END-TO-END DIFFERENTIABLE PHYSICS FOR LEARNING AND CONTROL

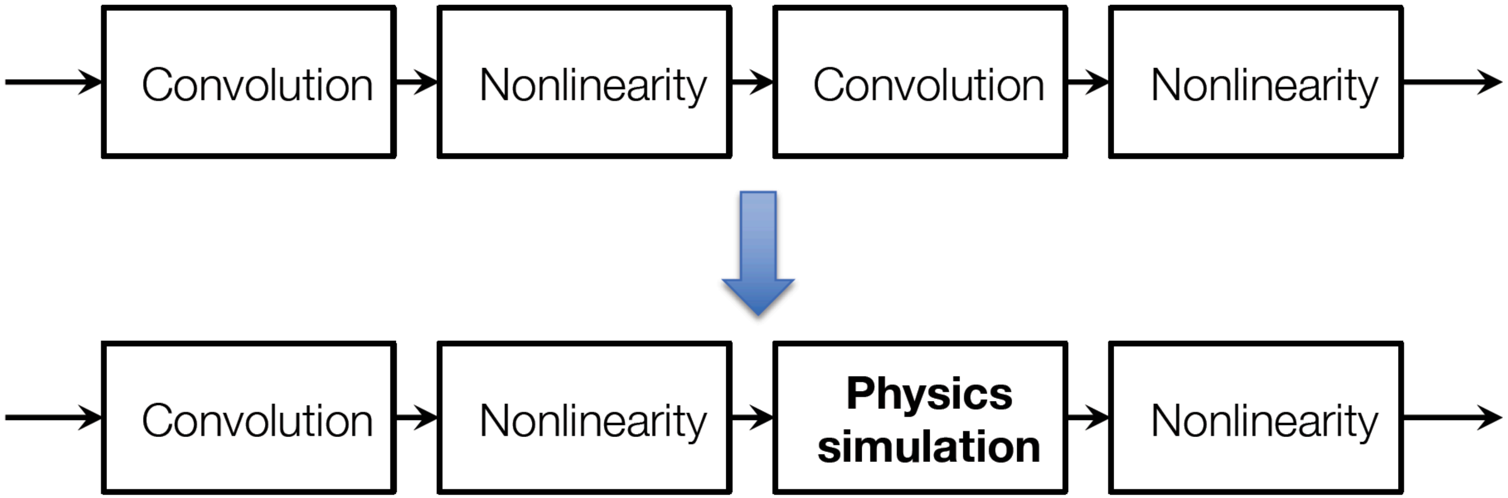
Filipe de Avila Belbute-Peres[1]

Kevin Smith[2] Kelsey Allen[2] Joshua Tenenbaum[2] Zico Kolter[1,3]

[1]School of Computer Science, Carnegie Mellon University
[2]Brain and Cognitive Sciences, Massachusetts Institute of Technology
[3]Bosch Center for Artificial Intelligence

MOTIVATION

Embed structured physics knowledge as a module in a larger end-to-end system



Requires the physics engine to be differentiable

PREVIOUS WORK

Others have done similar work in developing differentiable physics engines

- Automatic differentiation: Degrave, Hermans, Dambre, Wyffels, 2017.
  A *Differentiable Physics Engine for Deep Learning in Robotics*
- Numerical gradients: Todorov, Erez, Tassa, 2012.
  MuJoCo: A physics engine for model-based control
- Neural network-based: Battaglia et al., 2016; Chang et al., 2016; Lerer et al., 2016.

We formulate a physics engine that provides the analytical gradients in closed form

A DIFFERENTIABLE PHYSICS ENGINE IN 3 STEPS

1. Express equations of motion as an LCP

Discrete time approximation to Newtonian dynamics

$$\mathcal{M}\dot{v} = f^{(c)} + f \Rightarrow \mathcal{M}(v_{t+dt} - v_t) = dt f_t^{(c)} + dt f_t$$

