Executing Programs on Continuous States

Prof. Brian Williams
Dr. Andreas Hofmann
March 9th, 2016
Cognitive Robotics (16.412J / 6.834J)

photo courtesy MIT News

After lecture you will know how to…

- How to specify goal-directed motion planning problems as qualitative state plans (QSP).
- How to encode QSP motion planning as a constraint optimization problem.
- How to encode motion policies for under actuated robots using flow tubes.
- How to implement compliant, QSP motion execution by extending dynamic scheduling with flow tube execution.

Assignments

Today:

- T. Léauté and B. Williams, “Coordinating Agile Systems Through the Model-based Execution of Temporal Plans," *Proceedings of the Twentieth National Conference on Artificial Intelligence (AAAI-05),* Pittsburgh, PA, July 2005, pp. 114-120.
- A. Hofmann and B. Williams, “Exploiting Spatial and Temporal Flexibility for Plan Execution of Hybrid, Under-Actuated Systems," *Proceedings of the Twenty first National Conference on Artificial Intelligence (AAAI-06),* Boston, MA, July 2006, pp. 948-955.

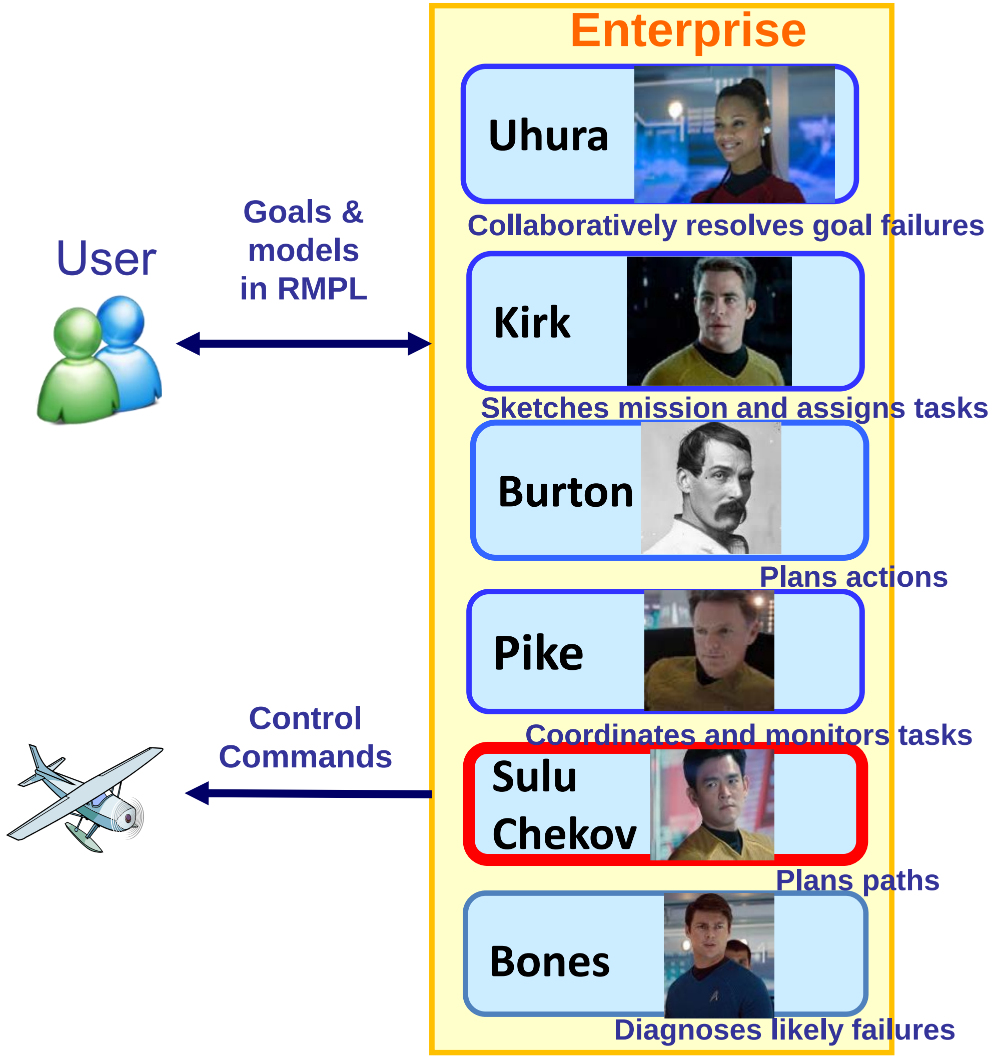
Next:

- P. Yu and B. Williams, “Continuously Relaxing Over-constrained Conditional Temporal Problems through Generalized Conflict Learning and Resolution,” *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI-13)*, Beijing, China, August 2013.

Homework:

- PSet #3 PDDL Modeling, out today, due today, Wed, March 9th.
- Advanced Lecture Sign up: Coming out soon
- Hybrid Estimation problem set: Coming out soon

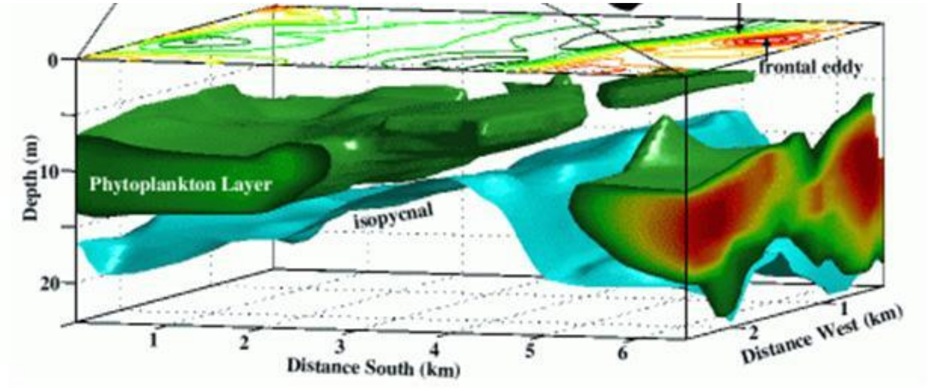
A single "cognitive system" language and executive.



Model-based Execution for Hybrid Systems

MBARI Dorado-class AUV:

- 6000m rated
- 20 hour operation
- Multibeam Sonars
- 3+ knots speed



Challenges:

- Long mission duration
- Limited communication
- GPS unavailable
- Uncertainty
  - tides and currents
  - estimation error

Dynamic Execution of State Plans

00:00 Go to x_1,y_1
00:20 Go to x_2,y_2
00:40 Go to x_3,y_3
…
04:10 Go to x_n,y_n

Command script

Commands

Plant

Leaute & Williams, AAAI 05

Dynamic Execution of State Plans

“Explore mapping region for at least 100s, then explore bloom region for at least 50s, then return to pickup region. Avoid obstacles at all times”

