Variational Integration Methods for Simulating and Designing Systems with Simultaneous Impact

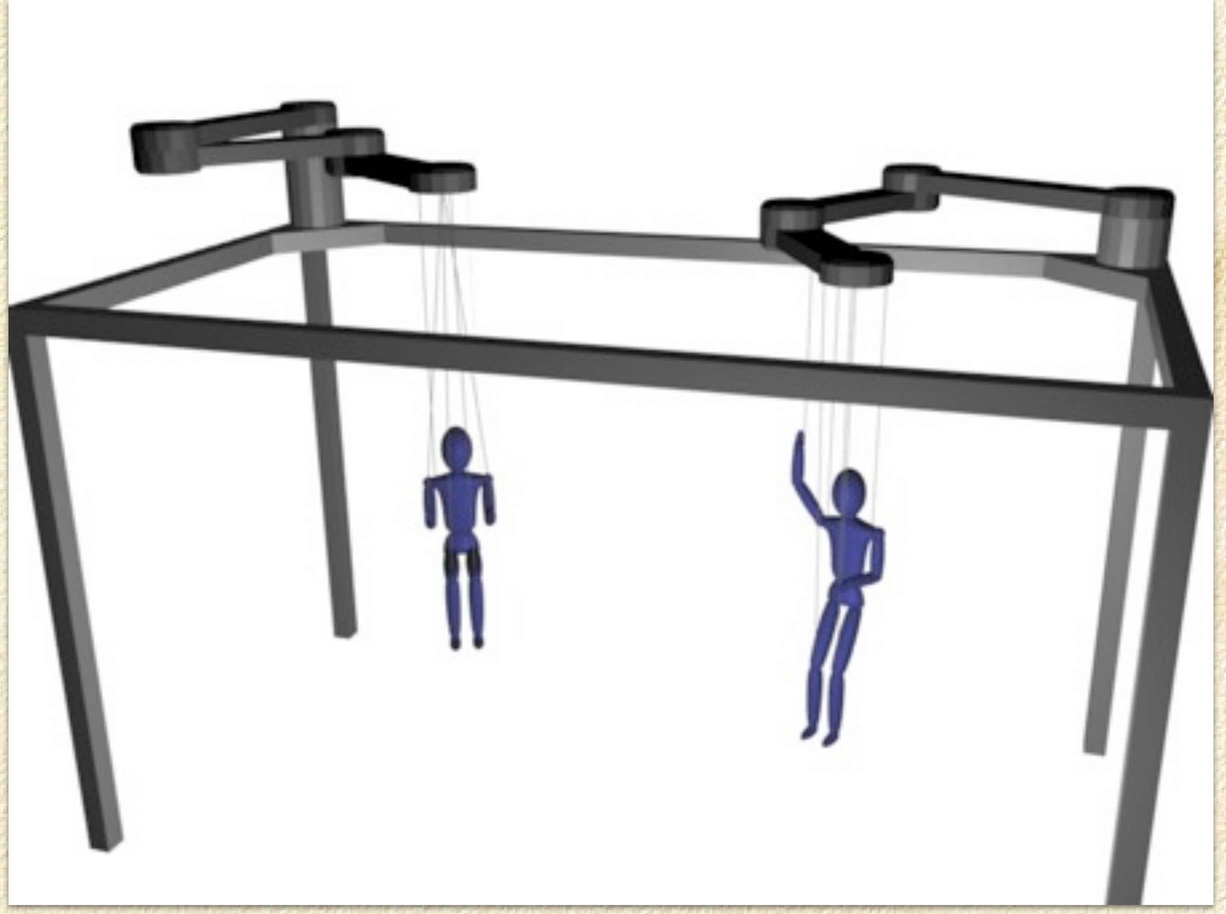
Todd Murphey
Northwestern University
June 27, 2010

Purpose of Simulation

- Simulation provides a way to predict physical behavior
- Depending on computational structures provided, it provides infrastructure for control design
- Examples include...

Marionettes as Complex Systems

- 40-50 DOF
- Nontrivial constraints
- Generalized coordinates for control analysis
- Force balance by hand is not feasible



Collaboration with Georgia Tech, Atlanta Center for Puppetry Arts, and Disney R&D/Imagineering

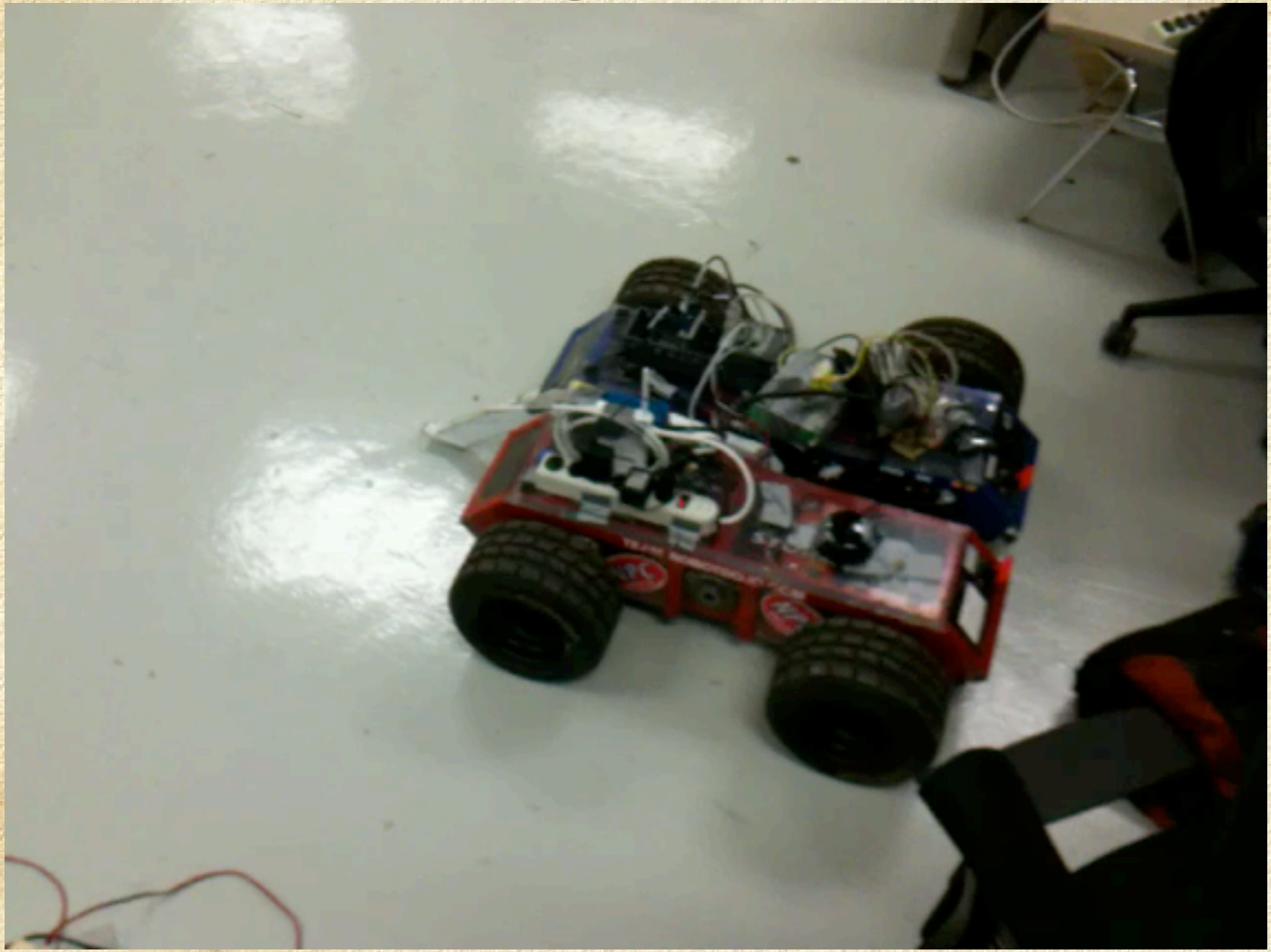
The Pygmalion Project

- Pygmalion is the story of a sculptor falling in love with his sculpture and begging the gods to bring it to life
- We took motion capture data of these dancers acting out a short version of the Pygmalion story
- Goal: automatically generate tracking control for marionettes

