Temporal Logic Motion Planning for Mobile Robots

Hadas Kress-Gazit, Georgios E. Fainekos, and George J. Pappas

hadaskg,fainekos,pappasg@grasp.upenn.edu

Problem Formulation

Given:

- **A Robot Model:** We consider a planar robot described by the configuration x(t) and controlled by the input u(t)
- **An Environment** : We consider a polygonal environment *P*, which includes "special interest" areas, π_i
- **Specifications:** A desired behavior for the robot expressed as a linear temporal logic (LTL_{-X}) formula φ.

  Such specifications may include:

  Reachability: The formula "$\Diamond\pi$" means "eventually go to π".

  Reachability while avoiding obstacles: The formula "$\neg (o_1 \vee o_2 \vee \ldots \vee o_n) \,\mathcal{U}\, \pi$" means "go to π while avoiding all obstacles o_i".

  Sequencing: "First go to π_1, then to π_2 and finally to π_3" is naturally captured by the formula "$\Diamond (\pi_1 \wedge \Diamond (\pi_2 \wedge \Diamond \pi_3))$".

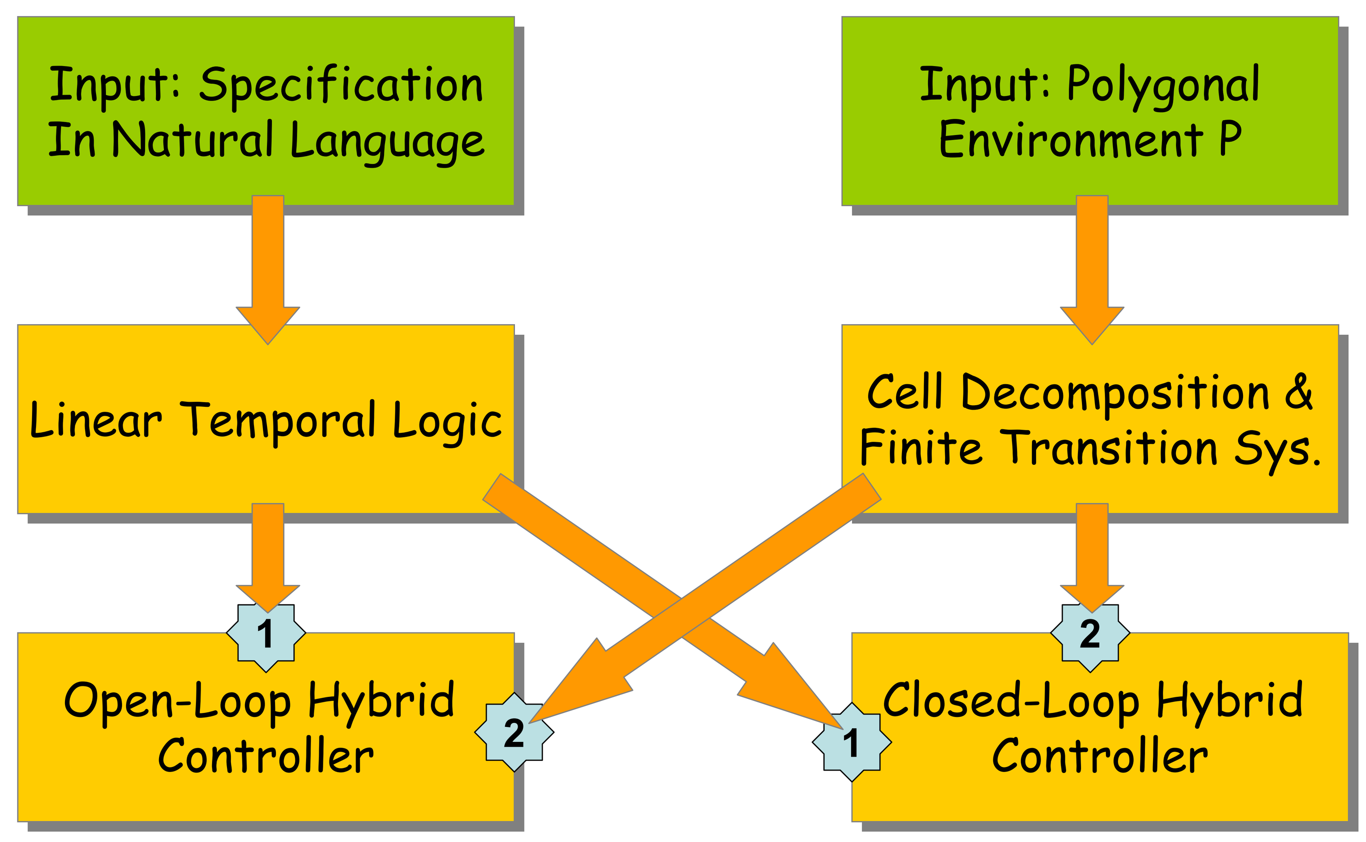
  Coverage: The formula "$\Diamond \pi_1 \wedge \Diamond \pi_2 \wedge \ldots \wedge \Diamond \pi_m$" reads as "visit all regions π_i, in no particular order"