State Plan

Sulu
Model-based Executive

Dynamics
+
Constraints

Observations

Commands

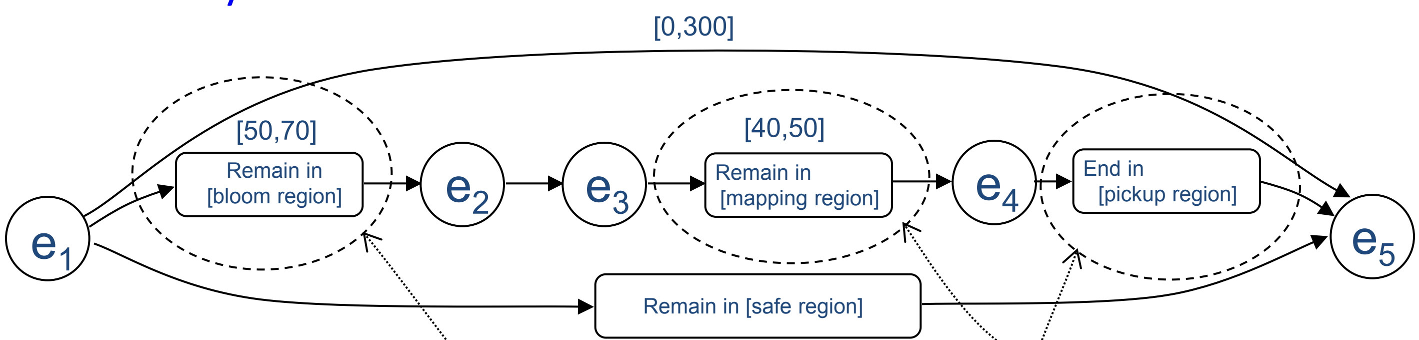
Plant

Optimal

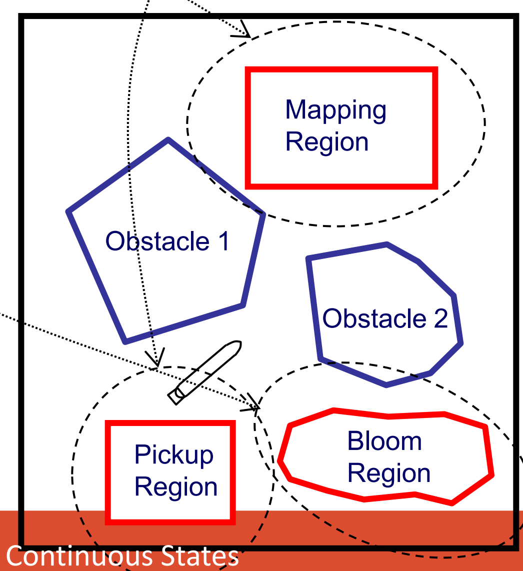
Leaute & Williams, AAAI 05

Sulu: Dynamic Execution of State Plans

A state plan is a model-based program that is unconditional, timed, and hybrid and provides flexibility in state and time.



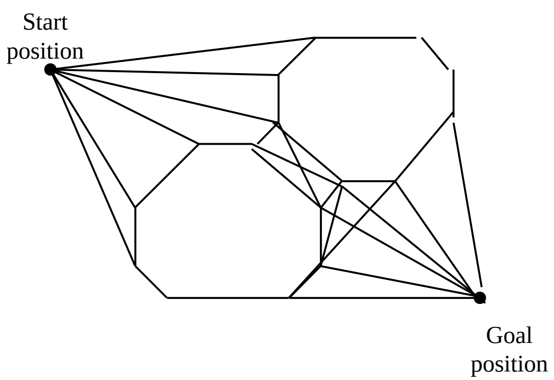
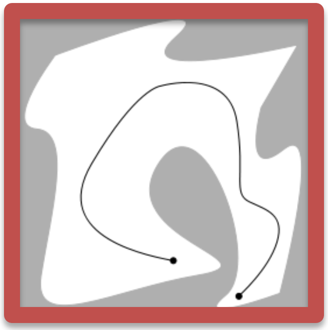
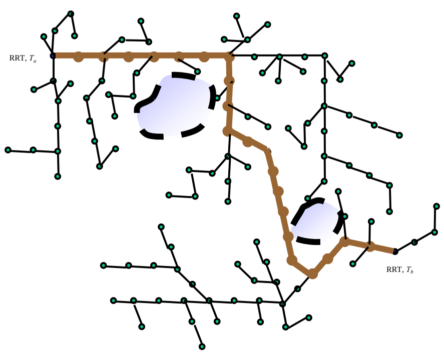
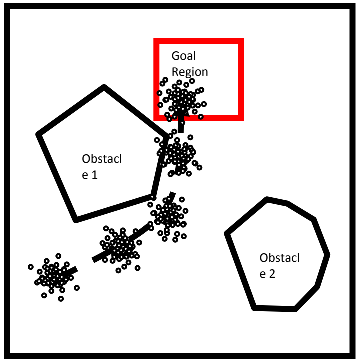
"Explore bloom region for between 50 and 70 seconds. Afterwards, explore mapping region for between 40s and 50s. End in the pickup region. Avoid obstacles at all times. Complete the mission within 300s"



Approach: Frame as Model-Predictive Control using Mixed Logic or Integer / Linear Programming.

[Leaute & Williams, AAAI 05]

Trajectory Planning

	Graph Search	Optimization
Exact		
Sample-based		

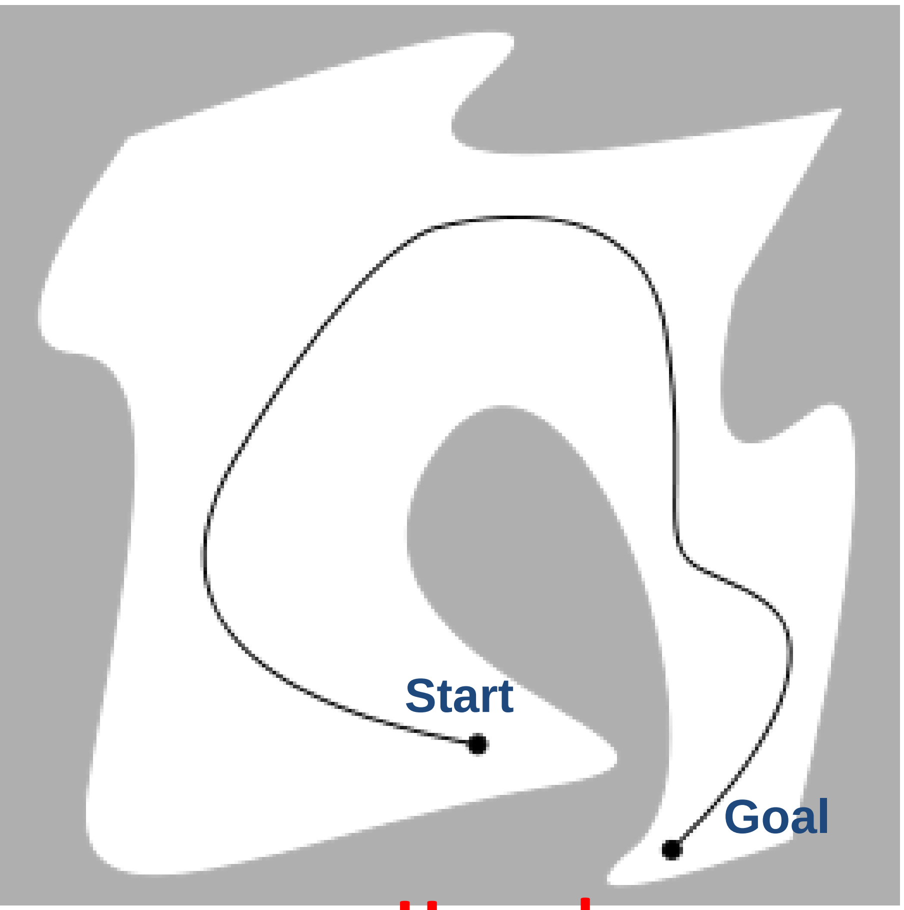
Risk-bounded

Probabilistic

Outline

- Goal-directed Motion Planning (Sulu)
- Compliant, Goal-directed Motion Planning for Under-actuated Robots (Chekov)

Trajectory Optimization



- Plan control trajectory = constraint optimization

$$\min_{p} J(p)$$

$$s.t.$$

$$p \in P$$

p: path
P: Set of feasible paths
J: cost function

How do we encode the constraints for goal-directed trajectory optimization?

