IROS 2009 Tutorial:
Filtering and Planning in I-Spaces
PART 6: POSSIBLE FUTURES

Steven M. LaValle

October 20, 2009

ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

Summary: 4 Main Parts

1. Physical sensors
   Examples of real sensors, how to use them, common characteristics

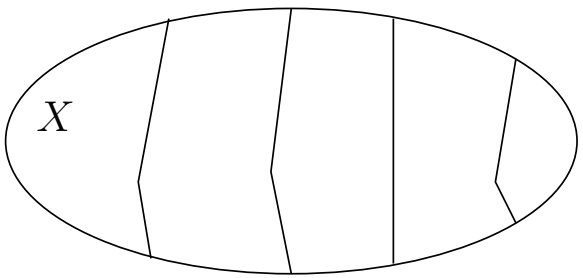
2. Virtual sensors
   Sensor mapping, depth, detection, relational, gap, field sensors, the sensor lattice, complications: disturbances, state-time, history-based

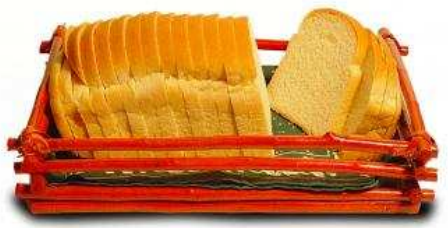
3. Filtering
   Spatial vs. temporal, general triangulation, I-spaces, nondeterministic, probabilistic, combinatorial filters, obstacles and beams, shadow I-spaces, gap navigation trees

4. Planning
   Filtering to planning, overall process, based examples, maze searching, gap navigation trees, landmarks and convex hulls, bug algorithms, sensorless manipulation

Crucial Points

- Virtual sensors: The interface from physical sensor to filters
- $h : X \rightarrow Y$ slices X into equivalence classes
- All basic sensors embed into the sensor lattice
- Spatial filters and triangulation intersects preimages
- Design an I-space for efficient temporal filtering
- All planning problems live in an I-space
- Design virtual sensors, filters, planning problems together around a task.





Research Topics: Virtual Sensors

Find more families of virtual sensors.

Study sensor lattices and how various sensors fit together

Develop better models for disturbed sensors: nondeterministic, probabilistic, ...

Develop more sensor models over state-time and state trajectory space.

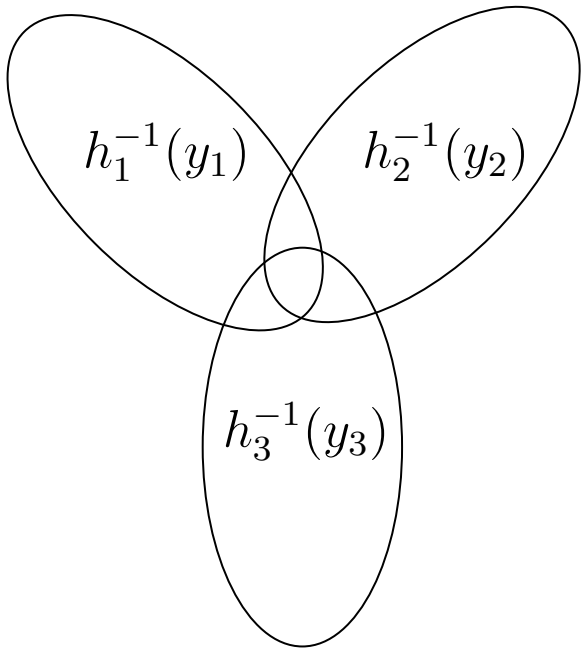
Experimental validation of virtual sensor models, with comparison of physical alternatives.

Research Topics: Spatial Filtering

Develop new families of triangulation methods

Robust alternatives to Bayesian, least-squares?

Distribution, asynchronous, triangulation over state-time space



ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

Research Topics: Temporal Filtering

Introduction of I-space lattices
If $h_1 \succeq h_2$, then does it produce a “better” filter?

Reduced complexity Bayesian filters, derived from combinatorial filters

New combinatorial filters from the virtual sensor models

Generalizing, extending, adapting shadow I-spaces, obstacles and beams, gap navigation trees.

Obstacle-and-beam SLAM

Planning

Move beyond particular, narrow examples of planning in small I-spaces.