  Recurrence: The formula "$\Box(\Diamond \pi_1 \wedge \Diamond \pi_2 \wedge \ldots \wedge \Diamond \pi_m)$" reads as "visit all regions π_i, in no particular order, infinitely often".
- **An initial condition:** Initial configuration of the robot, x(0)

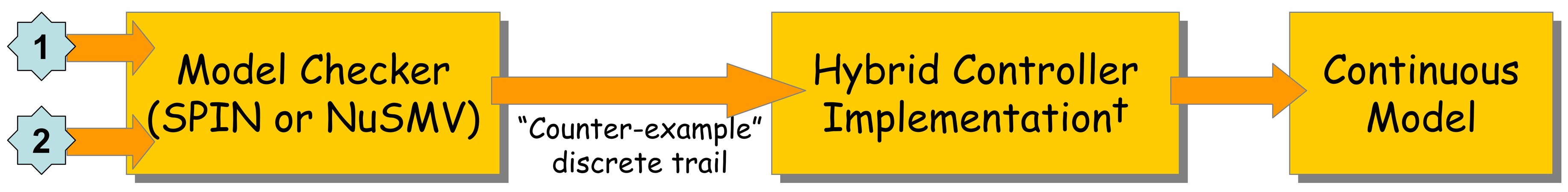
Find:

A **control input** u(t) such that x(t) satisfies φ.

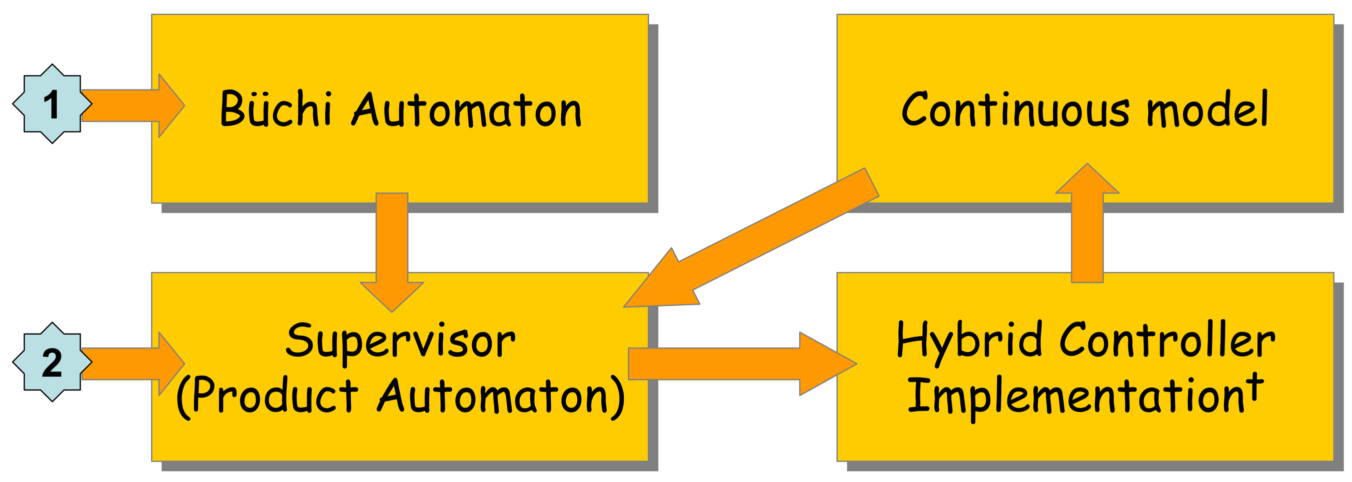
Overview of Temporal Logic Motion Planning



