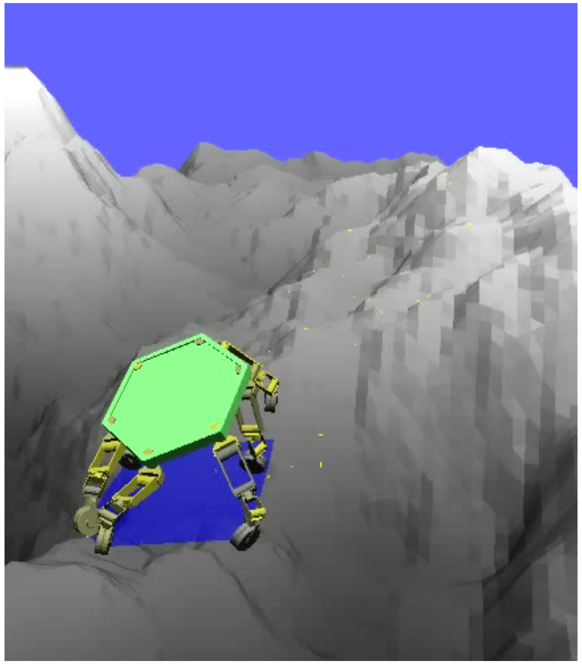
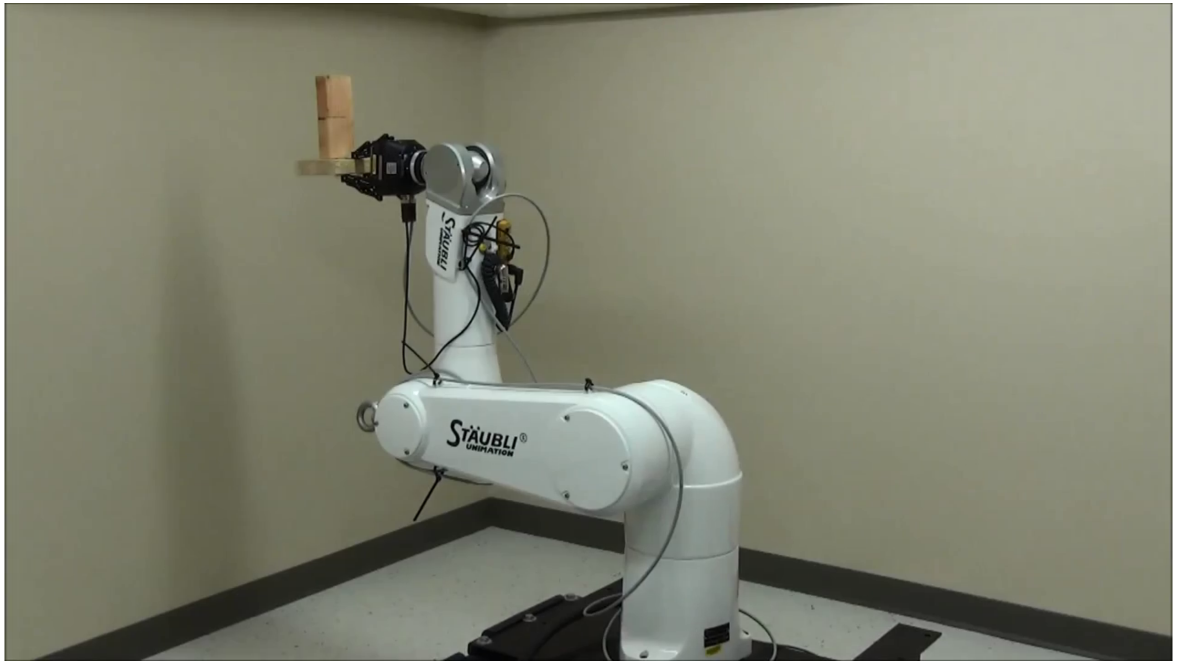
Planning for Manipulation and Locomotion is Unified.

Why not control?





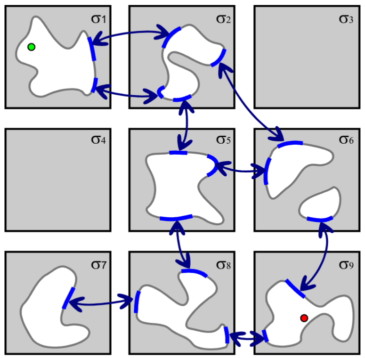
Kris Hauser
ECE & MEMS Departments
Duke University

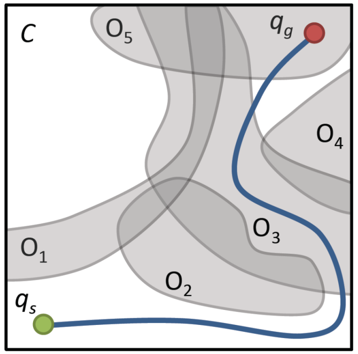
IML Intelligent Motion Lab

DUKE ROBOTICS

Summary

- Theoretically, planning with contact is unified
  - Multi-modal planning
  - Fast time scaling optimization
- Control = fast planning
- Thoughts about whole-body + manipulation control





Idea #1: All is equal in state space

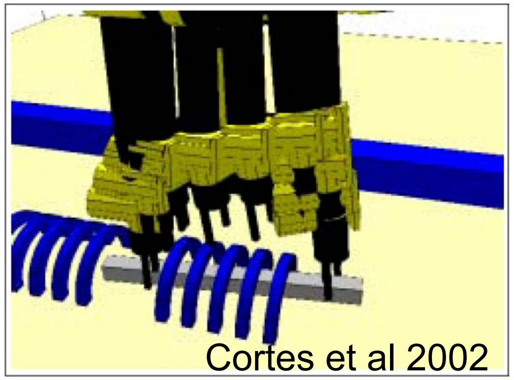
- *We already know* how to model manipulation and locomotion in a uniform manner…
  - In state space
  - Hybrid systems
  - Uncertainty: MDPs/POMDPs

Multi-Modal Planning

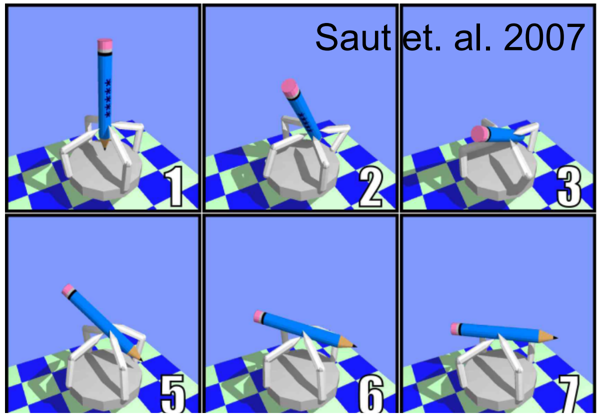
- The system must **switch** between continuous motion spaces
  - The discrete sequence of spaces must be chosen along with the continuous motion
- Instances
  - Reconfigurable robots
  - Switched actuators
  - Motion with tethers
  - Manipulation
  - Legged locomotion
  - LTL specifications

