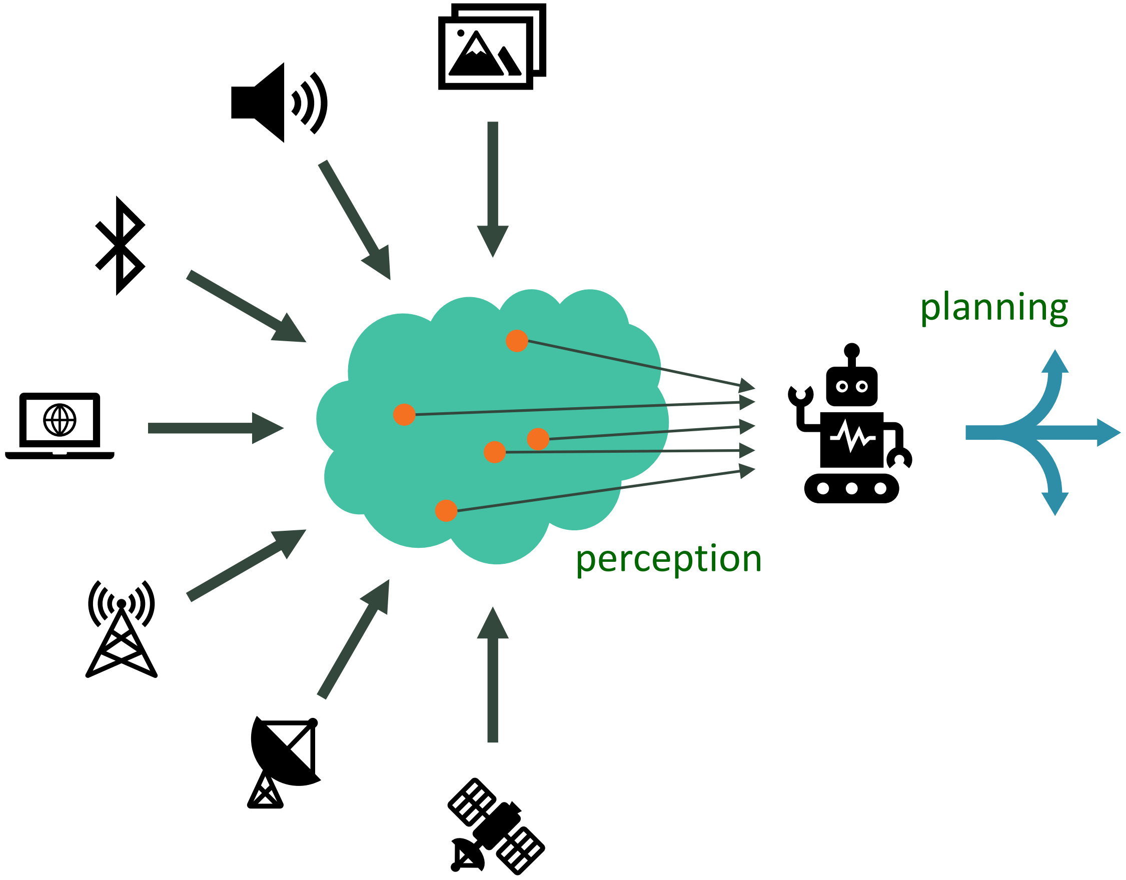
Task-Oriented Active Perception and Planning in Environments with Partially Known Semantics

MAHSA GHASEMI, ERDEM ARINÇ BULGUR, AND UFUK TOPCU

INTERNATIONAL CONFERENCE ON MACHINE LEARNING
JULY 12-18, 2020

Integrating Data into Decision Making Process



Setting

- Sequential decision making
- Partial knowledge of environment
- Continual information gathering