Robotic Marionettes

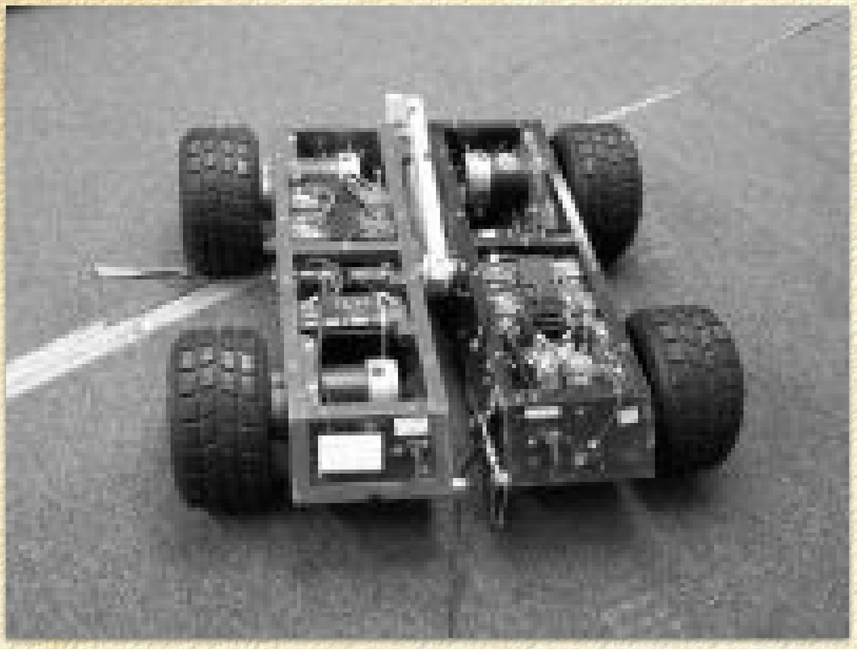
- marionette skeleton is actuated by three actuators
- notice the connections for the strings creates a design problem

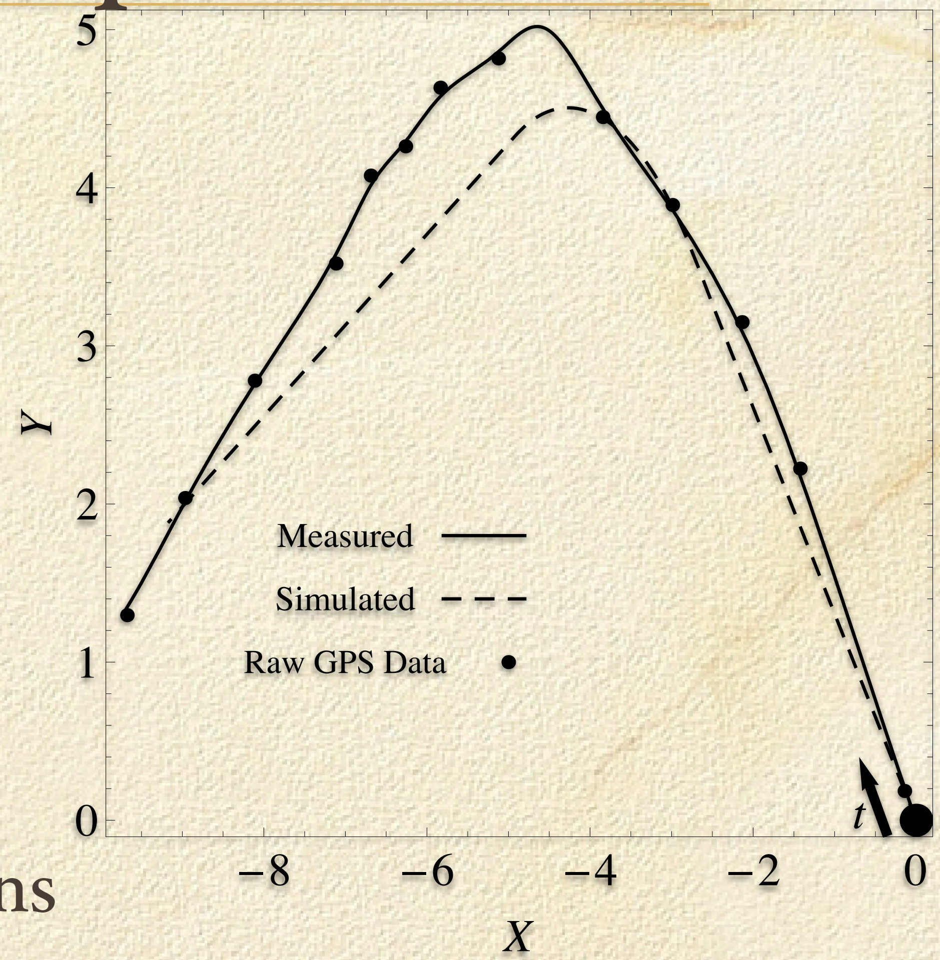
Skid-Steering Experiment



"Sticking" is real on hard surfaces

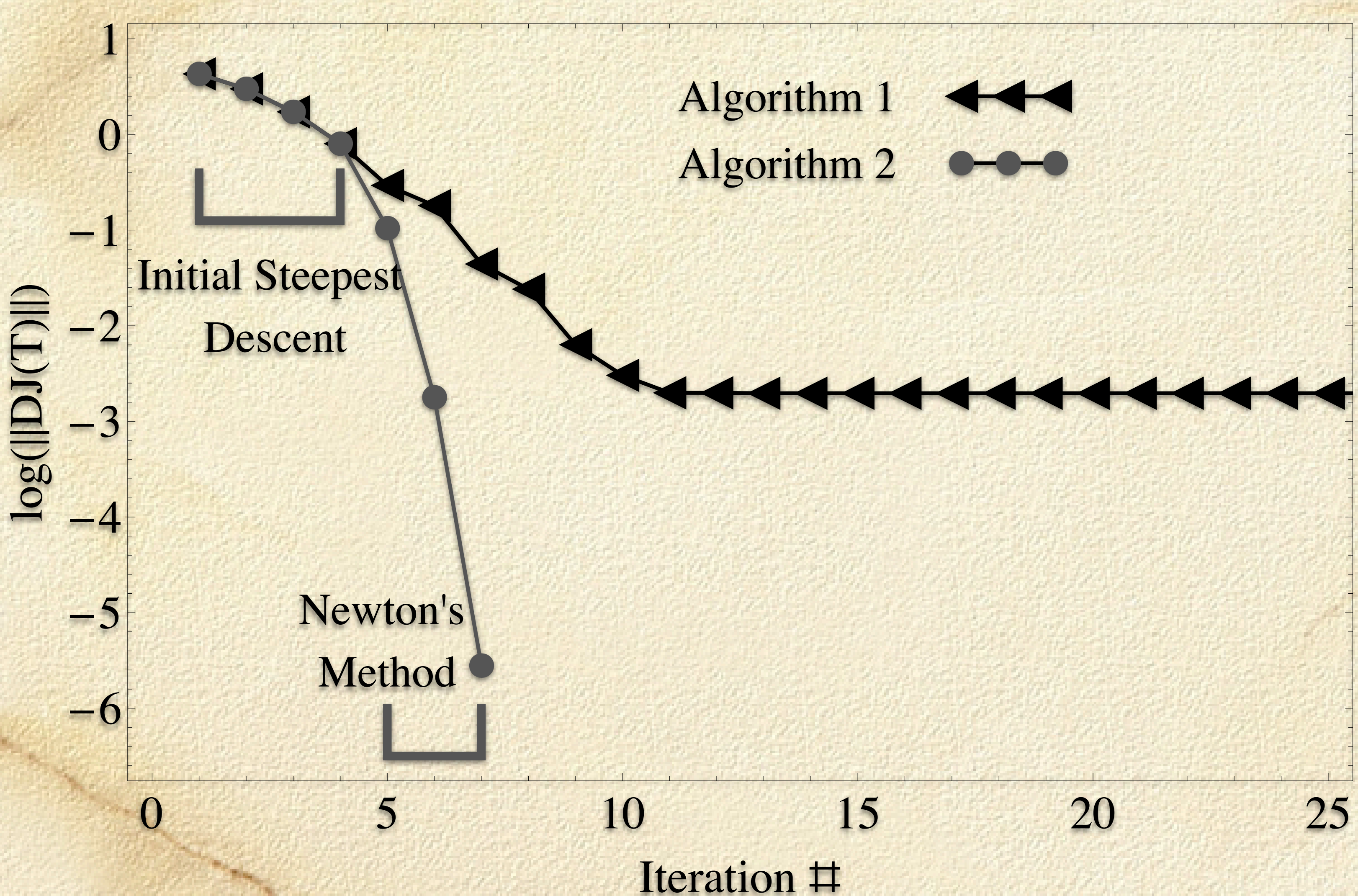
Skid-Steering Experiment





- Very noisy GPS data at 1 Hz
- Converges in just a few iterations
- Can run on-board optimizations over 10 seconds of data in 50 ms, so real-time is at least not crazy

Skid-Steering With Uncertainty



How Do We Model For Control?

