Reza Teshnizi (reza.teshnizi@gmail.com)

Behavior Recognition

In Dynamic Robotic Settings

Reza Teshnizi

Committee:

- Dylan Shell (Chair)
- Swaminathan Gopalswamy
- Steve Liu
- Bobak Mortazavi

June 2024

Reza Teshnizi (reza.teshnizi@gmail.com)

INTRODUCTION

Reza Teshnizi (reza.teshnizi@gmail.com)

Behavior

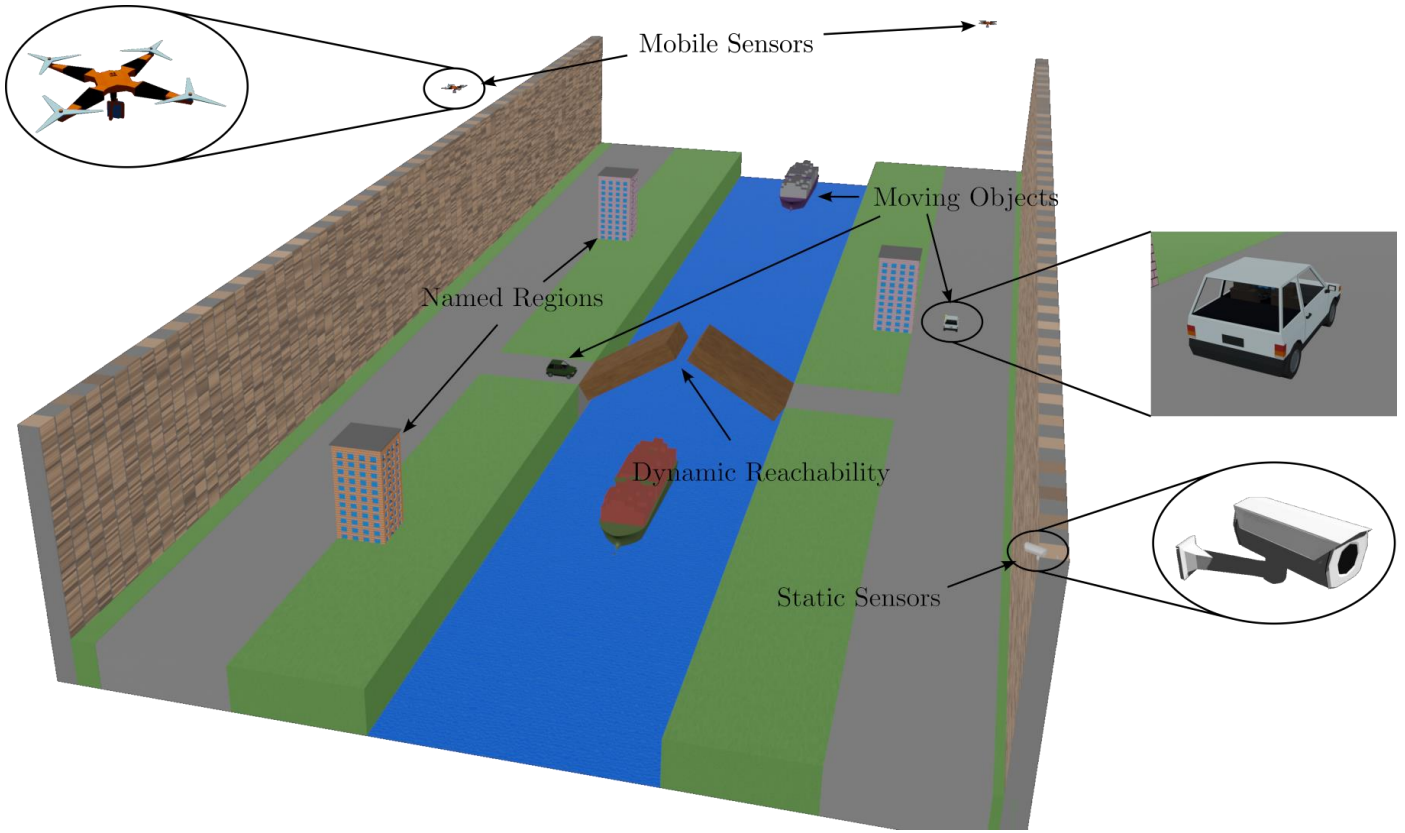
- Behavior is a concept, referring to a structured pattern of operating.
  - Target tracking
  - Situational awareness
  - Activity recognition
  - Anomaly detection
  - Reactive systems
  - ...

Reza Teshnizi (reza.teshnizi@gmail.com)

Desiderata

- Reasoning over high-level behavior of a single target
- Involving physical movements
- Large dynamic environments

- Information with rich semantics
- Detectable by realistic sensors
- Free of false positives



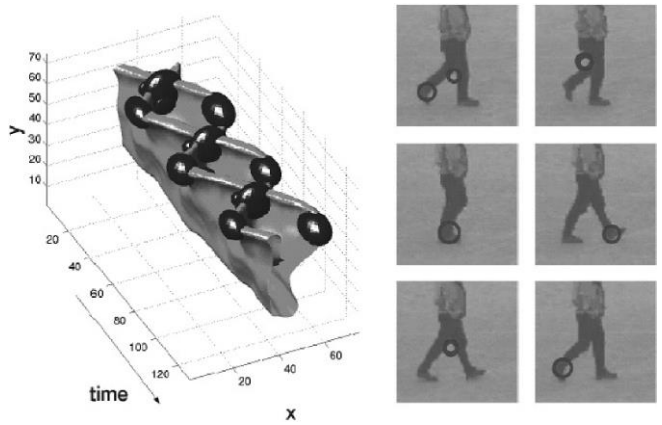
Research Contributions

1. Formalize specification of motion-based behavior
2. Provide algorithmic solution to behavior recognition
3. Incorporate behavior inference into perception processes

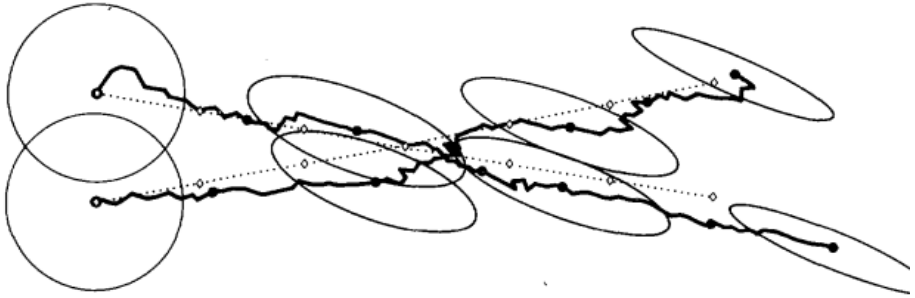
Reza Teshnizi (reza.teshnizi@gmail.com)

Probabilistic Models

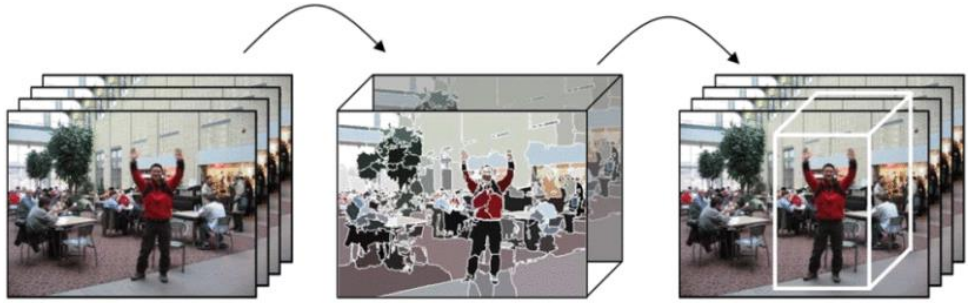
- Models evolve based on observations.



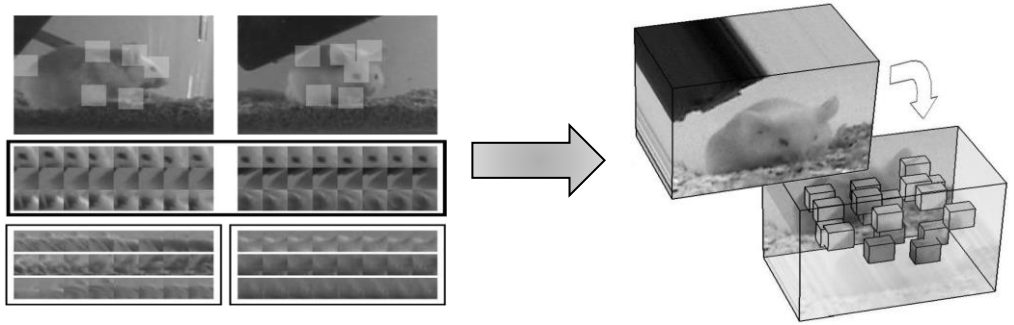
[Laptev and Lindeberg 2003]



[Fortmann et al. 1983]



[Ke et al. 2007]

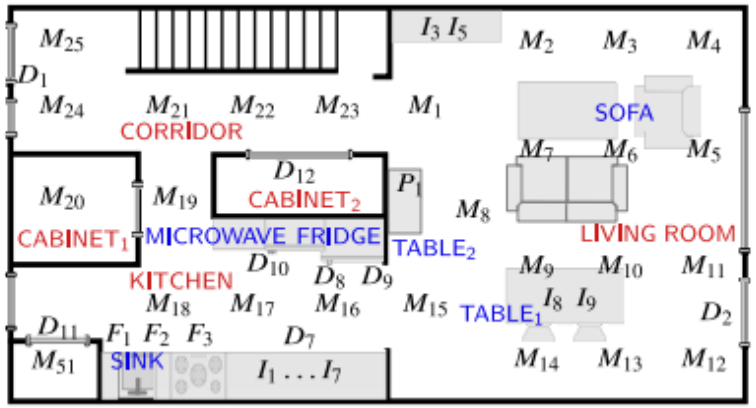


[P. Dollar et al. 2005]

Reza Teshnizi (reza.teshnizi@gmail.com)

Knowledge-based Models

- Choices of sensors and their placement matter.



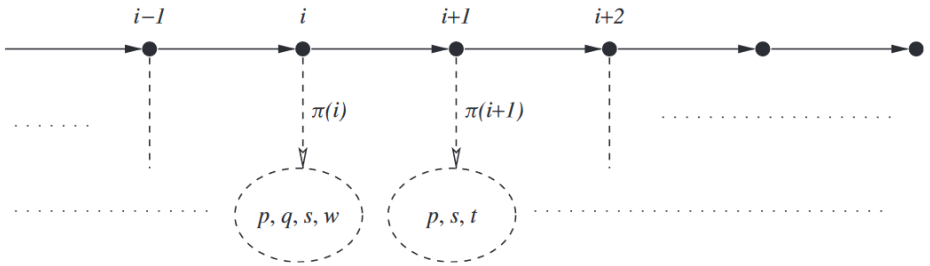
[L. Buoncompagni et al. 2022]

[L. Cheng and C. Nugent 2019]

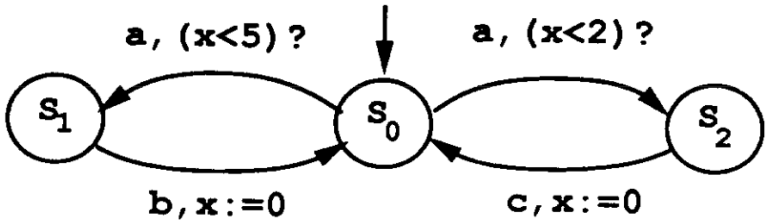
Reza Teshnizi (reza.teshnizi@gmail.com)

Temporal Reasoning

- Temporal Logic
- Reasoning over events that happen across time

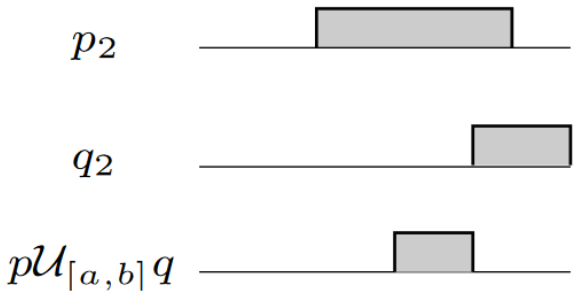


[M. Fisher 2011]



[R. Alur and D. Dill 1994]

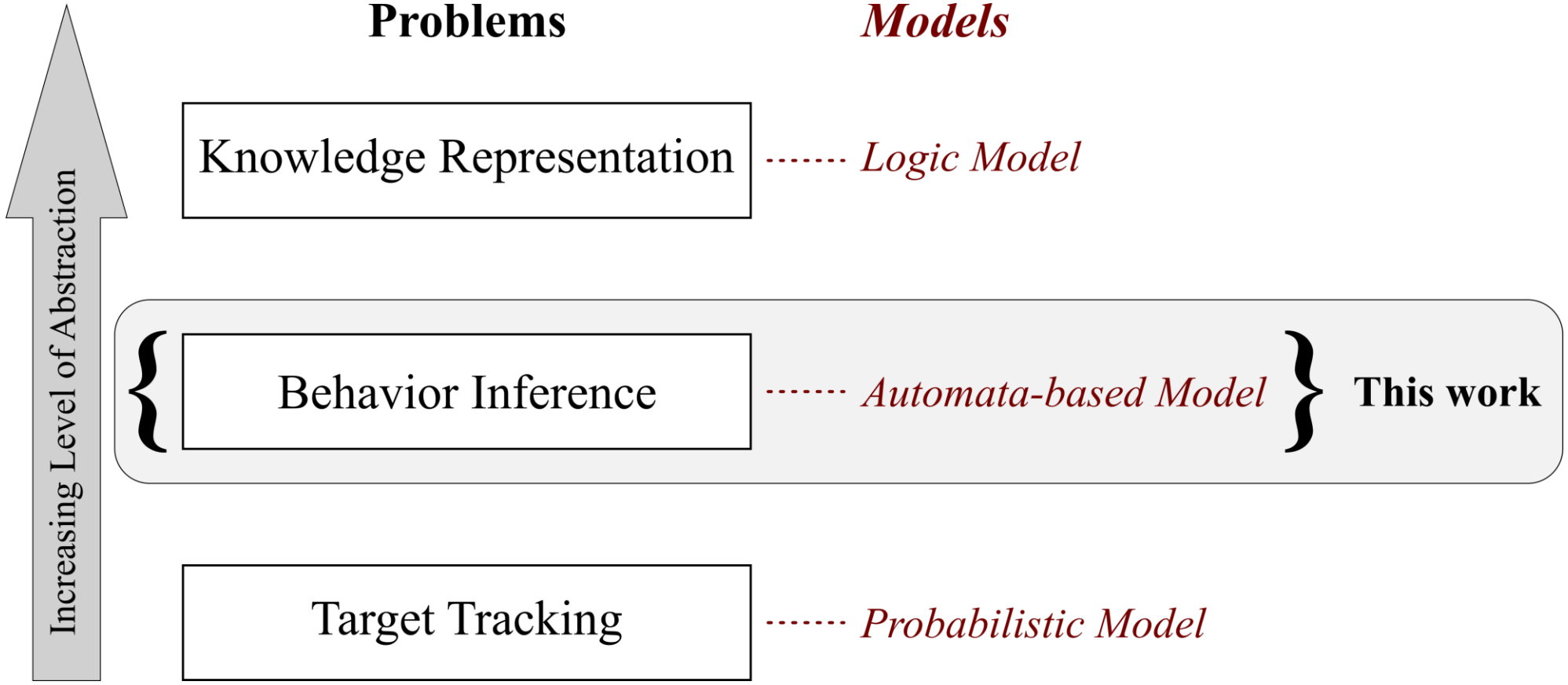
- Signal Temporal Logic (STL)
  - Predicate functions are static over time
  - Signals may be discontinuous



[O. Maler and D. Nickovic 2004]

Reza Teshnizi (reza.teshnizi@gmail.com)

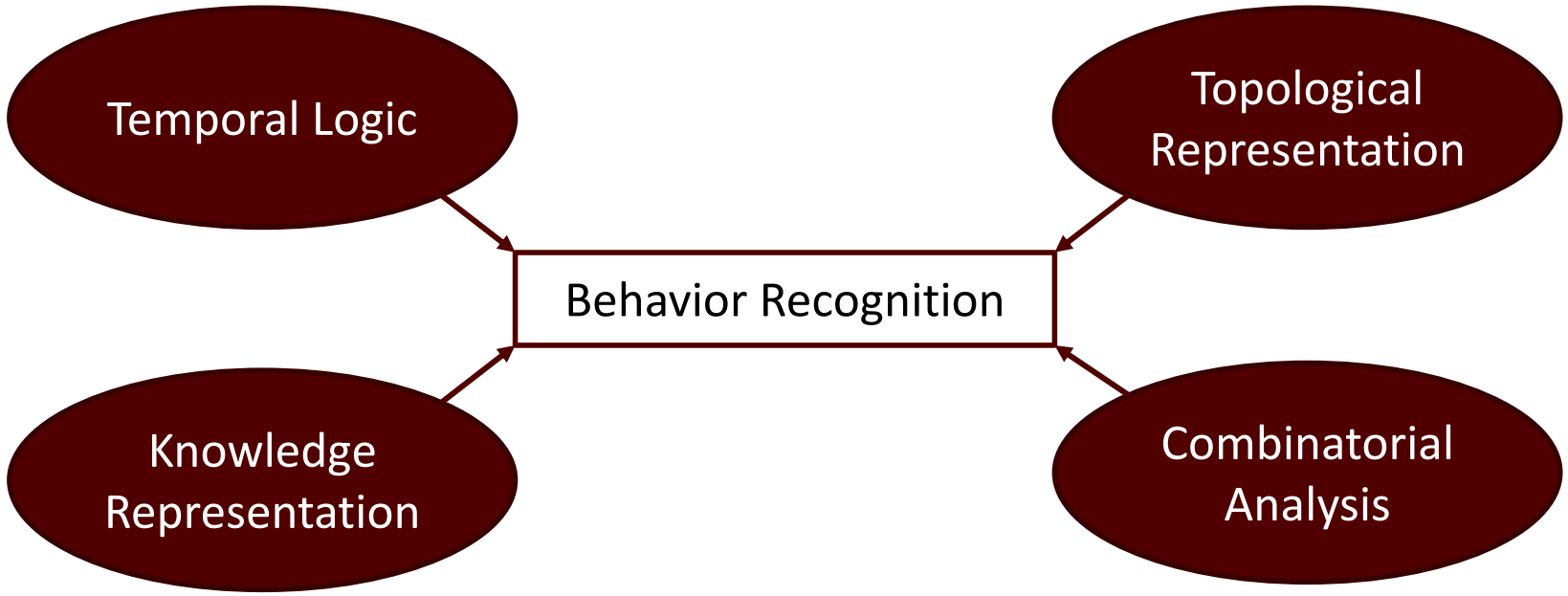
Conceptual Placement



Reza Teshnizi (reza.teshnizi@gmail.com)

Behavior Recognition

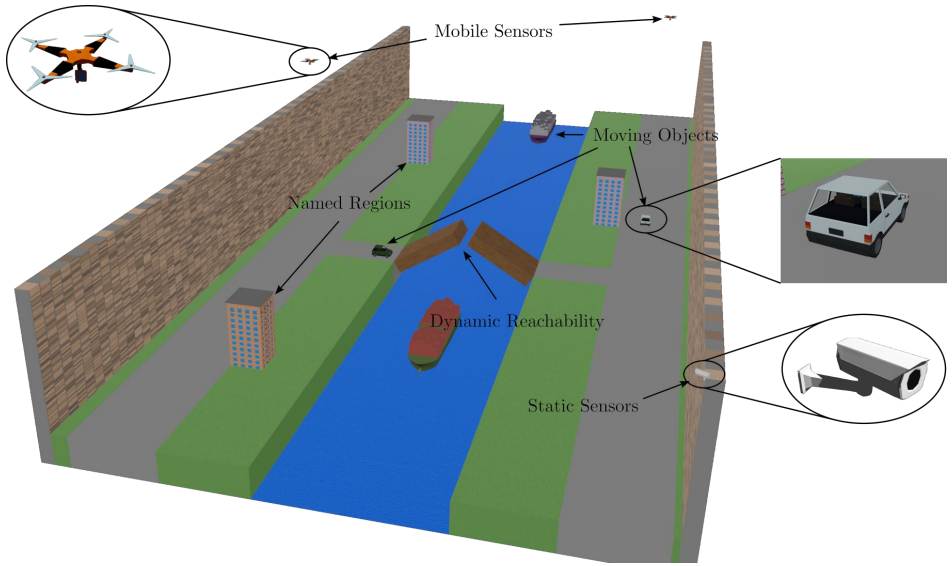
- How a robotic system can recognize patterns of movement
  - Formal specifications
  - Large dynamic environment
  - Network of mobile sensors
  - Free-moving target