Bretl 2006

Stilman et. al. 2007



Cortes et al 2002

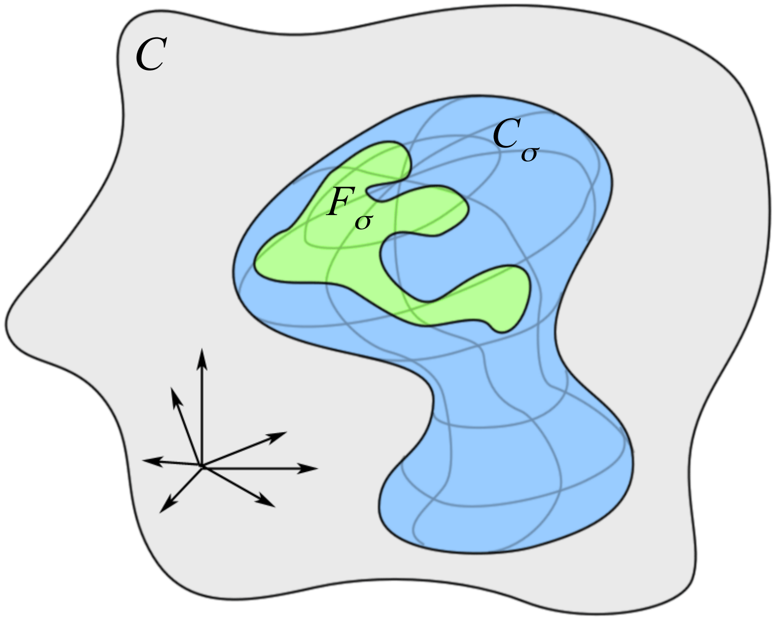


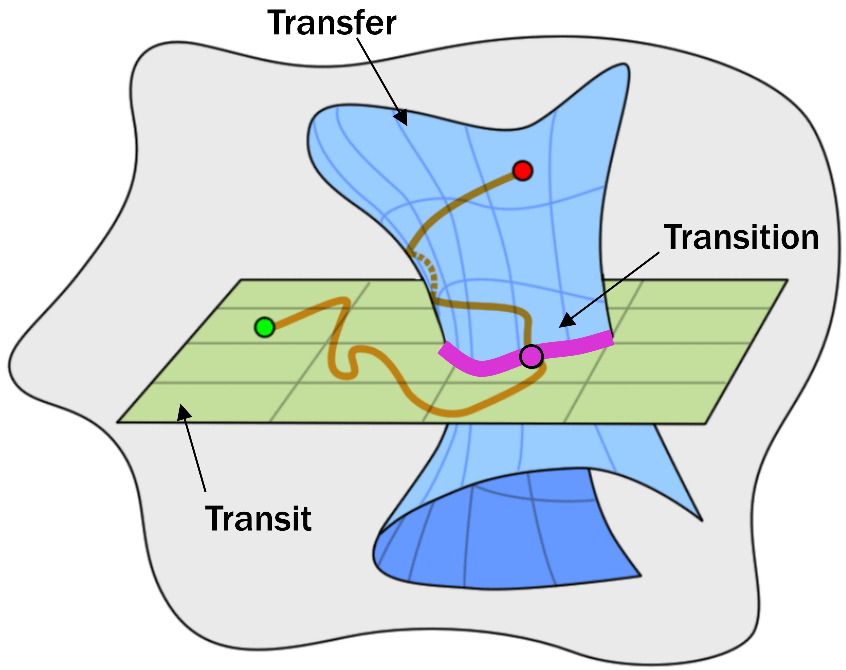
Saut et. al. 2007

Modes

[Hauser and Latombe 2009]
[Hauser and Ng Thow Hing 2010]

- A **mode** σ defines a continuous subspace of feasible motion $F\downarrow\sigma$
- Two modes intersect at a **transition**

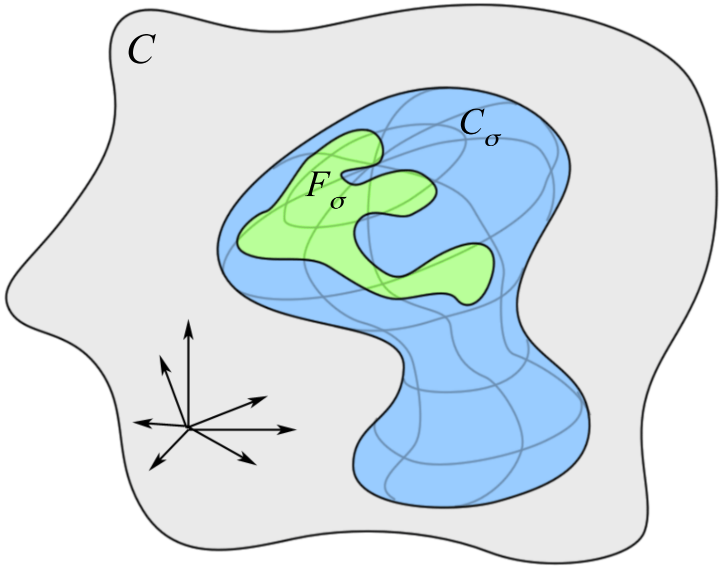


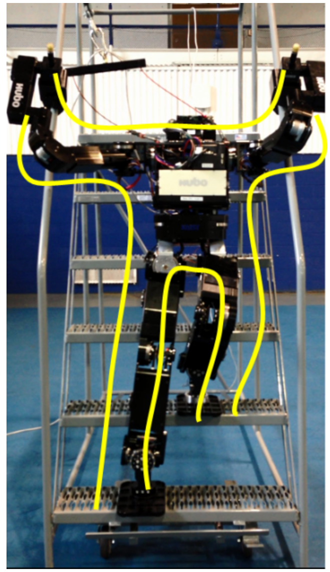


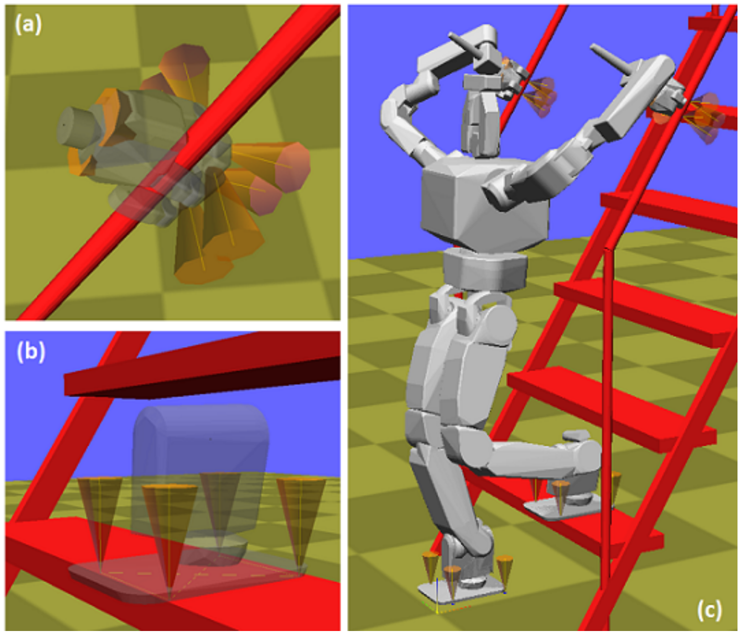
C_{robot} x C_{object}

Legged Robot Motion Constraints

- Full-body coordination: 6DOF "virtual base" + N joints
- Collision avoidance
- Joint limits
- Torque limits
- Frictional contact