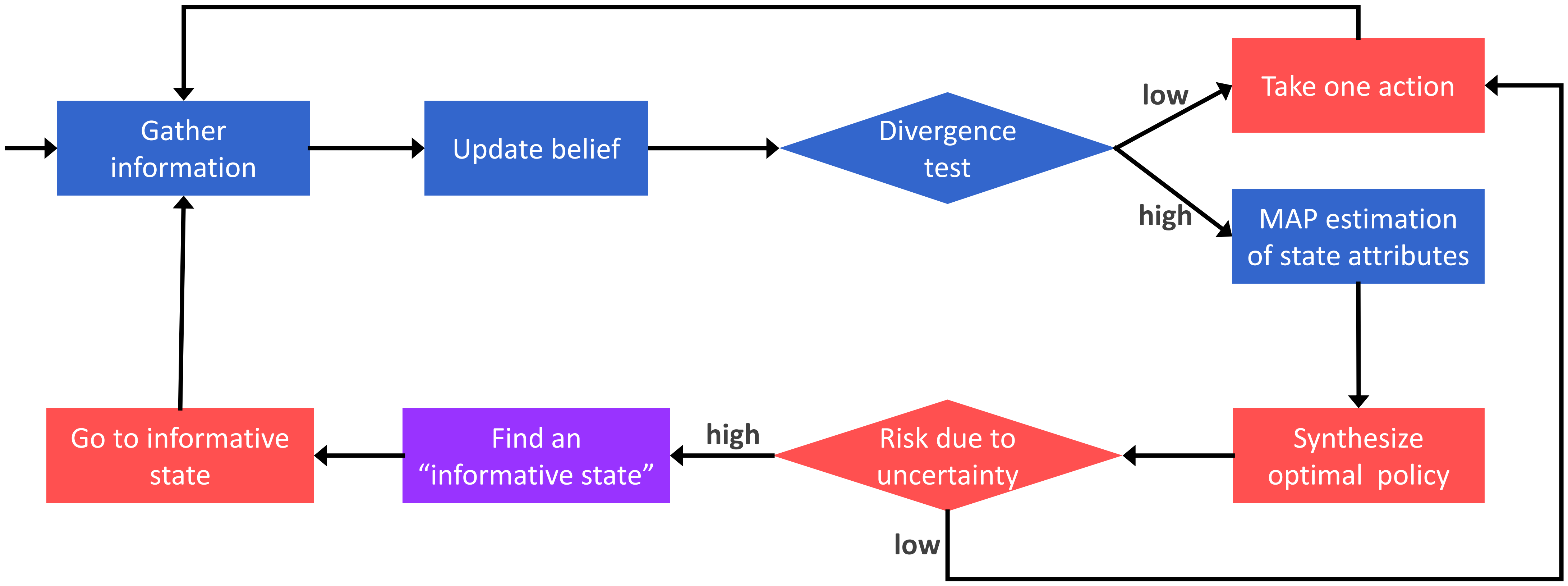
Challenge

How to simultaneously perceive and plan with efficiency and performance guarantee?

Contributions

1. Provide guarantee on task success
2. Characterize information utility
3. Guide active perception while planning

Task-Oriented Active Perception and Planning



System Dynamics as Markov Decision Process

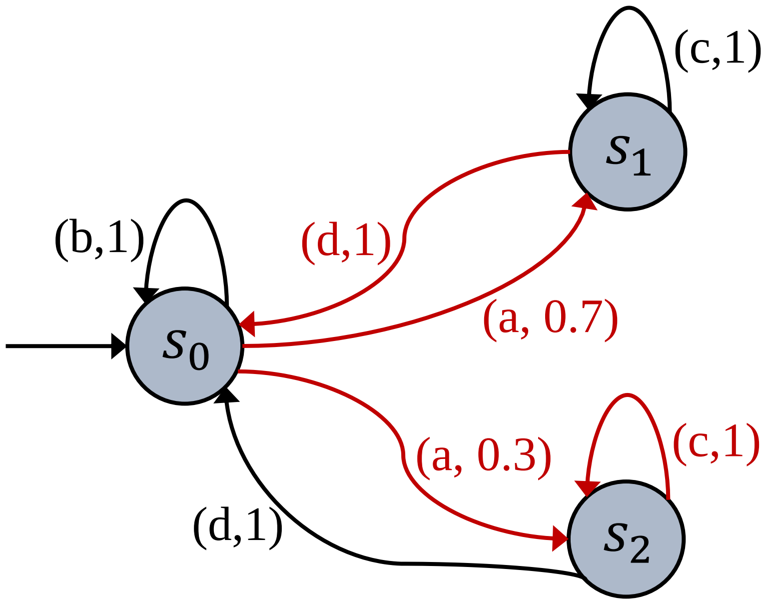
An MDP is a tuple $\mathcal{M} = (\mathcal{S}, s_{init}, \mathcal{A}, \mathcal{T})$

- $\mathcal{S}$ is a finite discrete state space
- s_{init} is an initial state
- $\mathcal{A}$ is a finite discrete action space
- $\mathcal{T} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0,1]$ is a probabilistic transition function such that for all $s \in \mathcal{S}$ and for all $a \in \mathcal{A}$, $\sum_{s' \in \mathcal{S}} \mathcal{T}(s, a, s') = 1$

Memoryless deterministic policies $\pi : \mathcal{S} \to \mathcal{A}$

Induced Markov chain $\mathcal{M}^\pi = (\mathcal{S}^\pi, s^\pi_{init}, \mathcal{T}^\pi)$

