COL865: Special Topics in Computer Applications

Physics-Based Animation

Paper discussions:
Selle et al. [2008],
Liu et al. [2013]

Discussion format

Short **5-minute intro** by lead

- Define the problem
- Describe the key ideas
- Compare with methods discussed in class
- Show some results

Class discussion

- Share questions, comments, new ideas
- Respond to each others' questions

Selle et al. [2008]:
“A Mass Spring Model for Hair Simulation”

The problem

Simulating hair and other elastic strands

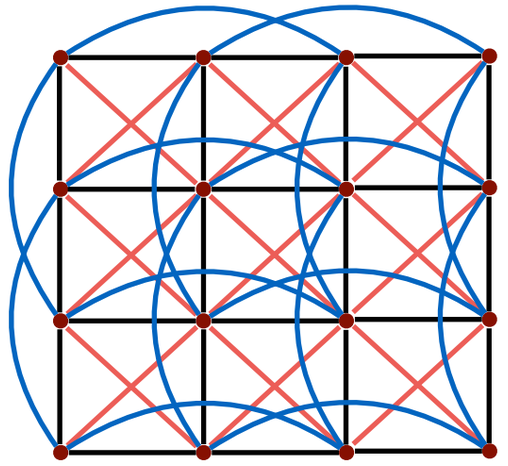
- How to model **twisting** behaviour?
- How to make **curly hair** remain curly?

Lots of other approaches, but mass-spring systems are simple and fast

Context

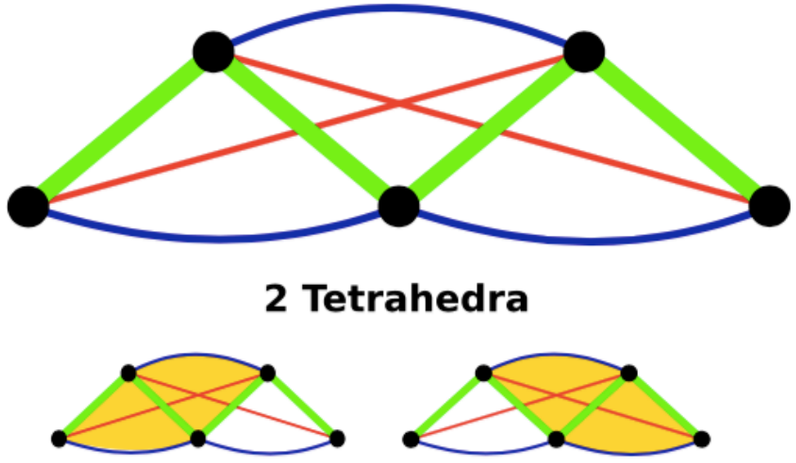
Mass-spring system for cloth:

- Various springs to model different deformation behaviours (structural, shear, bending)



This paper:

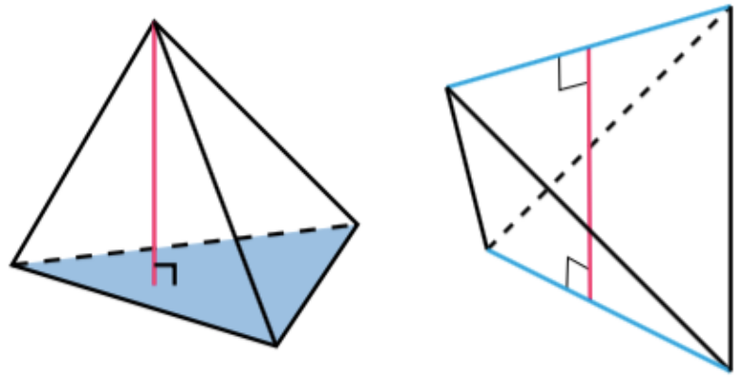
- Similar springs for hair: structural, bending, **torsion**
- New **altitude springs** to preserve shape



Key ideas

How to make spring models robust to collapse?