At all configurations $q \in SE(3) \times \mathbb{R}^N$

- $c_1(q)=d_1, \ldots, c_n(q)=d_n$
- Colliding(q)=false
- There exist joint torques τ and contact forces $f_1,\ldots,f_n$ to satisfy the *quasistatic balance equations*:

$$G(q) = \tau + \sum_i J_i^T(q) f_i$$

$$f_i \in FC_i \text{ for all } i=1,\ldots,n$$

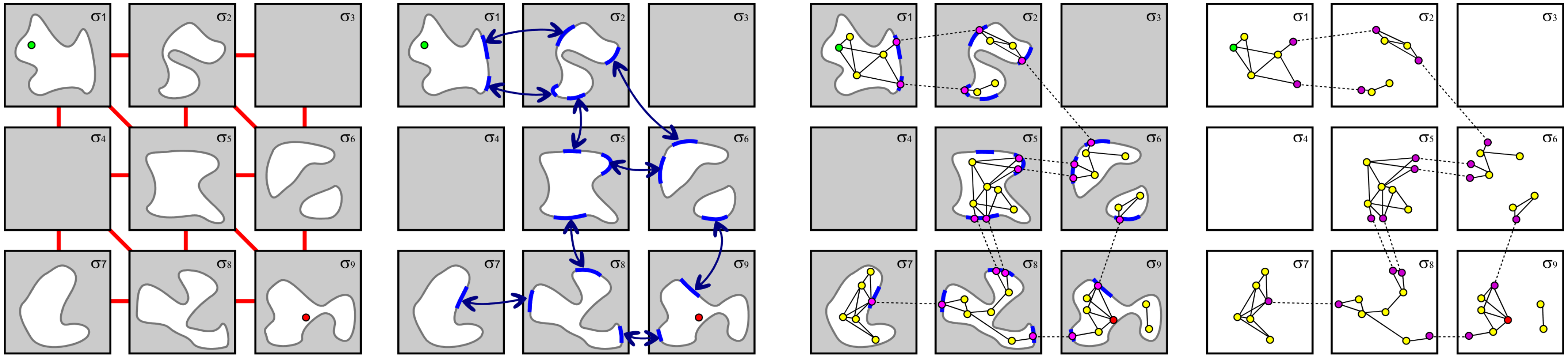
G: the generalized gravity vector
J_i: the Jacobian of the i'th contact point
FC_i: the i'th friction cone

Multi-modal planning: the discrete mode case

[Hauser and Latombe 2009]

- Given a discrete set of modes whose subspaces intersect, can plan reliably using a multi-modal planner

Mode graph | Transition regions | Sampling single-mode roadmaps | Search global roadmap

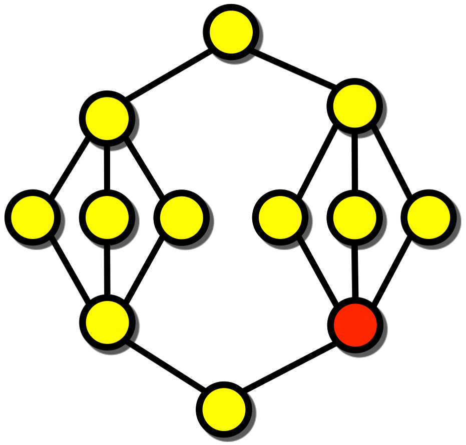


- MMPRM: probabilistic completeness, exponential convergence guarantees

Exploiting discrete structure

- The mode graph itself is a rich source of structure
- Modes often arise from **task-and-motion planning** problems => use efficient task planners for guidance

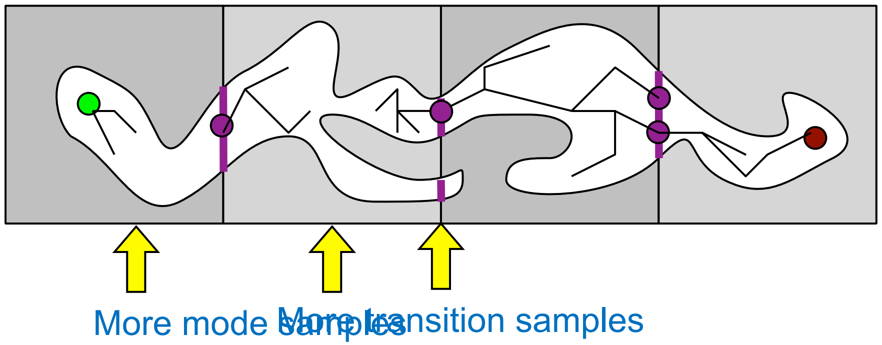
***Discrete strategy*: Selecting fruitful modes using search**



***Completeness*: eventually consider all modes**

Bretl et al 2004
Nielsen and Kavraki 2000

***Continuous strategy*: Biasing mode / transition samples**



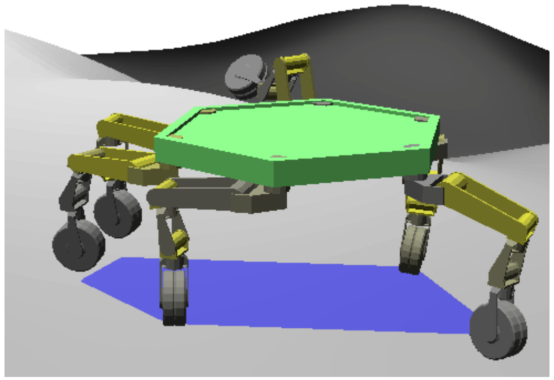
***Completeness*: each mode & transition sampled O(N) times**

[Hauser and Latombe 2009]

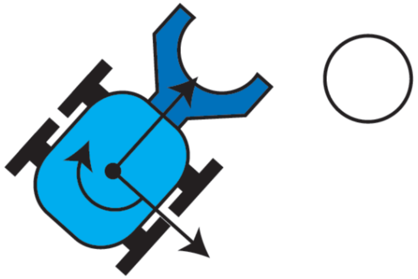
The continuous case: dimension and codimension

[Hauser and Ng Thow Hing 2010]

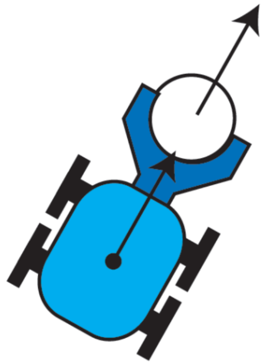
- Codimension: the variables *not involved in motion*, but involved in defining the shape of $F\downarrow\sigma$



Stance & footstep locations



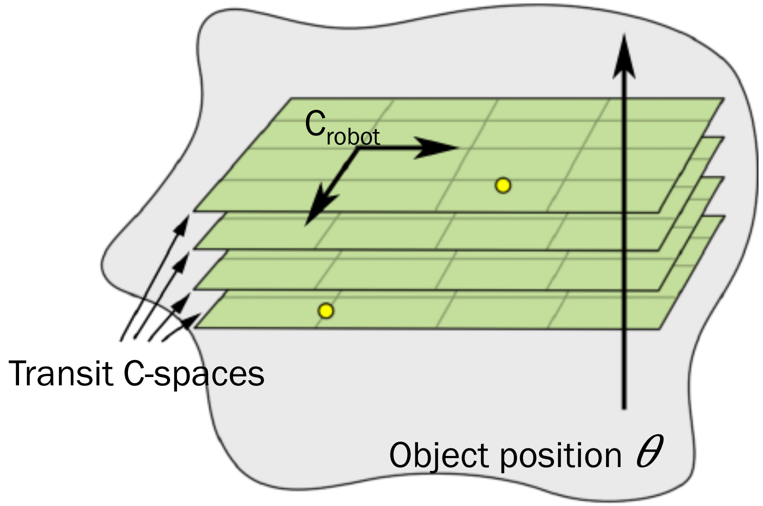
Transit space:
location of object



Transfer space:
grasp parameters

- Infinity of manifolds

Definition



- Define $\sigma=(m,\theta)$
  - m in 1,…,M the mode "family"
  - $\theta \in \Sigma\downarrow m$ the codimension
- $\Sigma\downarrow m$ is the codimensional space
- m indicates how codimensions map to constraint changes
- Tree-growing random sampling in codimension space: probabilistically complete [Hauser and Ng-Thow-Hing 2010]
- Dimensions in one family will become the codimension of other families
- *Rich source of structure*

Multi-modal structure

- $F(\mathit{transit}, T_{obj0}) = (q_{rob}, T_{obj})$ *s.t.*
  $q_{min} \leq q_{rob} \leq q_{max}$

Joint limits

$R(q_{rob}) \cap E = \emptyset$

Robot-env collision

$R(q_{rob}) \cap T_{obj} \cdot O = \emptyset$

Robot-obj collision

$T_{obj} = T_{obj0}$

Stationary object

- $F(\mathit{transfer}, T_{grasp}) = (q_{rob}, T_{obj})$ s.t.