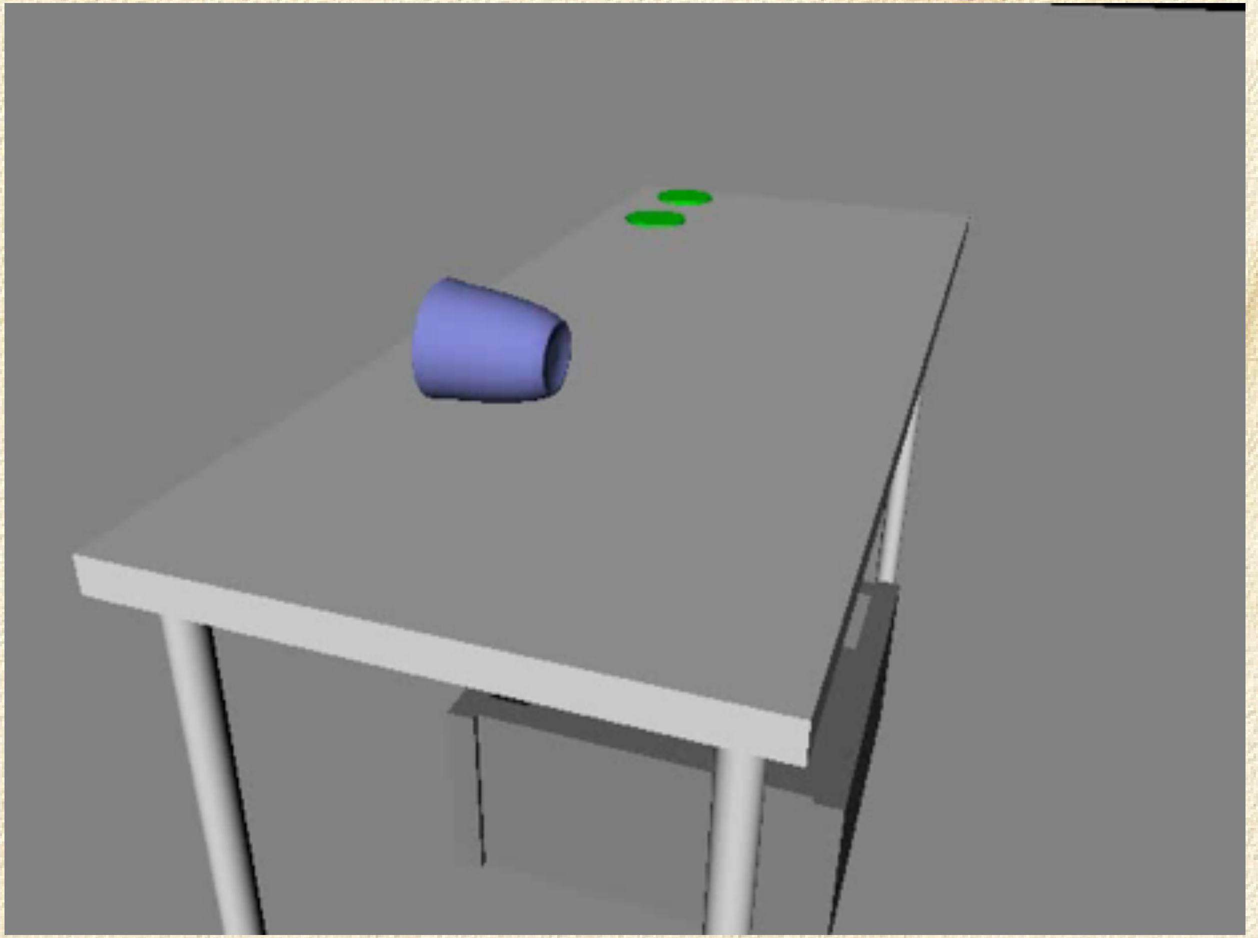
- Modeling for control is different from modeling for simulation
- Example simulation packages
  - Sim 20, ODE, DVC, OOPSMP
  - Model Predictive Control (MPC) approaches
  - Nonlinear Trajectory Generation (NTG)
- What do we need for control?
  - Goal: MATLAB for mechanical systems

What Do We Need?

- Automatic first and second linearizations
- closed kinematic chains
- friction, including Coulomb effects
- impacts, both elastic, inelastic, and plastic (maintaining contact after collision)
  - automatic determination of distinction between plastic and nonplastic impacts
- We are willing to give up some speed for analysis, similar to MATLAB

Tuning Simulations

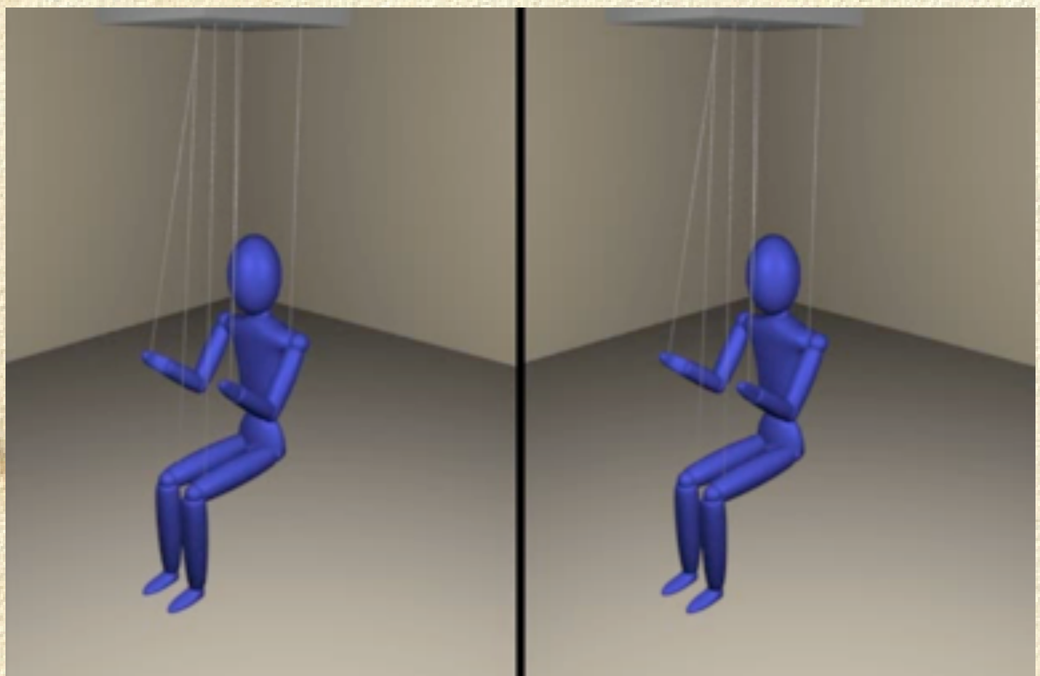
- Simulation methods often use artificial damping (explicitly or implicitly) to fix numerical instability
- System Identification leads to RHP poles when artificial stabilization is present
- Resulting control design is generally unstable