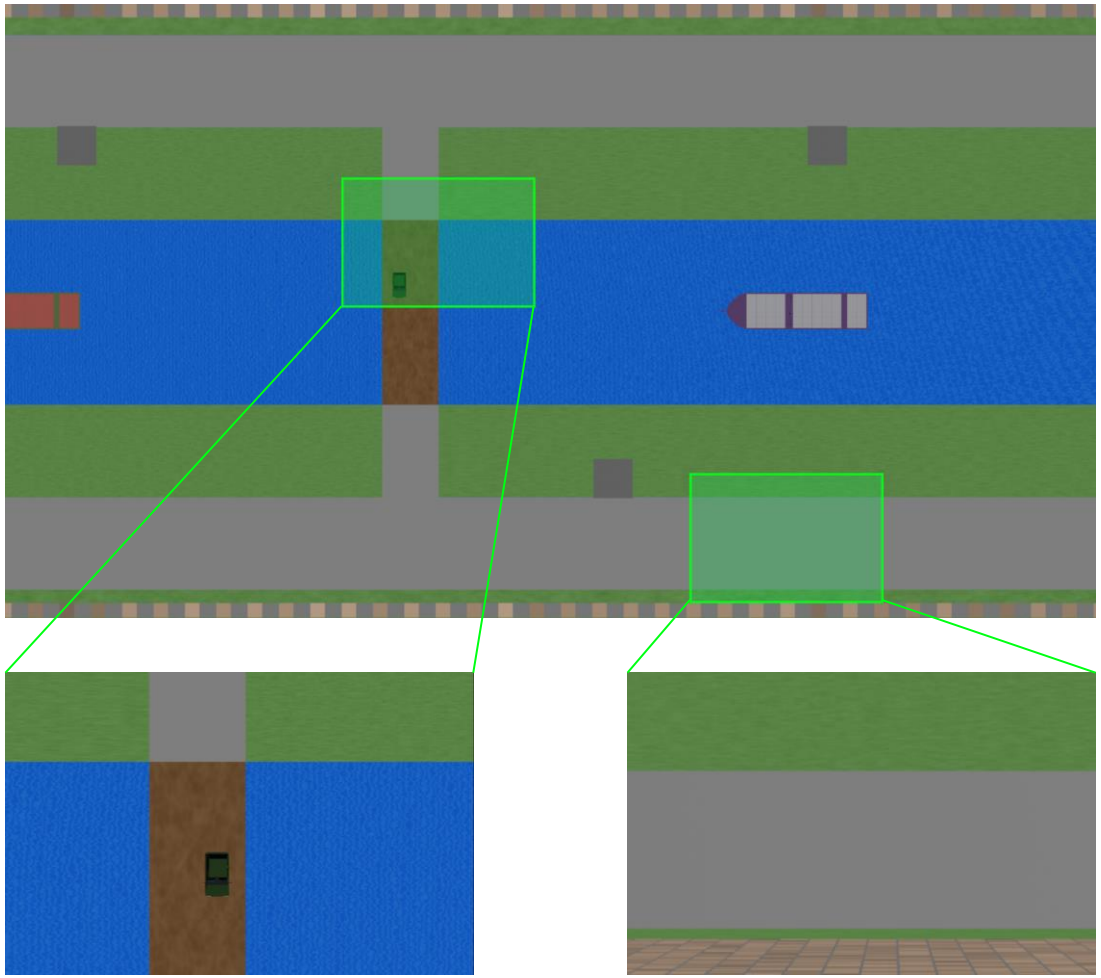


Reza Teshnizi (reza.teshnizi@gmail.com)

Monitoring Movements

- Limited field-of-view
- Long periods of time
- Mobile Sensors
- Spatially and temporally sparse observations

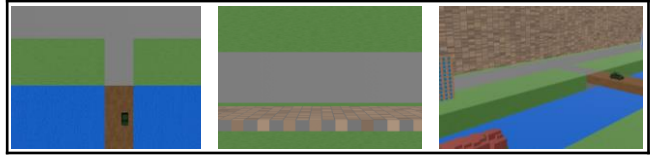


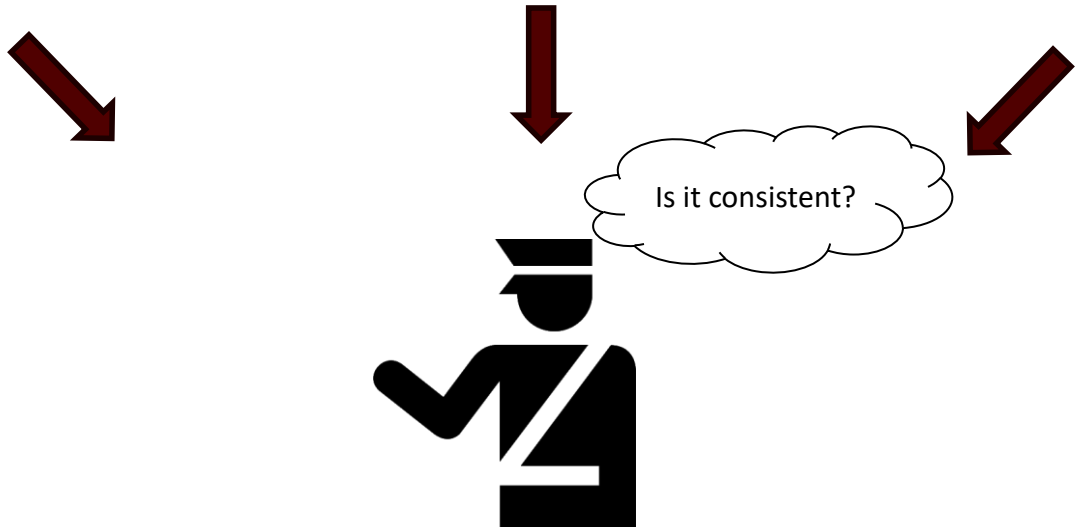


Reza Teshnizi (reza.teshnizi@gmail.com)

Input Information

- What defines a behavior recognition problem?

Specification of Environment	Specification of Behavior	Observations
In this environment there are buildings, streets, bridges, and water. The bridge is lifted from 10am to 11am…	Delivery driver arrives at Building Y at 9am. Delivery driver moves over streets and bridges…	

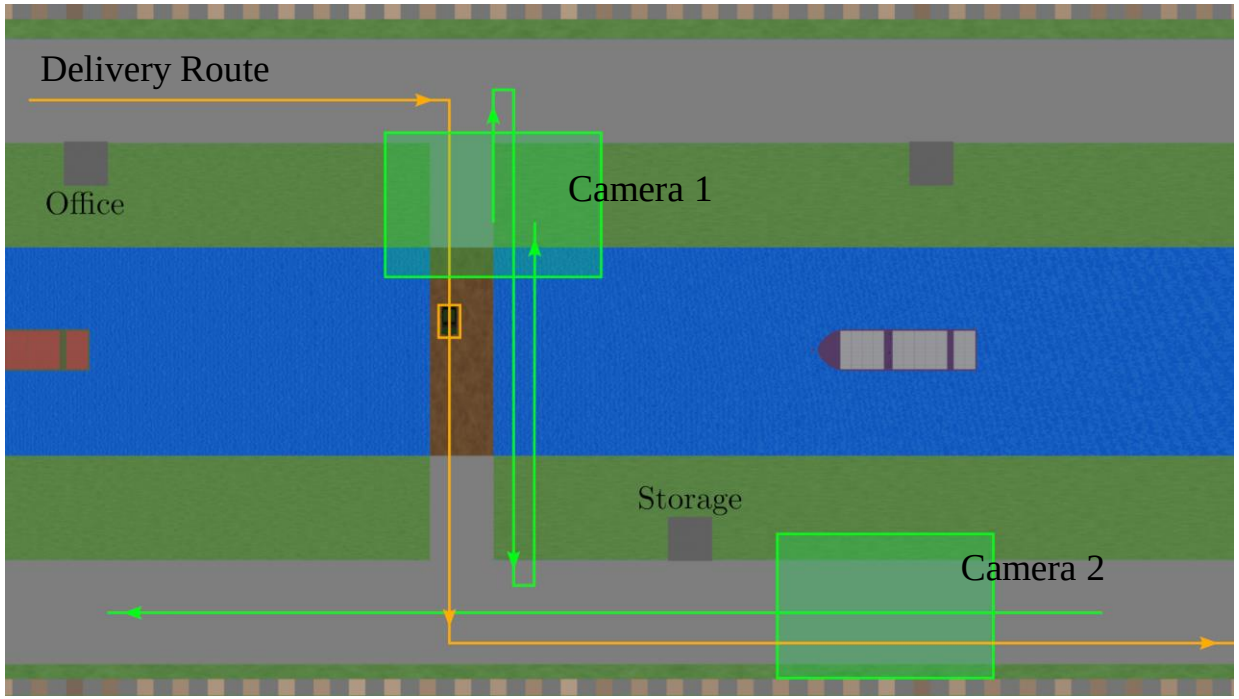


Reza Teshnizi (reza.teshnizi@gmail.com)

Visual Example: Tower Bridge

- Office contains valuable assets
- The reserves are kept at Storage
- The bridge is lifted at predefined times
- Environment under aerial monitoring

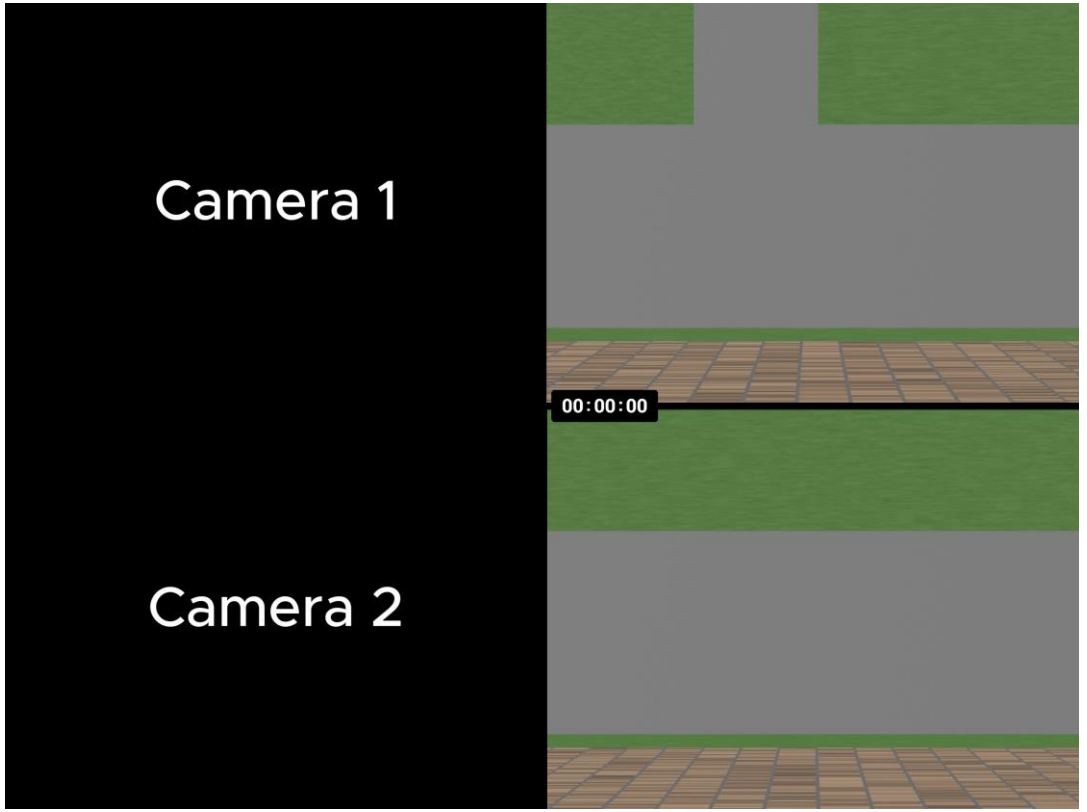
➢A delivery driver must transfer the reserves to storage every morning



Reza Teshnizi (reza.teshnizi@gmail.com)

Visual Example: Observations

- Did the target carry out the delivery task?



Reza Teshnizi (reza.teshnizi@gmail.com)

Research Contributions

1. Formalism of Motion-based Behavior Recognition
   - Automata-based specifications
   - Formulation of recognition problems

2. Algorithmic Filtering via Topological Representations
   - Minimal graph structures for the solution space
   - Logical abstraction of metric information

3. Perception Process Enabled with Behavior Inference
   - Modular architecture to integrate behavior models
   - Interfacing with a network of sensors and an ontology

Reza Teshnizi (reza.teshnizi@gmail.com)

Presentation Outline

Definitions and Formulation
- Formal Specifications
- Matching and Filtering

Algorithm and Abstraction
- Topological Representation
- Logical Descriptions

Implementation
- Modular Architecture
- Demonstration

Reza Teshnizi (reza.teshnizi@gmail.com)

I. Definitions and Formulation
II. Algorithm and Abstraction
III. Implementation

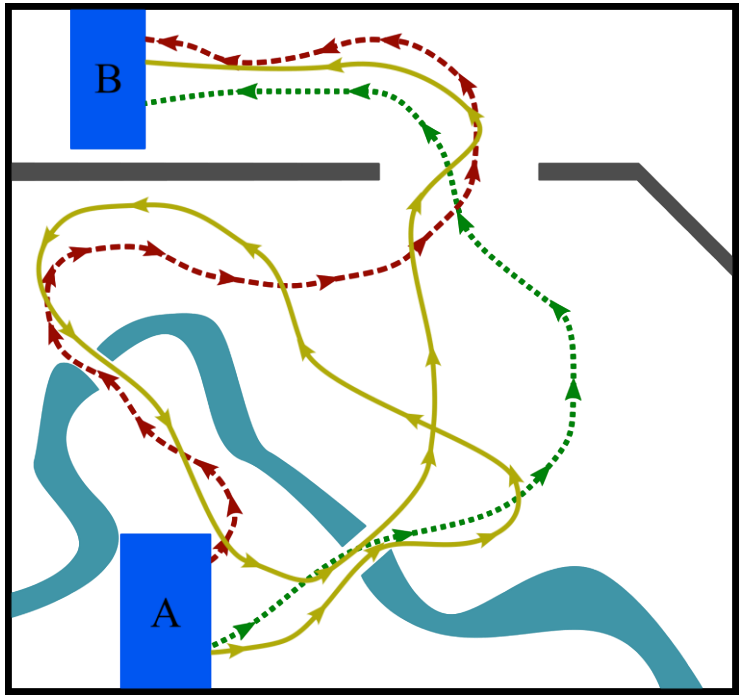
---

1. Formal Specifications
2. Matching and Filtering

Reza Teshnizi (reza.teshnizi@gmail.com)

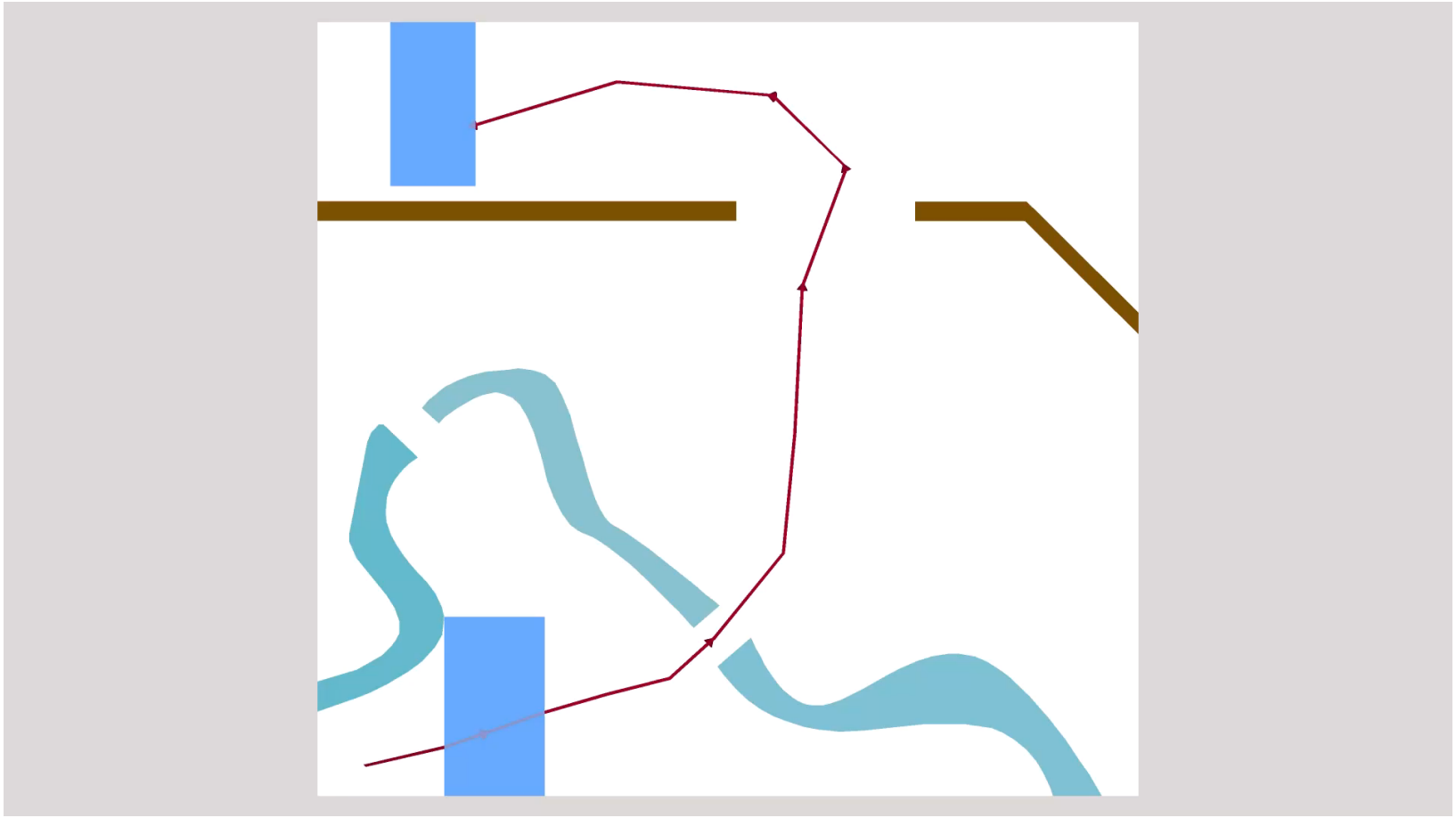
Motion-based Behavior

- Patterns of visitation
  - Facility A
  - Facility B



Reza Teshnizi (reza.teshnizi@gmail.com)

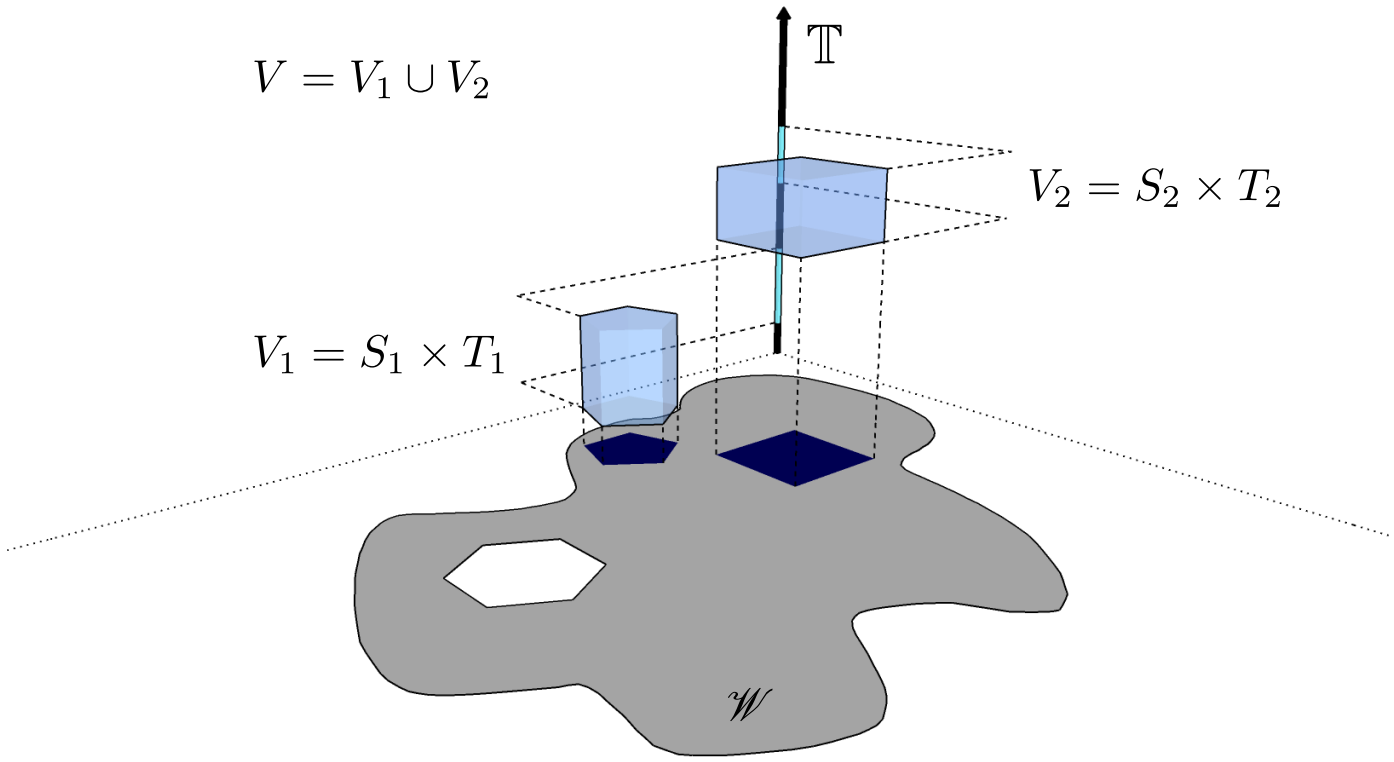
Trajectory



Reza Teshnizi (reza.teshnizi@gmail.com)

Behavior Automaton

- Space-time set
  - Spatial Sets $\subseteq \mathcal{W} \subseteq \mathbb{R} \times \mathbb{R}$
  - Temporal Sets $\subseteq \mathbb{T} = \mathbb{R}^{\geq 0}$
- Automaton $(\mathcal{W}, Q, \Sigma, \Delta, q_0, F)$
  - $\mathcal{W}$: free space
  - Q: States
  - Σ: Alphabet consisting of space-time sets
  - $\Delta \subseteq Q \times \Sigma \times Q$: Transition relation
  - q_0: Initial State
  - F: Accepting States
- Takes a trajectory as input

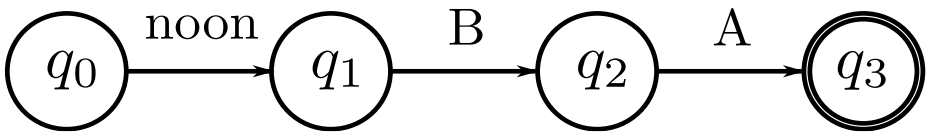


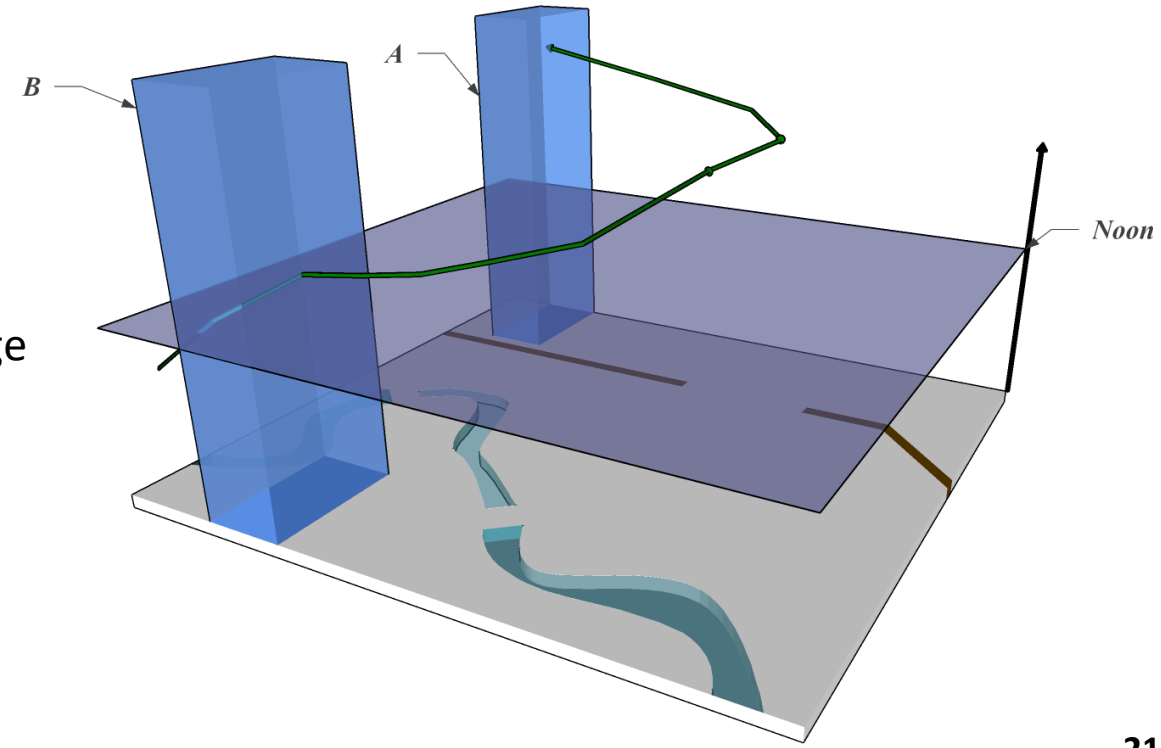
Reza Teshnizi (reza.teshnizi@gmail.com)

Behavior

- Accepted trajectories are members of a language describing the **behavior**.

