263-5805-00L
Modeling Elastic Objects (Finite Element Method)

Moritz Bächer

ETH zürich

Disney Research

Agenda

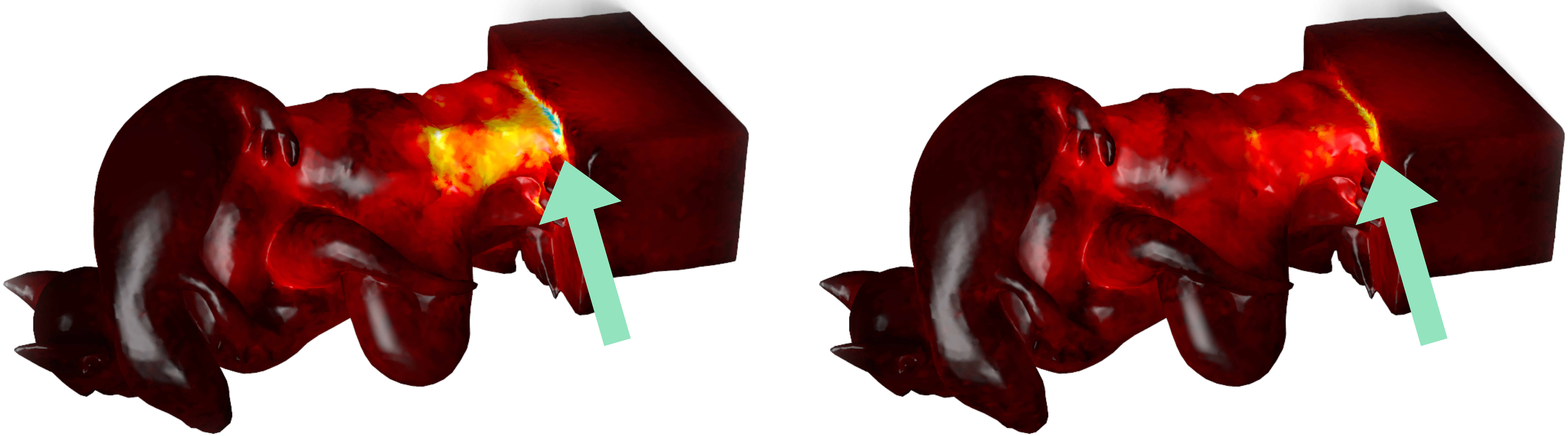
- Motivation
- Energy, forces, static vs. dynamic analysis
- Numerical time integration (explicit vs. implicit schemes)
- Continuum Mechanics: strain, stress, material models
- Linear vs. nonlinear FEM (Finite Element Method)
- Discretization and assembly

Motivation

[Zehnder et al. 2017]

Motivation

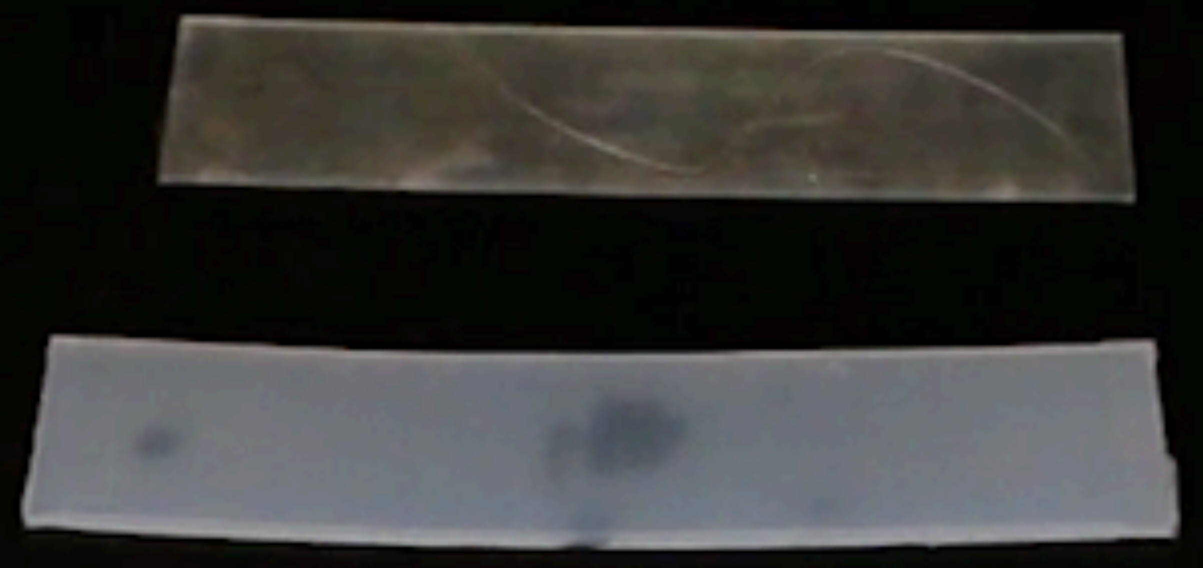
unoptimized optimized



[Zehnder et al. 2017]

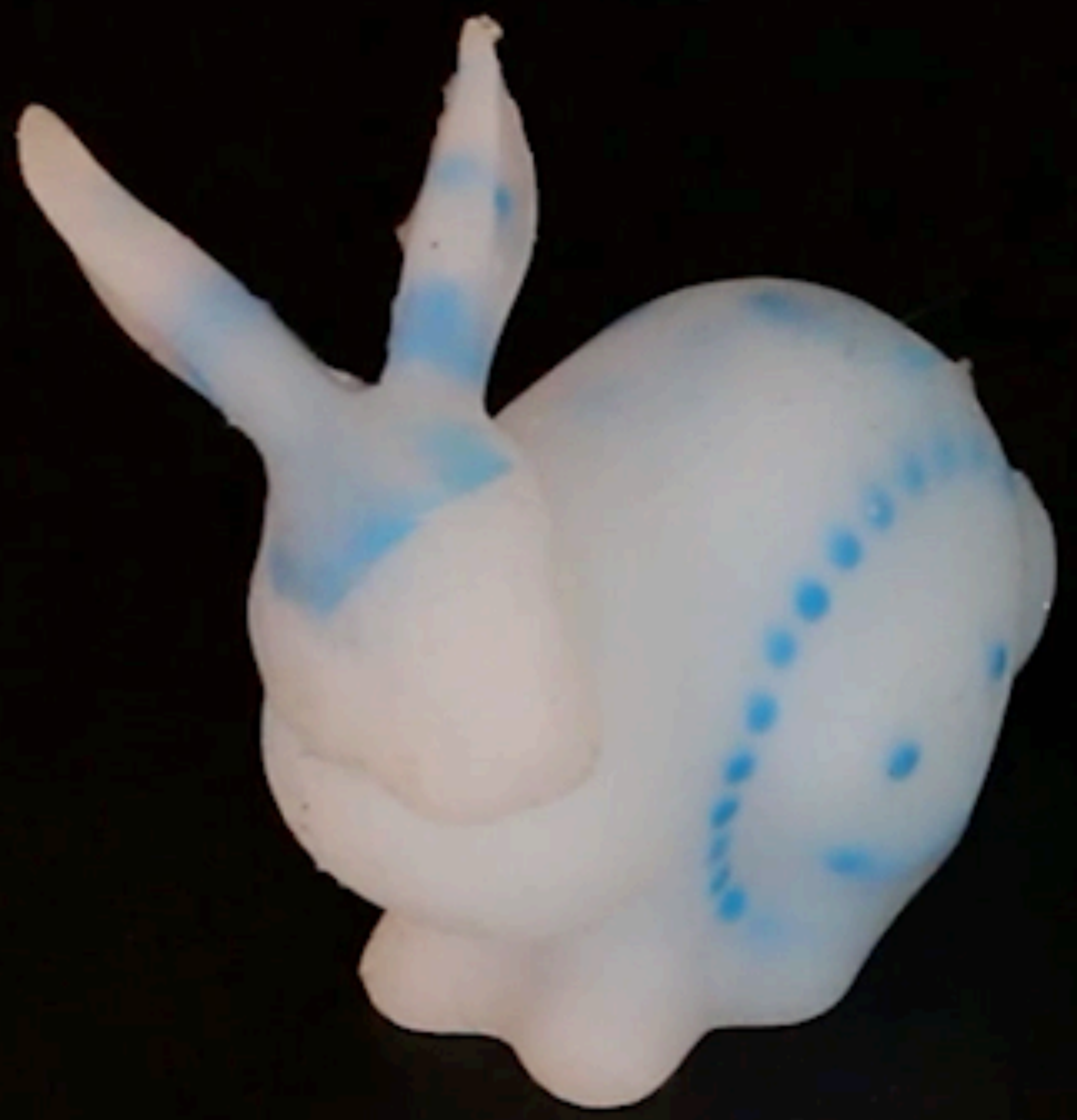
[Schumacher et al. 2018]

Motivation



[Zehnder et al. 2017]

Motivation



[Zehnder et al. 2017]

Agenda

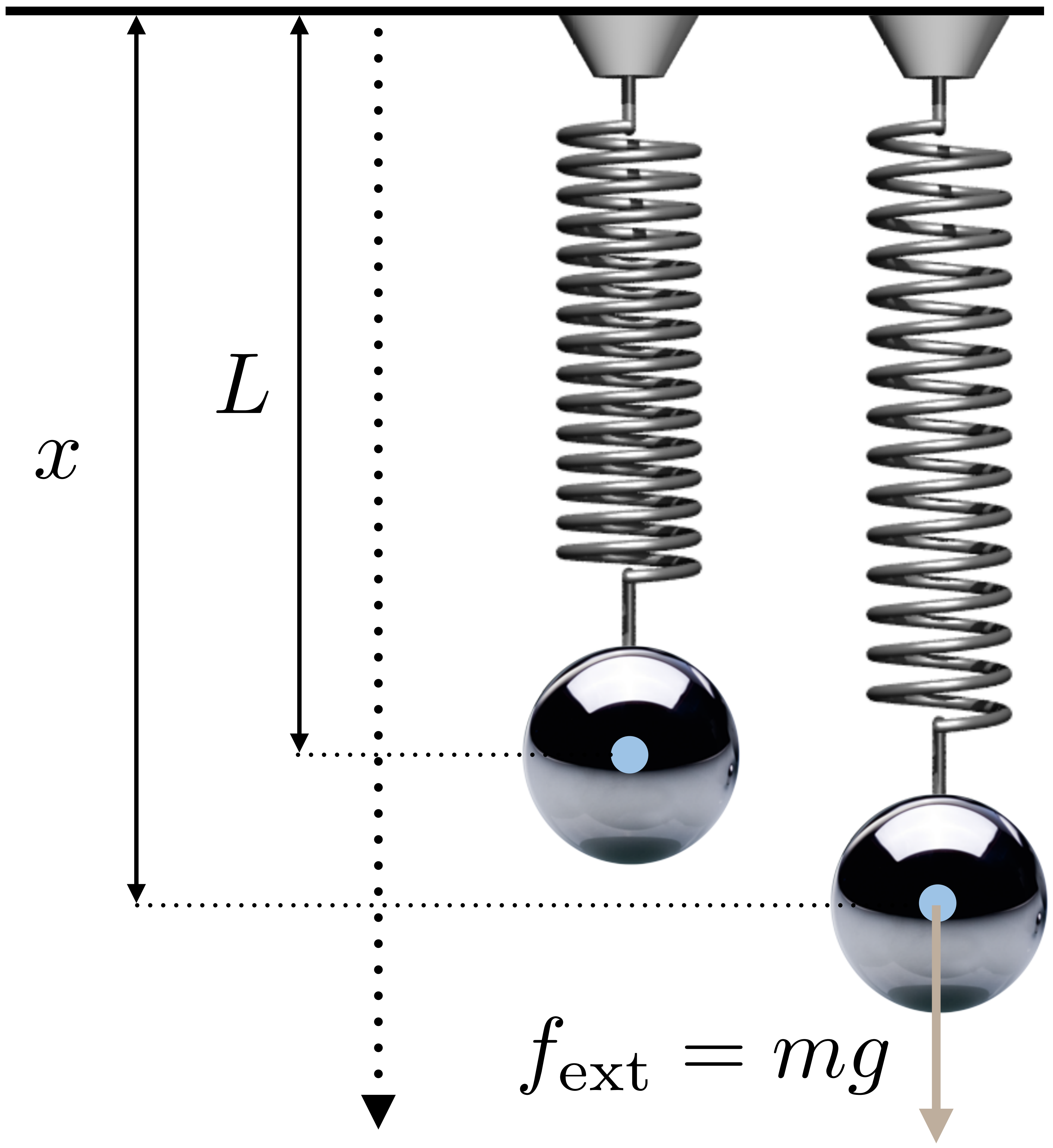
- Motivation
- Energy, forces, static vs. dynamic analysis
- Numerical time integration (explicit vs. implicit schemes)
- Continuum mechanics: strain, stress, material models
- Linear vs. nonlinear FEM (Finite Element Method)
- Discretization and assembly

Mass-Spring Systems

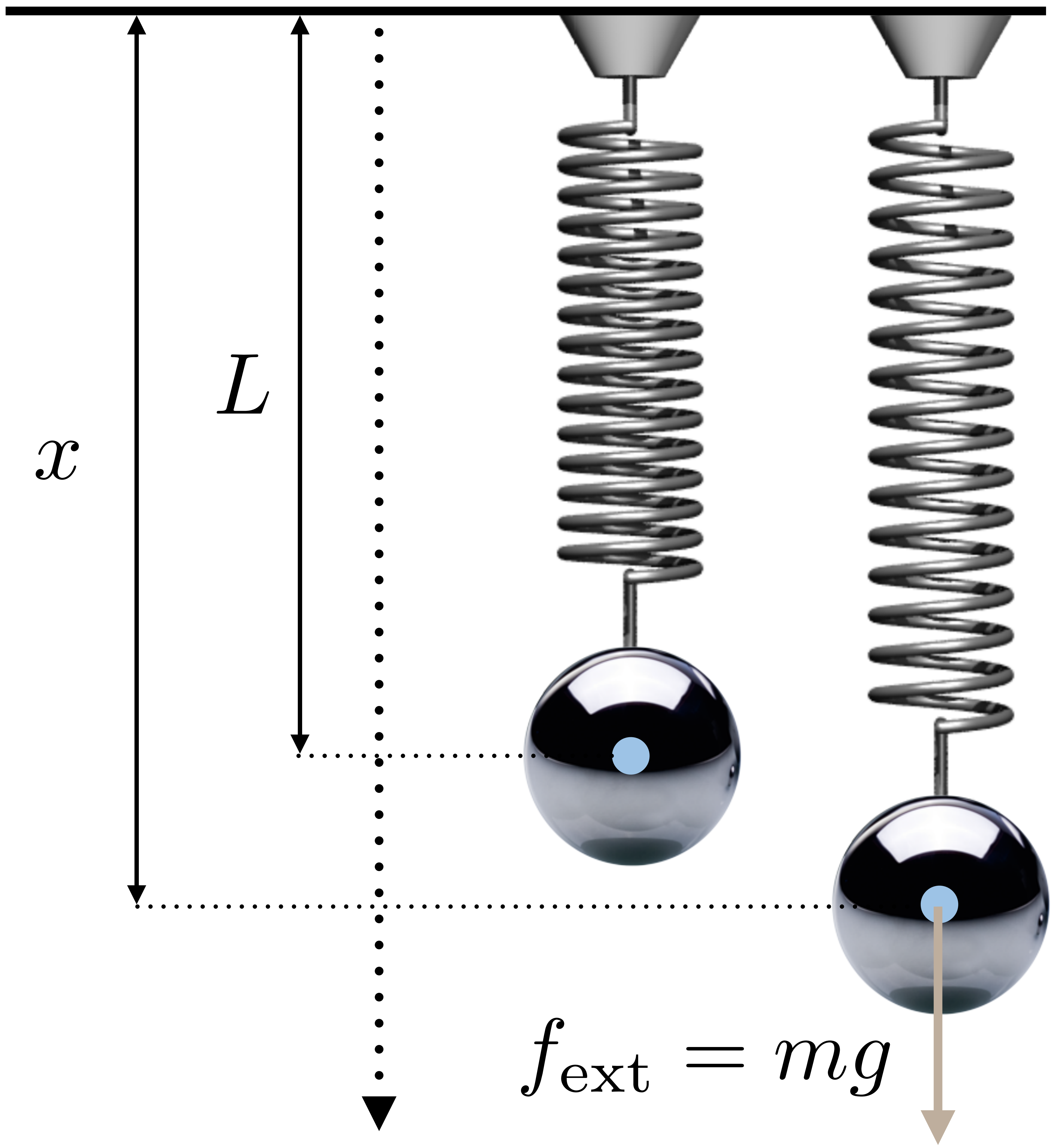
- Point masses
- Mass m
- Location $\mathbf{x}$

- Massless springs
- Stiffness k
- Rest length L

Energy



Energy

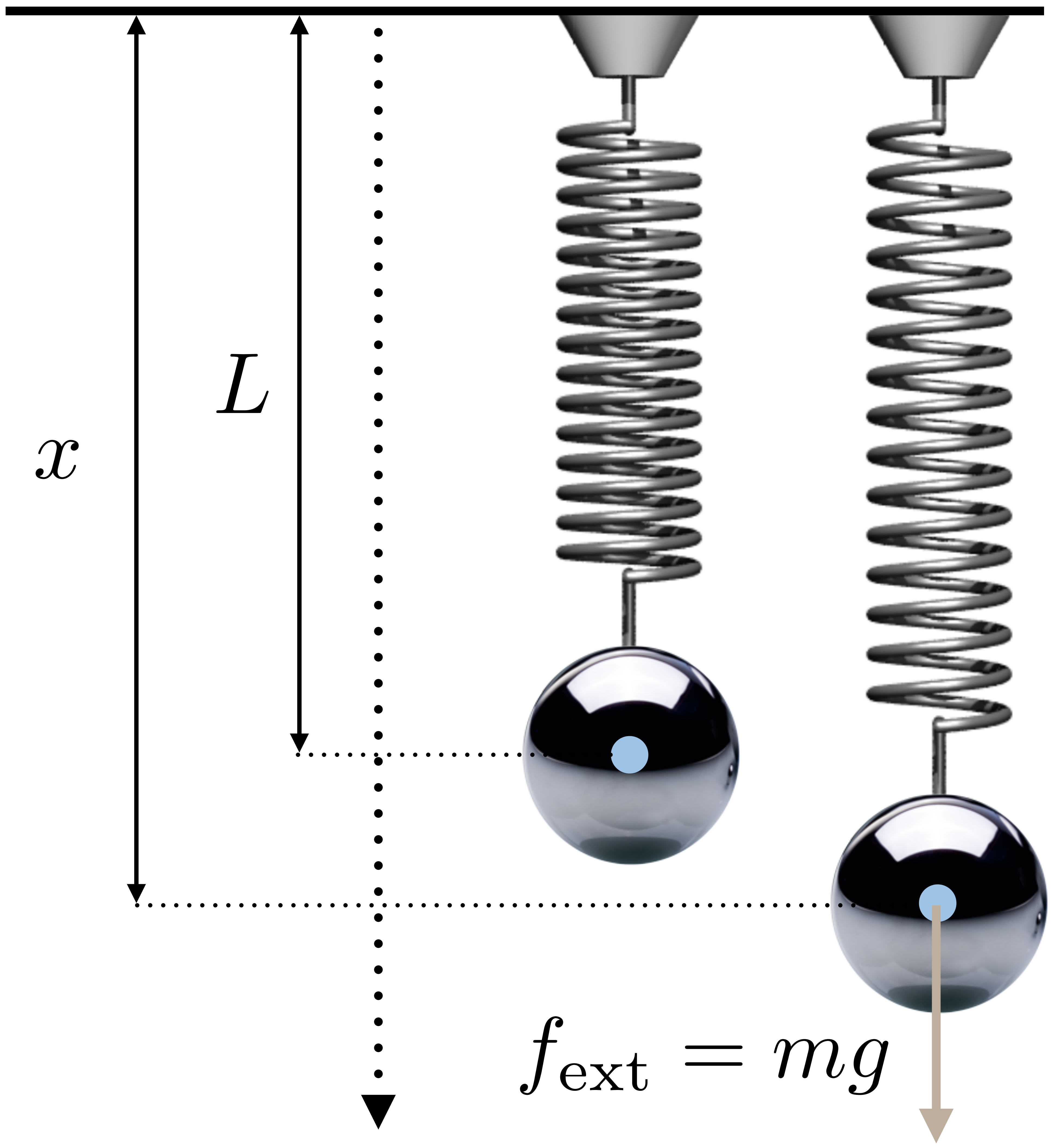


$$E(x) = E_{\text{int}}(x) - E_{\text{ext}}(x)$$
$$= \frac{1}{2}k(x-L)^2 - f_{\text{ext}}(x-L)$$

Potential energy

Work = force x displacement

Forces



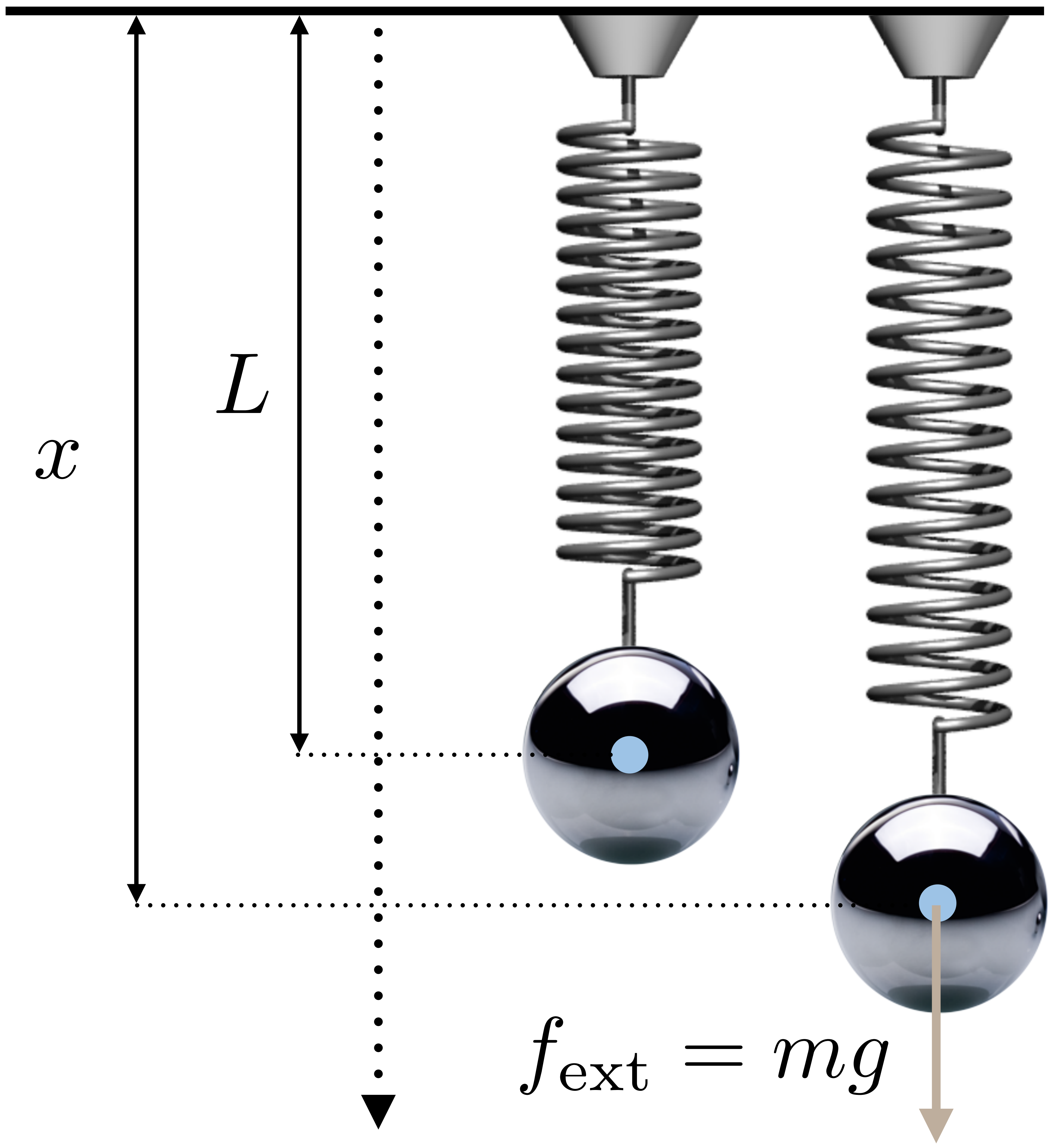
$$\frac{\mathrm{d}E(x)}{\mathrm{d}x} = \frac{\mathrm{d}E_{\text{int}}(x)}{\mathrm{d}x} - \frac{\mathrm{d}E_{\text{ext}}(x)}{\mathrm{d}x}$$

$$= f_{\text{int}}(x) - f_{\text{ext}}$$

$$= k(x - L) - f_{\text{ext}}$$

Hooke's law

Static Analysis



$$\frac{\mathrm{d}E(x)}{\mathrm{d}x} = \frac{\mathrm{d}E_{\text{int}}(x)}{\mathrm{d}x} - \frac{\mathrm{d}E_{\text{ext}}(x)}{\mathrm{d}x}$$

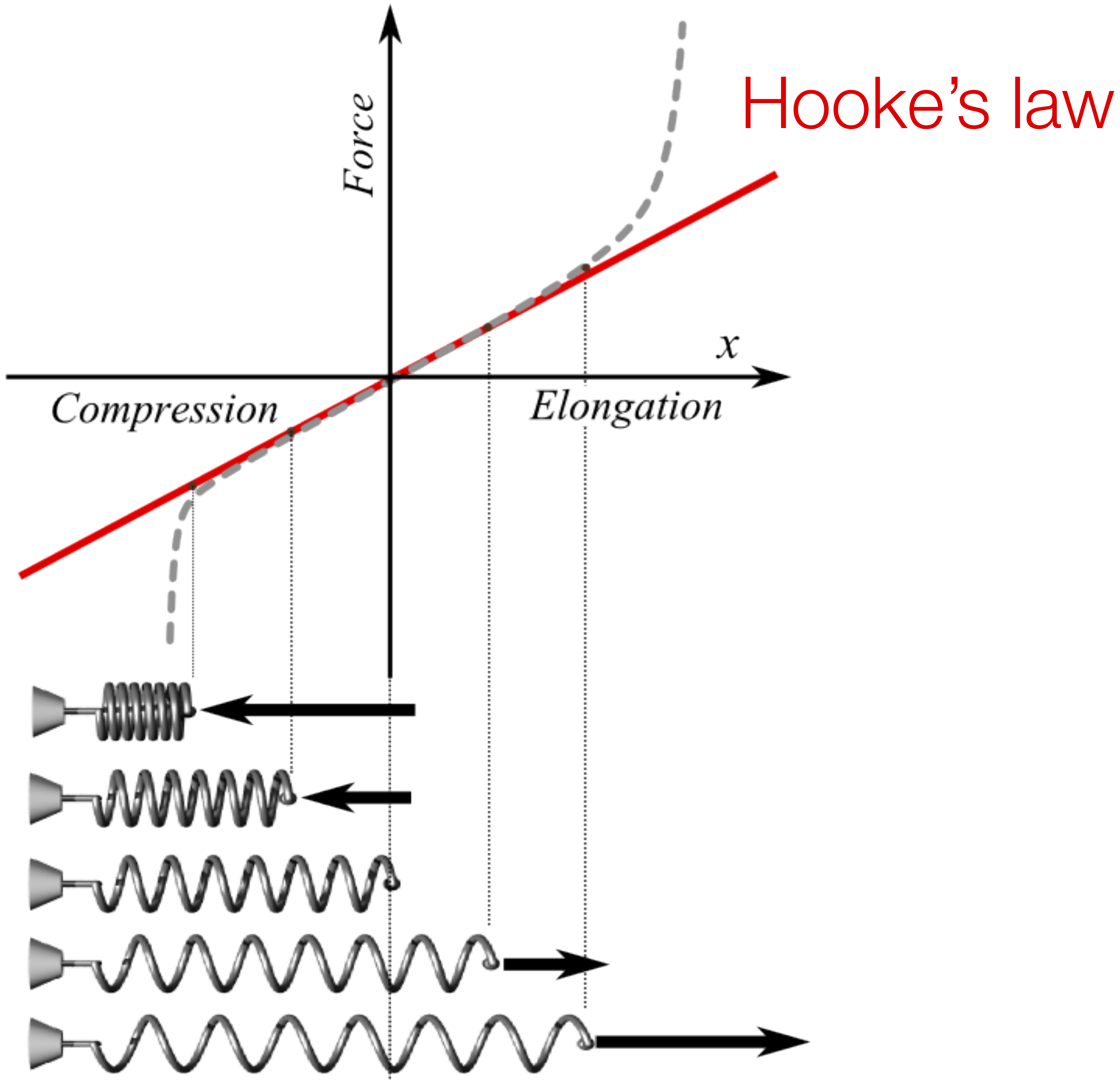
$$= f_{\text{int}}(x) - f_{\text{ext}} \stackrel{!}{=} 0$$