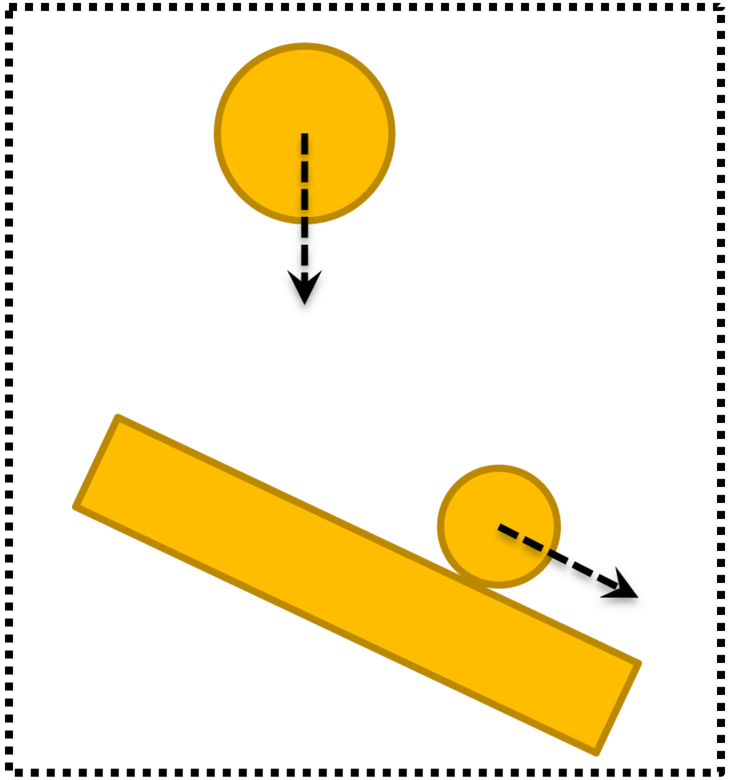
Add rigid body constraints to achieve LCP formulation

$$\mathcal{J}_e \lambda_e = 0$$ Equality constraints

$$(\lambda_c, \mathcal{J}_c v + c) \in \mathcal{C}$$ Contact constraints

$$(\lambda_f, \mathcal{J}_f v + E\gamma) \in \mathcal{C}$$
$$(\mu\lambda_c - E^T \lambda_f, \gamma) \in \mathcal{C}$$
Friction constraints

where $\mathcal{C}(a,b) = \{a \geq 0, b \geq 0, a^T b = 0\}$



[e.g. Anitescu and Potra, 1997]

A DIFFERENTIABLE PHYSICS ENGINE IN 3 STEPS

2. Differentiate optimality conditions of LCP

Optimality conditions for LCP can be written compactly as

$$\begin{aligned} \mathcal{M}x + A^T y + G^T z + q &= 0 \\ Ax &= 0 \\ Gx + Fz + s &= m \\ s \geq 0,\ \ z \geq 0,\ \ s^T z &\geq 0. \end{aligned}$$

Take matrix differentials

$$\begin{aligned} \mathrm{d}\mathcal{M}x + \mathcal{M}\mathrm{d}x + \mathrm{d}A^T y + A^T \mathrm{d}y + \mathrm{d}G^T z + G^T \mathrm{d}z + \mathrm{d}q &= 0 \\ \mathrm{d}Ax + A\mathrm{d}x &= 0 \\ \mathrm{d}z \circ (Gx + Fz - m) + z \circ (\mathrm{d}Gx + G\mathrm{d}x + \mathrm{d}Fz + F\mathrm{d}z - \mathrm{d}m) &= 0 \end{aligned}$$

Linear equations in unknowns (dx, dy, dz), simple to solve for desired differentials

[e.g. Magnus and Neudecker, “Matrix differential calculus”, 1988]