$q_{min} \leq q_{rob} \leq q_{max}$

Joint limits

$R(q_{rob}) \cap E = \emptyset$

Robot-env collision

$T_{obj} \cdot O \cap E = \emptyset$

Obj-env collision

$T_{obj} = T_{ee}(q_{rob}) \cdot T_{grasp}$

Stationary grasp

Multi-modal structure

- $F(transit, T_{obj}^{0}) = (q_{rob}, T_{obj})$ s.t.
  $q_{min} \leq q_{rob} \leq q_{max}$

$R(q_{rob}) \cap E = \emptyset$

$R(q_{rob}) \cap T_{obj} \cdot O = \emptyset$

$T_{obj} = T_{obj}^{0}$

Precomputed roadmaps

- $F(transfer, T_{grasp}) = (q_{rob}, T_{obj})$ s.t.

$q_{min} \leq q_{rob} \leq q_{max}$

$R(q_{rob}) \cap E = \emptyset$

$T_{obj} \cdot O \cap E = \emptyset$

$T_{obj} = T_{ee}(q_{rob}) \cdot T_{grasp}$

Multi-modal structure

- $F(transit, T_{obj}^{0}) \cap F(transfer, T_{grasp}) = (q_{rob}, T_{obj})$ s.t.
  $q_{min} \leq q_{rob} \leq q_{max}$

$$R(q_{rob}) \cap E = \emptyset$$

$$R(q_{rob}) \cap T_{obj} \cdot O = \emptyset$$

$$T_{obj} \cdot O \cap E = \emptyset$$

$$T_{obj} = T_{obj}^{0}$$

$$T_{obj} = T_{ee}(q_{rob}) \cdot T_{grasp}$$

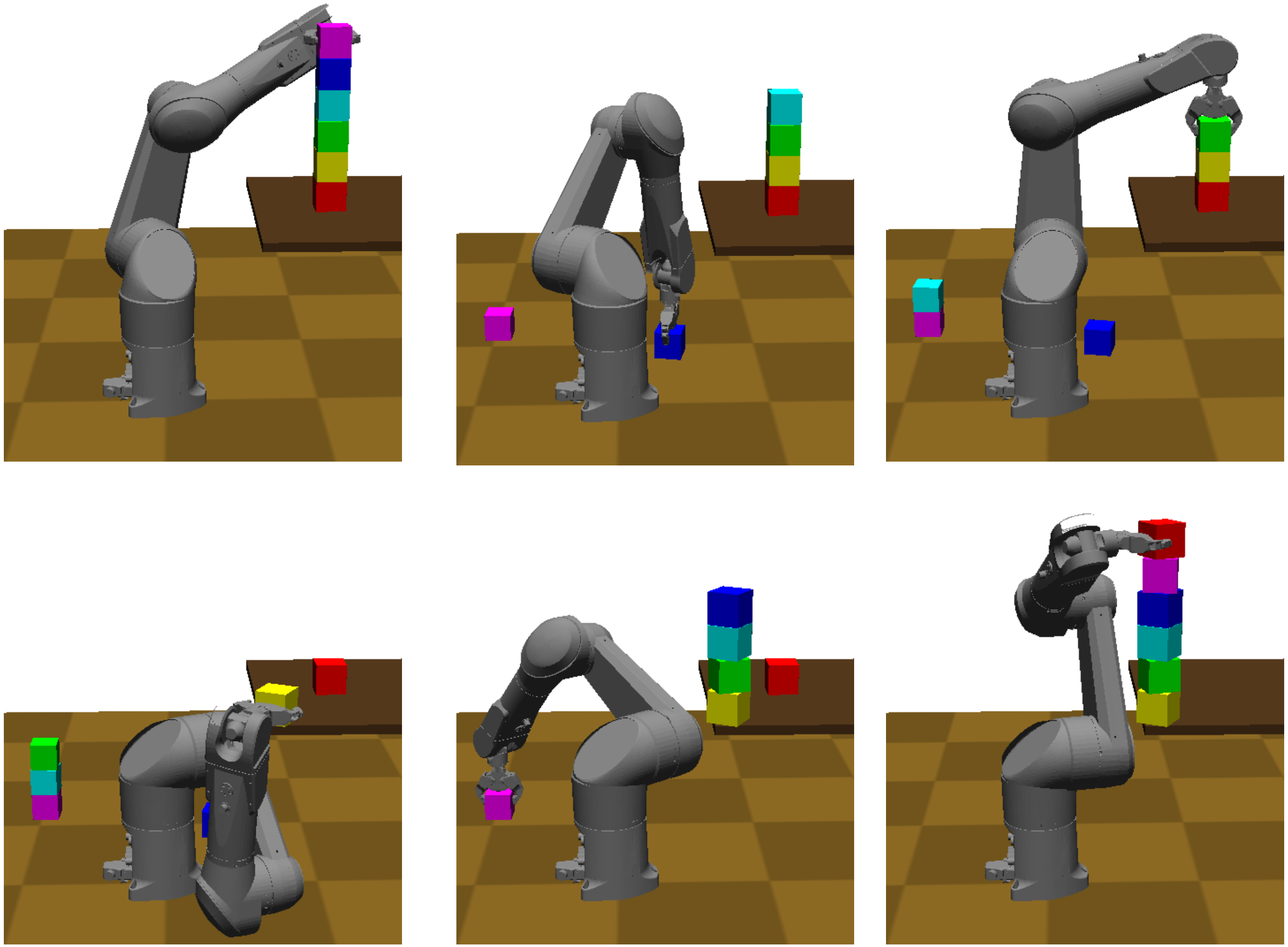
$$T_{obj}^{0} \in \text{Stable}$$

$$T_{grasp} \in Grasps$$

Object feasibility: precompute stable collision-free placements

Mode reachability: test workspace limits

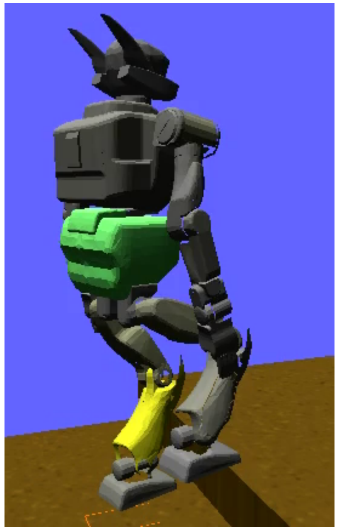
Multi-Object Pick and Place



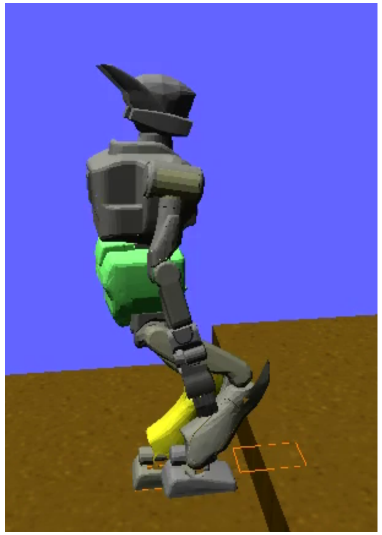
Exploiting Regularity: Motion Primitives