- Input
  - $(Q, \Sigma, \Delta, q_0, F)$ is a behavior specification
  - $\tau: \mathbb{T} \rightarrow \mathcal{W}$ is a trajectory
- Output
  - Indicating whether τ belongs to the behavior language





Reza Teshnizi (reza.teshnizi@gmail.com)

I. Definitions and Formulation
II. Algorithm and Abstraction
III. Implementation

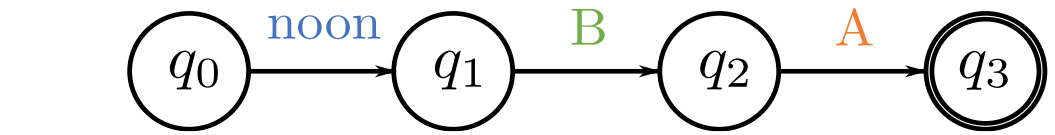
---

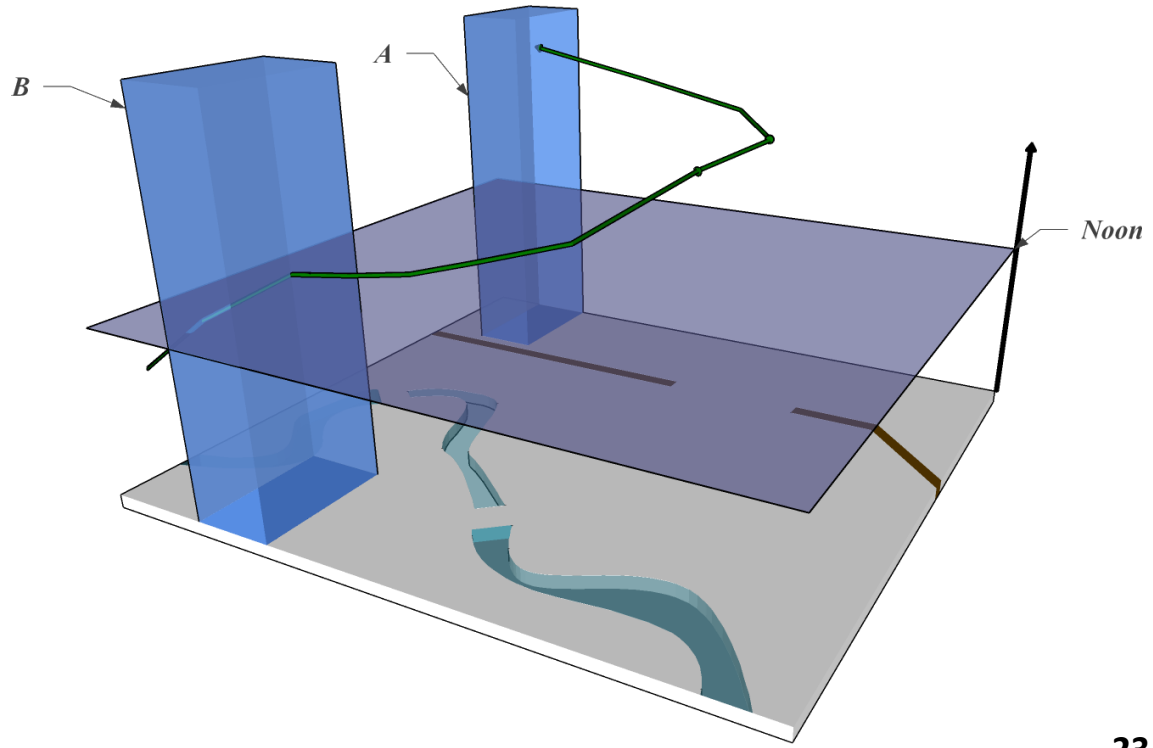
1. Formal Specifications
2. Matching and Filtering

Reza Teshnizi (reza.teshnizi@gmail.com)

Behavior Matching: Example 1

- Input
  - $\mathcal{W}$: the boundaries of free space,
  - $B = (Q, \Sigma, \Delta, q_0, F)$: a behavior specification,
  - τ: a trajectory.
- Solution either:
  - $t \in \mathbb{T}$, τ up to t is accepted by B, or
  - ∞ otherwise.





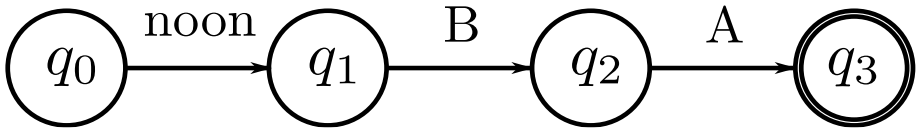
- $\exists t_1 \in \mathbb{T}: \tau(t_1) \in (\mathcal{W} \times \text{noon})$
- $\exists t_2 \in \mathbb{T}: t_2 \geq t_1 \text{ and } \tau(t_2) \in (B \times \mathbb{T})$
- $\exists t_3 \in \mathbb{T}: t_3 \geq t_2 \text{ and } \tau(t_3) \in (A \times \mathbb{T})$

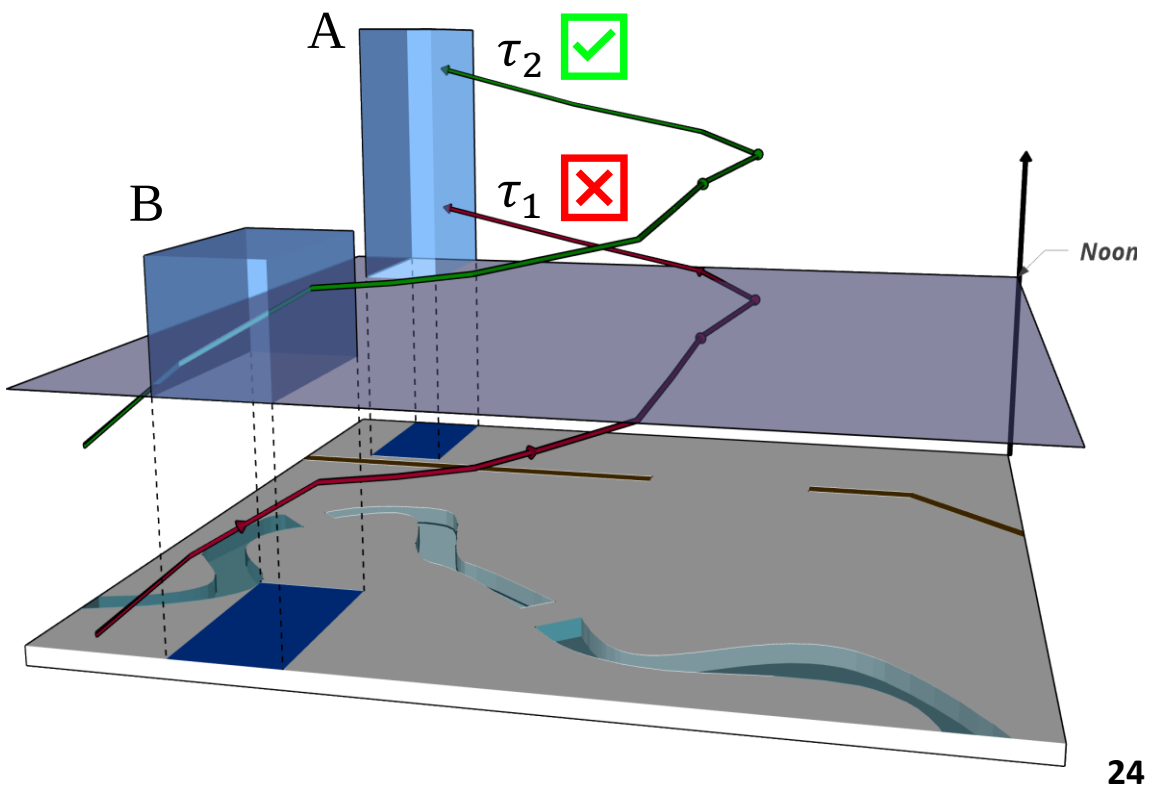
Reza Teshnizi (reza.teshnizi@gmail.com)

Behavior Matching: Example 1

- Input
  - $\mathcal{W}$: the free space,
  - $B = (Q, \Sigma, \Delta, q_0, F)$: a behavior specification,
  - τ: a trajectory.
- Solution either:
  - $t \in \mathbb{T}$, τ up to t is accepted by B, or
  - ∞ otherwise.

- $\exists t_1 \in \mathbb{T}: \tau(t_1) \in (\mathcal{W} \times \text{noon})$
- $\exists t_2 \in \mathbb{T}: t_2 \geq t_1 \text{ and } \tau(t_2) \in (B \times \mathbb{T})$
- $\exists t_3 \in \mathbb{T}: t_3 \geq t_2 \text{ and } \tau(t_3) \in (A \times \mathbb{T})$

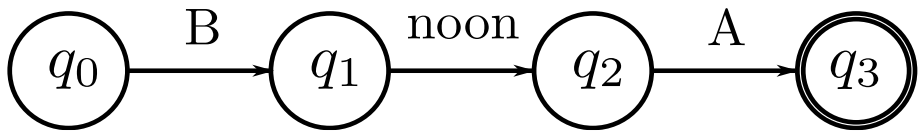


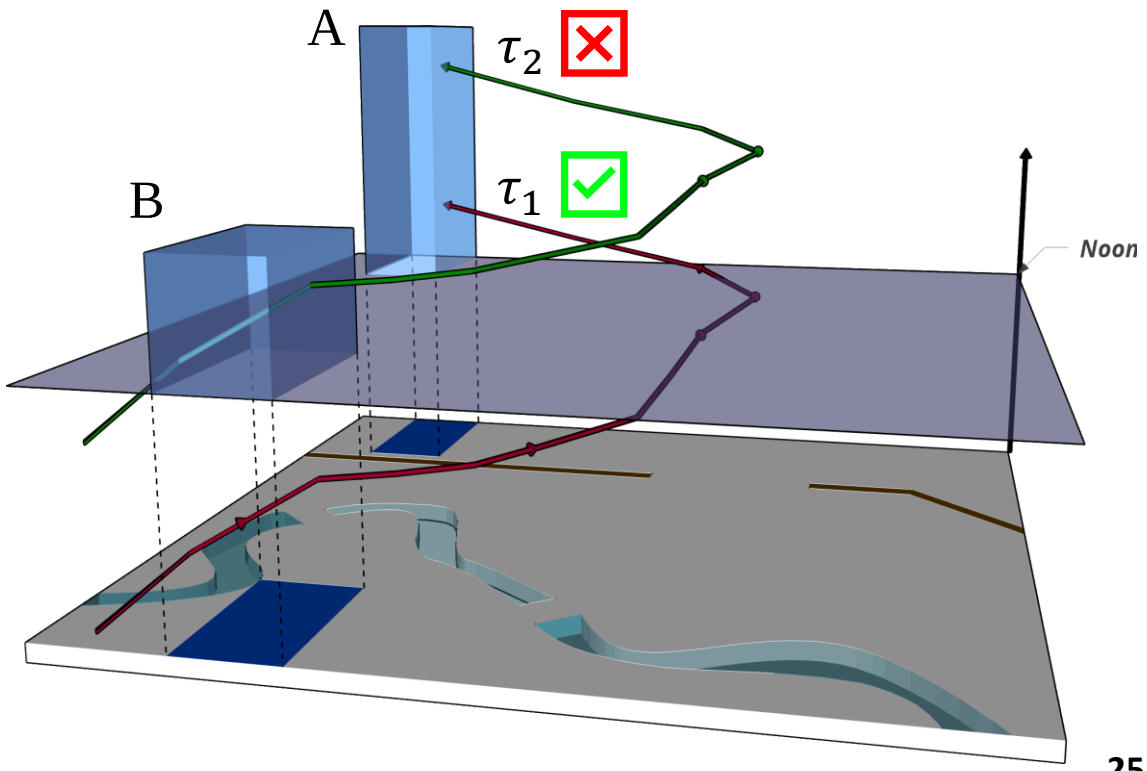


Reza Teshnizi (reza.teshnizi@gmail.com)

Behavior Matching: Example 2

- Input
  - $\mathcal{W}$: the free space,
  - $B = (Q, \Sigma, \Delta, q_0, F)$: a behavior specification,
  - τ: a trajectory.
- Solution either:
  - $t \in \mathbb{T}$, τ up to t is accepted by B, or
  - ∞ otherwise.



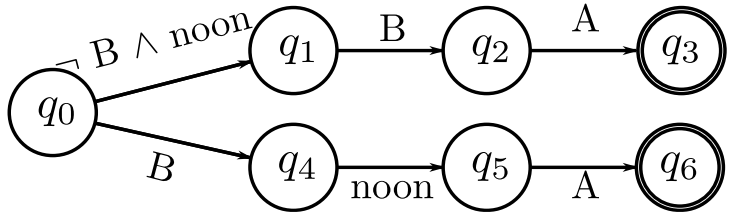


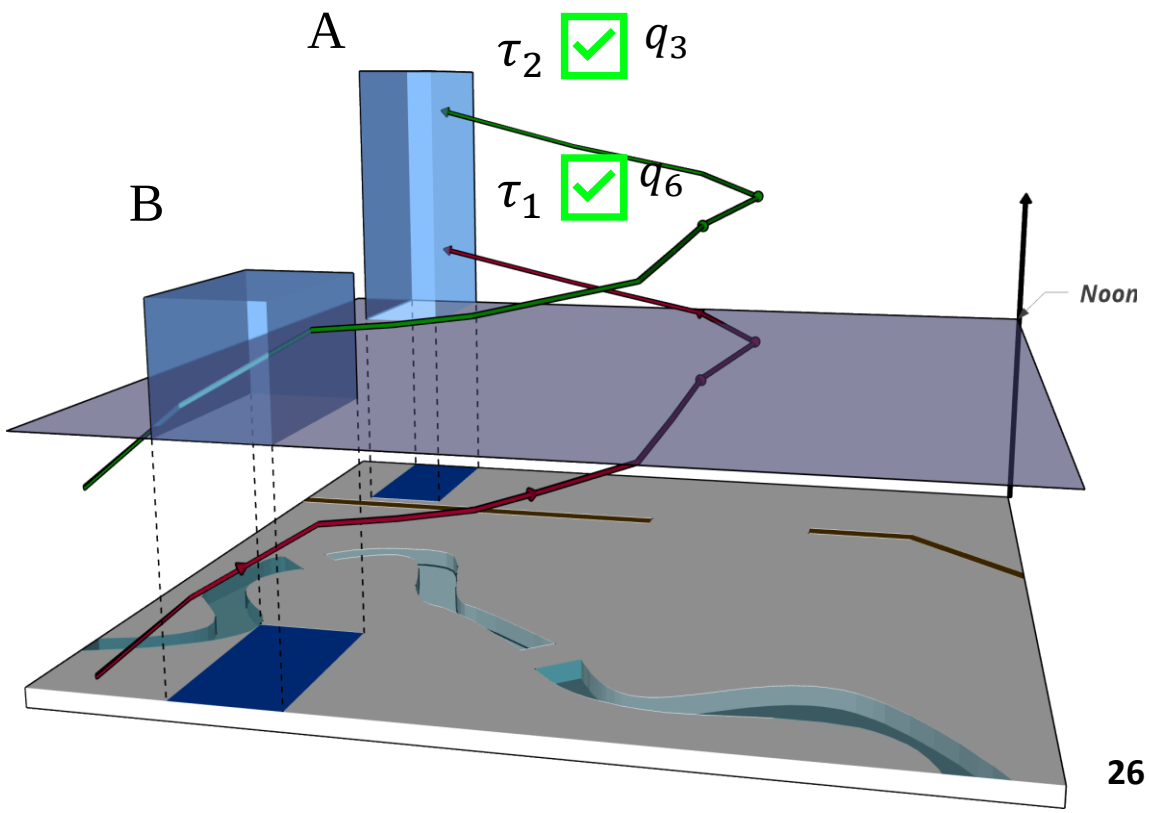
- $\exists t_1 \in \mathbb{T}: \tau(t_1) \in (B \times \mathbb{T})$
- $\exists t_2 \in \mathbb{T}: t_2 \geq t_1$ and $\tau(t_2) \in (\mathcal{W} \times \text{noon})$
- $\exists t_3 \in \mathbb{T}: t_3 \geq t_2$ and $\tau(t_3) \in (A \times \mathbb{T})$

Reza Teshnizi (reza.teshnizi@gmail.com)

Behavior Matching: Example 3

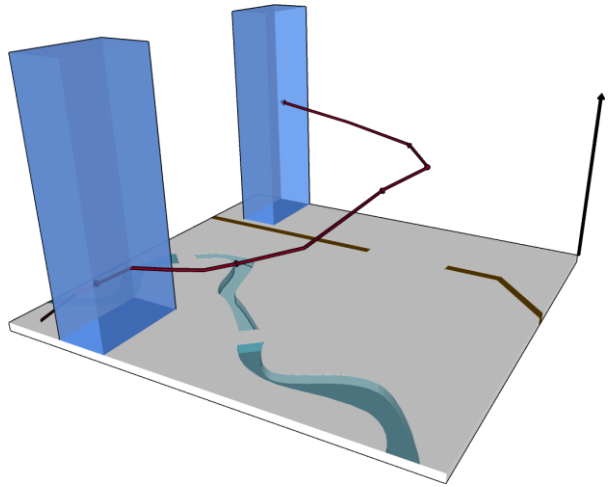
- Input
  - $\mathcal{W}$: the free space,
  - $B = (Q, \Sigma, \Delta, q_0, F)$: a behavior specification,
  - τ: a trajectory.
- Solution either:
  - $t \in \mathbb{T}$, τ up to t is accepted by B, or
  - ∞ otherwise.