A DIFFERENTIABLE PHYSICS ENGINE IN 3 STEPS

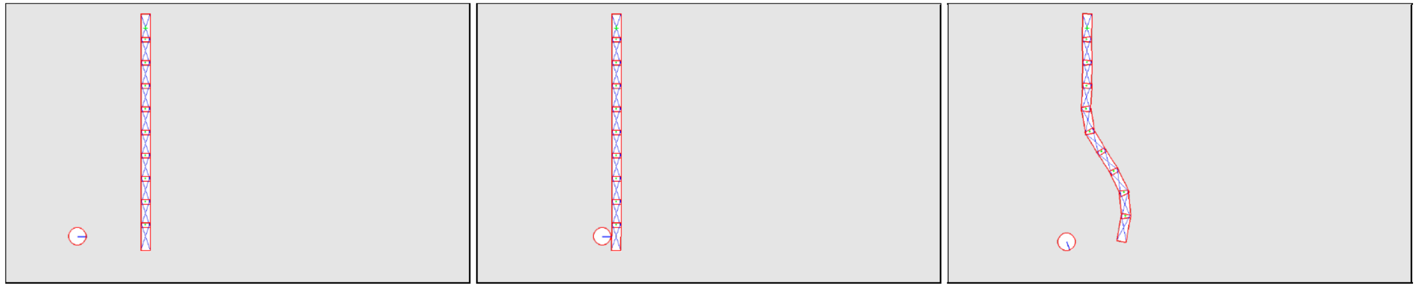
3. Efficiently compute backprop

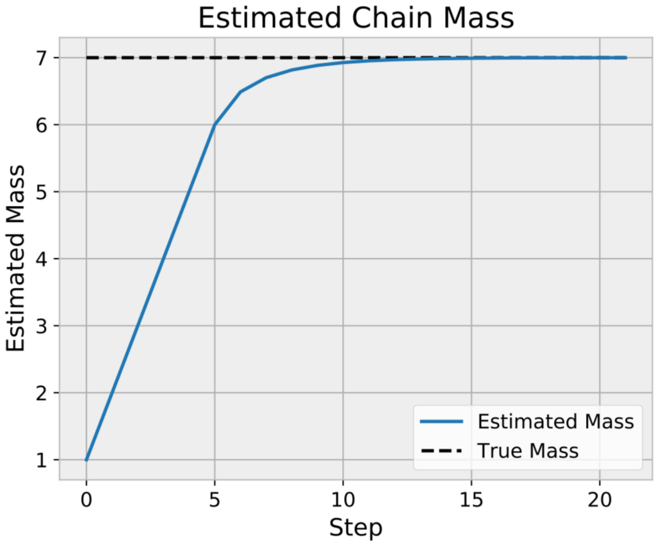
- Since we have already solved the LCP, we can compute the backward pass with just one additional solve based upon the LU-factorization of the LCP matrix

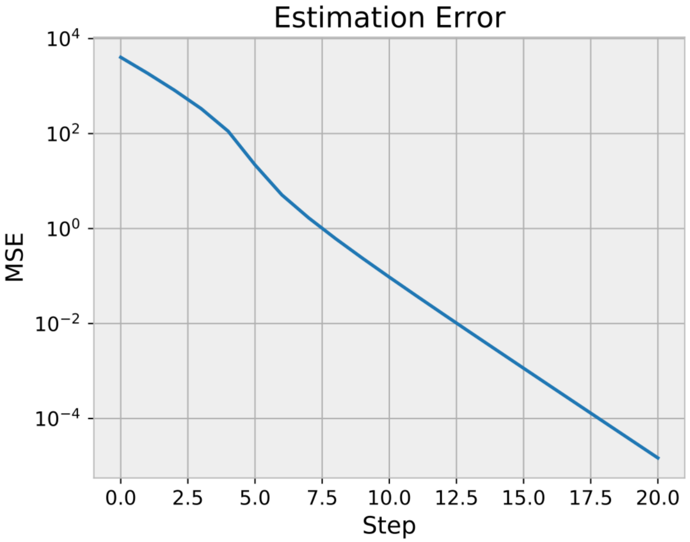
- We can effectively differentiate through the simulation at no additional cost to just running the simulation itself