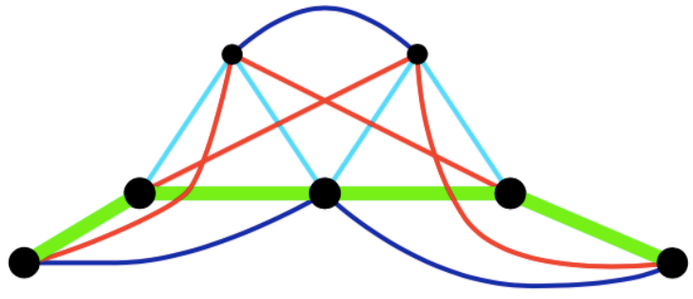
- Traditional spring-mass model once flattened will be stuck



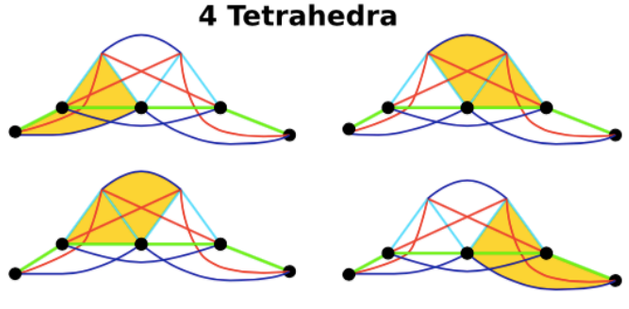
- Altitude springs to make tets “solid”

Create tets on consecutive curly segments

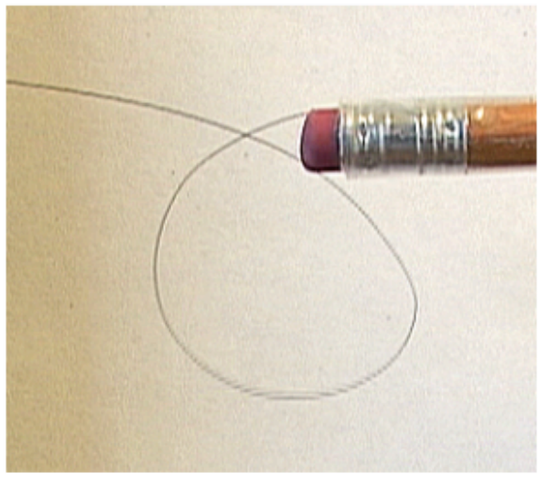
- Keeps helices shaped as helices



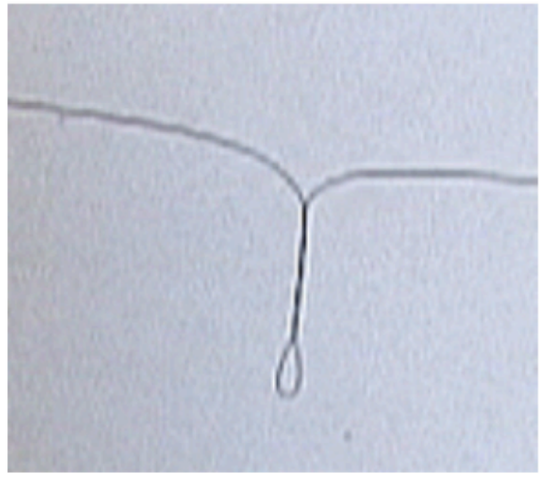
- Can’t make tets on straight segments? Add “fins” to each edge