Finite Horizon Trajectory Optimization

- Formulate as Linear (LP), Mixed Integer (MILP) or Mixed-Logic (MLLP) Program.

$$\min_{\mathbf{x}_{1:N},\mathbf{u}_{1:N}} J(\mathbf{x}_1 \cdots \mathbf{x}_N, \mathbf{u}_1 \cdots \mathbf{u}_N)$$ **Cost function**

$$s.t.$$

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k \quad (k = 0,1,\cdots N-1)$$ **Dynamics**

$$\mathbf{H}\mathbf{x}_\mathbf{k} \le \mathbf{g} \quad (k = 0,1,\cdots N)$$ **Spatial constraints**

$$\mathbf{x}_0 = \mathbf{x}_{\text{start}}$$ **Initial position and velocity**

$$\mathbf{x}_N = \mathbf{x}_{\text{goal}}$$ **Goal position and velocity**

$$-\mathbf{u}_{\max} \le \mathbf{u}_k \le \mathbf{u}_{\max} \quad (k = 0,1,\cdots N-1)$$ **Actuation limits**

$$\mathbf{x}_k \equiv \begin{pmatrix} x_k & y_k & \dot{x}_k & \dot{y}_k \end{pmatrix}^T, \ \mathbf{u}_k \equiv \begin{pmatrix} F_{x,k} & F_{y,k} \end{pmatrix}^T$$

Encoding Spatial Constraints

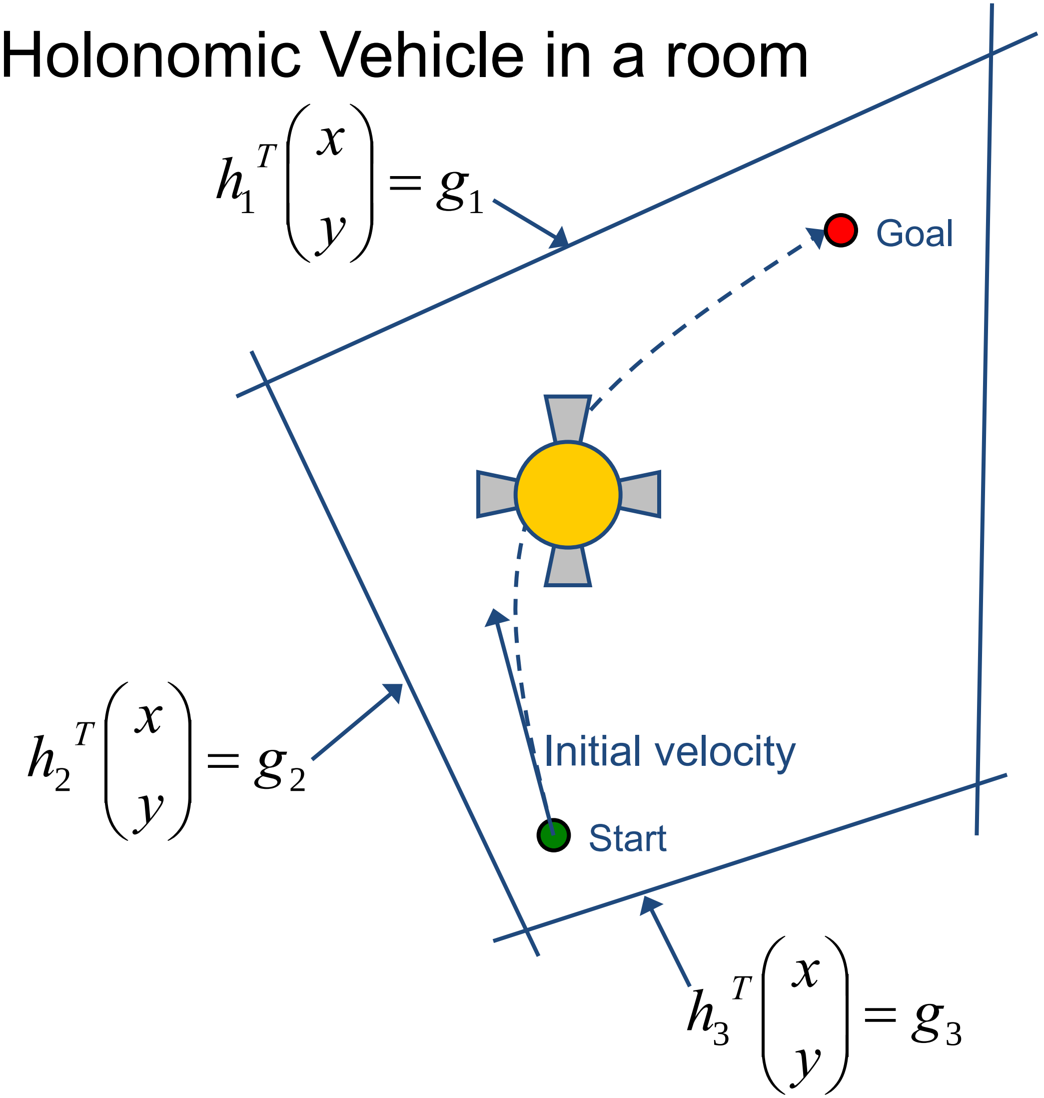
- 2-D Omni-dimensional Holonomic Vehicle in a room