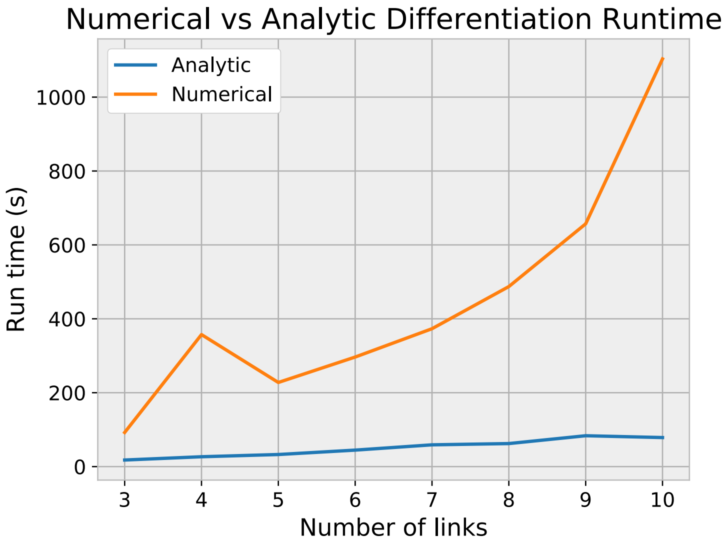
SYSTEM IDENTIFICATION

Learn mass of chain after observing collision



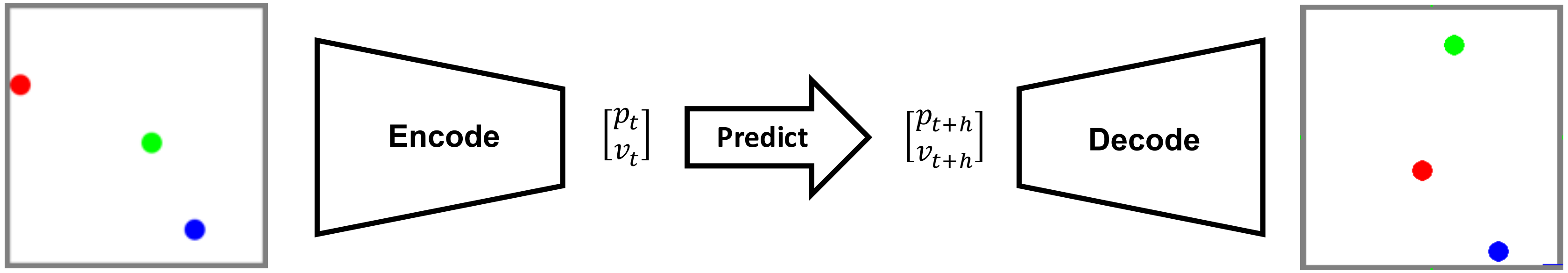




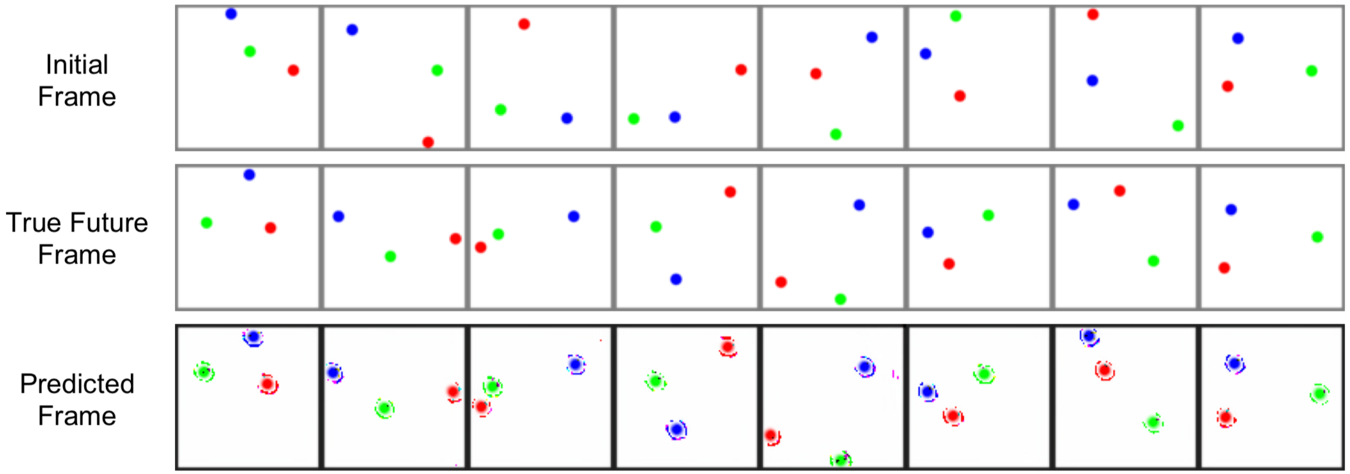


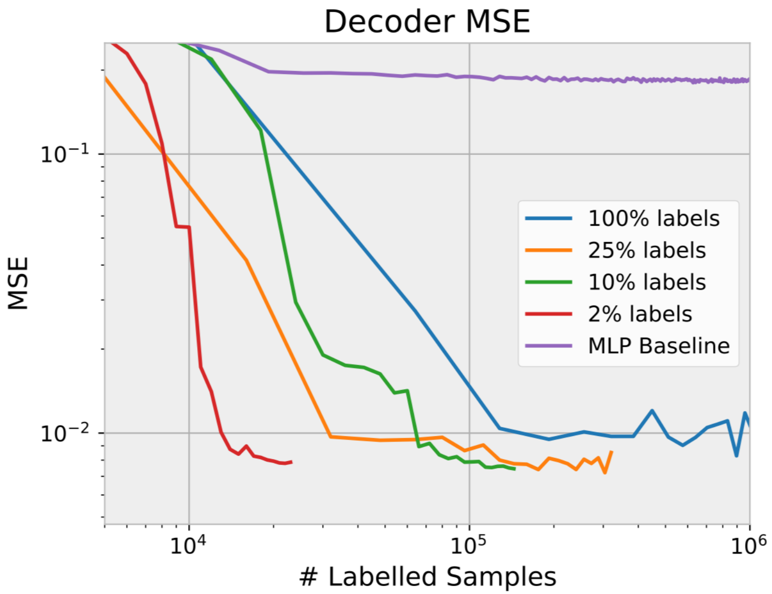
SIMULATION FOR VISUAL DYNAMICS

Predict evolution of simulated billiard balls from visual images

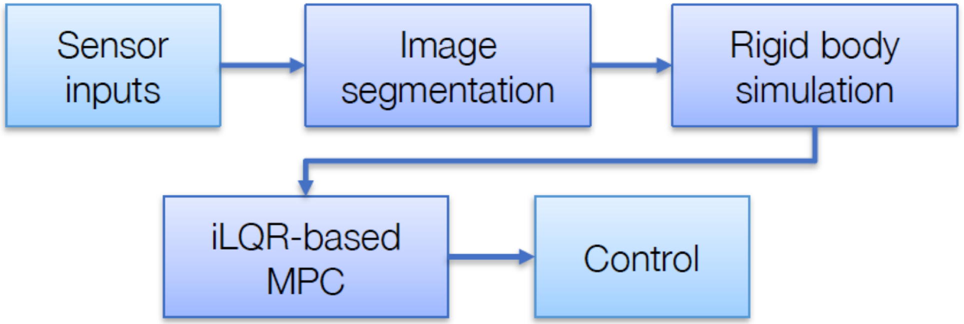


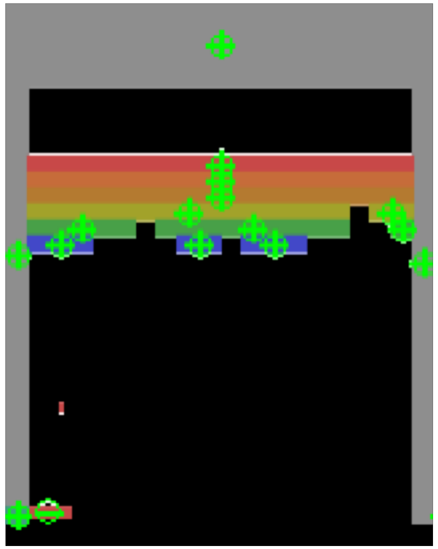