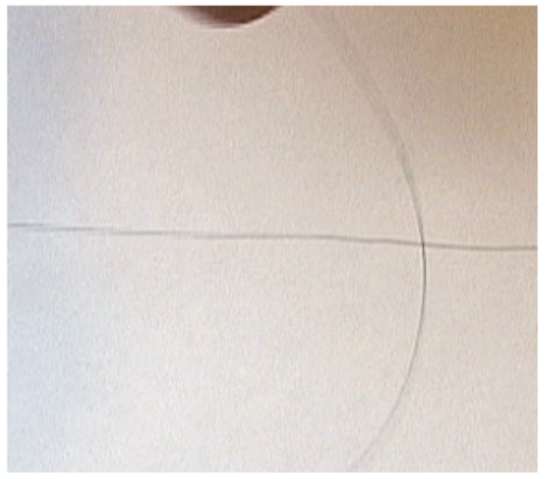
Results



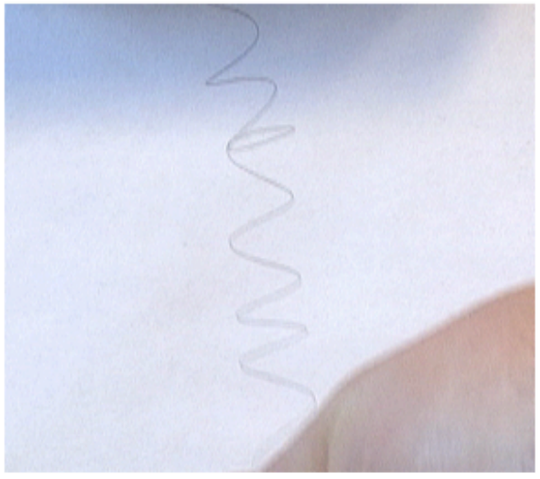
Buckling



Twisting



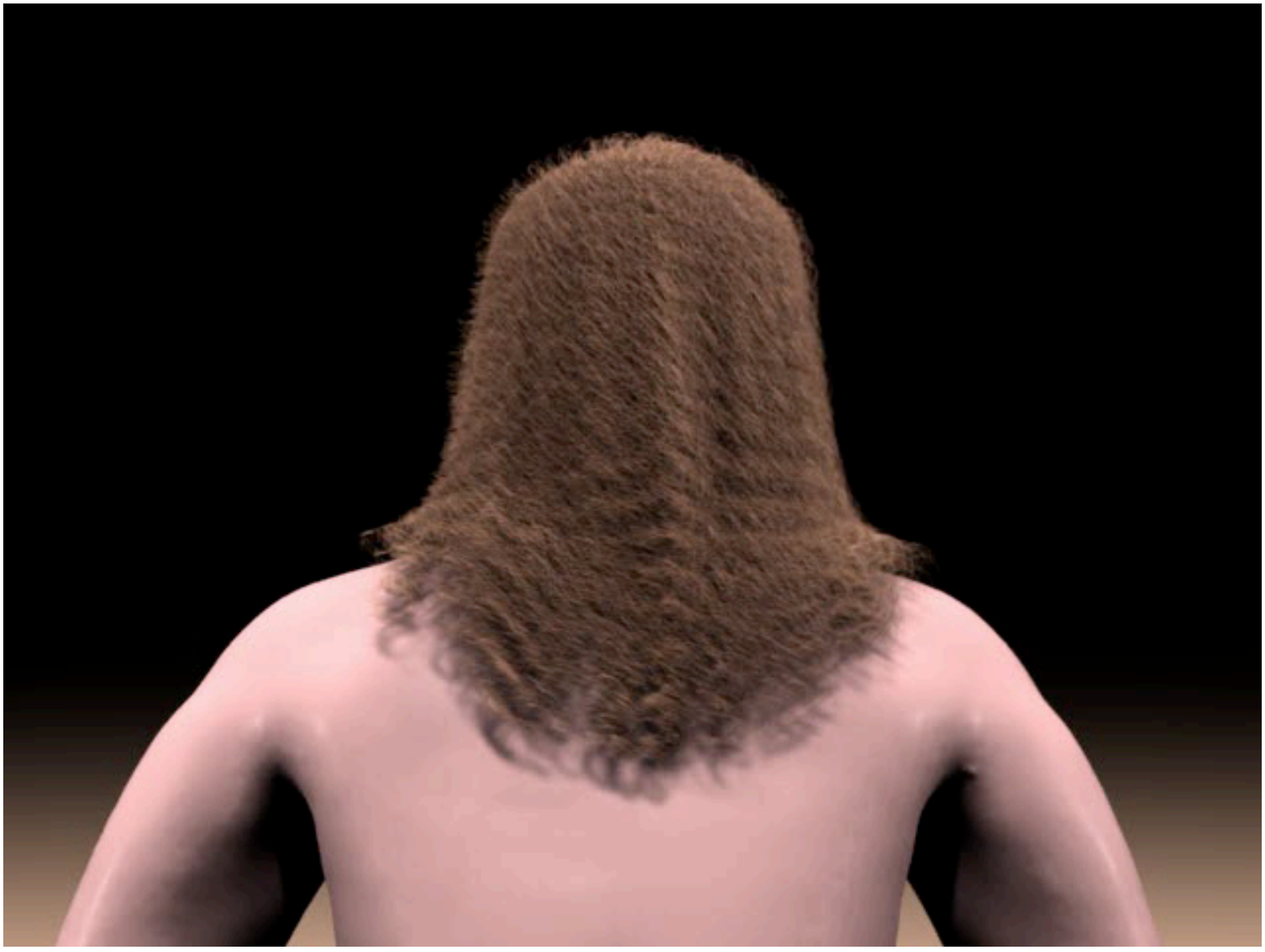
Stickiness



Curly hair

Results

Results



Liu et al. [2013]:
“Fast Simulation of Mass Spring Systems”

The problem

Implicit integration (e.g. backward Euler) has guaranteed stability, but requires solving large system of equations

$$\mathbf{x}^1 = \mathbf{x}^0 + \mathbf{v}^1 \, \Delta t$$
$$\mathbf{v}^1 = \mathbf{v}^0 + \mathbf{M}^{-1} \, \mathbf{f}(\mathbf{x}^1, \mathbf{v}^1) \, \Delta t$$

Newton's method:

$$(\mathbf{M} - \mathbf{J_x} \, \Delta t^2 - \mathbf{J_v} \, \Delta t) \, \Delta \mathbf{v} = (\mathbf{f}^0 + \mathbf{J_x} \, \mathbf{v}^0 \, \Delta t) \, \Delta t$$

- LHS matrix changes every time step
- System may be indefinite if Δt is large