Static equilibrium:
internal forces = external forces

$$k(x-L) = f_{\text{ext}} \longrightarrow x = \frac{f_{\text{ext}}}{k} + L$$

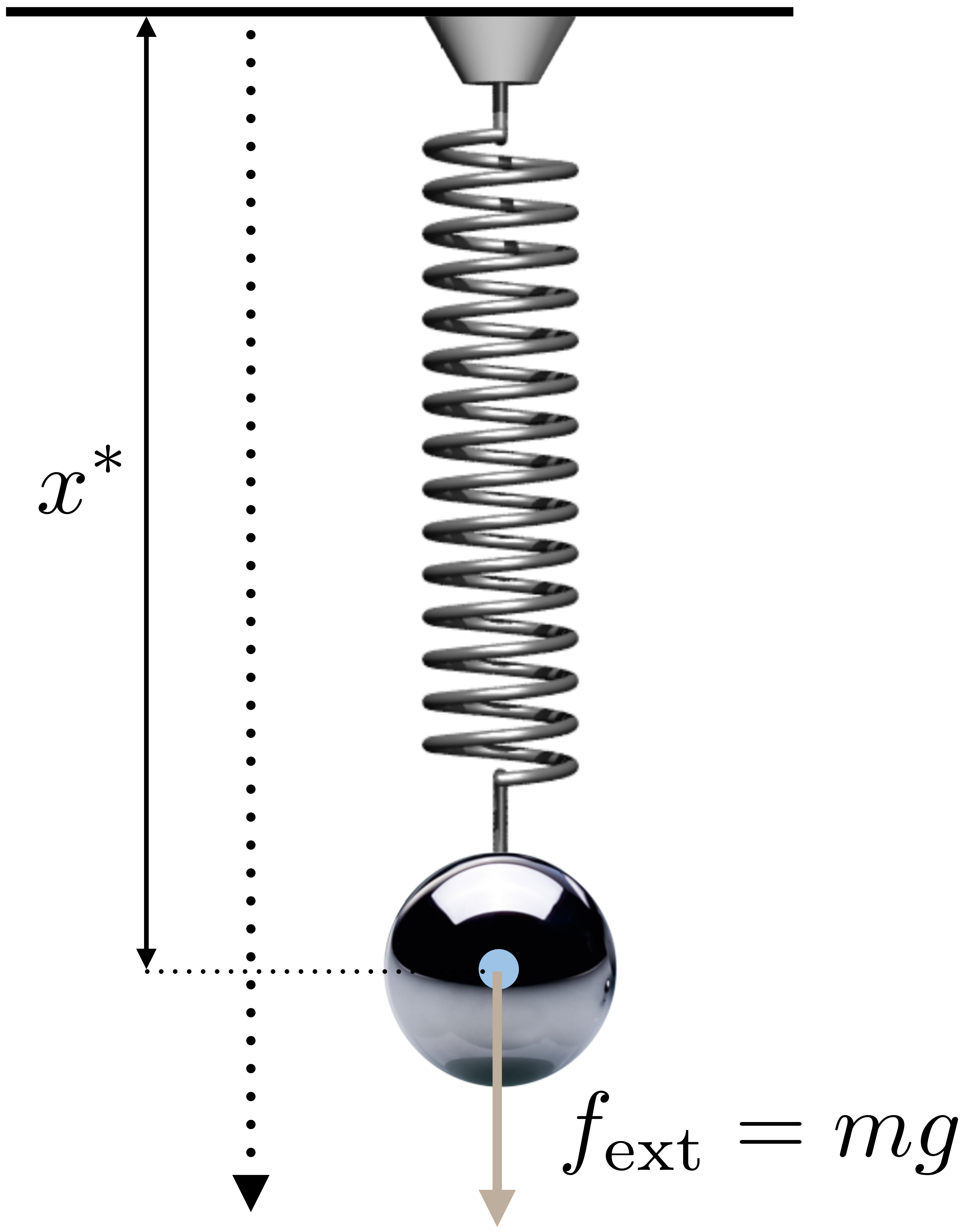
Static solution

Static Analysis



- Elastic springs
- Linear springs
  - small displacement
  - Hooke's law
- General: non-linear behavior
  - large displacements

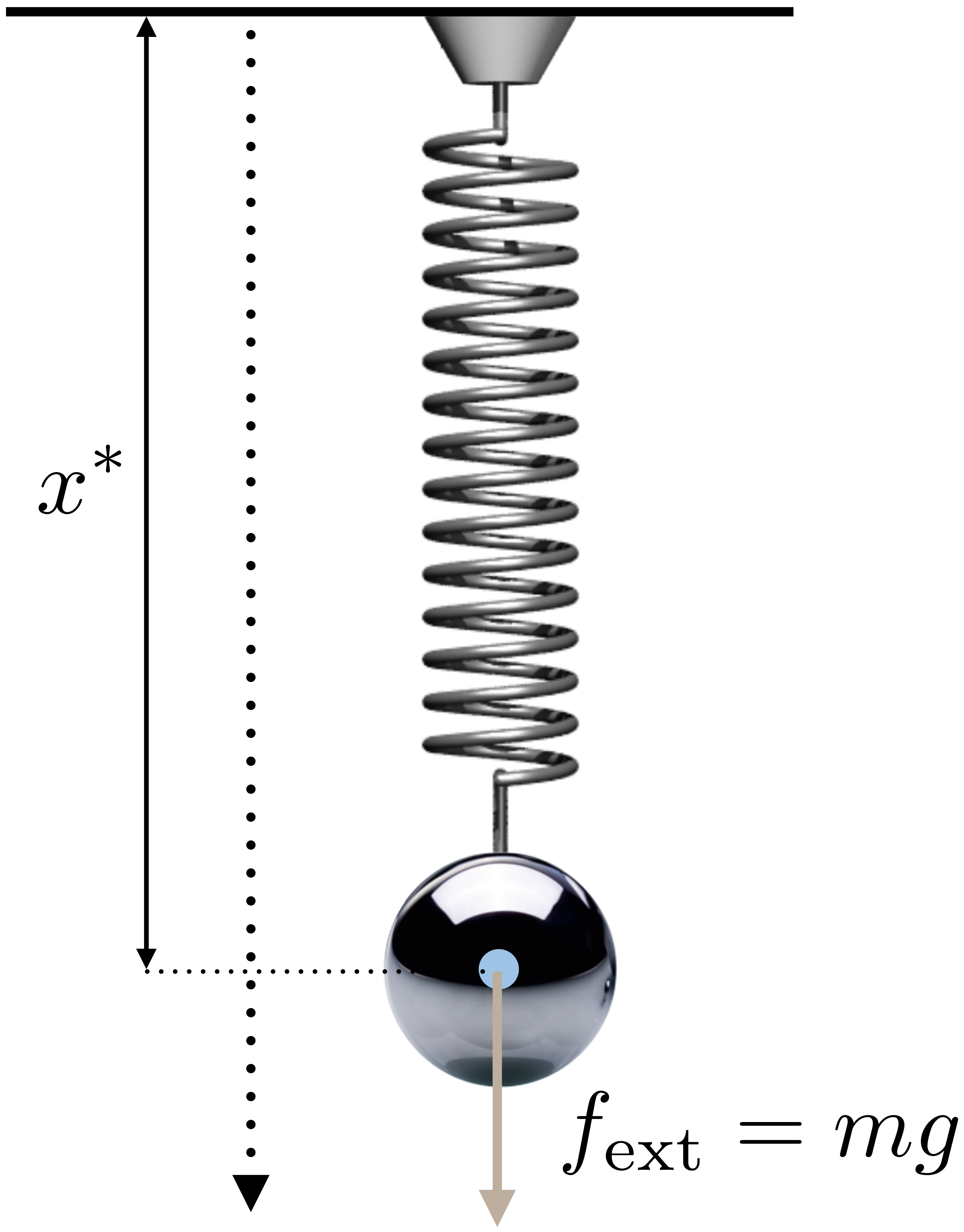
Static Analysis



- Minimize energy

$$\min_x f_{\text{static}}(x) \qquad f_{\text{static}}(x) = E(x)$$
$$= E_{\text{int}}(x) - E_{\text{ext}}(x)$$

Static Analysis

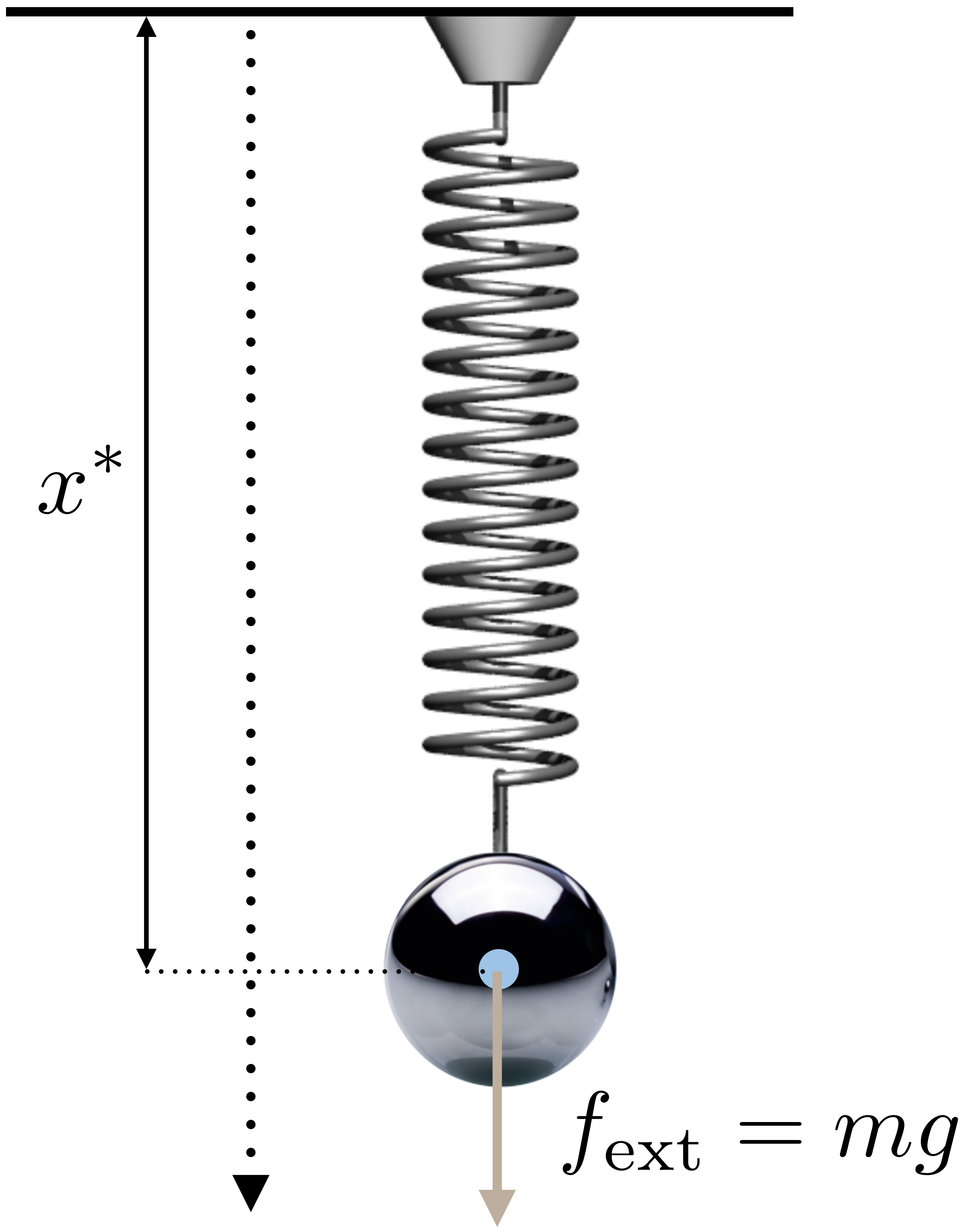


- Minimize energy

$$\min_x f_{\text{static}}(x) \quad f_{\text{static}}(x) = E(x)$$
$$= E_{\text{int}}(x) - E_{\text{ext}}(x)$$

- Minimum x^*
  - first derivative: zero $\quad E_x(x^*) \stackrel{!}{=} 0$
  - second derivative: positive $\quad E_{xx}(x^*) > 0$

Static Analysis



- Minimize energy

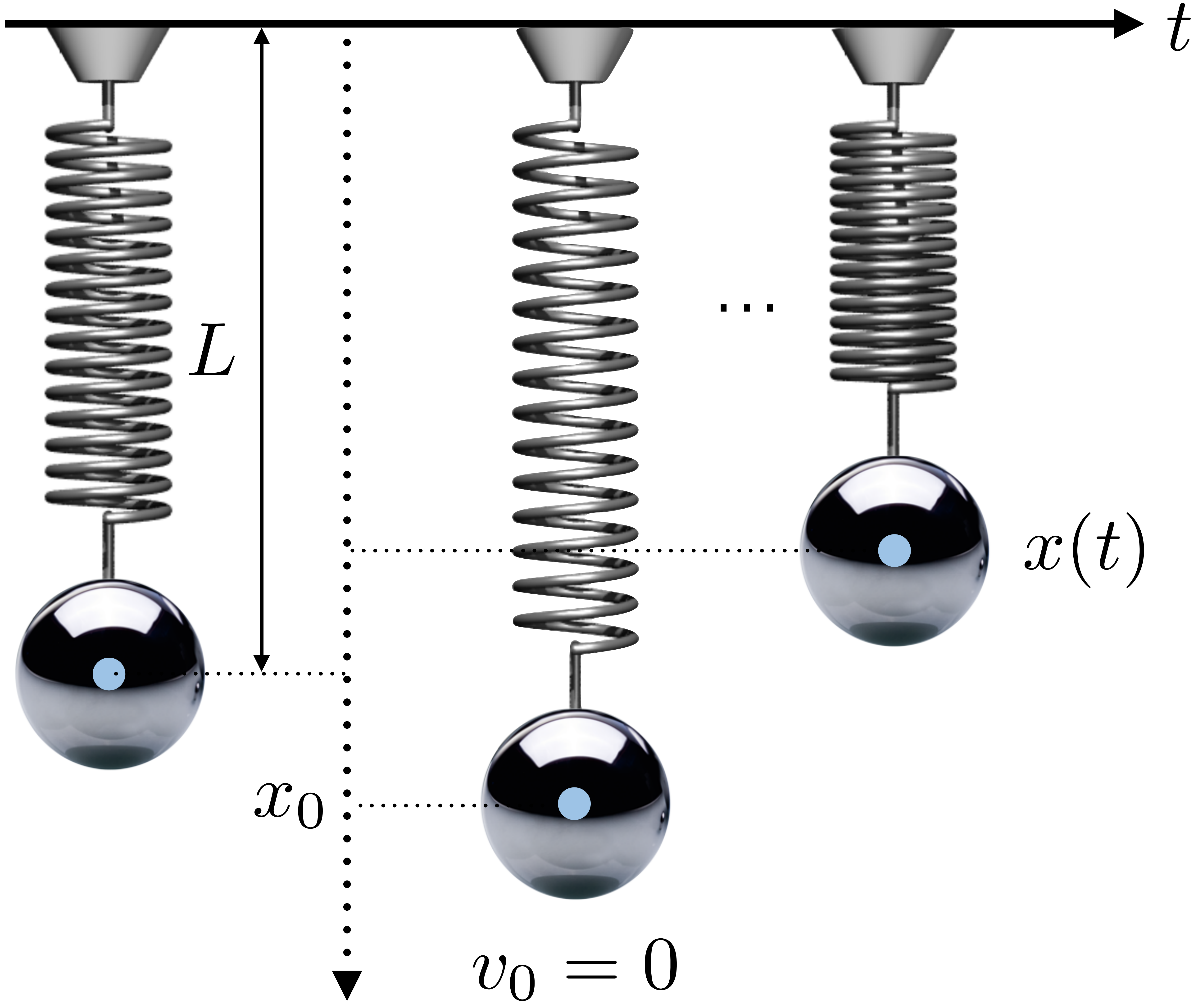
$$\min_x f_{\text{static}}(x) \quad f_{\text{static}}(x) = E(x)$$
$$= E_{\text{int}}(x) - E_{\text{ext}}(x)$$

- Minimum x^*
  - first derivative: zero $\quad E_x(x^*) \stackrel{!}{=} 0$
  - second derivative: positive $\quad E_{xx}(x^*) > 0$

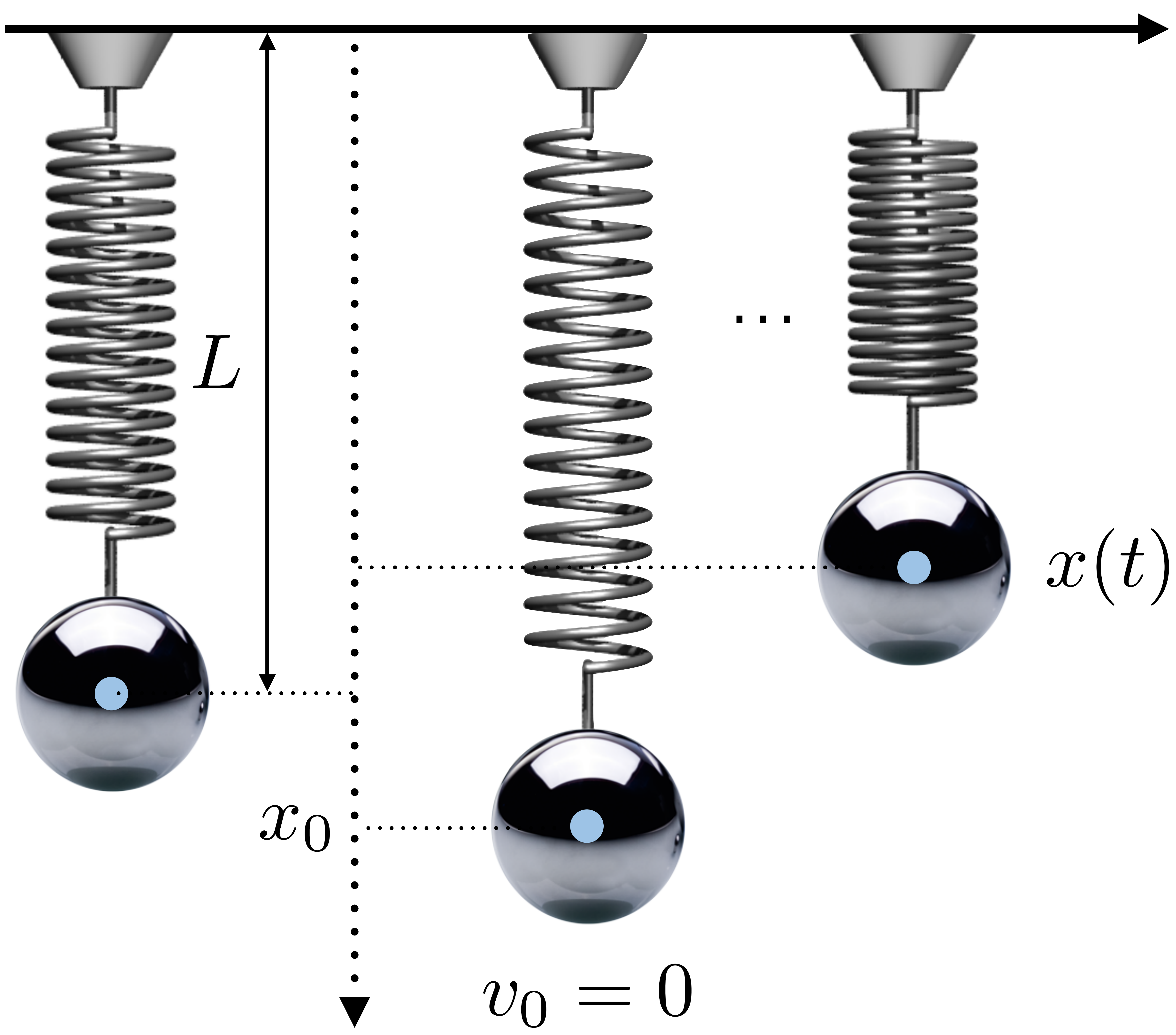
- Static equilibrium

$$E_x(x^*) = f_{\text{int}}(x^*) - f_{\text{ext}} \stackrel{!}{=} 0$$

Equations of Motion



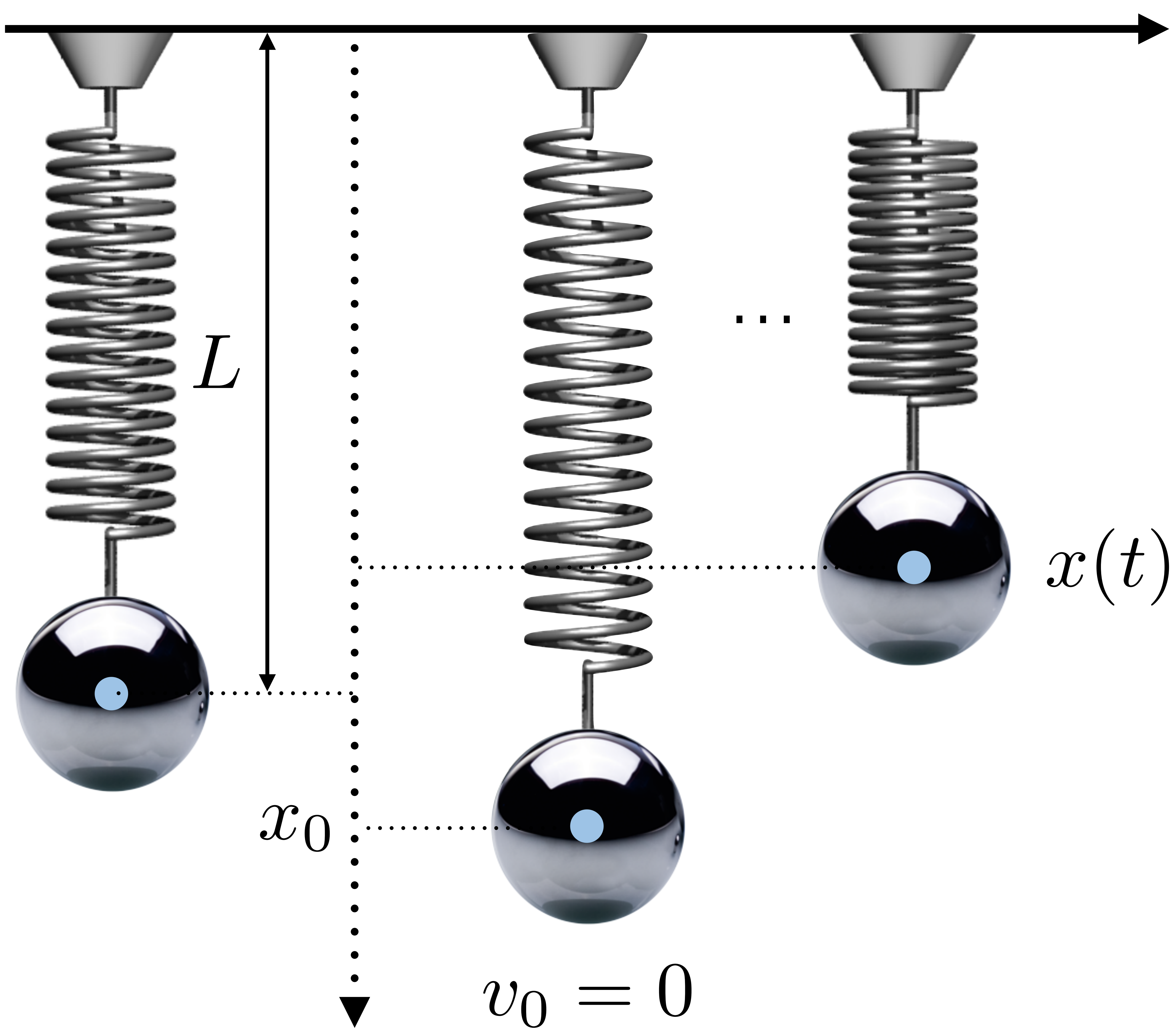
Equations of Motion



- Velocity

$$v(t) = \frac{\mathrm{d}x(t)}{\mathrm{d}t}$$

Equations of Motion



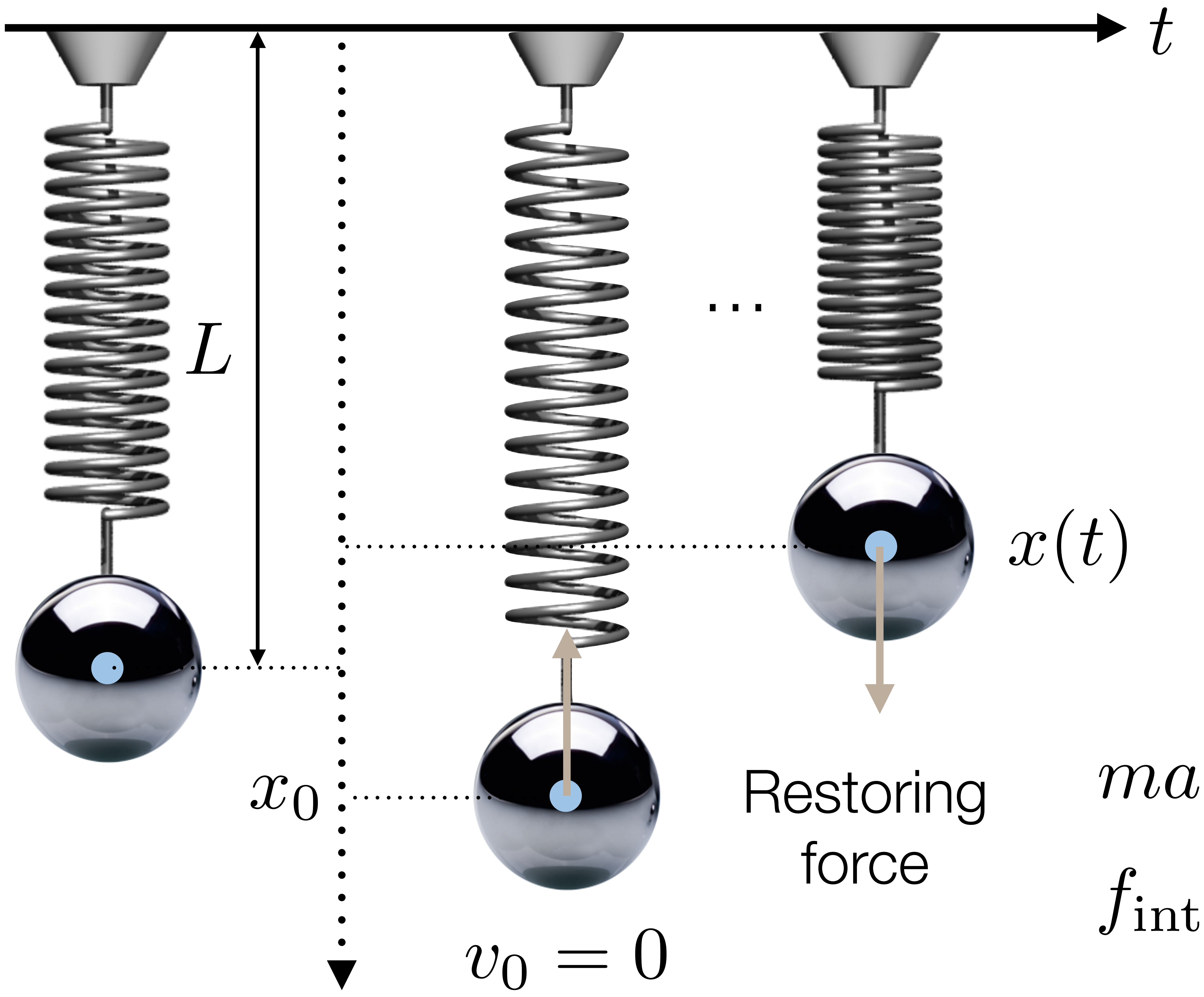
- Velocity

$$v(t) = \frac{\mathrm{d}x(t)}{\mathrm{d}t}$$

- Acceleration

$$a(t) = \frac{\mathrm{d}v(t)}{\mathrm{d}t} = \frac{\mathrm{d}^2x(t)}{\mathrm{d}t^2}$$

Equations of Motion



- Velocity

$$v(t) = \frac{\mathrm{d}x(t)}{\mathrm{d}t}$$

- Acceleration

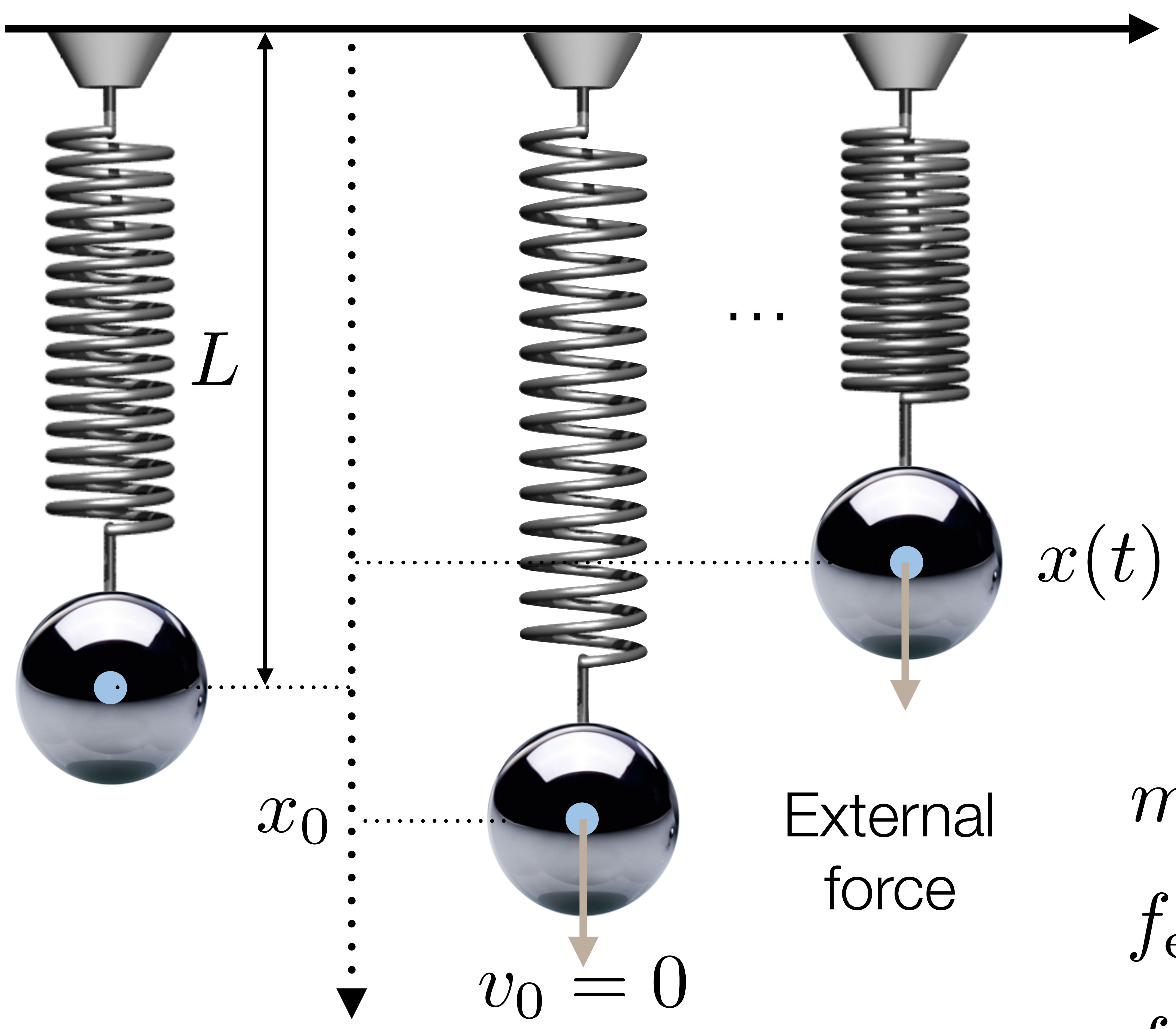
$$a(t) = \frac{\mathrm{d}v(t)}{\mathrm{d}t} = \frac{\mathrm{d}^2x(t)}{\mathrm{d}t^2}$$