- $\mathcal{S}^\pi = \mathcal{S}$
- $s^\pi_{init} = s_{init}$
- $\mathcal{T}^\pi : \mathcal{S} \times \mathcal{S} \to [0,1]$ is such that for all $s, s' \in \mathcal{S}$, $\mathcal{T}^\pi(s, s') = \mathcal{T}(s, \pi(s), s')$



An MDP

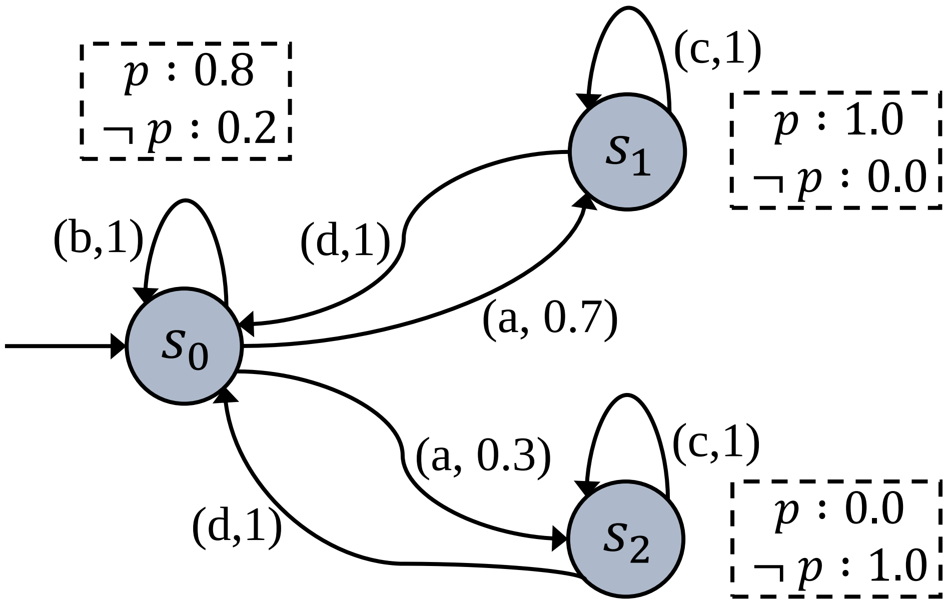
Environment Model and Observation Model

An environment model is a tuple $\mathcal{E} = (\mathcal{S}, \mathcal{AP}, \bar{\mathcal{L}})$

- $\mathcal{S}$ is a finite discrete state space
- $\mathcal{AP}$ is a set of atomic propositions
- $\bar{\mathcal{L}} : \mathcal{S} \to 2^{\mathcal{AP}}$ is a true labeling function

An observation model is a joint probability distribution $\mathcal{O} : \mathcal{S} \times \mathcal{S} \times \mathcal{AP} \times \{True, False\} \to [0, 1]$.

→ Belief at time t is a probabilistic labeling function $\mathcal{L}_t : \mathcal{S} \times 2^{\mathcal{AP}} \to [0, 1]$ such that for all $s \in \mathcal{S}$, $\sum_{P \subseteq \mathcal{AP}} \mathcal{L}_t(s, P) = 1$.



An MDP with partial semantics

Task Specification with Linear Temporal Logic

- **Linear temporal logic (LTL):** A formal language with logical and temporal operators

$\neg$: not	$\Diamond$: eventually
$\wedge$: and	$\Box$: always
$\vee$: or	$\mathcal{U}$: until

  - Suitable for high-level task specification
  - Verifiable
    - Qualitative (almost surely)
    - Quantitative (probabilistically)
  - Close to human language
    - Formal translation of natural language instructions into LTL specifications
      [E.g., LTLMoP toolkit by Finucane, Jing and Hadas Kress-Gazit, 2010]

Automaton Representation of Task

- Task specification as LTL formula (with probabilistic guarantee)

$$\varphi = (\neg obs\,\mathcal{U}\,door_1) \vee ((\neg door_2\,\mathcal{U}\,key) \wedge (\neg obs\,\mathcal{U}\,door_2))$$