Context

If $\mathbf{f}(\mathbf{x}) = -\nabla U(\mathbf{x})$, backward Euler is equivalent to optimization:

$$\mathbf{M}(\mathbf{x}^1 - \underbrace{\mathbf{x}^0 - \mathbf{v}^0\,\Delta t}_{\mathbf{y}}) = \mathbf{f}(\mathbf{x}^1, \mathbf{v}^1)\,\Delta t^2$$

$$\Rightarrow \quad \min_{\mathbf{x}} \tfrac{1}{2}\,(\mathbf{x}^1 - \mathbf{y})^{\mathsf{T}}\mathbf{M}(\mathbf{x}^1 - \mathbf{y}) + U(\mathbf{x}^1)\,\Delta t^2$$

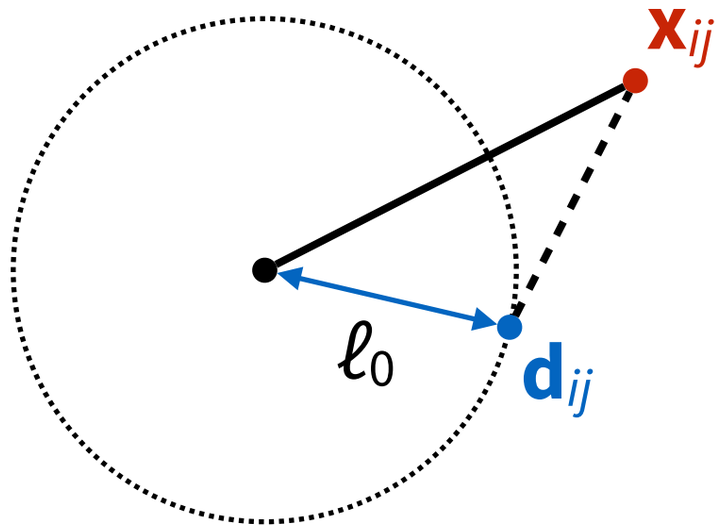
Solving with Newton's method = previous approach, but more robust (can do line search to ensure convergence)

But optimization form lets you do other tricks too…

Key ideas

For springs,

$$U_{ij} = \tfrac{1}{2}\, k_s\, (\|\mathbf{x}_{ij}\| - \ell_0)^2$$
$$= \min_{\mathbf{d}_{ij}} \tfrac{1}{2}\, k_s\, \|\mathbf{x}_{ij} - \mathbf{d}_{ij}\|^2$$