Open-Loop Hybrid Controller♣



Closed-Loop Hybrid Controller♥



♣ G. E. Fainekos, H. Kress-Gazit and G. J. Pappas, Temporal Logic Motion Planning for Mobile Robots, Proceedings of the International Conference on Robotics and Automation, Barcelona, April 2005

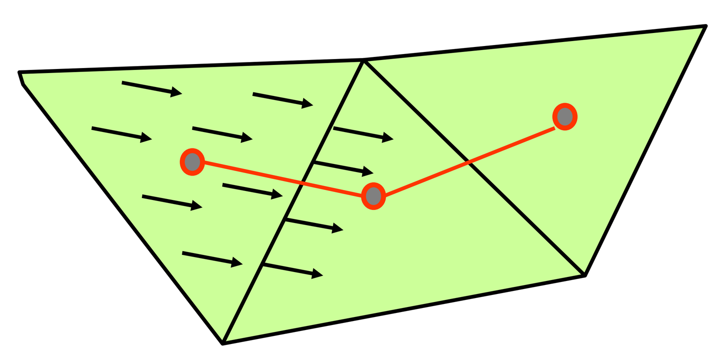
♥ G. E. Fainekos, H. Kress-Gazit and G. J. Pappas, Hybrid Controllers for Path Planning: A Temporal Logic Approach, 44th IEEE Conference on Decision and Control, Seville, Spain, 2005

Hybrid Controller Implementation†‡

A cell decomposition is a **bi-simulation** if the system can move between any two adjacent cells, without going through any other cell, regardless of the initial state.

Affine Controller†

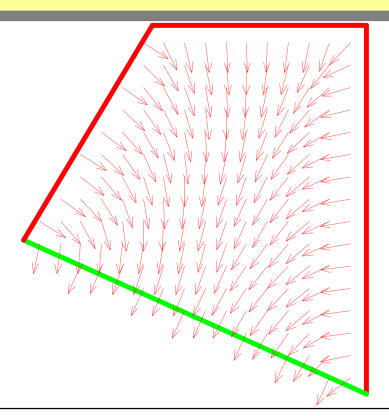
There exist (many) affine vector fields $\dot{x}_P = u_P \quad u_P = Ax + b \in U_P$ on any triangle, satisfying the bisimulation property.



Potential Function Controller‡

Using the gradient of a specific potential function as the controller $\dot{x}_P = u_P = -D\gamma$