or

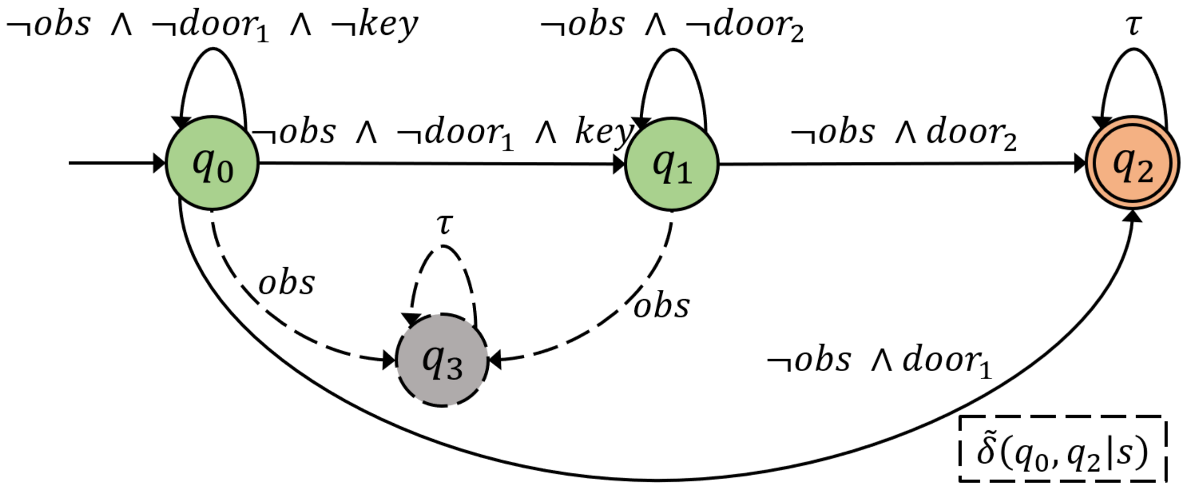
and

Do not crash with obstacles until you reach door 1

Do not go to door 2 until you find the key

Do not crash with obstacles until you reach door 2

- An LTL formula can be transformed into an automaton
  - A transition system for a task
  - Captures task progress
  - A run ending in the accepting state completes the task



An automaton

Formal Problem Statement

<u>Given</u>

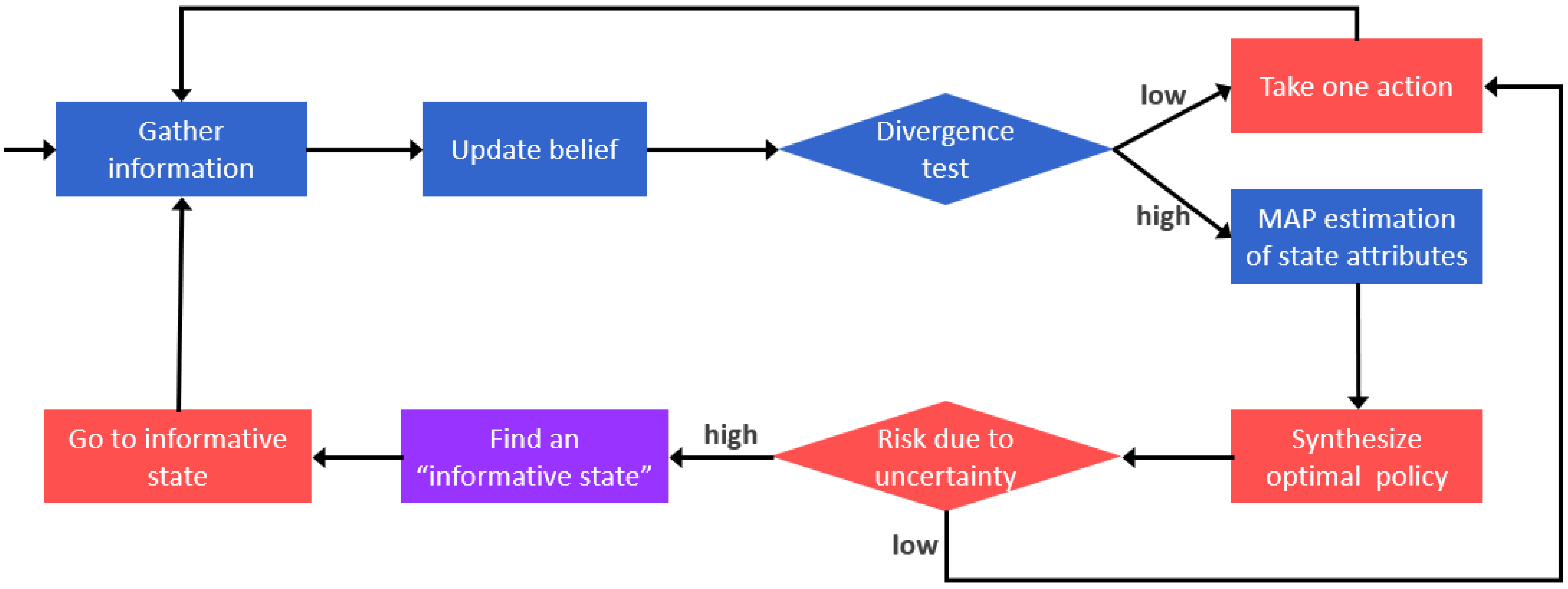
- An MDP $\mathcal{M} = (\mathcal{S}, s_{init}, \mathcal{A}, \mathcal{T})$
- An environment model with unknown labeling function $\mathcal{E} = (\mathcal{S}, \mathcal{AP}, \_)$
- An observation model $\mathcal{O}$
- A syntactically co-safe LTL task specification φ