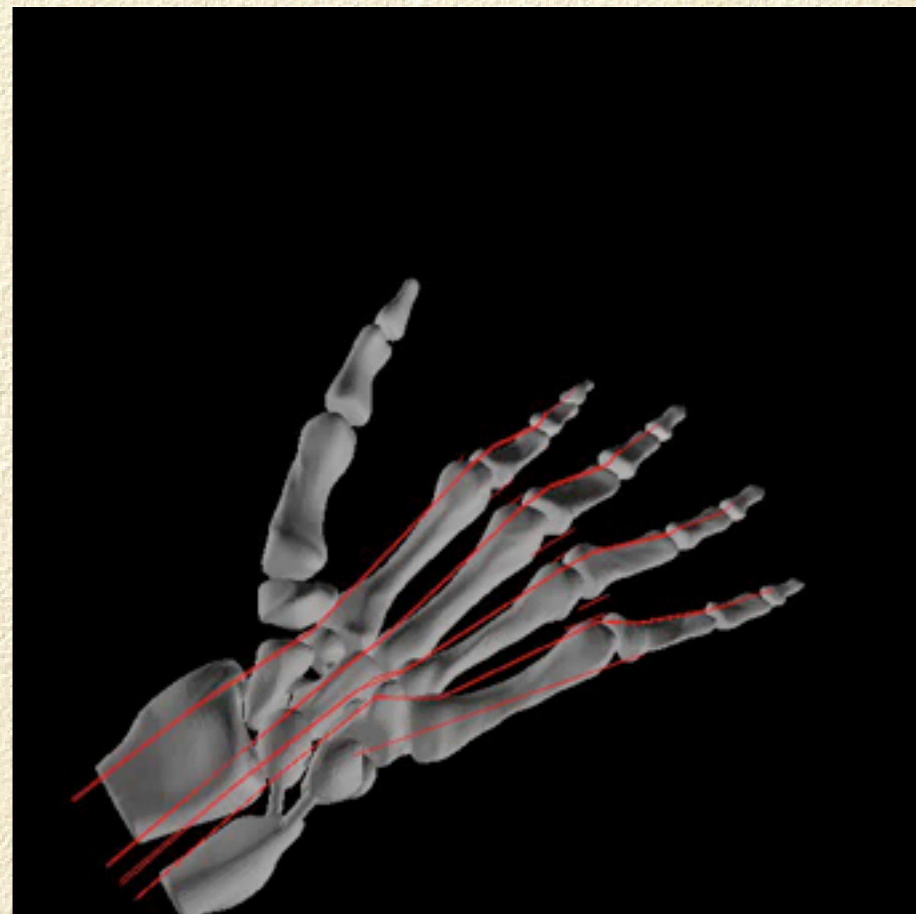
Intel simulation of a cup using ODE

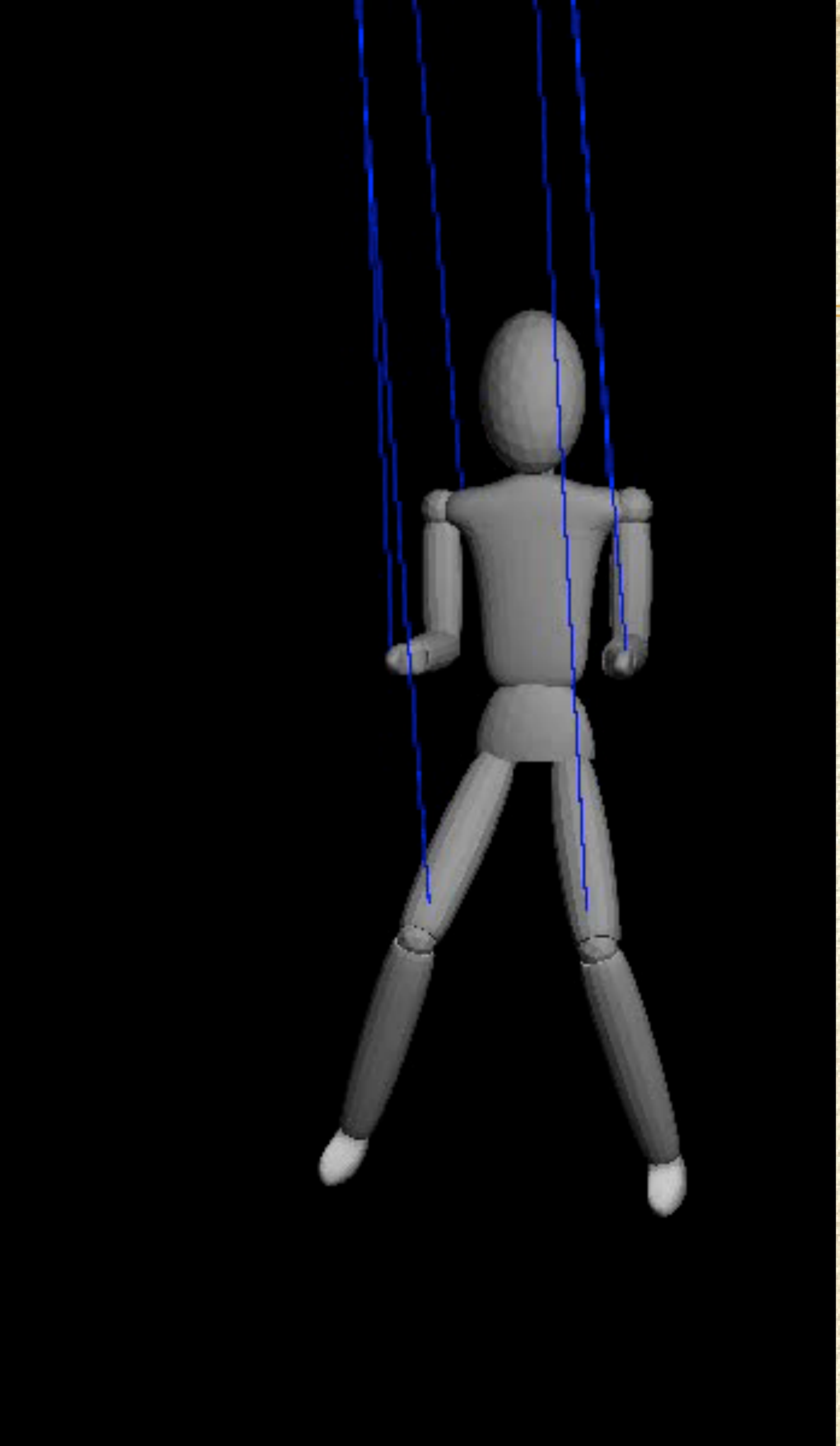
Tuning Simulations

How do we do simulations and calculate control laws for complex systems while avoiding simulation tuning?



plausible *not quite right!*





We use Variational Integrators

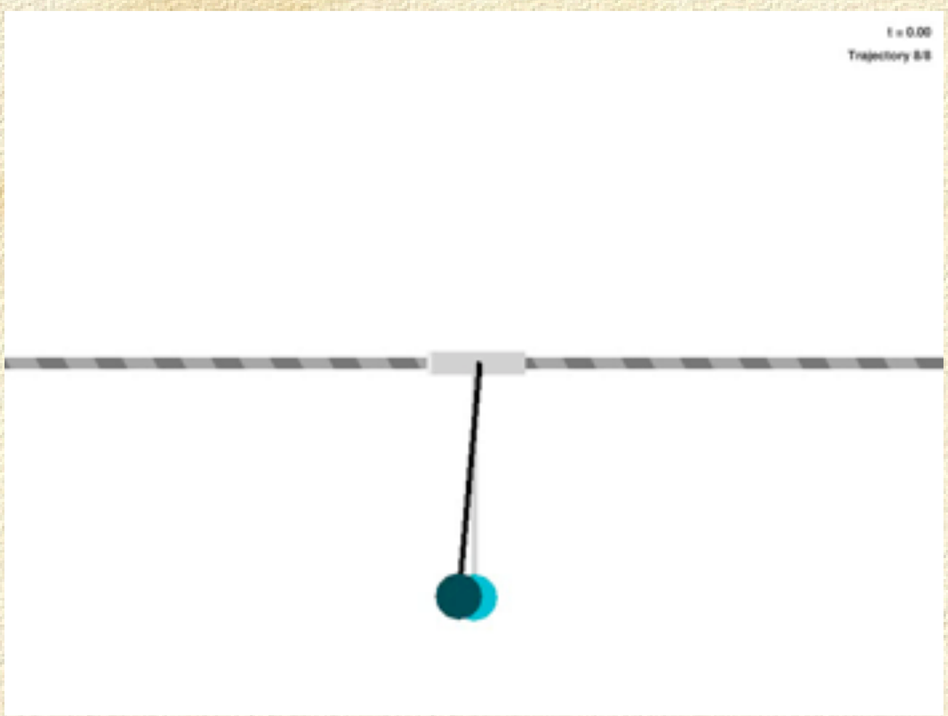
- Variational integrators are a particular choice of geometric DAE methods
- Five simulations, using time steps of 0.01, 0.02, 0.03, 0.05, and 0.10
- Constraints and other integrals of motion are maintained in the presence of nontrivial dynamics and external forcing
- Key thing--no tuning parameters except for dt (can be automated)