- Newton's 2nd law

$$ma(t) = -f_{\mathrm{int}}(t)$$

$$f_{\mathrm{int}}(t) = k\,(x(t) - L)$$

Equations of Motion



- Velocity

$$v(t) = \frac{\mathrm{d}x(t)}{\mathrm{d}t}$$

- Acceleration

$$a(t) = \frac{\mathrm{d}v(t)}{\mathrm{d}t} = \frac{\mathrm{d}^2x(t)}{\mathrm{d}t^2}$$

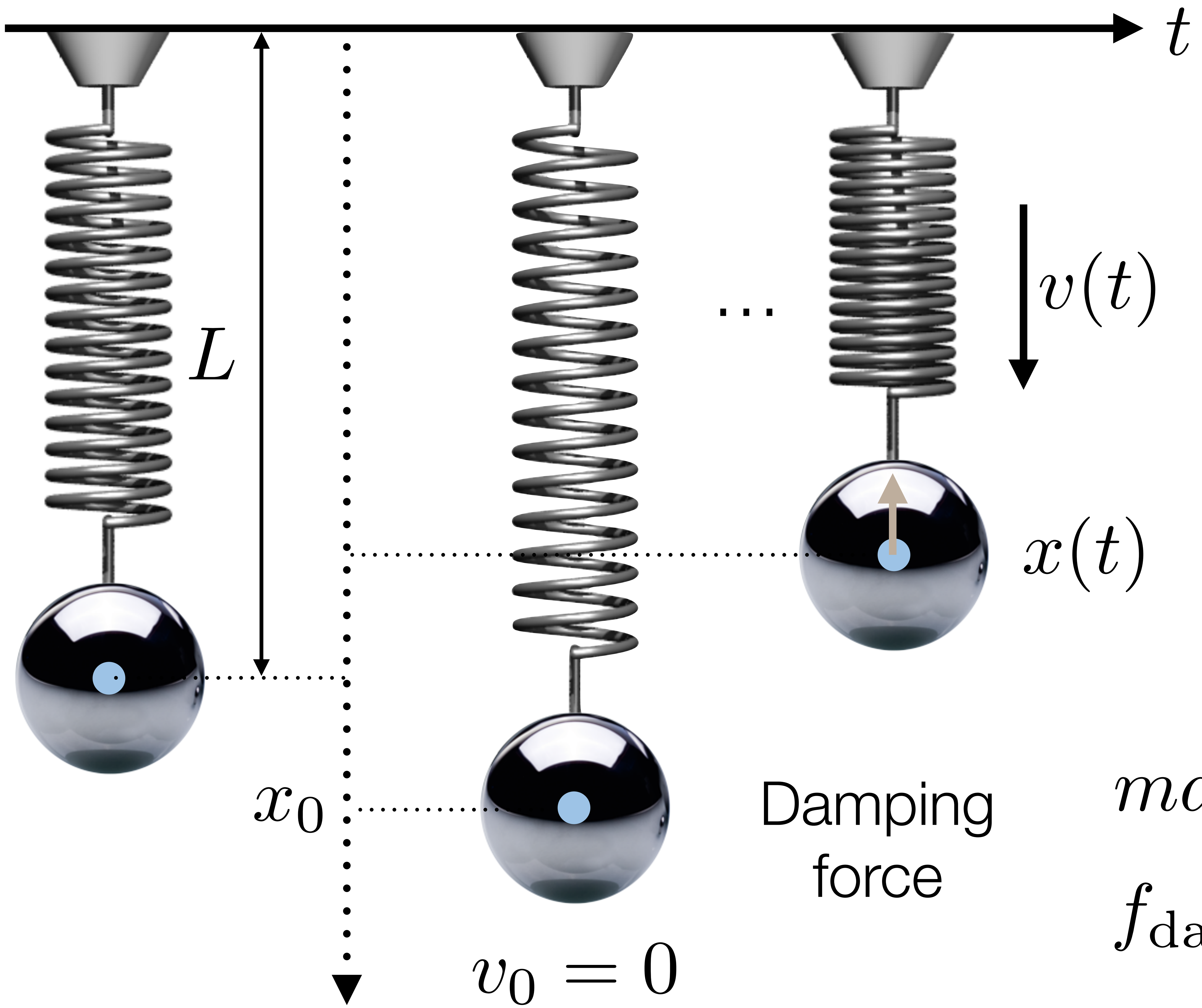
- Newton's 2nd law

$$ma(t) = -f_{\mathrm{int}}(t) + f_{\mathrm{ext}}(t)$$

$$f_{\mathrm{ext}} = mg$$

$$f_{\mathrm{ext}}(t) = mg + \cos(\omega t + \phi)$$

Equations of Motion



- Velocity

$$v(t) = \frac{\mathrm{d}x(t)}{\mathrm{d}t}$$

- Acceleration

$$a(t) = \frac{\mathrm{d}v(t)}{\mathrm{d}t} = \frac{\mathrm{d}^2x(t)}{\mathrm{d}t^2}$$

- Newton's 2nd law

$$ma(t) = -f_{\mathrm{int}}(t) + f_{\mathrm{ext}}(t) - f_{\mathrm{damp}}(t)$$

$$f_{\mathrm{damp}}(t) = \gamma v(t)$$

Control damping

Dynamic Analysis

$$ma(t) + f_{\mathrm{damp}}(t) = -f_{\mathrm{int}}(t) + f_{\mathrm{ext}}(t)$$

$$m\frac{\mathrm{d}^2x(t)}{\mathrm{d}t^2} + \gamma\frac{\mathrm{d}x(t)}{\mathrm{d}t} = -f_{\mathrm{int}}(t) + f_{\mathrm{ext}}(t)$$

2nd order ordinary differential equation (ODE)

$$x(t_0) = x_0 \qquad \frac{\mathrm{d}x(t_0)}{dt} = v_0$$

Initial value problem (IVP)

Dynamic Analysis

How do we determine motion $x(t)$ *?*

$$m\frac{\mathrm{d}^2x(t)}{\mathrm{d}t^2} + \gamma\frac{\mathrm{d}x(t)}{\mathrm{d}t} = -f_{\mathrm{int}}(t) + f_{\mathrm{ext}}(t)$$

2nd order ordinary differential equation (ODE)

$$x(t_0) = x_0 \qquad \frac{\mathrm{d}x(t_0)}{dt} = v_0$$

Initial value problem (IVP)

Agenda

- Motivation
- Energy, forces, static vs. dynamic analysis
- Numerical time integration (explicit vs. implicit schemes)
- Continuum mechanics: strain, stress, material models
- Linear vs. nonlinear FEM (Finite Element Method)
- Discretization and assembly

Dynamic Analysis

- Two coupled 1st order ODEs

$$\frac{\mathrm{d}x(t)}{dt} = v(t) \qquad \frac{\mathrm{d}v(t)}{dt} = \frac{1}{m}\left(-f_{\mathrm{int}}(t) + f_{\mathrm{ext}}(t) - \gamma v(t)\right)$$

Dynamic Analysis

- Two coupled 1st order ODEs

$$\frac{\mathrm{d}x(t)}{dt} = v(t) \qquad \frac{\mathrm{d}v(t)}{dt} = \frac{1}{m}\left(-f_{\text{int}}(t) + f_{\text{ext}}(t) - \gamma v(t)\right)$$

- Rewrite as one system of 1st order ODEs

$$\boxed{\begin{gathered}\mathbf{y}(t) = \begin{bmatrix} x(t) \\ v(t) \end{bmatrix} \quad \mathbf{y}'(t) = \begin{bmatrix} v(t) \\ \frac{1}{m}\left(-f_{\text{int}}(t) + f_{\text{ext}} - \gamma v(t)\right) \end{bmatrix} \\ \mathbf{y}(t_0) = \begin{bmatrix} x(t_0) \\ v(t_0) \end{bmatrix} = \begin{bmatrix} x_0 \\ v_0 \end{bmatrix}\end{gathered}}$$

Initial value problem (IVP)

Dynamic Analysis

Given system of 1st order ODEs with initial conditions, how do we solve for $\mathbf{y}(t)$ *?*

$$\mathbf{y}(t) = \begin{bmatrix} x(t) \\ v(t) \end{bmatrix} \qquad \mathbf{y}'(t) = \begin{bmatrix} v(t) \\ \frac{1}{m}\left(-f_{\text{int}}(t) + f_{\text{ext}} - \gamma v(t)\right) \end{bmatrix}$$

$$\mathbf{y}(t_0) = \begin{bmatrix} x(t_0) \\ v(t_0) \end{bmatrix} = \begin{bmatrix} x_0 \\ v_0 \end{bmatrix}$$

Initial value problem (IVP)

Time Integration

- General IVP
  - single ODE $y'(t) = f(t, y(t)) \quad y(t_0) = y_0$
  - system of ODEs $\mathbf{y}'(t) = \mathbf{f}(t, \mathbf{y}(t)) \quad \mathbf{y}(t_0) = \mathbf{y}_0$

Time Integration

- General IVP
  - single ODE $y'(t) = f(t, y(t)) \quad y(t_0) = y_0$
  - system of ODEs $\mathbf{y}'(t) = \mathbf{f}(t, \mathbf{y}(t)) \quad \mathbf{y}(t_0) = \mathbf{y}_0$

- Why time integration?

$$y(t+h) = y(t) + \int_t^{t+h} f(t, y(t))\mathrm{d}t \qquad y(t_0) = y_0$$

Solution at time t plus step h

Time Integration

- General IVP
  - single ODE $y'(t) = f(t, y(t)) \quad y(t_0) = y_0$
  - system of ODEs $\mathbf{y}'(t) = \mathbf{f}(t, \mathbf{y}(t)) \quad \mathbf{y}(t_0) = \mathbf{y}_0$

- Why time integration?

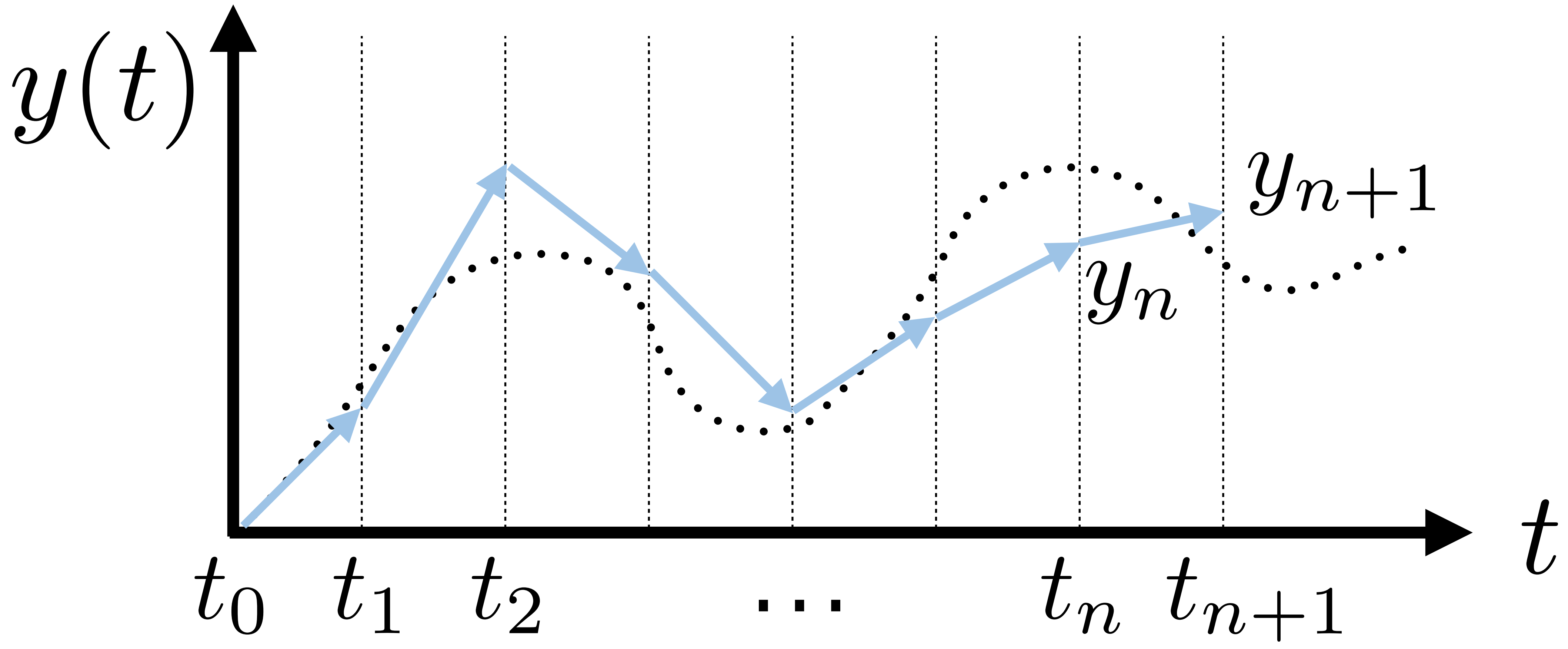
$$y(t+h) = y(t) + \int_t^{t+h} f(t, y(t))\mathrm{d}t \qquad y(t_0) = y_0$$

Solution at time t plus step h

- Solve IVP numerically $\longrightarrow$ numerical (time) integration

Numerical Time Integration

- Notation
  - $y(t)$ analytical solution
  - y_i approximate solution at $t_i = t_0 + ih$
  - h time step *(constant)*
- Problem: given y_n , compute y_{n+1}



Numerical Time Integration

How do we get from $y(t)$ *to* $y(t+h)$*?*

- Fundamental theorem of calculus

$$y(t+h) = y(t) + \int_t^{t+h} f(t, y(t))\mathrm{d}t$$

$$y(t+h) \approx y(t) + hf(t, y(t))$$

left-hand
rectangle method

- Taylor expansion (1st order approximation)

$$y(t+h) = y(t) + hy'(t) + O(h^2)$$

“forward”
Euler

Explicit Euler

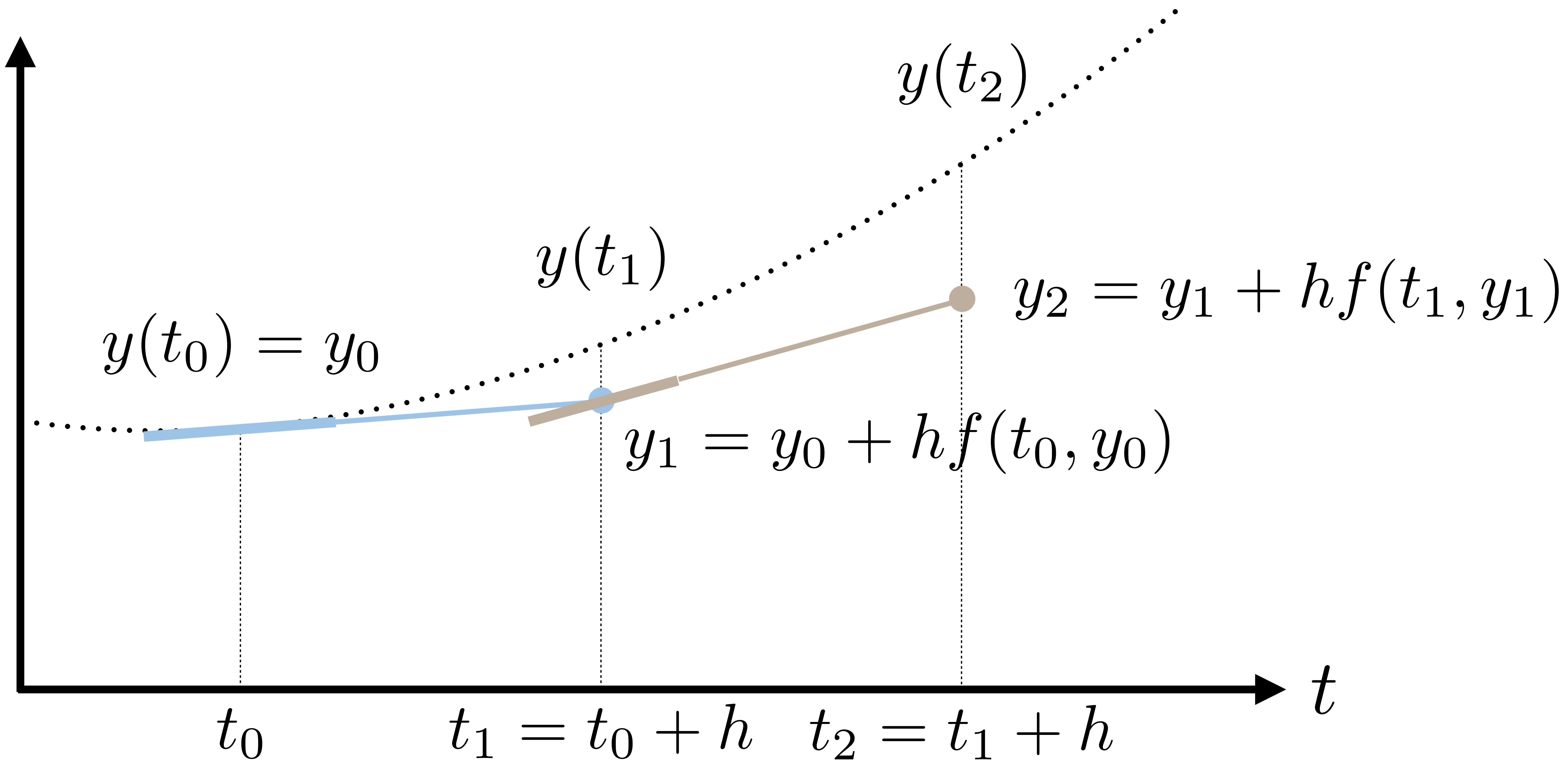
$$\boxed{y_{n+1} = y_n + hf(t_n, y_n)}$$

Euler step (1768)

- **Idea:** start at initial condition and take step into direction of tangent

- Iteration scheme: $y_0 \longrightarrow f(t_0, y_0) \longrightarrow y_1 \longrightarrow f(t_1, y_1) \longrightarrow \ldots$

Explicit Euler: Graphically



Explicit Euler: Mass-Spring System

- Set initial conditions: *Position* x_0

  Velocity v_0

Explicit Euler: Mass-Spring System

- Set initial conditions: *Position* x_0

  Velocity v_0

1. Evaluate derivatives: *Position* → *Velocity* $x'(t_n) = v(t_n)$

   Velocity → *Acceleration*

$$v'(t_n) = \frac{1}{m}(-f_{\text{int}}(t_n) + f_{\text{ext}}(t_n) - \gamma v(t_n))$$

Explicit Euler: Mass-Spring System

- Set initial conditions: *Position* x_0

  Velocity v_0

1. Evaluate derivatives: *Position* ⟶ *Velocity* $x'(t_n) = v(t_n)$

   Velocity ⟶ *Acceleration*

$$v'(t_n) = \frac{1}{m}\left(-f_{\text{int}}(t_n) + f_{\text{ext}}(t_n) - \gamma v(t_n)\right)$$

2. Euler step: *Position* $x(t_n + h) = x(t_n) + hx'(t_n)$

   Velocity $v(t_n + h) = v(t_n) + hv'(t_n)$

Analysis

How to evaluate integration schemes?

Criteria

- **Convergence:** do approximations converge to true solution, i.e., $h \to 0$ implies $y_i \to y(t_i)$?
- **Accuracy:** how fast does the error decrease as $h \to 0$?
- **Stability:** is the solution always bounded, i.e., $|y_n| < \infty$?
- **Efficiency:** is a given method a *good choice* for a given problem?

Analysis: Accuracy

- Numerical solution exhibits error

$$\left| \left[y_n + \int_{t_n}^{t_{n+1}} f(t, y(t)) \mathrm{d}t \right] - y_{n+1} \right|$$

local error *(single step)*

$$|y_i - y(t_i)|$$

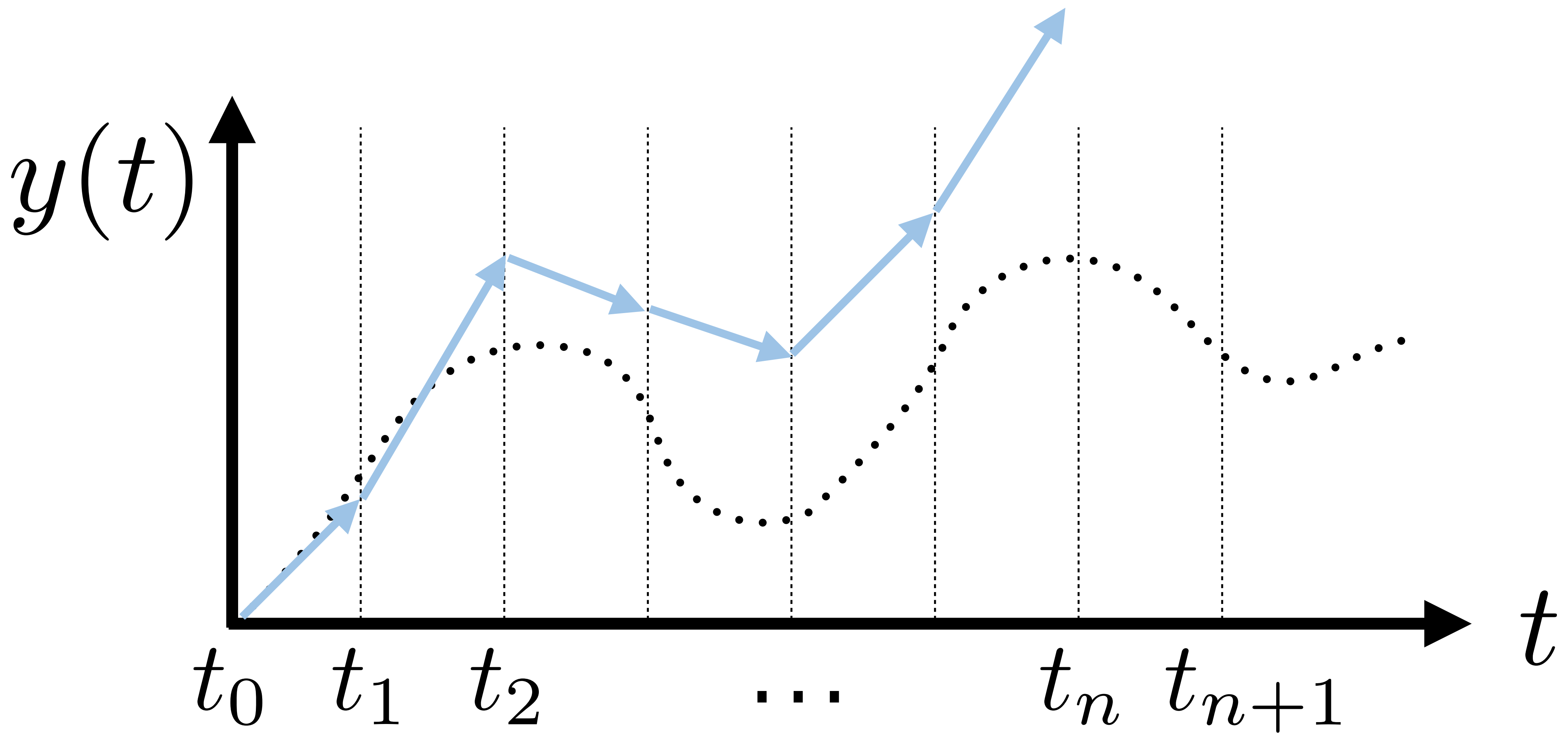
global error *(accumulated)*

- Error depends on the step size h
  - local error is $O(h^{p+1})$ $\longrightarrow$ global error is $O(h^p)$, method is of order p
  - explicit Euler makes $O(h^2)$ error per step: order 1

Analysis: Accuracy

- Numerical integration is inaccurate
- Error accumulates
- Error can cause instability

$$\underbrace{\boxed{y(t+h) = y(t) + hy'(t)}}_{\textbf{Euler step}} + \underbrace{\boxed{O(h^2)}}_{\textbf{error}}$$



How can we reduce error?

- reduce step size
- *improve accuracy*

Analysis: Higher Accuracy

- Taylor expansion (higher oder)

$$y(t+h) = y(t) + hy'(t) + \frac{1}{2}h^2y''(t) + O(h^3)$$

- Higher order integration schemes
  - *midpoint method*:
    accuracy: **order 2**, cost: **2 x** evaluations of f
  - *4th-order Runge-Kutta method (RK4):*
    accuracy: **order 4**, cost: **4 x** evaluations of f

Analysis: Stability

- Analyze
  - test equation $$y' = \lambda y \quad y(0) = 1 \quad \lambda < 0 \;\; t \geq 0$$
  - explicit Euler $$y_{n+1} = y_n + h\lambda y_n = (1 + h\lambda)\, y_n$$
- Solve recursion $$y_{n+1} = (1 + h\lambda)^{n+1}\, y_0$$

$$y_{n+1} < \infty \iff |1 + h\lambda| < 1$$

restricted step size *(explicit Euler)*

Analysis: Stability

- *Observations from test equation:* explicit Euler
  - requires small time steps for stable integration
  - inefficient since step size is determined by stability, not accuracy requirement
- Problems with this characteristic are termed *stiff*
- Do not use *explicit* methods for stiff problems, use *implicit* methods instead:
  - explicit methods: y_{n+1} expressed with *known* quantities (e.g, y_n , $f(t_n, y_n)$)
  - implicit methods: y_{n+1} expressed with un*known* quantities (e.g, $f(t_{n+1}, y_{n+1})$)

Analysis: Implicit Euler

How do we get from $y(t)$ *to* $y(t+h)$*?*

- Fundamental theorem of calculus

$$y(t+h) = y(t) + \int_t^{t+h} f(t, y(t))\mathrm{d}t$$

$$y(t+h) \approx y(t) + hf(t+h, y(t+h))$$

right-hand rectangle method

- Taylor expansion (1st order approximation)

$$y(t_{n+1} - h) \approx y(t_{n+1}) - hy'(t_{n+1}) + O(h^2)$$

“backward” Euler

Analysis: Stability

- Analyze
  - test equation $y' = \lambda y \quad y(0) = 1 \quad \lambda < 0 \;\; t \geq 0$
  - implicit Euler $y_{n+1} = y_n + h\lambda y_{n+1} \longrightarrow y_{n+1} = \frac{1}{1 - h\lambda} y_n$
- Solve recursion

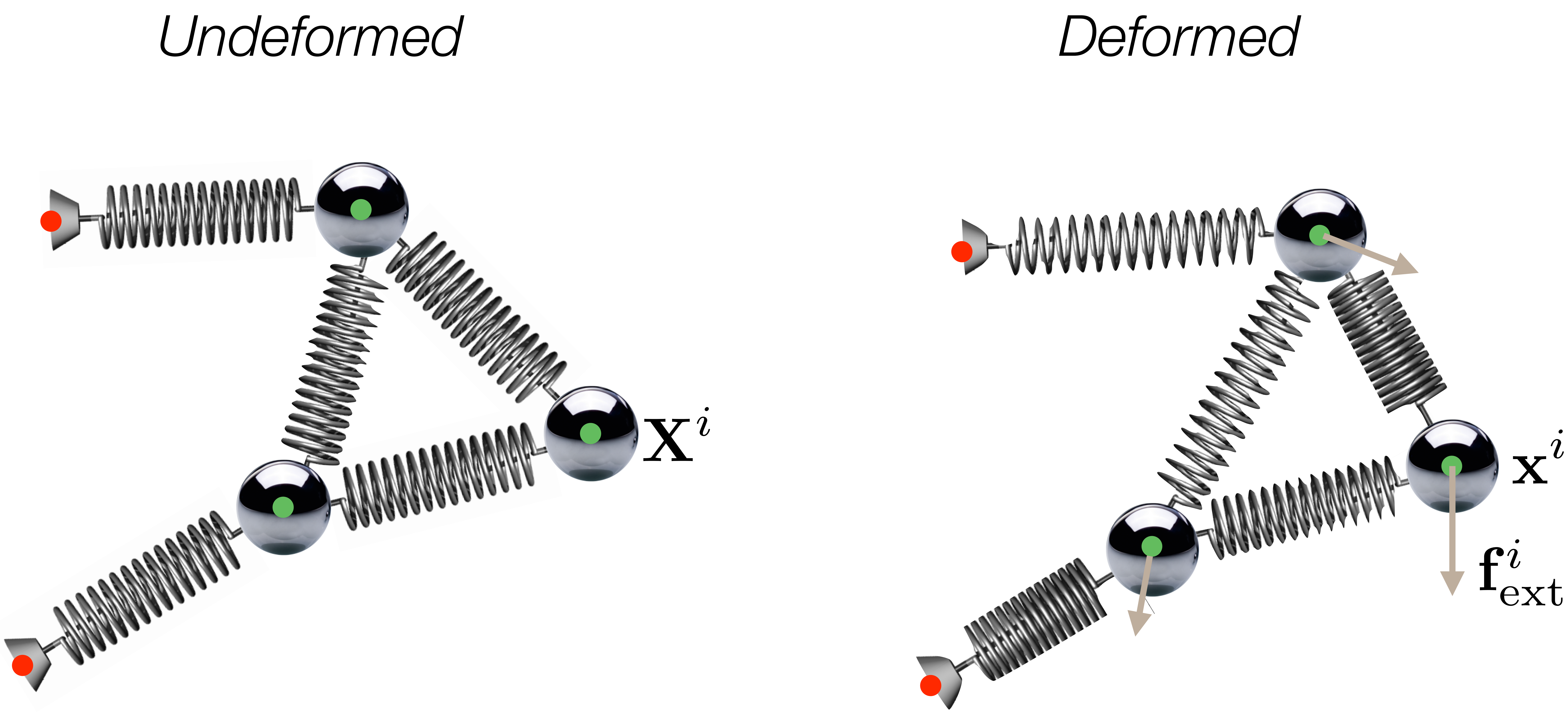
$$y_{n+1} = \left(\frac{1}{1 - h\lambda} \right)^{n+1} y_0$$

implicit Euler
is stable for all
$h > 0$

Agenda

- Motivation
- Energy, forces, static vs. dynamic analysis
- Numerical time integration (explicit vs. implicit schemes)
- Assembly: energy, forces, stiffness matrix
- Continuum mechanics: strain, stress, material models
- Linear vs. nonlinear FEM (Finite Element Method)

