Runtime Monitoring under Uncertainty in Autonomous Cyber-Physical Systems

Hazem Torfah

Department of Computer Science and Engineering
Chalmers University of Technology

@Workshop on Formal Verification and Learning for Quantitative Systems | FSTTCS 25 | Dec 16, 2025

WASP | WALLENBERG AI, AUTONOMOUS SYSTEMS AND SOFTWARE PROGRAM

CHALMERS UNIVERSITY OF TECHNOLOGY

Autonomous systems today …

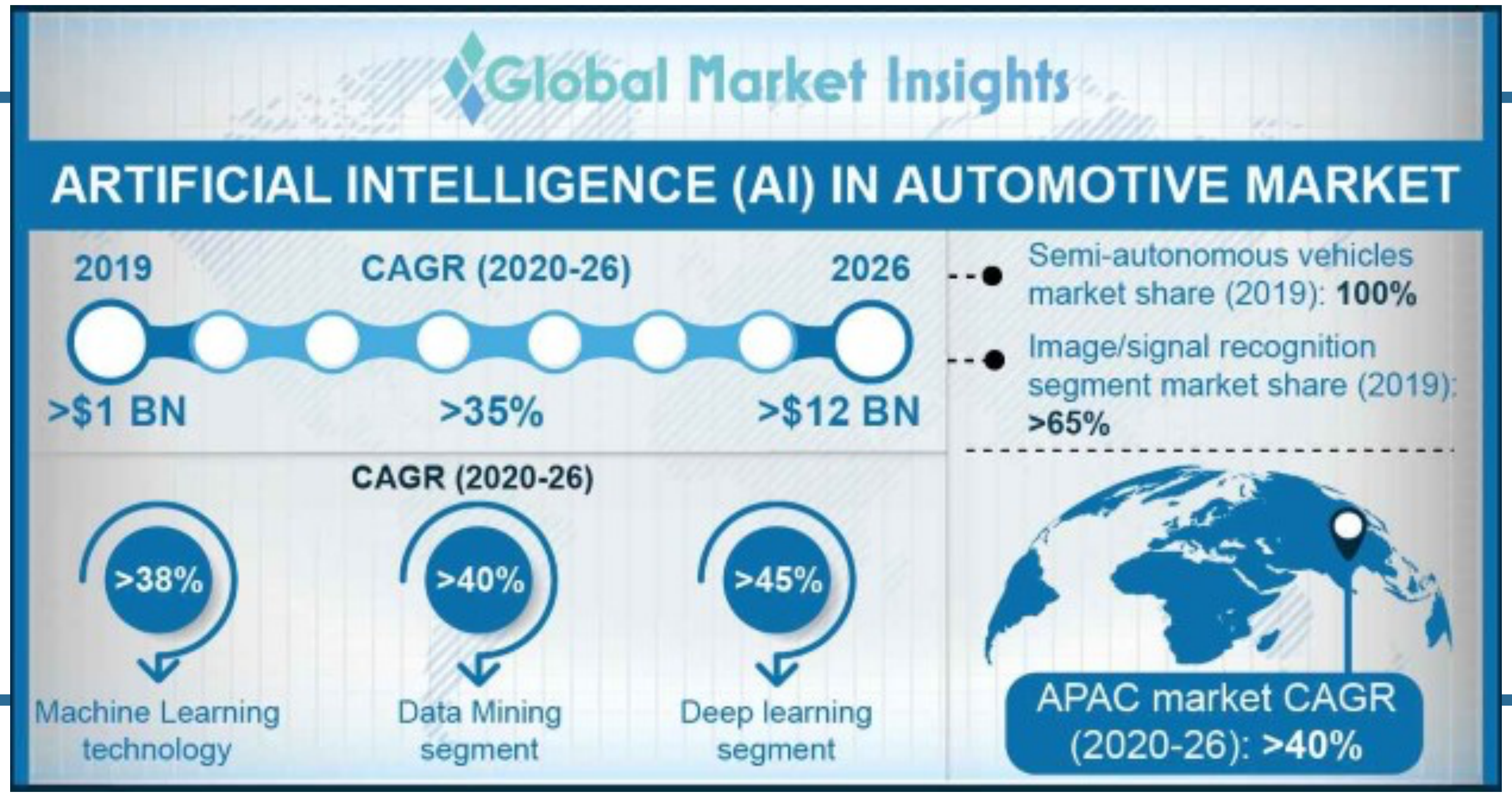
http://volvotrucks.com

https://jpl.nasa.gov/

https://waymo.com

Increased ML in Autonomy

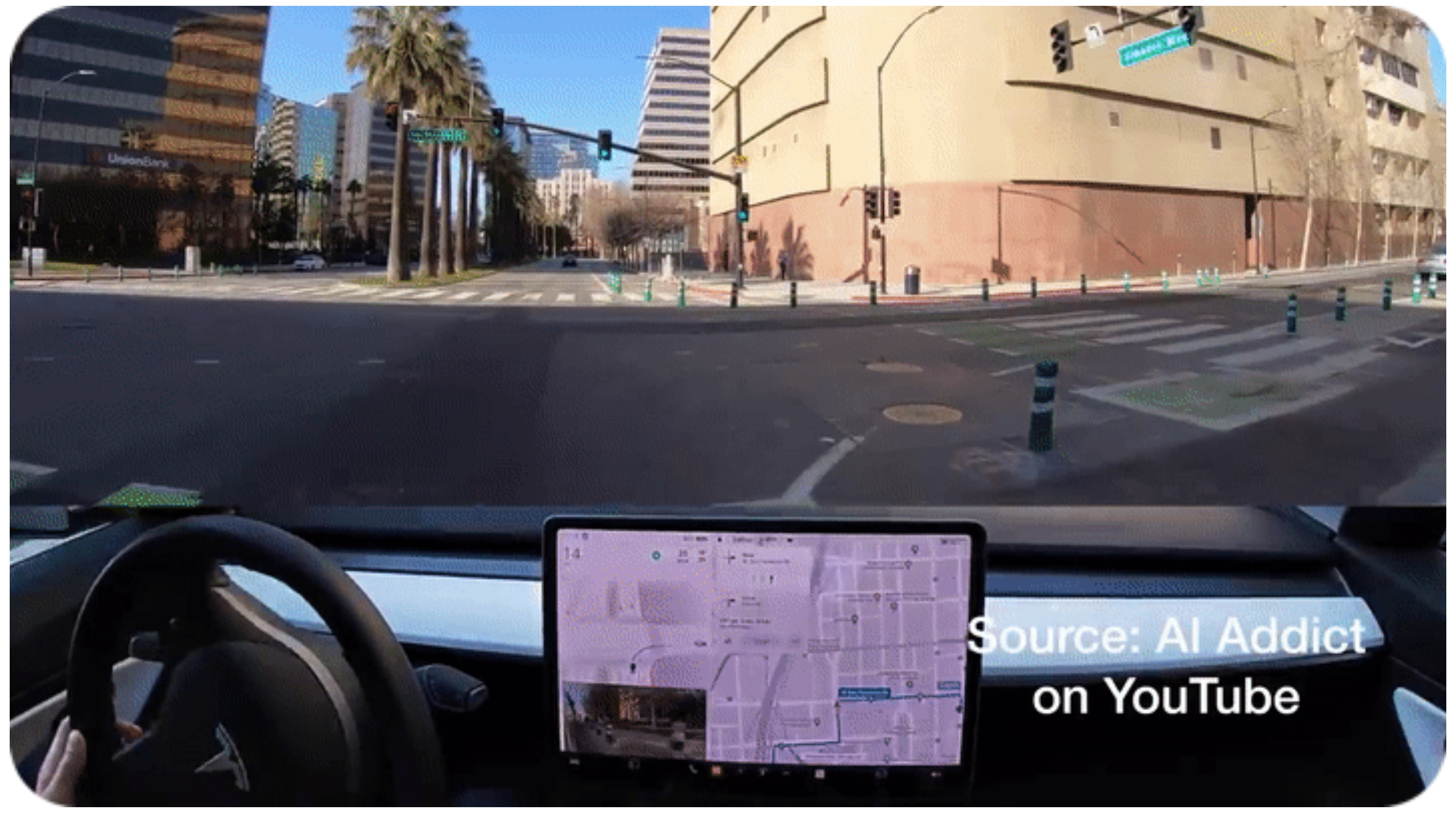
- **Trend:**
  - Data-driven machine learning techniques are essential to perform perception tasks
  - The use of ML techniques is projected to grow



gminsights.com

Autonomous systems today …

http://nvidia.com



Bell APT Autonomous Cargo Drone Crashes in Texas

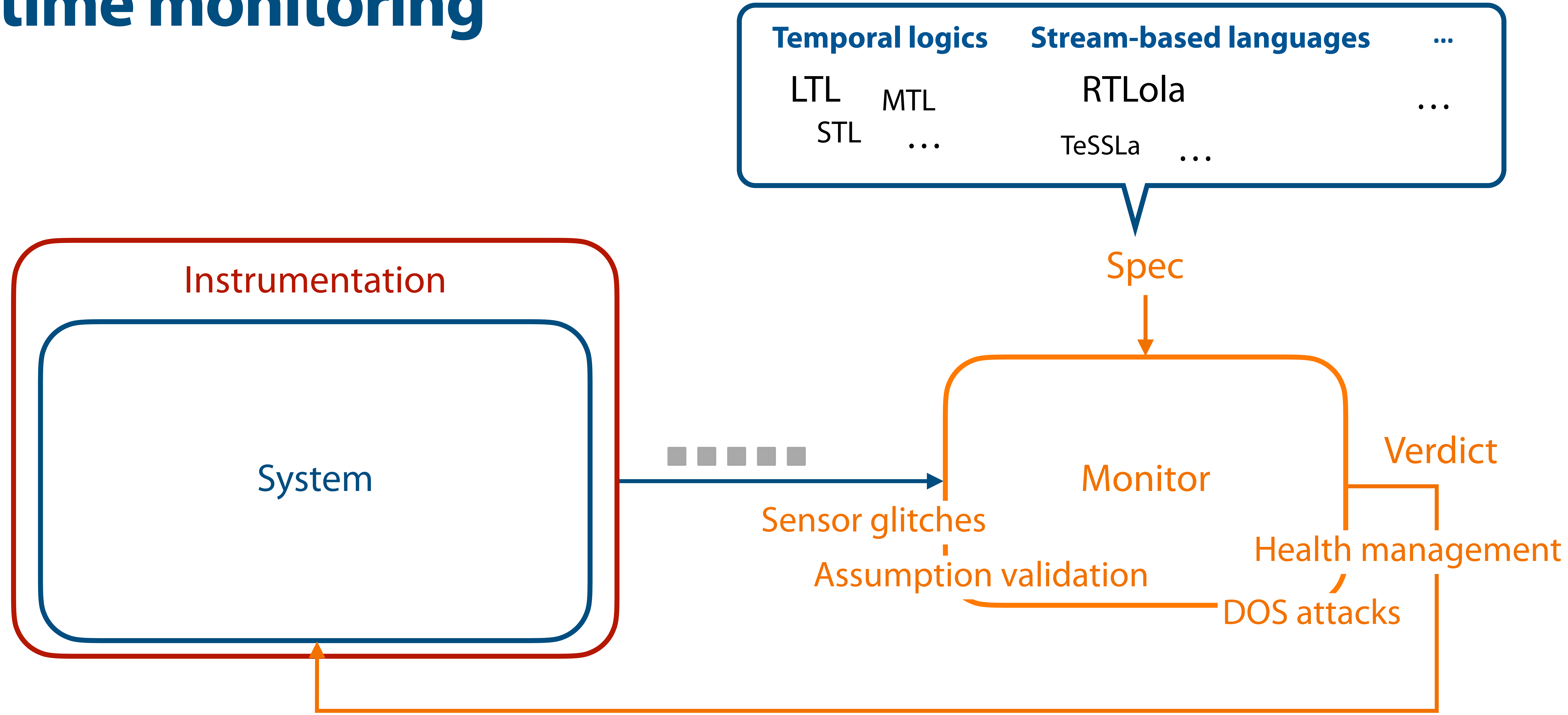
Cruise to pay $1.5M penalty in connection with San Francisco pedestrian accident, NHTSA says

…

- **Growing concerns:**
  - Unpredictability hinders deployment and adoption in safety-critical applications
  - Neural models (e.g. DNNs) are brittle: unanticipated changes in the environment may cause faulty behavior

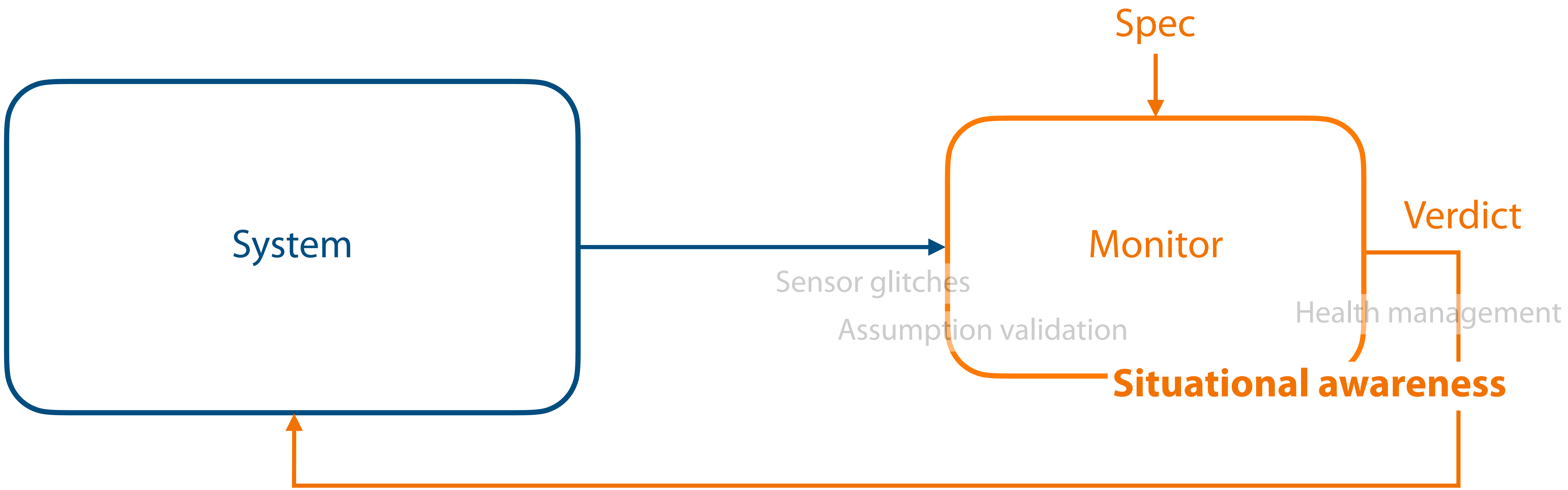
How can we raise assurance in such systems?

Runtime monitoring

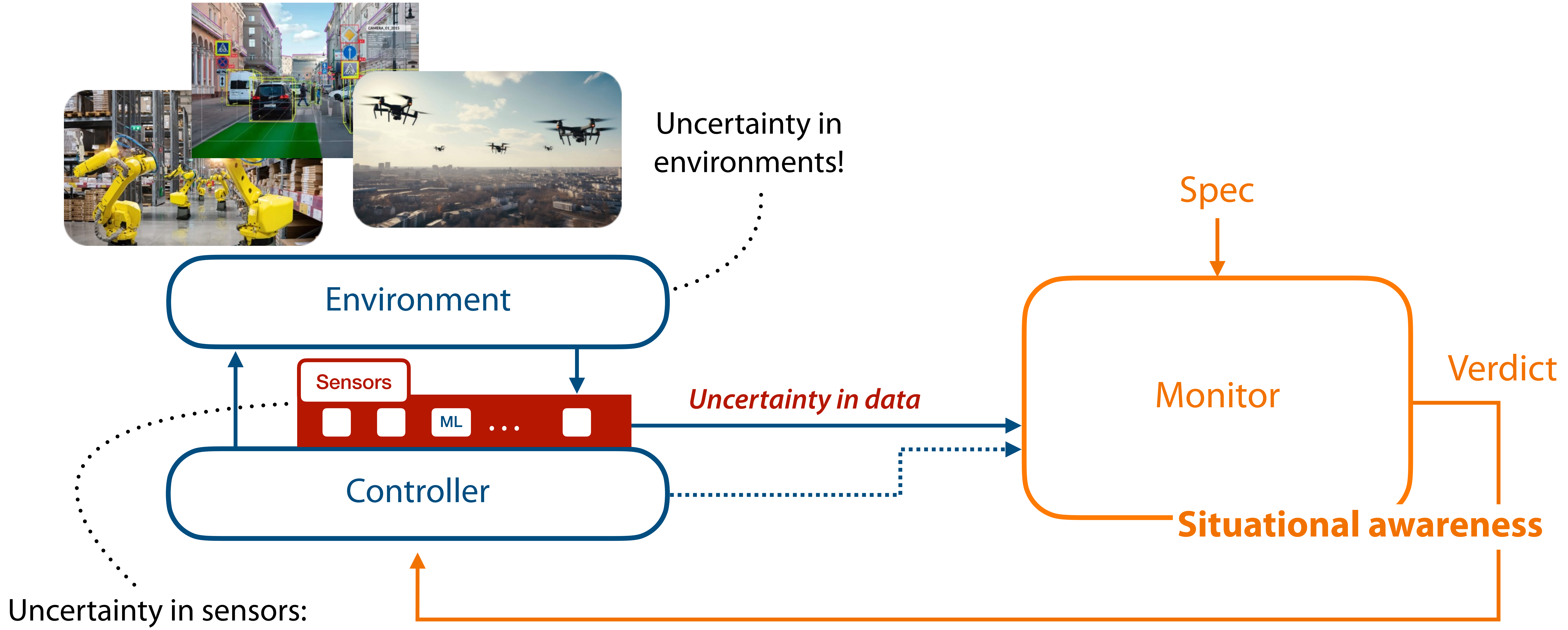


Runtime monitoring is a technique based on ***extracting*** information from a system at runtime and ***evaluating*** this information to determine whether an execution of the system satisfies or violates a given property, and consequently ***deploying*** the necessary recovery mechanisms.