Beta available at *http://trep.sourceforge.net*

Graph-Based Linearization

- Based on the continuous mechanics or discrete variational mechanics, we can compute *exact* linearization for the discrete time system
  - no root solving required
  - no matrix exponentials as an approximation
- Same thing for second derivatives
- This includes constraints and external forcing
- Closed kinematic chains generate linearization that *exactly* preserves the constraints

Graph-Based Optimization



- Only a few iterations for each step
- The only specification is the graph
- All quantities are automatically calculated
- Quadratic convergence gives good scaling properties

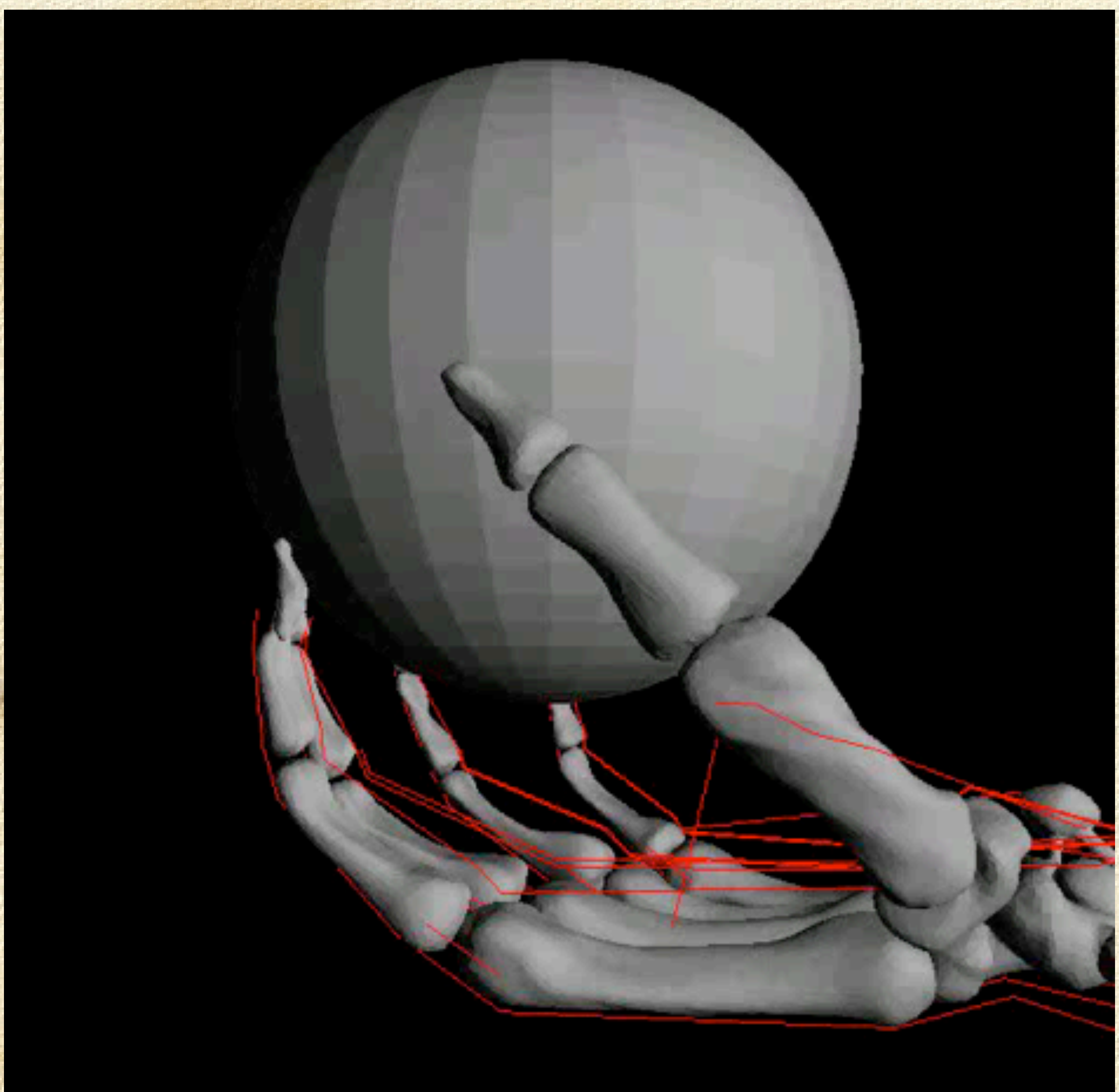
Optimal Control of the Hand

Halfway through optimization

Fully optimized

- Idea: Use "stable" interpretation of EMG signals to drive real prosthetic or FES stimulation

Constrained Hand



- Same calculations are stable for grasping
- Feedback control captures role of compliance in stability

Collisions

- Challenges in collision modeling include
  - distinguishing between plastic and nonplastic impacts
    - articulated bodies can experience plastic impacts even with completely elastic impacts
- A key result in variational integration is that there is a Modified Hamiltonian that the discrete updates exactly compute
  - this turns out to be very helpful in impact modeling

Collisions

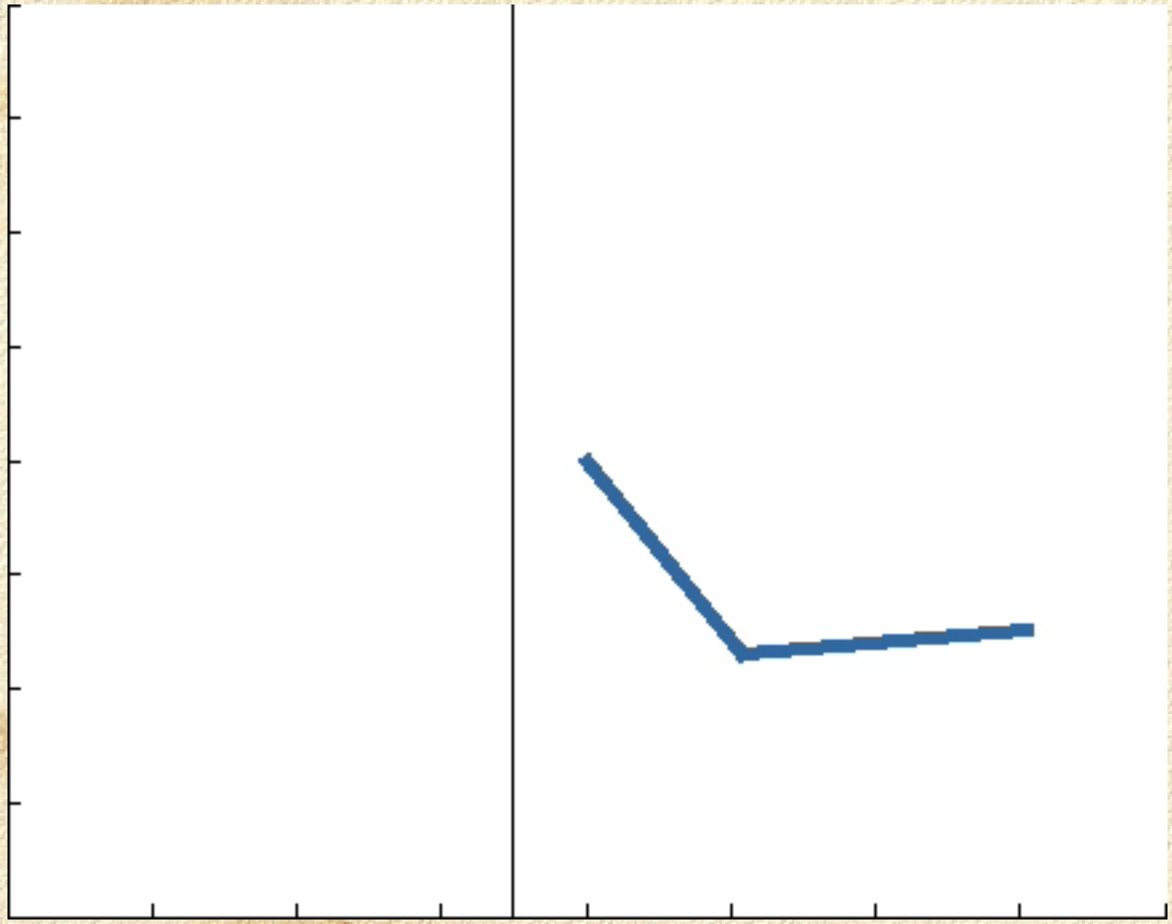
- Two common approaches
  - using continuous-time impact equations using an approximation of the velocity (CTEC)
  - discrete-time impact equations using momentum at an intermediate point (DTEC)
- Both can lead to nontrivial energy losses (10-15% in a single "reasonable" time step)
- We use the Modified Hamiltonian to update the impact equations
- Leads to much better energy and control calculations