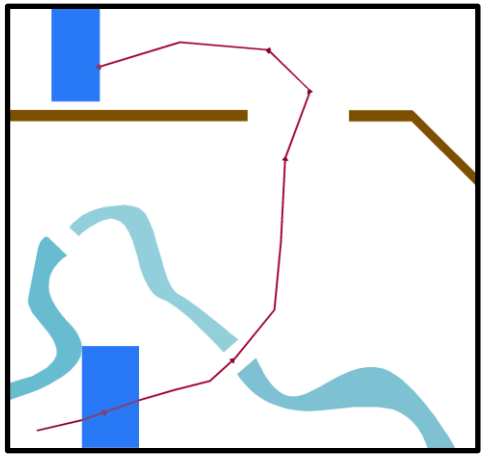




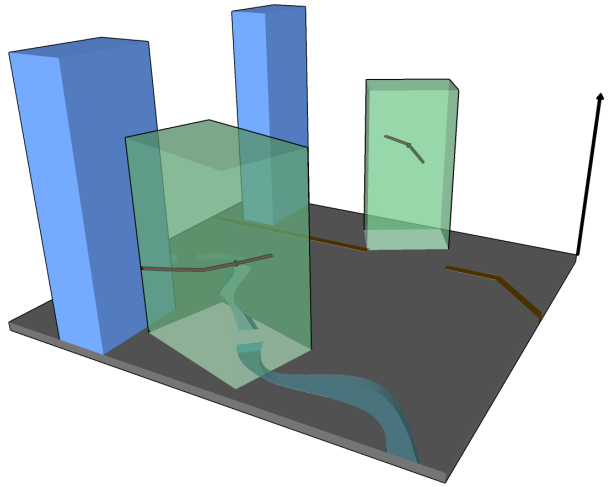
Reza Teshnizi (reza.teshnizi@gmail.com)

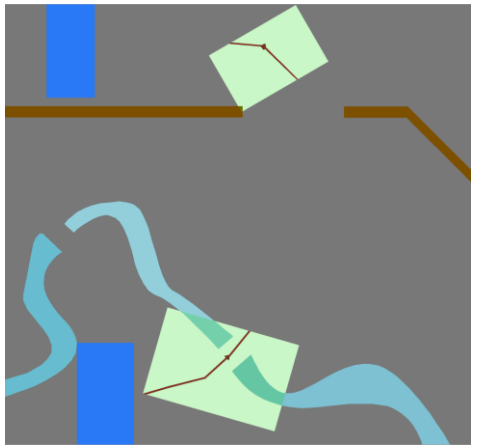
Partial Observability





Limited Field-of-view

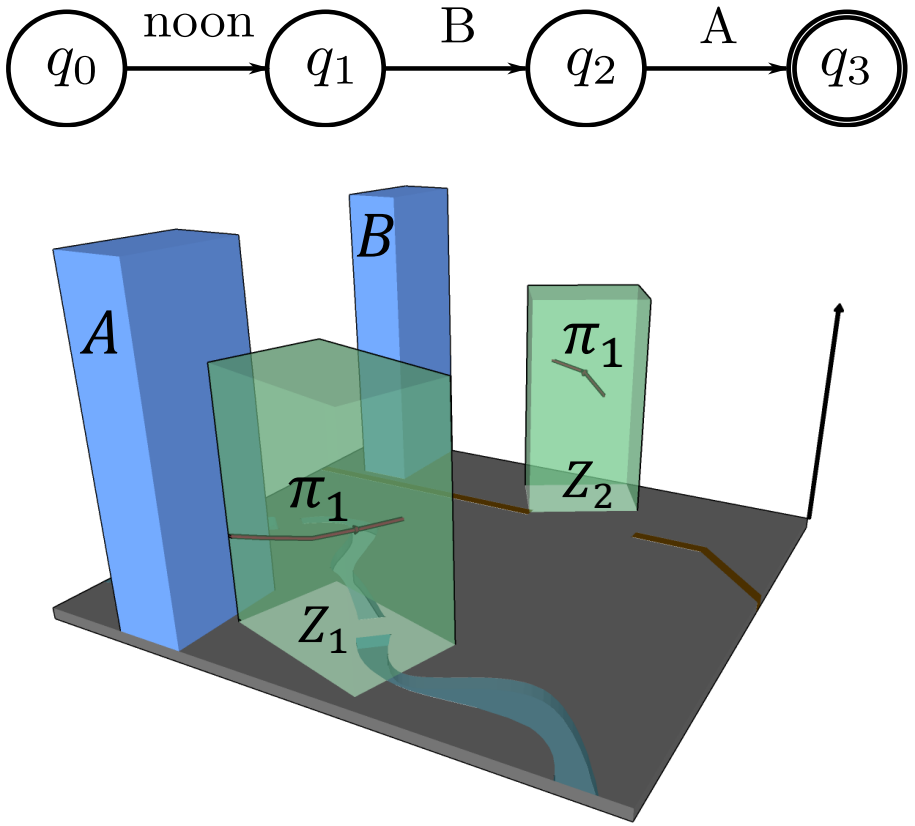




Reza Teshnizi (reza.teshnizi@gmail.com)

Behavior Filtering

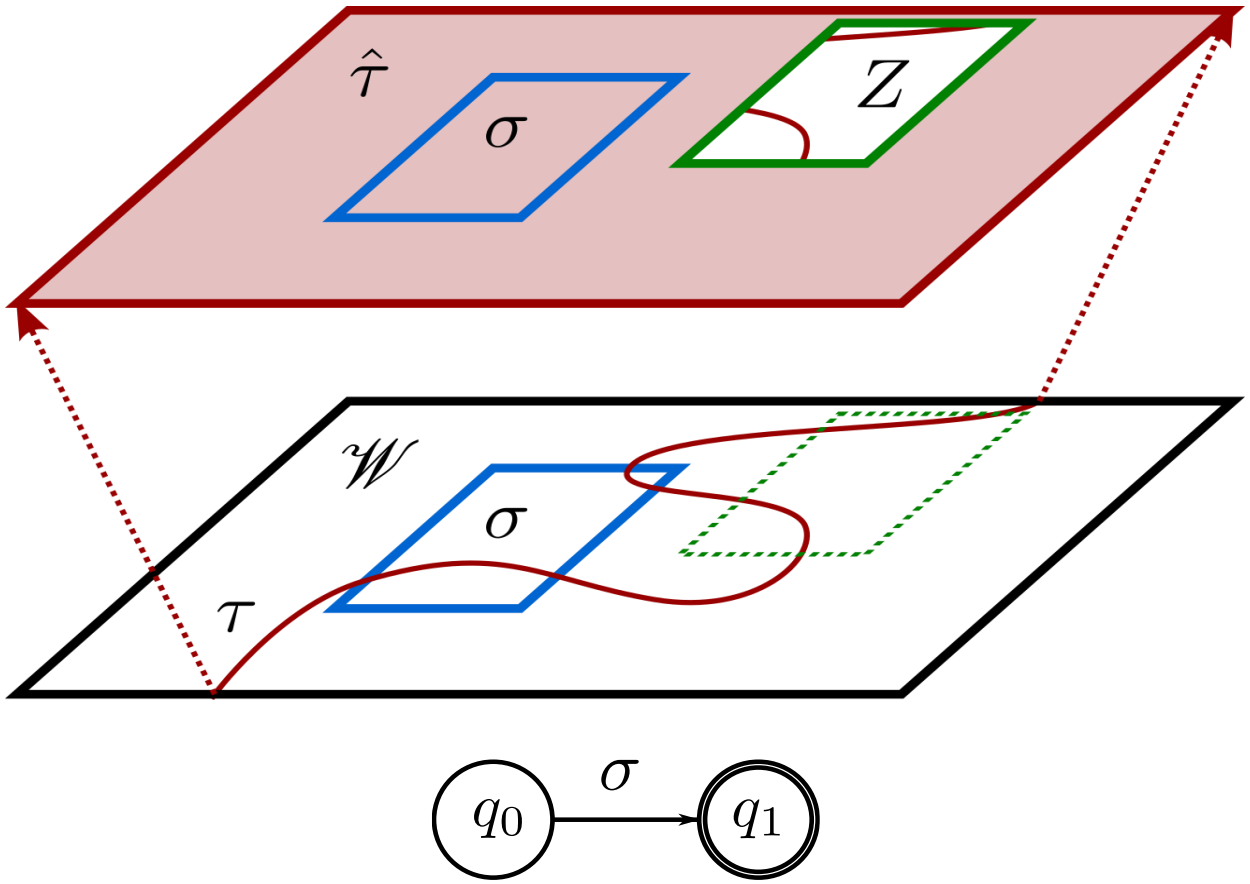
- Input
  - $\mathcal{W}$: the free space,
  - $B = (Q, \Sigma, \Delta, q_0, F)$: a behavior specification,
  - Z: field-of-view,
  - $\Pi = \{\pi_1, \pi_2, \ldots, \pi_{|\Pi|}\}$: a set of tracklets
    - $\pi_i: T_i \to Z$, and
    - $T_i \cap T_j = \emptyset$ for $i \neq j$.
- Solution either:
  - a trajectory $\tau: \mathbb{T} \to \mathcal{W}$
    - τ is accepted by B and
    - $\forall \pi_i \in \Pi: \big(t \in T_i \Rightarrow \pi_i(t) = \tau(t)\big)$
  - $\emptyset$ which indicates no such trajectory exists



Reza Teshnizi (reza.teshnizi@gmail.com)

Lifting From Points to Sets

- $\exists (x, y, t) \in \mathcal{W} \times \mathbb{T} : (x, y, t) \in \hat{\tau} \cap \sigma$
- $\exists t \in \mathbb{T} : \tau(t) \in \sigma$



Reza Teshnizi (reza.teshnizi@gmail.com)

I. Definitions and Formulation

II. Algorithm and Abstraction

III. Implementation

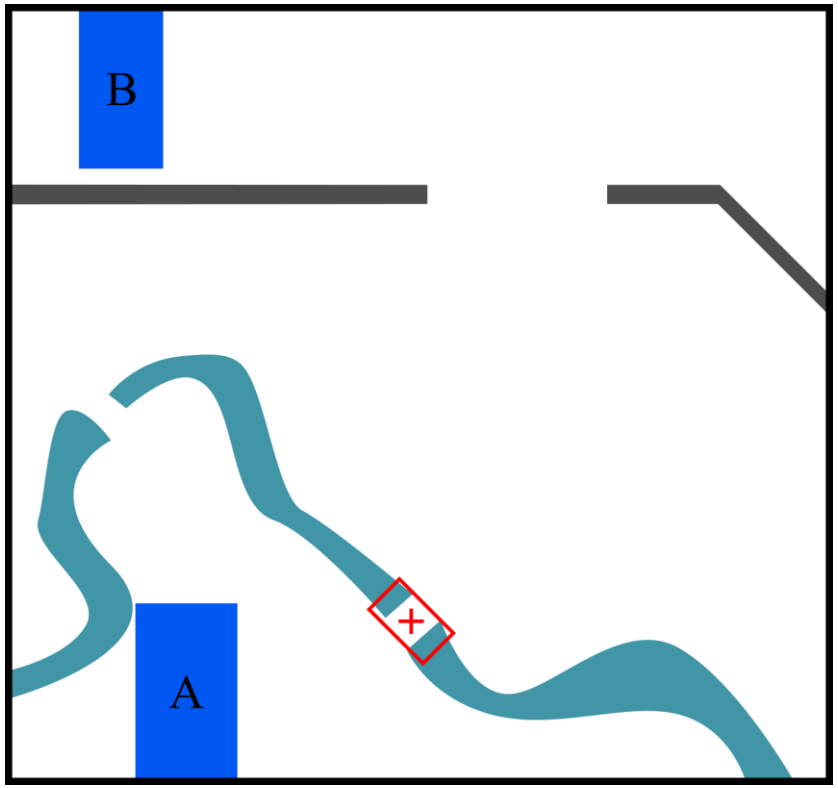
---

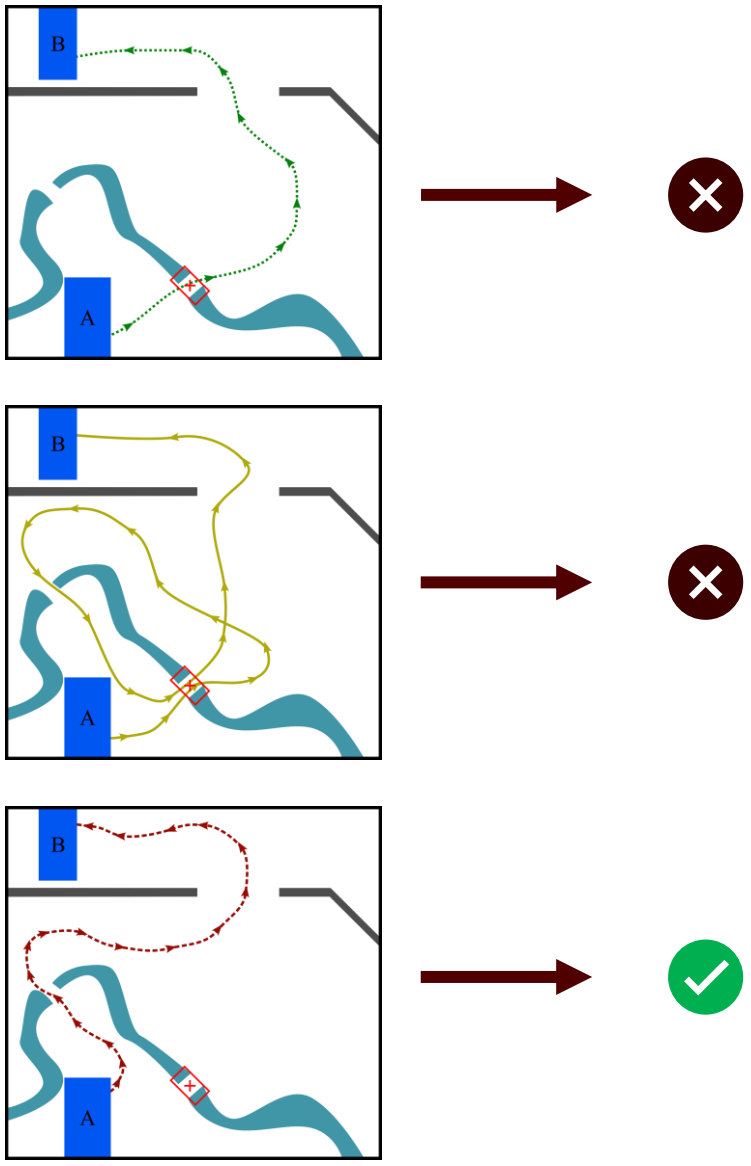
1. Topological Representation
2. Logical Descriptions

Reza Teshnizi (reza.teshnizi@gmail.com)

Combinatorial Motion Analysis

- Movements are continuous

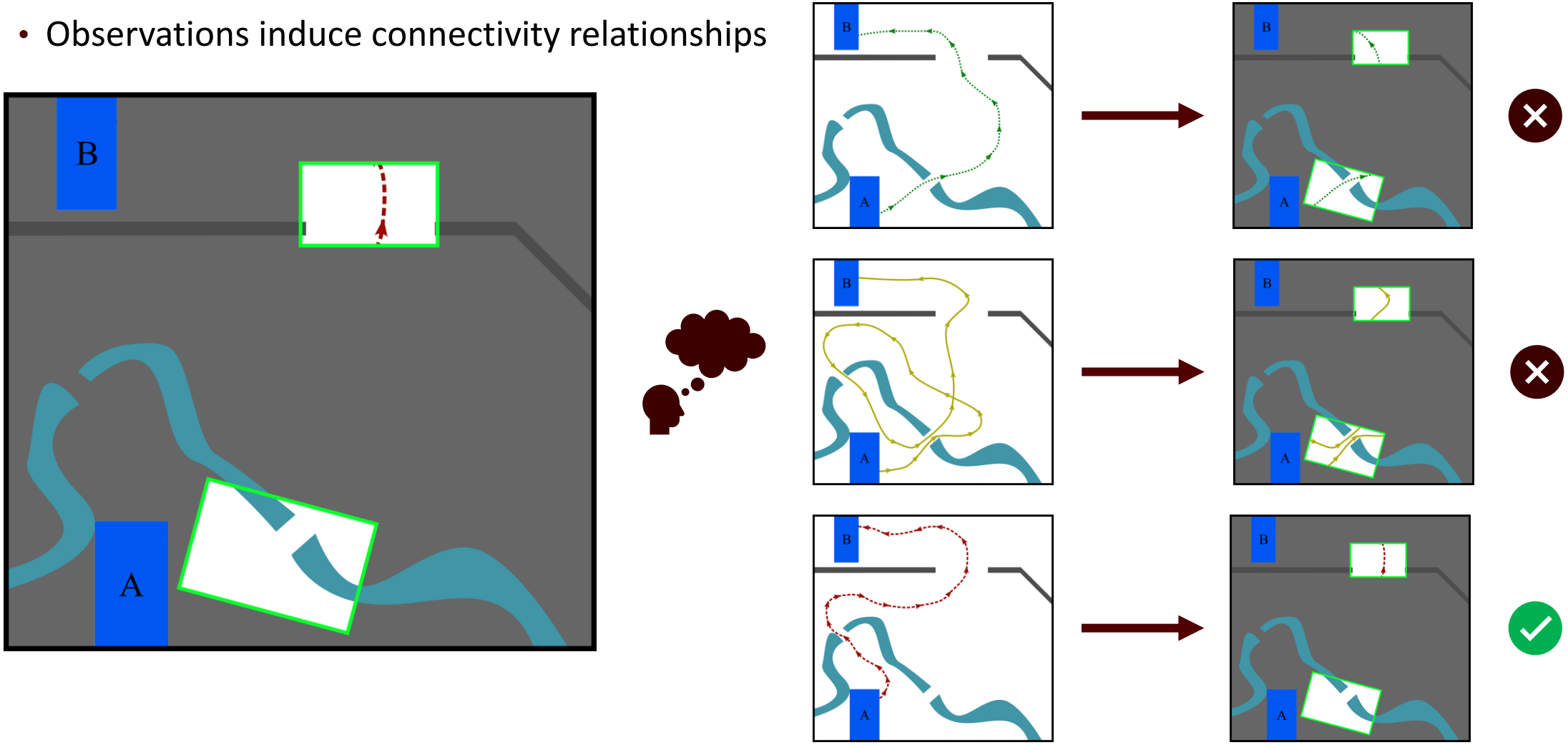




Reza Teshnizi (reza.teshnizi@gmail.com)

Field-of-view Observations

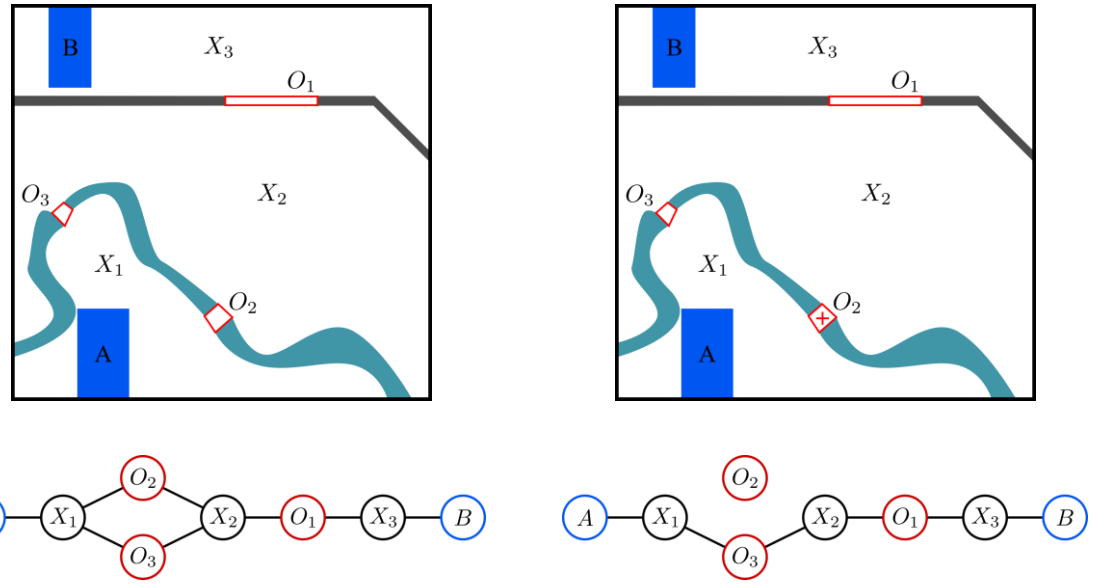
- Observations induce connectivity relationships



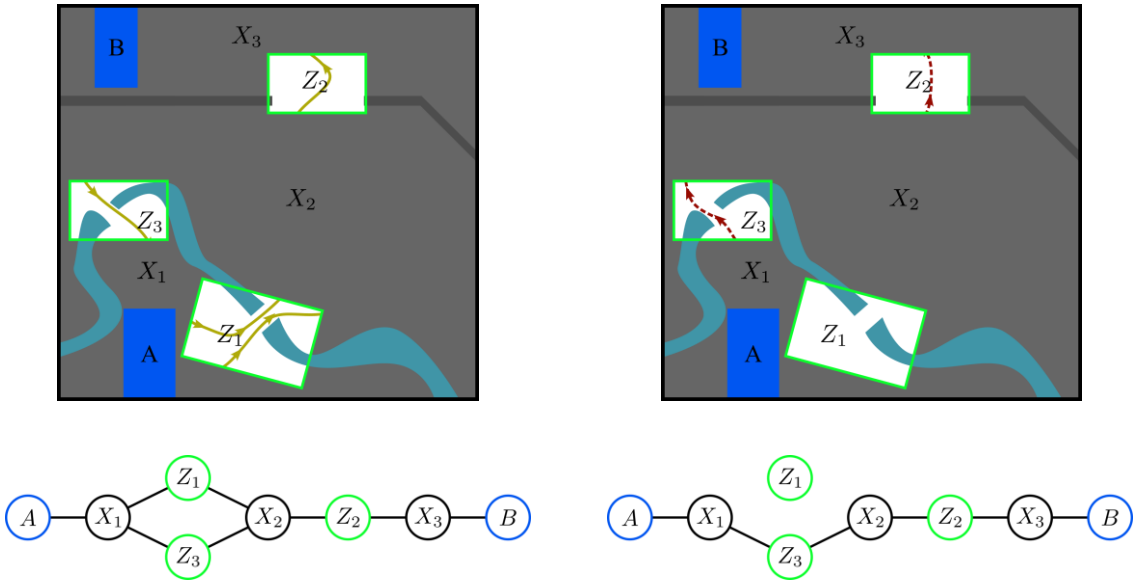
Reza Teshnizi (reza.teshnizi@gmail.com)

Spatial Decomposition

Dynamic Spaces



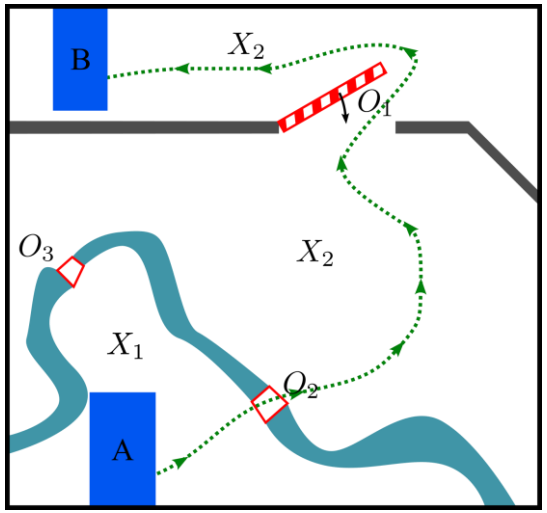
Field-of-view

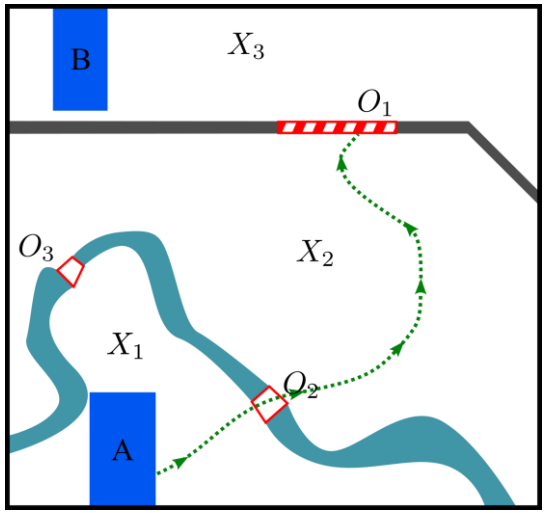


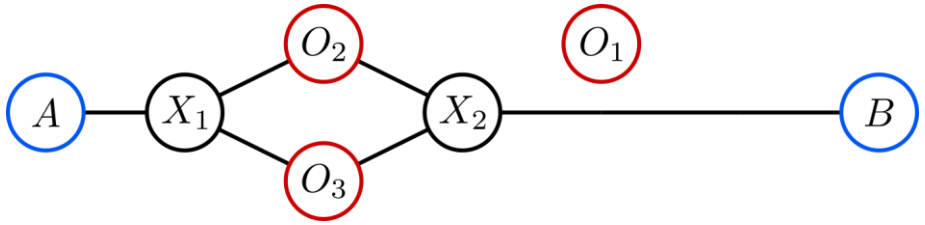
The possible trajectories of a moving target **induces** equivalence relationships over subspaces.