Substantially better performance and data efficiency by integrating physics engine

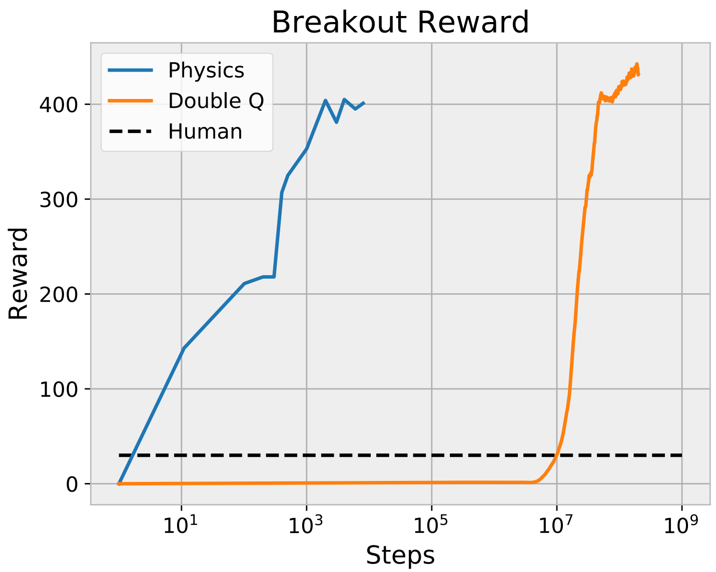




MODEL-BASED CONTROL



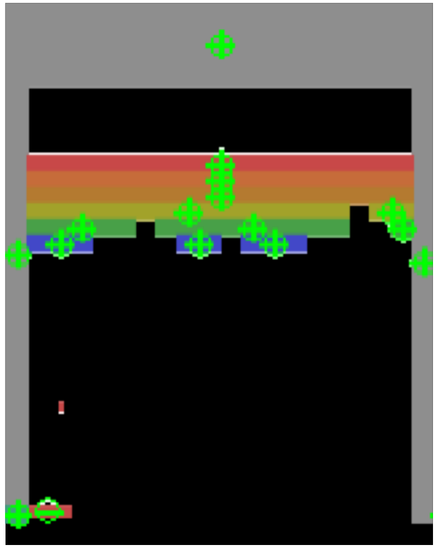




- Parameters learned from data
- iLQR used for control with the differentiable model

SUMMARY

Integrating structured constraints such as physical simulation into machine learning is a promising direction for more efficient learning.



Poster #38

Code at https://github.com/locuslab/lcp-physics

Thank you!