Static Analysis

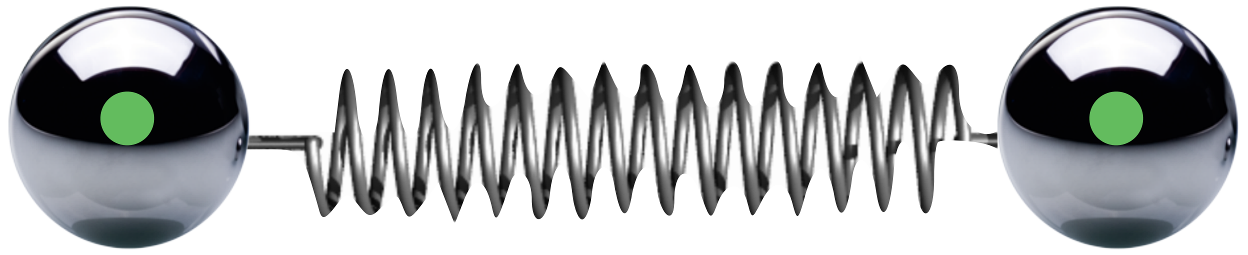


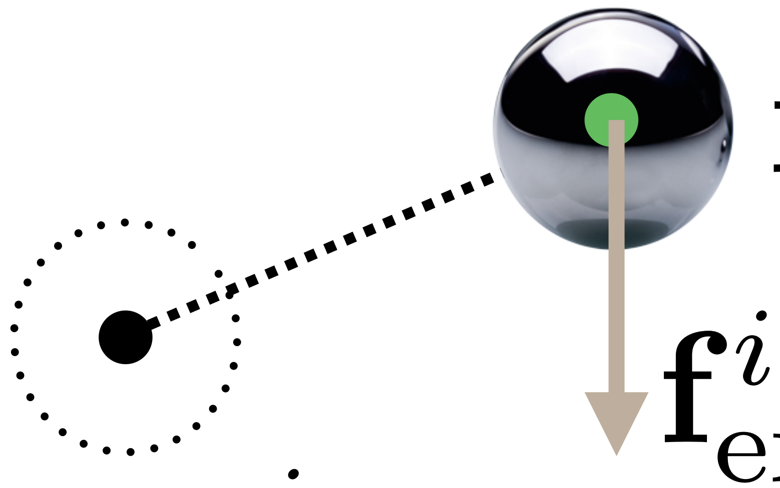
Static Analysis

- Energy

$$
\begin{aligned}
E(\mathbf{x}) &= E_{\text{int}}(\mathbf{x}) - E_{\text{ext}}(\mathbf{x}) \\
&= \sum_{(i,j)} E_{\text{int}}^{(i,j)}(\mathbf{x}^i, \mathbf{x}^j) - \sum_i E_{\text{ext}}^i(\mathbf{x}^i) \\
&= \sum_{(i,j)} \frac{1}{2} k \left(\|\mathbf{x}^i - \mathbf{x}^j\| - L \right)^2 - \sum_i \left(\mathbf{f}_{\text{ext}}^i \right)^T \left(\mathbf{x}^i - \mathbf{X}^i \right)
\end{aligned}
$$

$$
\mathbf{x} = \begin{bmatrix} \vdots \\ \mathbf{x}^i \\ \vdots \end{bmatrix} \in \mathbb{R}^{3n}
$$

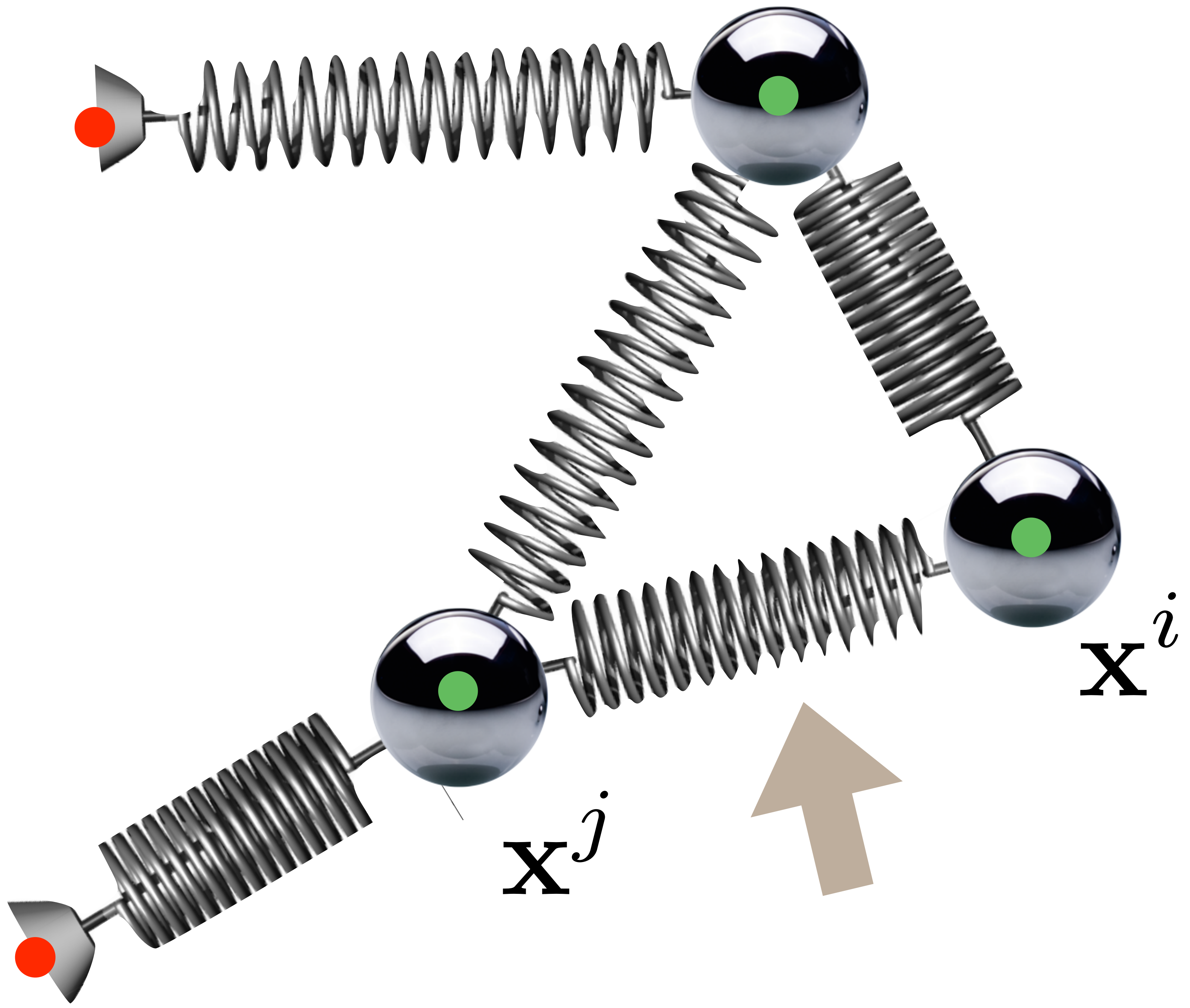




Static Analysis

- Forces $\nabla E(\mathbf{x}) = \nabla E_{\text{int}}(\mathbf{x}) - \nabla E_{\text{ext}}(\mathbf{x})$

$$= \sum_{(i,j)} \nabla E_{\text{int}}^{(i,j)}(\mathbf{x}^i, \mathbf{x}^j) - \sum_i \nabla E_{\text{ext}}^i(\mathbf{x}^i)$$



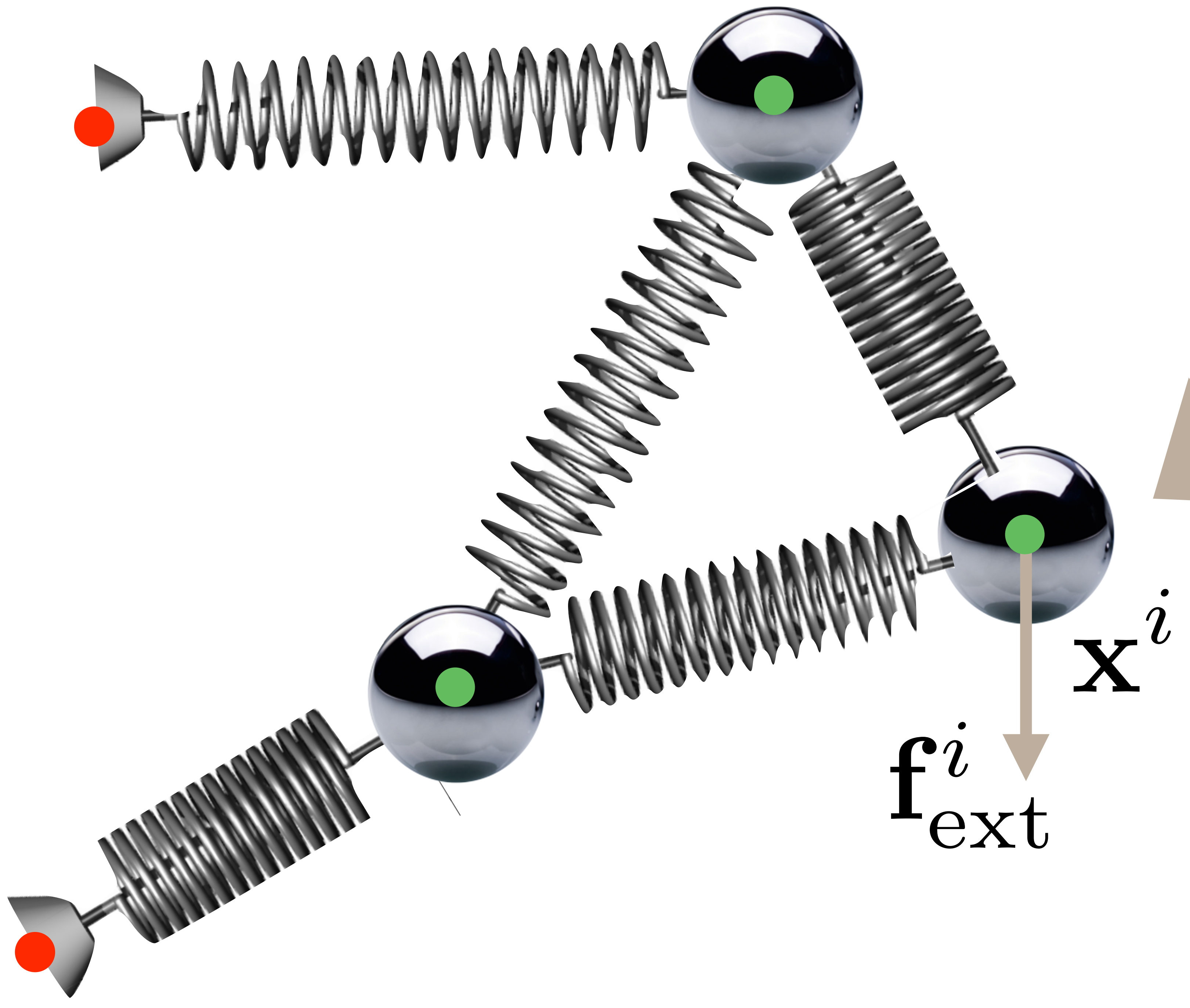
$$\frac{\partial E_{\text{int}}^{(i,j)}(\mathbf{x}^i, \mathbf{x}^j)}{\partial \mathbf{x}^i} = +k\left(\|\mathbf{x}^i - \mathbf{x}^j\| - L\right)\frac{\mathbf{x}^i - \mathbf{x}^j}{\|\mathbf{x}^i - \mathbf{x}^j\|}$$

$$\frac{\partial E_{\text{int}}^{(i,j)}(\mathbf{x}^i, \mathbf{x}^j)}{\partial \mathbf{x}^j} = -k\left(\|\mathbf{x}^i - \mathbf{x}^j\| - L\right)\frac{\mathbf{x}^i - \mathbf{x}^j}{\|\mathbf{x}^i - \mathbf{x}^j\|}$$

Static Analysis

- Forces $\nabla E(\mathbf{x}) = \nabla E_{\text{int}}(\mathbf{x}) - \nabla E_{\text{ext}}(\mathbf{x})$

$$= \sum_{(i,j)} \nabla E_{\text{int}}^{(i,j)}(\mathbf{x}^i, \mathbf{x}^j) - \sum_i \nabla E_{\text{ext}}^i(\mathbf{x}^i)$$

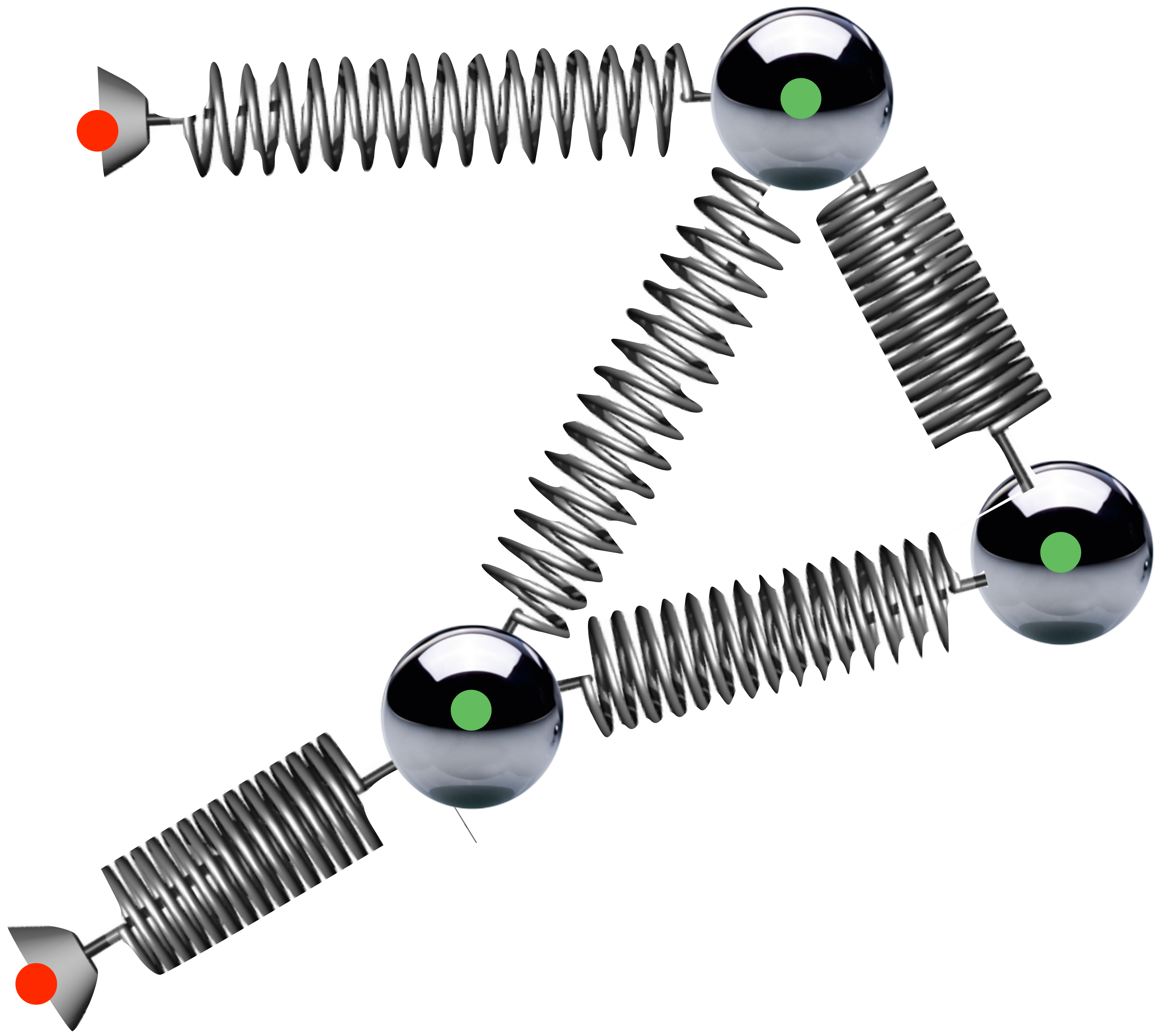


$$\frac{\partial E_{\text{ext}}^i(\mathbf{x}^i)}{\partial \mathbf{x}^i} = \mathbf{f}_{\text{ext}}^i$$

Static Analysis

- Forces $\nabla E(\mathbf{x}) = \nabla E_{\text{int}}(\mathbf{x}) - \nabla E_{\text{ext}}(\mathbf{x})$

$$= \sum_{(i,j)} \nabla E_{\text{int}}^{(i,j)}(\mathbf{x}^i, \mathbf{x}^j) - \sum_i \nabla E_{\text{ext}}^i(\mathbf{x}^i)$$

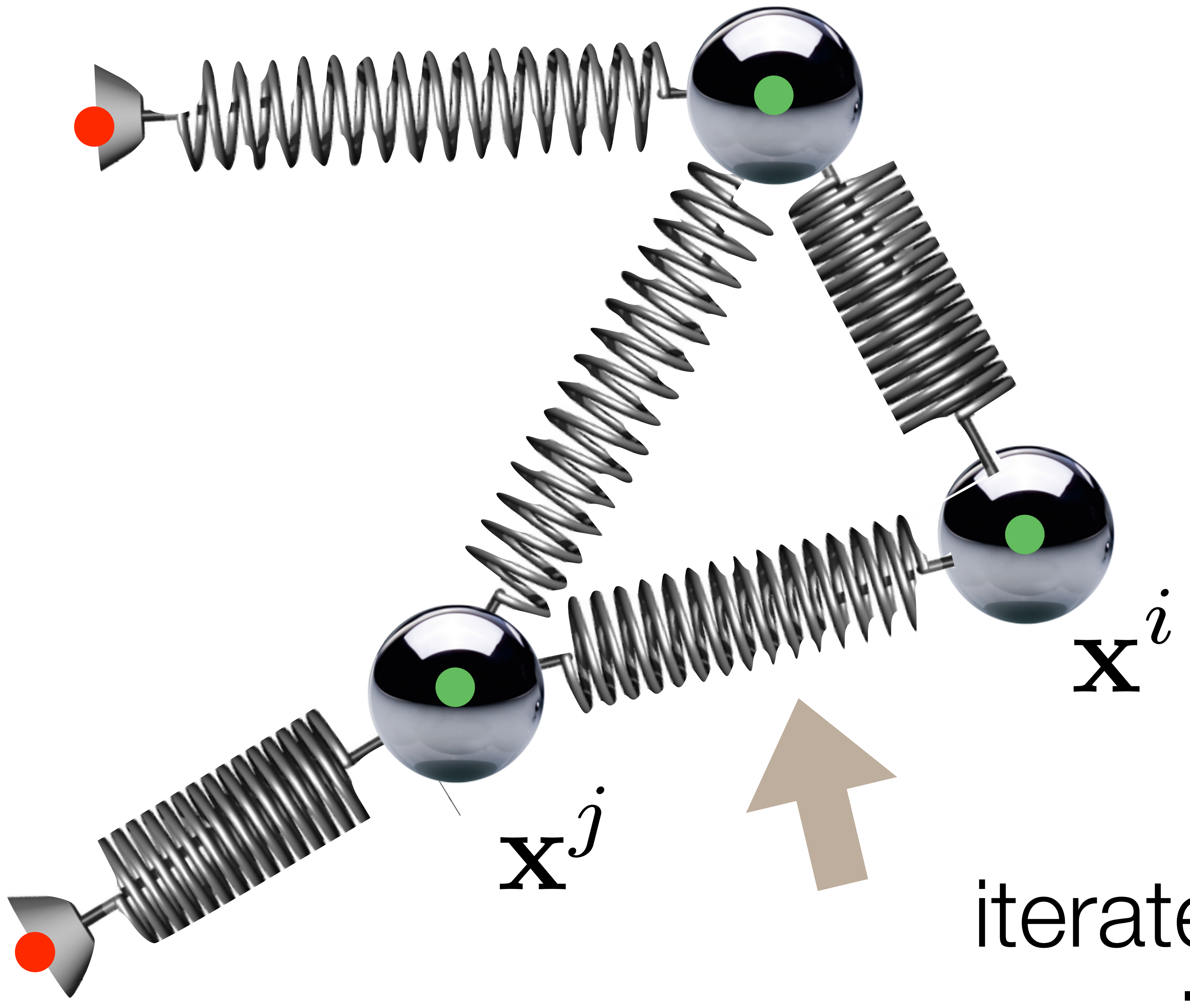


$$\nabla E(\mathbf{x}) = \begin{matrix} = \mathbf{0} \\ = \mathbf{0} \\ = \mathbf{0} \\ = \mathbf{0} \\ = \mathbf{0} \end{matrix}$$

Static Analysis

- Forces $\nabla E(\mathbf{x}) = \nabla E_{\text{int}}(\mathbf{x}) - \nabla E_{\text{ext}}(\mathbf{x})$

$$= \sum_{(i,j)} \nabla E_{\text{int}}^{(i,j)}(\mathbf{x}^i, \mathbf{x}^j) - \sum_i \nabla E_{\text{ext}}^i(\mathbf{x}^i)$$



iterate
over springs

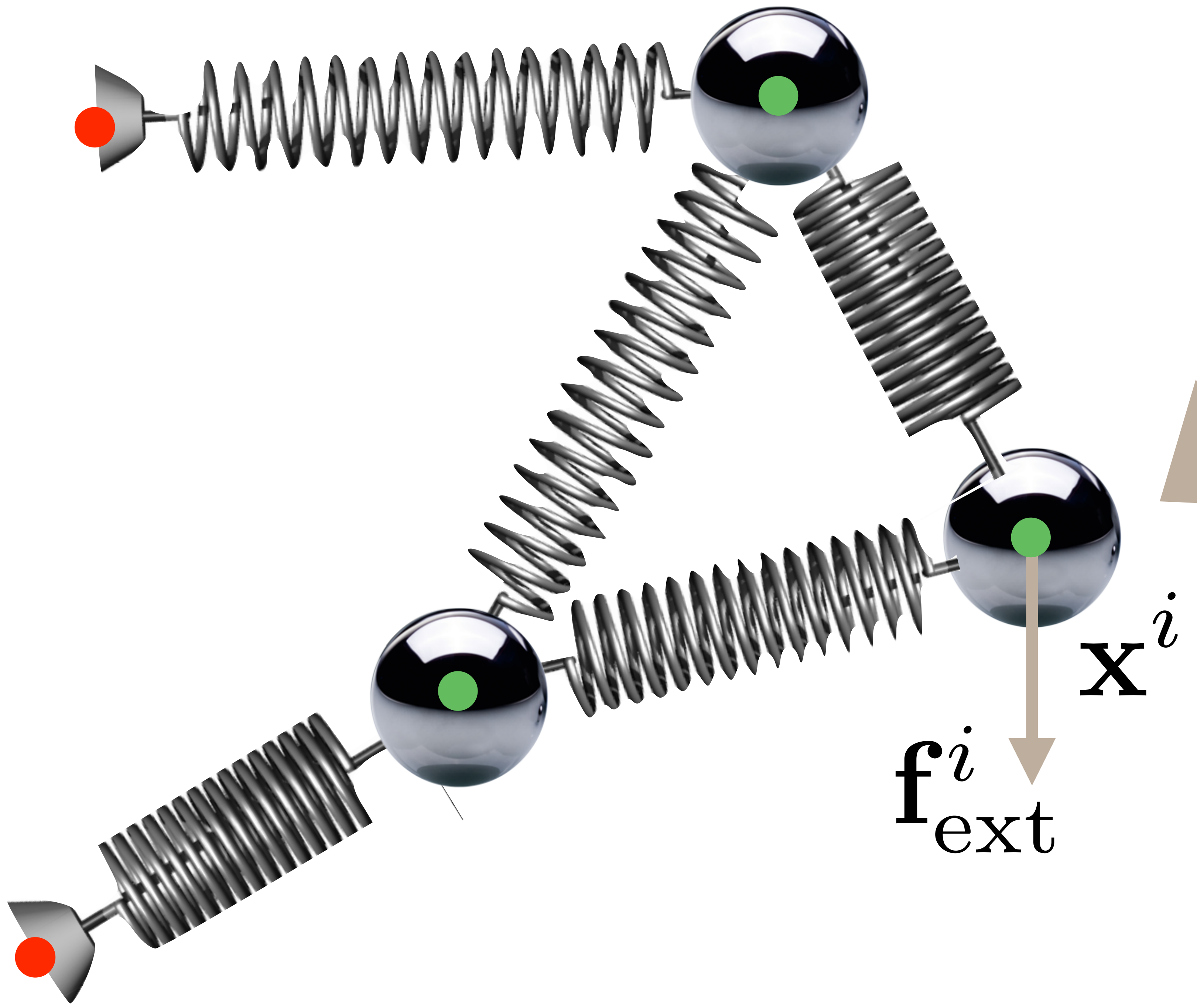
$$\nabla E(\mathbf{x}) = \quad \square \mathrel{+}= \frac{\partial E_{\text{int}}^{(i,j)}(\mathbf{x}^i, \mathbf{x}^j)}{\partial \mathbf{x}^i}$$

$$\square \mathrel{+}= \frac{\partial E_{\text{int}}^{(i,j)}(\mathbf{x}^i, \mathbf{x}^j)}{\partial \mathbf{x}^j}$$

Static Analysis

- Forces $\nabla E(\mathbf{x}) = \nabla E_{\text{int}}(\mathbf{x}) - \nabla E_{\text{ext}}(\mathbf{x})$

$$= \sum_{(i,j)} \nabla E_{\text{int}}^{(i,j)}(\mathbf{x}^i, \mathbf{x}^j) - \sum_i \nabla E_{\text{ext}}^i(\mathbf{x}^i)$$



iterate
over masses

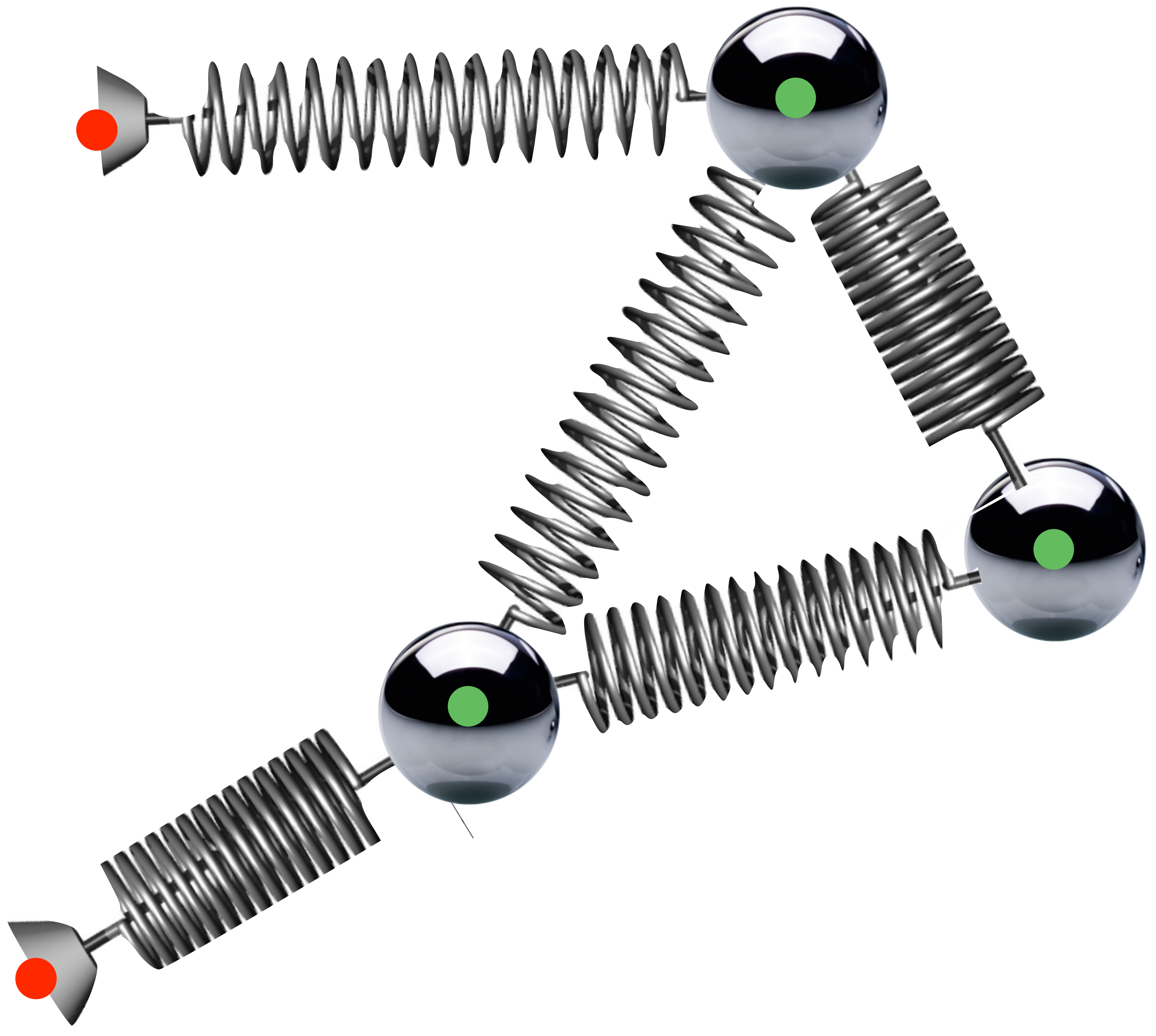
$$- = \frac{\partial E^i_{\text{ext}}(\mathbf{x}^i)}{\partial \mathbf{x}^i}$$

$$\nabla E(\mathbf{x}) =$$

Static Analysis

- Stiffness matrix $\nabla^2 E(\mathbf{x}) = \nabla^2 E_{\text{int}}(\mathbf{x}) - \nabla^2 E_{\text{ext}}(\mathbf{x})$

$$= \sum_{(i,j)} \nabla^2 E_{\text{int}}^{(i,j)}(\mathbf{x}^i, \mathbf{x}^j)$$

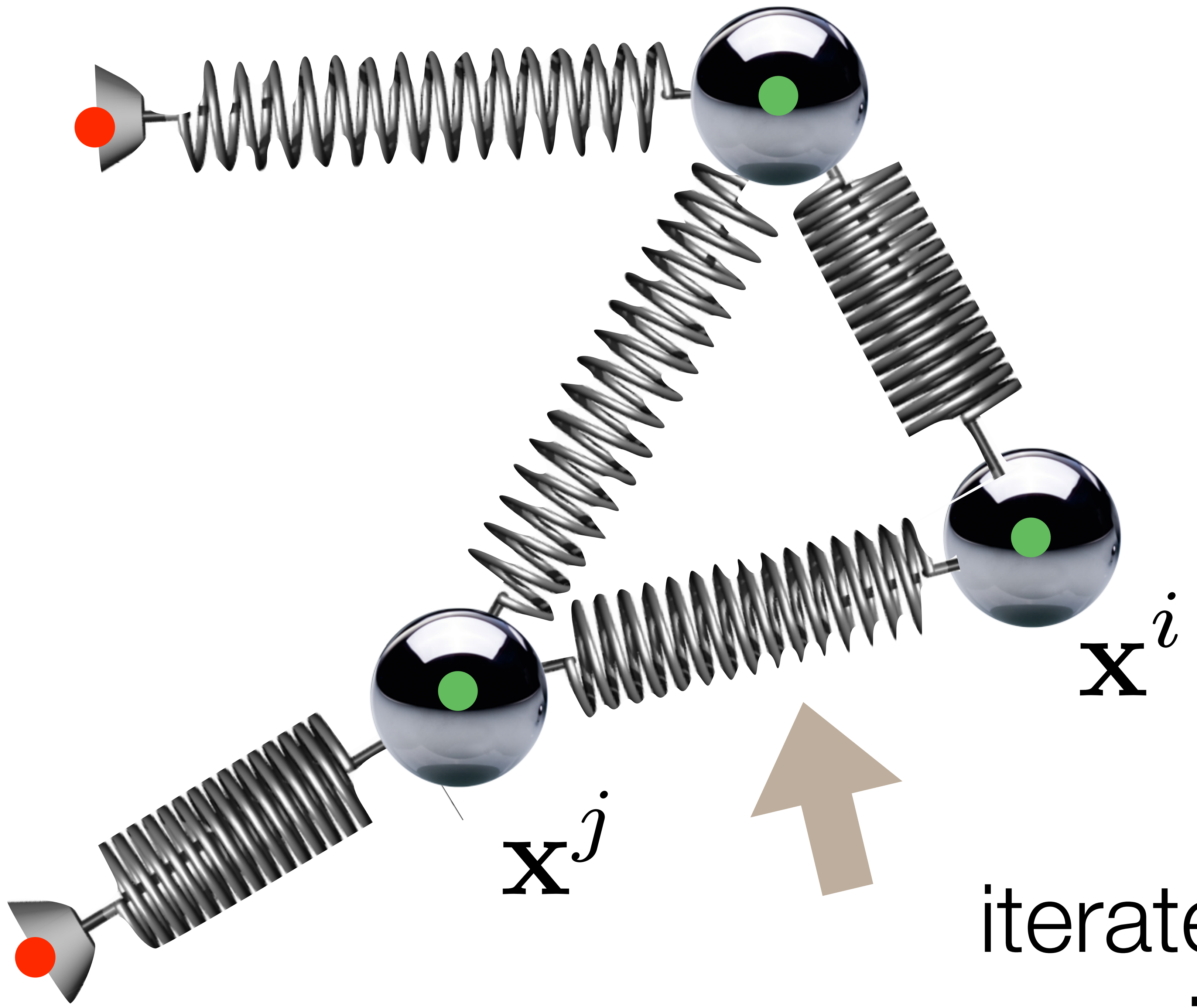


$$\nabla^2 E(\mathbf{x}) = \begin{matrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{matrix}$$