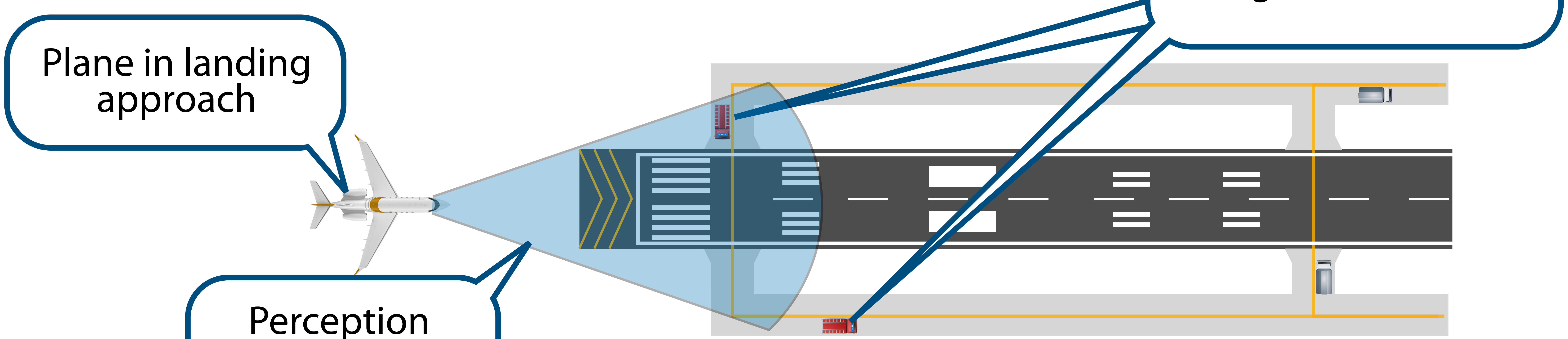
Runtime monitoring in autonomous systems



Runtime monitoring in Autonomous Systems

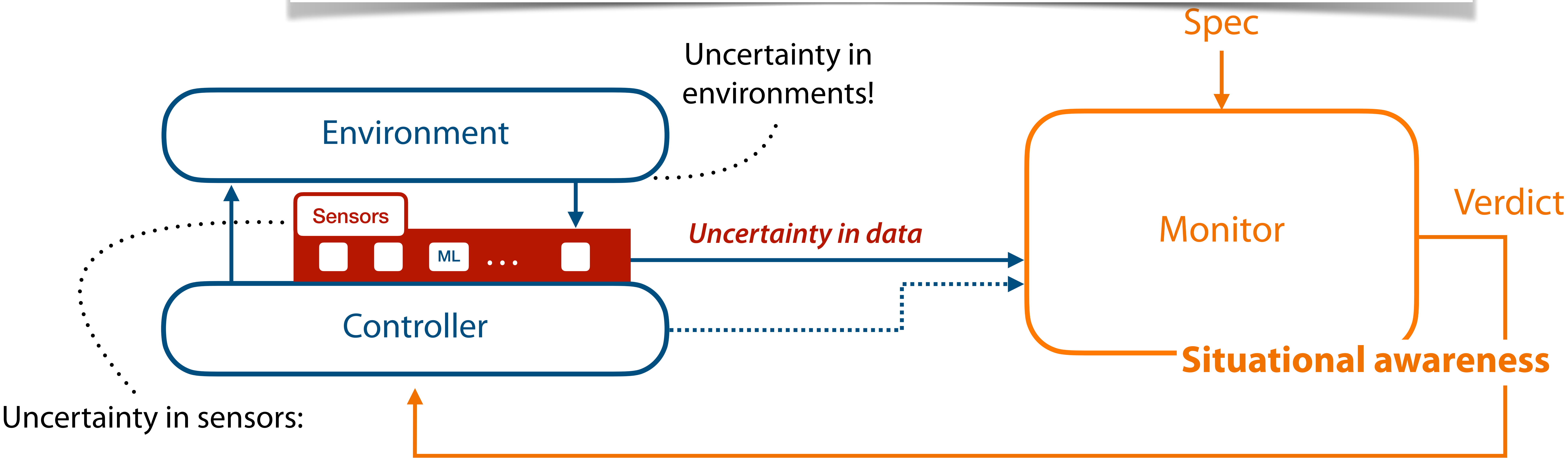


Runtime monitoring in autonomous sy

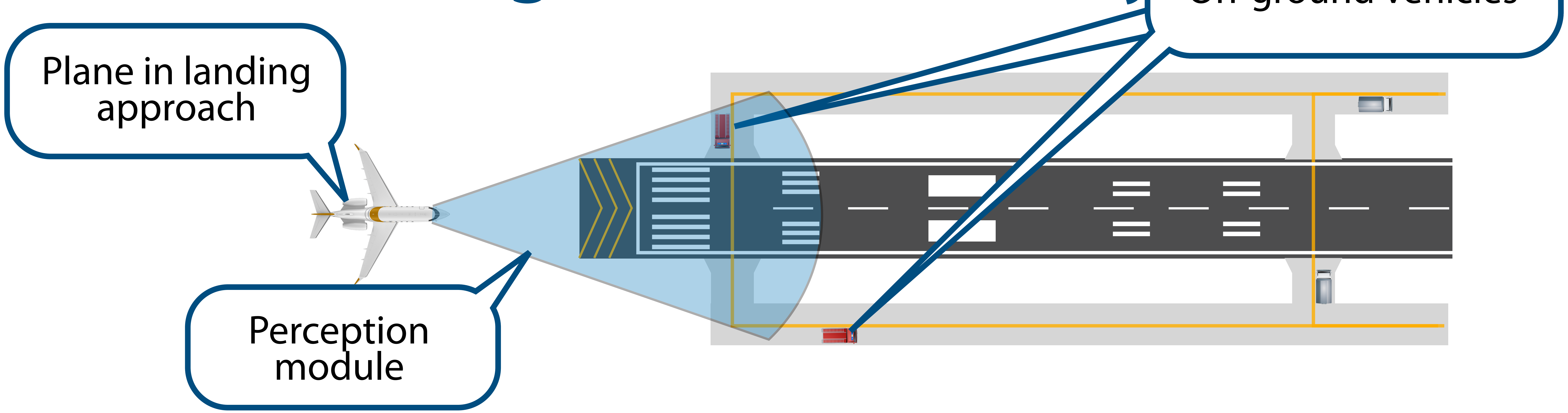


Example is inspired by a challenge problem provided by Boeing in the DARPA Assured Autonomy program.

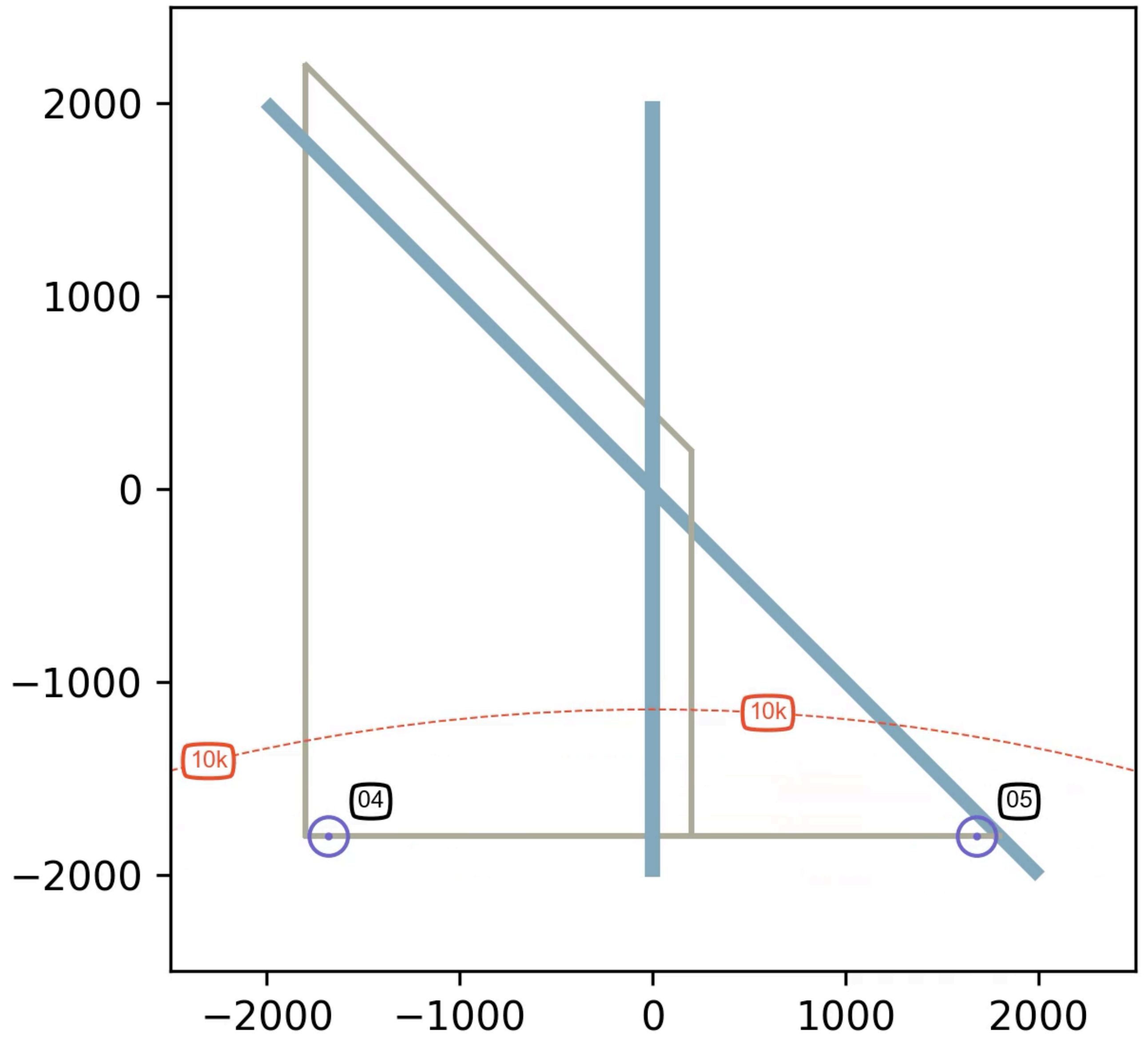
Runtime monitor: Estimate the risk of colliding with one of the vehicles