[Hauser et al 2008]
[Luo et al 2014]

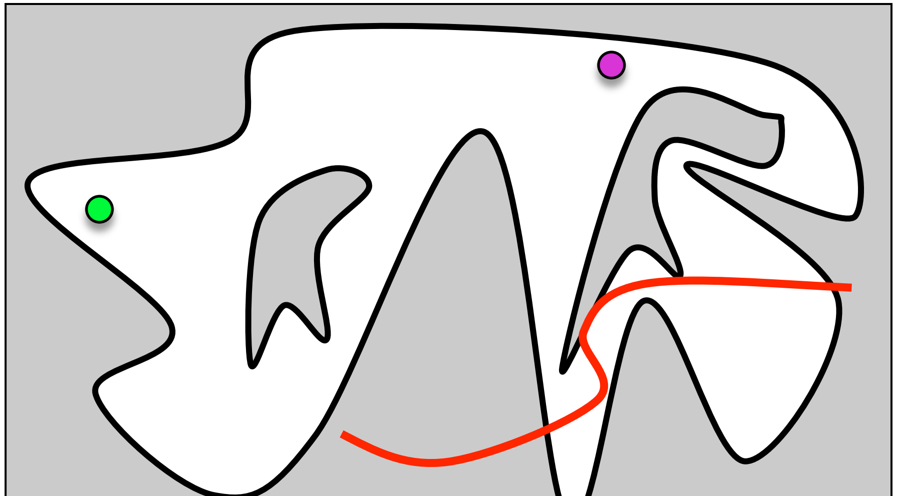
- A small change in θ makes a small change in $F\downarrow(m,\theta)$
- Existing solutions can be warped to similar modes



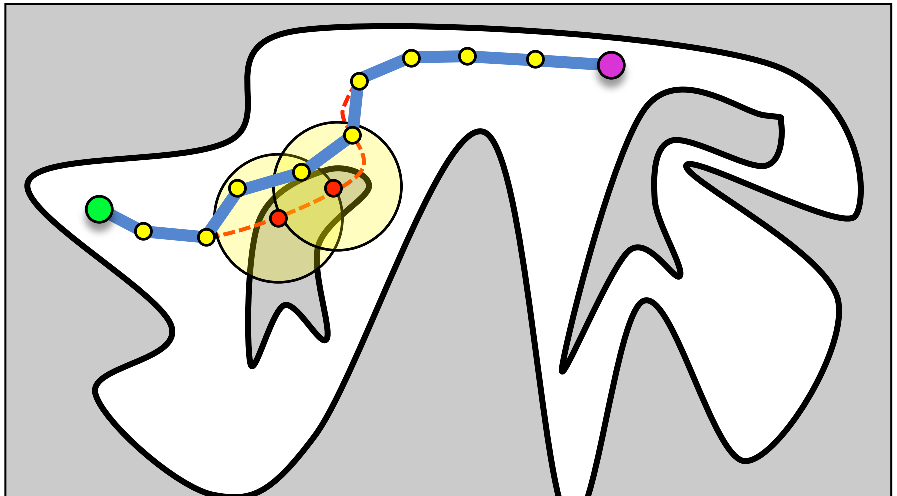
Original motion clip (flat ground step)