Static Analysis

- Stiffness matrix $\nabla^2 E(\mathbf{x}) = \nabla^2 E_{\text{int}}(\mathbf{x}) - \nabla^2 E_{\text{ext}}(\mathbf{x})$

$$= \sum_{(i,j)} \nabla^2 E_{\text{int}}^{(i,j)}(\mathbf{x}^i, \mathbf{x}^j)$$



iterate
over springs

$$\nabla^2 E(\mathbf{x}) =$$

$$+= \frac{\partial^2 E_{\text{int}}^{(i,j)}(\mathbf{x}^i, \mathbf{x}^j)}{\partial \mathbf{x}^i \partial \mathbf{x}^i}$$

$$+= \frac{\partial^2 E_{\text{int}}^{(i,j)}(\mathbf{x}^i, \mathbf{x}^j)}{\partial \mathbf{x}^i \partial \mathbf{x}^j}$$

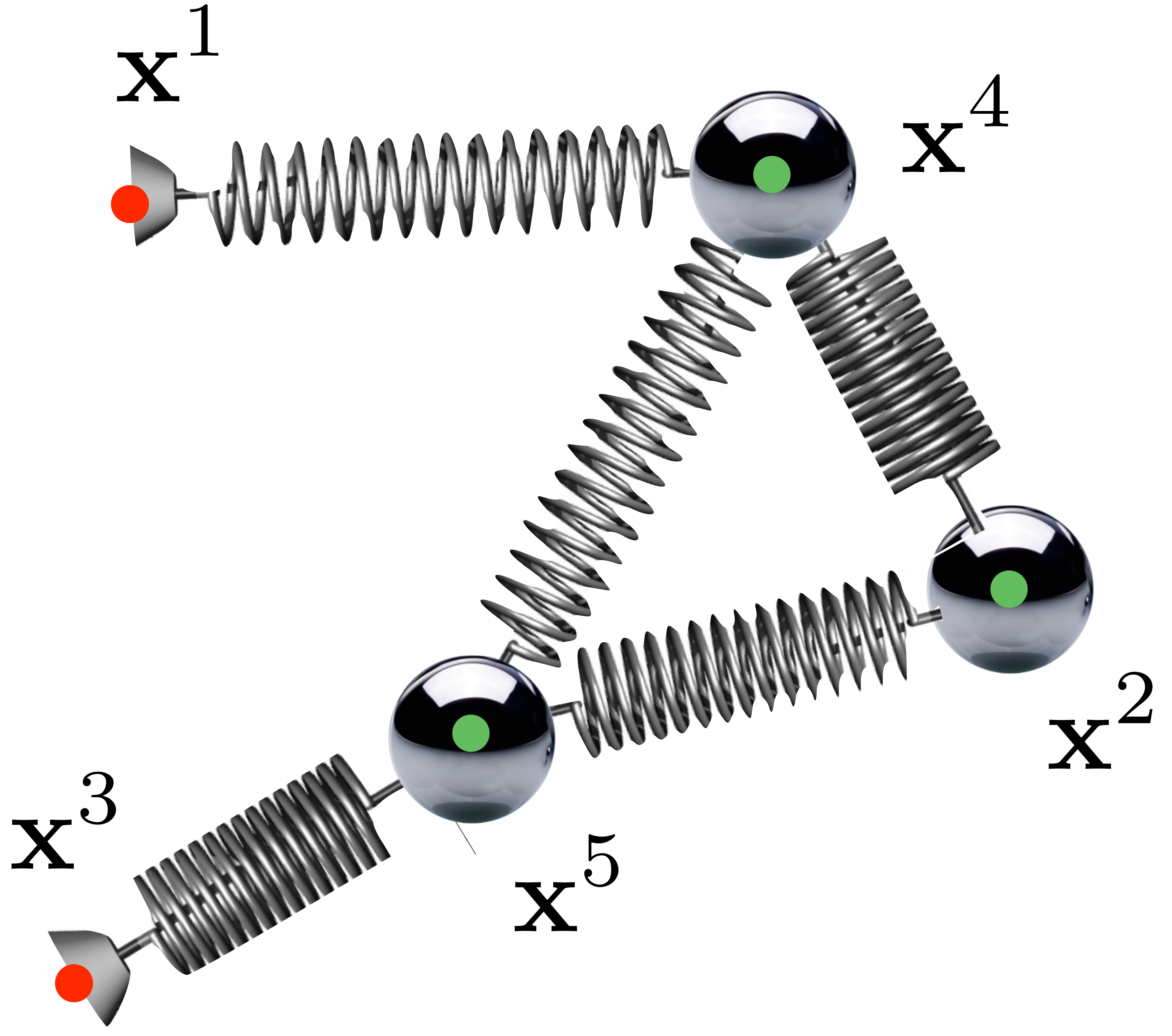
$$+= \frac{\partial^2 E_{\text{int}}^{(i,j)}(\mathbf{x}^i, \mathbf{x}^j)}{\partial \mathbf{x}^j \partial \mathbf{x}^i}$$

$$+= \frac{\partial^2 E_{\text{int}}^{(i,j)}(\mathbf{x}^i, \mathbf{x}^j)}{\partial \mathbf{x}^j \partial \mathbf{x}^j}$$

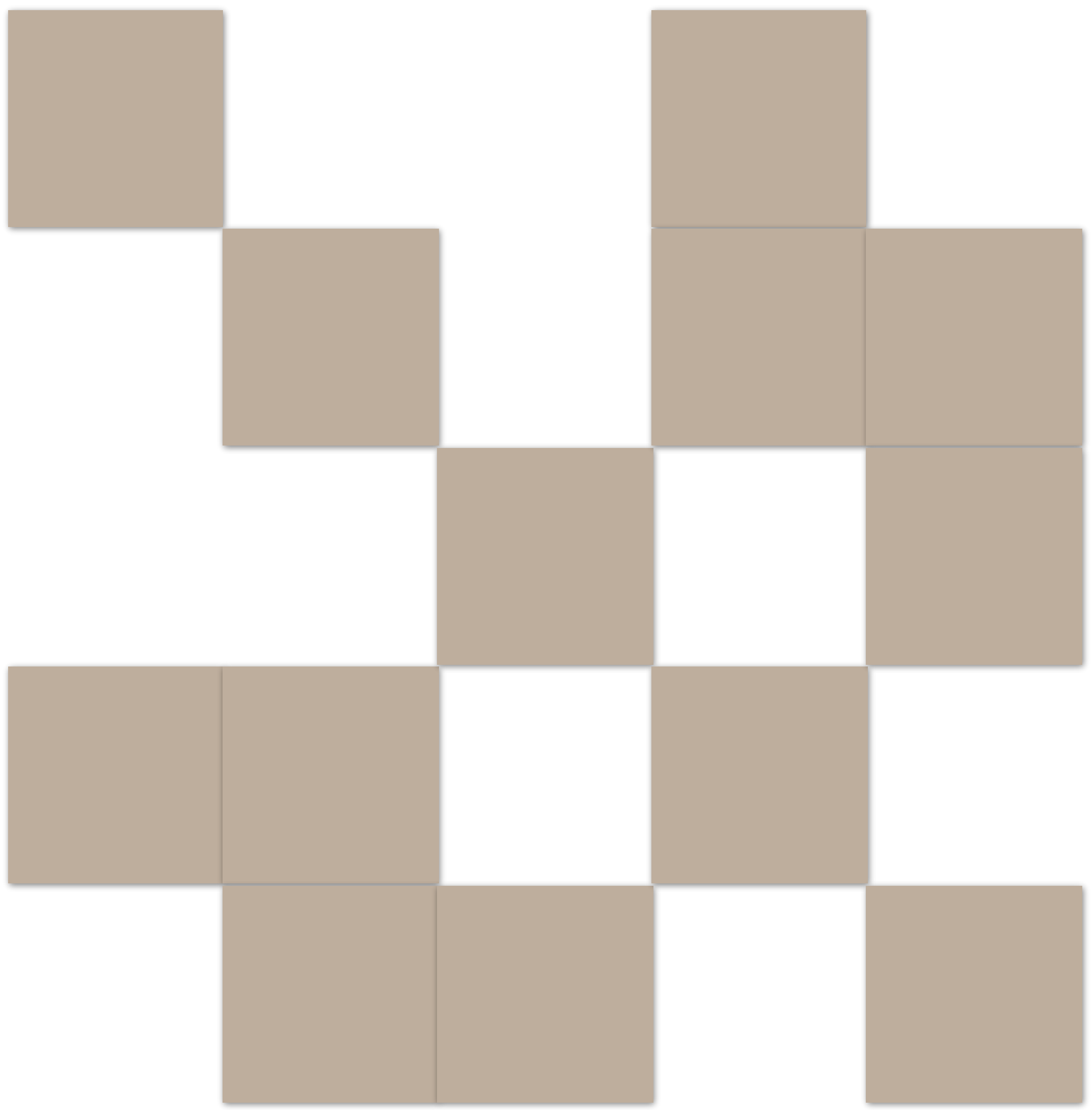
Static Analysis

- Stiffness matrix $\nabla^2 E(\mathbf{x}) = \nabla^2 E_{\text{int}}(\mathbf{x}) - \nabla^2 E_{\text{ext}}(\mathbf{x})$

$$= \sum_{(i,j)} \nabla^2 E_{\text{int}}^{(i,j)}(\mathbf{x}^i, \mathbf{x}^j)$$



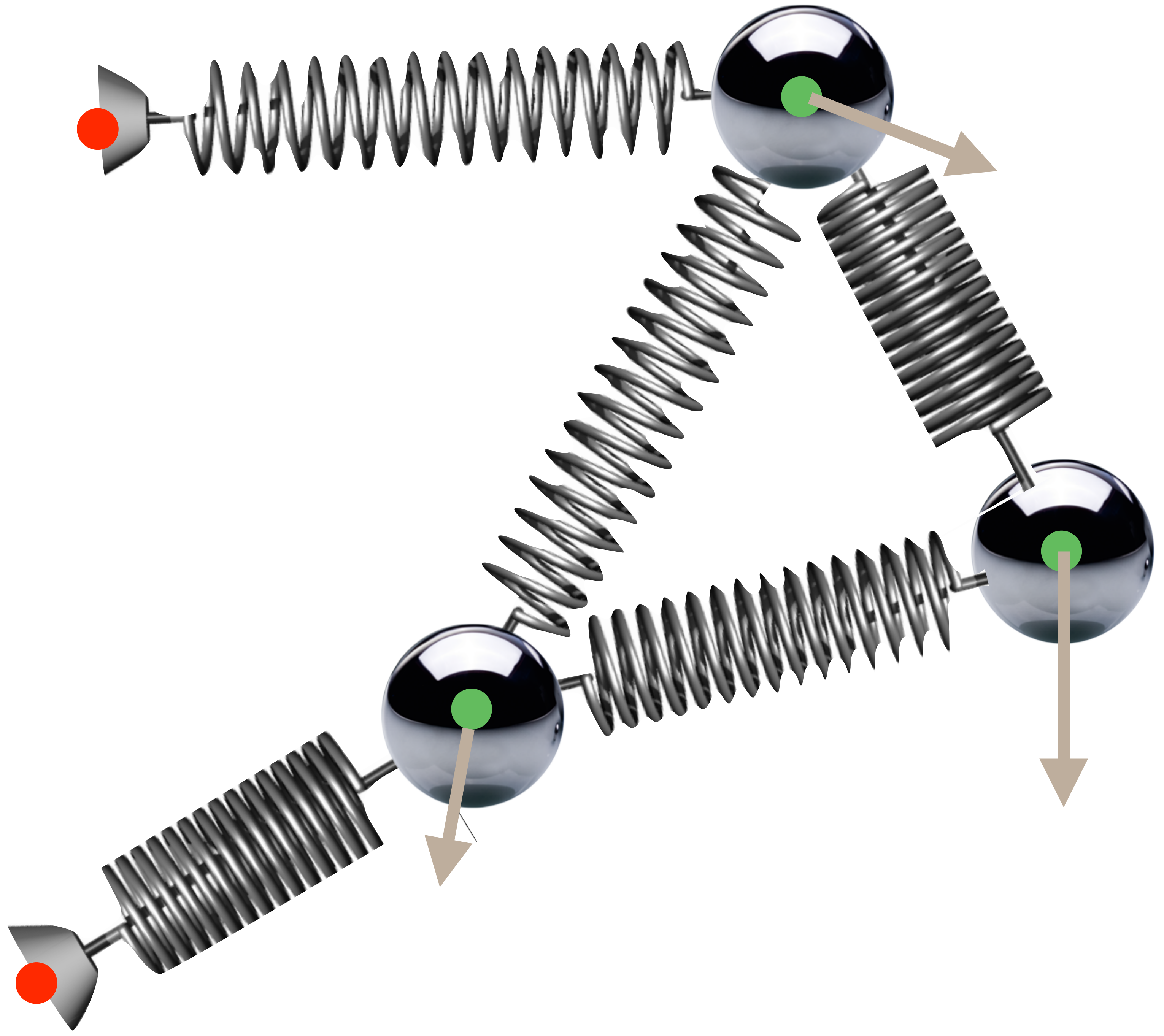
$\nabla^2 E(\mathbf{x}) =$



sparse matrix

symmetric matrix

Static Analysis



- Minimize energy

$$\min_{\mathbf{x}} f_{\text{static}}(\mathbf{x}) \qquad f_{\text{static}}(\mathbf{x}) = E(\mathbf{x})$$
$$= E_{\text{int}}(\mathbf{x}) - E_{\text{ext}}(\mathbf{x})$$

- Minimum $\mathbf{x}^*$
  - gradient: zero $\qquad \nabla E(\mathbf{x}^*) \stackrel{!}{=} \mathbf{o}$
  - Hessian: positive definite

$$\forall \mathbf{p} \neq \mathbf{o} : \mathbf{p}^T \nabla^2 E(\mathbf{x}^*)\mathbf{p} > 0$$

- Static equilibrium

$$\nabla E(\mathbf{x}^*) = \mathbf{f}_{\text{int}}(\mathbf{x}^*) - \mathbf{f}_{\text{ext}} \stackrel{!}{=} \mathbf{o}$$

Dynamic Analysis

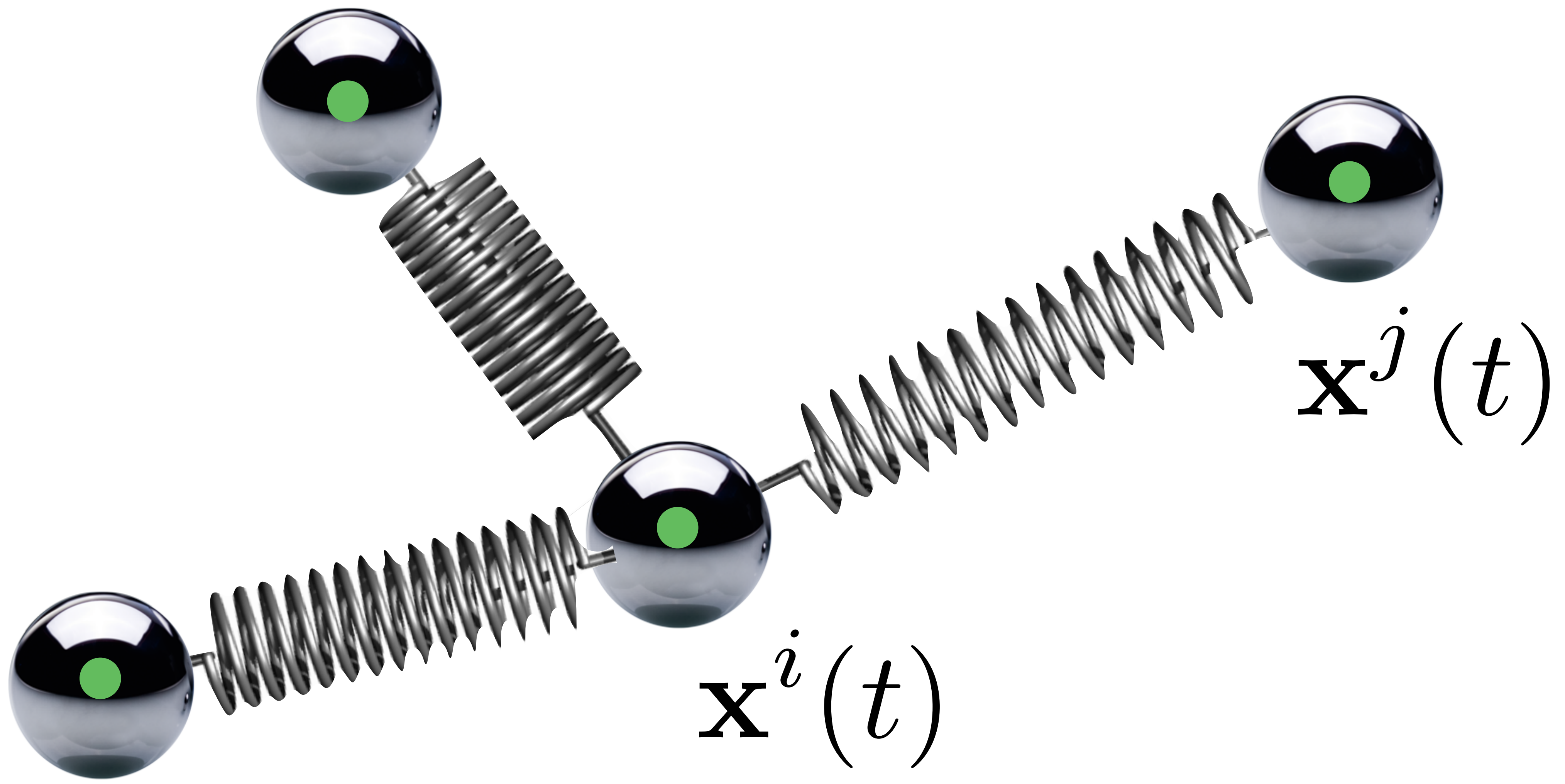
$$m^i \frac{\mathrm{d}^2\mathbf{x}^i(t)}{\mathrm{d}t^2} + \gamma \frac{\mathrm{d}\mathbf{x}^i(t)}{\mathrm{d}t} = -\sum_j \mathbf{f}_{\mathrm{int}}^{(i,j)}(\mathbf{x}^i(t), \mathbf{x}^j(t)) + \mathbf{f}_{\mathrm{ext}}^i$$

2nd order ordinary differential equation (ODE)

$$\mathbf{x}^i(t_0) = \mathbf{x}_0^i \qquad \frac{\mathrm{d}\mathbf{x}^i(t_0)}{\mathrm{d}t} = \mathbf{v}_o^i$$

Initial value problem (IVP)

How do we determine motion $\mathbf{x}^i(t)$*?*



Dynamic Analysis

- Explicit Euler

$$\mathbf{x}^i(t+h) = \mathbf{x}^i(t) + h\mathbf{v}^i(t)$$

$$\mathbf{v}^i(t+h) = \mathbf{v}^i(t) + h\mathbf{a}^i(t)$$

$$\mathbf{a}^i(t) = \frac{1}{m^i}\left(-\sum_j \mathbf{f}_{\text{int}}^{(i,j)}\left(\mathbf{x}^i(t), \mathbf{x}^j(t)\right) + \mathbf{f}_{\text{ext}}^i - \gamma\mathbf{v}^i(t)\right)$$

Dynamic Analysis

- Implicit Euler

$$\mathbf{x}^i(t+h) = \mathbf{x}^i(t) + h\mathbf{v}^i(t+h)$$

$$\mathbf{v}^i(t+h) = \mathbf{v}^i(t) + h\mathbf{a}^i(t+h)$$

$$\mathbf{a}^i(t+h) = \frac{1}{m^i}\left(-\sum_j \mathbf{f}_{\text{int}}^{(i,j)}\left(\mathbf{x}^i(t+h), \mathbf{x}^j(t+h)\right) + \mathbf{f}_{\text{ext}}^i - \gamma\mathbf{v}^i(t+h)\right)$$

multiply both sides with m^i

Dynamic Analysis

- Implicit Euler

$$\mathbf{x}^i(t+h) = \mathbf{x}^i(t) + h\mathbf{v}^i(t+h)$$

$$\mathbf{v}^i(t+h) = \mathbf{v}^i(t) + h\mathbf{a}^i(t+h)$$

$$m^i\mathbf{a}^i(t+h) = -\sum_j \mathbf{f}_{\text{int}}^{(i,j)}\left(\mathbf{x}^i(t+h), \mathbf{x}^j(t+h)\right) + \mathbf{f}_{\text{ext}}^i - \gamma\mathbf{v}^i(t+h)$$

Newton's 2nd law

move internal and damping forces to left-hand side

Dynamic Analysis

- Implicit Euler

$$\mathbf{x}^i(t+h) = \mathbf{x}^i(t) + h\mathbf{v}^i(t+h)$$

$$\mathbf{v}^i(t+h) = \mathbf{v}^i(t) + h\mathbf{a}^i(t+h)$$

$$m^i\mathbf{a}^i(t+h) + \sum_j \mathbf{f}_{\text{int}}^{(i,j)}\left(\mathbf{x}^i(t+h), \mathbf{x}^j(t+h)\right) + \gamma\mathbf{v}^i(t+h) = \mathbf{f}_{\text{ext}}^i$$

“dynamic” equilibrium

Dynamic Analysis

- Implicit Euler

$$\mathbf{x}(t+h) = \mathbf{x}(t) + h\mathbf{v}(t+h)$$

$$\mathbf{v}(t+h) = \mathbf{v}(t) + h\mathbf{a}(t+h)$$

$$\mathbf{M}\mathbf{a}(t+h) + \nabla E_{\text{int}}(\mathbf{x}(t+h)) + \gamma\mathbf{v}(t+h) = \mathbf{f}_{\text{ext}}$$

$$\mathbf{M} = \underbrace{\begin{bmatrix} \ddots & & & & \\ & m^i & & & \\ & & m^i & & \\ & & & m^i & \\ & & & & \ddots \end{bmatrix}}_{\mathbb{R}^{3n\times 3n}} \quad \mathbf{x} = \begin{bmatrix} \vdots \\ \mathbf{x}^i \\ \vdots \end{bmatrix} \quad \mathbf{v} = \begin{bmatrix} \vdots \\ \mathbf{v}^i \\ \vdots \end{bmatrix} \quad \mathbf{a} = \begin{bmatrix} \vdots \\ \mathbf{a}^i \\ \vdots \end{bmatrix} \quad \mathbf{f}_{\text{ext}} = \begin{bmatrix} \vdots \\ \mathbf{f}^i_{\text{ext}} \\ \vdots \end{bmatrix}$$

$$\mathbb{R}^{3n}$$

Dynamic Analysis

- Implicit Euler

$$\mathbf{x}_n = \mathbf{x}_p + h\mathbf{v}_n$$

$$\mathbf{v}_n = \mathbf{v}_p + h\mathbf{a}_n$$

p previous, *known*

n next, *unknown*

$$\mathbf{M}\mathbf{a}_n + \nabla E_{\text{int}}(\mathbf{x}_n) + \gamma\mathbf{v}_n = \mathbf{f}_{\text{ext}}$$

Dynamic Analysis

- Implicit Euler

$$\mathbf{x}_n = \mathbf{x}_p + h\mathbf{v}_n$$

$$\mathbf{v}_n = \mathbf{v}_p + h\mathbf{a}_n$$

p previous, *known*

n next, *unknown*

$$\mathbf{M}\mathbf{a}_n + \nabla E_{\text{int}}(\mathbf{x}_n) + \gamma\mathbf{v}_n = \mathbf{f}_{\text{ext}}$$

$$\mathbf{x}_n = \mathbf{x}_p + h\mathbf{v}_n \longrightarrow \mathbf{v}_n(\mathbf{x}_n) = \frac{\mathbf{x}_n - \mathbf{x}_p}{h}$$

$$\mathbf{v}_n = \mathbf{v}_p + h\mathbf{a}_n \longrightarrow \mathbf{a}_n(\mathbf{x}_n) = \frac{\mathbf{v}_n(\mathbf{x}_n) - \mathbf{v}_p}{h} = \frac{\mathbf{v}_n(\mathbf{x}_n)}{h} - \frac{\mathbf{v}_p}{h} = \frac{\mathbf{x}_n - \mathbf{x}_p}{h^2} - \frac{\mathbf{v}_p}{h}$$

Dynamic Analysis

- Implicit Euler

$$\mathbf{M}\mathbf{a}_n(\mathbf{x}_n^*) + \nabla E_{\text{int}}(\mathbf{x}_n^*) + \gamma \mathbf{v}_n(\mathbf{x}_n^*) - \mathbf{f}_{\text{ext}} \stackrel{!}{=} \mathbf{o}$$

Find $\mathbf{x}_n^*$ that fulfills this "dynamic" equilibrium.

$$\mathbf{v}_n(\mathbf{x}_n) = \frac{\mathbf{x}_n - \mathbf{x}_p}{h}$$

$$\mathbf{a}_n(\mathbf{x}_n) = \frac{\mathbf{v}_n(\mathbf{x}_n) - \mathbf{v}_p}{h} = \frac{\mathbf{v}_n(\mathbf{x}_n)}{h} - \frac{\mathbf{v}_p}{h} = \frac{\mathbf{x}_n - \mathbf{x}_p}{h^2} - \frac{\mathbf{v}_p}{h}$$

Dynamic Analysis

- Implicit Euler

$$\min_{\mathbf{x}_n} f_{\text{dynamic}}(\mathbf{x}_n)$$

$$\begin{aligned} f_{\text{dynamic}}(\mathbf{x}_n) = & \frac{h^2}{2}\left(\mathbf{a}_n(\mathbf{x}_n)\right)^T \mathbf{M}\mathbf{a}_n(\mathbf{x}_n) && \text{“inertia”} \\ & + E_{\text{int}}(\mathbf{x}_n) && \text{internal energy} \\ & + \frac{h}{2}\gamma\left(\mathbf{v}_n(\mathbf{x}_n)\right)^T \mathbf{v}_n(\mathbf{x}_n) && \text{“damping”} \\ & - \mathbf{f}_{\text{ext}}^T(\mathbf{x}_n - \mathbf{X}) && \text{external energy} \end{aligned}$$

