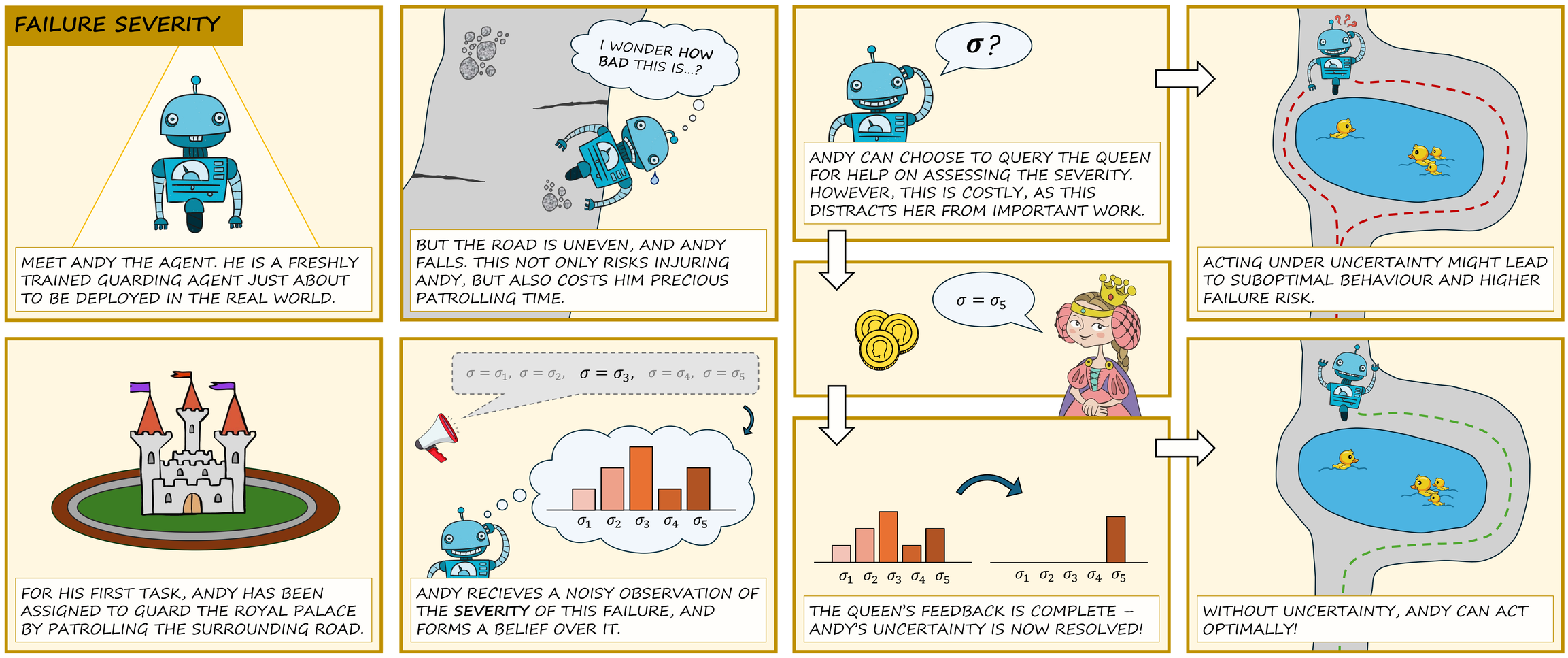
Planning under Uncertain Failure Severity

Rebecka Winqvist, rebwin@kth.se

Anna Gautier, Jana Tumova

KTH VETENSKAP OCH KONST

WASP WALLENBERG AI, AUTONOMOUS SYSTEMS AND SOFTWARE PROGRAM



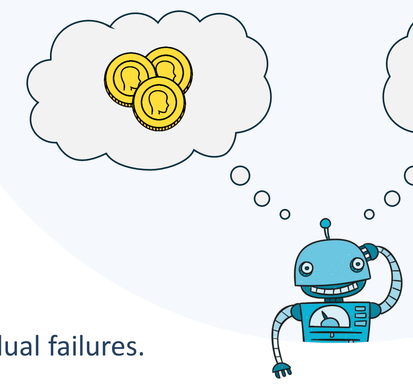
BACKGROUND

An agent operates in a known domain $\mathcal{D}$, described by a Goal-MDP:

$$\mathcal{D} = \langle \mathcal{S}_d, \mathcal{A}_d, T_d, R_d, \mathcal{G}_d \rangle.$$

Each action failure is associated with a **severity level** $\sigma \in \Sigma$ that reflects its impact on the agents ability to reach its goal.