Better understand complexity of I-spaces and planning in them.

Sampling-based planning in I-space.

Combinatorial planning in I-spaces.

Planning shadow I-spaces, obstacle-and-beam spaces, gap tree spaces

I-Spaces meet Theoretical Computer Science

Decidability: With certain sensors and filters, what tasks can be accomplished?

Complexity: How much does it cost to achieve a task? How is cost measured?

Can we compare the power of machines, similar to DFAs, PDAs, Turing machines? Sensor lattices can help.

Obstacles and beams lead to regular languages and automata. How can these be minimized? Recall Myhill-Nerode.

Decidability Example

There are two 1D environments embedded in $\mathbb{R}^3$:

1. E_1 is a unit circle in the x_1x_2 plane.
2. E_2 is an infinite helix that projects onto the unit circle:

$$E_2 = \{(\cos\theta, \sin\theta, \theta) \in \mathbb{R}^3 \mid \theta \in \mathbb{R}\}$$

The robot has only a projection sensor: $(x_1, x_2, x_3) \mapsto (x_1, x_2)$.
Its "altitude" is known.

Two actions:

1. Move distance ϵ counterclockwise
2. Move distance ϵ clockwise

Decidability Example

Task 1: A treasure has been placed at an unknown location in E_1 or E_2.

Move the robot until it crosses the treasure.

The problem is decidable.
Solution: Use the "lost cow" strategy over θ.

Task 2: Decide whether a treasure exists in E_1 or E_2.

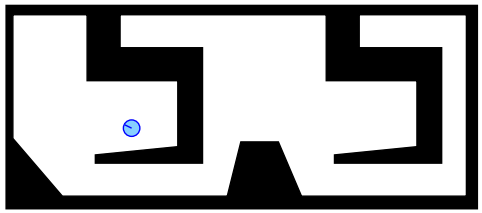
The problem is *not* decidable.

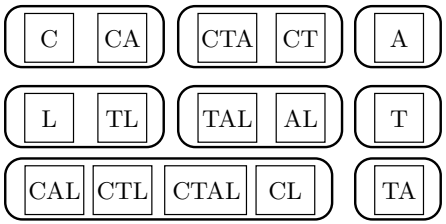
After traversing a full revolution from $\theta = 0$ to $\theta = 2\pi$, here is the resulting I-state:

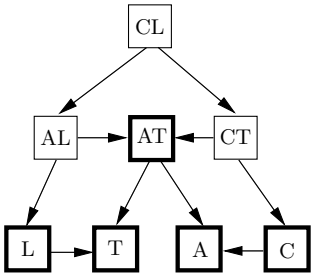
"Either the environment is a circle and there is not treasure, OR the environment is a helix and their may or may not be a treasure"

Comparing Power of Robotic Systems

O'Kane, LaValle RSS 2007







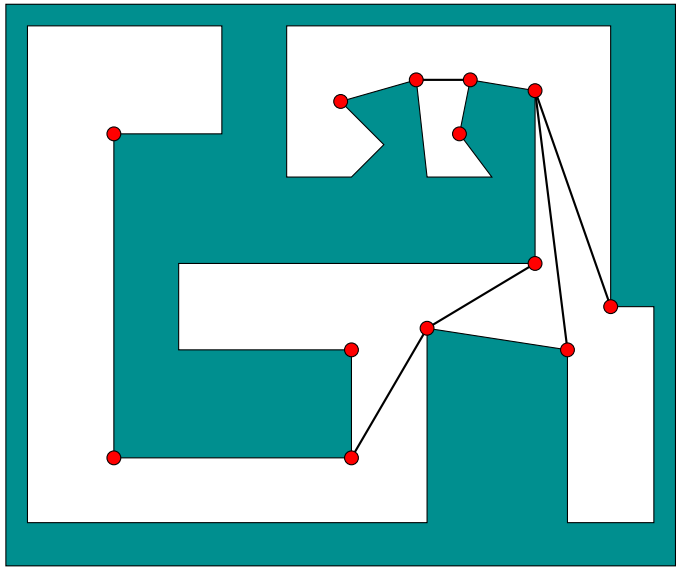
Localization in a given simple polygon ($X \subset SE(2)$)