<u>Find</u>

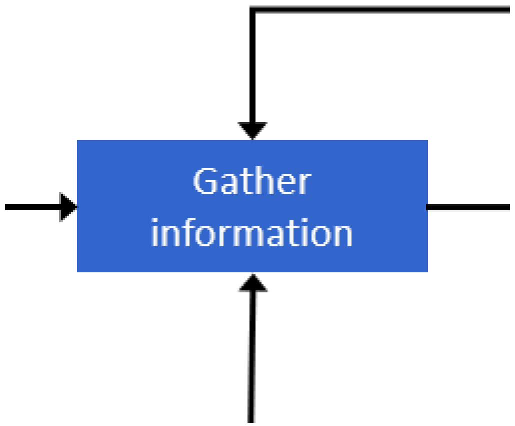
A policy π that maximizes the probability of satisfying the task conditioned on the true labeling function, i.e.,

$$\pi^* = \text{argmax}_\pi Pr(\mathcal{M}^\pi \models \varphi \,|\, \bar{\mathcal{L}})$$

Task-Oriented Active Perception and Planning



Task-Oriented Active Perception and Planning



Perception module receives data sampled according to the observation model $\mathcal{O}$

Task-Oriented Active Perception and Planning

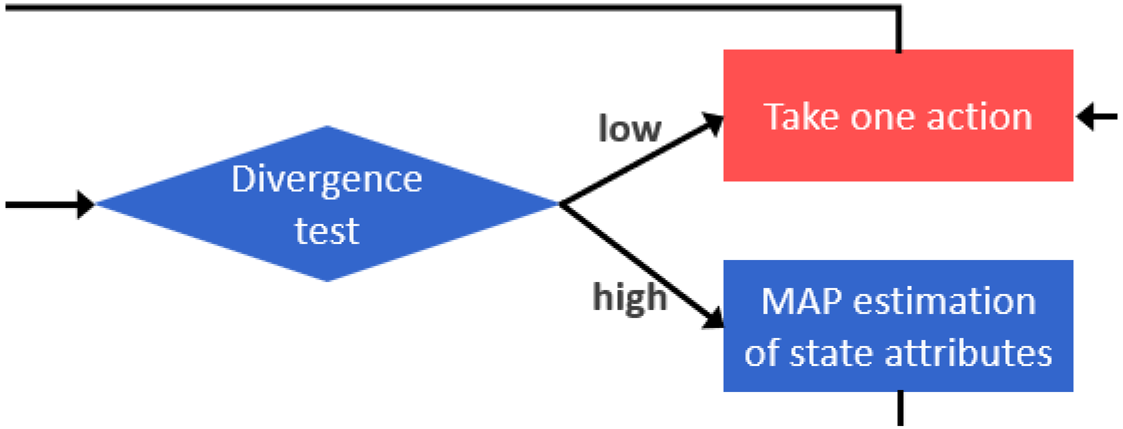
The agent updates its learned model of the environment in a Bayesian approach

- Assumption: Atomic propositions are mutually independent
- Frequentist update if an observation model unavailable

Task-Oriented Active Perception and Planning

The agent checks whether its learned model of the environment has significantly changed

- Jensen-Shannon divergence
- A hyperparameter determining the frequency of replanning



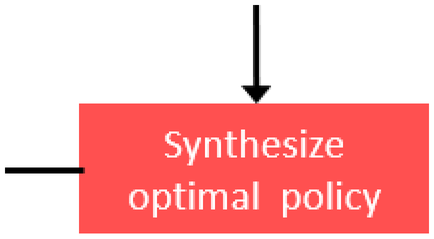
The agent estimates the most probable environment configuration

- According to the current model of the environment
- Maximum a posteriori estimation

Task-Oriented Active Perception and Planning

The agent synthesizes an optimal policy according to the estimated environment configuration