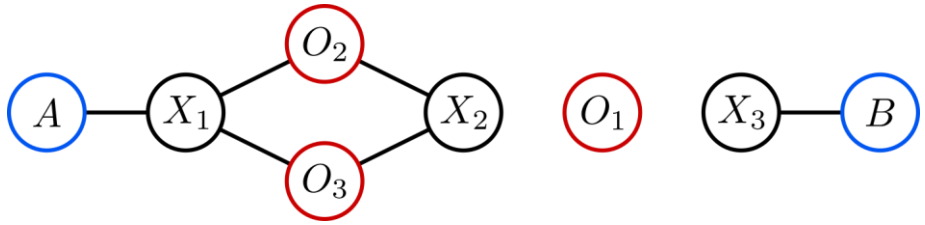
Reza Teshnizi (reza.teshnizi@gmail.com)

Temporal Decomposition







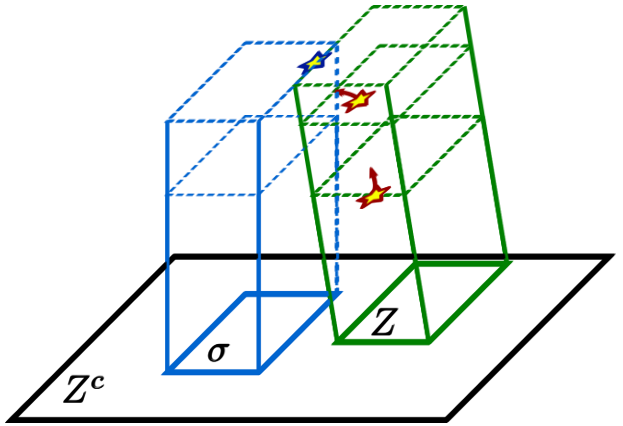


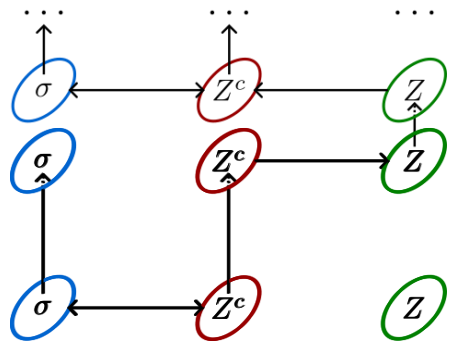
Movements of path-connected components **change** equivalence relationships over subspaces.

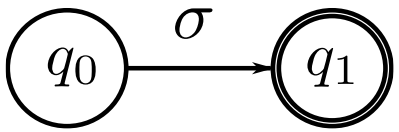
Reza Teshnizi (reza.teshnizi@gmail.com)

Overall Procedure

- Decompose the domain of time
  - Disjoint intervals containing at most one event
  - Dynamic obstructions, target observations, or collision among boundaries of moving components
- Compute connectivity graph
  - Decompose the domain of space into path-connected components for a given target
- Compute information graph
  - Determine spatiotemporal connectivity between the nodes of the successive connectivity graphs
- Search for paths
  - Find paths on the information graph that satisfy the alphabet and end in an accepting state.







Reza Teshnizi (reza.teshnizi@gmail.com)

I. DEFINITIONS AND FORMULATION

II. ALGORITHM AND ABSTRACTION

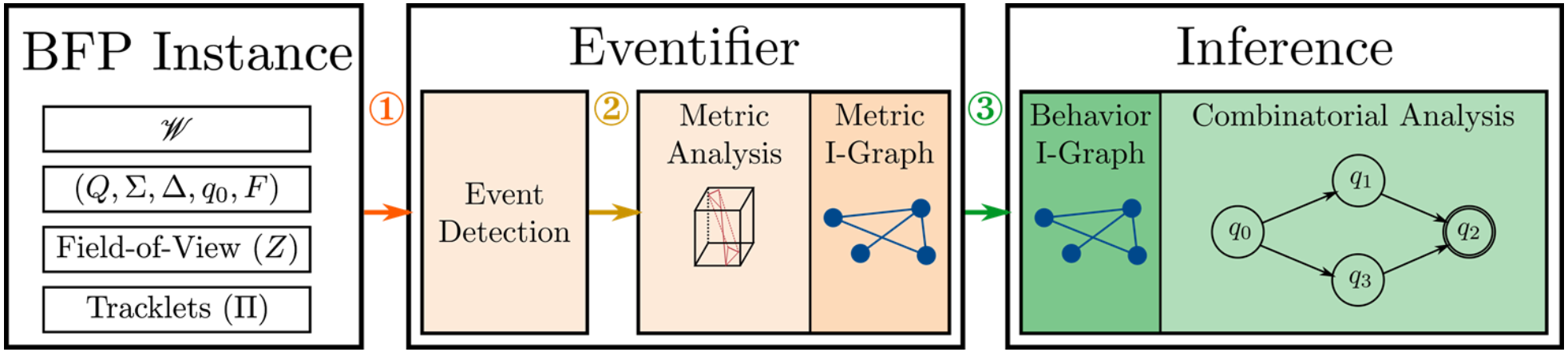
III. IMPLEMENTATION

---

1. Topological Representation
2. Logical Descriptions

Reza Teshnizi (reza.teshnizi@gmail.com)

Domains of Analysis



① Metric information about the boundaries of space and time.

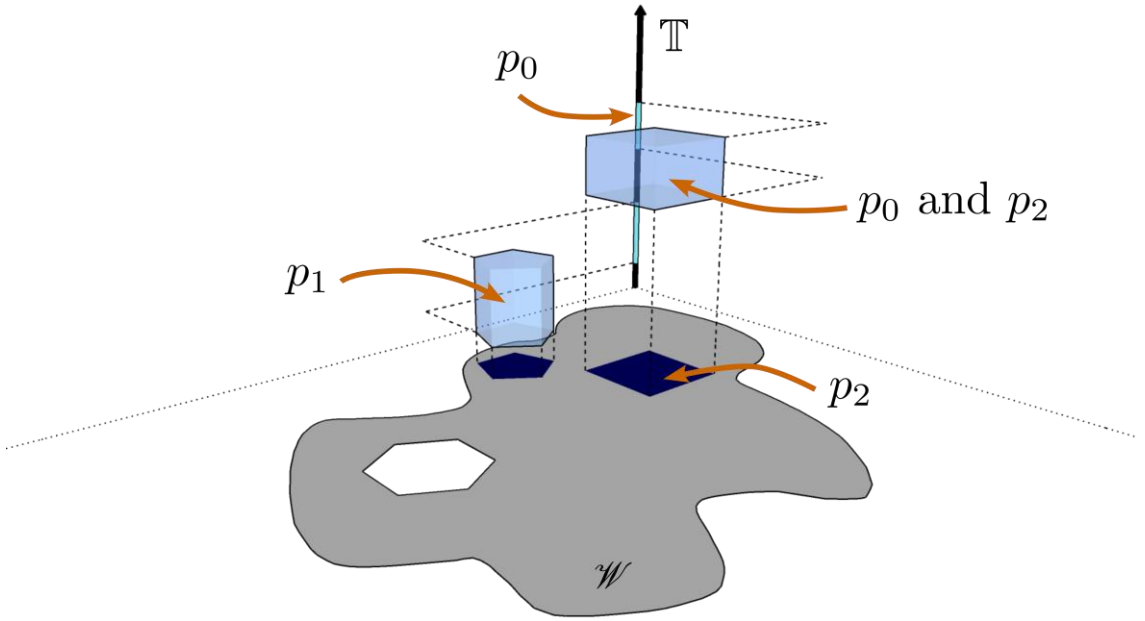
② Metric information decomposed temporally according to critical events.

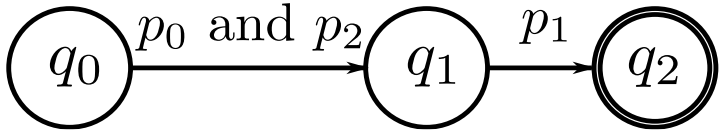
③ Information graph representations of spatiotemporal connectivity.

Reza Teshnizi (reza.teshnizi@gmail.com)

Propositional Behavior Automata

- Descriptions of space-time sets
  - PROP $= \{p_1, p_2, \dots, p_n\}$
  - is an office, is a restaurant, …
  - is a weekend, is a morning, …
  - is a muddy road, is green, …
- Well-formed Propositional Formula (WFF)
  - "is an office **and <u>not</u>** is a weekend"
- Propositional BA $(Q, \Sigma, \Delta, q_0, F)$
  - $\Sigma \subseteq$ WFF over symbols in PROP

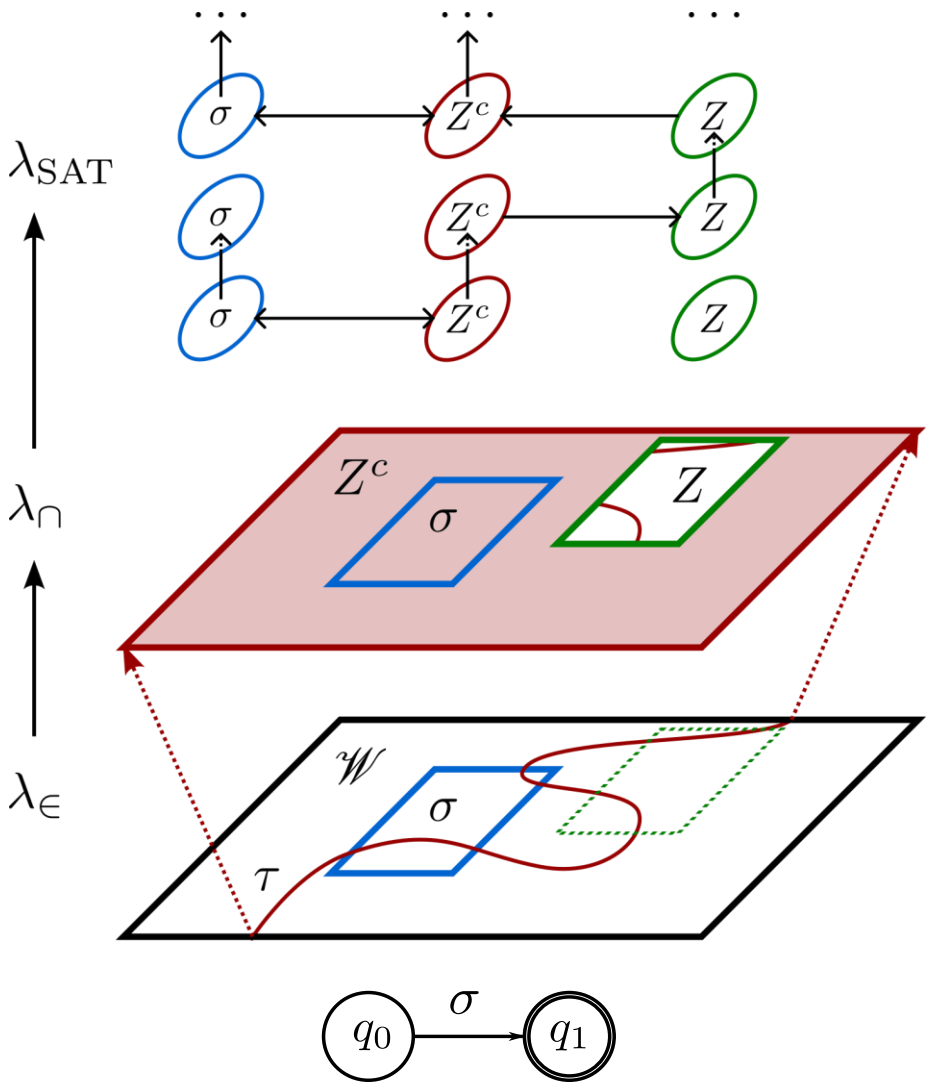




Reza Teshnizi (reza.teshnizi@gmail.com)

Descriptive Behavior Filtering

- Input
  - $\mathcal{W}$: the free space,
  - PROP: atomic predicate symbols,
  - I: an interpretation of WFF,
  - $B = (Q, \Sigma, \Delta, q_0, F)$: a behavior specification,
  - Z: field-of-view,
  - $\Pi = \{\pi_1, \pi_2, \ldots, \pi_{|\Pi|}\}$: a set of tracklets
- Solution either:
  - a trajectory $\tau: \mathbb{T} \to \mathcal{W}$
  - $\emptyset$ which indicates no such trajectory exists



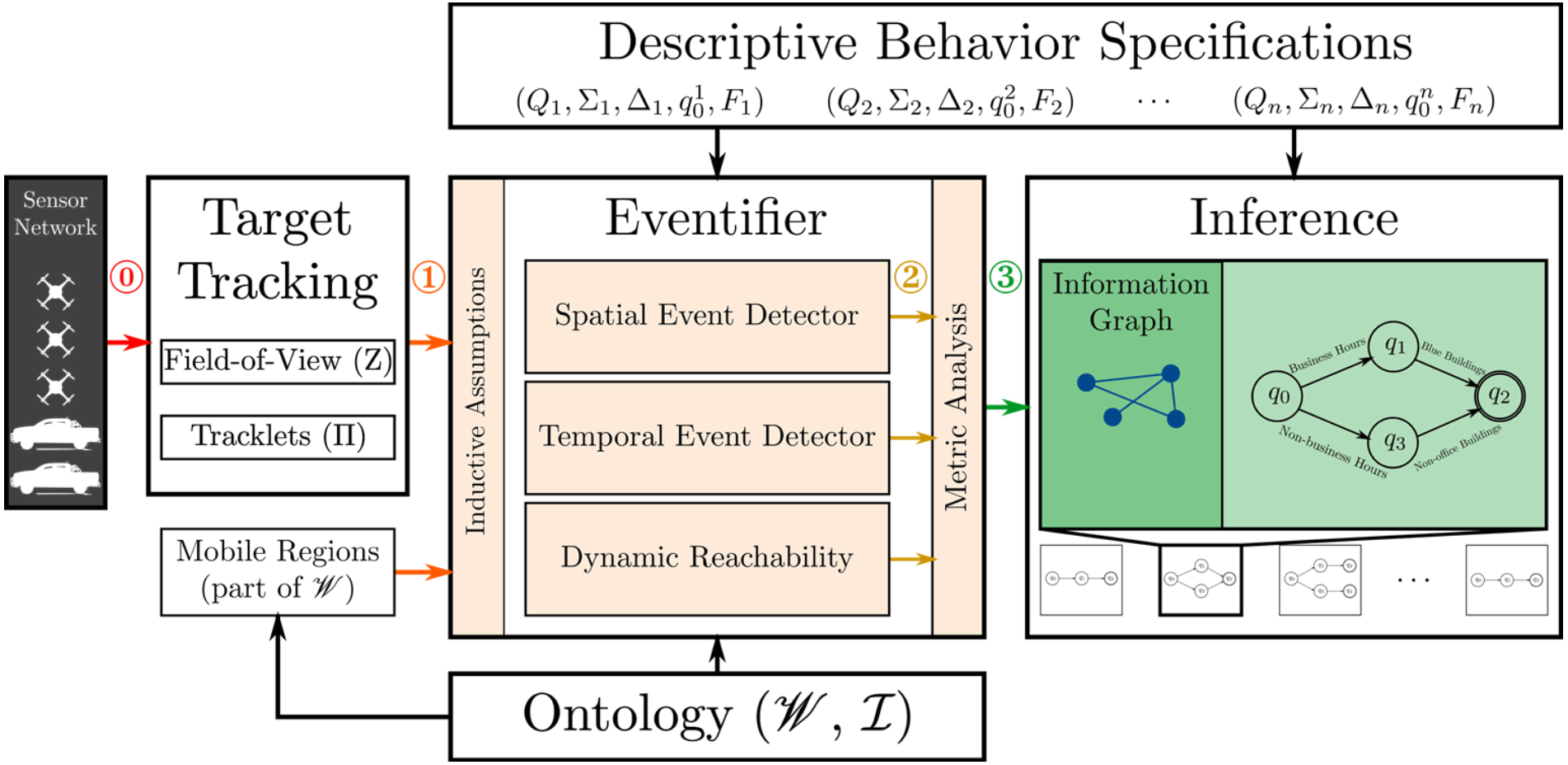
Reza Teshnizi (reza.teshnizi@gmail.com)

I. Definitions and Formulation
II. Algorithm and Abstraction
III. Implementation

1. Modular Architecture
2. Demonstration

Reza Teshnizi (reza.teshnizi@gmail.com)

Perception Architecture

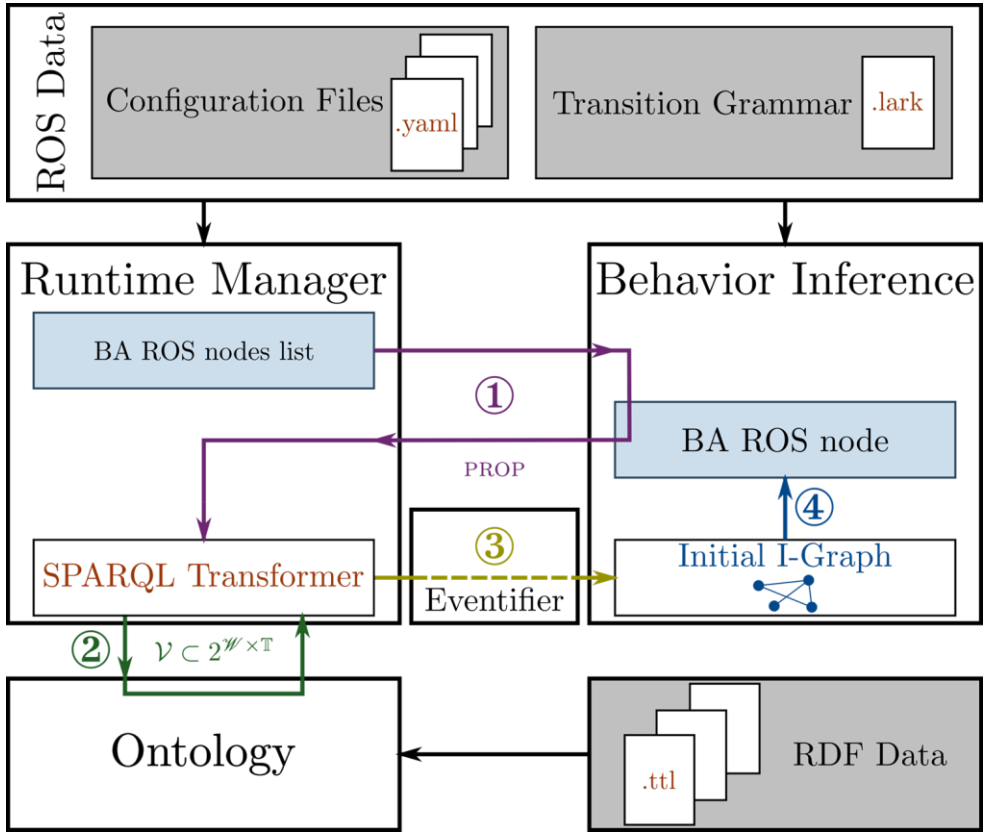


Reza Teshnizi (reza.teshnizi@gmail.com)

Interfacing with an Ontology

- Automaton $(Q, \Sigma, \Delta, q_0, F)$
  - PROP $= \{p_1, p_2, \dots, p_n\}$
  - $\Sigma \subseteq$ WFF over symbols in PROP

- $O: \mathcal{V} \rightarrow 2^{\text{PROP}}$
  - $\mathcal{V} = \{V_1, V_2, \dots, V_n\}$,
  - $V_i \subseteq \mathcal{W} \times \mathbb{T}$,
  - $V_i \neq V_j \Leftrightarrow O(V_i) \neq O(V_j)$, and
  - $\forall (x, y, t) \in V_i: p_i \in \text{PROP} \Leftrightarrow (x, y, t) \vDash p_i$.