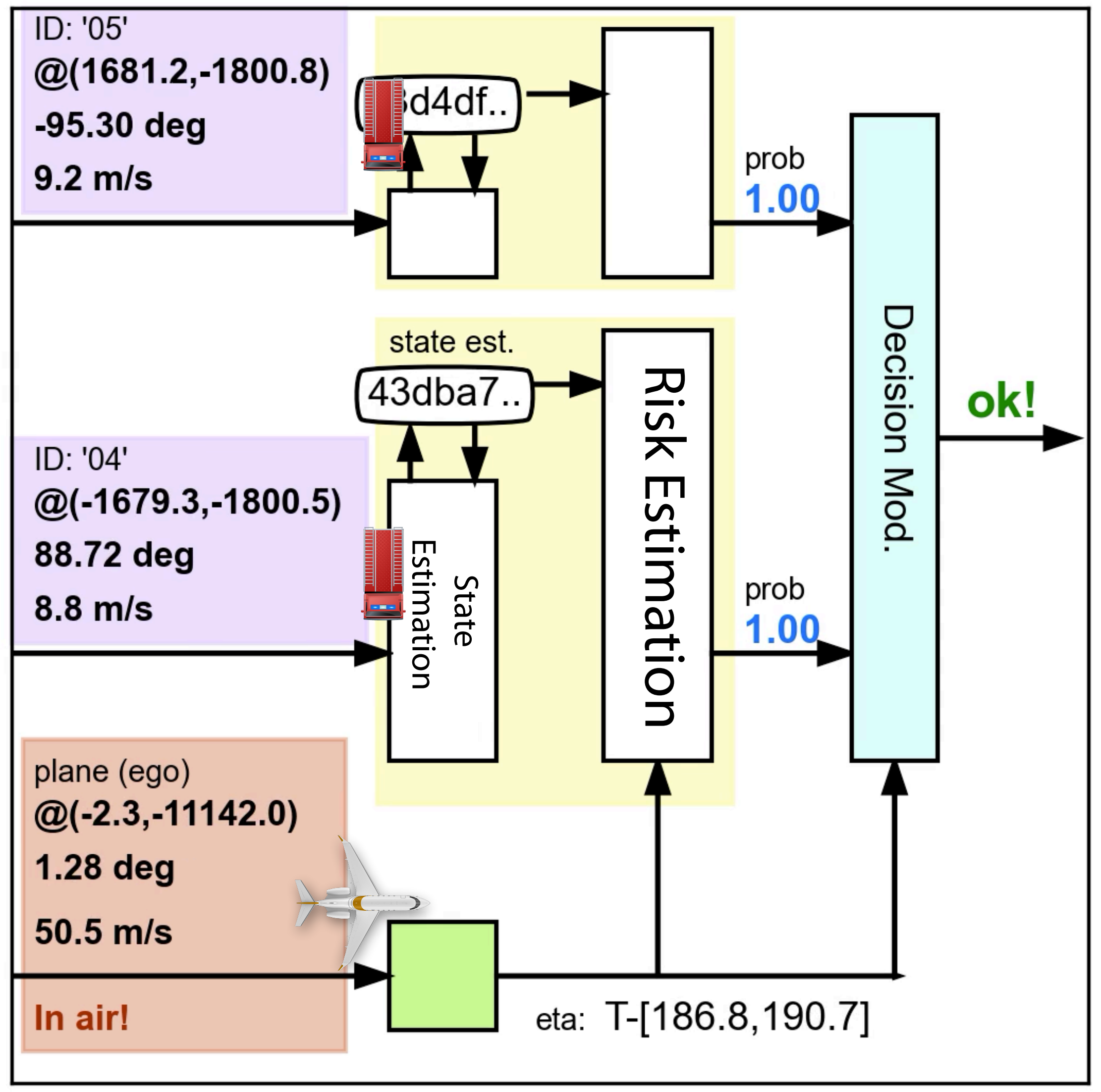


Runtime monitoring in autonomous sy

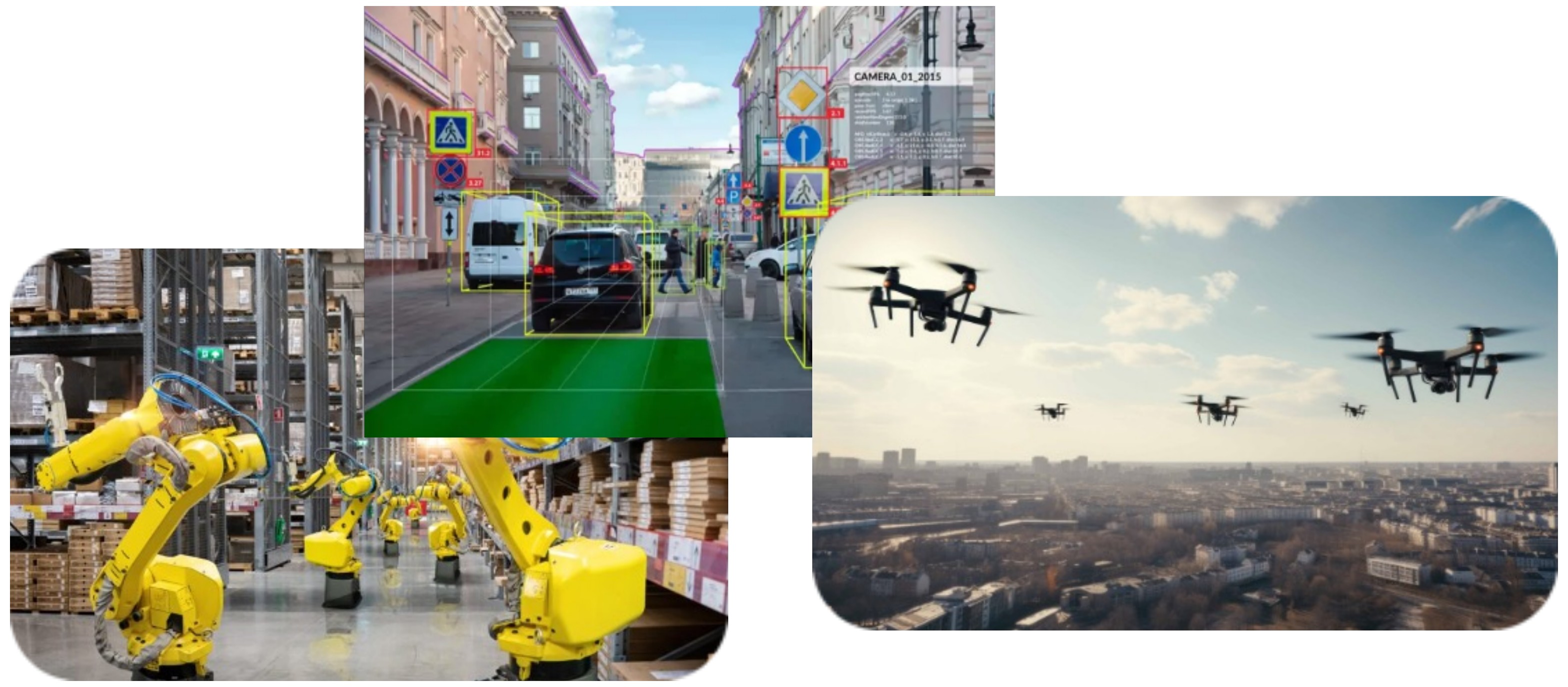


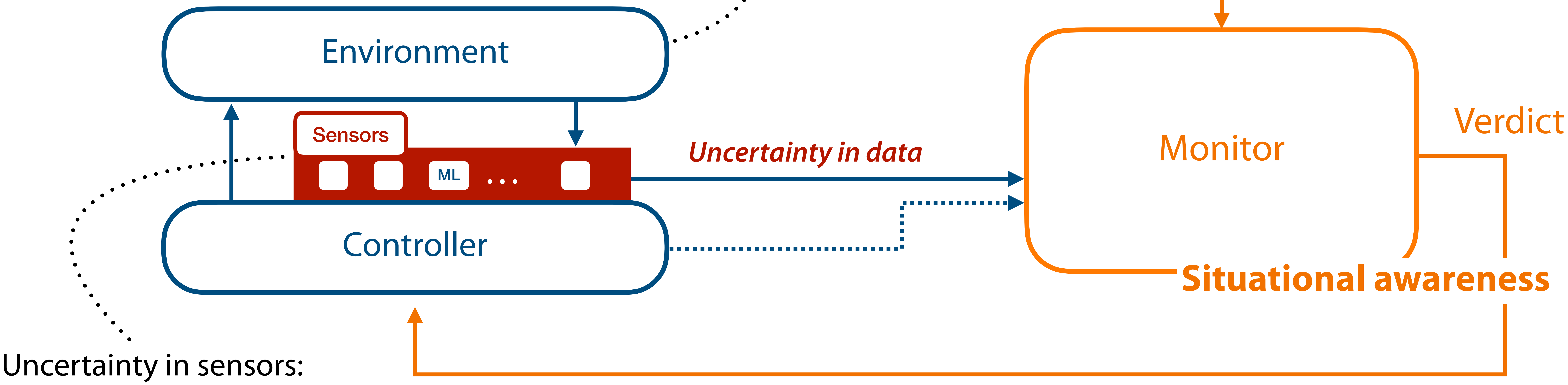
Runtime monitor: Estimate the risk of colliding with one of the vehicles





Runtime monitoring in autonomous systems





How can we build monitors that are resilient to uncertainty in the information received about the environment?

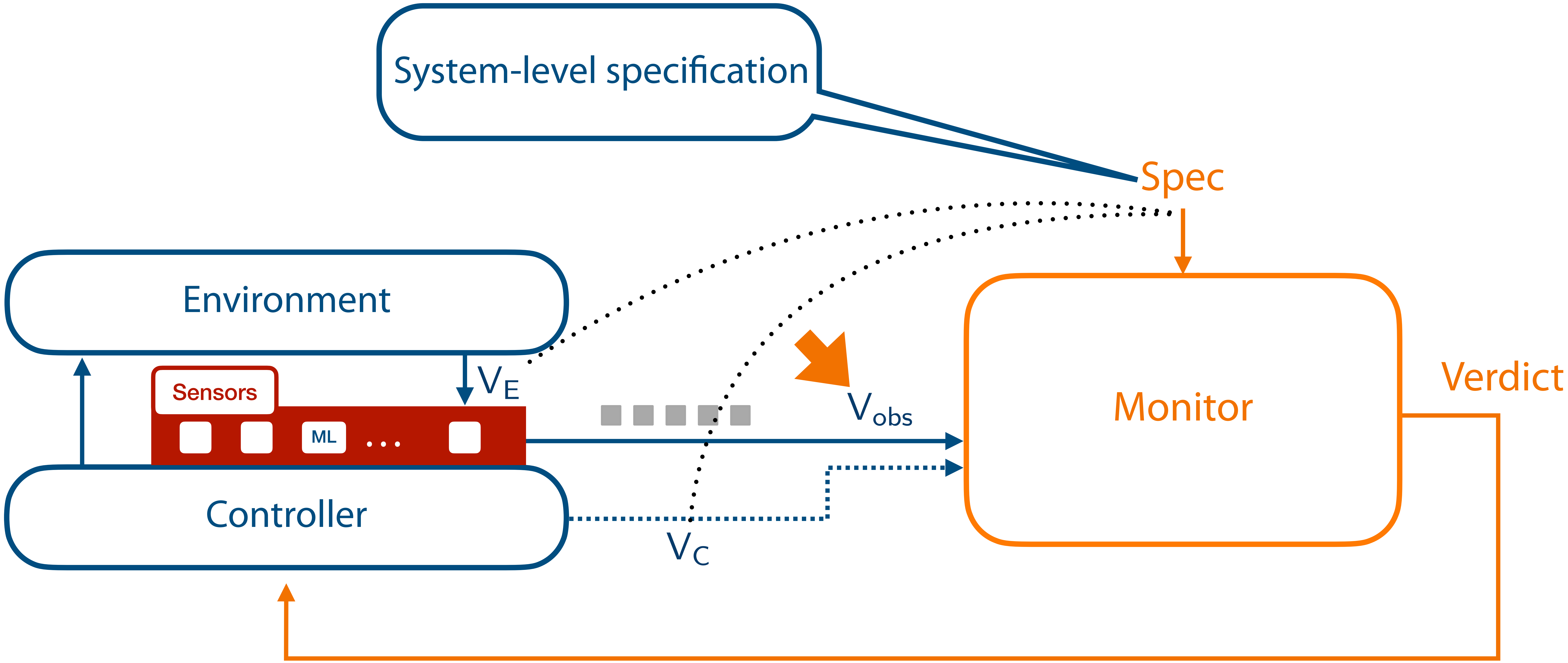
What is the risk of issuing a verdict?

In this talk

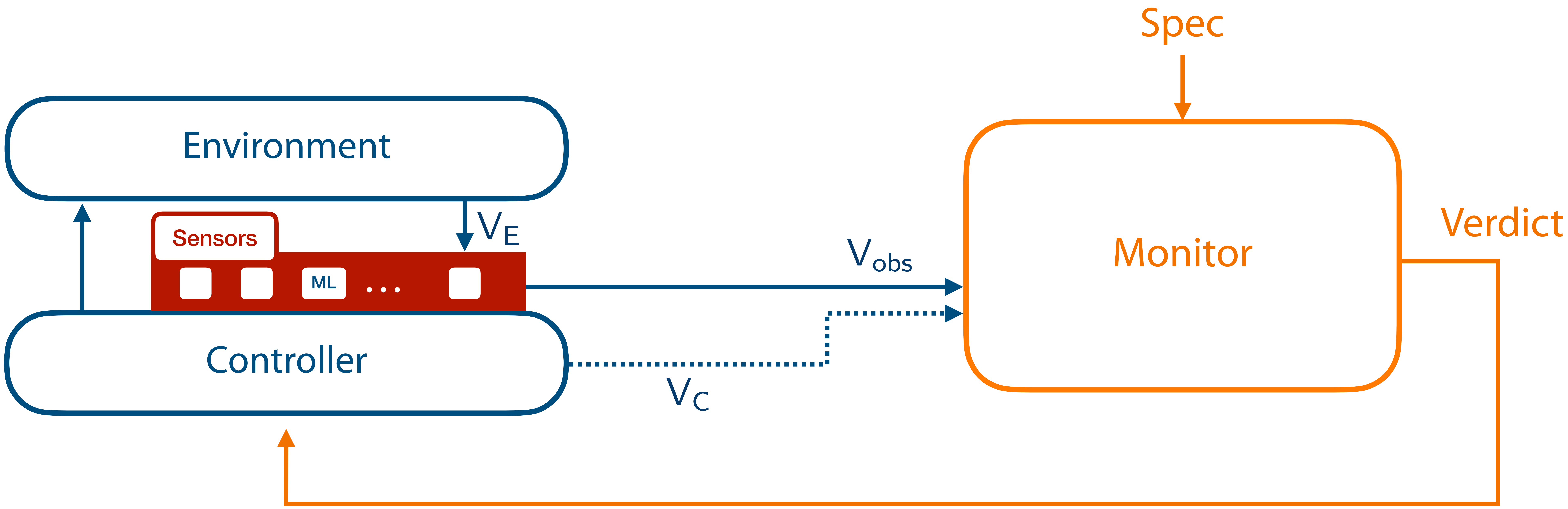
- **Runtime monitoring under uncertainty**
  - *formal problem statement*
- **Monitoring over Markov decision processes**
  - *from exponential to polynomial complexity*
- **Learning robust uncertainty models for monitoring**
  - *point-based models vs interval-based models*
  - *model-based vs model-free*

Problem statement

Runtime assurance in AI-based autonomy



Problem Statement



Let: $V_{sys} = V_E \cup V_C$ and $\Sigma_{sys}, \Sigma_{obs}$ denote the set of valuations over the variables V_{sys}, V_{obs}

Problem Statement: Given a distribution $d \in \mathrm{Dist}(\Sigma^*_{sys})$, a mapping $\mu\colon \Sigma^*_{sys} \to \mathrm{Dist}(\Sigma^*_{obs})$, a risk function $r\colon \Sigma^*_{sys} \to \mathbb{R}$, and a constant λ, learn a monitor M such that:

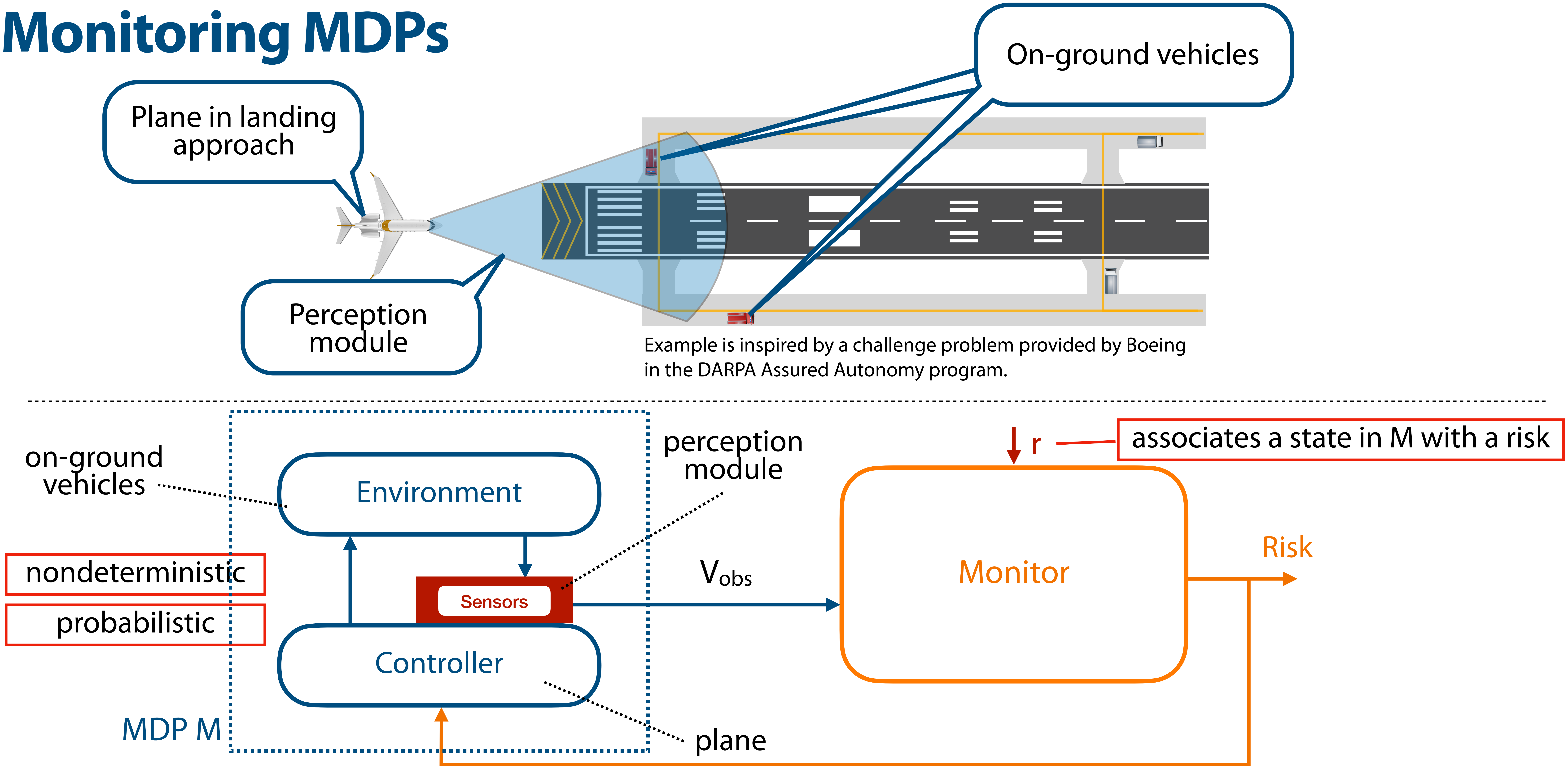
$$\forall \sigma_{obs}.\ M(\sigma_{obs}) \Leftrightarrow \sum_{\sigma_{sys} \in \Sigma^*_{sys}} d(\sigma_{sys}) \cdot \mu(\sigma_{sys})(\sigma_{obs}) \cdot r(\sigma_{sys}) > \lambda$$

Junges, Seshia, Torfah. Active Learning of Runtime Monitors under Uncertainty. iFM 2024

Monitoring over Markov decision processes

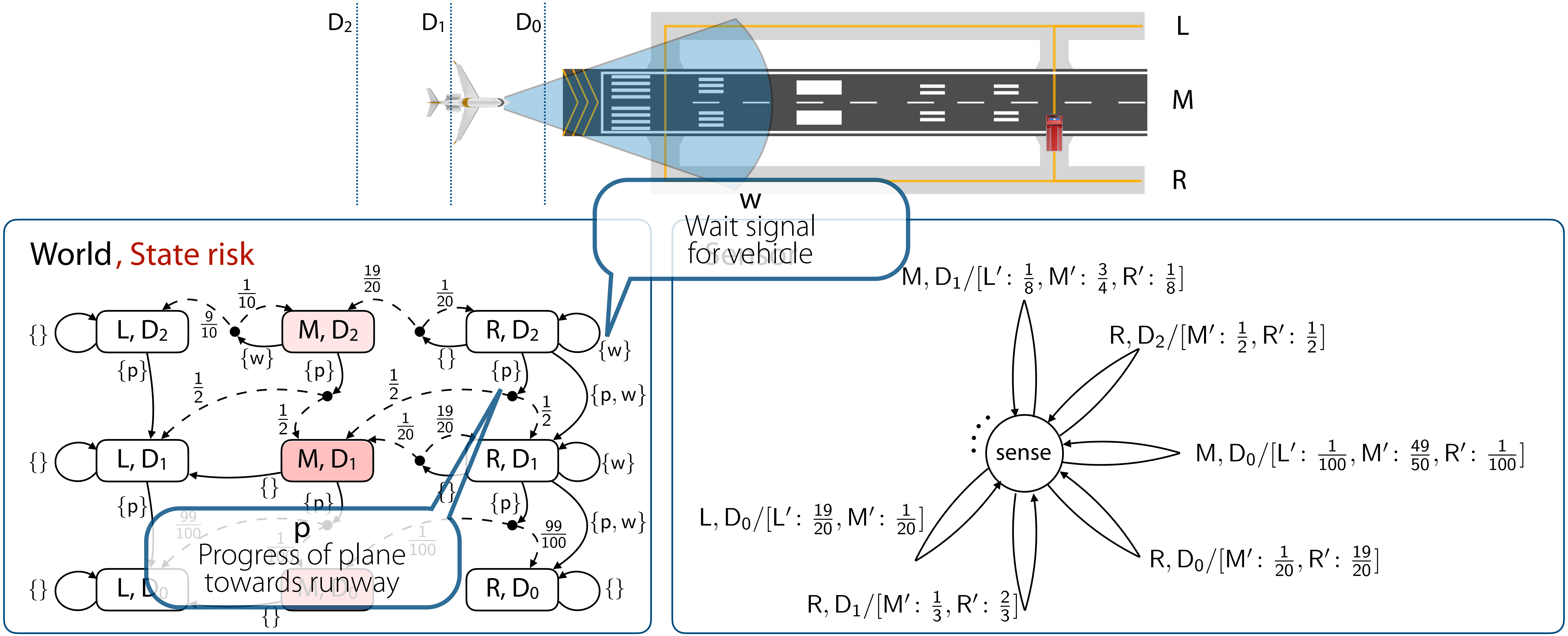
Junges, Torfah, Seshia. CAV 2021

Monitoring MDPs

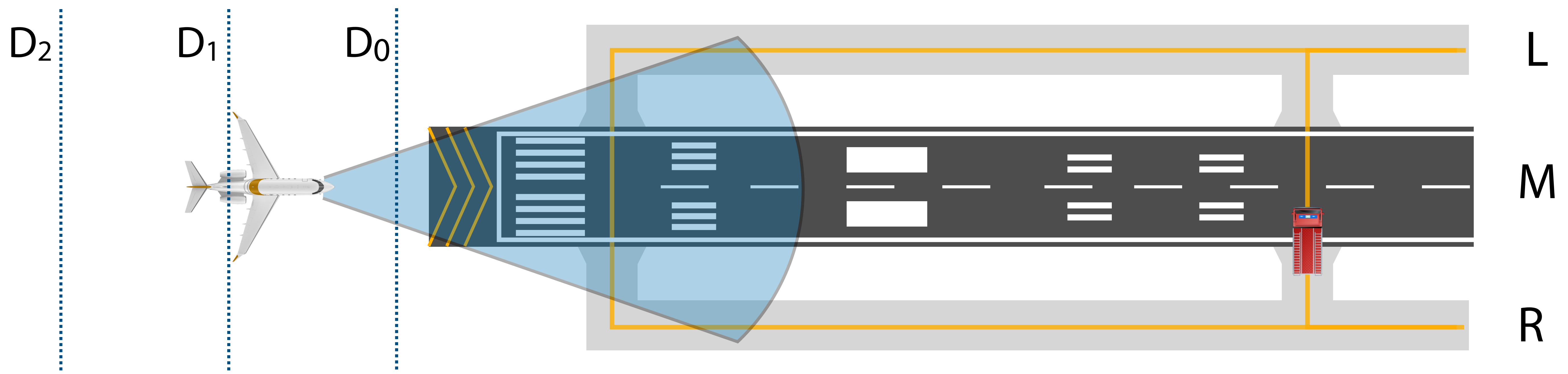


Given M and r, what is the risk after observing a trace τ over V_{obs} ?