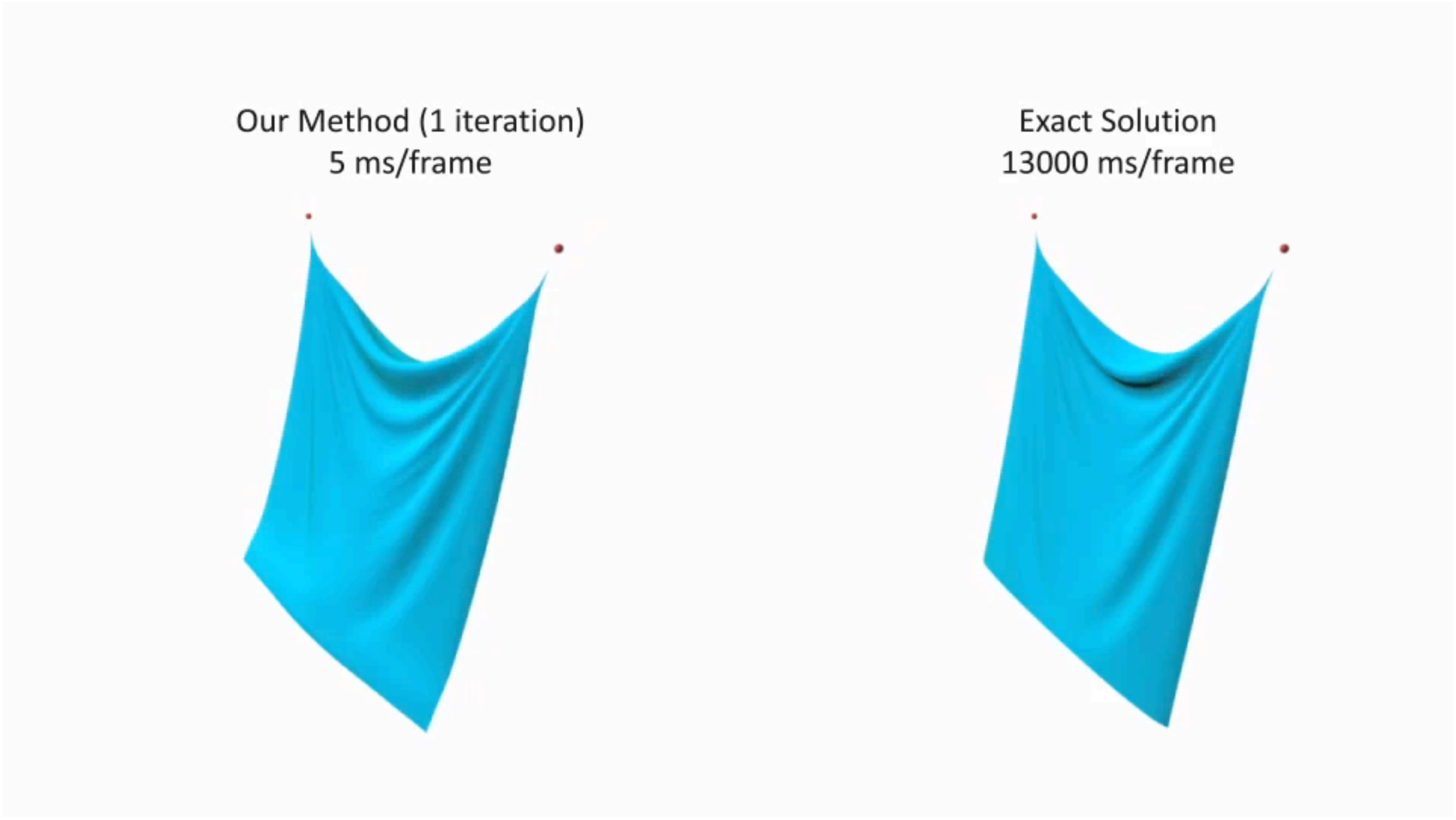
Therefore,

$$\min_{\mathbf{x}} \tfrac{1}{2}\, (\mathbf{x} - \mathbf{y})^{\mathsf{T}} \mathbf{M} (\mathbf{x} - \mathbf{y}) + U(\mathbf{x})\, \Delta t^2$$
$$\Leftrightarrow \quad \min_{\mathbf{x}} \min_{\mathbf{d}} \tfrac{1}{2}\, (\mathbf{x} - \mathbf{y})^{\mathsf{T}} \mathbf{M} (\mathbf{x} - \mathbf{y}) + U(\mathbf{x}, \mathbf{d})\, \Delta t^2$$