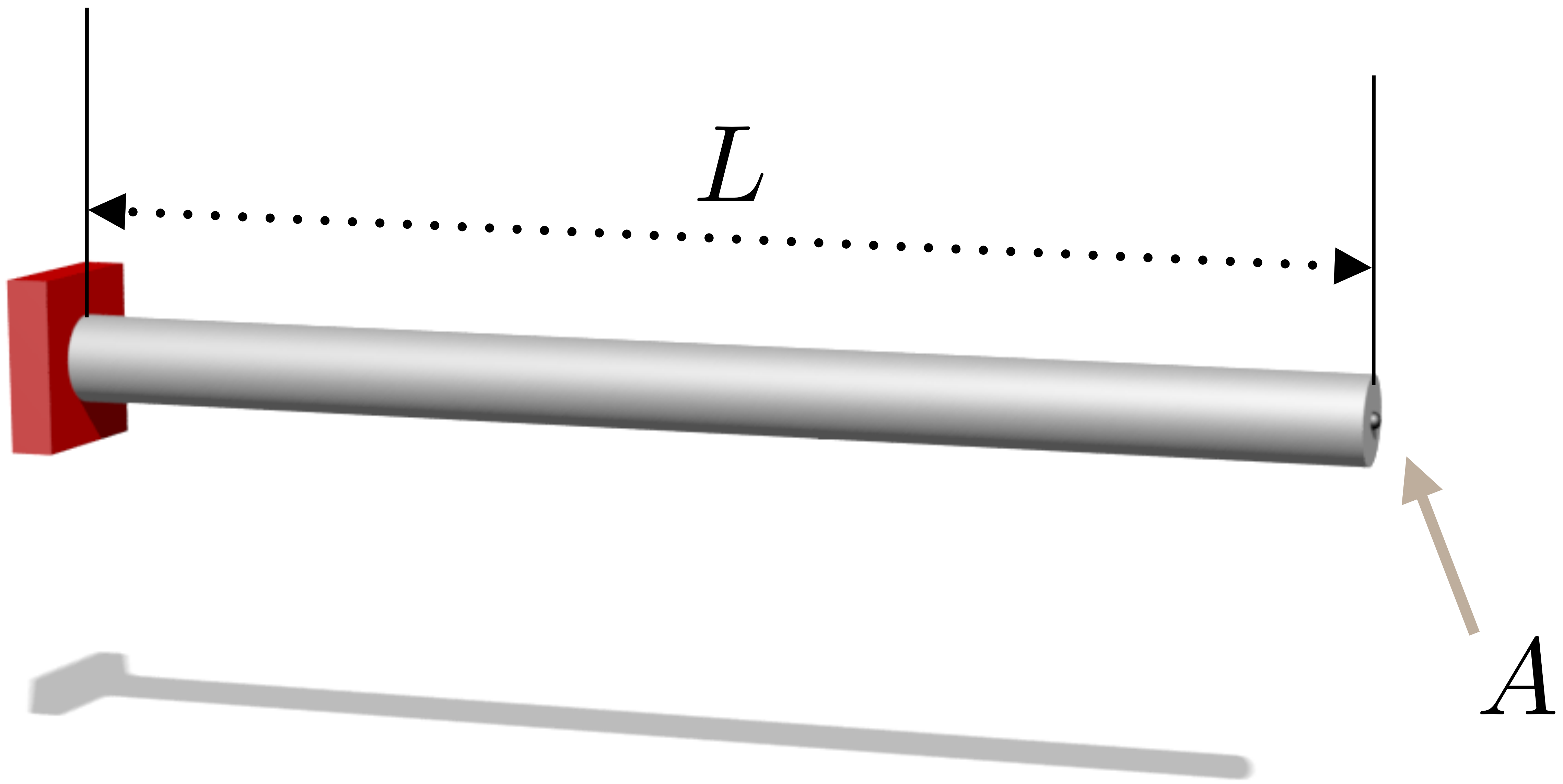
Agenda

- Motivation
- Energy, forces, static vs. dynamic analysis
- Numerical time integration (explicit vs. implicit schemes)
- Assembly: energy, forces, stiffness matrix
- Continuum mechanics: strain, stress, material models
- Linear vs. nonlinear FEM (Finite Element Method)

Elastic Rod

L rest length

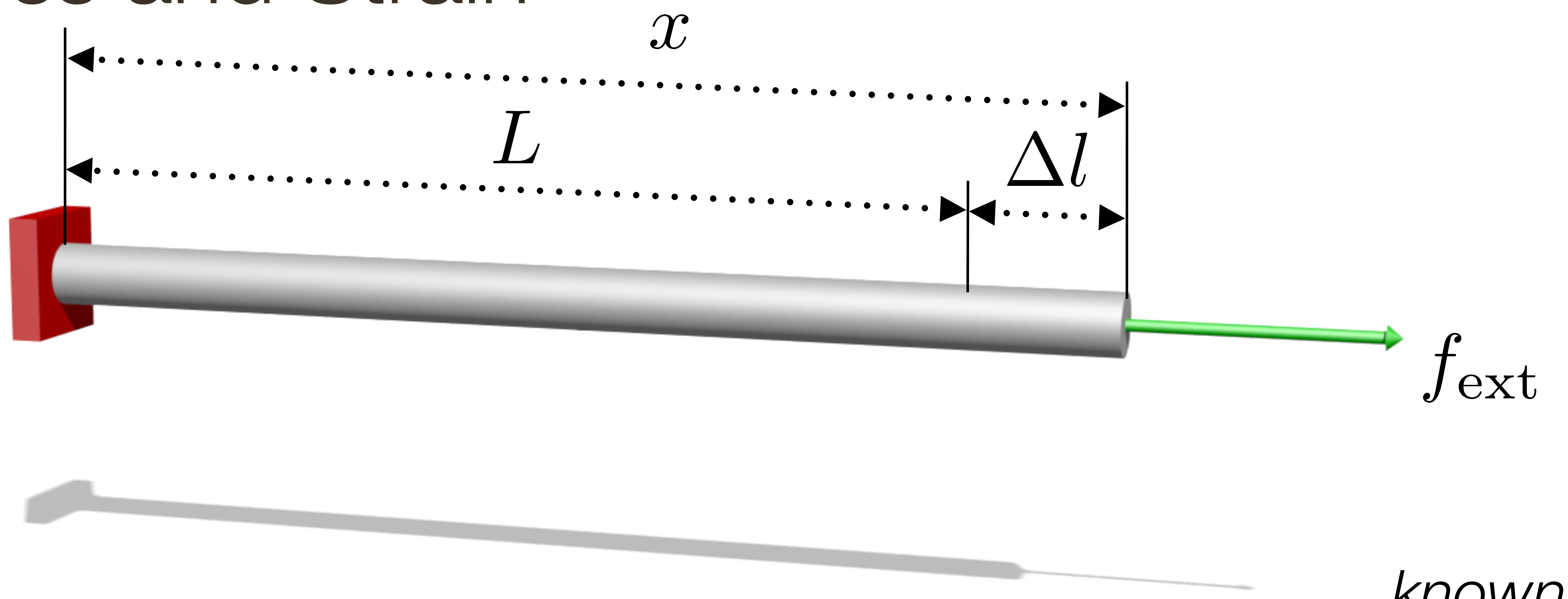
A cross-section



Elastic Rod: Stress and Strain

L rest length

A cross-section



unknown				*known*
deformation	**strain**	Hooke's law	**stress**	force
x	$\varepsilon = \dfrac{x - L}{L}$	$\sigma = E\varepsilon$	$\sigma = \dfrac{f_{\text{ext}}}{A}$	f_{ext}
	relative stretch	E Young's modulus	*force density*	

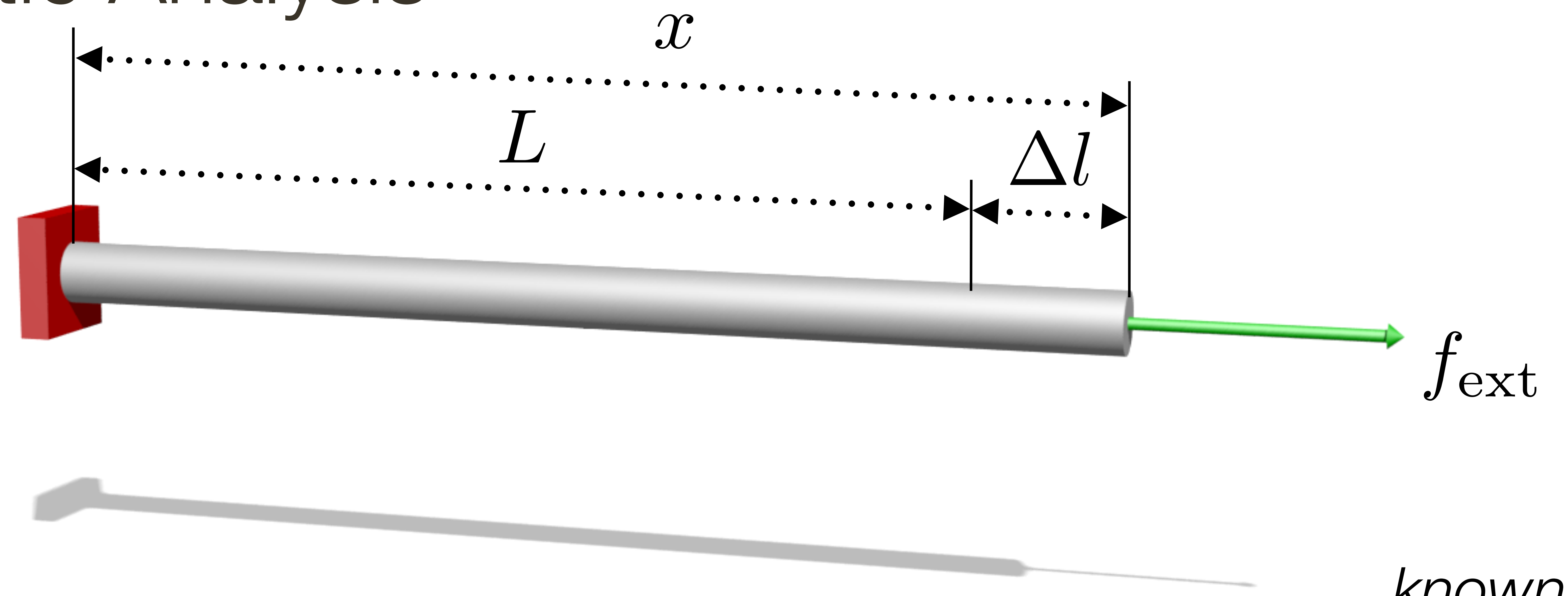
ETH zürich

Disney Research

Elastic Rod: Static Analysis

L rest length

A cross-section



unknown

deformation

x

static solution

$$x = \frac{f_{\text{ext}} L}{EA} + L$$

known

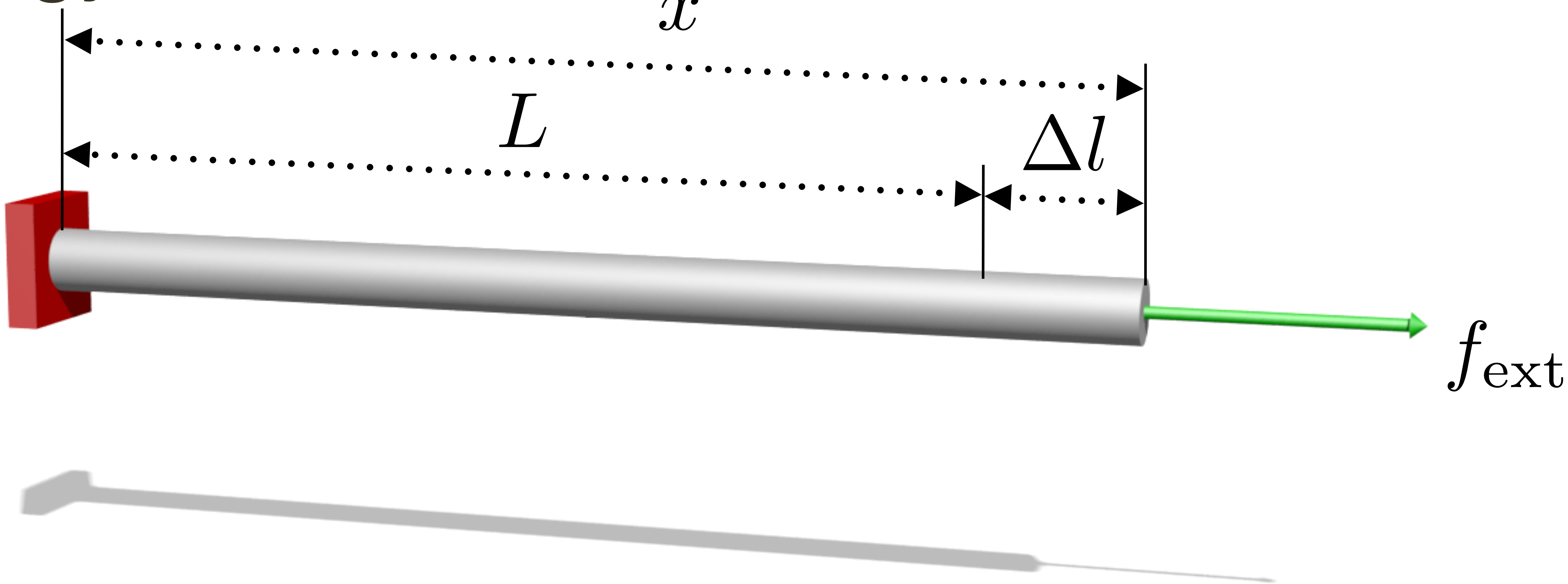
force

f_{ext}

Elastic Rod: Energy

L rest length

A cross-section



$$f_{\text{static}}(x) = U(x) - W(x)$$
$$= \Psi(x)V - f_{\text{ext}}(x - L)$$

strain energy density

$$\Psi(x) = \frac{1}{2}E\varepsilon^2(x)$$

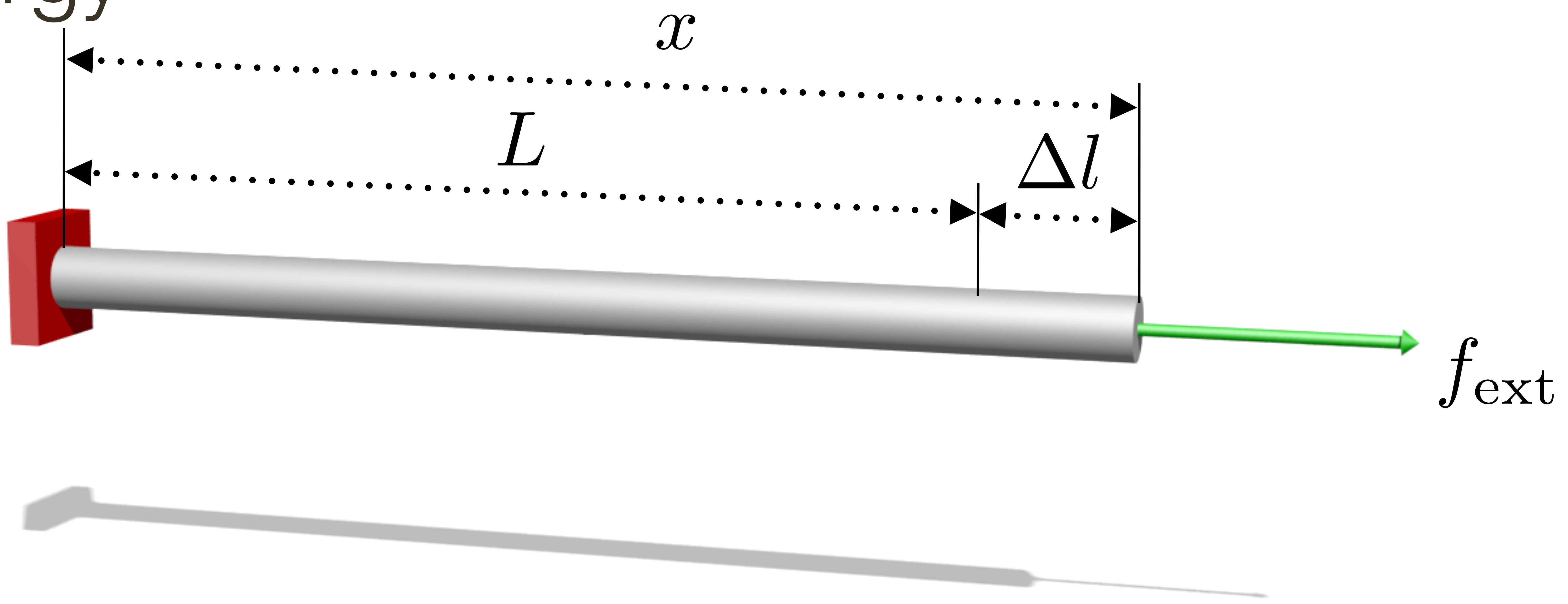
volume

$$V = AL$$

Elastic Rod: Energy

L rest length

A cross-section



$$\frac{\mathrm{d}f_{\text{static}}(x)}{\mathrm{d}x} = E\varepsilon(x)A - f_{\text{ext}} \stackrel{!}{=} 0$$

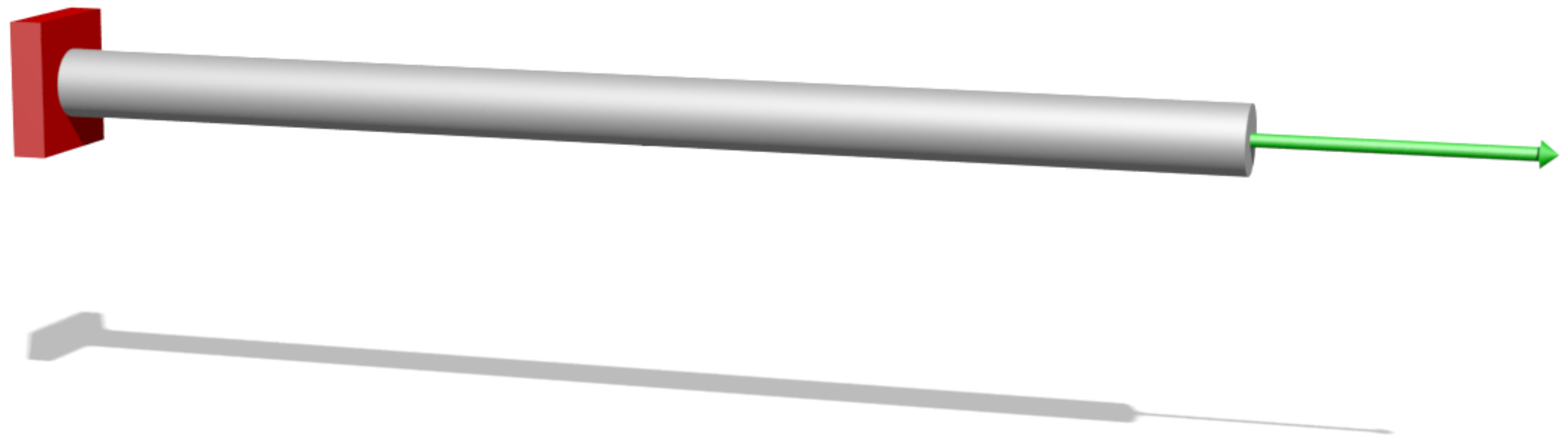
strain

$$\varepsilon(x) = \frac{x - L}{L}$$

static solution

$$\boxed{x = \frac{f_{\text{ext}}L}{EA} + L}$$

Elastic Rod: Energy

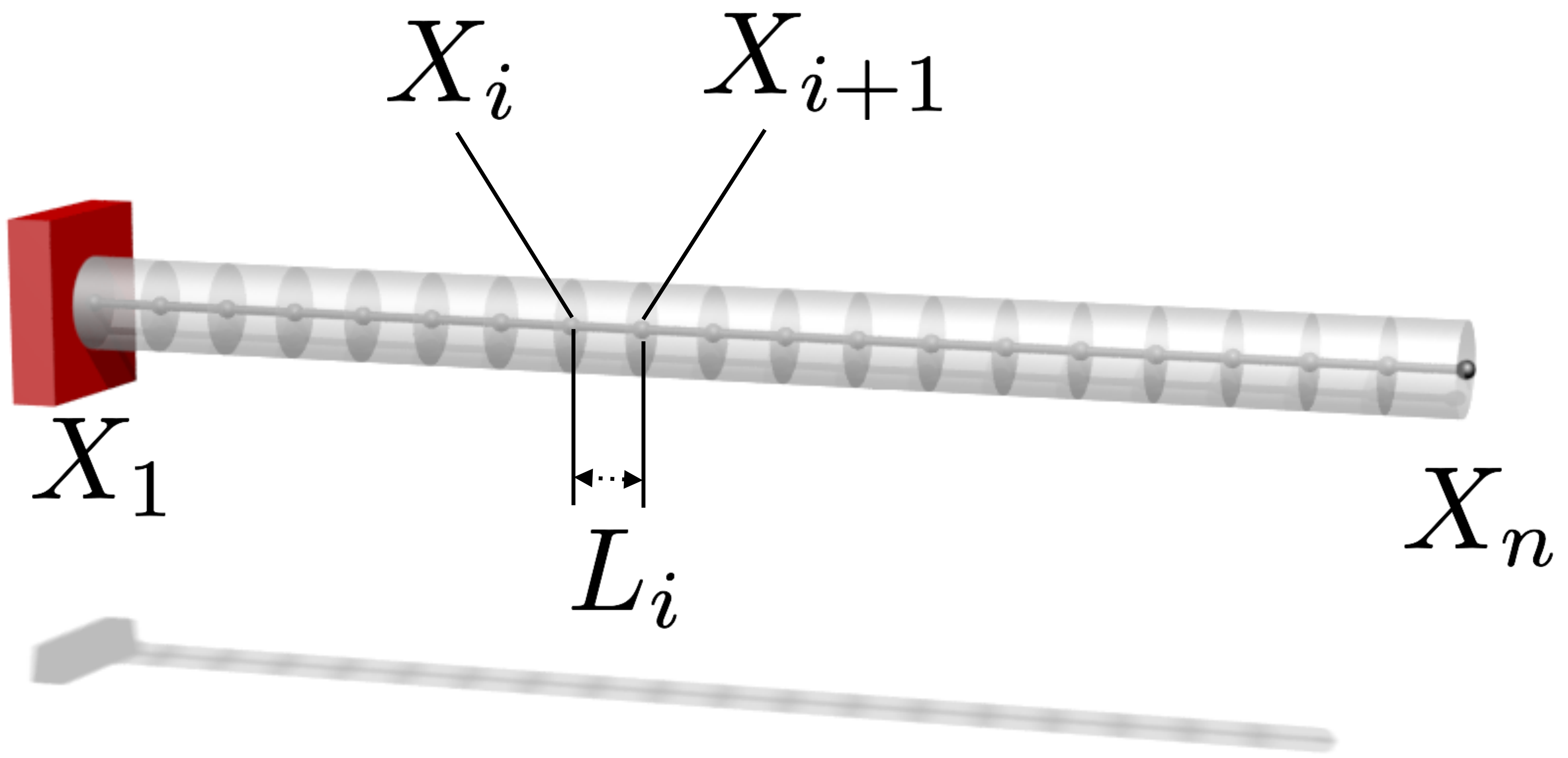


Principle of minimum potential energy

A mechanical system in static equilibrium will assume a state of minimum potential energy.

Elastic Rod: Finite Element Discretization

$$\mathbf{X} = \begin{bmatrix} X_1 \\ \vdots \\ X_n \end{bmatrix}$$



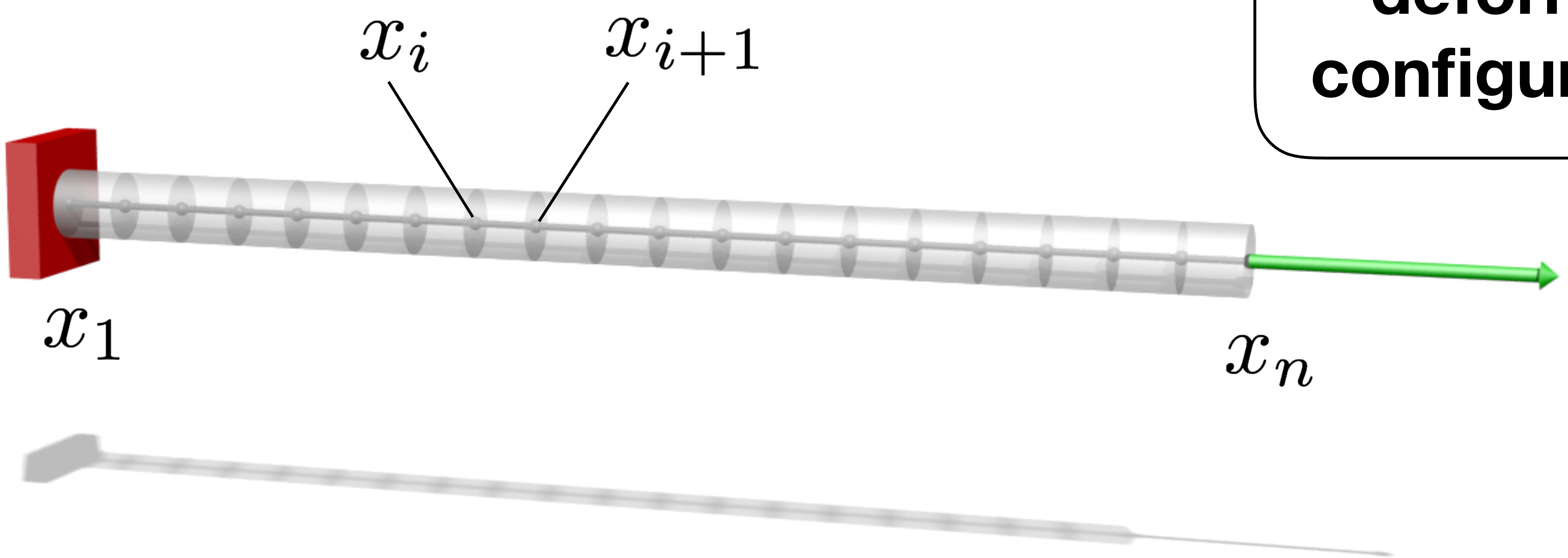
undeformed configuration

element rest length

$$L_i = X_{i+1} - X_i$$

Elastic Rod: Finite Element Discretization

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$$



deformed configuration

element rest length

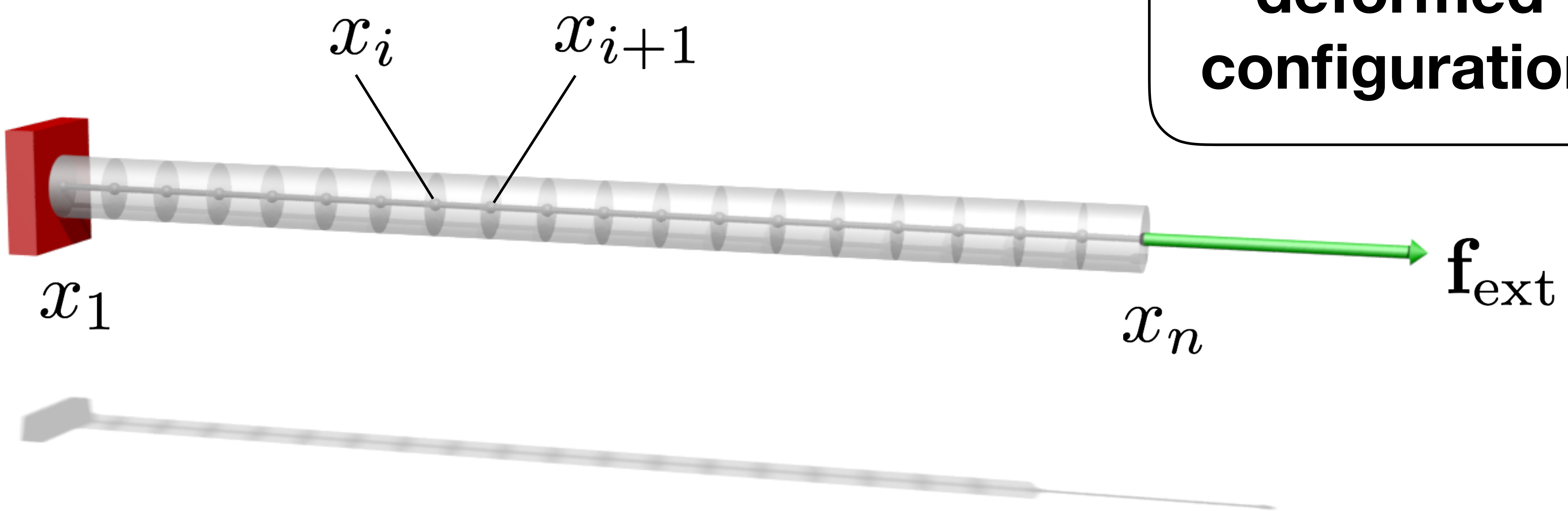
$$L_i = X_{i+1} - X_i$$

element strain

$$\varepsilon_i = \frac{x_{i+1} - x_i - L_i}{L_i}$$

Elastic Rod: Finite Element Discretization

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$$



deformed configuration

element rest length

$$L_i = X_{i+1} - X_i$$

element strain

$$\varepsilon_i = \frac{x_{i+1} - x_i - L_i}{L_i}$$

energy

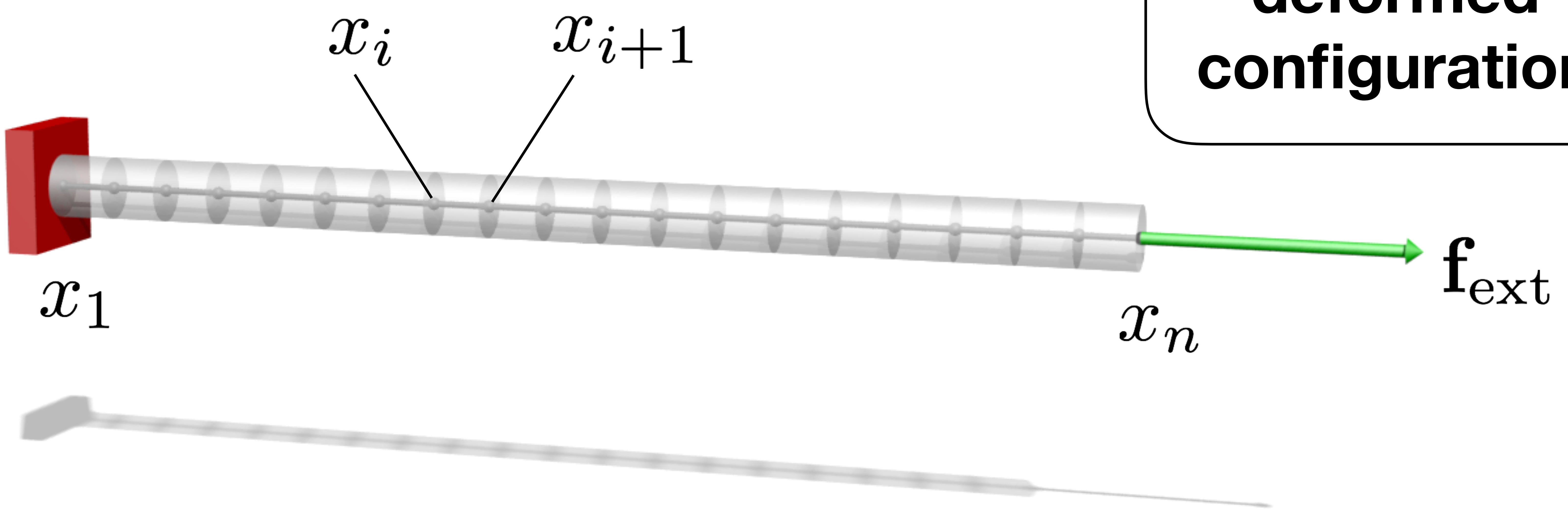
$$f_{\text{static}}(\mathbf{x}) = \sum_{i=1}^{n-1} U_i(\mathbf{x}) - \mathbf{f}_{\text{ext}}^T(\mathbf{x} - \mathbf{X})$$

$$U_i(\mathbf{x}) = \Psi_i(\mathbf{x})V_i = \frac{1}{2}E\varepsilon_i^2(\mathbf{x})AL_i$$