Monitoring MDPs



Monitoring MDPs



World, State risk

W, r

Sensor

S

$$W, r \otimes S \rightarrow M, r$$

What is the risk after observing a trace τ?

$$\mathsf{Risk}(\tau) = \sup_{\sigma \in \mathsf{Strategies}(M)} \sum_{\pi \in \mathsf{Paths}(M, |\tau|)} \mathsf{Pr}_\sigma(\pi \mid \tau) \cdot \mathsf{r}(\mathsf{last}(\pi))$$

Algorithmic Approach

Filtering

Too expensive
Needs to track
exponential number of beliefs.

- Algorithmic analysis of nonlinear hybrid systems. IEEE Trans. Autom Control 1998. T. Henzinger et al.

- Runtime Verification with State Estimation. RV 11. S. Stoller et. al -

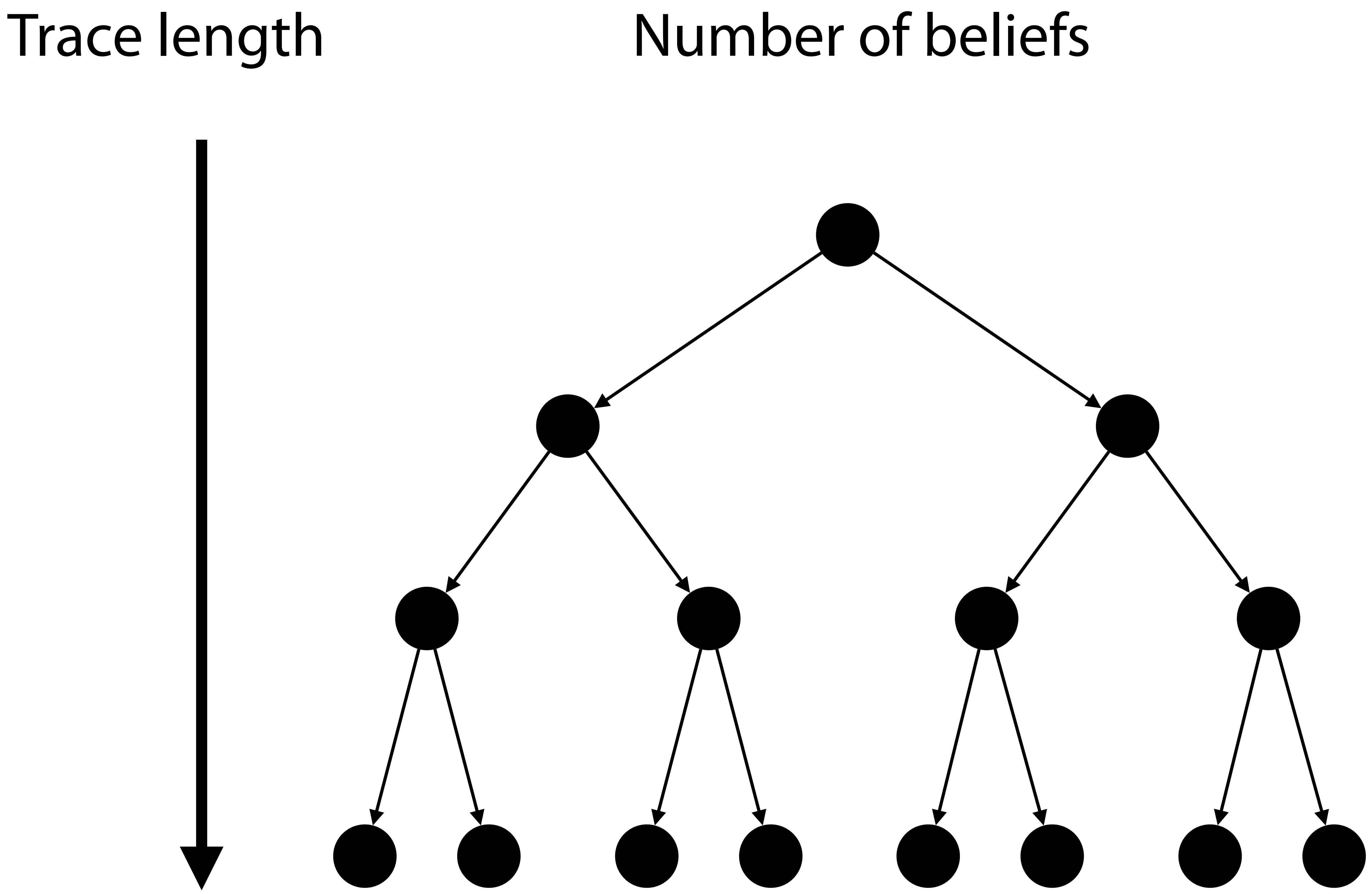
- Runtime Verification of Stochastic, Faulty Systems. RV 10. M. Wilcox, B Williams -

- Monitoring Temporal Properties of Stochastic Systems. VMCAI 2008. A. Sistla, A.R. Srinivas -

Algorithmic Approach

Filtering

Too expensive
Needs to track
exponential number of beliefs.



Algorithmic Approach

Filtering	Filtering with Convex Hull	Unrolling with Model checking
Too expensive **Needs to track** **exponential** number of beliefs.	Only tracks vertices of the convex hull. **Reduces number of beliefs dramatically**	Unroll MDP by the length of the trace then compute conditional reachability probability. **Runs in polynomial time**

- *Algorithmic analysis of nonlinear hybrid systems. IEEE Trans. Autom Control 1998. T. Henzinger et al.*
- *Runtime Verification with State Estimation. RV 11. S. Stoller et. al -*
- *Runtime Verification of Stochastic, Faulty Systems. RV 10. M. Wilcox, B Williams -*
- *Monitoring Temporal Properties of Stochastic Systems. VMCAI 2008. A. Sistla, A.R. Srinivas -*

Algorithmic Approach

Filtering

Too expensive!
Needs to track
exponential number of beliefs.

Filtering with
Convex Hull

Only tracks vertices
of the convex hull.
Reduces number of beliefs dramatically

Based on a geometric interpretation
of distributions

Unrolling with
Model checking

Unroll MDP by
the length of the trace then compute
conditional reachability probability.
Runs in polynomial time

Convex-hull Forward Filtering

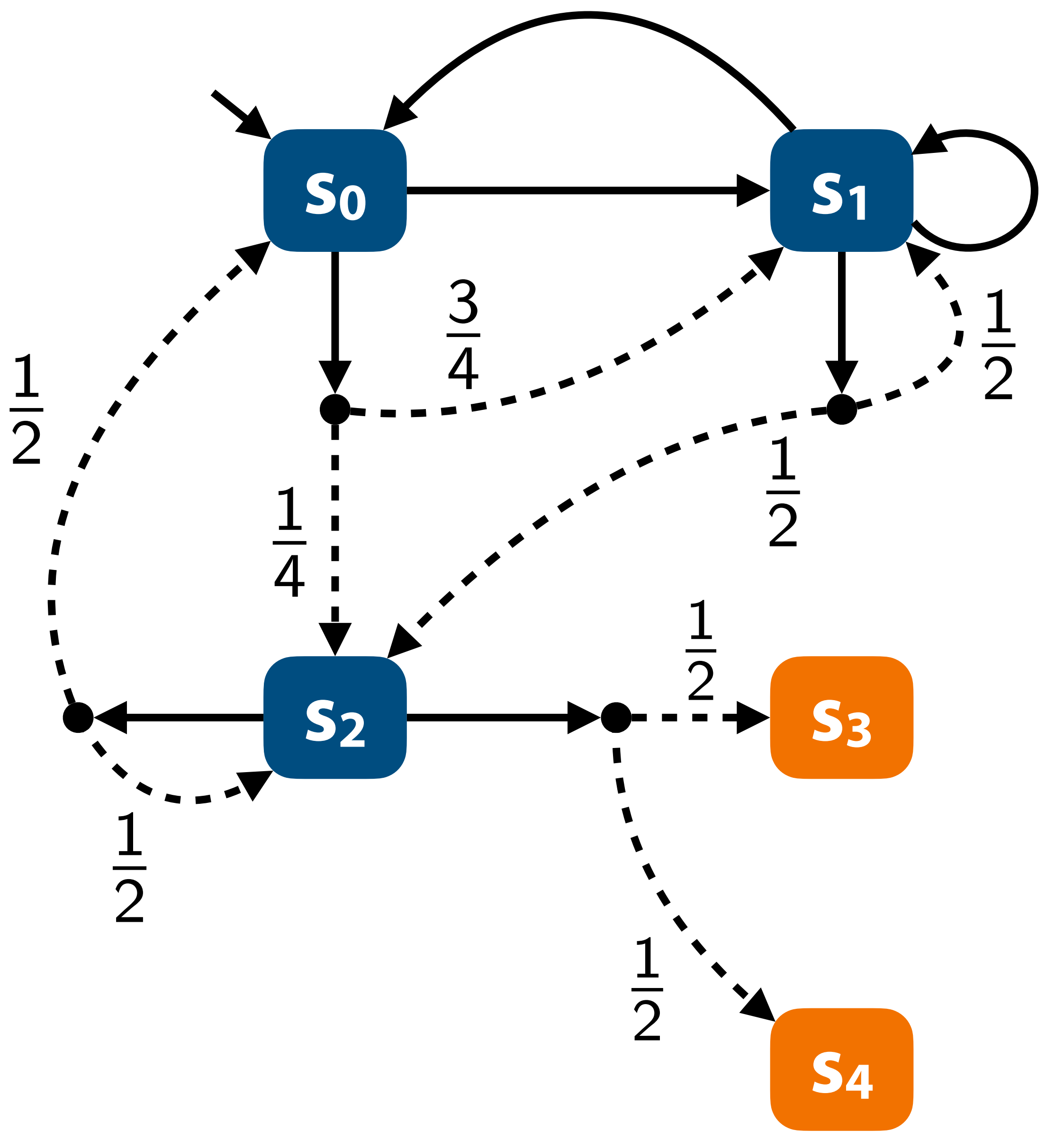
Number of reached states

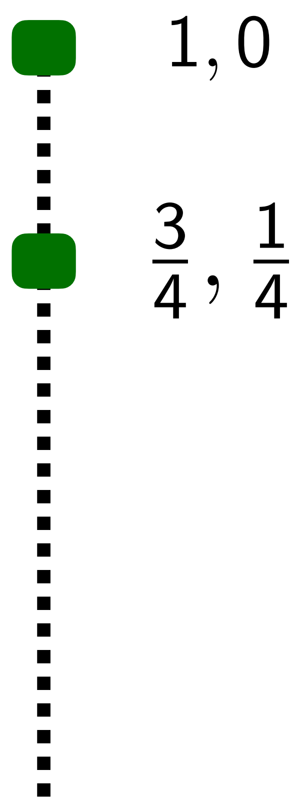
Idea:

- Interpret a belief as a point in $\mathbb{R}^{|S|-1}$
- Compute vertices of convex hull
- Eliminate all interior points

Vertex of convex hull

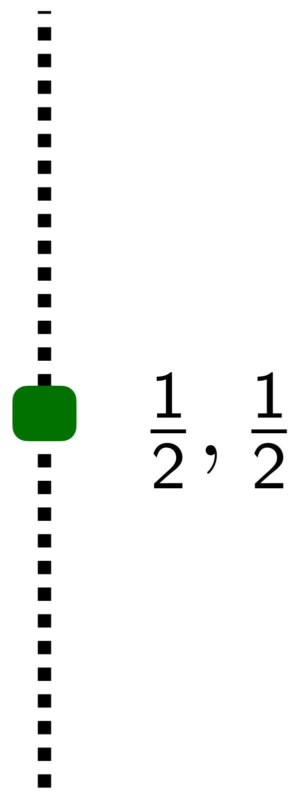
Can be eliminated! Point is a linear combination of vertices





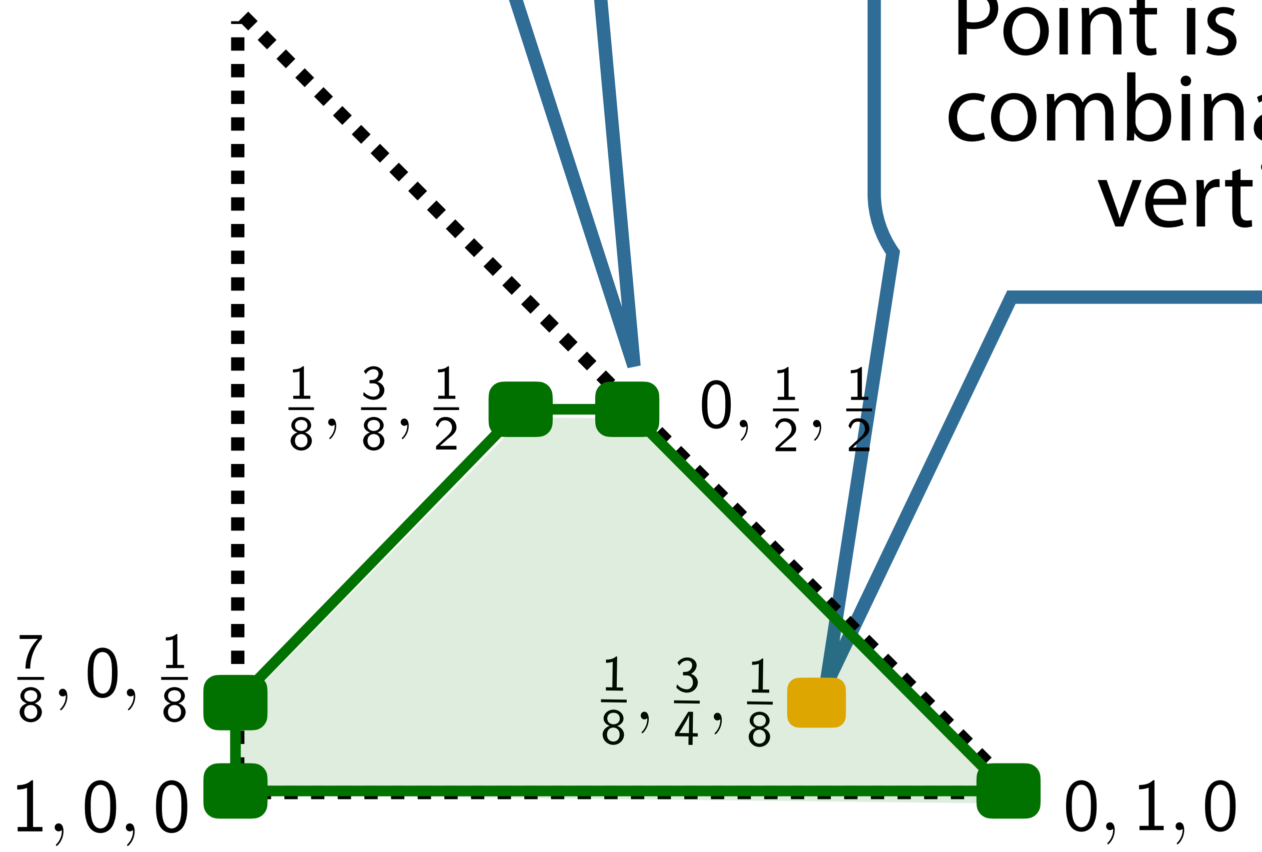
b b

s1 s2



b b o

s3 s4



b b b

s0 s1 s2

Algorithmic Approach

Filtering

Too expensive
Needs to track
exponential number of beliefs.