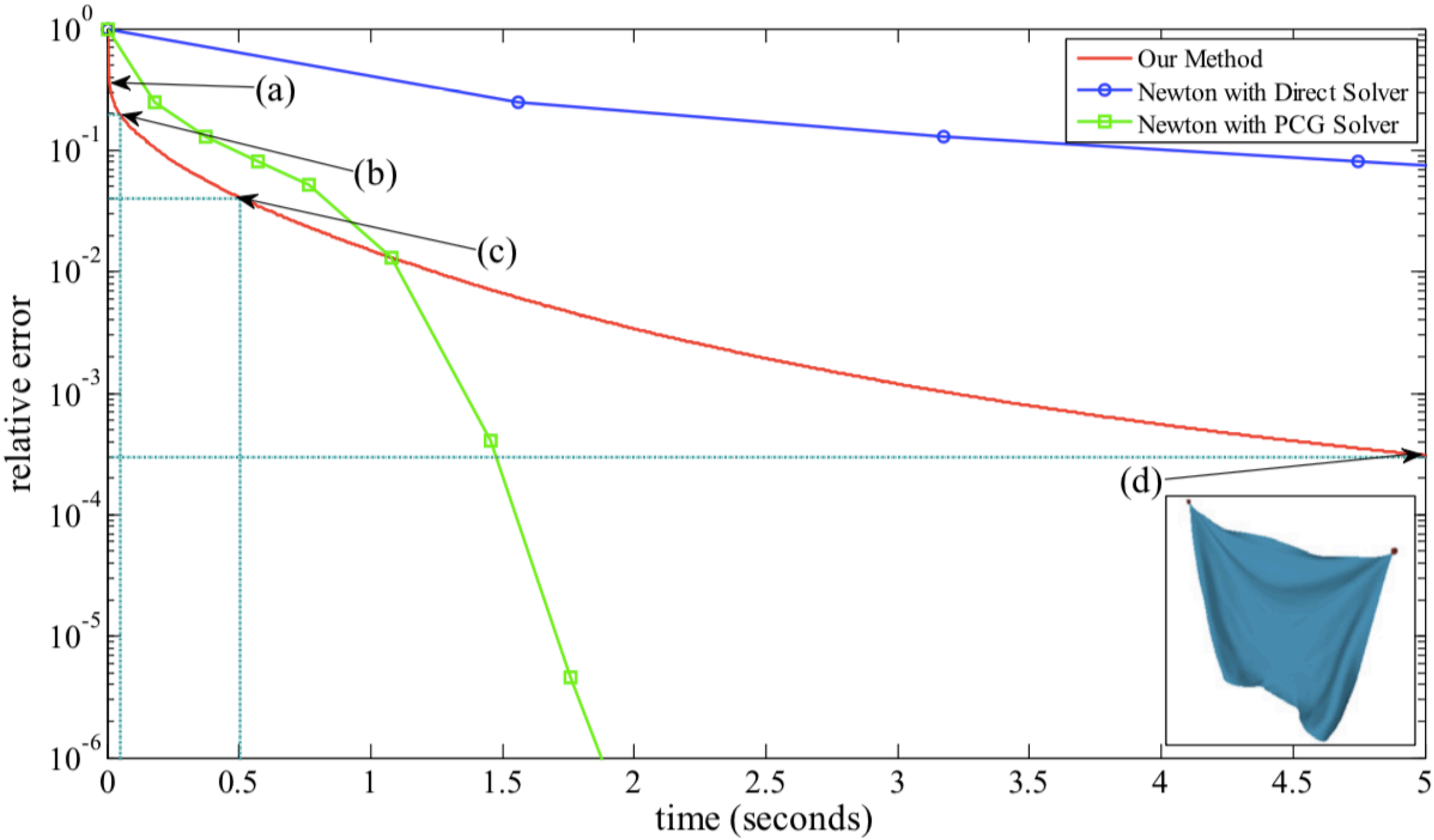
Minimize over $\mathbf{x}$ then over $\mathbf{d}$ alternately