Spatial constraints:
Vehicle must be in the room

$$\bigwedge_{n=1}^{4} h_n^T \begin{pmatrix} x \\ y \end{pmatrix} \le g_n$$

or

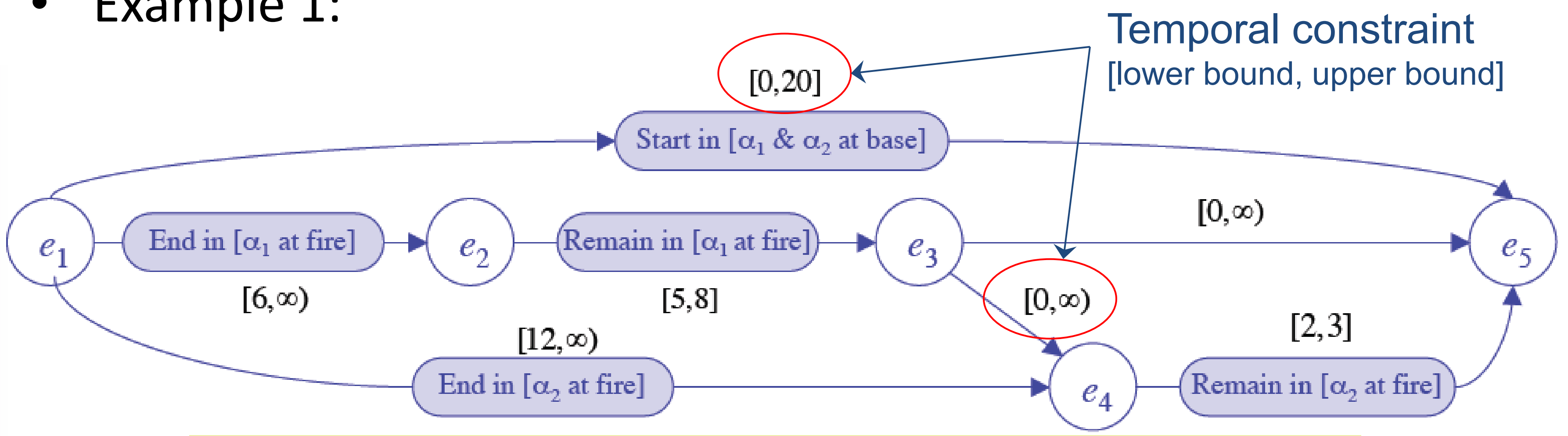
$$\mathbf{Hx} \le \mathbf{g}$$

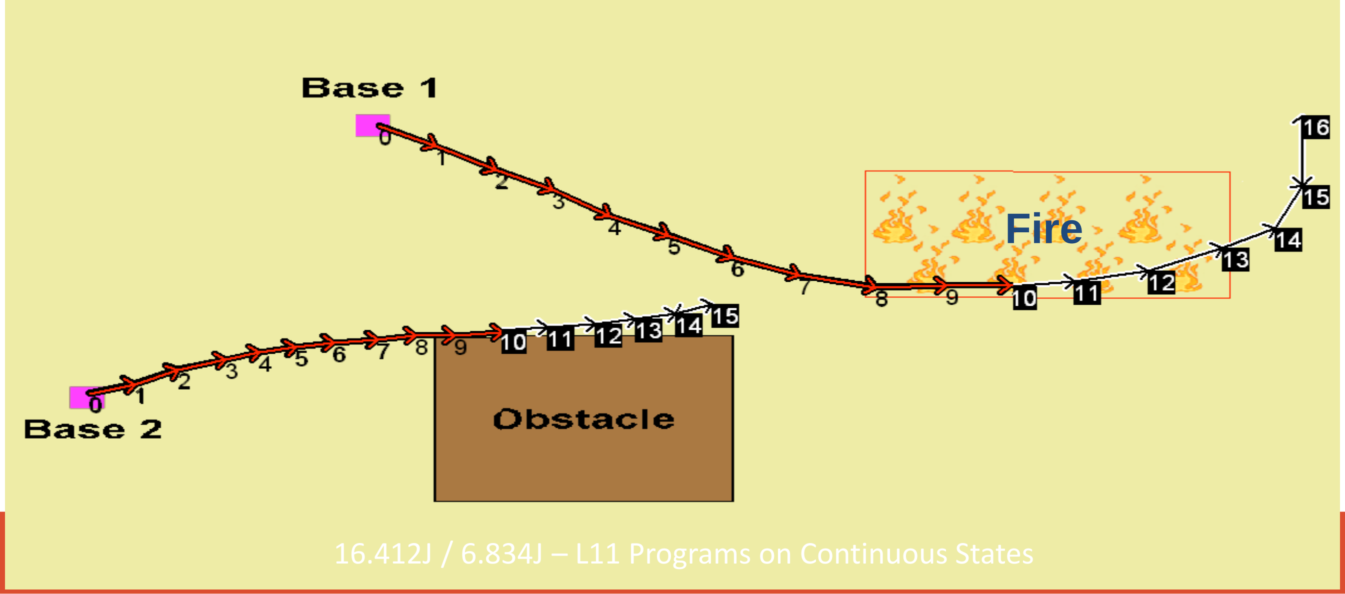


Encoding Qualitative State Plans

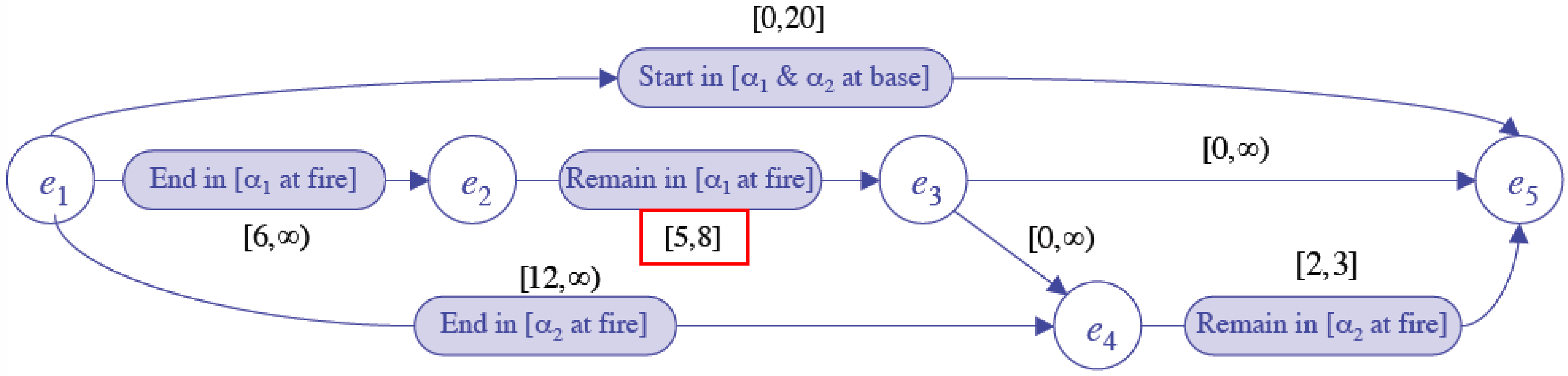
Sulu [Leaute & Williams, AAAI05]

- Example 1:





Encoding Temporal Constraints

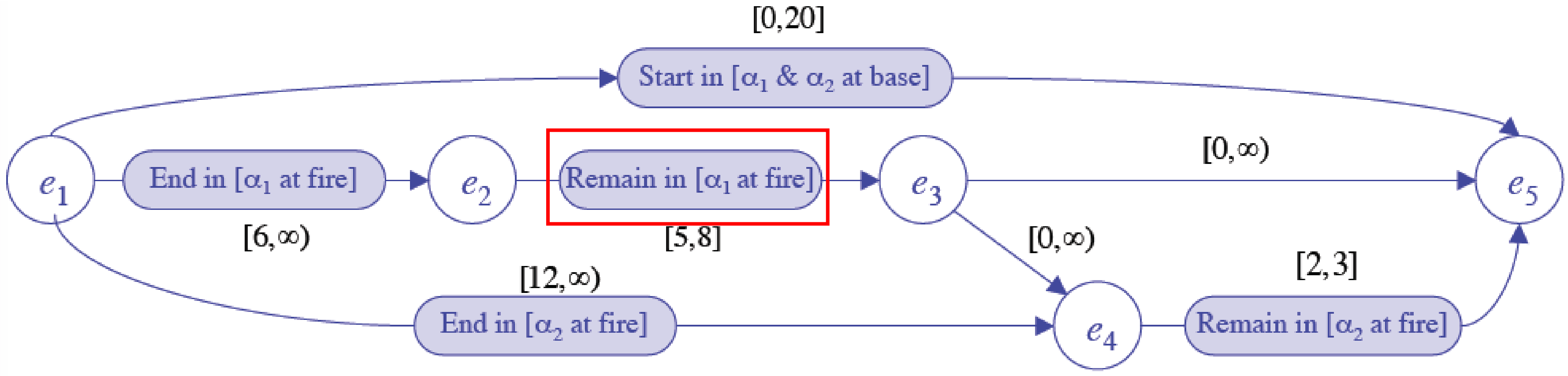


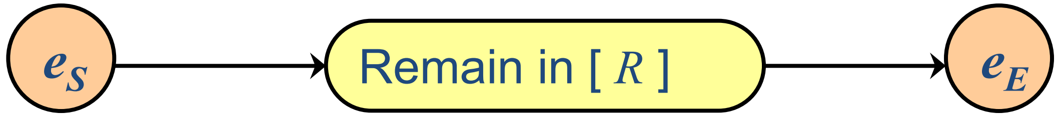
$$LB \leq T(e_E) - T(e_S) \leq UB$$

$T(e_E), T(e_S)$: decision variables

•Thomas Léauté, "*Coordinating Agile Systems through the Model-based Execution of Temporal Plans*, " S. M. Thesis, Massachusetts Institute of Technology, August 2005.
•Thomas Léauté, Brian Williams, "Coordinating Agile Systems Through the Model-based Execution of Temporal Plans," *Proceedings of the Twentieth National Conference on Artificial Intelligence (AAAI-05),* Pittsburgh, PA, July 2005, pp. 114-120.

Encoding “Remain In” Constraints

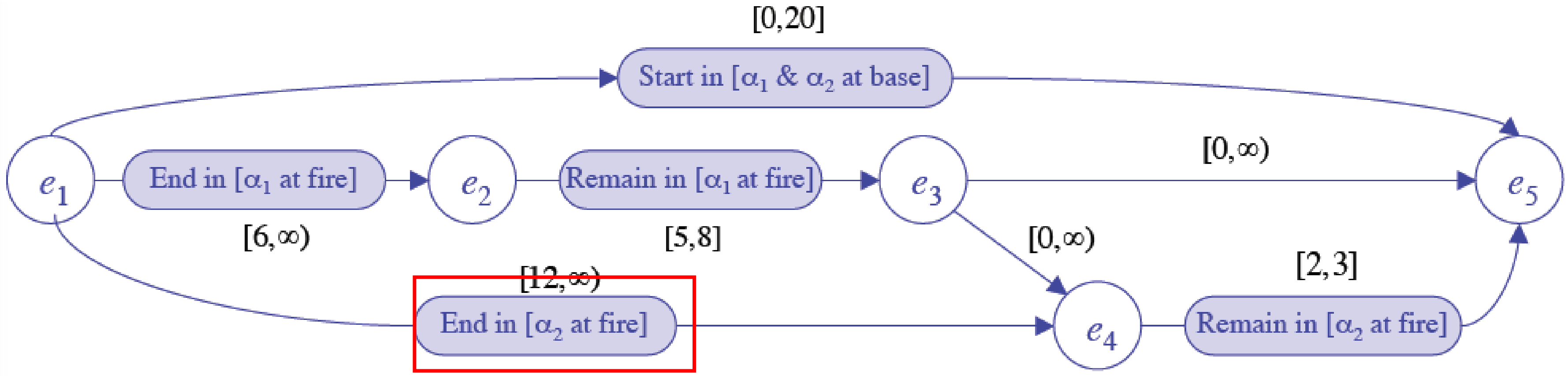




$$\bigwedge_{k=0}^{k=N} \left\{ T(e_S) \le t_k \le T(e_E) \Rightarrow \mathbf{x}_k \in R \right\}$$

•Thomas Léauté, "*Coordinating Agile Systems through the Model-based Execution of Temporal Plans*, " S. M. Thesis, Massachusetts Institute of Technology, August 2005.
•Thomas Léauté, Brian Williams, “Coordinating Agile Systems Through the Model-based Execution of Temporal Plans," *Proceedings of the Twentieth National Conference on Artificial Intelligence (AAAI-05),* Pittsburgh, PA, July 2005, pp. 114-120.