Collisions

- Main differences between methods are along an accuracy versus complexity scale
- Main point: MH only needs to be evaluated during the impact map

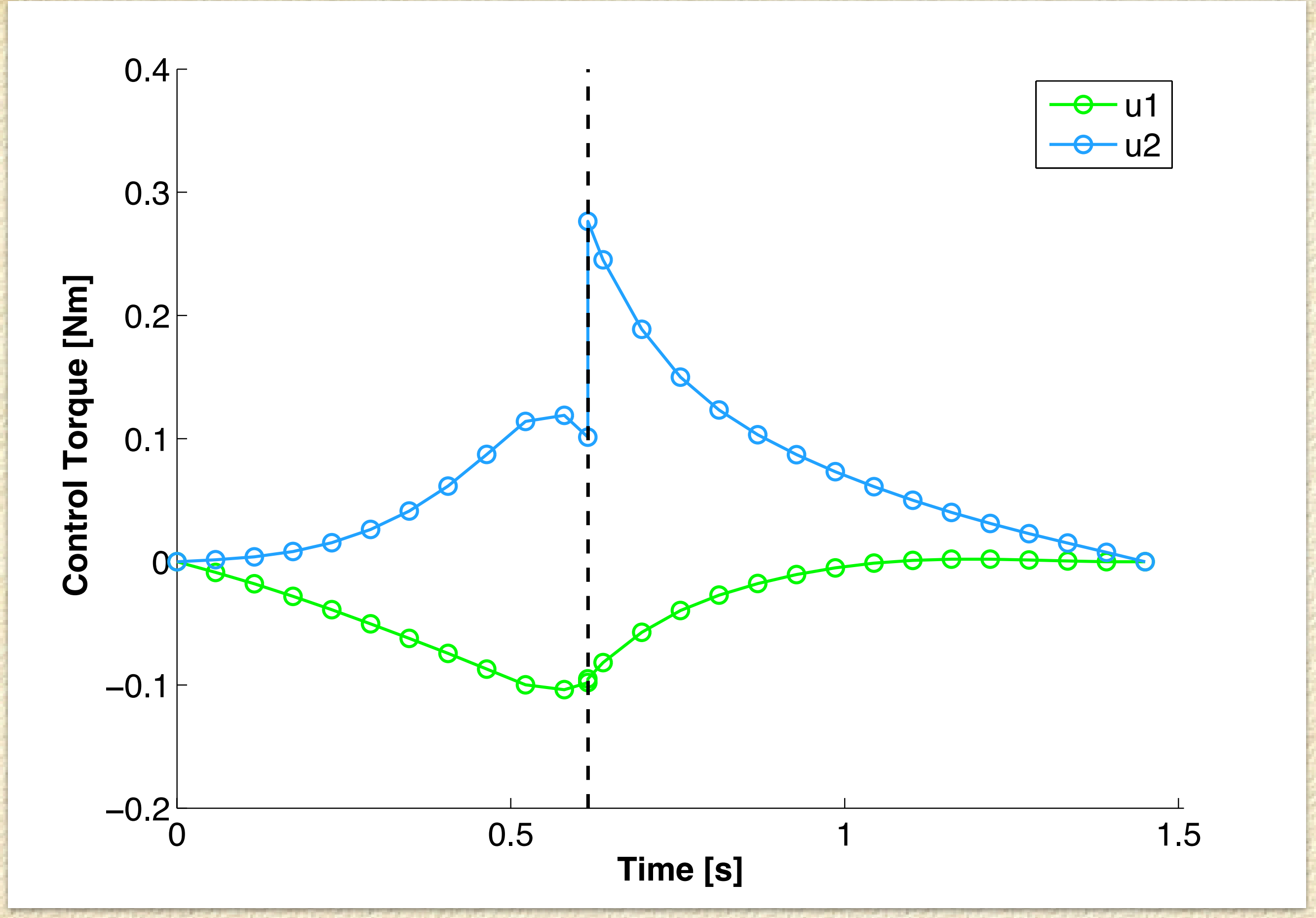
Method	CTEC	DTEC	MHC
Symplectic	No	Yes	Sometimes
Conservation of Momentum	Yes	Yes	Yes
Scalar Conserved Quantity	CE	DE	MH
Implementation	Easy	Hard	Harder
Impact map	Explicit	Implicit	Implicit
Computational cost	Low	Medium	Medium-High
Dense impact behavior	Good	Bad	Good
L^2 Error	Highest	Med	Lowest
Structured L^2 Convergence	Yes	No	Yes

Collisions for a Double Pendulum



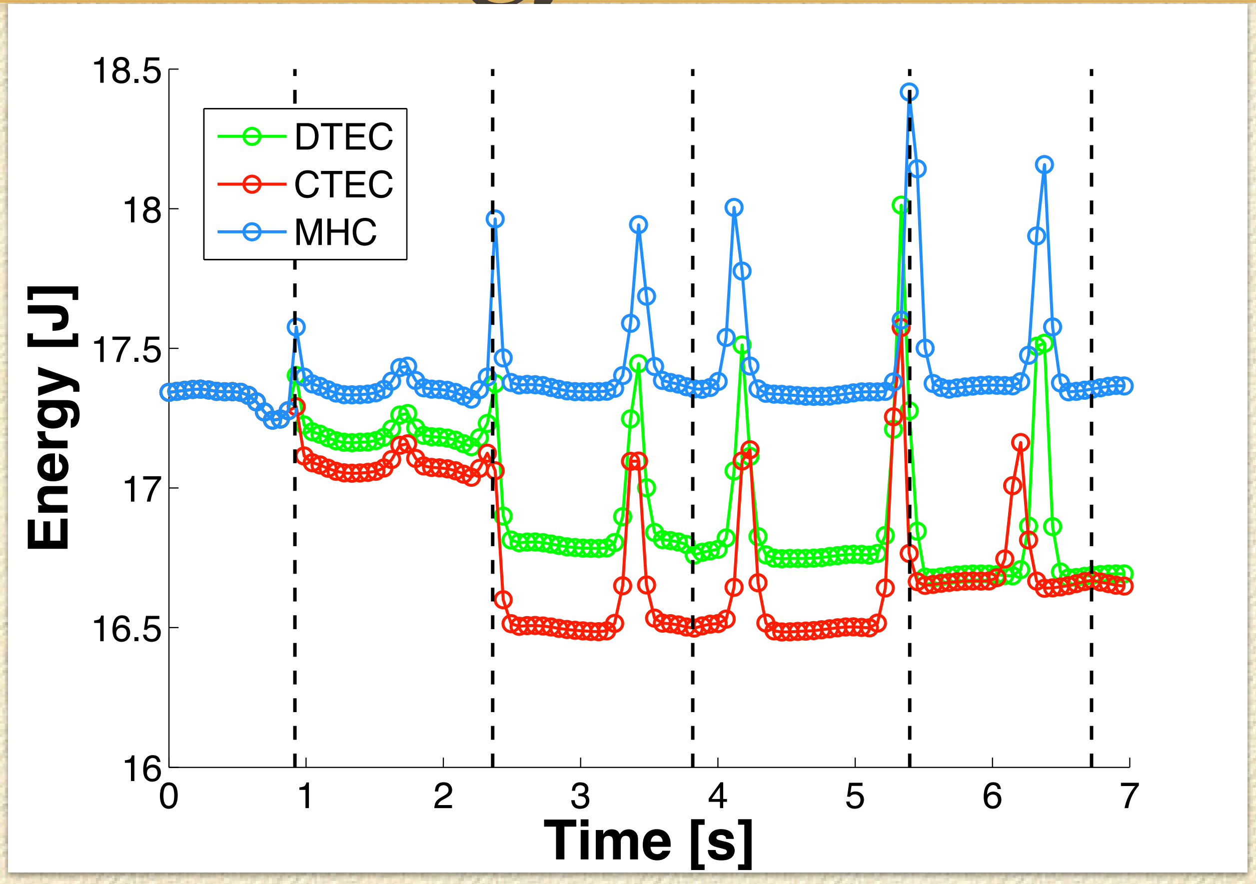
- We use a simple nonlinear double pendulum to test impact maps
- This system is surprisingly sensitive to choice of impact map