Elastic Rod: Finite Element Discretization

deformed configuration

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$$



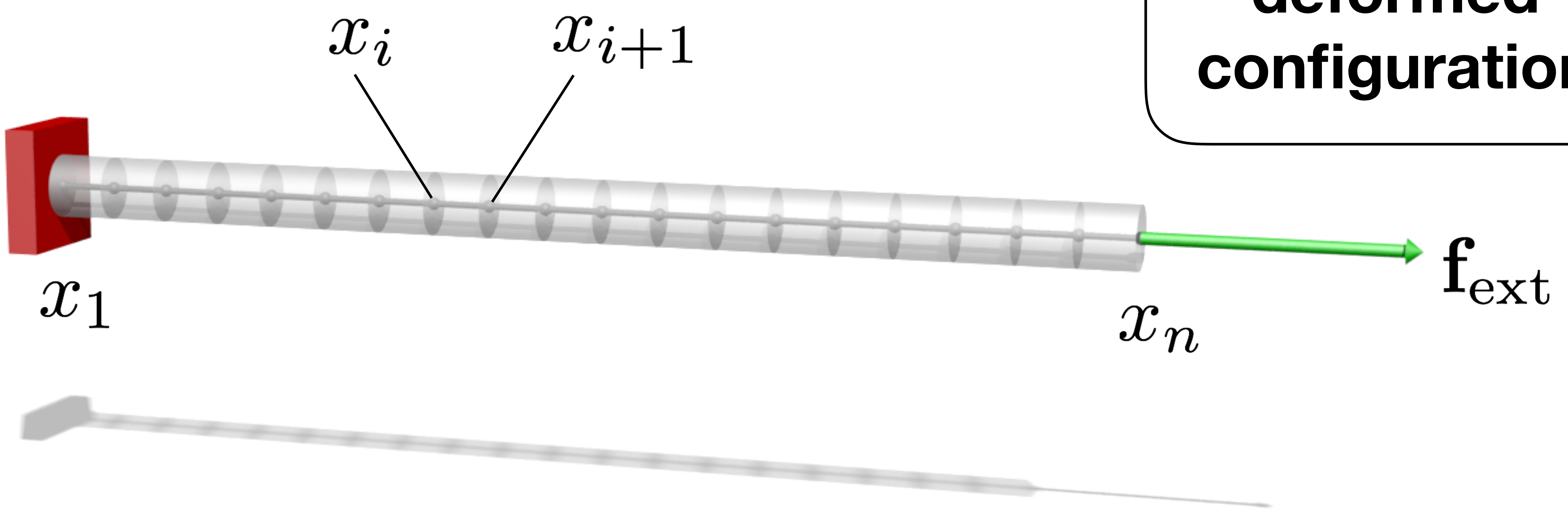
energy gradient

$$\nabla f_{\text{static}}(\mathbf{x}) = \sum_{i=1}^{n-1} \nabla U_i(\mathbf{x}) - \mathbf{f}_{\text{ext}}$$

$$\nabla U_i(\mathbf{x}) = \begin{bmatrix} \frac{\partial U_i}{\partial x_i} \\ \frac{\partial U_i}{\partial x_{i+1}} \end{bmatrix} = \begin{bmatrix} -E\varepsilon_i A \\ E\varepsilon_i A \end{bmatrix}$$

Elastic Rod: Finite Element Discretization

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$$



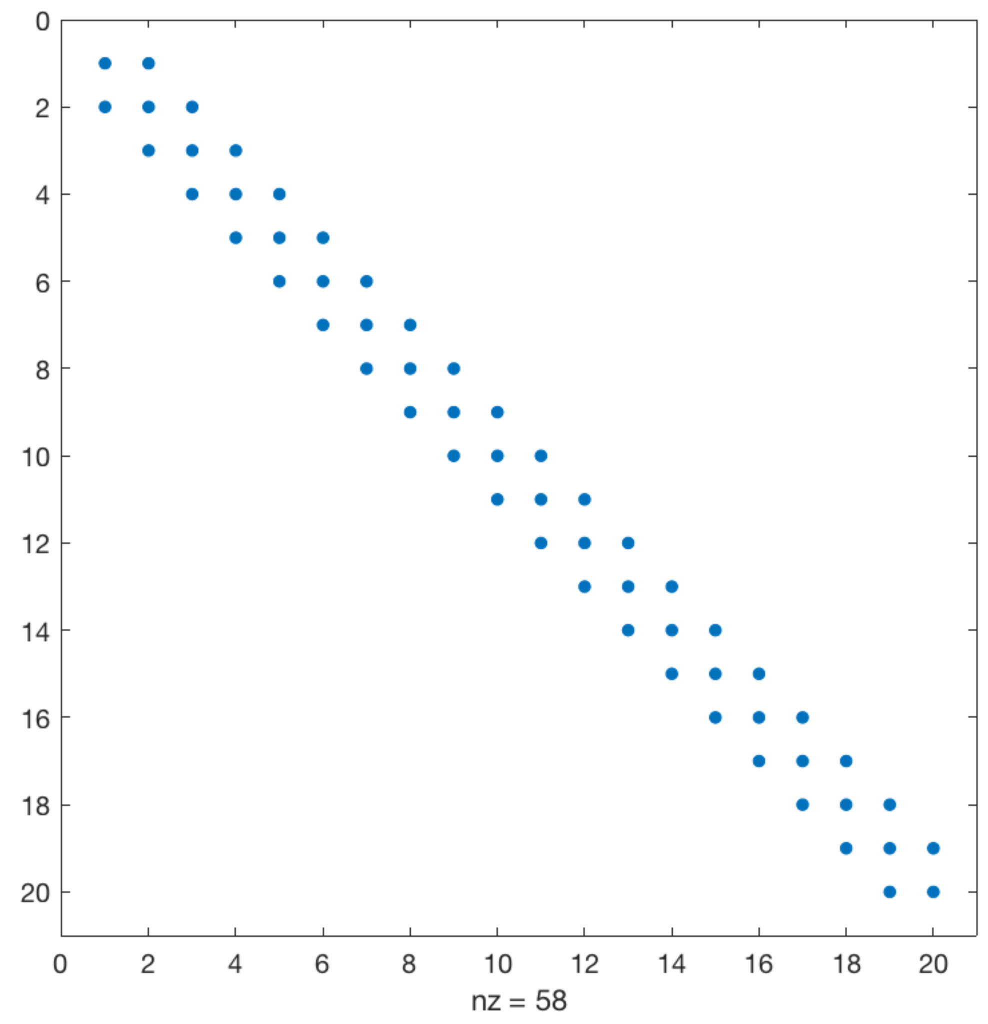
deformed configuration

constant stiffness matrix

energy Hessian

$$\mathbf{K} = \nabla^2 f_{\text{static}}(\mathbf{x}) = \sum_{i=1}^{n-1} \nabla^2 U_i$$

$$\nabla^2 U_i = \begin{bmatrix} \frac{\partial^2 U_i}{\partial x_i^2} & \frac{\partial^2 U_i}{\partial x_i \partial x_{i+1}} \\ \frac{\partial^2 U_i}{\partial x_{i+1} \partial x_i} & \frac{\partial^2 U_i}{\partial x_{i+1}^2} \end{bmatrix} = \begin{bmatrix} \frac{EA}{L_i} & -\frac{EA}{L_i} \\ -\frac{EA}{L_i} & \frac{EA}{L_i} \end{bmatrix}$$



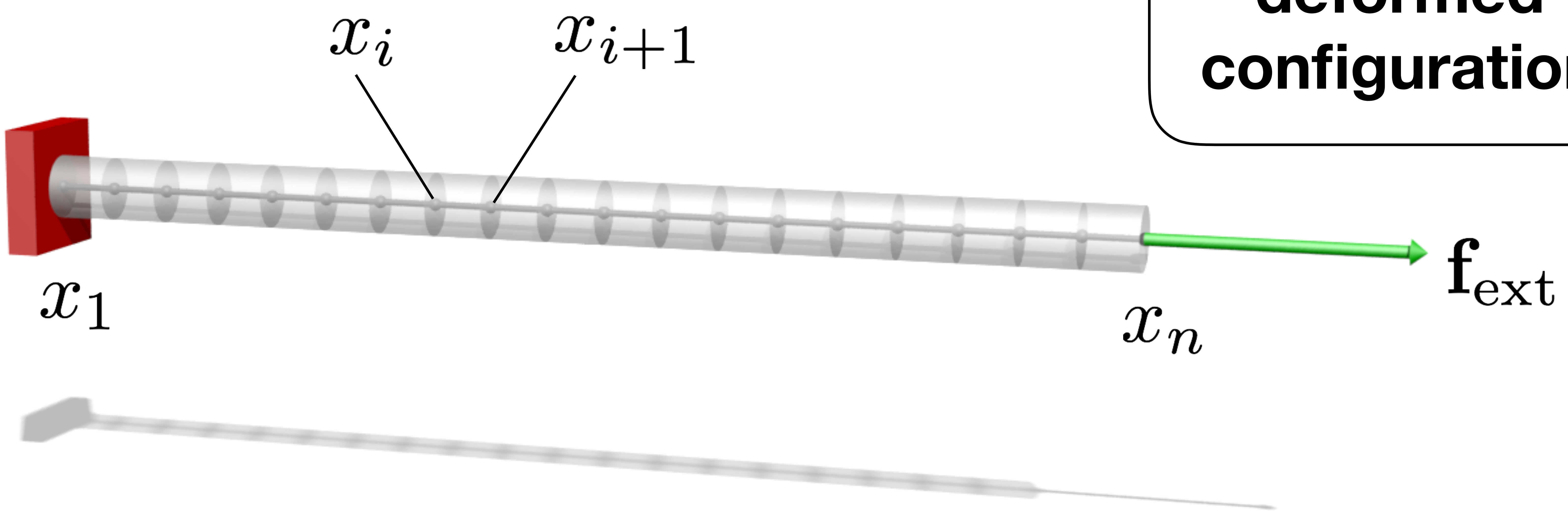
sparsity pattern

ETH zürich

Disney Research

Elastic Rod: Linear Elasticity

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$$



deformed configuration

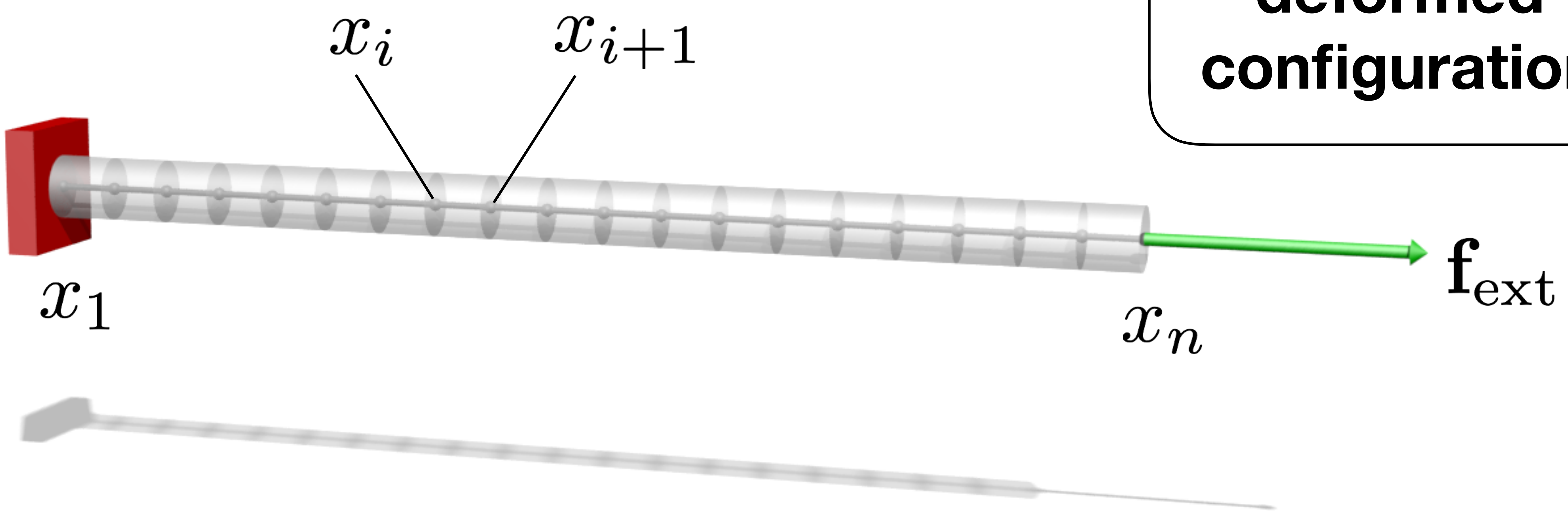
constant stiffness matrix $\nabla^2 f_{\text{static}}(\mathbf{x}) = \mathbf{K}$

linear forces $\nabla f_{\text{static}}(\mathbf{x}) = \mathbf{K}(\mathbf{x} - \mathbf{X}) - \mathbf{f}_{\text{ext}}$

quadratic energy $f_{\text{static}}(\mathbf{x}) = \frac{1}{2}(\mathbf{x} - \mathbf{X})^T \mathbf{K}(\mathbf{x} - \mathbf{X}) - \mathbf{f}_{\text{ext}}^T(\mathbf{x} - \mathbf{X})$

Elastic Rod: Linear Elasticity

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$$



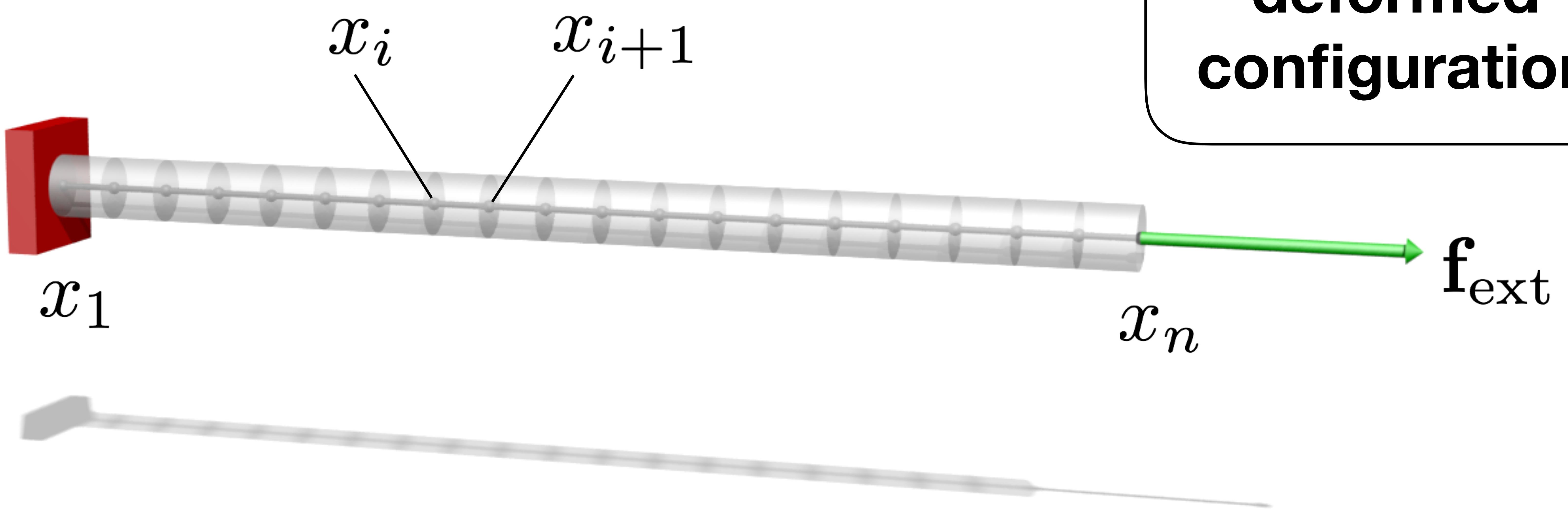
deformed configuration

$$\nabla f_{\text{static}}(\mathbf{x}^*) = \mathbf{K}\,(\mathbf{x}^* - \mathbf{X}) - \mathbf{f}_{\text{ext}} \stackrel{!}{=} \mathbf{o}$$

1. solve sparse linear system $\mathbf{K}\mathbf{u}^* = \mathbf{f}_{\text{ext}}$ $\quad$ $\mathbf{u}^*$ displacement
2. compute deformation $\mathbf{x}^* = \mathbf{X} + \mathbf{u}^*$

Elastic Rod: Boundary Conditions

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$$



deformed configuration

- *eigenvalue decomposition* of $\mathbf{K}$: one *eigenvalue* is *zero*
- stiffness matrix *not* positive definite: *unstable* equilibrium
- missing *Dirichlet* condition: fix one node (e.g., $x_1 = 0$)

Enforcing Dirichlet Conditions

$$\mathbf{K}\mathbf{u} = \mathbf{f} \qquad\qquad u_2 = v$$

$$\left[\begin{array}{c|c|ccc} & a & & & \\ \hline a & b & c & d & e \\ \hline & c & & & \\ & d & & & \\ & e & & & \end{array}\right] \left[\begin{array}{c} u_1 \\ \hline u_2 \\ \hline u_3 \\ u_4 \\ u_5 \end{array}\right] = \left[\begin{array}{c} f_1 \\ \hline f_2 \\ \hline f_3 \\ f_4 \\ f_5 \end{array}\right]$$

[source: Peter Kaufmann]

Enforcing Dirichlet Conditions

$$\mathbf{Ku} = \mathbf{f} \qquad\qquad u_2 = v$$

$$\left[\begin{array}{c|c|ccc} & 0 & & & \\ \hline a & 0 & c & d & e \\ \hline & 0 & & & \\ & 0 & & & \\ & 0 & & & \end{array}\right] \left[\begin{array}{c} u_1 \\ \hline u_2 \\ \hline u_3 \\ u_4 \\ u_5 \end{array}\right] = \left[\begin{array}{c} f_1 \\ \hline f_2 \\ \hline f_3 \\ f_4 \\ f_5 \end{array}\right] - v \left[\begin{array}{c} a \\ \hline b \\ \hline c \\ d \\ e \end{array}\right]$$

[source: Peter Kaufmann]

Enforcing Dirichlet Conditions

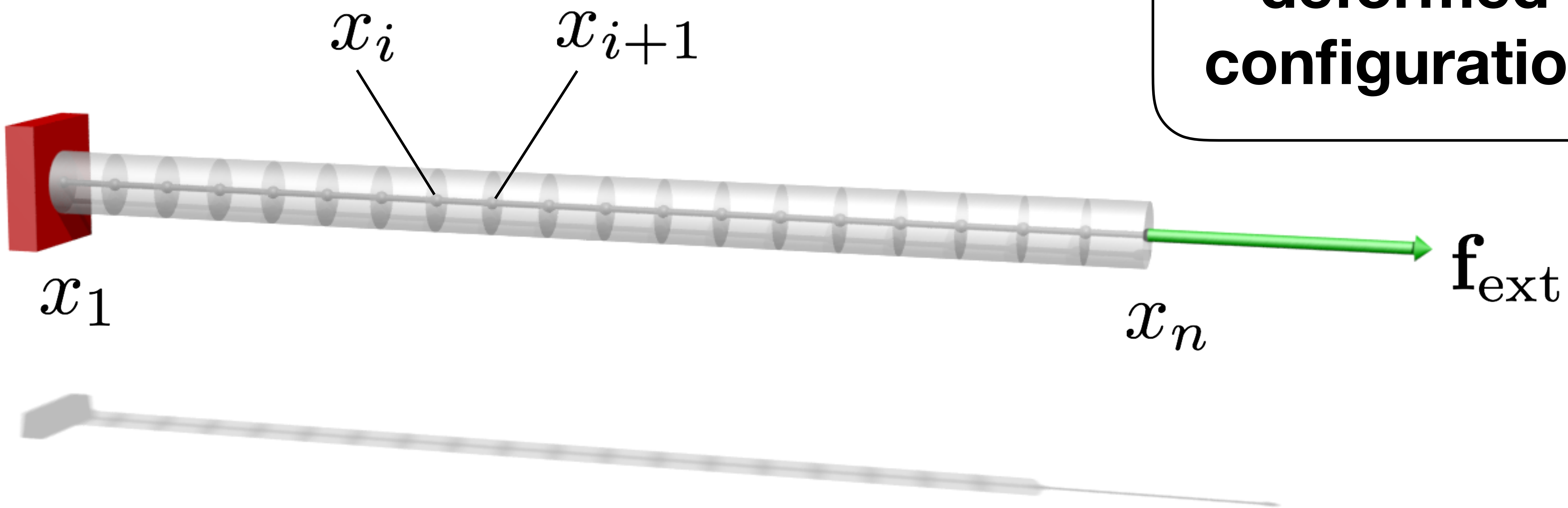
$$\mathbf{Ku} = \mathbf{f} \qquad\qquad u_2 = v$$

$$\left[\begin{array}{c|c|ccc} & 0 & & & \\ \hline 0 & 1 & 0 & 0 & 0 \\ \hline & 0 & & & \\ & 0 & & & \\ & 0 & & & \end{array}\right] \left[\begin{array}{c} u_1 \\ \hline u_2 \\ \hline u_3 \\ u_4 \\ u_5 \end{array}\right] = \left[\begin{array}{c} f_1 - va \\ \hline v \\ \hline f_3 - vc \\ f_4 - vd \\ f_5 - ve \end{array}\right]$$

[source: Peter Kaufmann]

Elastic Rod: Boundary Conditions

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$$



deformed configuration

$$\mathbf{K}\mathbf{u}^* = \mathbf{f}_{\text{ext}}$$

$$u_1 = 0$$

$$\left[\begin{array}{c|ccc} 1 & 0 & \dots & 0 \\ \hline 0 & & & \\ \vdots & & & \\ 0 & & & \end{array}\right] \left[\begin{array}{c} u_1 \\ \hline u_2 \\ \vdots \\ u_n \end{array}\right] = \left[\begin{array}{c} 0 \\ \hline 0 \\ \vdots \\ f_{\text{ext}} \end{array}\right]$$

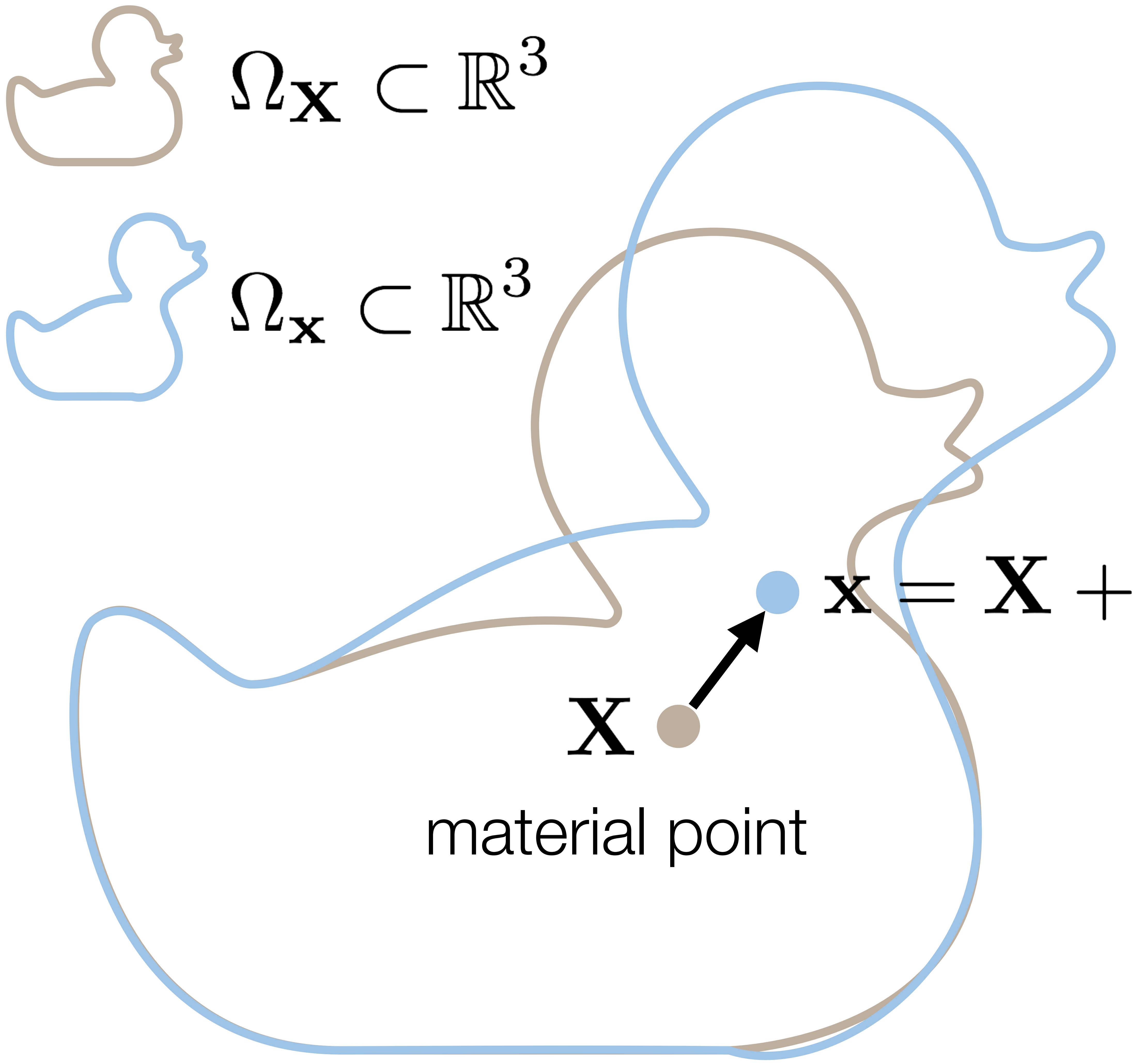
Agenda

- Motivation
- Energy, forces, static vs. dynamic analysis
- Numerical time integration (explicit vs. implicit schemes)
- Assembly: energy, forces, stiffness matrix
- Continuum mechanics: strain, stress, material models
- Linear vs. nonlinear FEM (Finite Element Method)

Continuum Mechanics in 3D: Deformation

$\Omega_{\mathbf{X}} \subset \mathbb{R}^3$

$\Omega_{\mathbf{x}} \subset \mathbb{R}^3$



$\mathbf{X} \in \Omega_{\mathbf{X}}$ *undeformed* or *rest* state

$\mathbf{x} \in \Omega_{\mathbf{x}}$ *deformed* state

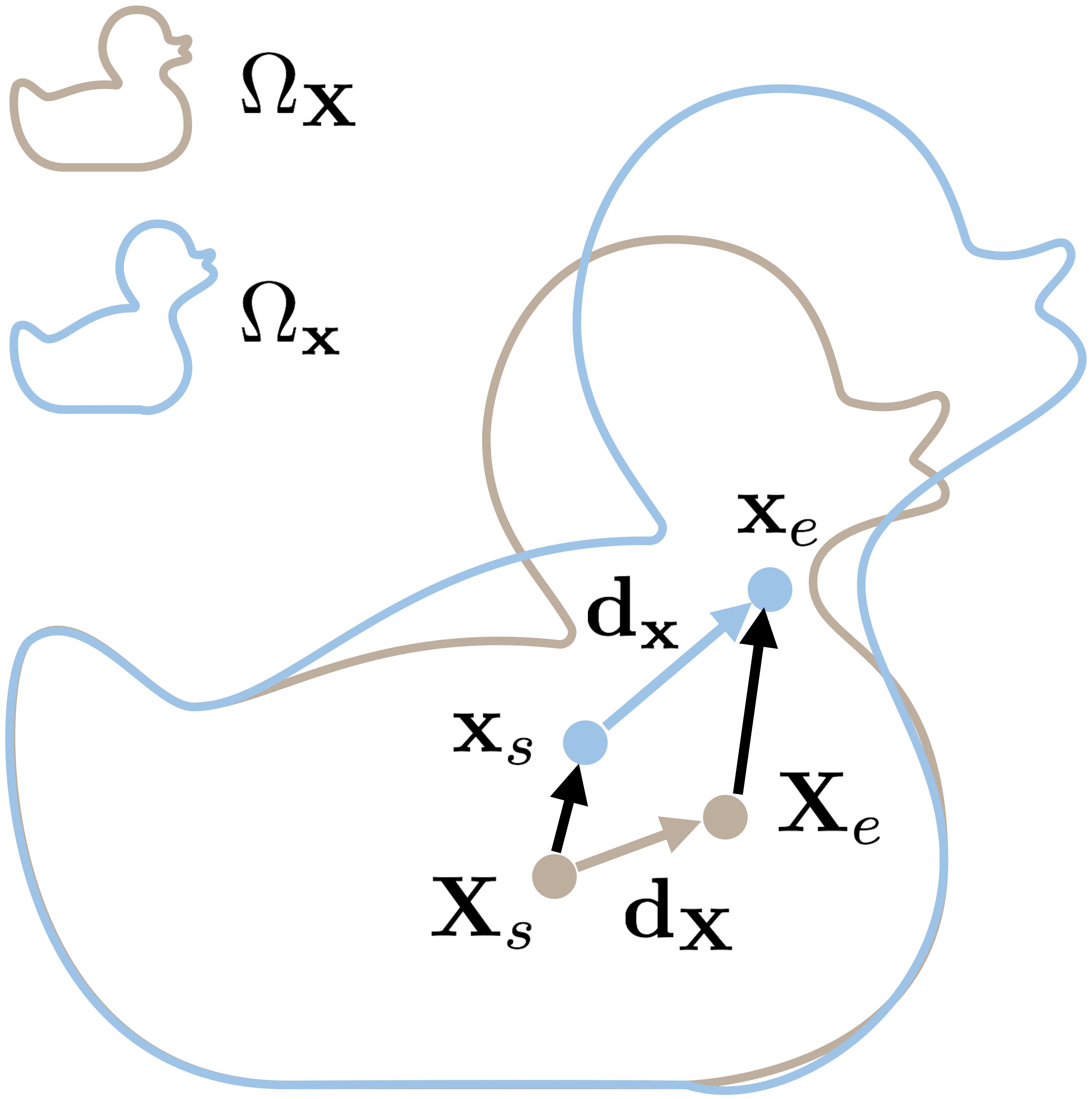
Displacement Field

$$\mathbf{u} : \Omega_{\mathbf{X}} \rightarrow \Omega_{\mathbf{x}}$$

$$\mathbf{x} = \mathbf{X} + \mathbf{u}(\mathbf{X})$$

$$\mathbf{u}(\mathbf{X}) = \begin{bmatrix} u(X, Y, Z) \\ v(X, Y, Z) \\ w(X, Y, Z) \end{bmatrix}$$