Encoding "End In" Constraints

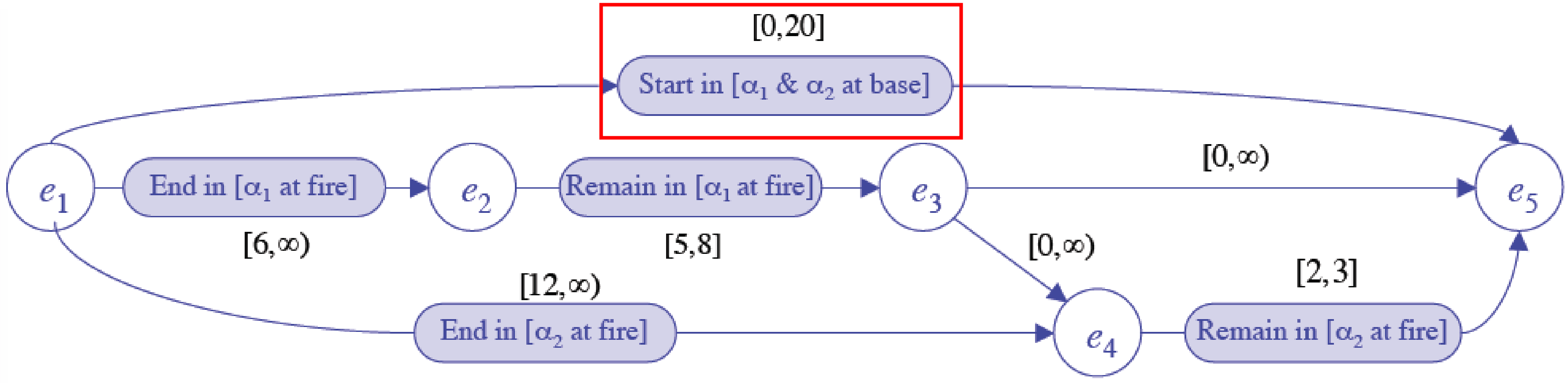


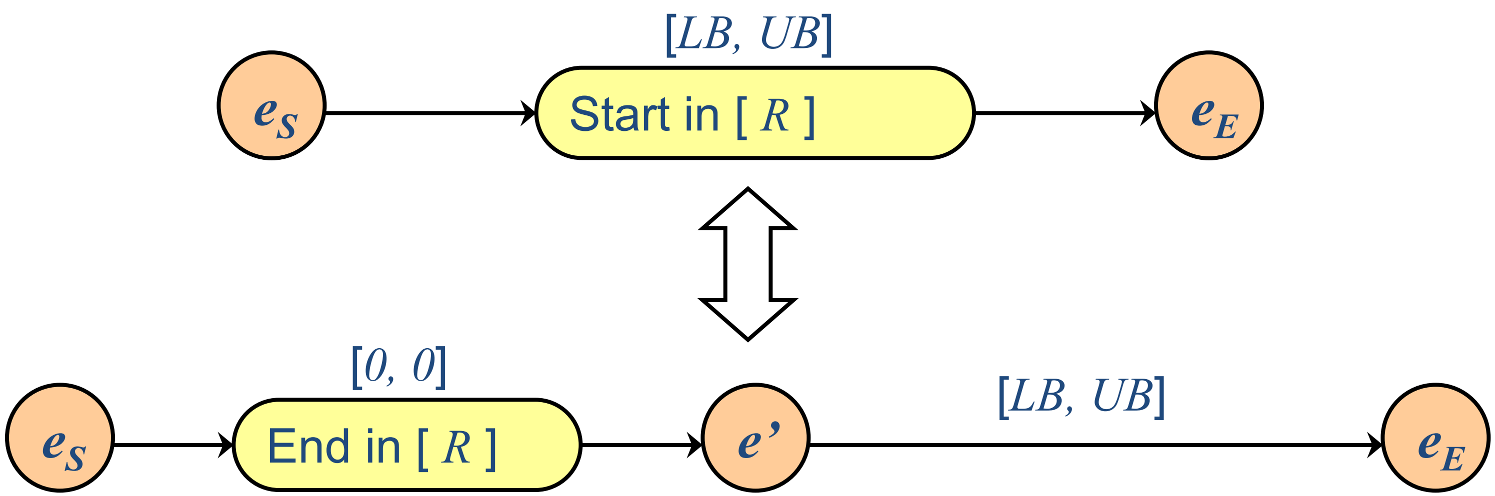
Variable

$$\bigvee_{k=0}^{k=N} \left\{ t_k - \frac{\Delta t}{2} \le T(e_E) \le t_k + \frac{\Delta t}{2} \wedge \mathbf{x}_k \in R \right\}$$

Constant

Encoding "Start In" Constraints

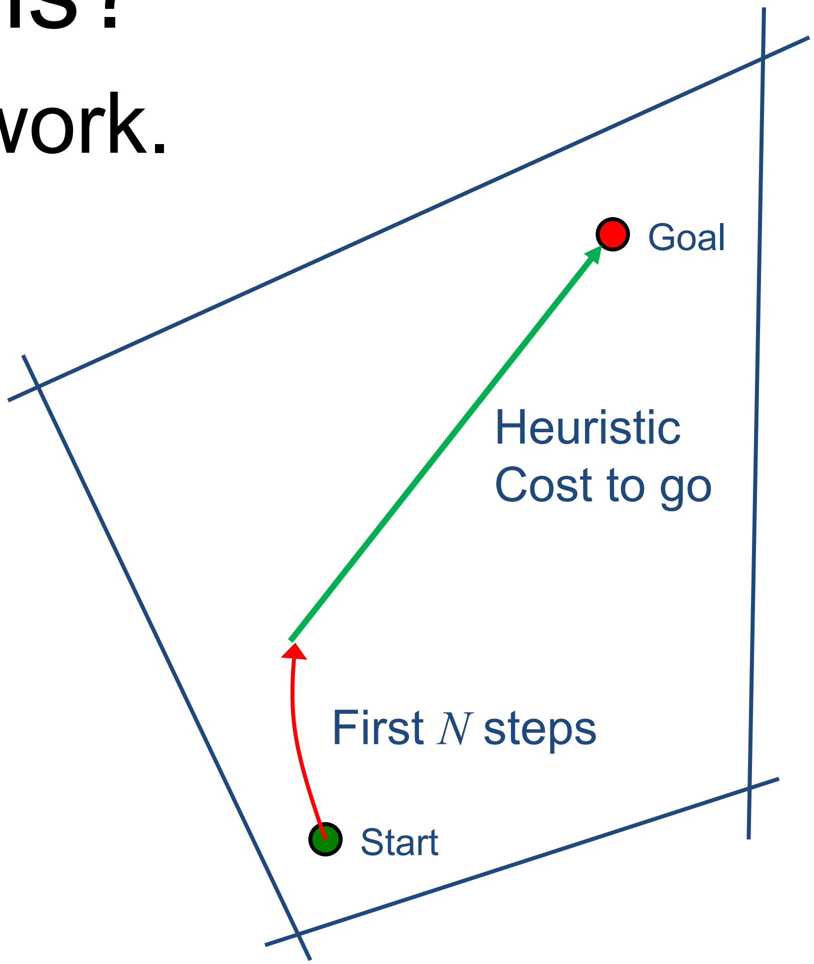




•Thomas Léauté, "*Coordinating Agile Systems through the Model-based Execution of Temporal Plans*, " S. M. Thesis, Massachusetts Institute of Technology, August 2005.
•Thomas Léauté, Brian Williams, "Coordinating Agile Systems Through the Model-based Execution of Temporal Plans," *Proceedings of the Twentieth National Conference on Artificial Intelligence (AAAI-05),* Pittsburgh, PA, July 2005, pp. 114-120.