- Generating the product MDP (dynamics + task)
- Computing the optimal policy using a linear program



Task-Oriented Active Perception and Planning

The agent assesses the risk due to the perception uncertainties

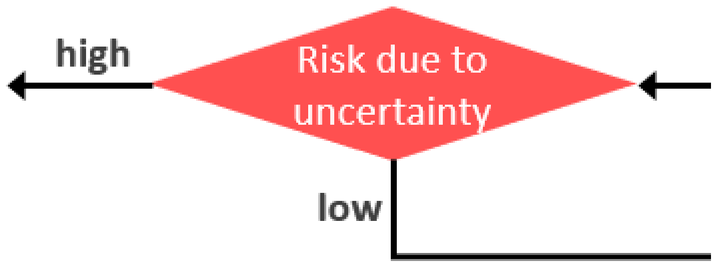
- Statistical verification of the induced Markov chain

$$\hat{\mathbb{E}}_{\mathcal{L}\sim Dist(\mathcal{L})}\left[Pr\left(\mathcal{M}_{\mathcal{D}}^{\pi_t} \models \varphi\right)\right] = \frac{1}{N}\sum_{i=1}^{N} Pr\left(\mathcal{M}_{\mathcal{D}}^{\pi_t} \models \varphi | \mathcal{L}_i\right)$$

- Defining a risk parameter

$$\mathcal{R}(\mathcal{M}_{\mathcal{D}}, \pi_t, \mathcal{L}_t, \varphi) = \left| Pr(\mathcal{M}_{\mathcal{D}}^{\pi_t} \models \varphi \,|\, \hat{\mathcal{L}}_t) - \mathbb{E}_{\mathcal{L}\sim Dist(\mathcal{L})}\left[Pr\left(\mathcal{M}_{\mathcal{D}}^{\pi_t} \models \varphi\right)\right] \right|$$

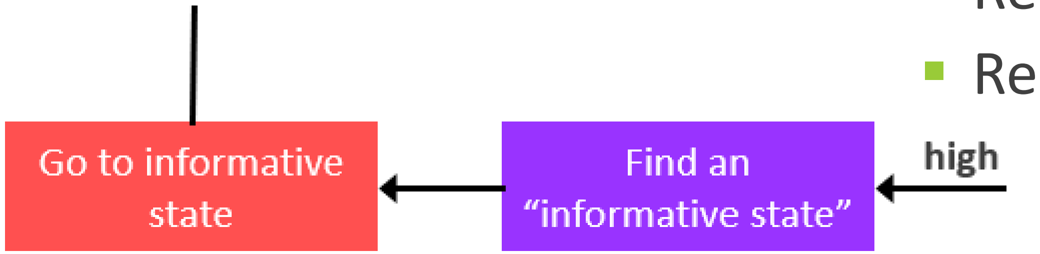
- A hyperparameter determining the willingness of the agent to risk



Task-Oriented Active Perception and Planning

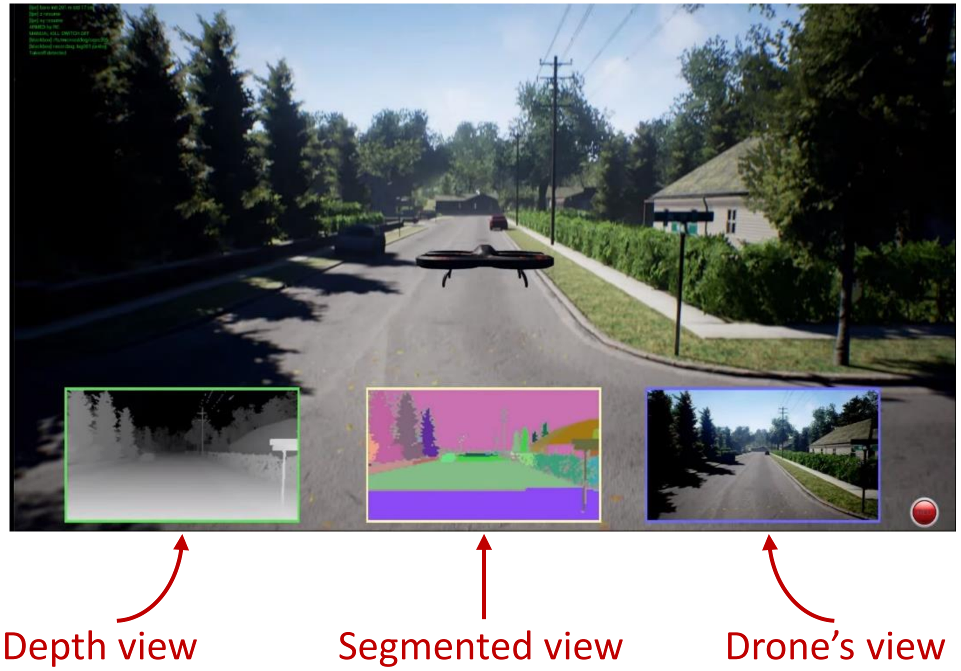
The agent finds an active perception strategy to reduce its perception uncertainty