Filtering with
Convex Hull

Only tracks vertices
of the convex hull.
Reduces number of beliefs
dramatically

Unrolling with
Model checking

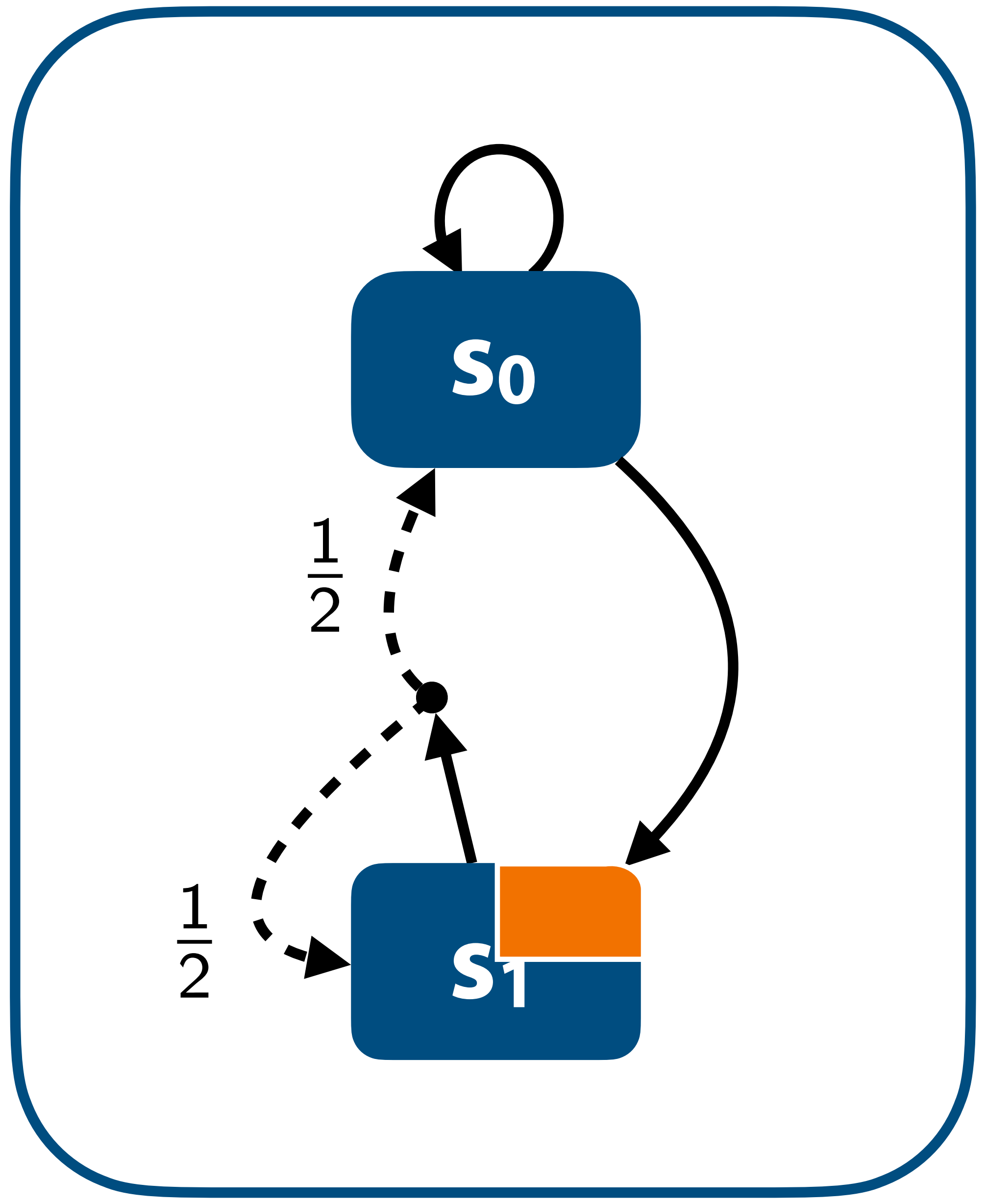
Unroll MDP by
the length of the trace then compute
conditional reachability probability.
Runs in polynomial time

Unrolling with Model Checking

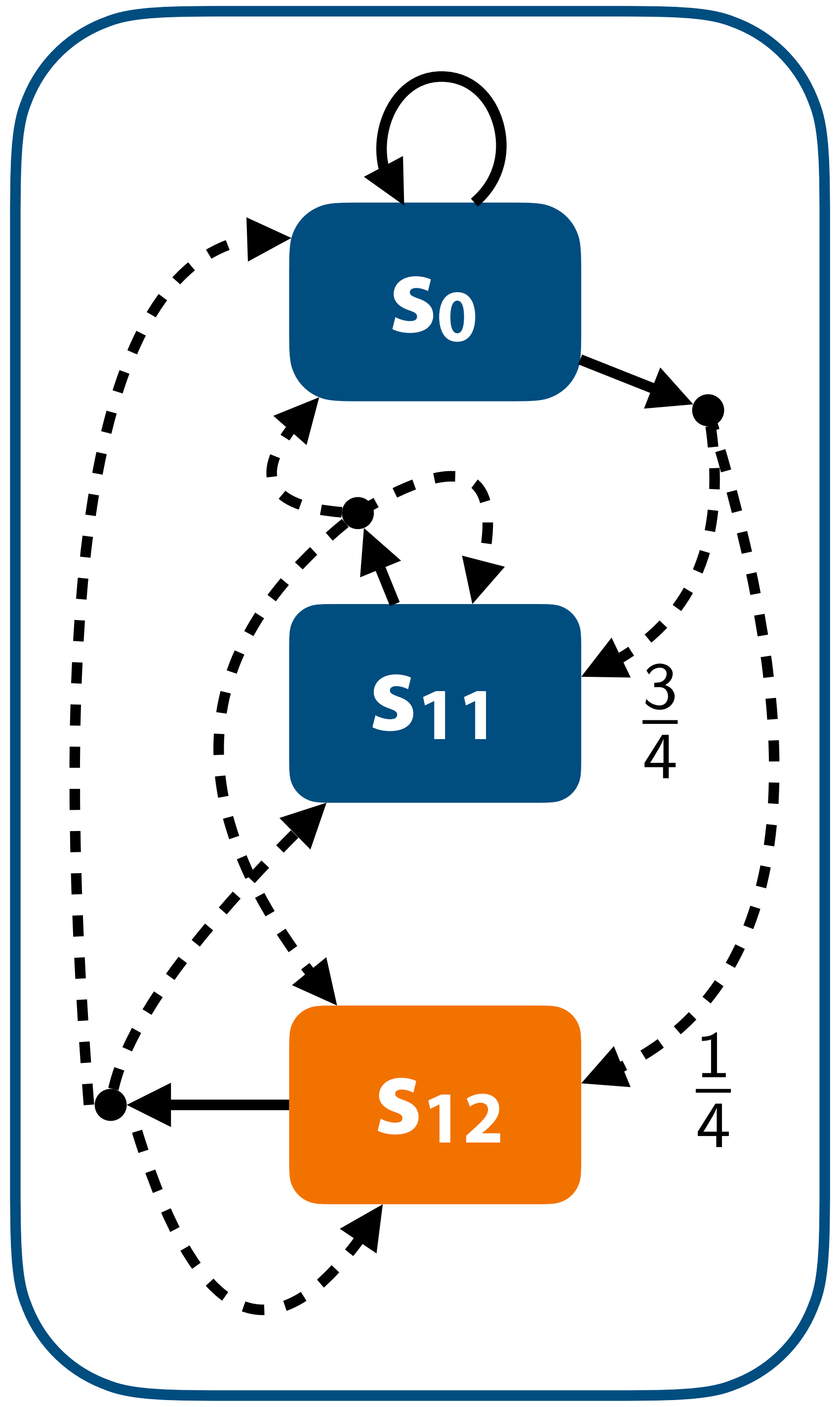
Polynomial in initial MDP

Idea:

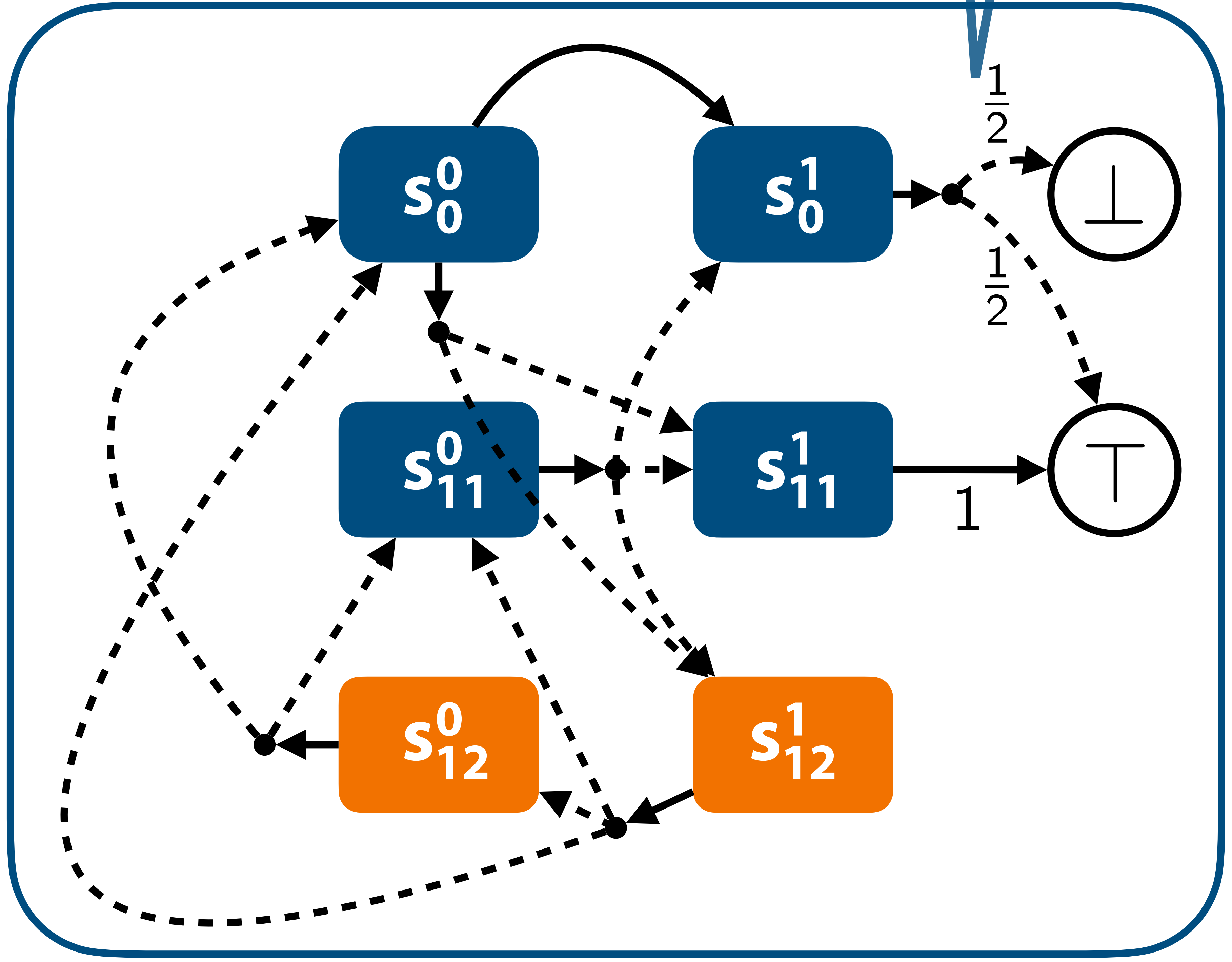
- Incorporate risk and observation probabilities as part of the transition relation
- Unroll MDP by the length of the observation trace
- Compute maximal conditional reachability probability on unrolled MDP [C. Baier et al. TACAS 2014]



$\Rightarrow$



$\Rightarrow$



$\iota = \{s_0, s_1 \mapsto \frac{1}{2}\}$

$r(s_0) = 1, r(s_1) = 2$

Encode observations probabilities as part as the transition function

$|\mathbf{b}\ \mathbf{b}| = 2 \quad \Longrightarrow \quad$ unroll once

Normalize risk: $r'(s_0) = \frac{1}{2}, r'(s_1) = \frac{2}{2}$

Experiments

Success rate: percentage of traces where no monitoring step was delayed by more than 1sec

Bench	MDP size	MDP transitions	Trace length	Filtering		Unrolling w model checking		
				Rate %	Avg # Beliefs	Rate %	Avg size	Max size
Airport A	20232	106012	100	**0**	-	100	556	629
			500	**0**	-	100	1460	1647
Airport B	20910	114143	100	**0**	-	100	524	599
			500	**0**	-	100	1075	1258
Airport C	41820	308474	100	**0**	-	100	1000	1183
			500	**0**	-	**22**	2097	2297
Refuel A	45073	2431691	100	**2**	1473	100	325	409
			500	**2**	1873	100	1071	2409
Refuel B	90154	9725277	100	**0**	-	100	608	732
			500	**0**	-	**92**	2171	4688
Evade I	377101	2022295	100	**26**	2055	**98**	332	363
			500	**4**	20524	**90**	1655	1891
Evade V A	1001	5318	100	**26**	274	100	134	241
			500	**26**	674	100	538	671
Evade V B	2161	11817	100	**0**	-	100	319	861
			500	**0**	-	**98**	777	1484

Experiments

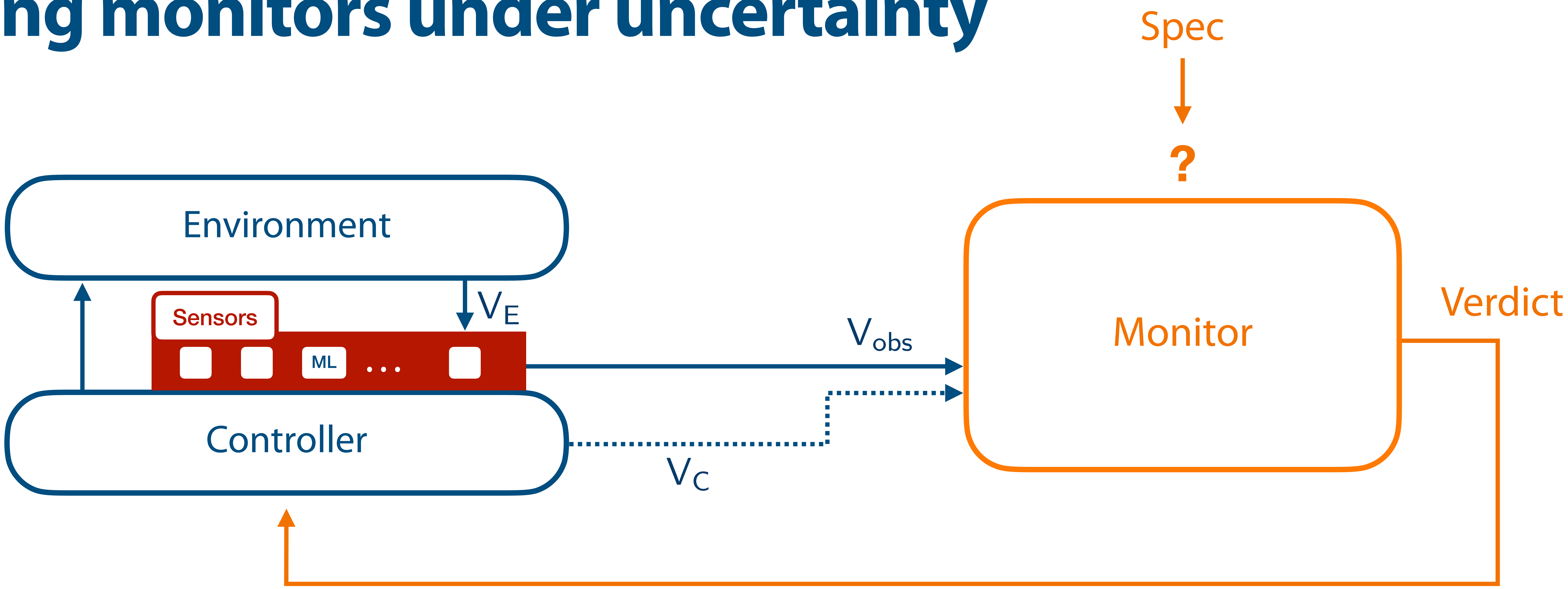
Success rate: percentage of traces where no monitoring step was delayed by more than 1sec

Bench	MDP size	MDP transitions	Trace length	Filtering		Unrolling w model checking		
				Rate %	Avg # Beliefs	Rate %	Avg size	Max size
Airport A	20232	106012	100	0	-	100	556	629
			500	0	-	100	1460	1647
Airport B	20910	114143	100	0	-	100	524	599
			500	0	-	100	1075	1258
Airport C	41820	308474	100	0	-	100	1000	1183
			500	0	-	22	2097	2297
Refuel A	45073	2431691	100	2	1473	100	325	409
			500	2	1873	100	1071	2409
Refuel B	90154	9725277	100	0	-	100	608	732
			500	0	-	92	2171	4688
Evade I	377101	2022295	100	26	2055	98	332	363
			500	4	20524	90	1655	1891
Evade V A	1001	5318	100	26	274	100	134	241
			500	26	674	100	538	671
Evade V B	2161	11817	100	0	-	100	319	861
			500	0	-	98	777	1484

Learning robust uncertainty models

Skurka, van der Maas, Junges, Torfah. AAMAS 2026

Learning monitors under uncertainty



Let: $V_{sys} = V_E \cup V_C$ and $\Sigma_{sys}, \Sigma_{obs}$ denote the set of valuations over the variables V_{sys}, V_{obs}