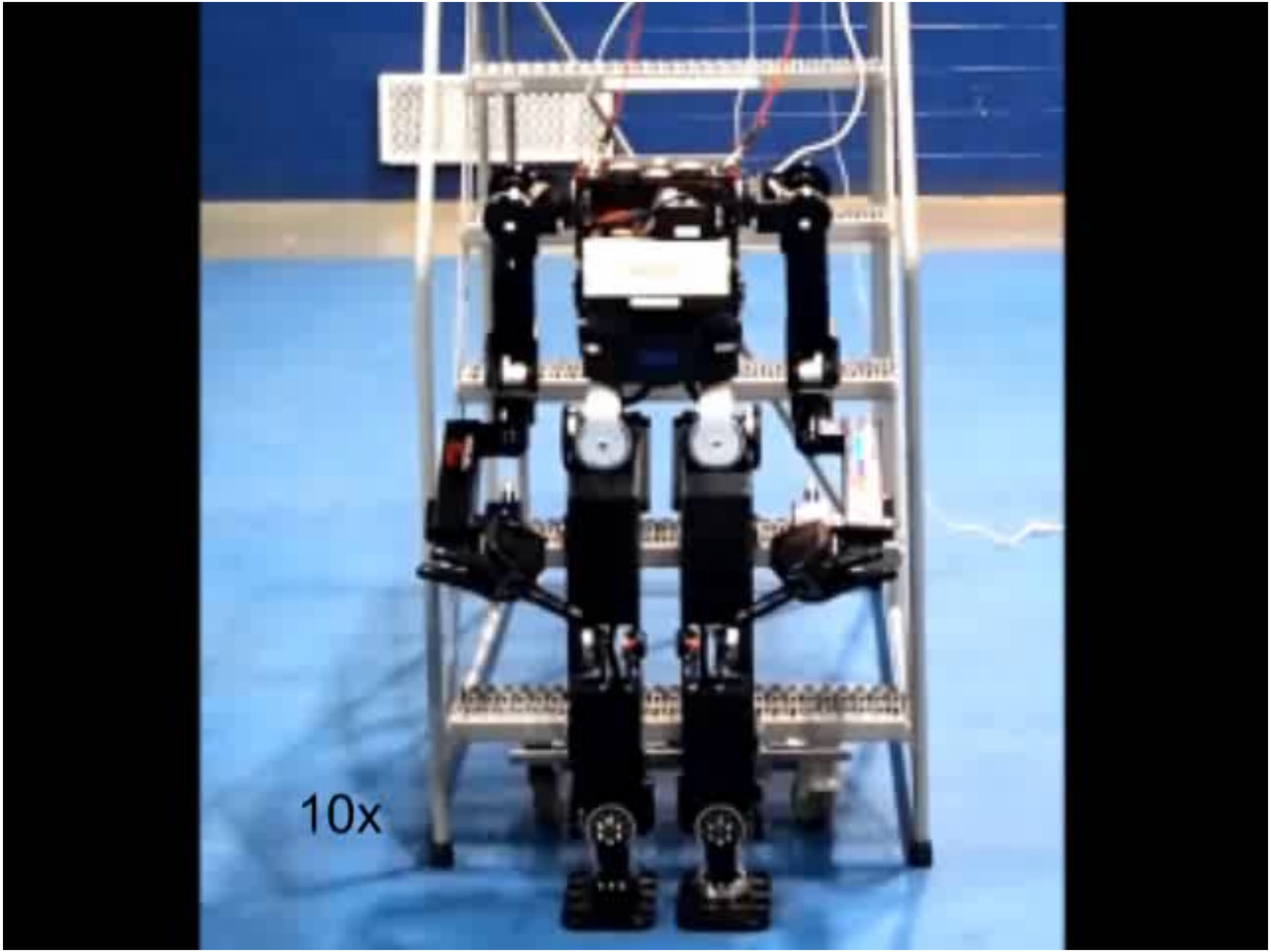
Adapted to stair



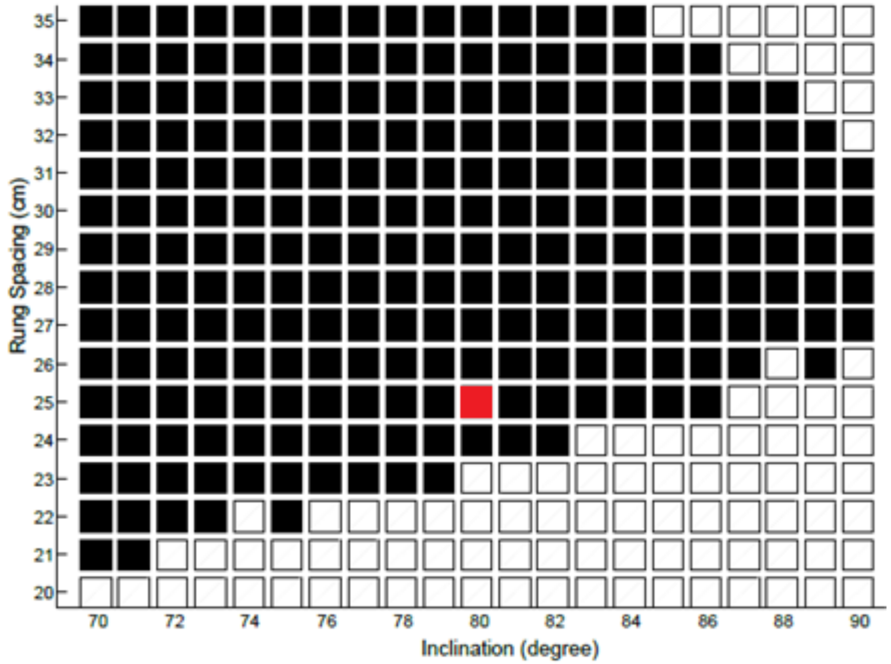
1) Transform primitive



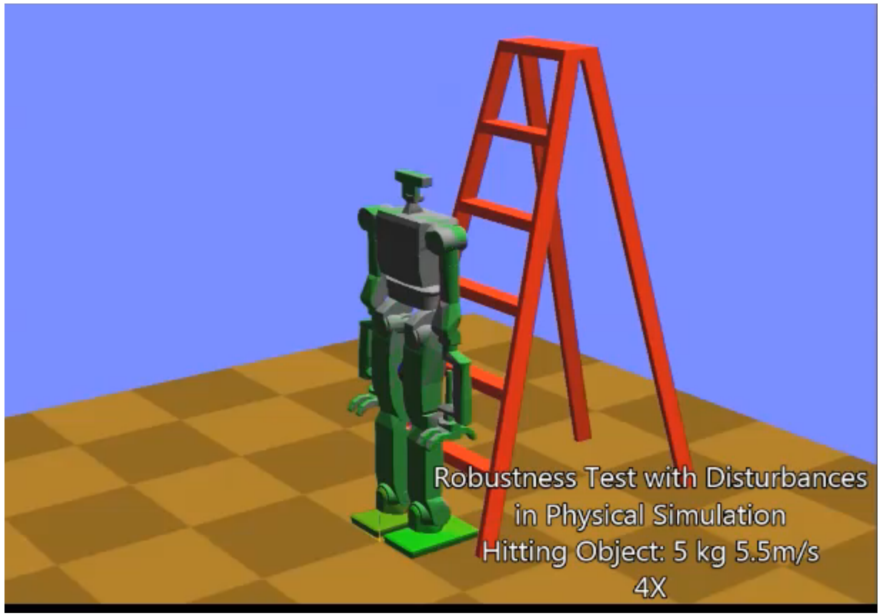
2) Bias sampling


10x

- **Robustness of primitive adaptation**
  - Ladders with varying inclination [70°, 90°] and rung spacing [20 cm, 35 cm]
  - Using motion primitives designed specifically for 80°, 25cm
  - 72% success rate

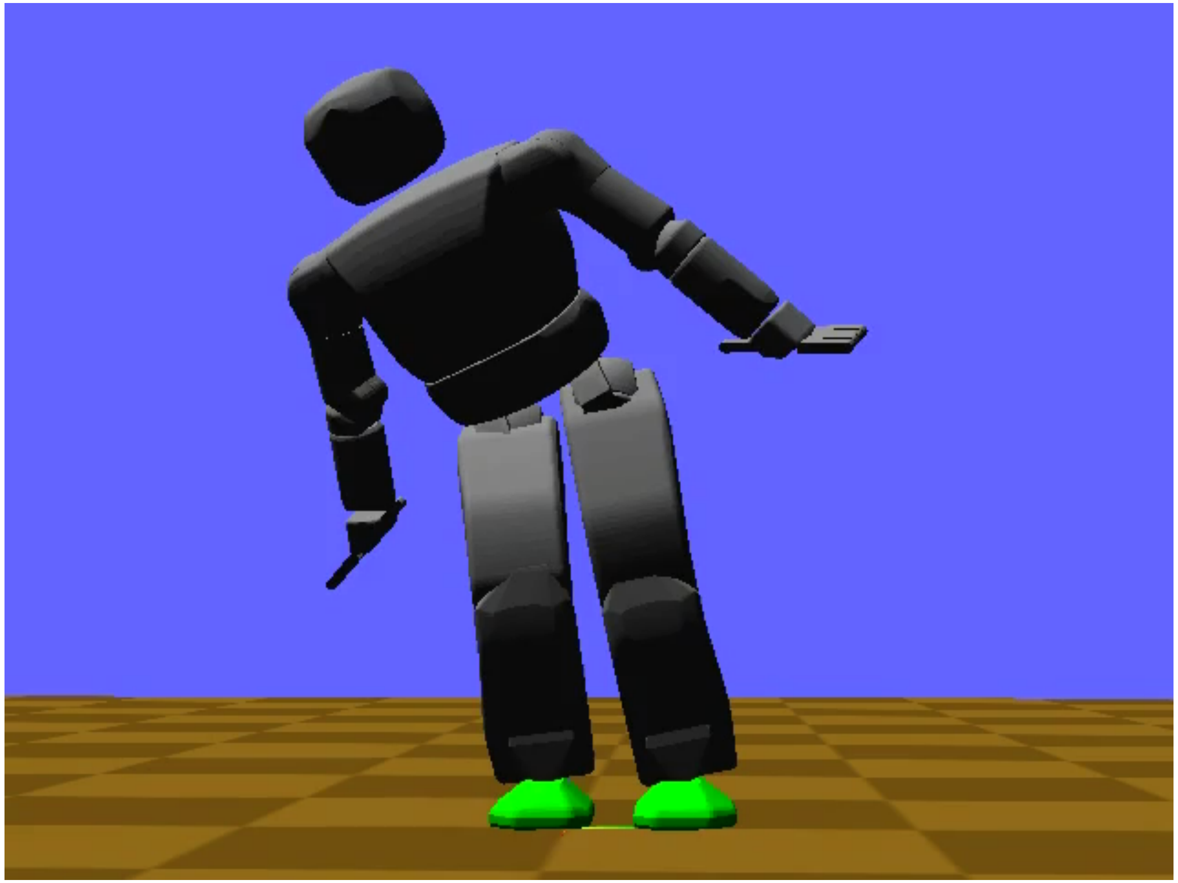


- **Robustness of controller**
  - Compliant arm: errors in hand placement tolerated as long as rail in finger range
  - Up to 2cm variation in rung vertical position tolerated
  - Disturbances: up to 15kg dropped on robot at 5.5m/s