- Local search over a bounded horizon
- Criteria:
  - Forward and backward reachability
  - Remaining in the same stage of the task
  - Reducing task-related uncertainty



Drone Navigation in Simulated Urban Environment

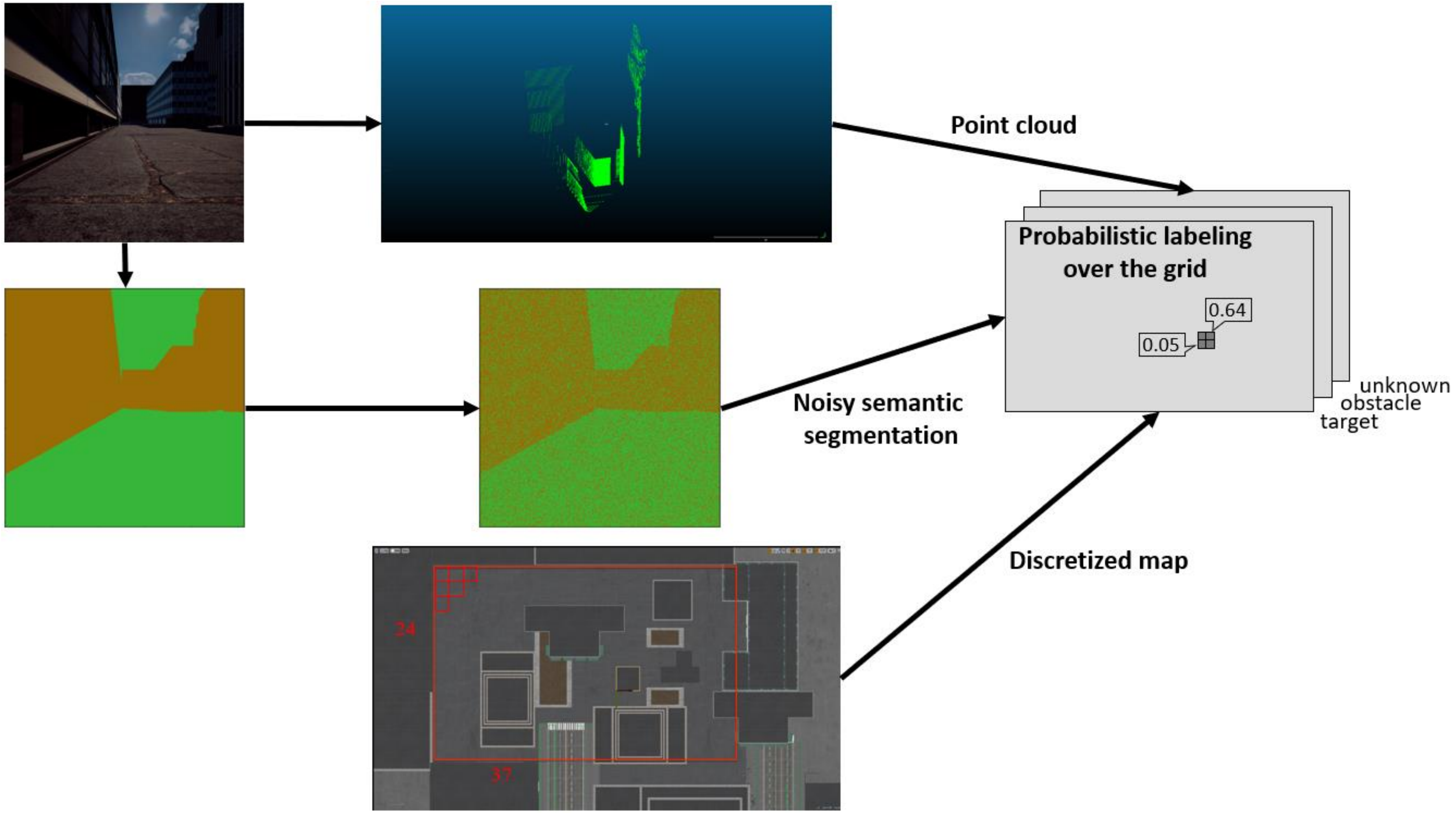
- AirSim[1] simulation environment
- A drone navigating in an urban environment
- **Task:** Reach a flagged building while avoiding collision
- **Dynamics:** Planar motion with constant altitude



