eg. PDDL actions
action: action-identifier
pre: test
effects: $effect_1$, $effect_2$, ..., $effect_n$

Operational Models
How to perform the task?

method-name(arg_1, ..., arg_k)
task: task-identifier
pre: test
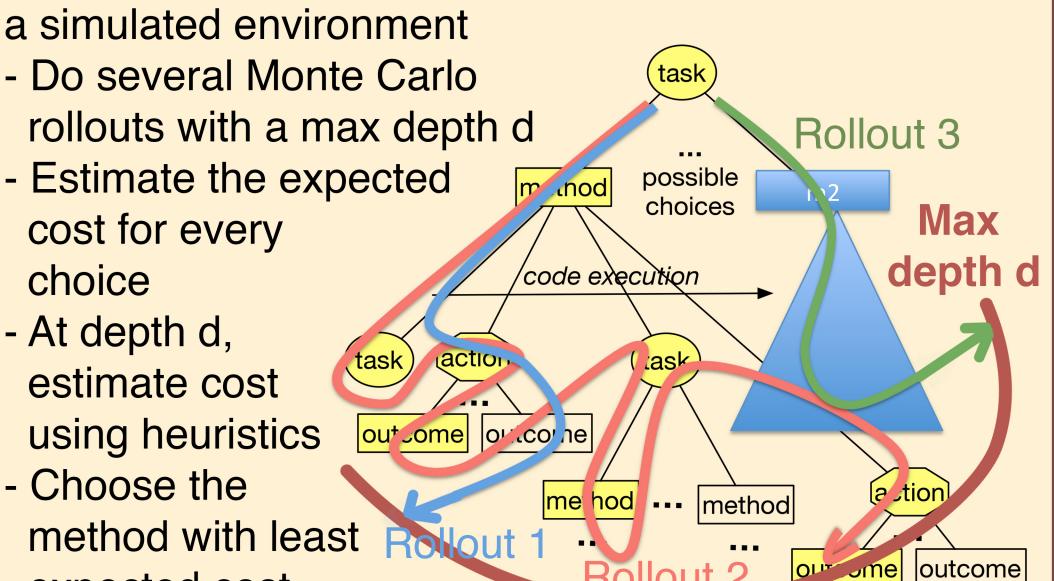body: computer program to generate commands and more tasks

Problem:
Are they consistent?

Our Contribution
We developed a planner RAEplan-LookAhead that uses the actor's operational models for online planning.

Input stream of tasks that may lead to more tasks

Actor (RAE)

Which method to choose to refine a task?
- Tough choice
- Nondeterministic actions
- Dynamic, partially observable environment
- May lead to failure or make the situation worse

Solution: Do online planning using the operational models

Planning Algorithm: RAEplan-LookAhead

Idea: Do rollout simulations of tasks and commands in a simulated environment
- Do several Monte Carlo rollouts with a max depth d
- Estimate the expected cost for every choice
- At depth d, estimate cost using heuristics
- Choose the method with least expected cost

Rollout 3
...
possible choices
task
method
code execution
task | action
task
outcome | outcome
method ... method
action
Rollout 1
Rollout 2
outcome | outcome
Max depth d

Experimental Evaluation

- Simulated Domain: **Search and Rescue**
- Measure Efficiency (inverse of cost), acting time and planning time

b = number of methods looked at for every task
Depth = maximum depth of the rollouts

Survivors located