Control Calculations



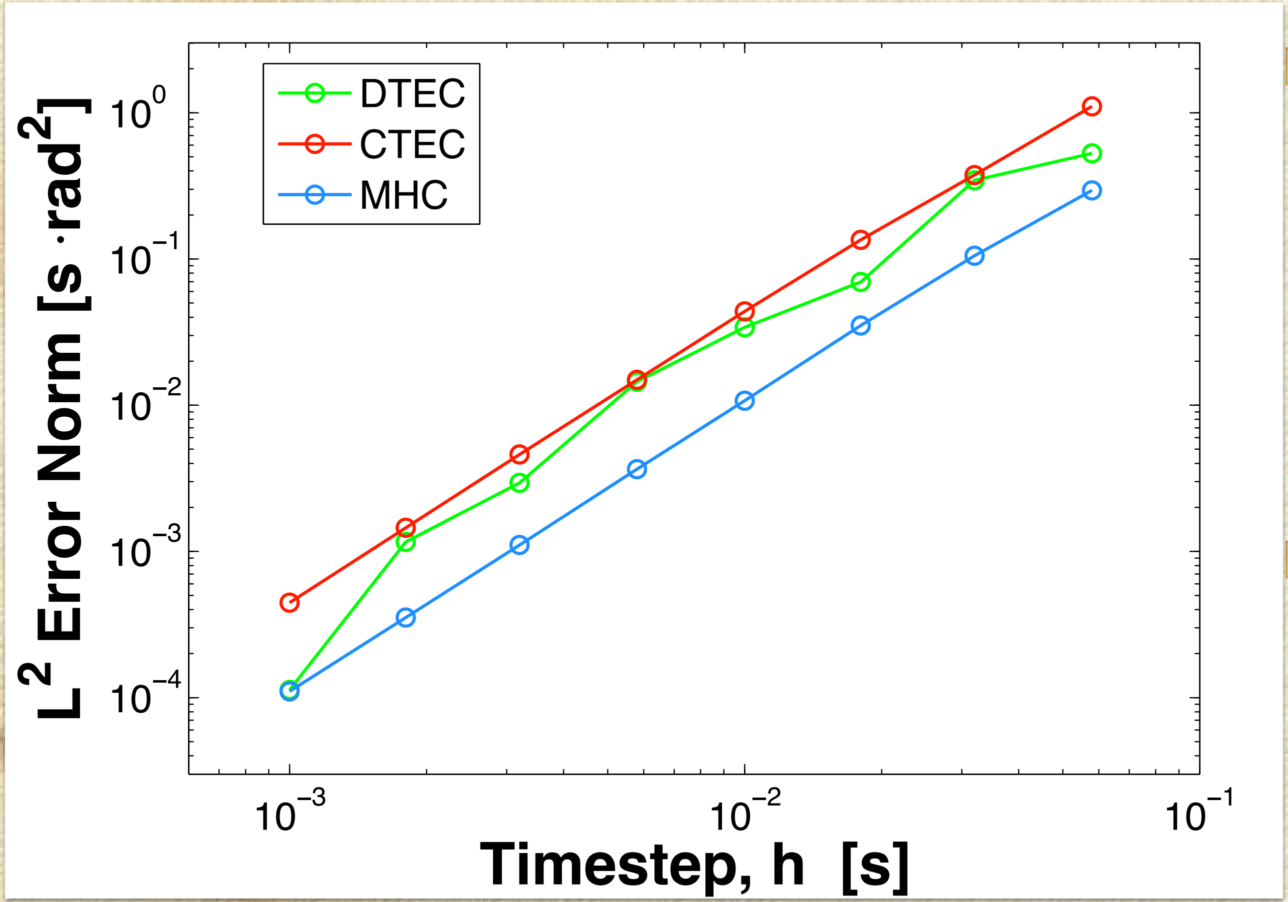
- Control calculations are dramatically affected by impact map.
- The "free dynamics" (optimal has u1=u2=0) become nontrivial due to bad representation of impact map

Energy Behavior



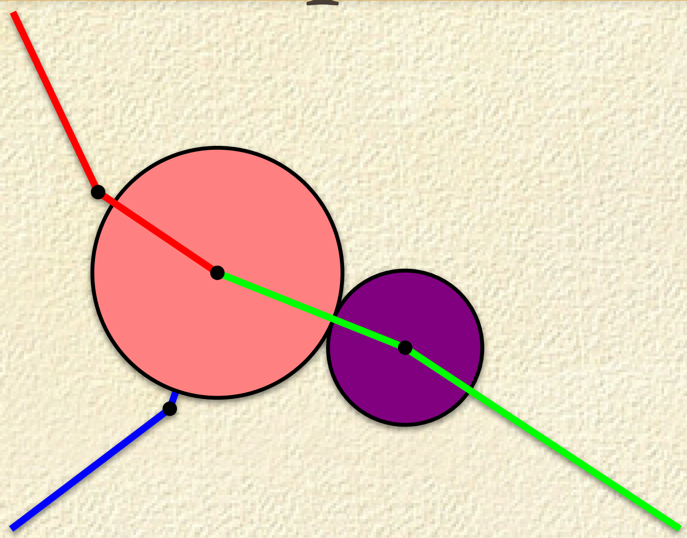
- The energy behavior is one way to explain why
- Each impact has a dramatic affect on the trajectory

What does updating just the impact with the MH buy us?