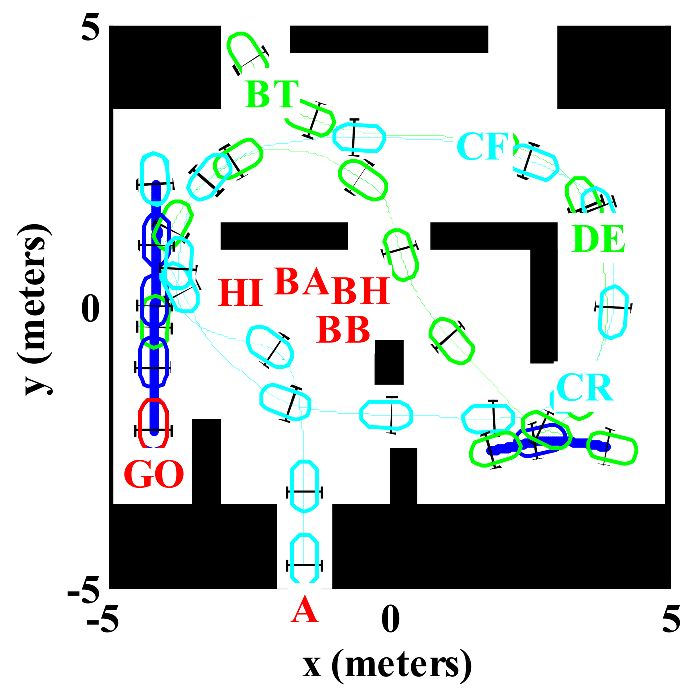
The bisimulation property is satisfied



† C. Belta, V. Isler, and G. J. Pappas, "Discrete abstractions for robot motion planning and control", IEEE Transactions on Robotics, 21(5):864-874, October 2005.

‡ D. C. Conner, A. A. Rizzi and H. Choset, "Composition of Local Potential Functions for Global Robot Control and Navigation", IEEE/RSJ Int'l. Conf. on Intelligent Robots and Systems, Las Vegas, October 2003.

Examples



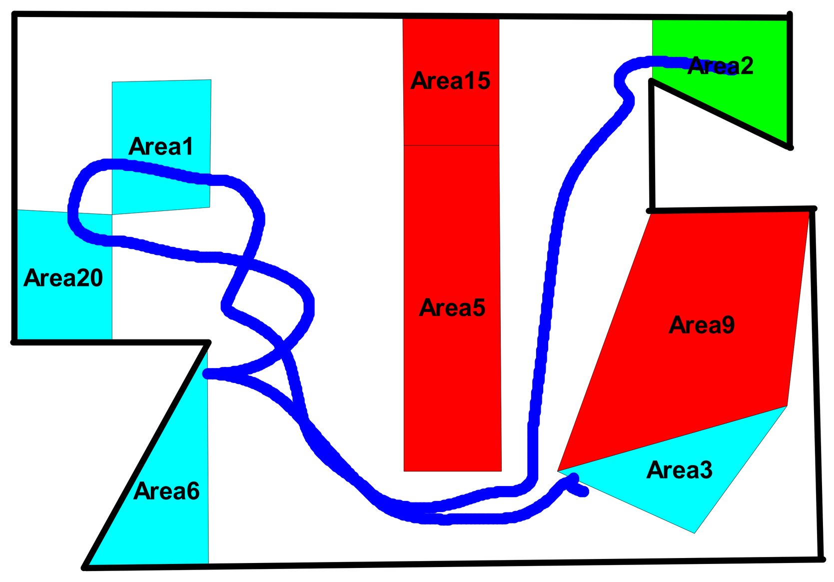
Spec: Start in BT, go to DE, then to GO and then cover CF,CR,A – all this while avoiding BA,BB,BH,HI.

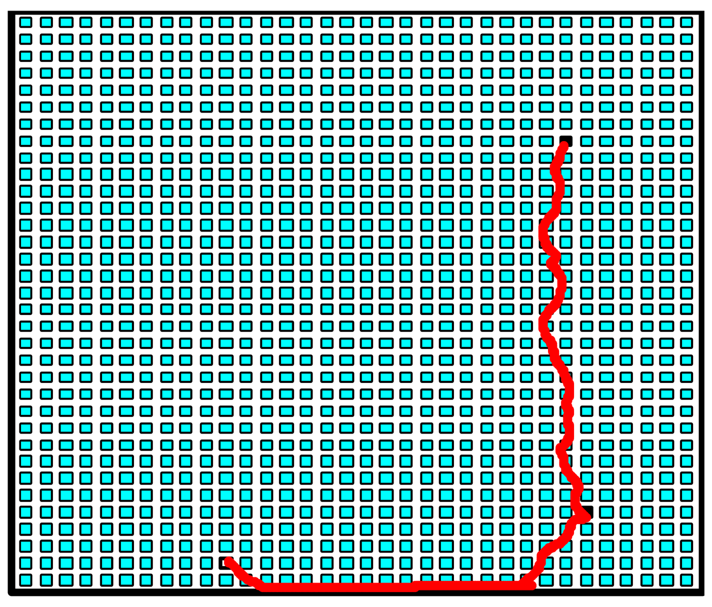
Controller for non-holonomic, convex bodied mobile robot