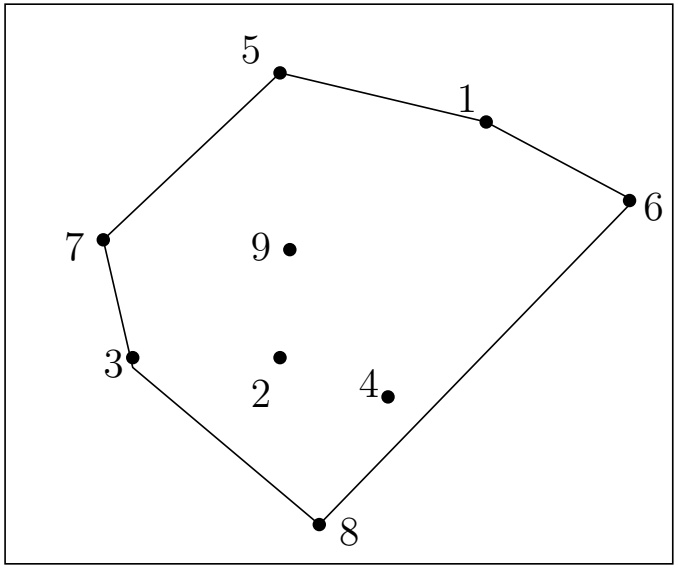
15 robot models are shown to belong to 8 equivalence classes

A domination hierarchy is obtained, in which CL is the most powerful.

I-Spaces meet Computational Geometry

We keep finding known structures from computational geometry:





What else awaits?

Why is this happening?

I-Spaces meet Control Theory

Control theory is “estimation obsessive”:

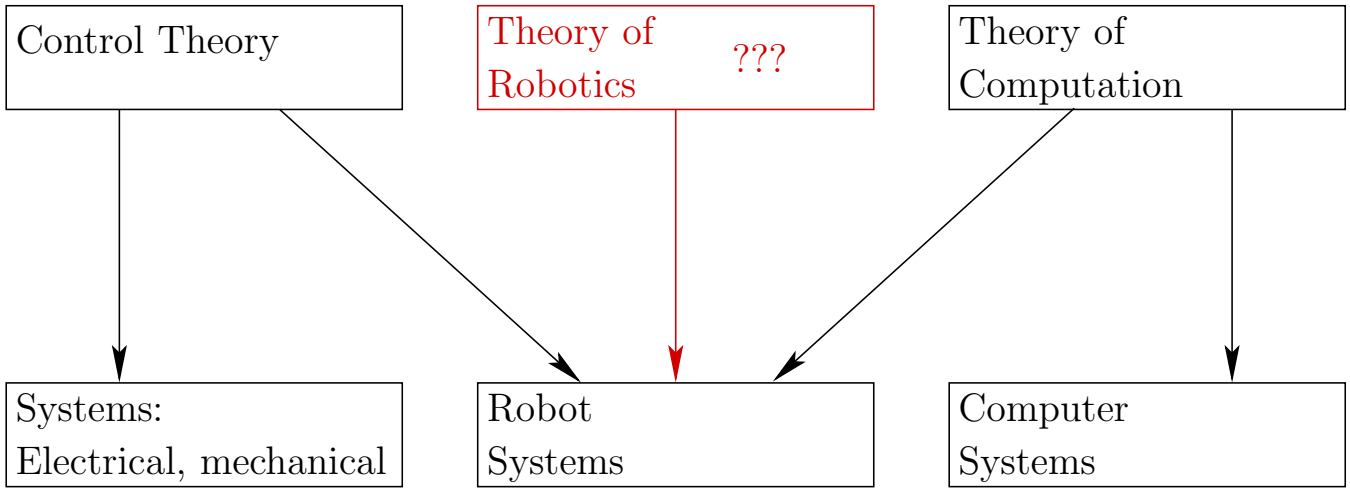
- Make the sensors and filters powerful enough to recover the state x.
- Then pretend that we can use perfect state feedback $\pi : X \rightarrow U$.

“open loop” control really means perfect time feedback (LaValle, Egerstedt, CDC 2007)

Hybrid systems mix computation and system theory, but I-spaces are neglected at present.

Robotics: Where is Our Theory?

We are often the “application” field of others.



What mathematical foundations are special to us?