The Role of Dynamics

- Generating a geometric path is relatively easy, but…

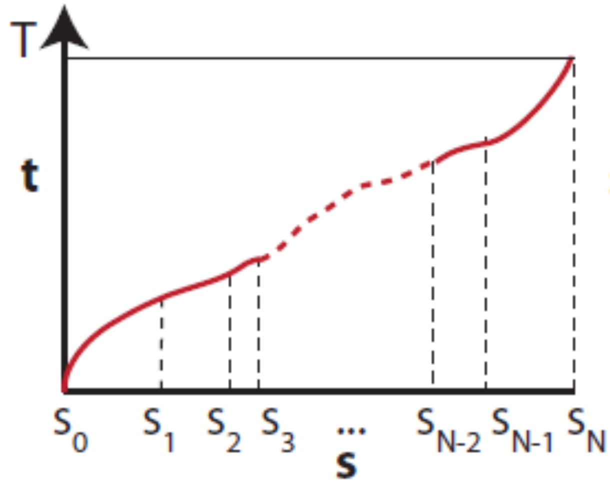


Direct execution of geometric path

Problem Definition

[Hauser, RSS 2013]
[Hauser, ICRA 2014]
[Hauser, IJRR 2014]
[Luo and Hauser, RSS 2015]

- Dynamic constraints
  - Velocity constraints $v_{min} \leq q \leq v_{max}$
  - Acceleration constraints $a_{min} \leq q \leq a_{max}$
  - Torque constraints $\tau_{min} \leq \tau \leq$



Dynamic time-scaling with contact

Input

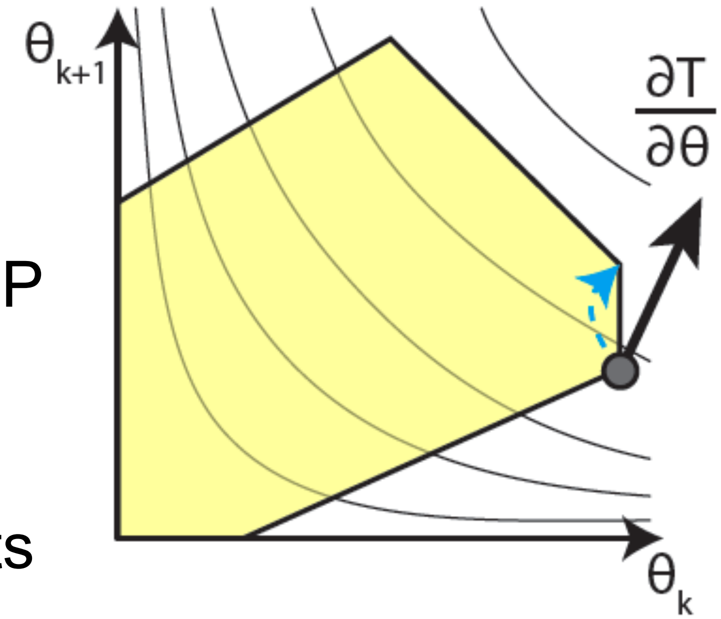
- Geometric path $p(s):[0,1] \to \mathbb{R}^n$ (continuous first derivatives)
- Stance $\sigma(s)$ maps a point along path to the set of active contact points
- Dynamic constraints

Output

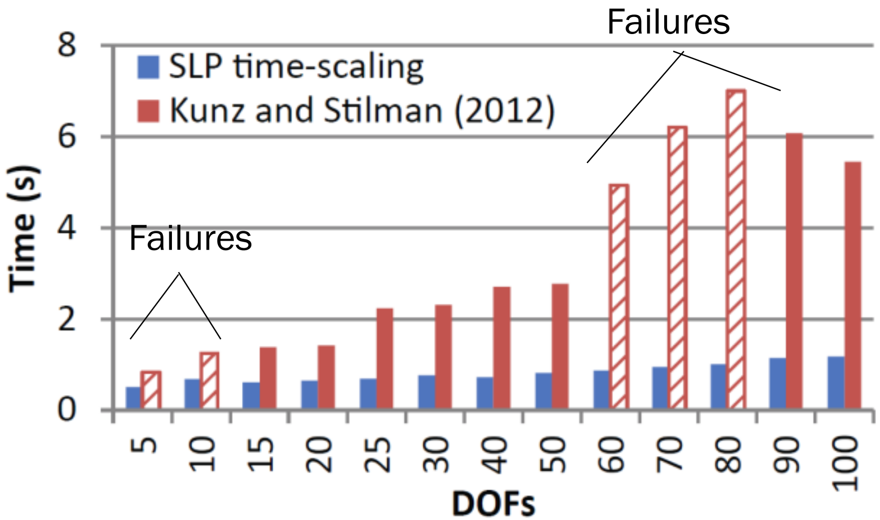
- Time scaling $s(t):[0,T] \to [0,1]$ such that trajectory $q(t)=p(s(t))$ satisfies dynamic constraints at each time point t

Efficient Convex Optimization Formulation

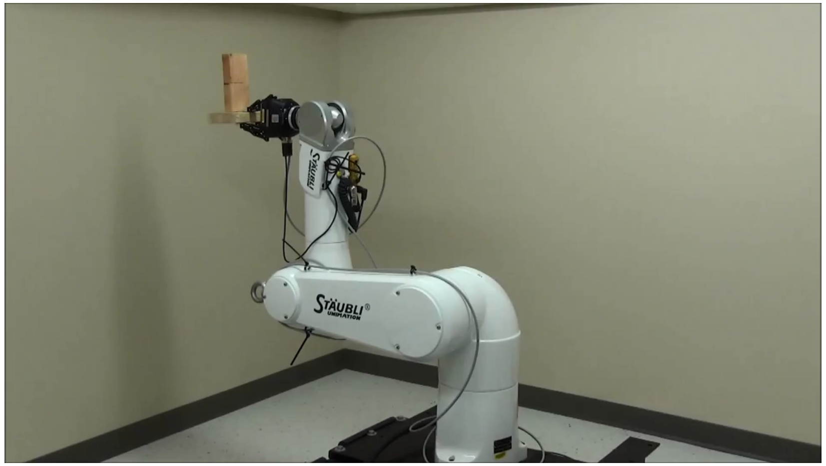
- Convex optimization formulation of time-scaling [Verschure et al 2009]
  - Unique minimum
  - Solved quickly/robustly using SLP techniques [Hauser 2013]
  - “Bulletproof”
- Fast pruning of irrelevant constraints
- Solution times ~1s for 100D+ robots