Problem Statement: Given a distribution $d \in \mathrm{Dist}(\Sigma^*_{sys})$, a mapping $\mu\colon \Sigma^*_{sys} \to \mathrm{Dist}(\Sigma^*_{obs})$, a risk function $r\colon \Sigma^*_{sys} \to \mathbb{R}$, and a constant λ, learn a monitor M such that:

$$\forall \sigma_{obs}.\ M(\sigma_{obs}) \Leftrightarrow \sum_{\sigma_{sys} \in \Sigma^*_{sys}} d(\sigma_{sys}) \cdot \mu(\sigma_{sys})(\sigma_{obs}) \cdot r(\sigma_{sys}) > \lambda$$

Learning monitors under uncertainty

Let: $V_{sys} = V_E \cup V_C$ and $\Sigma_{sys}, \Sigma_{obs}$ denote the set of valuations over the variables V_{sys}, V_{obs}

Problem Statement: Given a distribution $d \in Dist(\Sigma^*_{sys})$, a mapping $\mu\colon \Sigma^*_{sys} \to Dist(\Sigma^*_{obs})$, a risk function $r\colon \Sigma^*_{sys} \to \mathbb{R}$, and a constant λ, find a monitor M such that:

$$\forall \sigma_{obs}.\ M(\sigma_{obs}) \Leftrightarrow \sum_{\sigma_{sys} \in \Sigma^*_{sys}} d(\sigma_{sys}) \cdot \mu(\sigma_{sys})(\sigma_{obs}) \cdot r(\sigma_{sys}) > \lambda$$

$$M^* \in \operatorname*{arg\,min}_{M\colon \Sigma_{sys} \to [0,1]} \sum_{\sigma_{sys} \in \Sigma^*_{sys}} d(\sigma_{sys}) \cdot \mu(\sigma_{sys})(\sigma_{obs}) \cdot \ell(M(\sigma_{obs}), r(\sigma_{sys}))$$

Problem Statement: Let an ideal monitor M^*, a set of admissible monitors $\mathcal{M}$, and a distance function $\delta\colon (\Sigma^*_{obs} \to [0,1])^2 \to \mathbb{R}$, find a monitor M, s.t.,

$$M \in \operatorname*{arg\,min}_{M' \in \mathcal{M}} \delta(M^*, M')$$

Learning monitors under uncertainty

Let: $V_{sys} = V_E \cup V_C$ and $\Sigma_{sys}, \Sigma_{obs}$ denote the set of valuations over the variables V_{sys}, V_{obs}

Problem Statement: Let an ideal monitor M^*, a set of admissible monitors $\mathcal{M}$, and a distance function $\delta : (\Sigma_{obs}^* \to [0,1])^2 \to \mathbb{R}$, find a monitor M, s.t.,

$$M \in \arg\min_{M' \in \mathcal{M}} \delta(M^*, M')$$

Interpret the problem of one learning Markovian uncertainty models, specifically iHMMs

Assume system under observation is modeled by a hidden Markov model.

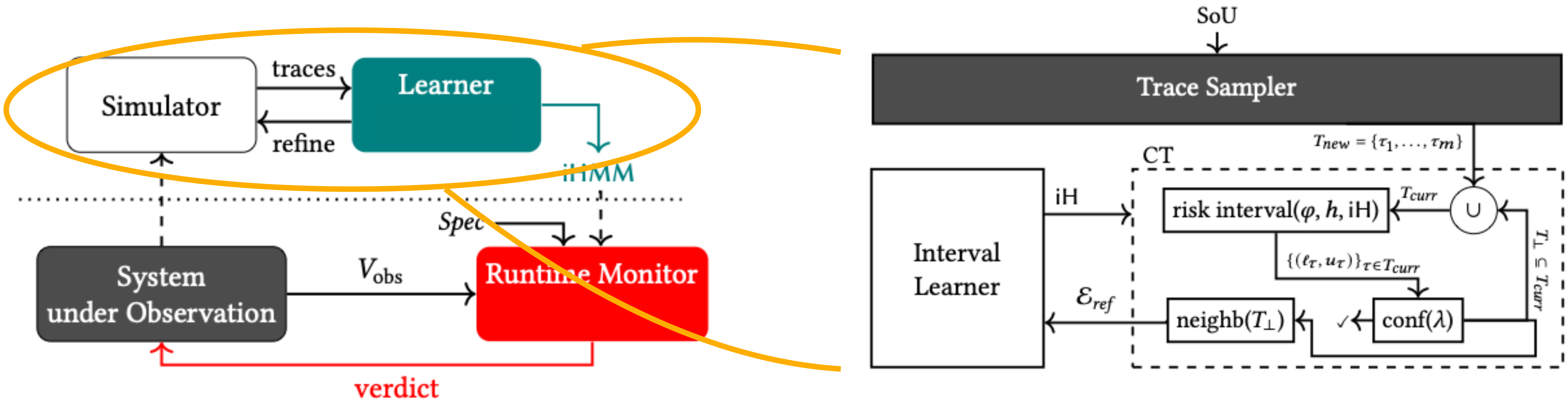
Monitors over iHMMs:

- let r_φ^h be a risk function defined over a specification φ and horizon h
- iHMM monitor: $M_{iH}(\tau) = \max_{H \in iH} \mathbb{E}_H \left[r_\varphi^h(\pi_\tau) \mid \tau \right]$

Problem Statement: Given system under observation H, and a set of iHMMs $\mathcal{H}$, and a distance function $\delta : (\Sigma_{obs}^* \to [0,1])^2 \to \mathbb{R}$, find an iHMM iH, s.t.,

$$M_{iH} \in \arg\min_{iH' \in \mathcal{H}} \delta(M_H^*, M_{iH'})$$

Learning interval hidden Markov models



Using a technique known by *linear update of probability intervals (LUI) (Suilen et al.)*, we can guarantee convergence to the ideal monitor.

Theorem. *LUI yields a sequence of iHMMs, which induces a sequence of monitors that converges to the ideal monitor.*

Experiments

Name	States	Transitions	$\|\tau\|$	h
SnL-10x10(*)	101	502	15	5
evadeV-5-3	1 001	3 878	10	20
evadeV-6-3(*)	2 161	8 667	12	20
airportA-7-10-10(*)	1 170	5 557	15	25
airportA-7-40-20	10 760	56 577	125	35
airportB-7-40-20(*)	21 520	152 170	125	35

RQ 1: Model-based vs model-free monitoring

RQ 2: iHMM vs HMM learned monitors

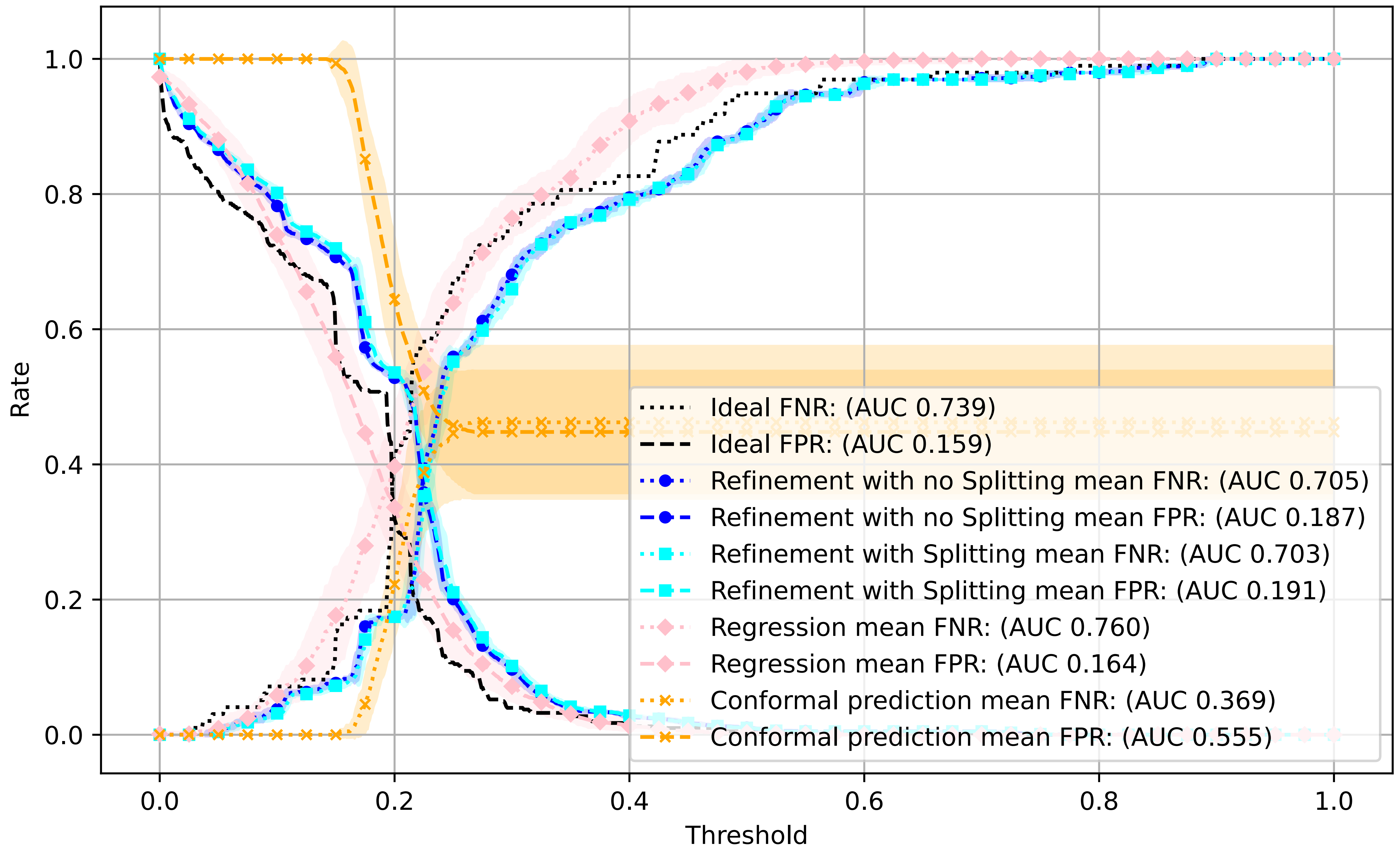
RQ 3: No refinement vs refinement

(*) – additional coarse model version

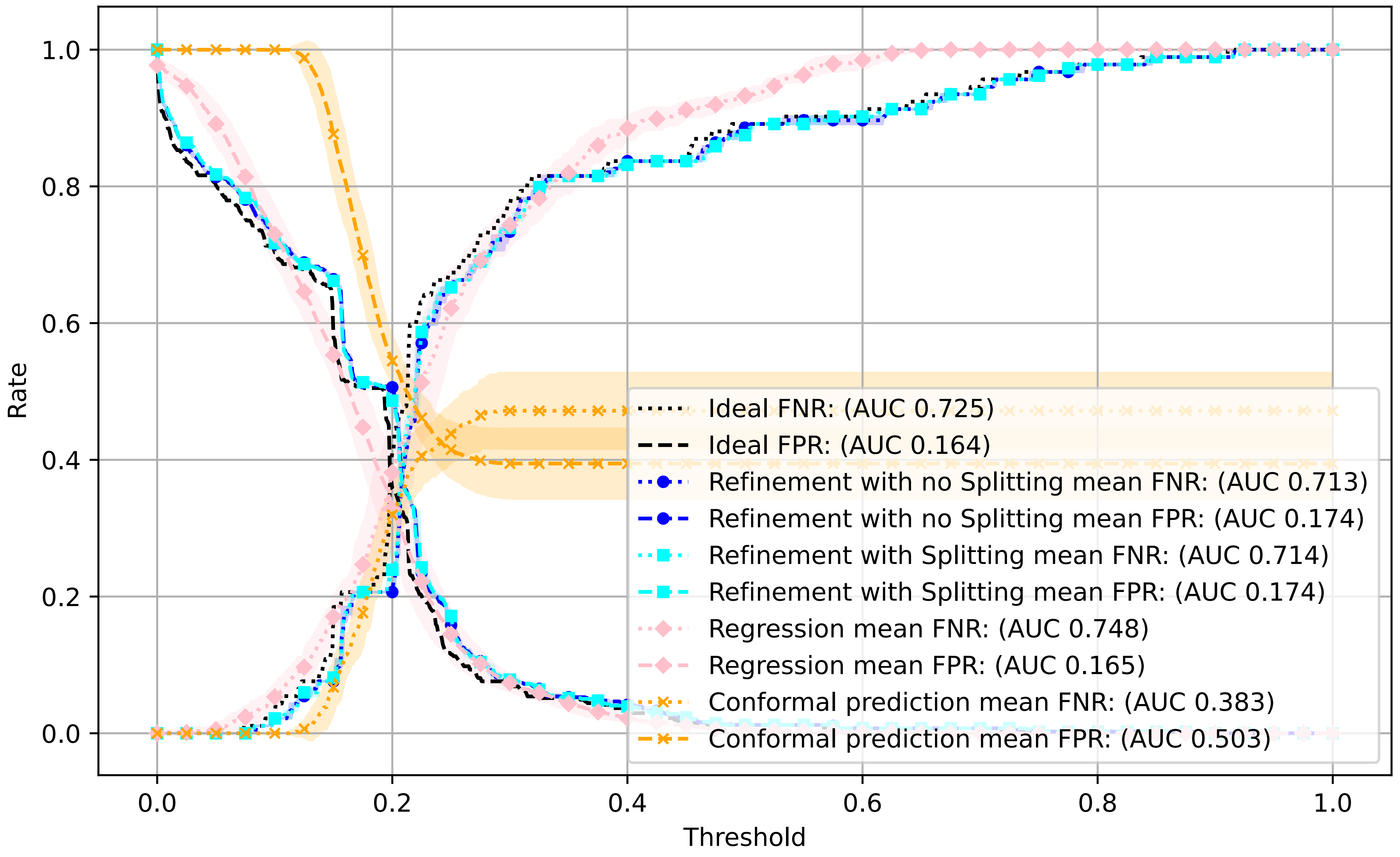
2 different stopping condition thresholds (0.001, 0.01 / 0.01, 0.1)

RQ1: Model-based vs model-free

Robust models vs regression vs conformal prediction

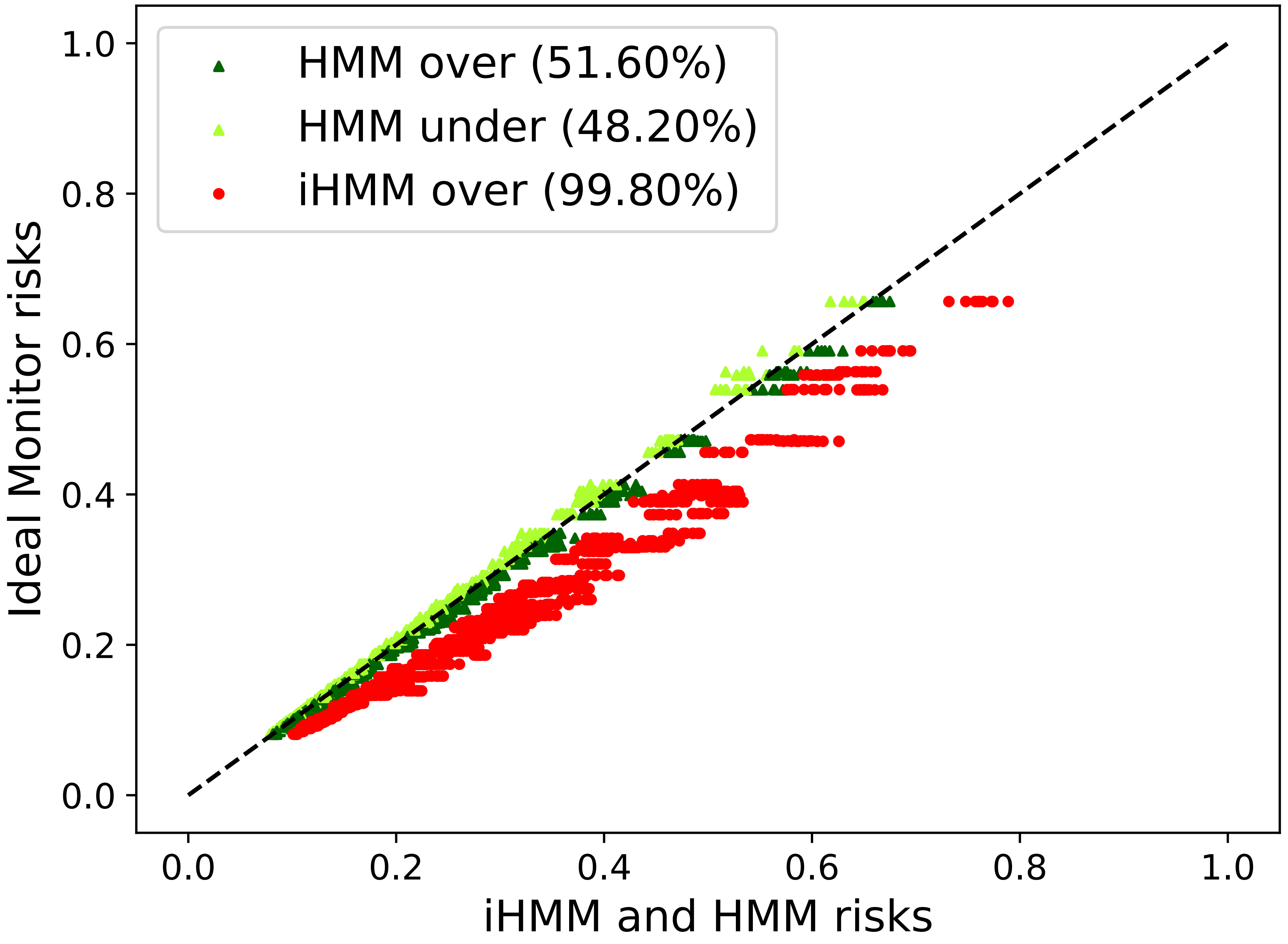


Comparing FNR and FPR between model-based and model-free methods - airportA-7-10-10, $\theta = 0.01$