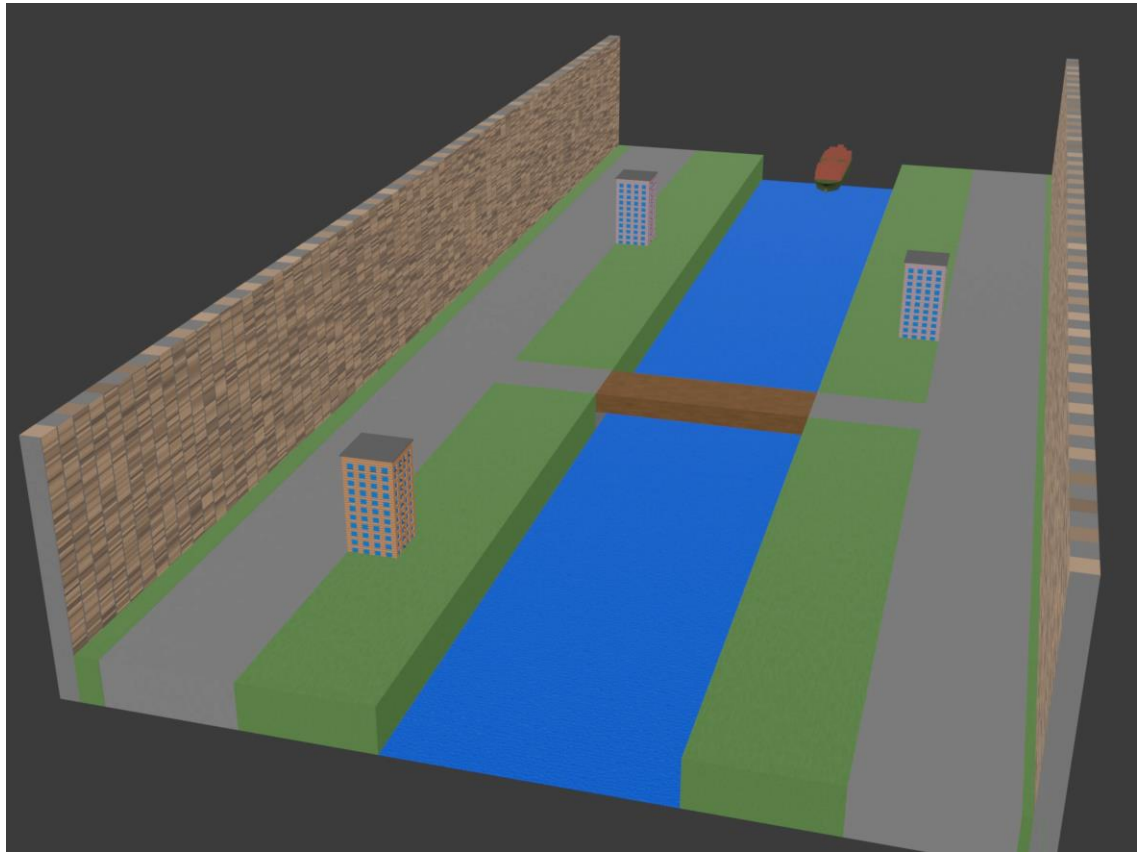
Reza Teshnizi (reza.teshnizi@gmail.com)

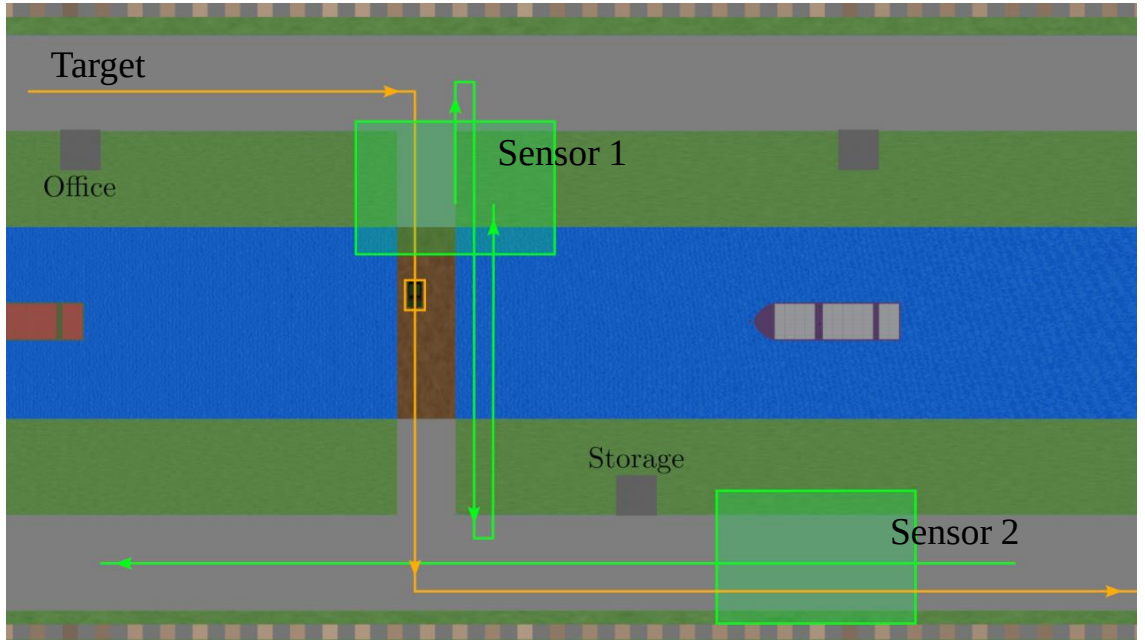
I. Definitions and Formulation
II. Algorithm and Abstraction
III. Implementation

1. Modular Architecture
2. Demonstration

Reza Teshnizi (reza.teshnizi@gmail.com)

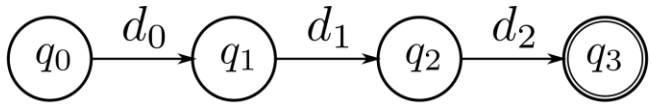
Demonstration: Environment





Reza Teshnizi (reza.teshnizi@gmail.com)

Demonstration: Specification

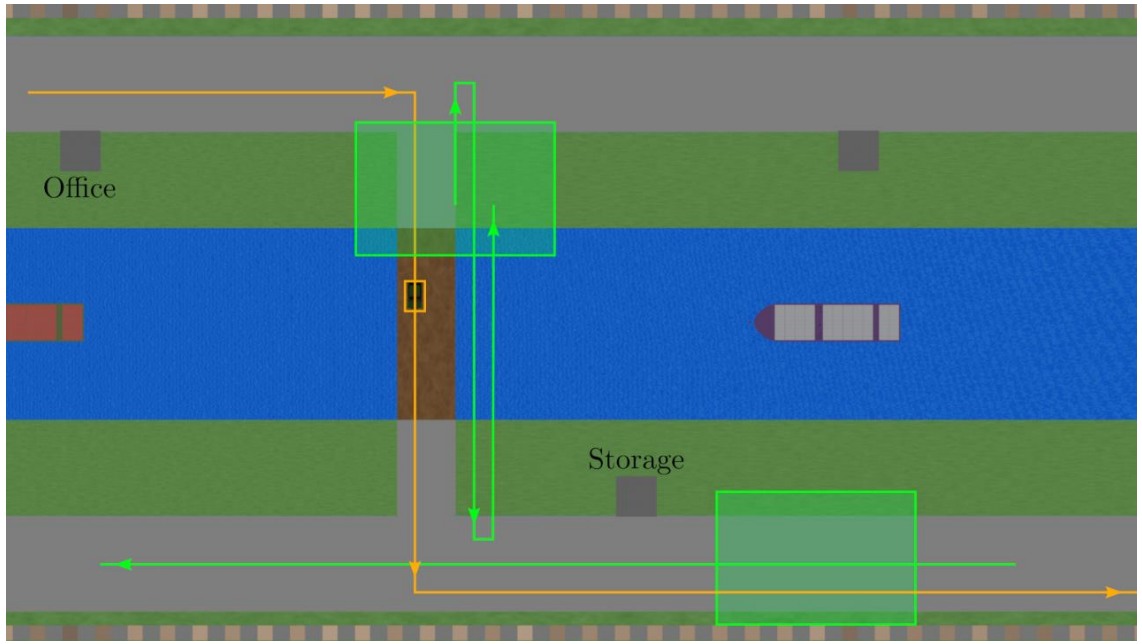



```
d0: name == "office" AND name == "morning"
d1: material.name == "street" OR material.name == "bridge"
d2: name == "storage"
```

↓Symbolized

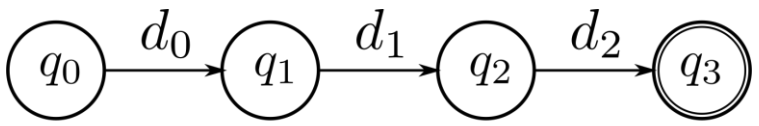
```
p0: material.name == "street"
p1: name == "morning"
p2: name == "storage"
p3: material.name == "bridge"
p4: name == "office"

d0: p4 AND p1
d1: p0 OR p3
d2: p2
```



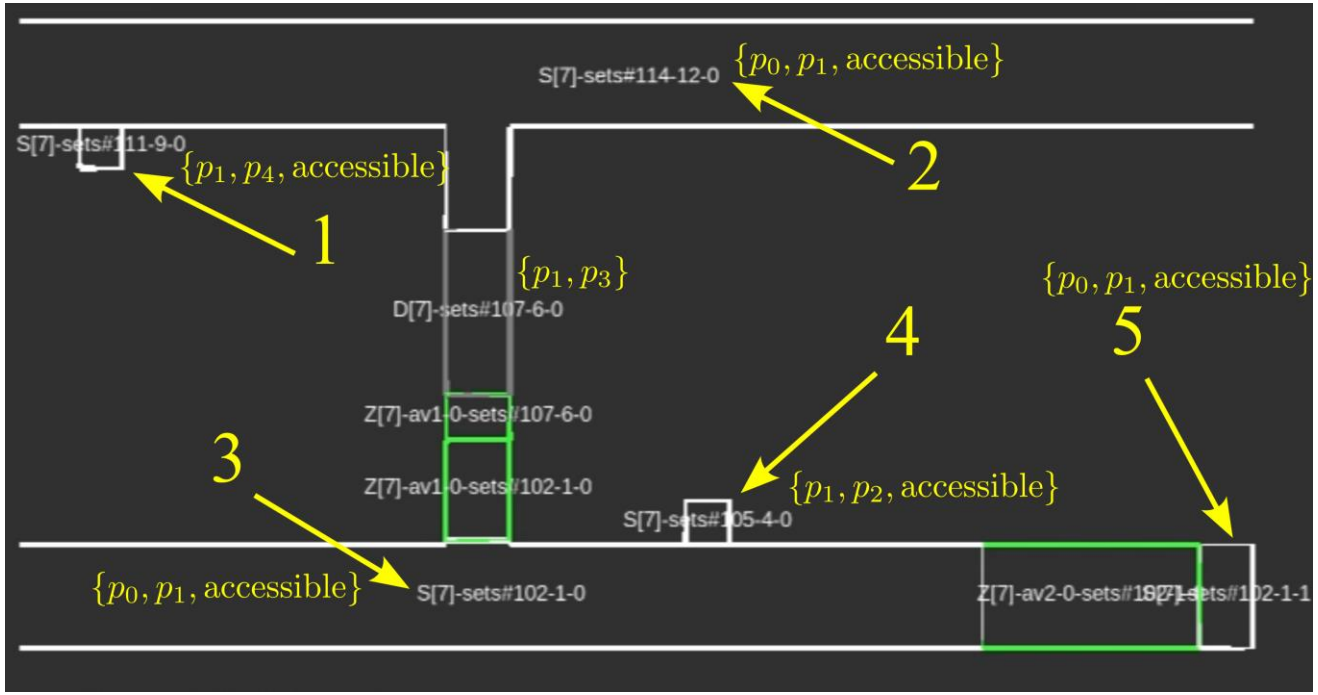
Reza Teshnizi (reza.teshnizi@gmail.com)

Demonstration: Initialization



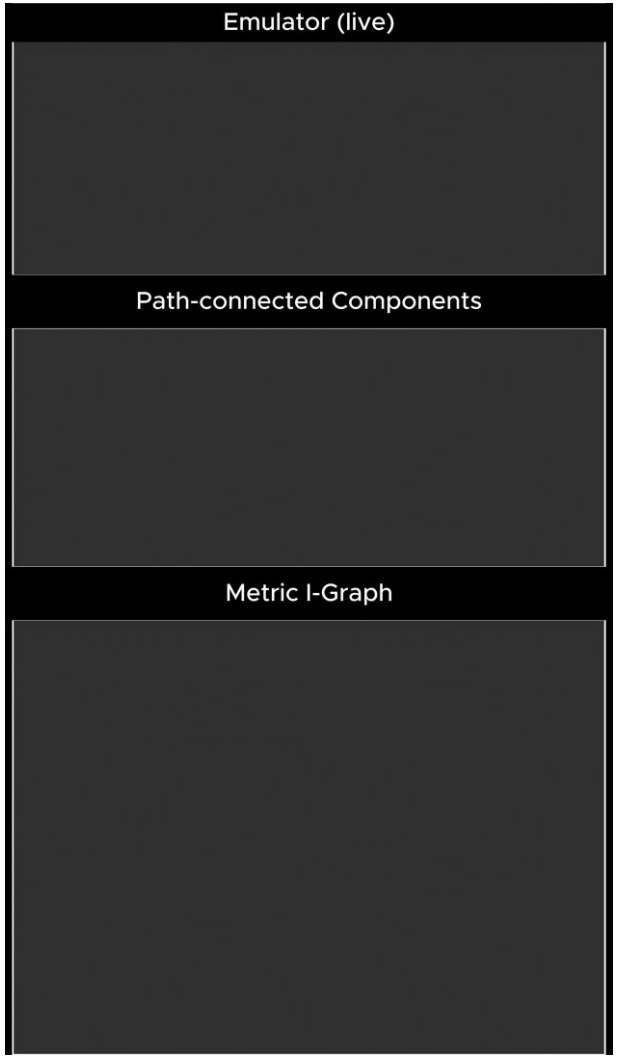

```
d0: p4 AND p1
d1: p0 OR p3
d2: p2

p0: material.name == "street"
p1: name == "morning"
p2: name == "storage"
p3: material.name == "bridge"
p4: name == "office"
```



Reza Teshnizi (reza.teshnizi@gmail.com)

Demonstration



Reza Teshnizi (reza.teshnizi@gmail.com)

CONCLUSION

Reza Teshnizi (reza.teshnizi@gmail.com)

Research Contributions

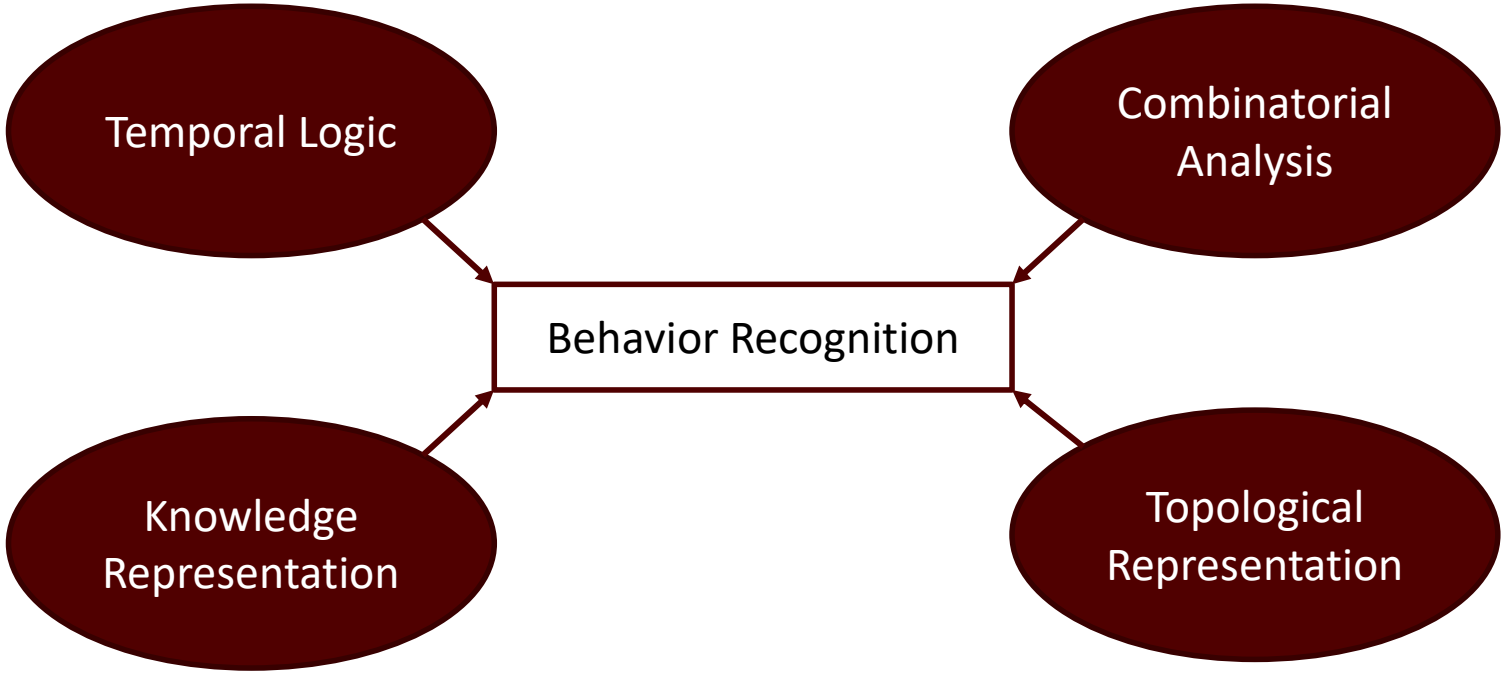
1. Formalism of Motion-based Behavior Recognition
   - Automata-based specifications
   - Formulation of recognition problems

2. Algorithmic Filtering via Topological Representations
   - Minimal graph structures for the solution space
   - Logical abstraction of metric information

3. Perception Process Enabled with Behavior Inference
   - Modular architecture to integrate behavior models
   - Interfacing with a network of sensors and an ontology

Reza Teshnizi (reza.teshnizi@gmail.com)

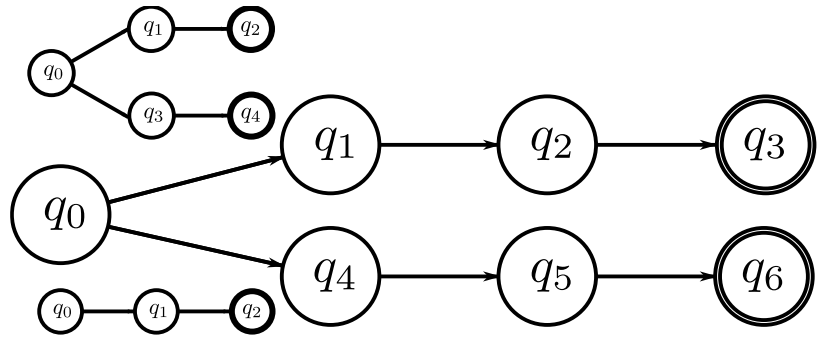
Summary



Reza Teshnizi (reza.teshnizi@gmail.com)

Future Extensions

- Multi-agent behavior specification
  - Spatiotemporal constraints over multiple trajectories
- Hierarchical specifications
  - Automata that use other automata as transition predicates
- Decentralized architecture
- A measure of estimation confidence



Acknowledgments

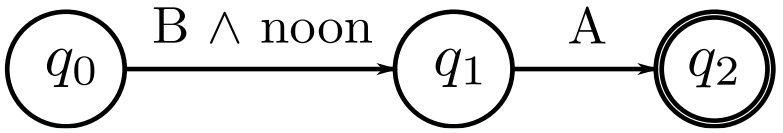
- The Bush Combat Development Complex at Rellis Campus
  - Dr. Swaminathan Gopalswamy
  - Dr. Lucas Krakow
  - Dr. Dylan Shell
- DAIR Lab
  - Dr. Diptanil Chaudhuri and Reza Oftadeh
- My parents and friends.

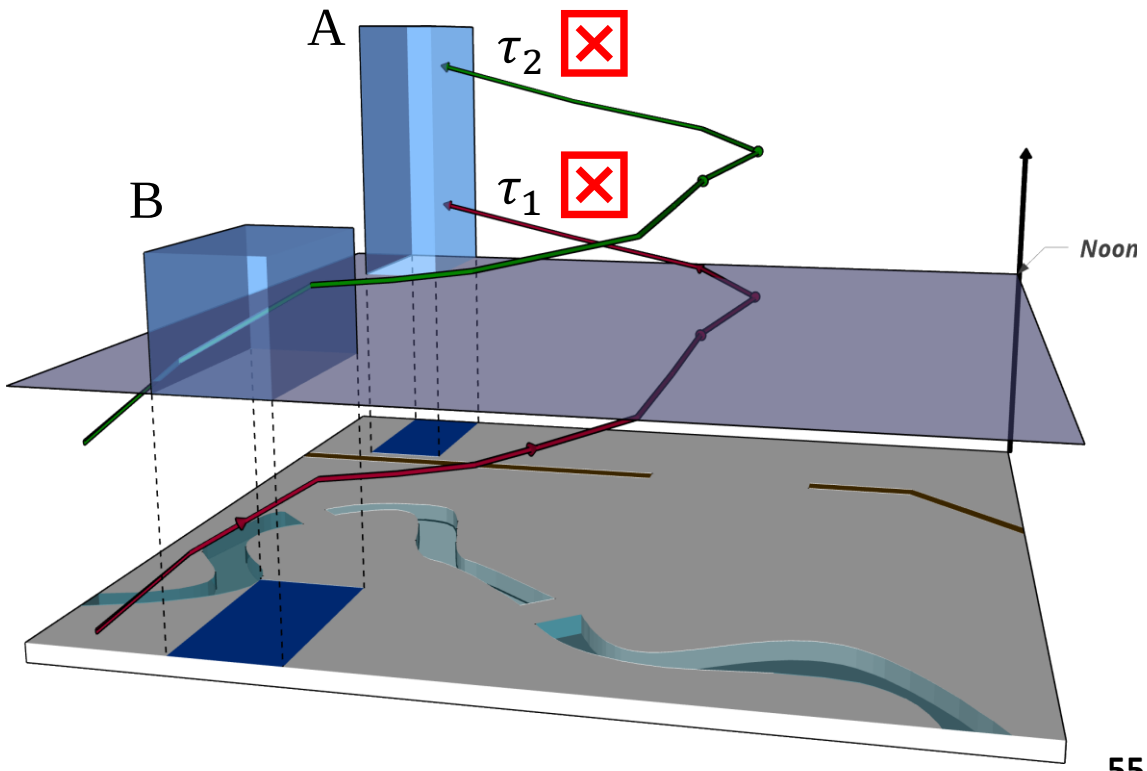
Reza Teshnizi (reza.teshnizi@gmail.com)

EXTRA CONTENT

Behavior Matching: Example 4

- Input
  - $\mathcal{W}$: the free space,
  - $B = (Q, \Sigma, \Delta, q_0, F)$: a behavior specification,
  - τ: a trajectory.
- Output either:
  - $t \in \mathbb{T}$, τ up to t is accepted by B, or
  - ∞ otherwise.