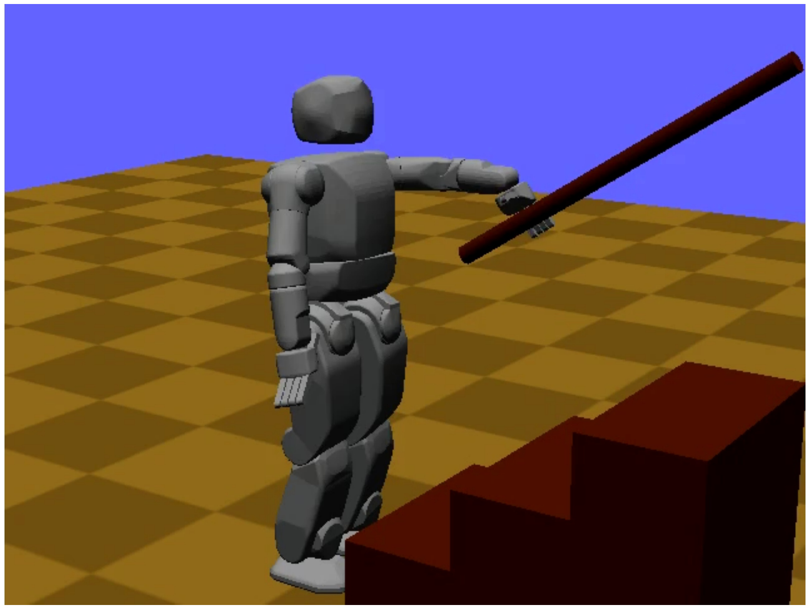


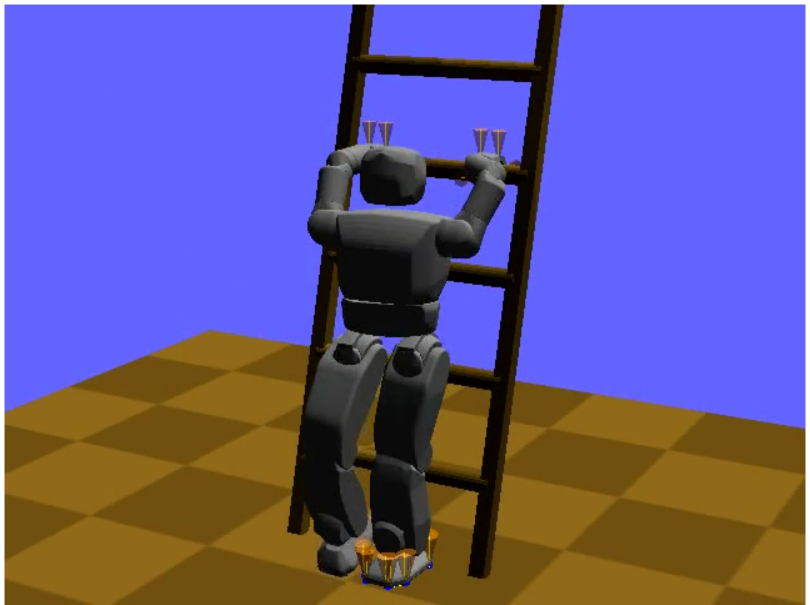
MInTOS trajectory optimizer: http://motion.pratt.duke.edu/mintos



Faster and more reliable than existing techniques



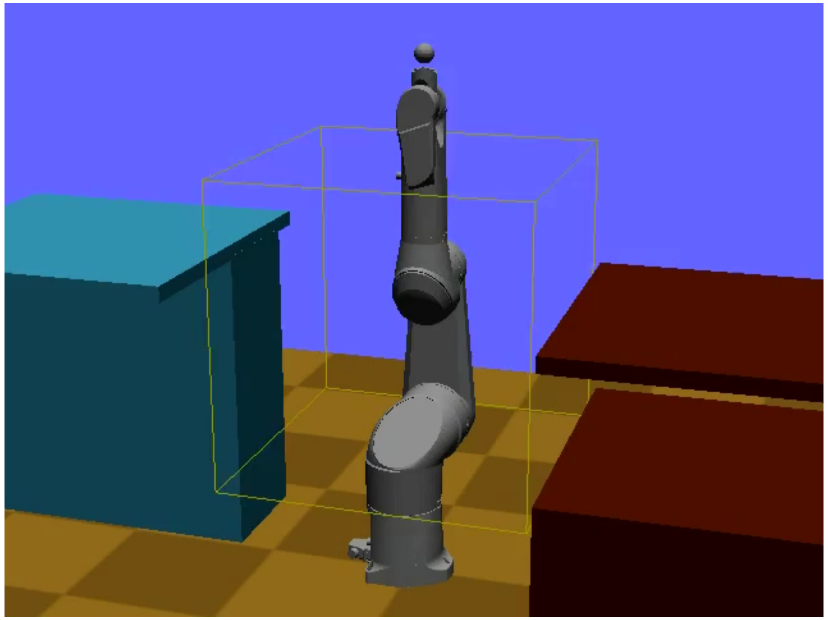




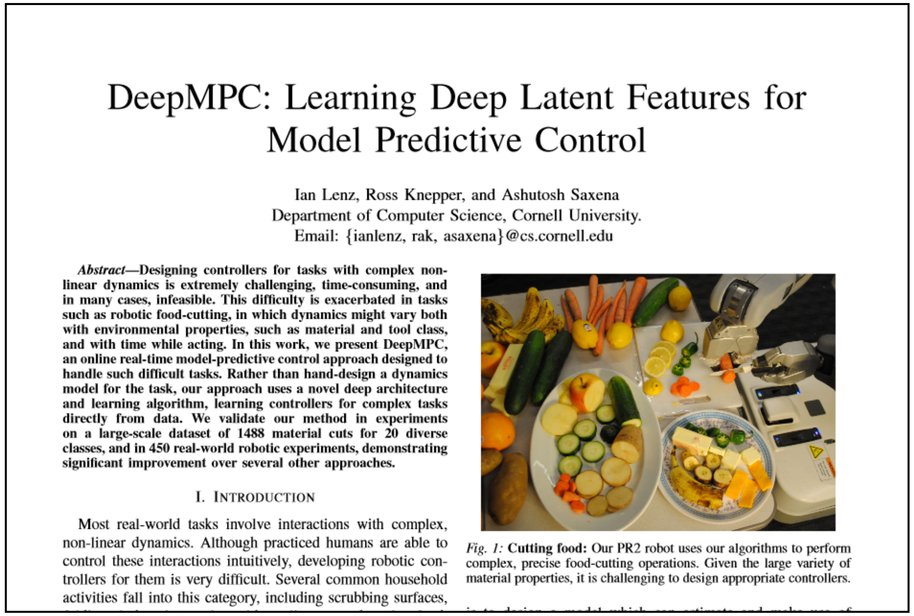
Hauser, RSS 2013, ICRA 2014 and, IJRR 2014

Control = Fast Replanning

- MPC is becoming fast enough for relatively high-D problems… without contact



K. Hauser, WAFR 2010, Autonomous Robots 2011



DeepMPC: Learning Deep Latent Features for Model Predictive Control

Ian Lenz, Ross Knepper, and Ashutosh Saxena
Department of Computer Science, Cornell University.
Email: {ianlenz, rak, asaxena}@cs.cornell.edu

Abstract—Designing controllers for tasks with complex non-linear dynamics is extremely challenging, time-consuming, and in many cases, infeasible. This difficulty is exacerbated in tasks such as robotic food-cutting, in which dynamics might vary both with environmental properties, such as material and tool class, and with time while acting. In this work, we present DeepMPC, an online real-time model-predictive control approach designed to handle such difficult tasks. Rather than hand-design a dynamics model for the task, our approach uses a novel deep architecture and learning algorithm, learning controllers for complex tasks directly from data. We validate our method in experiments on a large-scale dataset of 1488 material cuts for 20 diverse classes, and in 450 real-world robotic experiments, demonstrating significant improvement over several other approaches.

I. Introduction

Most real-world tasks involve interactions with complex, non-linear dynamics. Although practiced humans are able to control these interactions intuitively, developing robotic controllers for them is very difficult. Several common household activities fall into this category, including scrubbing surfaces,

Fig. 1: **Cutting food:** Our PR2 robot uses our algorithms to perform complex, precise food-cutting operations. Given the large variety of material properties, it is challenging to design appropriate controllers.

Lenz et al, RSS 2015

…and many others…

Idea #2: Where's the disconnect?

- In the *hacks*
- E.g.
  - LIPM: a idealization of biped locomotion
  - Grid planning for a 2D cylinder : an approximation of biped navigation
  - Force closure: a idealization of grasping
  - Guarded moves: an idealization of optimal policy under uncertainty
  - Visual servoing: uncertainty management
- Reasonable approximations… in some instances
- Made for computational convenience, mathematical elegance, laziness, time pressure

Thoughts

- These “hacks” are incredibly useful
  - Managing uncertainty
  - Responsive motion
- MPC not (yet) fast enough for full-body problems with contact
- At the moment, a “unified framework” will need to *exploit* hacks, NOT replace them

Thank you!

- Students + collaborators
  - Tim Bretl
  - Jean-Claude Latombe
  - Paul Oh
  - C.S. George Lee
  - Jingru Luo
  - Yajia Zhang
- Sponsors
  - DARPA Robotics Challenge Track A
  - NSF RI #1218534