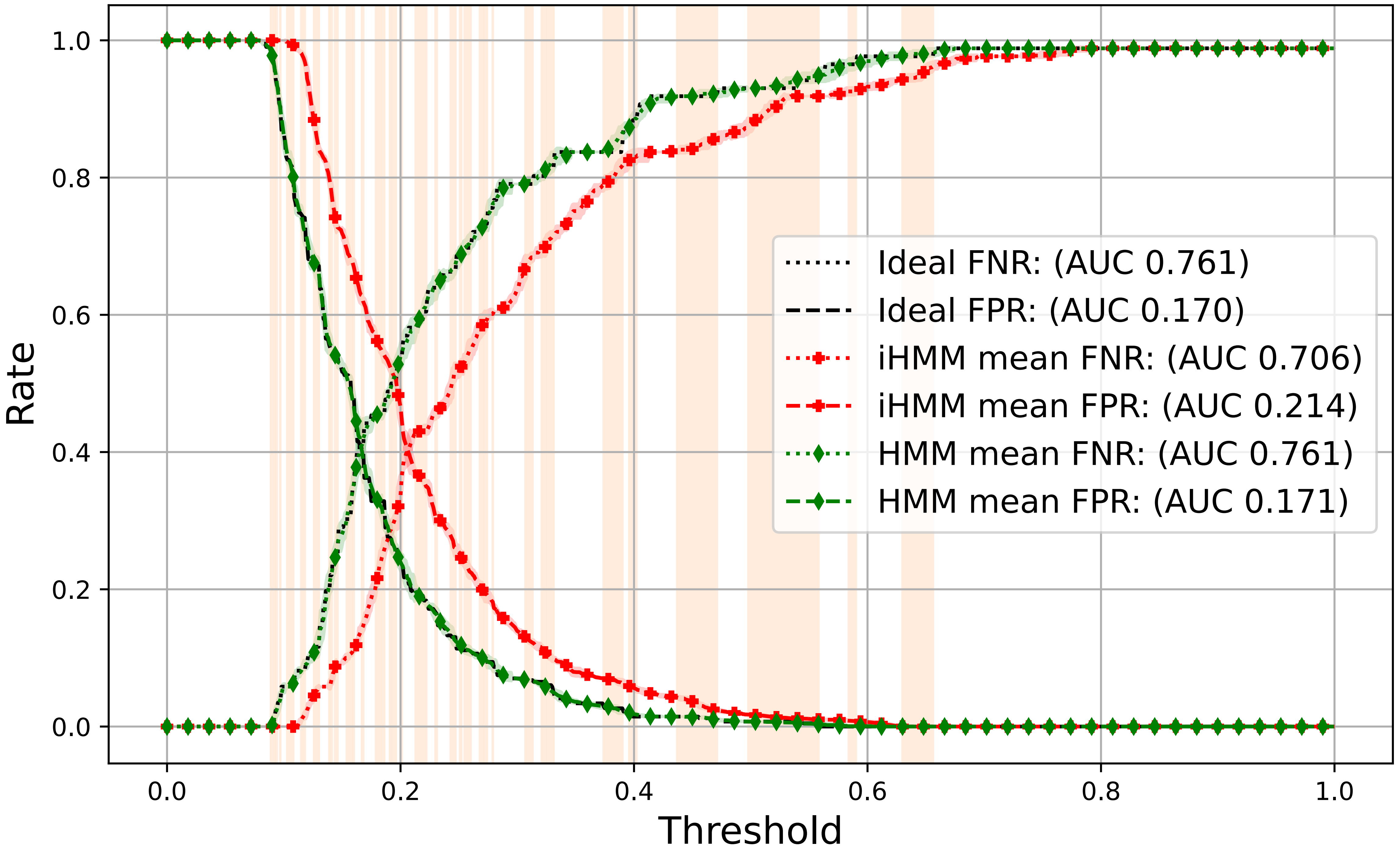


Comparing FNR and FPR between model-based and model-free methods - airportA-7-10-10, $\theta = 0.001$

RQ2: iHMM vs HMM learned monitors

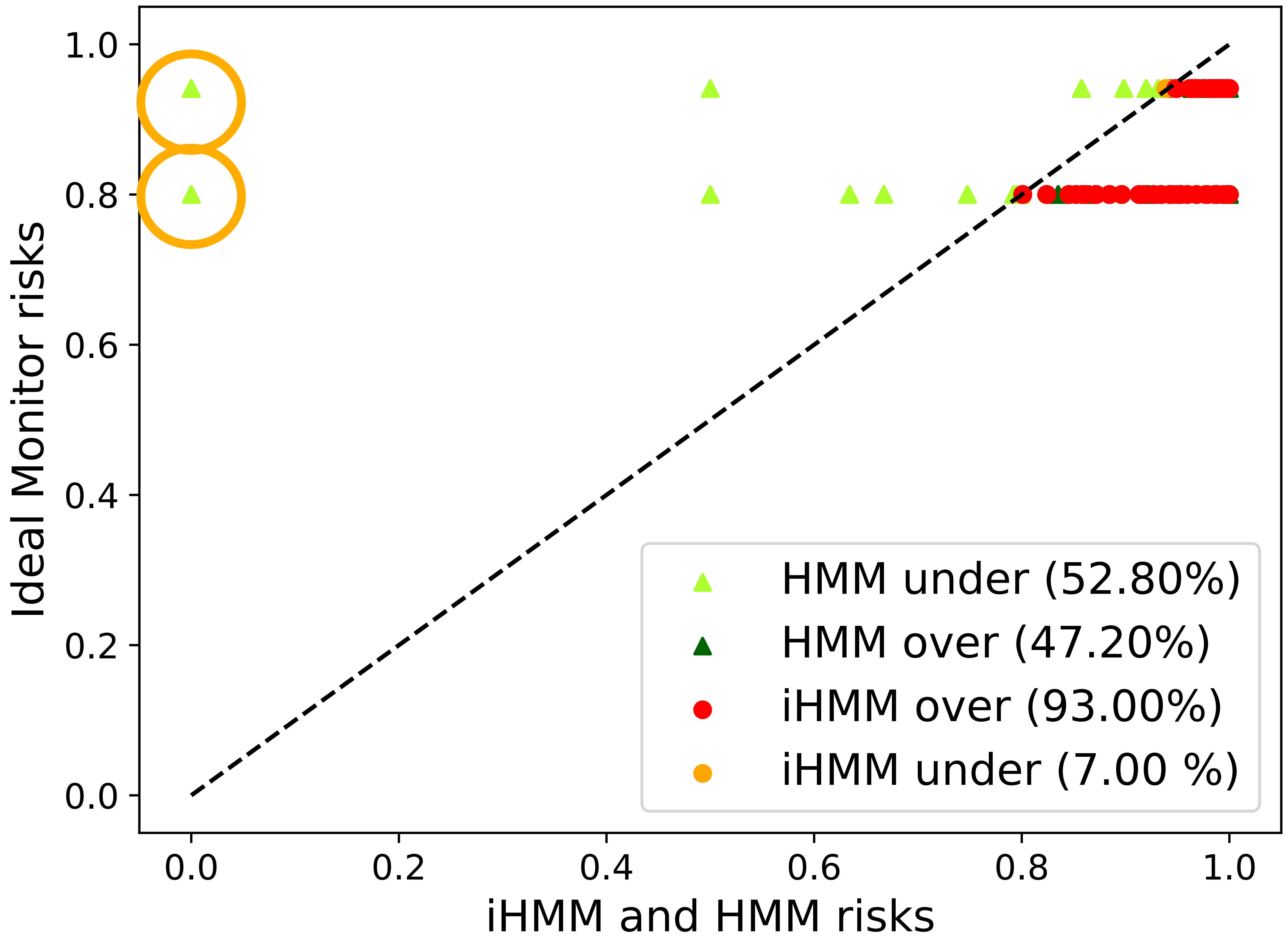


Comparing risk estimates between HMM and iHMM monitors – evadeV-5-3, θ = 0.01



Comparing FNR and FPR between HMM and iHMM monitors – evadeV-5-3, θ = 0.01

RQ2: iHMM vs HMM learned monitors



Comparing risk estimates between HMM and iHMM monitors – unlikely-15 , $\theta = 0.01$

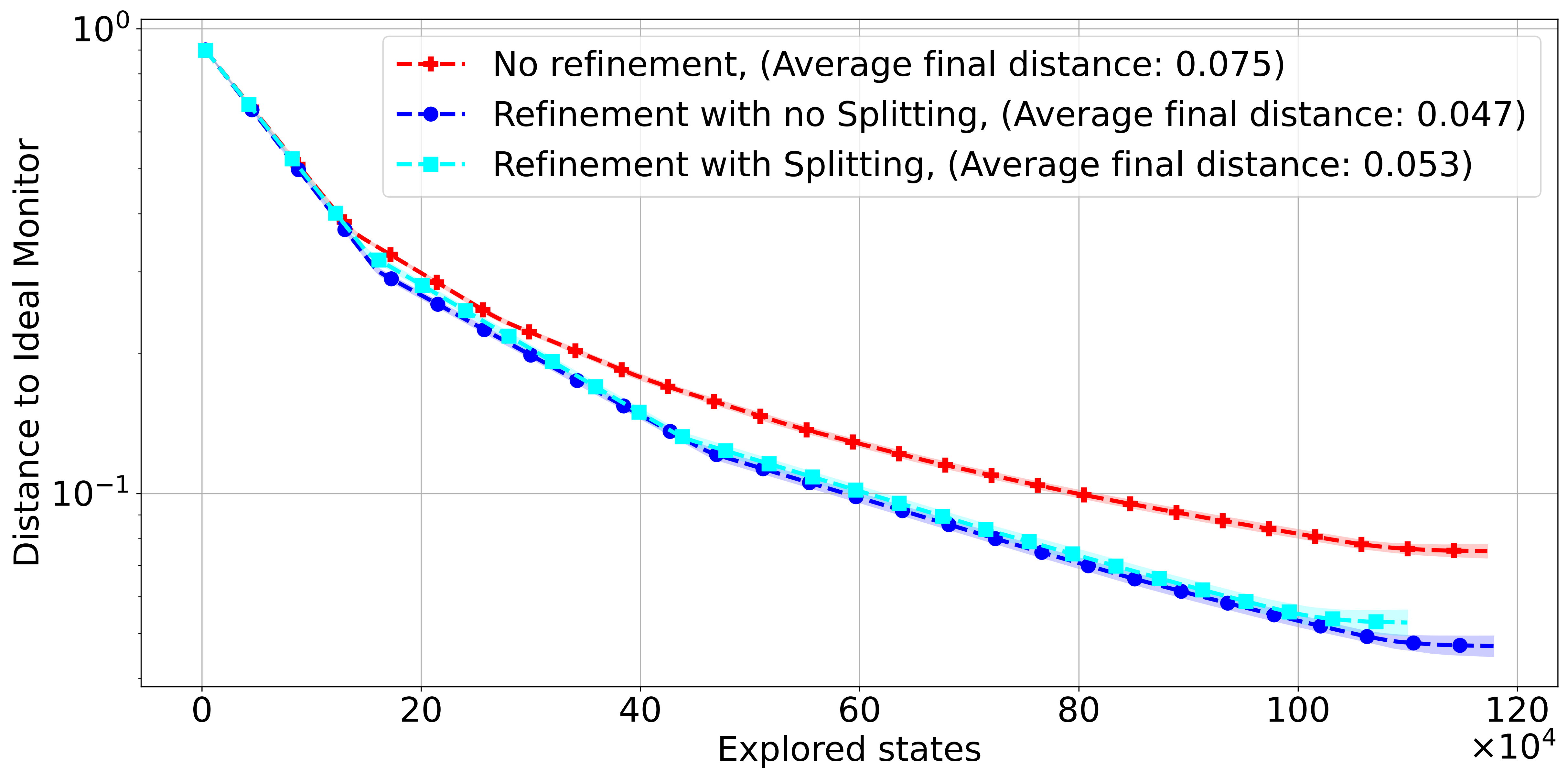
HMM

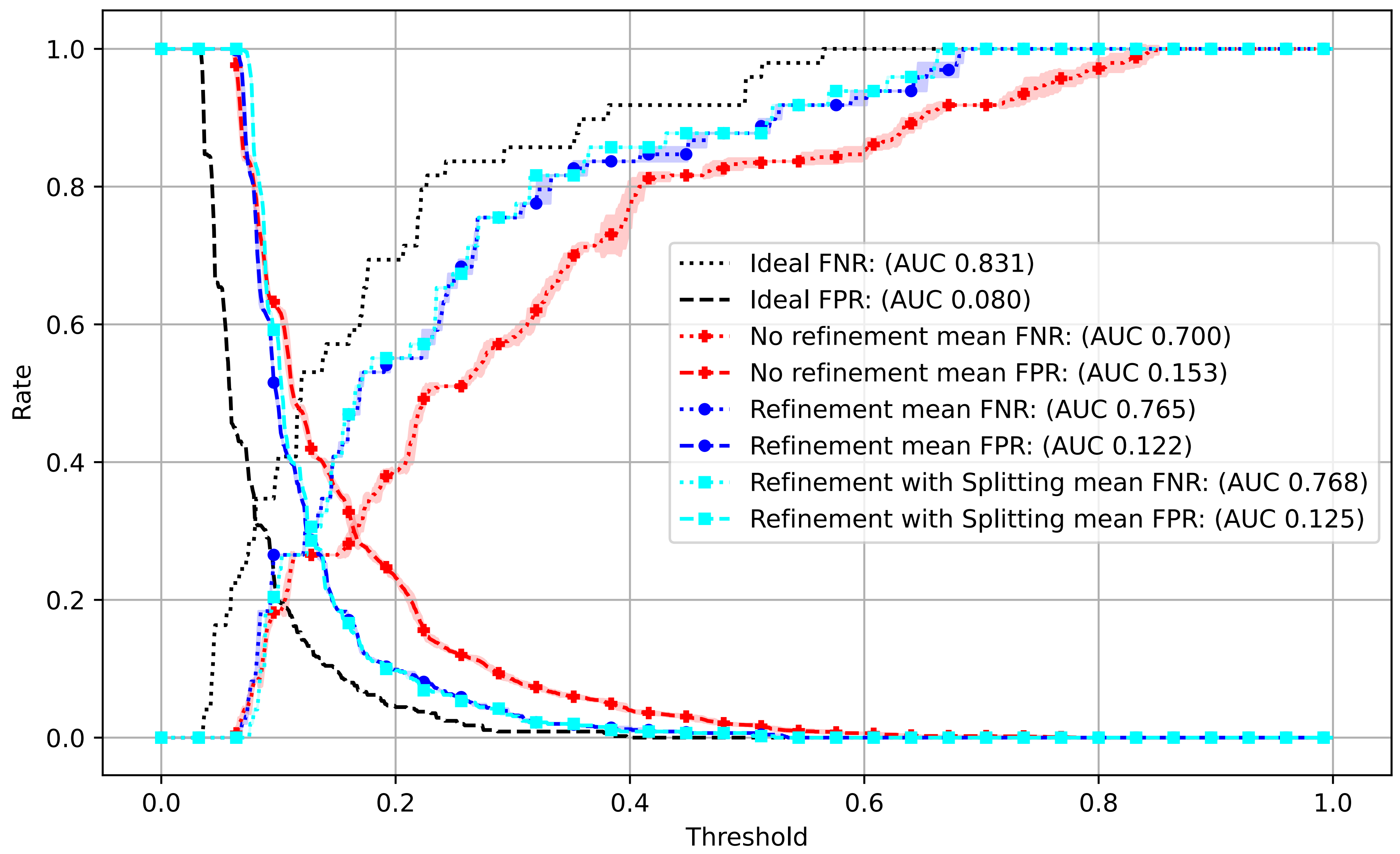
Underestimate the risk of unexplored paths