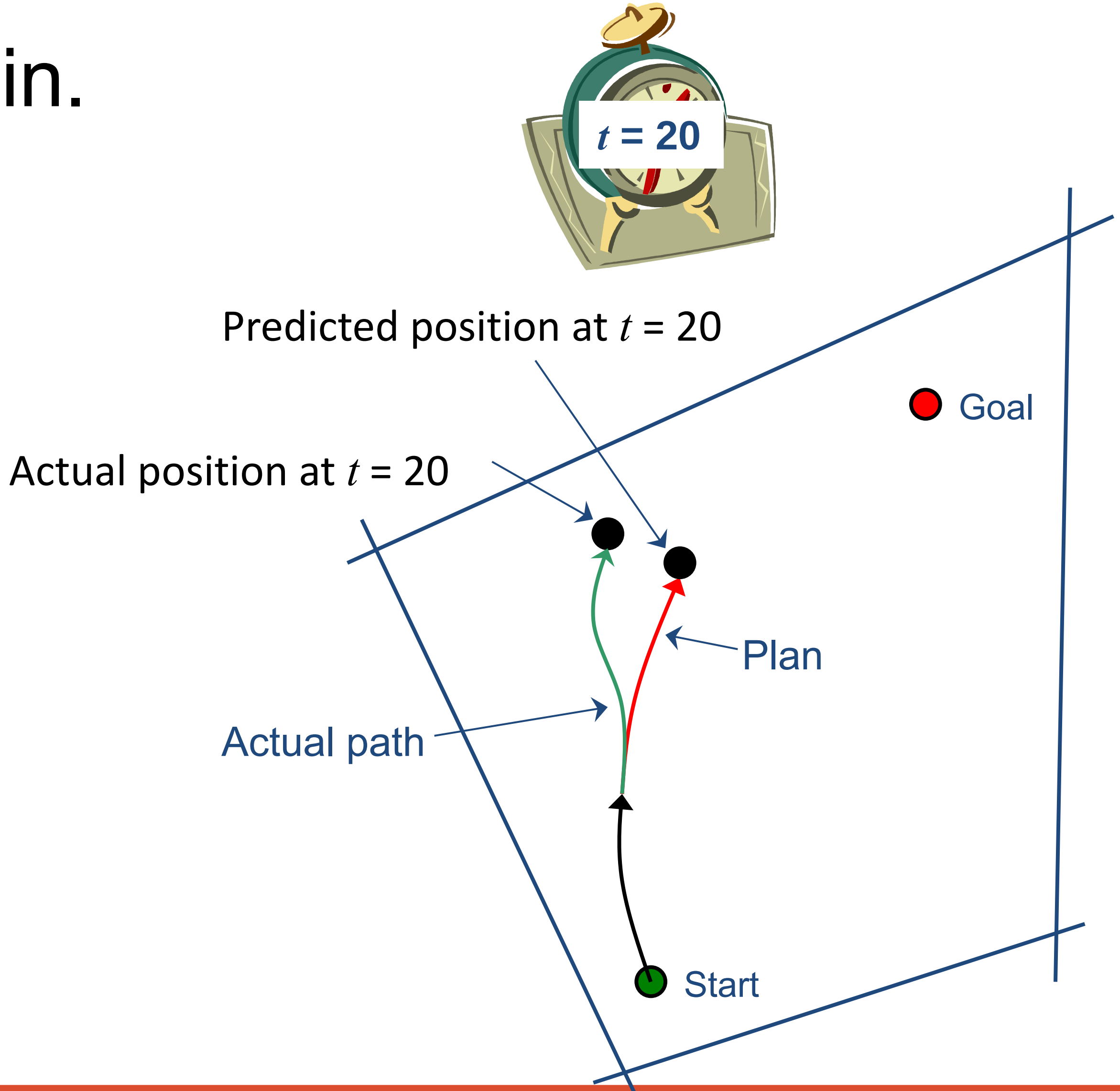
How do we plan over long horizons?

- Patchwork.



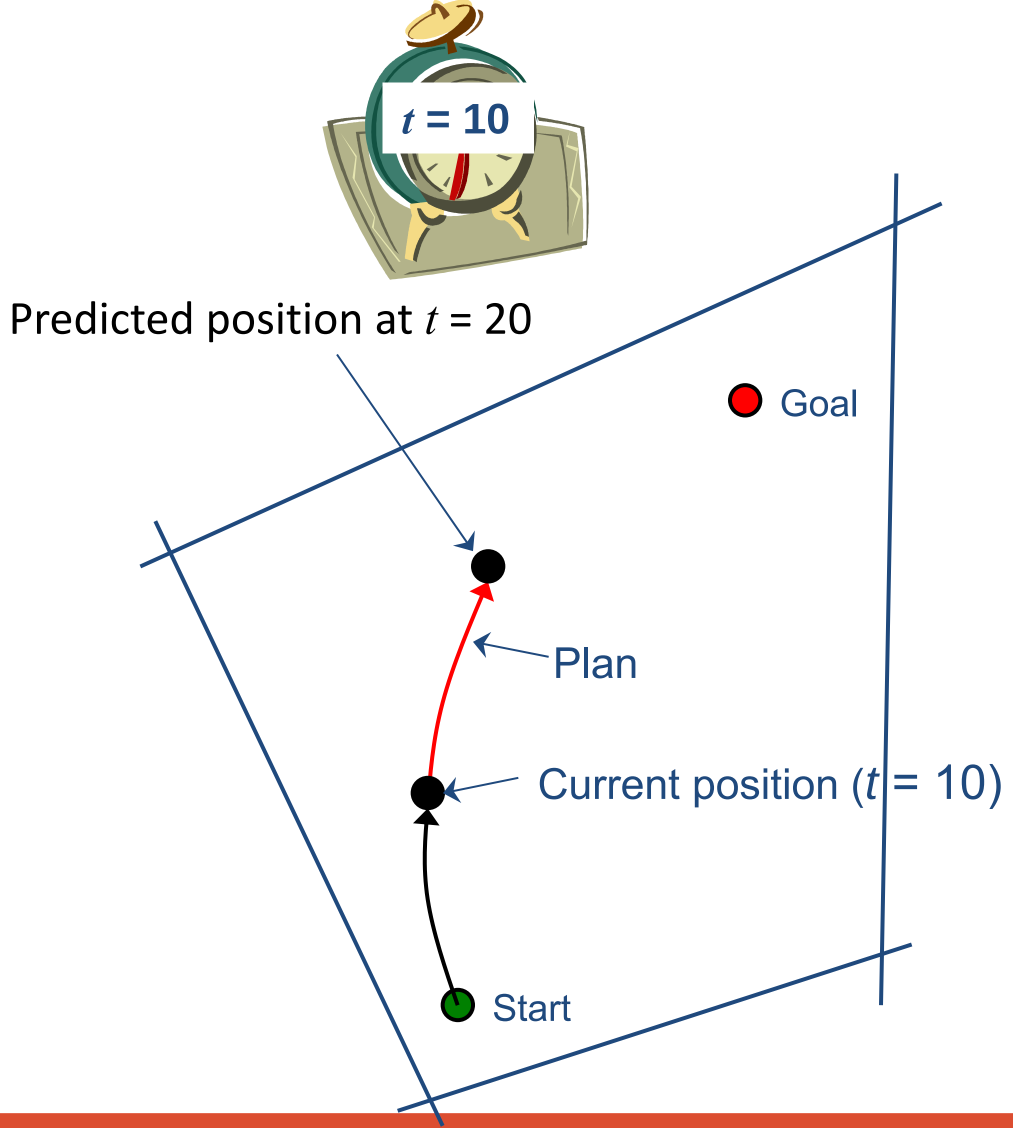
How do we handle disturbances?