- **Sensing:**
  - Exact localization
  - 4 RGB cameras with 90° field of view
  - 4 depth sensing cameras with 90° field of view

[1] From https://github.com/microsoft/AirSim

Processing Image and Depth Data



Simulation Results

Navigation with exact knowledge of the semantic labeling

Navigation with the proposed task-oriented active perception and planning

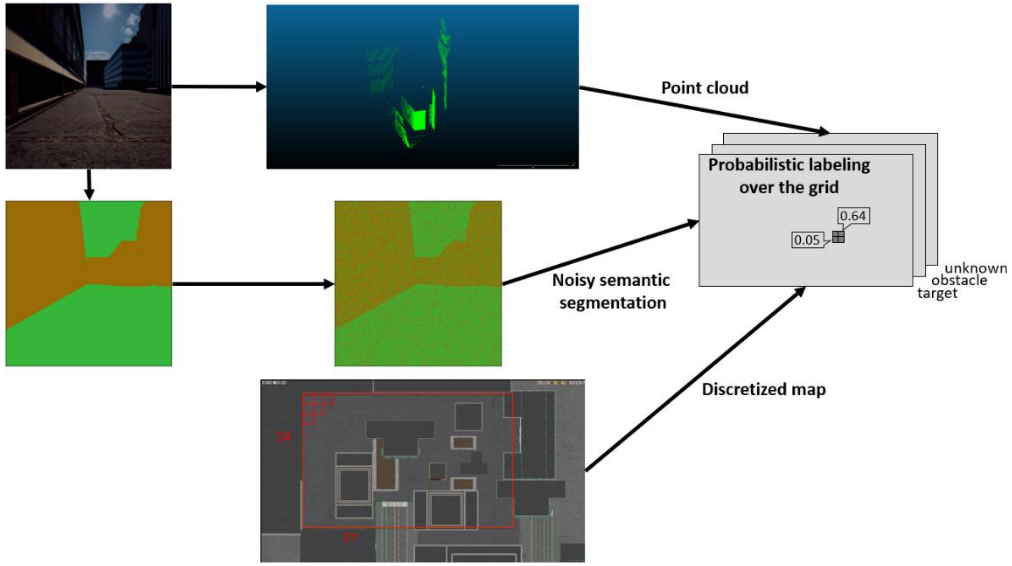
Conclusion and Future Directions

Conclusion:

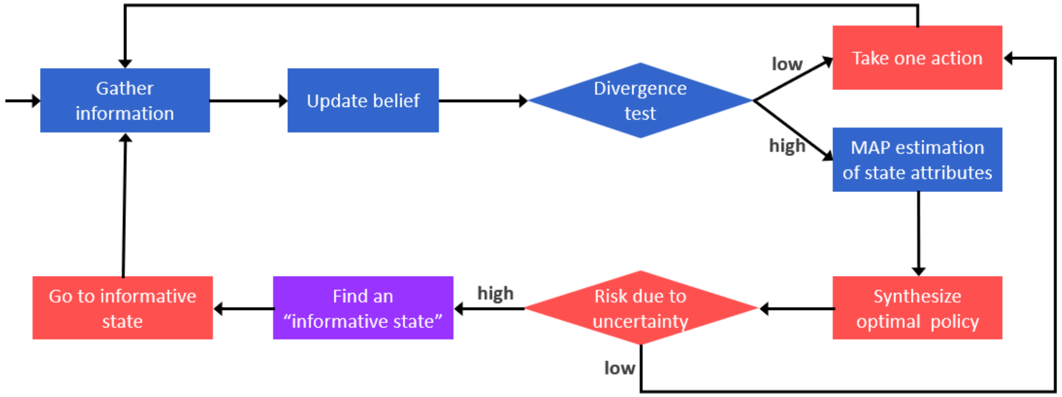
- Studied planning in environments with partially known semantics
  - Guarantee over task performance
  - Assessment of risk due to imperfect knowledge
- Proposed a task-oriented active perception and planning framework that integrates learning through perception with decision-making under uncertainty

Future Directions:

- Extending the framework to settings with uncertain or unknown dynamics
- Using calibrated neural networks for perception module
- Incorporating side knowledge on the correlation between the atomic propositions



Thank you!



Task-Oriented Active Perception and Planning in Environments with Partially Known Semantics

Mahsa Ghasemi, Erdem Arınç Bulgur, and Ufuk Topcu