iHMM

Overestimate the risk of unexplored paths

RQ3: No refinement vs refinement

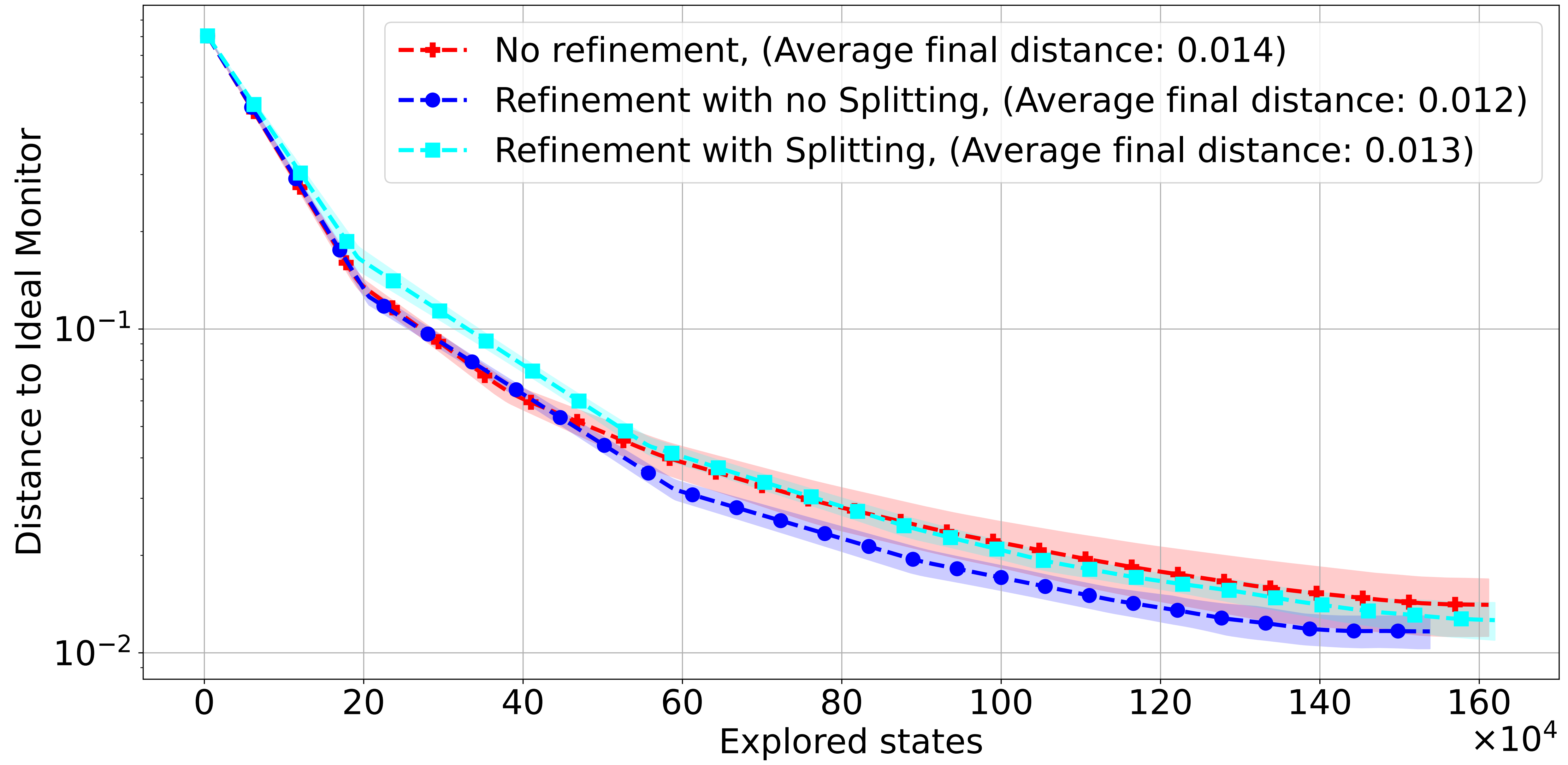




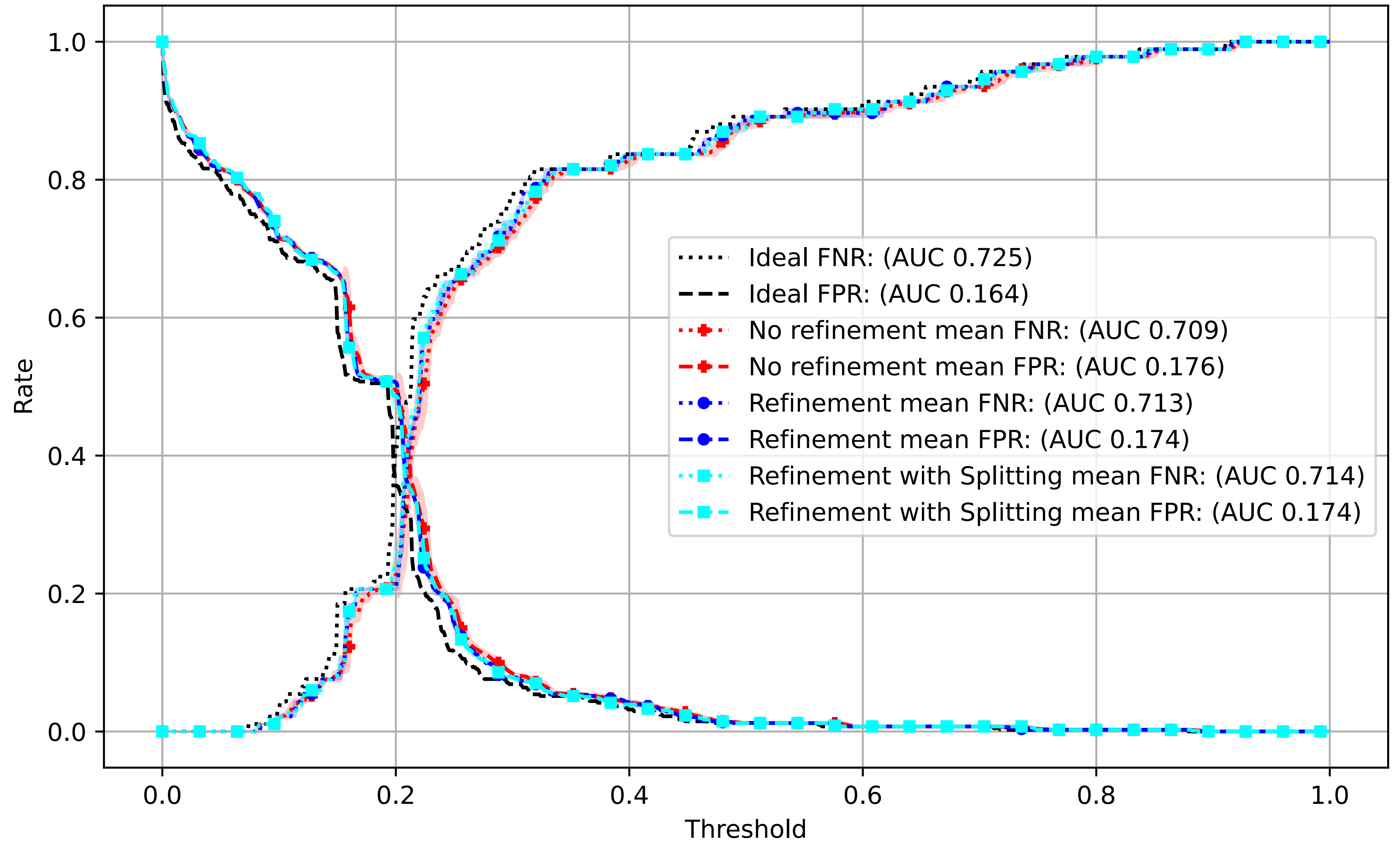
Comparing stance to Ideal Monitor between refinement and no refinement - Coarse evadeV-6-3 , $\theta = 0.01$

Comparing FNR and FPR between learning with and without refinement - Coarse evadeV-6-3 , $\theta = 0.01$

RQ3: No refinement vs refinement



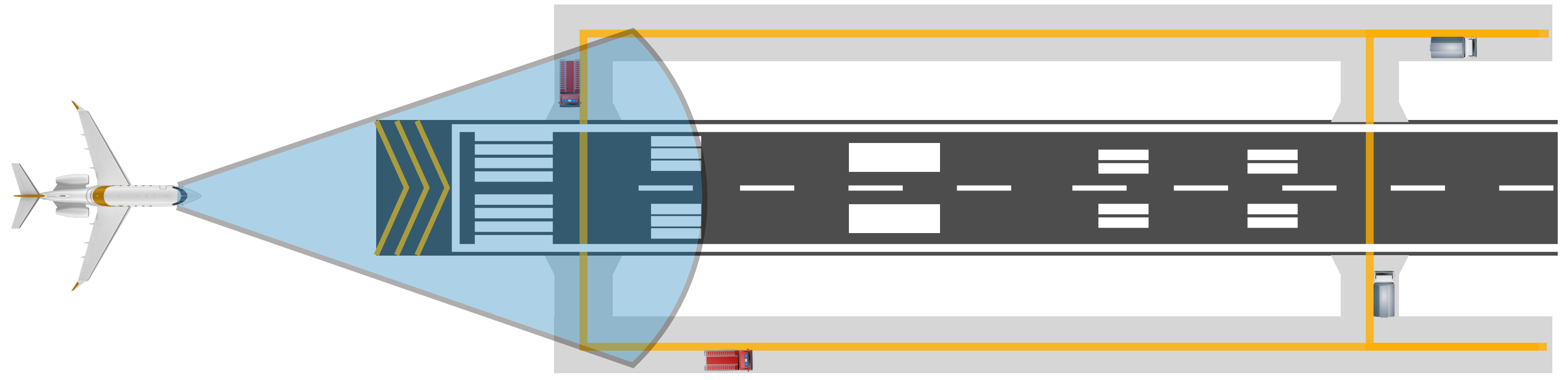
Comparing stance to Ideal Monitor between refinement and no refinement - airportA-7-10-10, θ = 0.001



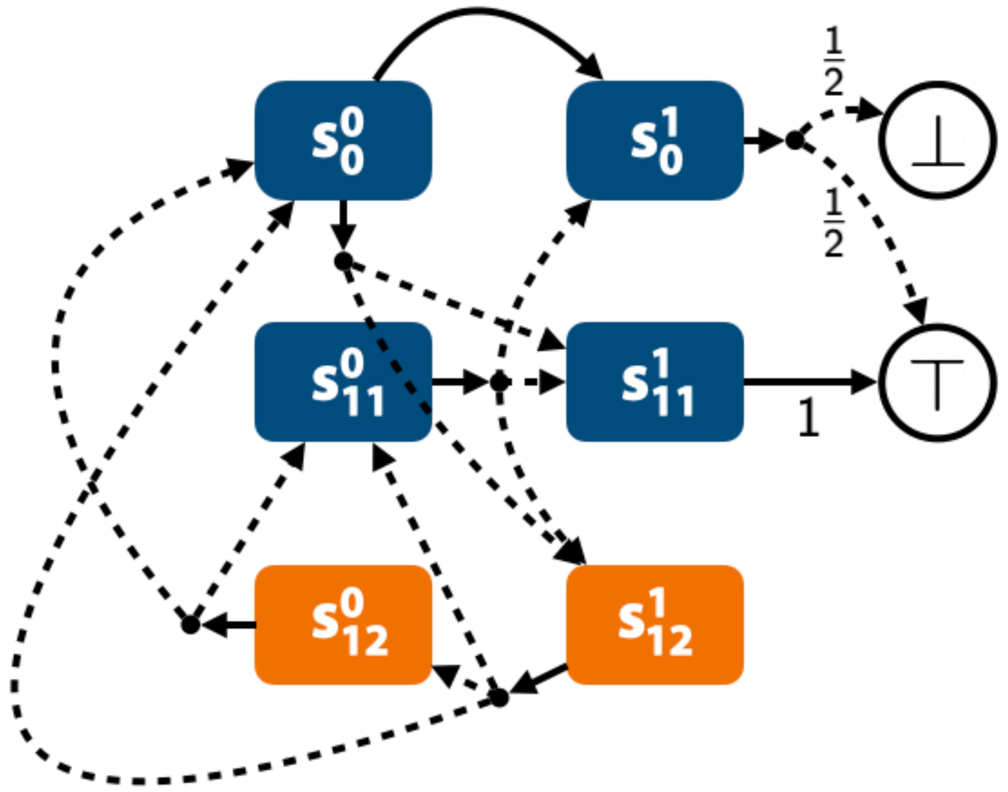
Comparing FNR and FPR between learning with and without refinement - airportA-7-10-10, θ = 0.001

Recap of today

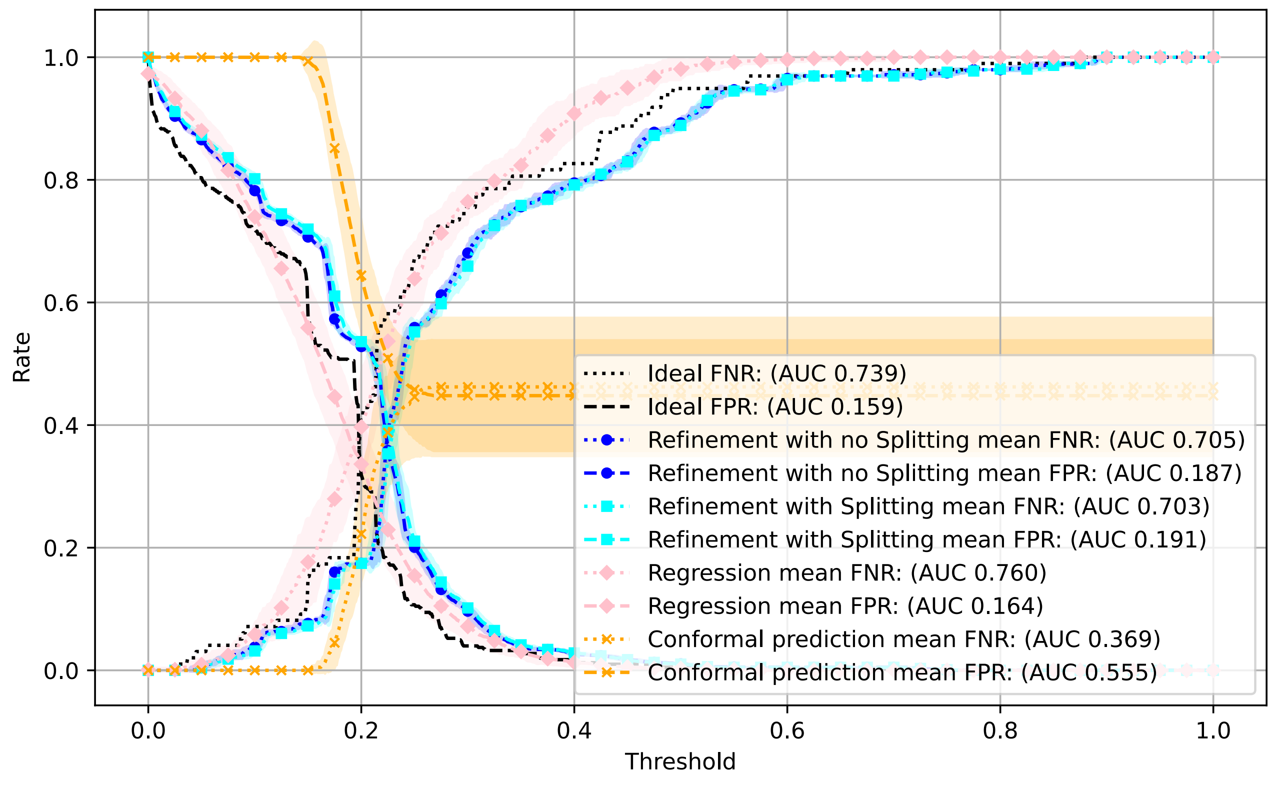
- **Runtime monitoring under uncertainty**
  - *how do we construct robust monitors over noisy data*



- **Monitoring over Markov decision processes**
  - *polynomial monitoring algorithm for MDPs*

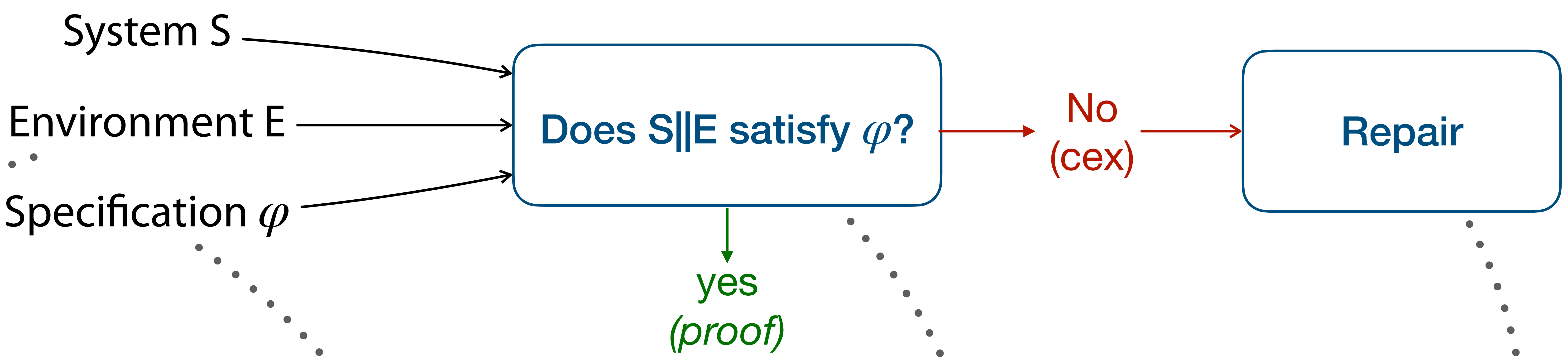


- **Learning robust uncertainty models for monitoring**
  - *model-based approaches outperform model-free approaches*
  - *robust models allow us to maintain safety without being overly conservative*



Verifying autonomous systems

Formal Verification



Challenges

Environment modeling
we do not have good models of the behaviors of agents

Specifying ML tasks
ML tasks are not easily formally specified

Scalability
ML models have high-dimensional input and state spaces

Black-box models
detecting unexpected scenarios at runtime

Principles

Probabilistic modeling
probabilistic scenario description languages

Specification formalisms for ML
- relational specifications
- from system-level to component-level specs

Compositional and statistical verification methods
combine compositional formal methods with statistical learning theory

Runtime assurance
- runtime monitoring and enforcement


Safe and Trustworthy
Autonomous Reasoning
The goal of our research group is to develop theoretical foundations and techniques for the construction of safe, reliable, and secure autonomous cyber-physical systems. Our work spans from formal specification, verification, synthesis, to runtime verification.

LATEST NEWS