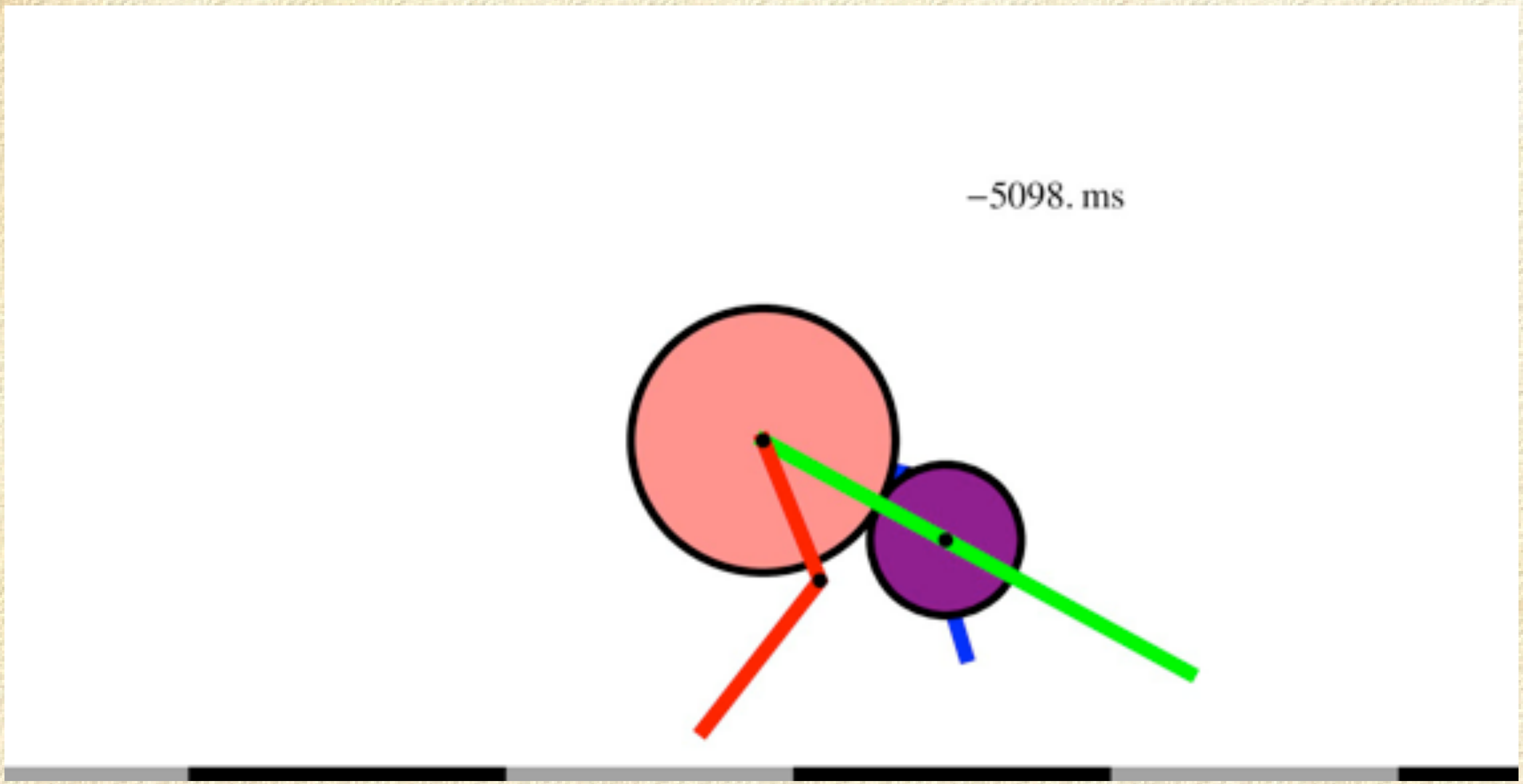
- Using the MH at impacts leads to nearly an order of magnitude improvement in accuracy
- For simulation this may or may not be worth the extra computation, but for control it improves physical meaning of simulation

Mechanical Design for Uniqueness



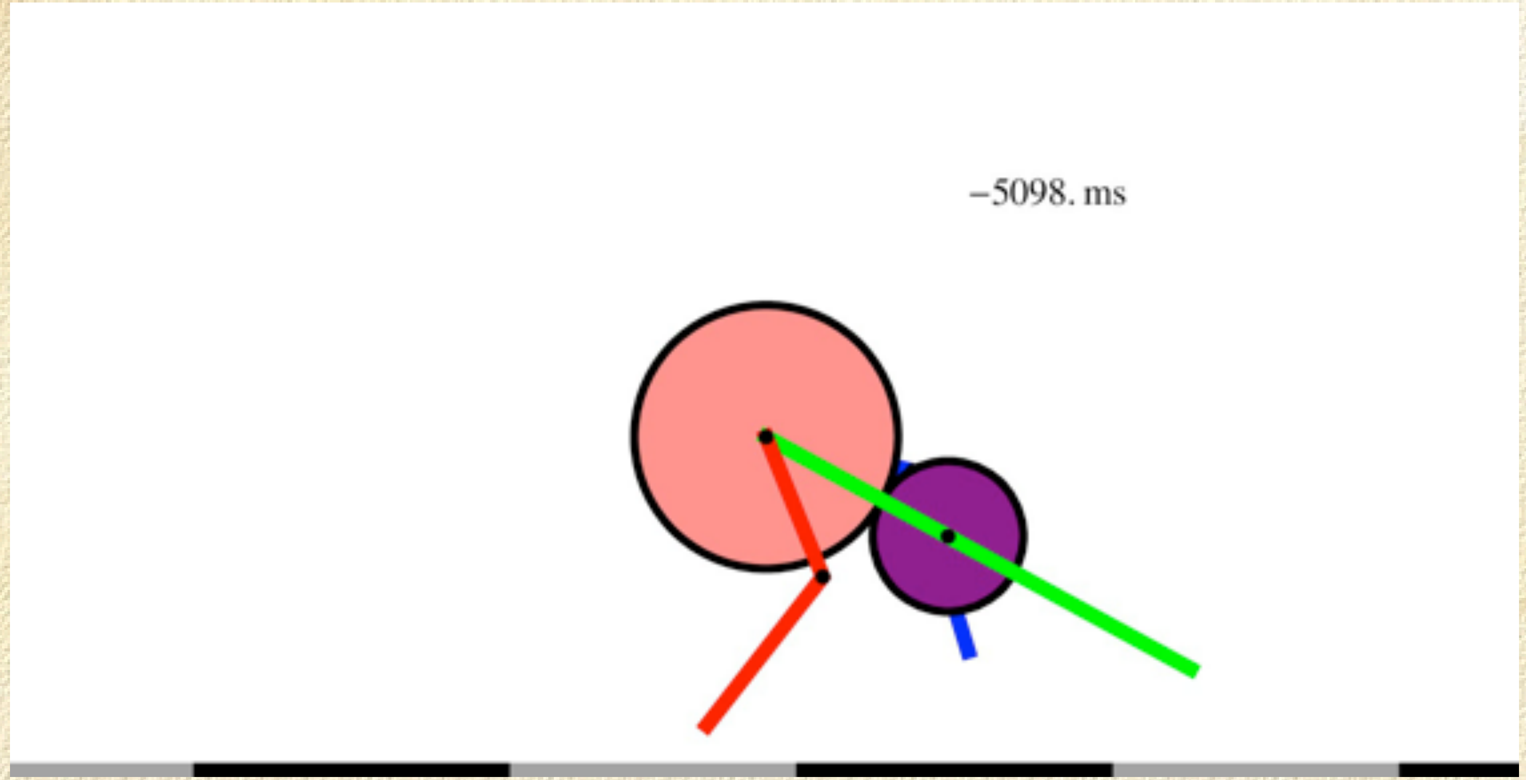
- We know that simultaneous impact need not have unique outcomes
- But some systems (e.g., Newton's cradle) do have unique impact maps when looking at variational analysis
- Can we design a system to have unique impacts?

Mechanical Design for Uniqueness



- Variational representation of simultaneous impact helps us obtain an algebraic conditions for uniqueness
- By optimizing this quantity over mechanical and gait parameters, we obtain unique impact outcomes

Mechanical Design for Uniqueness



- This simulation is unique--nowhere is there a choice being made between solutions

Conclusions

- Software needs:
  - First and second variations of discrete trajectories, computed exactly
  - Compatibility with other software (OMPL for planning, ROS for implementation, SNOPT for optimization comparisons) for integration and comparison
  - Representations of contact that have known properties with respect to error, convergence, conserved quantities, and effect on planning
- Our *trep* software provides many of these capabilities for forced, constrained systems, but we are only now getting to the point of implementing collisions

Support and Thanks

- Magnus Egerstedt, Jerrold Marsden, Wendy Murray
- Atlanta Center for Puppetry Arts
- Lanny Smoot/Disney Imagineering
- Graduate Students: Tim Caldwell, Elizabeth Jochum, **Elliot Johnson**, Lauren Miller, Jarvis Schultz, **Vlad Seghete**, Krissy Snyder, Matthew Travers, Andrew Wilson
- Undergraduates: Corrina Gibson, Matanya Horowitz, Kirk Nichols, and Tasnim Tanveer
- Postdocs: **David Pekarek** and Benjamin Tovar