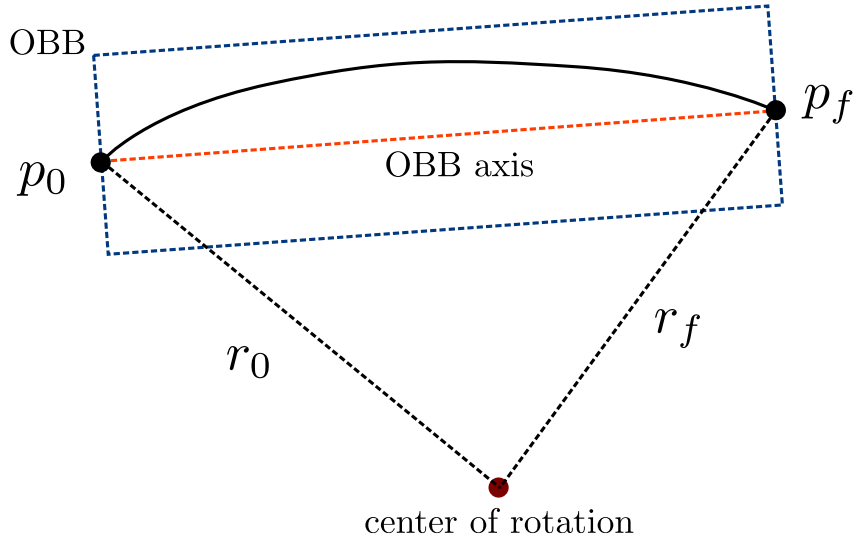




- $\exists t_1 \in \mathbb{T}: \tau(t_1) \in (B \times \text{noon})$
- $\exists t_2 \in \mathbb{T}: t_2 \geq t_3$ and $\tau(t_2) \in (A \times \mathbb{T})$

Continuous Time Collision Detection

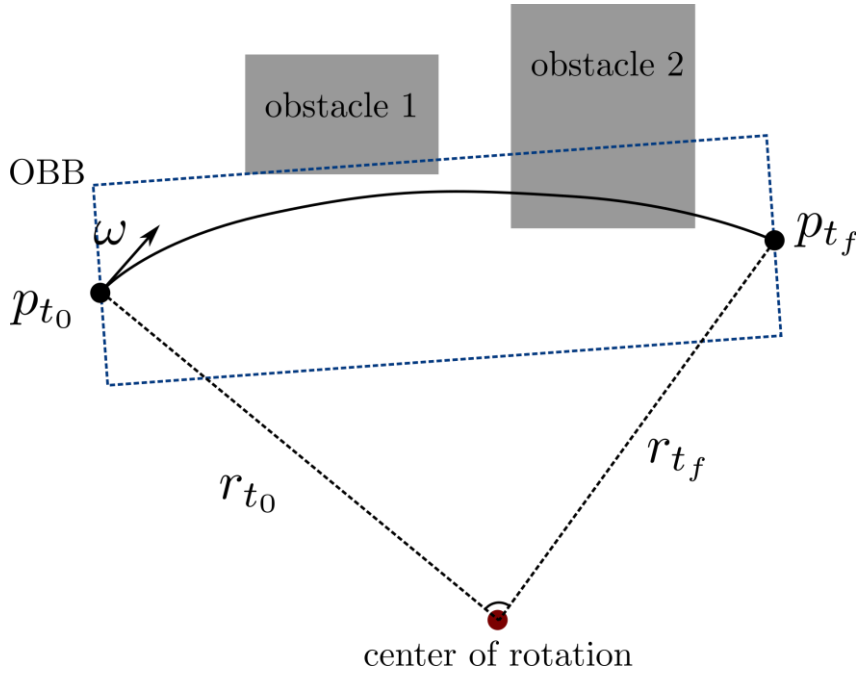
- Rigid body motion Bounding Box
  - rigid body transformation t_0 to t_f
  - Distance from center of rotation r
  - Fixed angular velocity ω
- Extend by $r \times \omega \times \Delta t$



SCIAVICCO, L. AND SICILIANO, B. 2000. Modeling and Control of Robot Manipulators, 2nd Ed. Springer-Verlag, London.

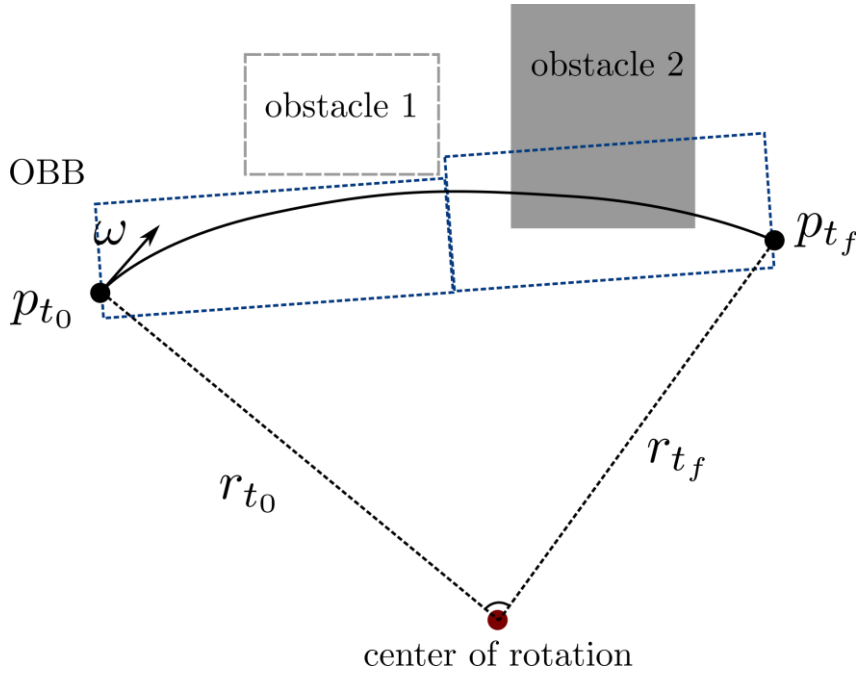
Continuous Time Collision Detection

- Check BB against all the edges
- Make a list of the colliding edges



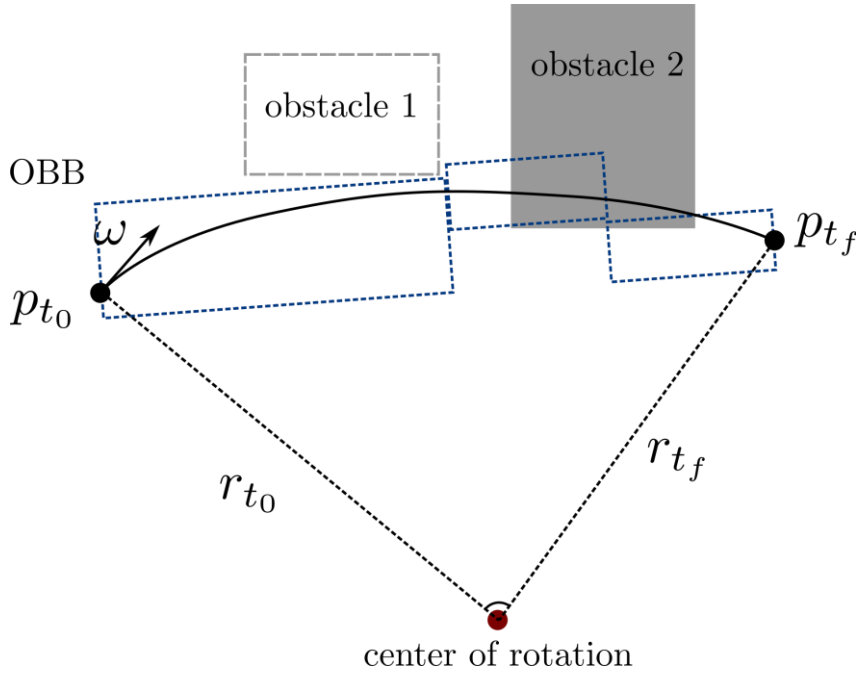
Continuous Time Collision Detection

- Cut the interval in half
- Create BB for the smaller intervals
- Test against the list of edges from previous step
- Remove any edge that does not collide anymore

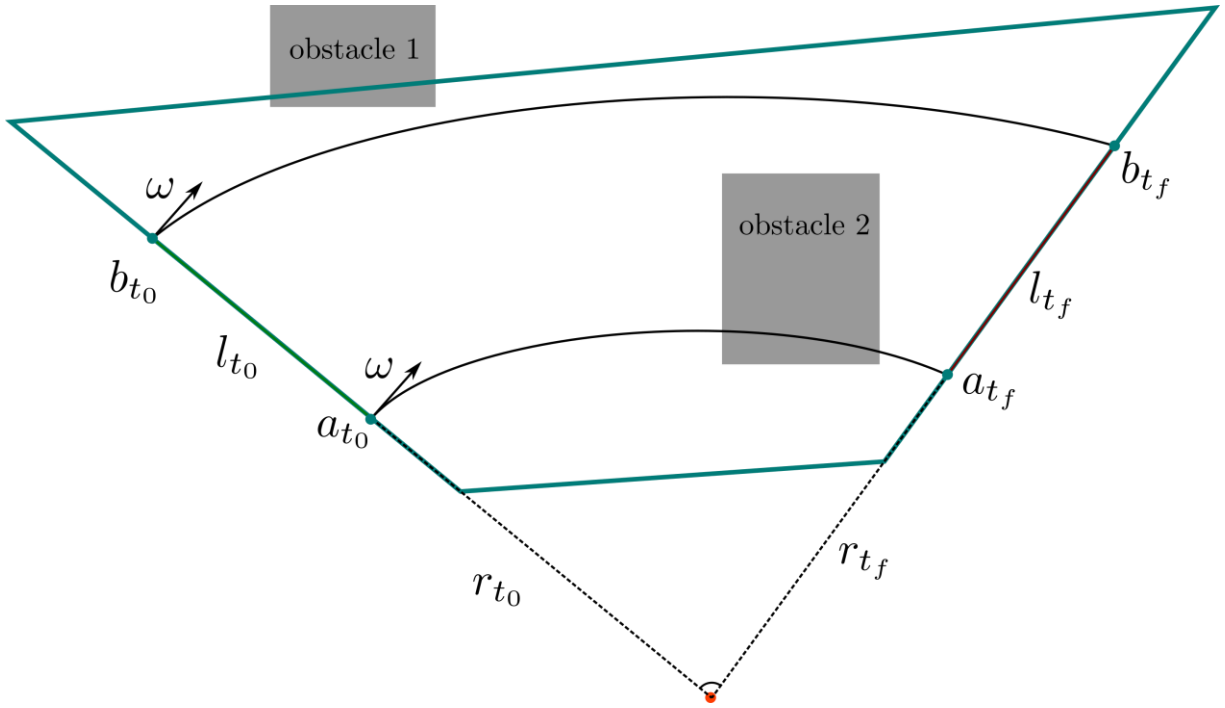


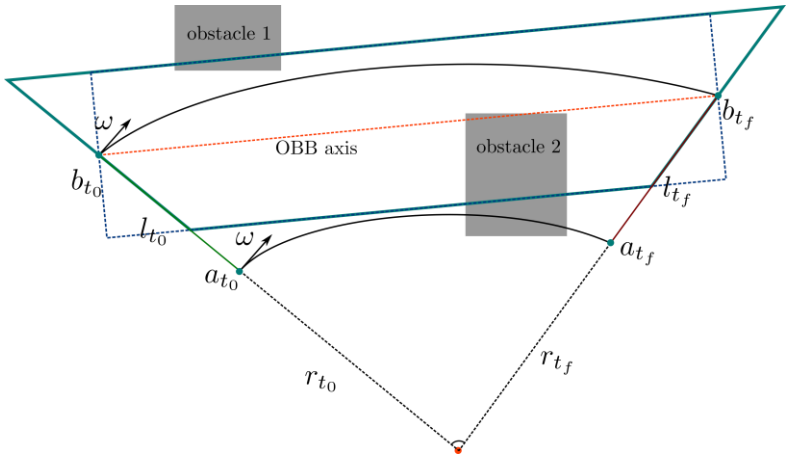
Continuous Time Collision Detection

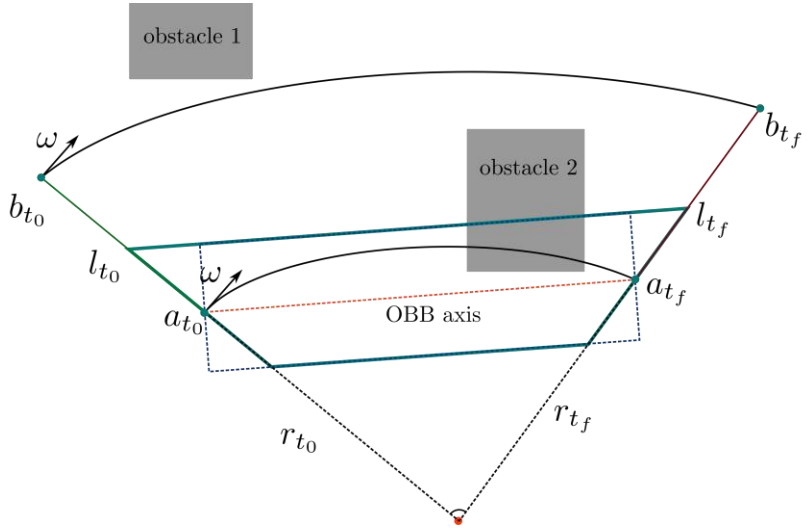
- Repeat this until each interval has at most one collision



Continuous Time Collision Detection

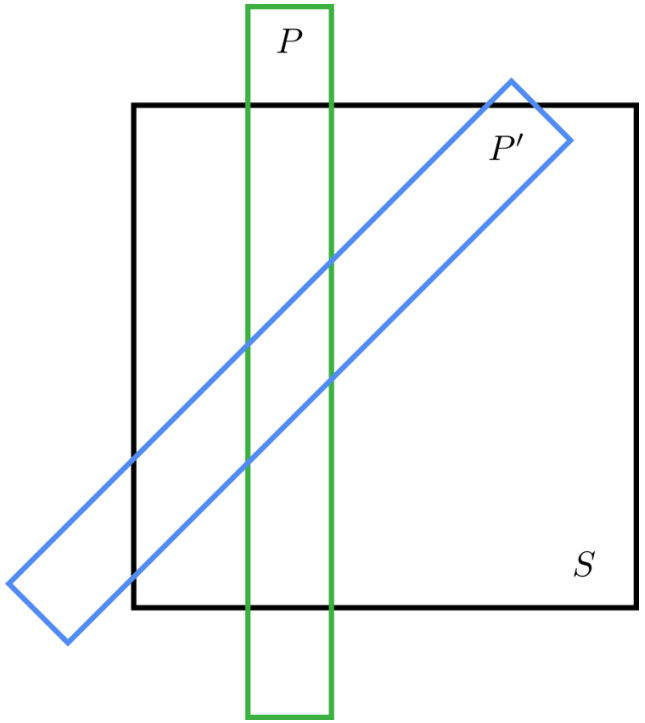


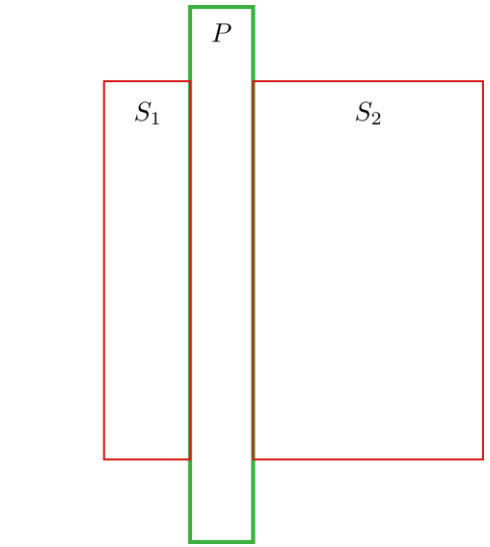


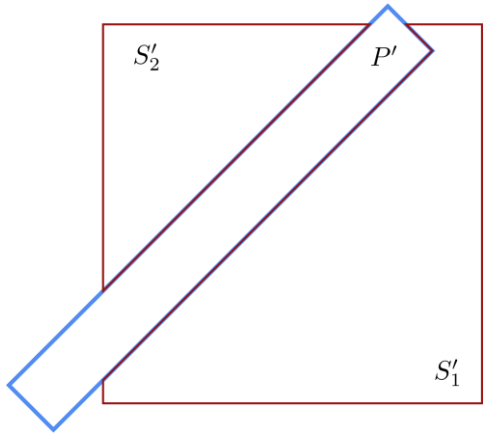


Reza Teshnizi (reza.teshnizi@gmail.com)

Spatiotemporal Connectivity

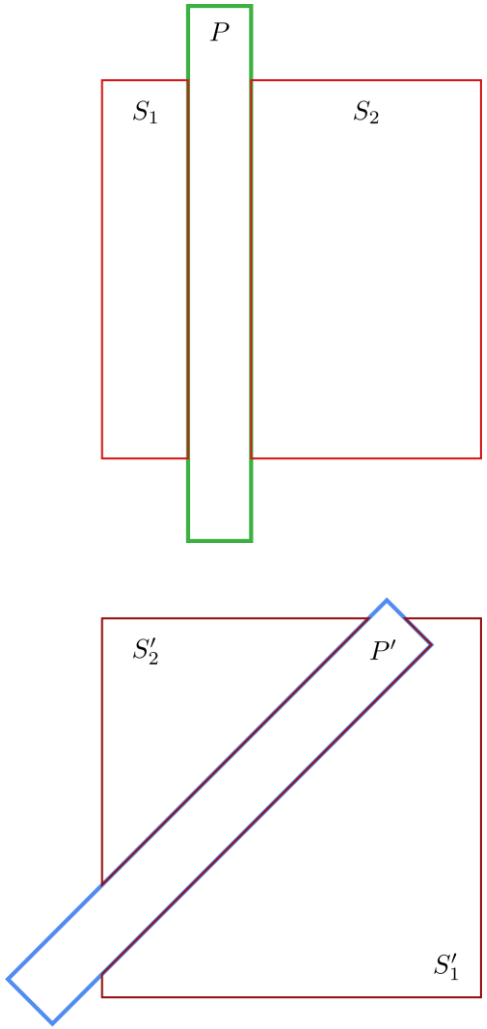


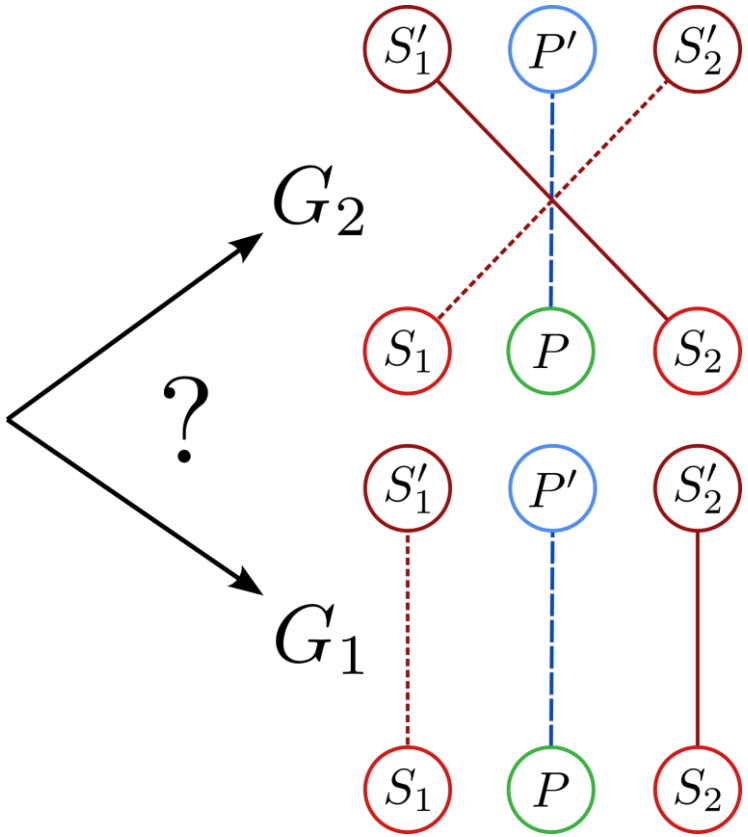




Reza Teshnizi (reza.teshnizi@gmail.com)

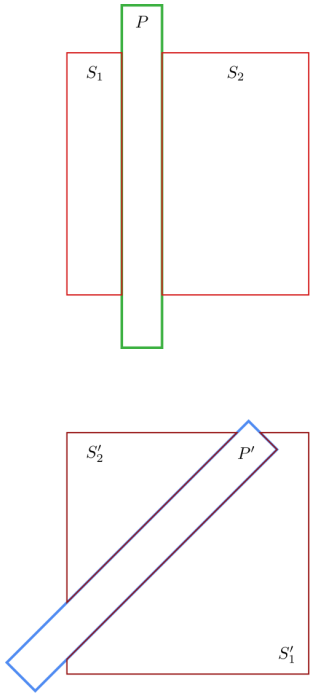
Spatiotemporal Connectivity

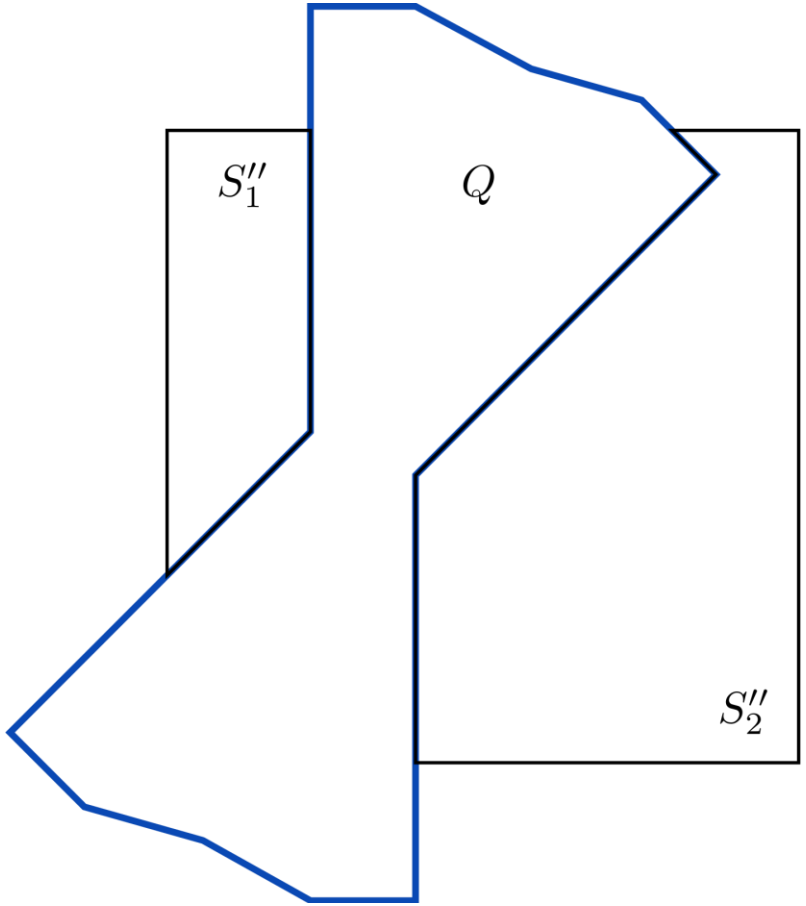


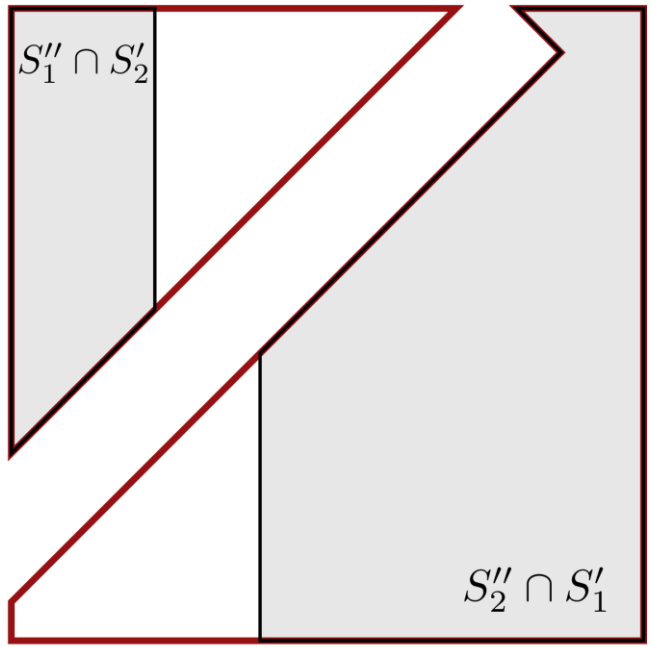


Reza Teshnizi (reza.teshnizi@gmail.com)