- The agent receives **noisy observations** of the severity level of individual failures.
- Severity from multiple failures **accumulate over time**.
- The agent keeps a belief over the **total accumulated** severity ξ.
- The agent can choose to **query for expert feedback** on ξ, but only *right after* a failure occurs.



APPROACH

The domain $\mathcal{D}$ is known, the **accumulated severity** ξ is not.

We model this using a **Mixed Observation Markov Decision Process (MOMDP)** with conditional observations:

$$\mathcal{M} = \langle \mathcal{S}, \mathcal{S}_O, \mathcal{A}, T, R, \mathcal{O}, O, s_0, b_0, \mathcal{T}, \mathcal{G} \rangle .$$

- Observable states factors: domain state $\mathcal{S}_d$, domain action outcome
- Hidden state factors: accumulated severity ξ
- Actions: domain actions $\mathcal{A}_d$ and query action
- Observations are received in states:

$$\mathcal{S}_O^{(1)} = \{\text{action outcome} = \text{"fail"}, \text{ no query}\} \quad \textit{(noisy, σ)}$$
$$\mathcal{S}_O^{(2)} = \{\text{action outcome} = \text{"fail"}, \text{ query}\} \quad \textit{(complete, ξ)}$$

Problems:

- Acting under uncertainty may lead to suboptimal performance
- Expert feedback is costly and not uniformly available

Question: How do we best account for failure severity during planning?

Fail-then-Query (FTQ) Algorithm

- Domain actions $\mathcal{A}_d$ only influence state transitions in $\mathcal{D}$.
- Query actions resolve uncertainty.

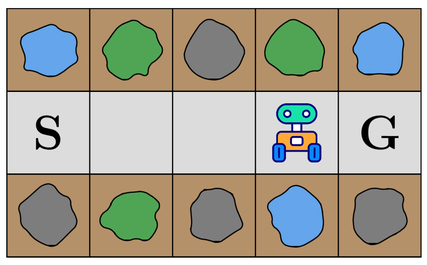
↳ Use Q_{MDP} heuristic and evaluate query value after failures!

PRELIMINARY RESULTS

Simulation

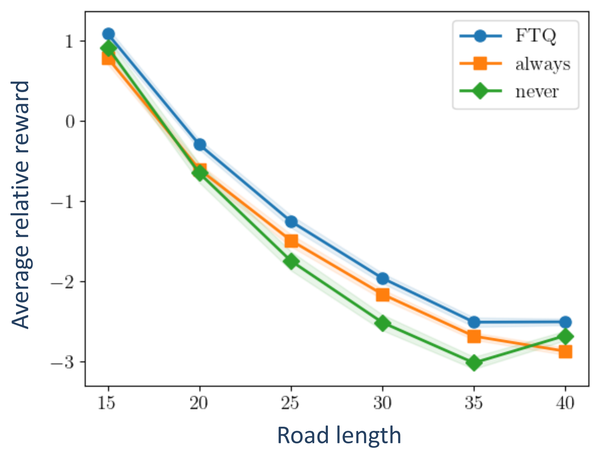
We evaluate our FTQ planner in a **Rocky Road[1] environment** of varying lengths.

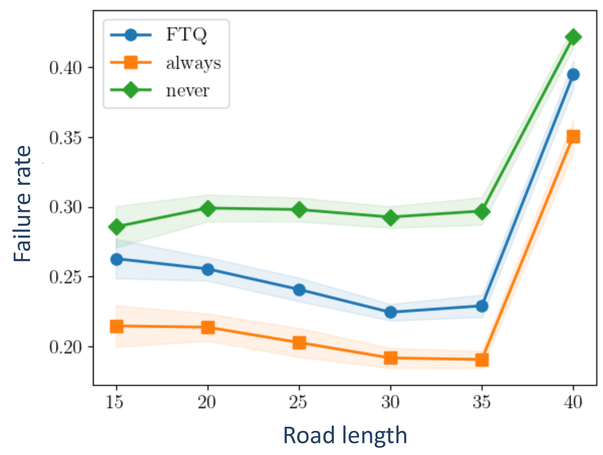
The Rocky Road landscape features a mix of safe and hazardous terrains. Starting from the initial position (S), the agent's objective is to reach the goal position (G), while avoiding hazardous terrain along the way, as this terrain type may cause damage to the agent.



A Rocky Road environment of length 5.

Comparing FTQ to two baselines: "always query" & "never query"





[1]*Adapted from OpenAI Gym's Frozen Lake environment:* G. Brockman et al., OpenAI Gym.arXiv:1606.01540

Icons by Freepik and by creator logturnal on Freepik