- World uncertain.



Execution Horizon < Planning Horizon

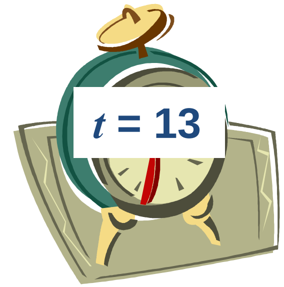
- 3 seconds later….

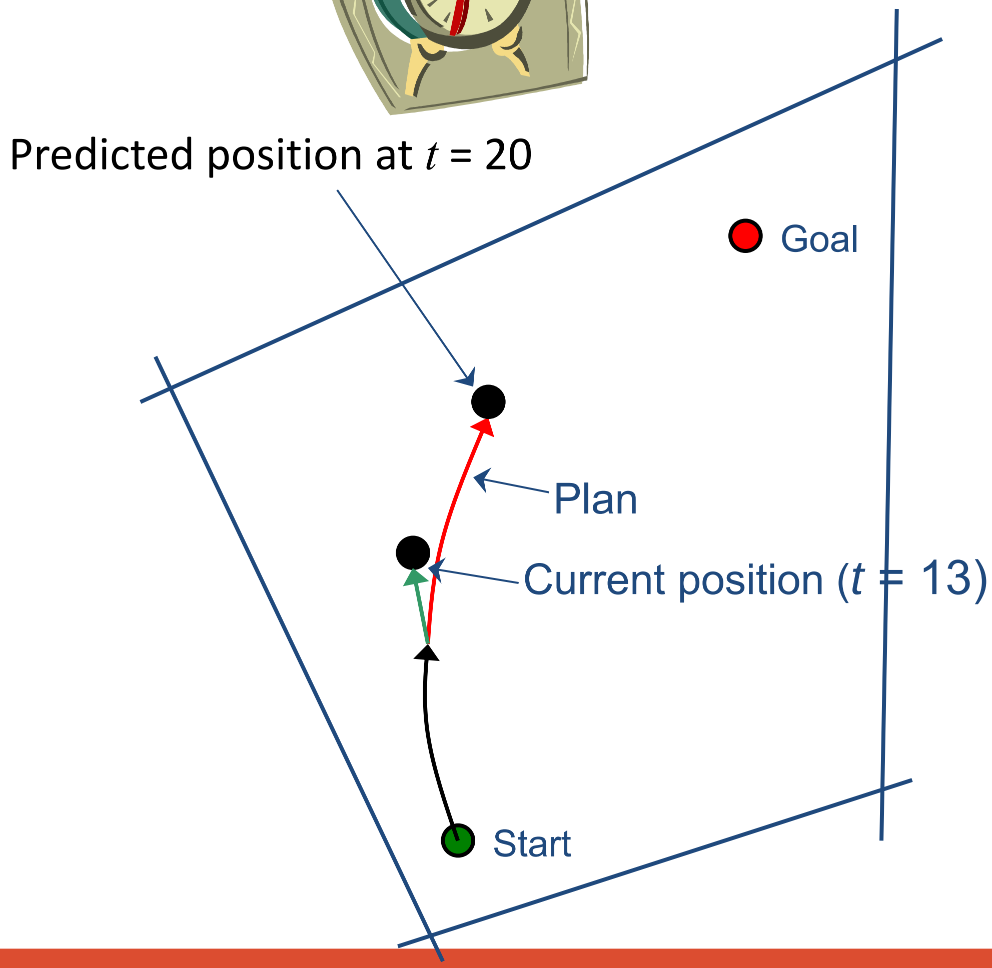


Execution Horizon < Planning Horizon

- 3 seconds later….

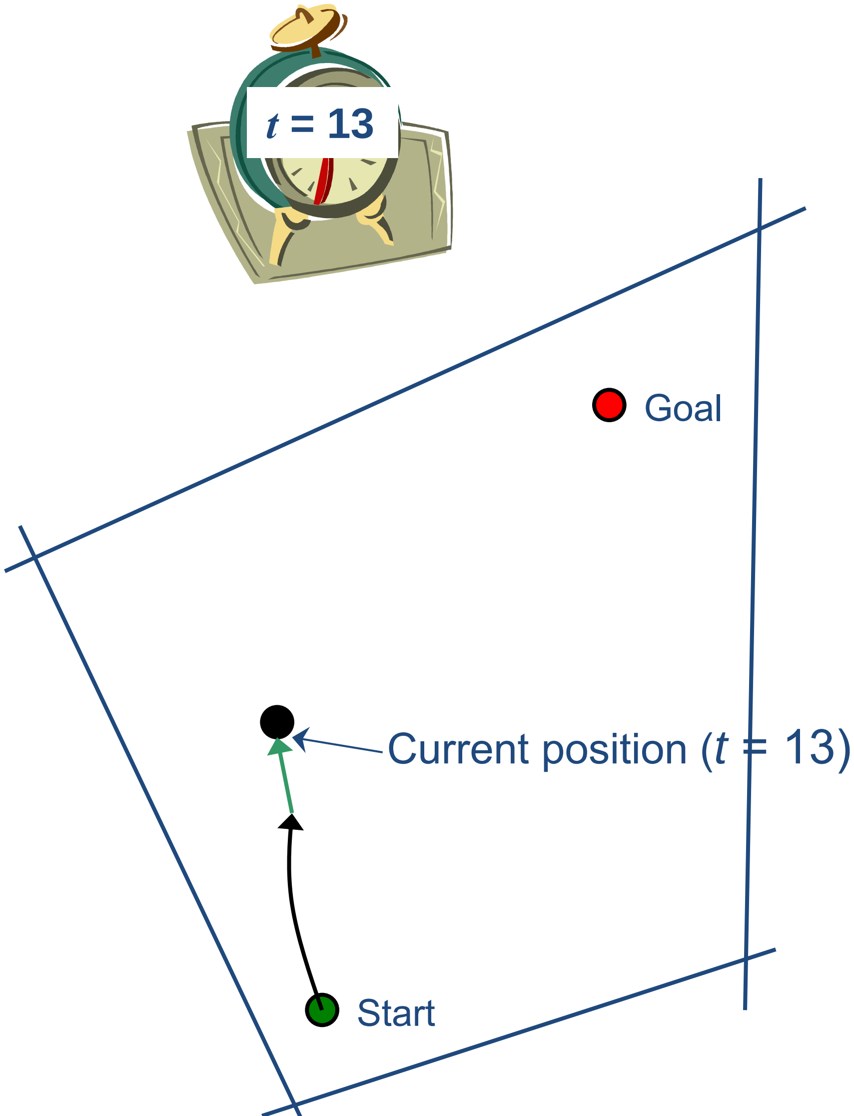
- Position a little bit off from the plan.