*Ongoing work with David C. Conner from CMU

Spec: First go to area 1 and then cover areas 3, 6, 20 – all this, while avoiding areas 5,9,15

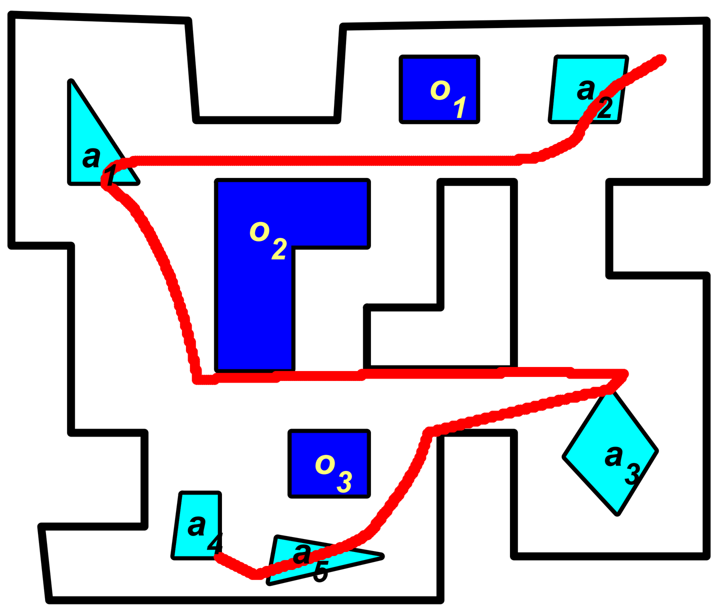
Potential function controller





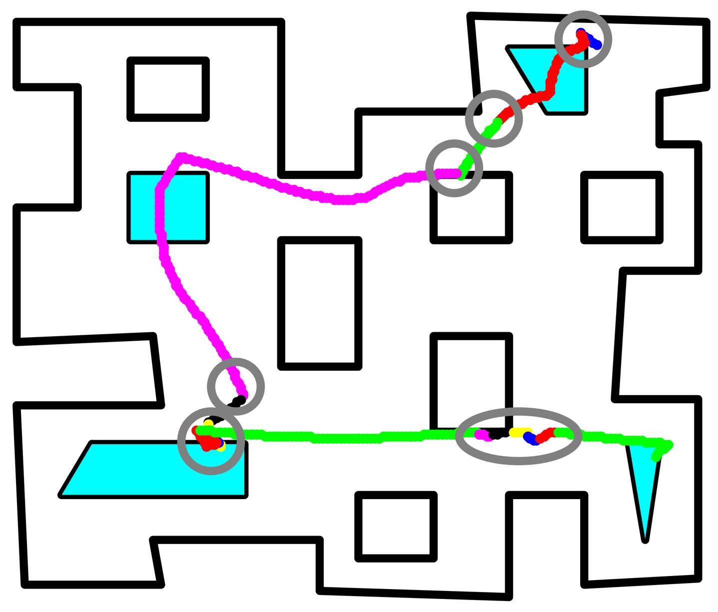
Spec: Go to the two black rooms

Problem Size	Computation time
1156 observables	Triangulation:
9250 triangles	A few seconds
Solution path:	NuSMV:
145 triangles	55 seconds
145 Affine controllers	Matlab:
	90 seconds



Spec: Go to area 2, then to area 1 and then cover areas 3, 4, 5 – all this, while avoiding obstacles o_1, o_2, o_3

Affine controller



Spec: Visit all highlighted areas

Affine controller

Trajectory created by the **closed-loop hybrid** controller. Each divergence from the planned discrete path due to **sampling errors** is circled.