Spatiotemporal Connectivity

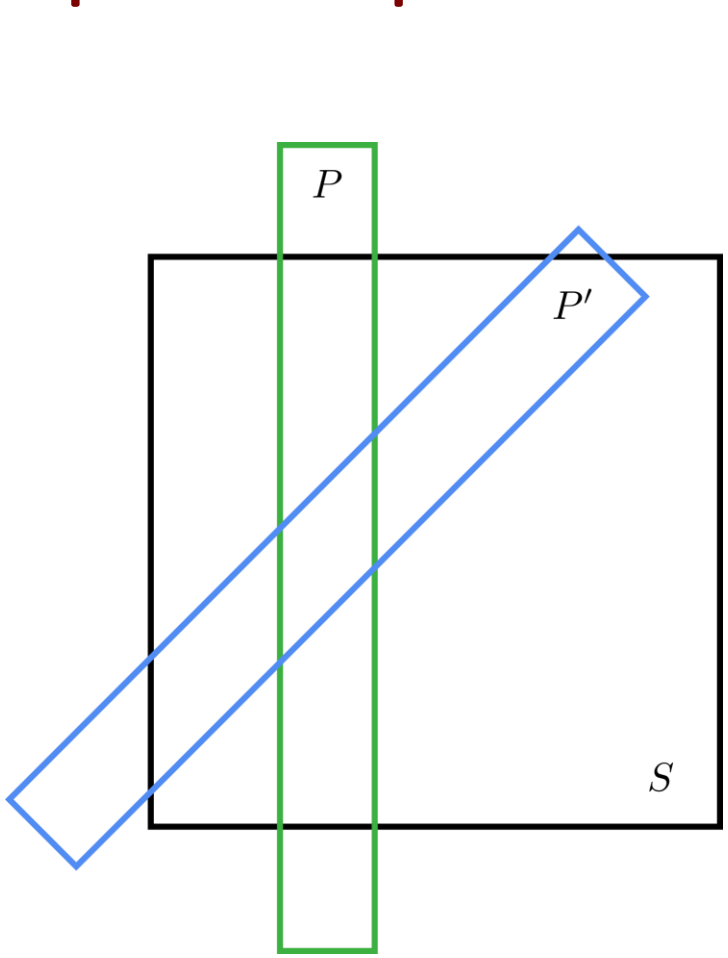


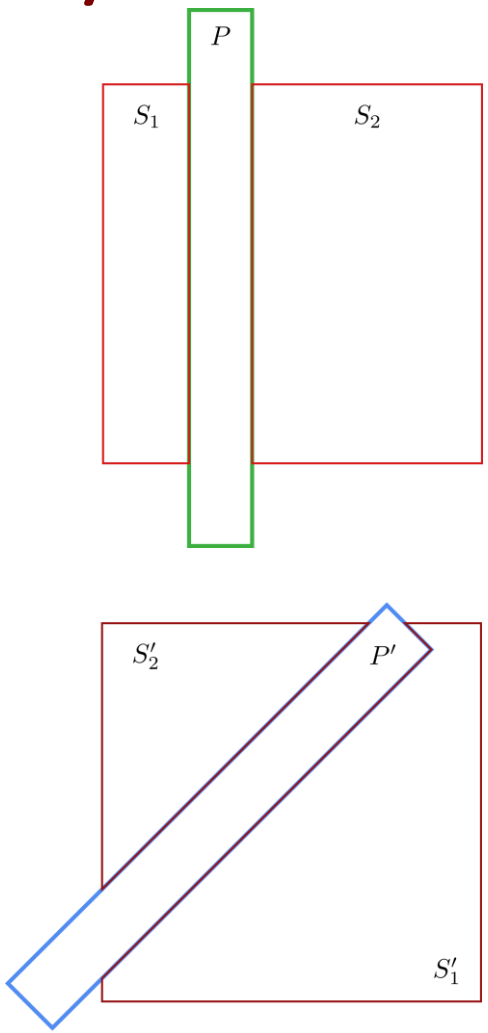


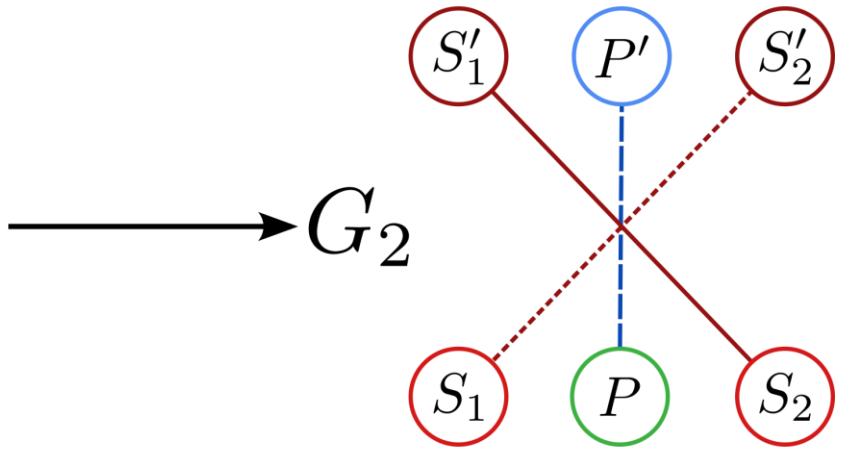


Reza Teshnizi (reza.teshnizi@gmail.com)

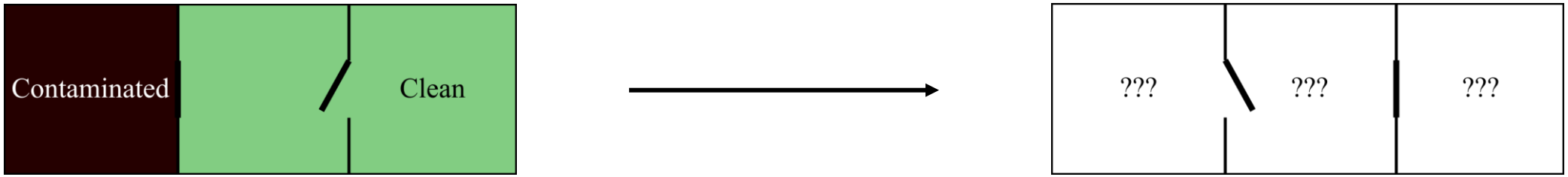
Spatiotemporal Connectivity



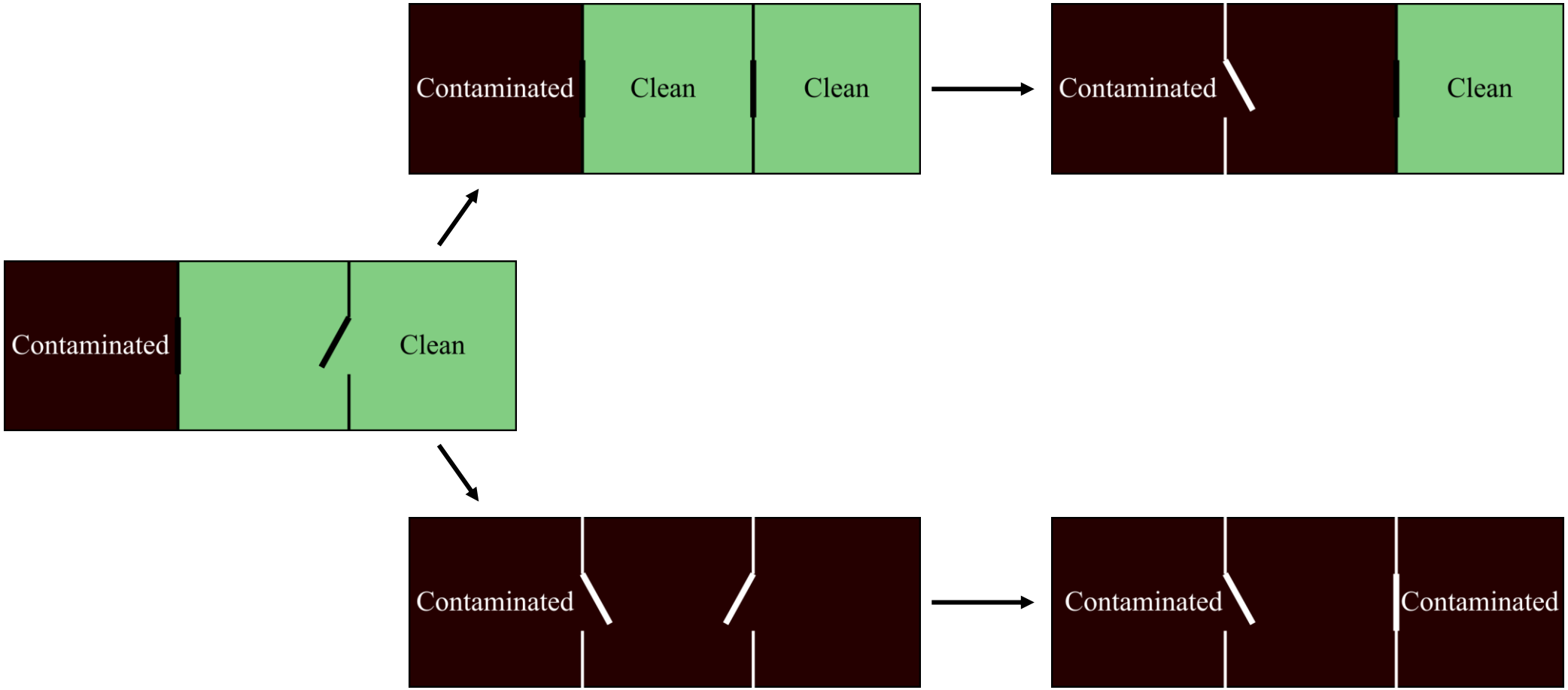




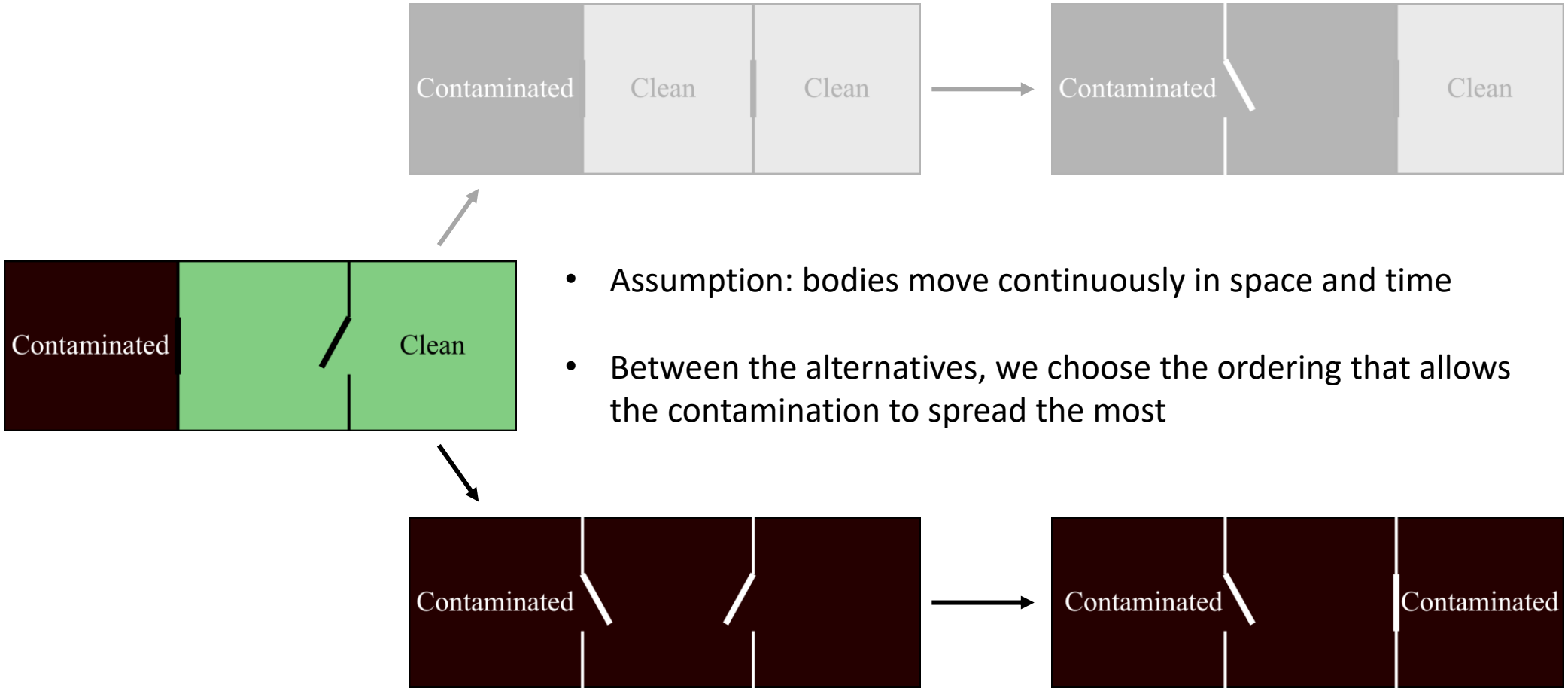
Simultaneous Events



Simultaneous Events



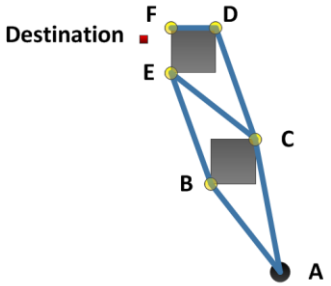
Simultaneous Events

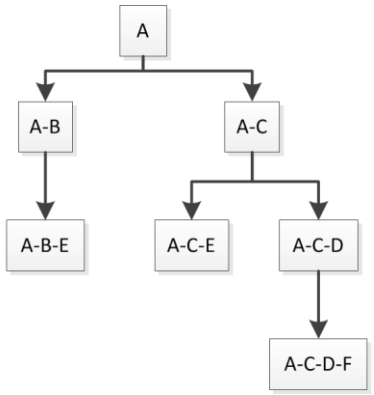


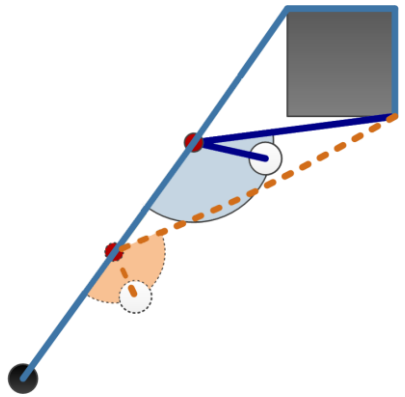
- Assumption: bodies move continuously in space and time
- Between the alternatives, we choose the ordering that allows the contamination to spread the most

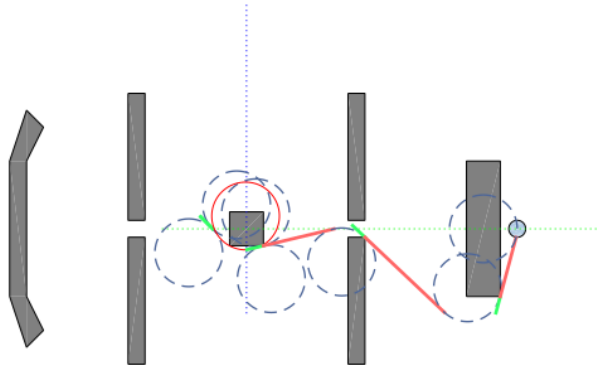
My Research Background

- R. Teshnizi, and D. Shell, 2014, "Computing cell-based decompositions dynamically for planning motions of tethered robots." In *ICRA*.
- R. Teshnizi, and D. Shell, 2016, "Planning motions for a planar robot attached to a stiff tether." In *ICRA*.

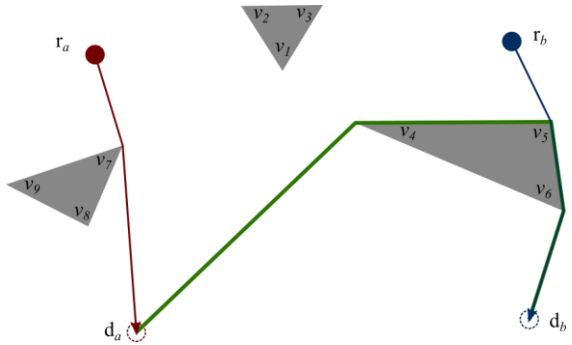


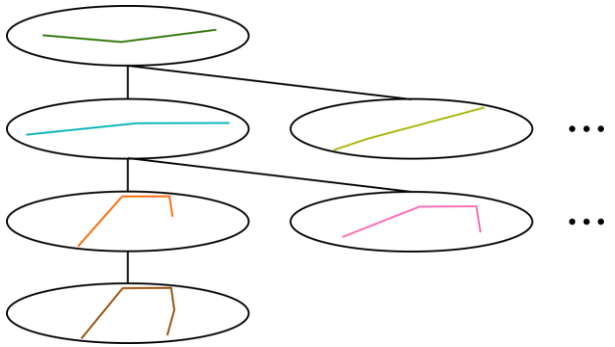


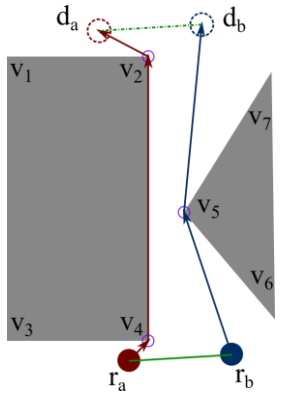




- R. Teshnizi, and D. Shell, 2021, "Motion planning for a pair of tethered robots.“ In Autonomous Robots.







Reza Teshnizi (reza.teshnizi@gmail.com)

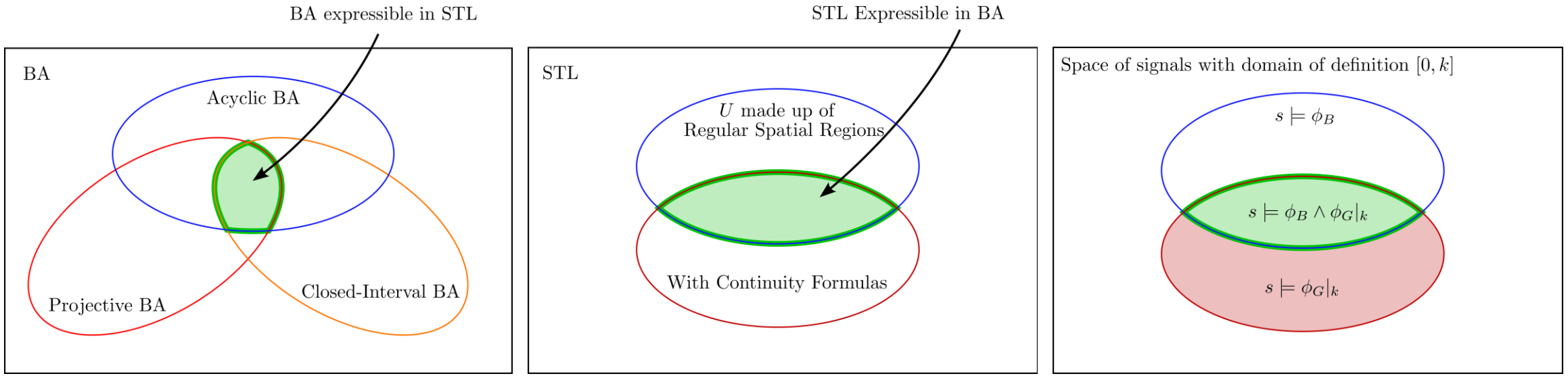
Basis of Comparison – Signal Temporal Logic (STL)

- Monitoring properties of real-valued signals
- s: $\mathbb{R}^{\geq 0} \rightarrow \mathbb{R}$
- $U = \{\mu_1, \ldots, \mu_m\}$ predicate functions of the form $\mu_i: \mathbb{R} \rightarrow \{T, F\}$

- Grammar $\varphi := \mu \mid \neg\varphi \mid \varphi_1 \vee \varphi_2 \mid \varphi_1 \mathcal{U}_{[a,b]} \varphi_2$
- Satisfying a formula:
  - $(s,t) \vDash \mu \quad \Leftrightarrow \mu\big(s(t)\big) = T$
  - $(s,t) \vDash \neg\varphi \quad \Leftrightarrow (s,t) \nvDash \varphi$
  - $(s,t) \vDash \varphi_1 \vee \varphi_2 \quad \Leftrightarrow (s,t) \vDash \varphi_1 \text{ or } (s,t) \vDash \varphi_2$
  - $(s,t) \vDash \varphi_1 \mathcal{U}_{[a,b]} \varphi_2 \Leftrightarrow \exists t_1 \in [t+a, t+b], (s,t_1) \vDash \varphi_2 \text{ and } \forall t_2 \in [t, t_1], (s,t) \vDash \varphi_1$

Reza Teshnizi (reza.teshnizi@gmail.com)

Expressive Power Comparison



$$\phi_B := \bigvee_{r \in R} \left(\bigwedge_{0 \le i < |i|} L(\sigma_i) \right)$$

$$s \models \varphi \iff s \models \varphi \wedge \phi_U$$

$$\phi_G \Big|_k := \bigwedge_{v_\sigma \in V_\Sigma} \left(\bigwedge_{v_{\sigma'} \in (V_\Sigma \setminus N_\sigma)} \phi_{(\sigma,\sigma')} \Big|_k \right)$$