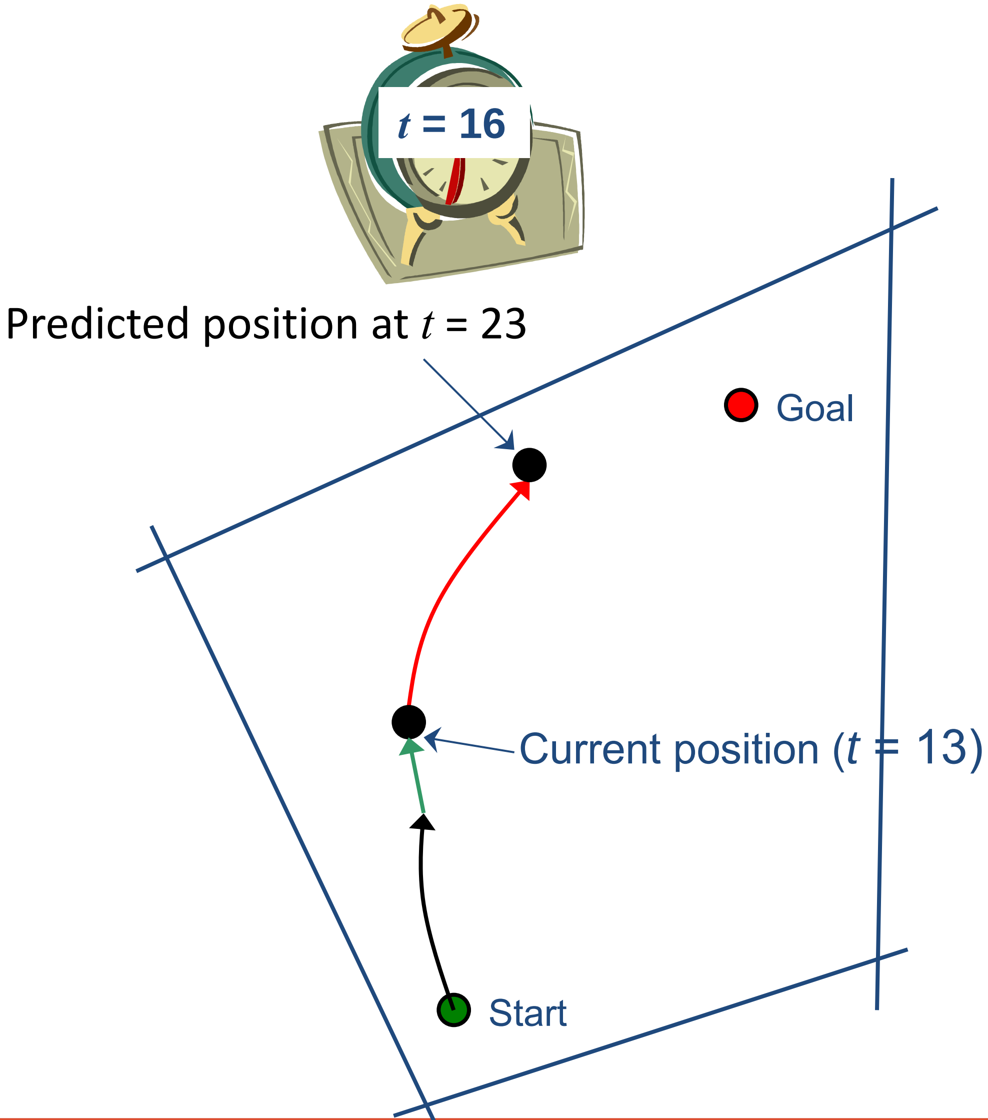
Execution Horizon < Planning Horizon

- Abandon the plan after t = 13.



Model-Predictive, Receding Horizon Control

- Abandon the plan after t = 13.
- Replan for another planning horizon.
- Repeat.



Formulation of Receding Horizon Control

$$\min_{\mathbf{x}_{1:N},\mathbf{u}_{1:N}} J(\mathbf{x}_1 \cdots \mathbf{x}_N, \mathbf{u}_1 \cdots \mathbf{u}_N) + f(\mathbf{x}_N)$$

Cost function

Cost-to-go function

$s.t.$

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k \quad (k = 0,1,\cdots N-1)$$ **Dynamics**

$$\mathbf{H}\mathbf{x}_\mathbf{k} \le \mathbf{g} \quad (k = 0,1,\cdots N)$$ **Spatial constraints**

$$\mathbf{x}_0 = \mathbf{x}_{\text{start}}$$ **Initial position and velocity**

$$\mathbf{x}_N = \mathbf{x}_{\text{goal}}$$ **Goal position and velocity**

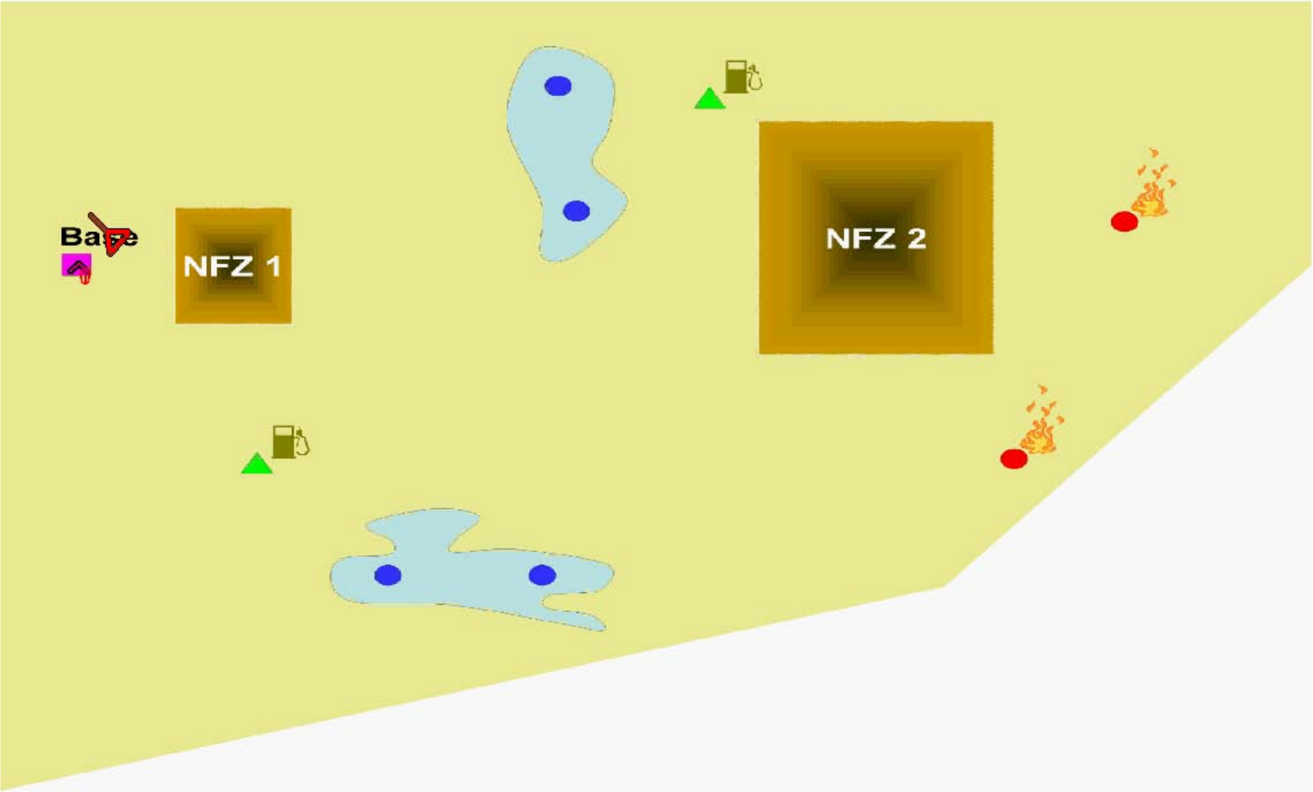
$$-\mathbf{u}_{\max} \le \mathbf{u}_k \le \mathbf{u}_{\max} \quad (k = 0,1,\cdots N-1)$$ **Thrust limits**

$$\mathbf{x}_k \equiv (x_k \quad y_k \quad \dot{x}_k \quad \dot{y}_k)^T, \quad \mathbf{u}_k \equiv (F_{x,k} \quad F_{y,k})^T$$

Fire-fighting Example: Goal-directed Receding Horizon Control

Operator states goals:

"Fires out at Locations 1 & 2
and back to Base within an hour."



Location 1

Location 2

- Plans and schedules activities.
- Routes and "flies" vehicle to achieve plan.

Outline

- Goal-directed Motion Planning (Sulu)
- Compliant, Goal-directed Motion Planning for Under-actuated Robots (Chekov)

Key takeaways

- Motions need to be generated in light of higher level goals.
- For under actuated system, activities and motions couple through state and temporal constraints.
- Compliance is achieved be pre-computing policies for schedules and motions, and by coordinating them in real-time.

A single “cognitive system” language and executive.