- Min over $\mathbf{x}$ = linear solve with *fixed* matrix
- Min over $\mathbf{d}$ = trivial rescaling to ℓ_0

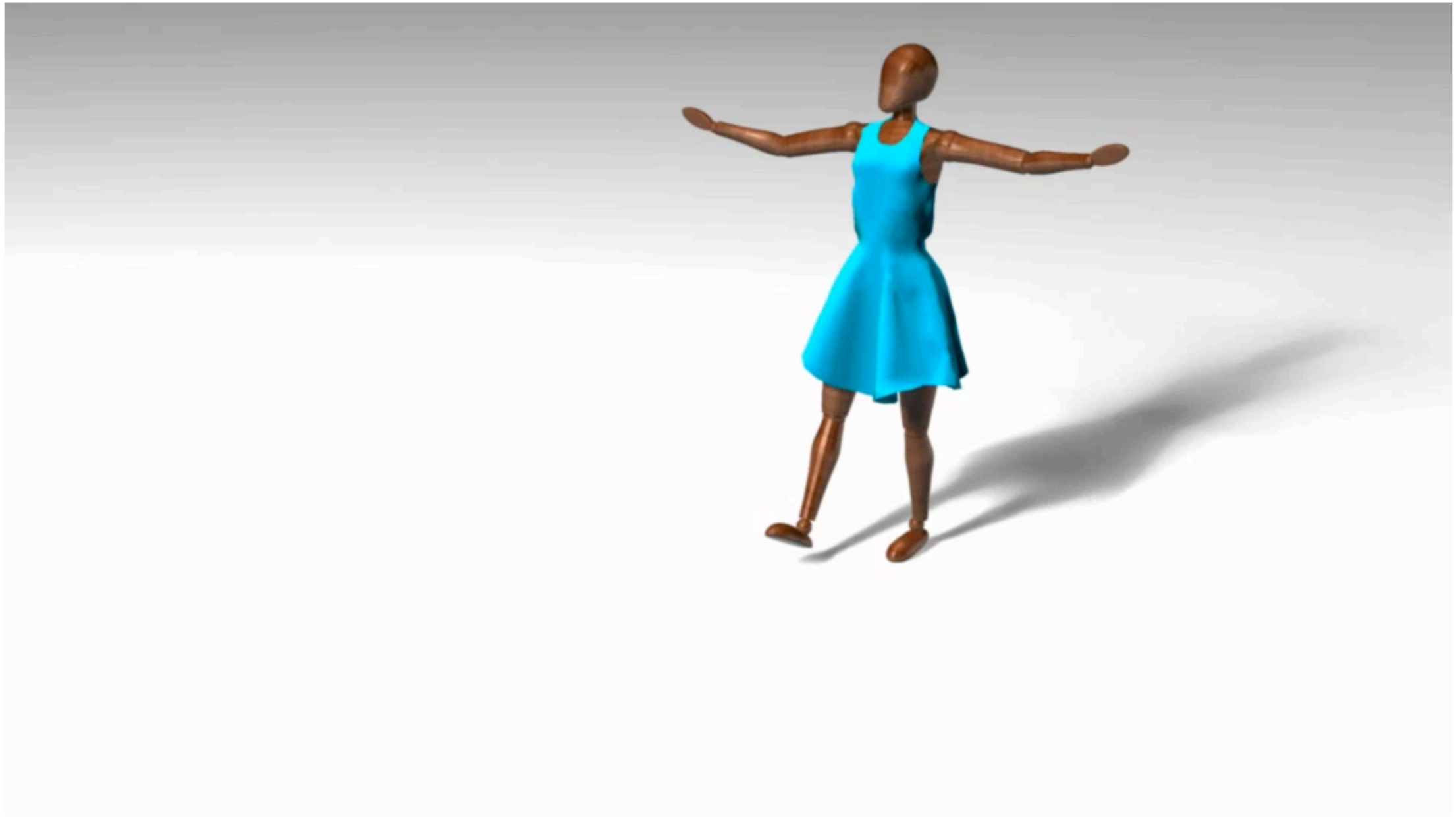
Results


Our Method (1 iteration)
5 ms/frame
Exact Solution
13000 ms/frame

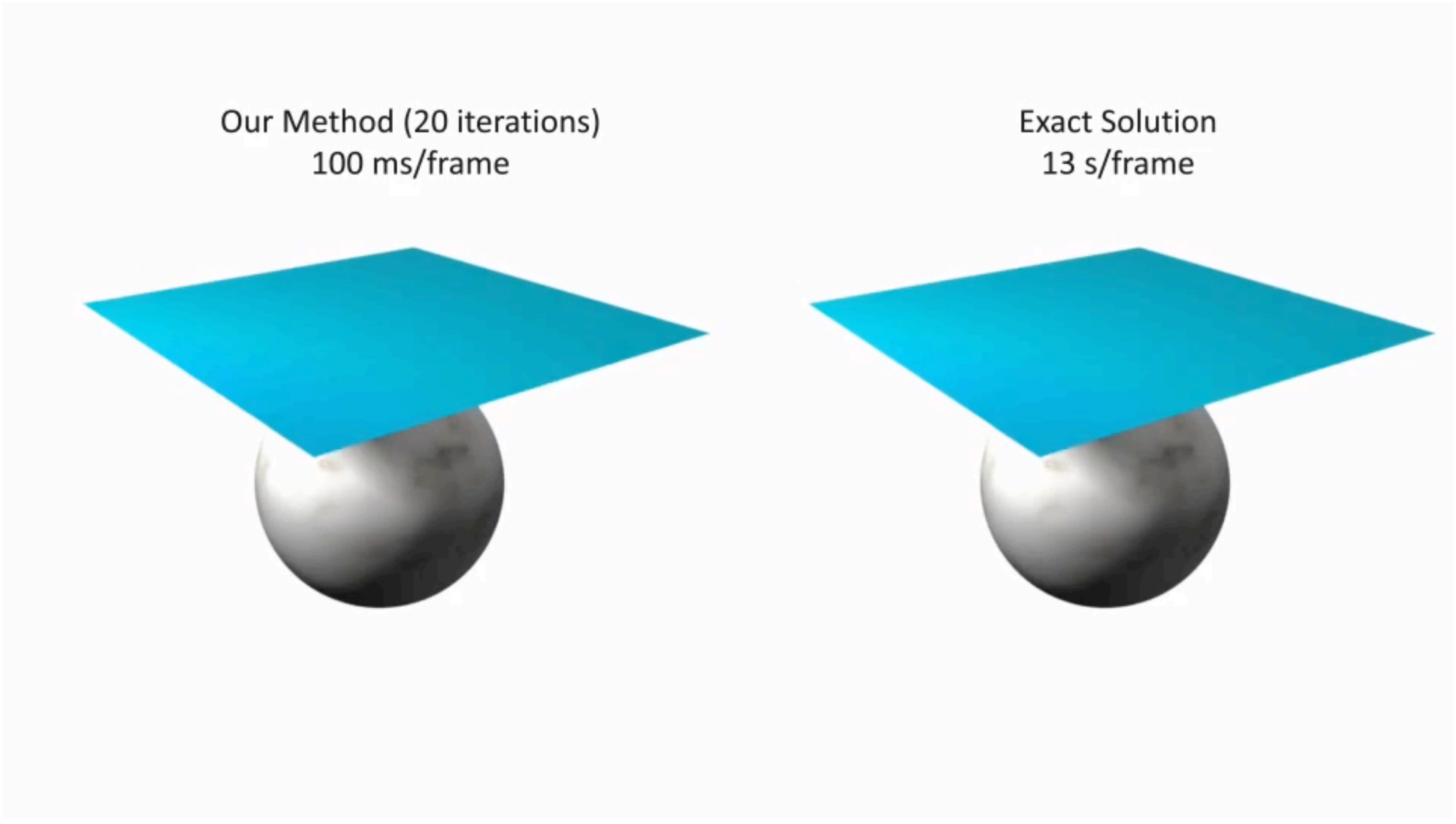
Results


Our Method
Newton with Direct Solver
Newton with PCG Solver
(a)
(b)
(c)
(d)
relative error
time (seconds)
10^{0}
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
10^{-6}
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5

Results



Results


Our Method (20 iterations)
100 ms/frame
Exact Solution
13 s/frame