Continuum Mechanics in 3D: Deformation



- *infinitesimal* vector
  - *undeformed* $\mathbf{d}_{\mathbf{X}} = \mathbf{X}_e - \mathbf{X}_s$
  - *deformed* $\mathbf{d}_{\mathbf{x}} = \mathbf{x}_e - \mathbf{x}_s$

$$
\begin{aligned}
\mathbf{d}_{\mathbf{x}} &= \mathbf{x}_e - \mathbf{x}_s \\
&= \mathbf{X}_e + \mathbf{u}(\mathbf{X}_e) - \mathbf{X}_s - \mathbf{u}(\mathbf{X}_s) \\
&= \mathbf{d}_{\mathbf{X}} + \mathbf{u}(\mathbf{X}_s + \mathbf{d}_{\mathbf{X}}) - \mathbf{u}(\mathbf{X}_s) \\
&\approx \mathbf{d}_{\mathbf{X}} + \mathbf{u}(\mathbf{X}_s) + \nabla_{\mathbf{X}}\mathbf{u}(\mathbf{X}_s)\mathbf{d}_{\mathbf{X}} - \mathbf{u}(\mathbf{X}_s) \\
&= (\mathbf{I} + \nabla_{\mathbf{X}}\mathbf{u}(\mathbf{X}_s))\,\mathbf{d}_{\mathbf{X}}
\end{aligned}
$$

- *deformation gradient*

$$\mathbf{F} = \mathbf{I} + \nabla_{\mathbf{X}}\mathbf{u}$$

Continuum Mechanics in 3D: Deformation Gradient

- Deformation gradient $\mathbf{F} = \mathbf{I} + \nabla_{\mathbf{X}} \mathbf{u}$ maps undeformed vectors to deformed vectors: $\mathbf{d_x} = \mathbf{F}\mathbf{d_X}$

$$\mathbf{u}(\mathbf{X}) = \begin{bmatrix} u(X,Y,Z) \\ v(X,Y,Z) \\ w(X,Y,Z) \end{bmatrix} \qquad \nabla\mathbf{u} = \begin{bmatrix} \partial_X u & \partial_Y u & \partial_Z u \\ \partial_X v & \partial_Y v & \partial_Z v \\ \partial_X w & \partial_Y w & \partial_Z w \end{bmatrix}$$

- Alternative form:

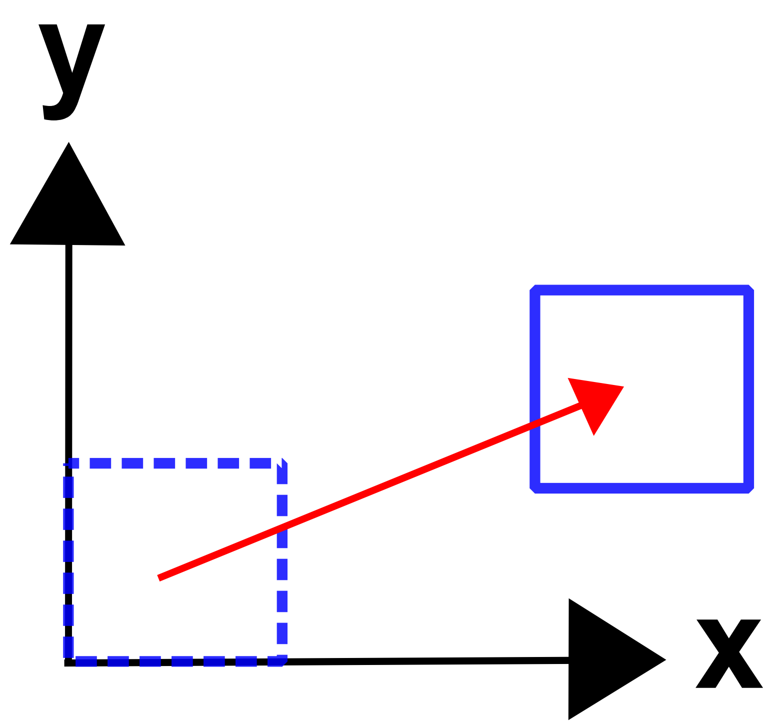
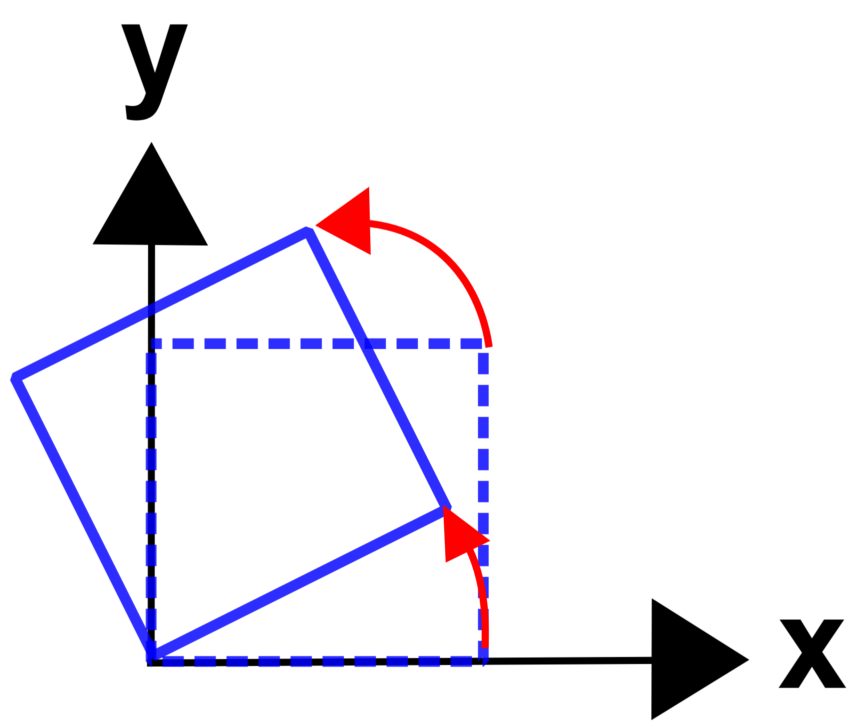
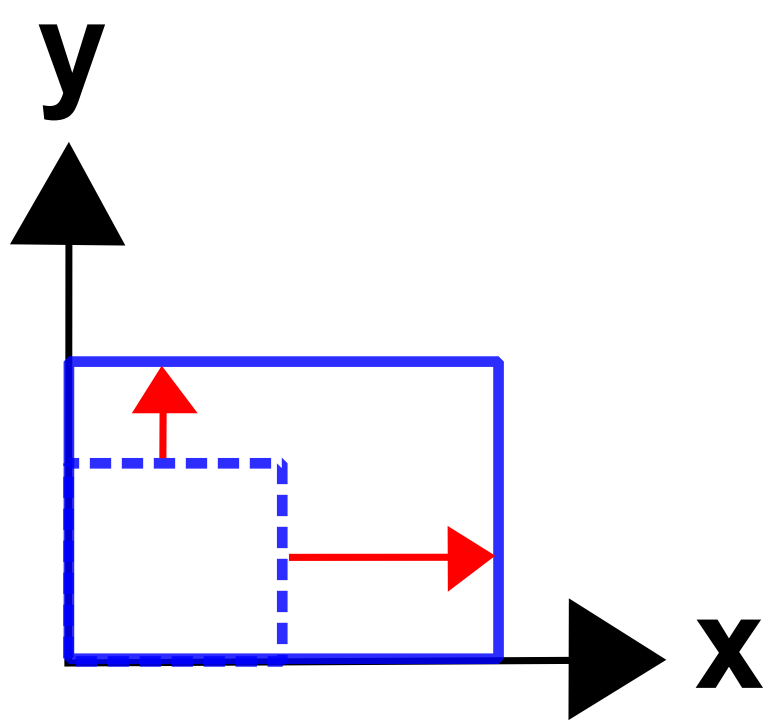
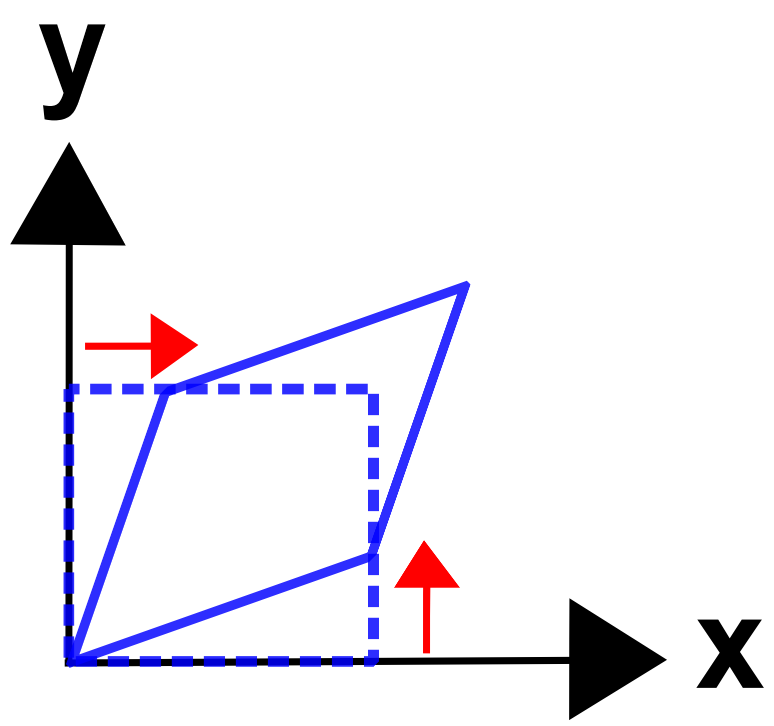
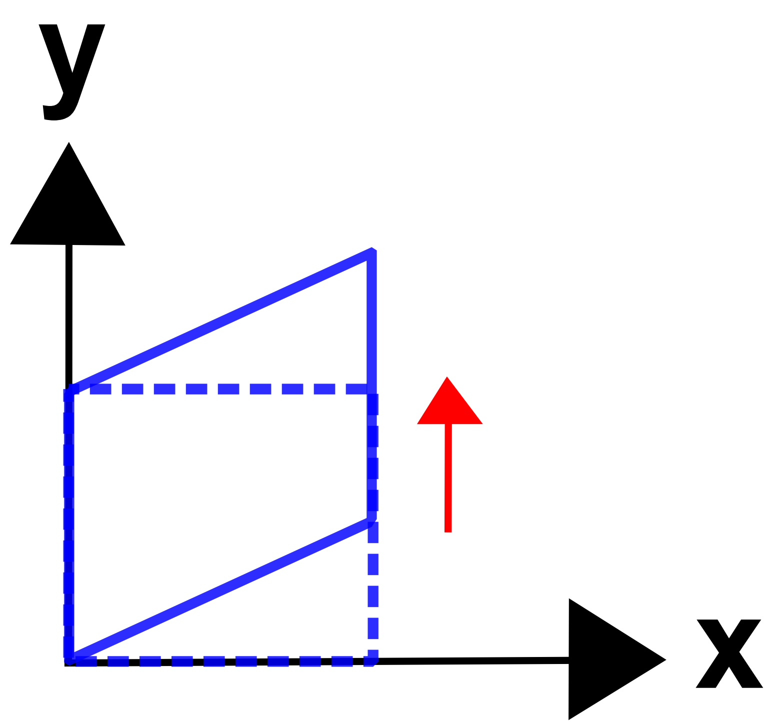
$$\mathbf{x} = \mathbf{X} + \mathbf{u}$$

$$\mathbf{F} = \nabla_{\mathbf{X}} \mathbf{x} = \frac{\partial \mathbf{x}}{\partial \mathbf{X}}$$

$$\begin{aligned} \mathbf{F} &= \nabla_{\mathbf{X}} (\mathbf{X} + \mathbf{u}) \\ &= \nabla_{\mathbf{X}} \mathbf{X} + \nabla_{\mathbf{X}} \mathbf{u} \\ &= \mathbf{I} + \nabla_{\mathbf{X}} \mathbf{u} \end{aligned}$$

$$\nabla_{\mathbf{X}} \mathbf{X} = \frac{\partial \mathbf{X}}{\partial \mathbf{X}} = \mathbf{I}$$

Continuum Mechanics in 3D: Deformation Gradient

translation	rotation	stretch	shear	rotation + shear
				
$x = X + 5$ $y = Y + 2$	$x = X\cos\theta + Y\sin\theta$ $y = X\sin\theta + Y\cos\theta$	$x = 2X + 0Y$ $y = 0X + 1.5Y$	$x = 1X + 0.5Y$ $y = 0.5X + 1Y$	$x = 1X + 0Y$ $y = 0.5X + 1Y$
$\mathbf{F} = \mathbf{I}$	$\mathbf{F} = \begin{bmatrix} \cos\theta & \sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$	$\mathbf{F} = \begin{bmatrix} 2 & 0 \\ 0 & 1.5 \end{bmatrix}$	$\mathbf{F} = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix}$	$\mathbf{F} = \begin{bmatrix} 1 & 0 \\ 0.5 & 1 \end{bmatrix}$
translational invariant	***not*** *rotational* invariant			

[source: http://www.continuummechanics.org/]

Continuum Mechanics in 3D: Nonlinear Strain

- Deformation gradient $\mathbf{F} = \mathbf{I} + \nabla_{\mathbf{X}} \mathbf{u}$ maps undeformed vectors to deformed vectors: $\mathbf{d_x} = \mathbf{F} \mathbf{d_X}$

- Measure change in length *(squared)* in all directions:

$$
\begin{aligned}
\|\mathbf{d_x}\|^2 - \|\mathbf{d_X}\|^2 &= \mathbf{d_x}^T \mathbf{d_x} - \mathbf{d_X}^T \mathbf{d_X} \\
&= \mathbf{d_X}^T \mathbf{F}^T \mathbf{F} \mathbf{d_X} - \mathbf{d_X}^T \mathbf{d_X} \\
&= \mathbf{d_X}^T \left(\mathbf{F}^T \mathbf{F} - \mathbf{I} \right) \mathbf{d_X}
\end{aligned}
$$

- **Green strain**:

$$\boxed{\mathbf{E} = \frac{1}{2} \left(\mathbf{F}^T \mathbf{F} - \mathbf{I} \right)}$$

Continuum Mechanics in 3D: Linear Strain

- Green strain is quadratic in 1st derivatives of displacements

$$\mathbf{E} = \frac{1}{2}\left(\mathbf{F}^T\mathbf{F} - \mathbf{I}\right) = \frac{1}{2}\left(\nabla_{\mathbf{X}}\mathbf{u} + \nabla_{\mathbf{X}}\mathbf{u}^T + \nabla_{\mathbf{X}}\mathbf{u}^T\nabla_{\mathbf{X}}\mathbf{u}\right)$$

- Neglecting quadratic terms leads to linear **Cauchy strain**

$$\boxed{\varepsilon = \frac{1}{2}\left(\nabla_{\mathbf{X}}\mathbf{u} + \nabla_{\mathbf{X}}\mathbf{u}^T\right) = \frac{1}{2}\left(\mathbf{F} + \mathbf{F}^T\right) - \mathbf{I}}$$

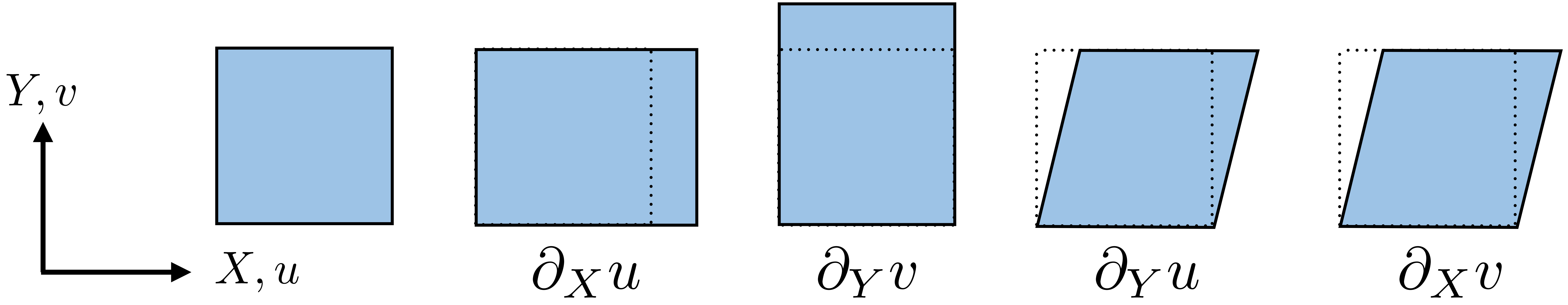
Continuum Mechanics in 3D: Linear Strain

- Linear Cauchy strain

$$\boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_x & \gamma_{xy} & \gamma_{xz} \\ \gamma_{xy} & \varepsilon_y & \gamma_{yz} \\ \gamma_{xz} & \gamma_{yz} & \varepsilon_z \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 2\partial_X u & \partial_Y u + \partial_X v & \partial_Z u + \partial_X w \\ \partial_X v + \partial_Y u & 2\partial_Y v & \partial_Z v + \partial_Y w \\ \partial_X w + \partial_Z u & \partial_Y w + \partial_Z v & 2\partial_Z w \end{bmatrix}$$

ε_i : normal strains $\qquad$ γ_{ij} : shear strains

- Geometric interpretation (2D)



Continuum Mechanics in 3D: Cauchy vs. Green Strain

- Polar decomposition $\mathbf{F} = \mathbf{RU}$ $\quad \mathbf{R}$: rotation $\quad \mathbf{U}$: stretch + shear

- Nonlinear **Green strain** is rotation-invariant

$$
\begin{aligned}
\mathbf{E} &= \frac{1}{2}\left(\mathbf{F}^T\mathbf{F} - \mathbf{I}\right) \\
&= \frac{1}{2}\left(\mathbf{U}^T\underbrace{\mathbf{R}^T\mathbf{R}}_{\mathbf{I}}\mathbf{U} - \mathbf{I}\right) = \frac{1}{2}\left(\underbrace{\mathbf{U}^T\mathbf{U}}_{\text{deformation}} - \mathbf{I}\right)
\end{aligned}
$$

- Linear **Cauchy strain** is *not* rotation-invariant

$$
\boldsymbol{\varepsilon} = \frac{1}{2}\left(\mathbf{F} + \mathbf{F}^T\right) - \mathbf{I} = \frac{1}{2}\left(\mathbf{RU} + \mathbf{U}^T\mathbf{R}^T\right) - \mathbf{I}
$$

rotation does not cancel out

Continuum Mechanics in 3D: Cauchy vs. Green Strain

[M. Müller, J. Dorsey, L. McMillan, R. Jagnow, B. Cutler,
Stable Real-Time Deformations, SCA 2002]

Continuum Mechanics in 3D: Material Model

- Material model links strain to energy (and stress)

- Linear isotropic material (generalized Hooke's law)
  - strain energy density $\Psi = \frac{1}{2}\lambda \mathrm{tr}(\boldsymbol{\varepsilon})^2 + \mu \mathrm{tr}(\boldsymbol{\varepsilon}^2)$
  - material constants: Lamé parameters λ and μ

$$\boxed{\mathrm{tr}(\boldsymbol{\varepsilon}) = \sum_i \varepsilon_{ii}}$$

- Interpretation
  - $\mathrm{tr}(\boldsymbol{\varepsilon}^2) = \|\boldsymbol{\varepsilon}\|_F^2$ penalizes all strain components equally
  - $\mathrm{tr}(\boldsymbol{\varepsilon})^2$ penalizes dilations, i.e., volume changes

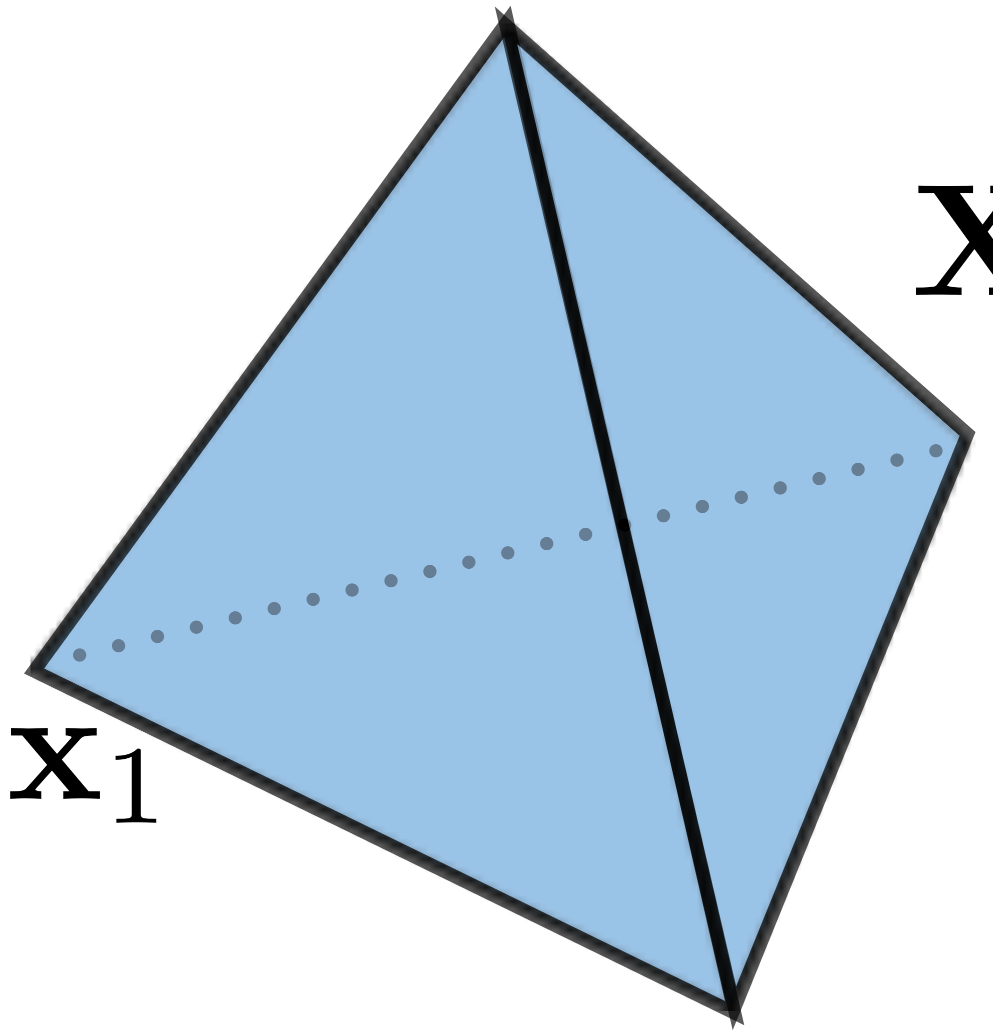
Finite Elements and Deformation Gradient

- Interpolate using shape functions

$$\mathbf{X}(\boldsymbol{\xi}) = \sum_{i=1}^{n_e} N_i(\boldsymbol{\xi})\mathbf{X}_i \qquad \mathbf{x}(\boldsymbol{\xi}) = \sum_{i=1}^{n_e} N_i(\boldsymbol{\xi})\mathbf{x}_i$$

undeformed configuration

deformed configuration



$\boldsymbol{\xi}$ *: elemental coordinates*

- Deformation gradient

$$\mathbf{F} = \frac{\partial \mathbf{x}}{\partial \mathbf{X}} = \frac{\partial \mathbf{x}(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}} \left(\frac{\partial \mathbf{X}(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}} \right)^{-1}$$

$$\frac{\partial \mathbf{X}(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}} = \sum_{i=1}^{n_e} \mathbf{X}_i \left(\frac{\partial N_i(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}} \right)^T$$

$$\frac{\partial \mathbf{x}(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}} = \sum_{i=1}^{n_e} \mathbf{x}_i \left(\frac{\partial N_i(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}} \right)^T$$

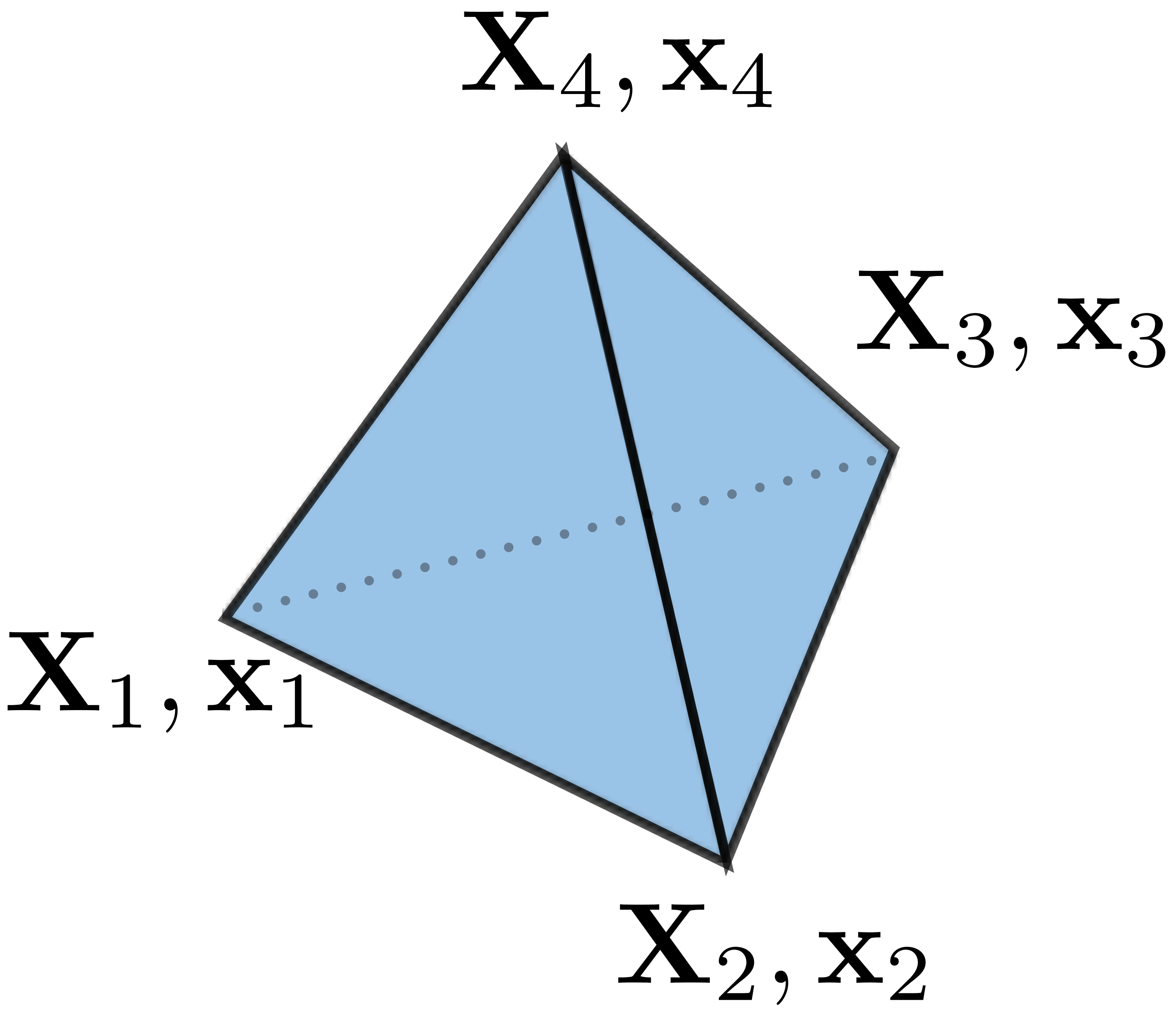
Linear Tetrahedral Elements

- Shape functions $\boldsymbol{\xi}$ *: elemental coordinates*

$$N_1(\boldsymbol{\xi}) = \xi_1 \quad N_2(\boldsymbol{\xi}) = \xi_2 \quad N_3(\boldsymbol{\xi}) = \xi_3$$

$$N_4(\boldsymbol{\xi}) = 1 - \xi_1 - \xi_2 - \xi_3$$

$$\frac{\partial N_1}{\partial \boldsymbol{\xi}} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \quad \frac{\partial N_2}{\partial \boldsymbol{\xi}} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \quad \frac{\partial N_3}{\partial \boldsymbol{\xi}} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad \frac{\partial N_3}{\partial \boldsymbol{\xi}} = \begin{bmatrix} -1 \\ -1 \\ -1 \end{bmatrix}$$



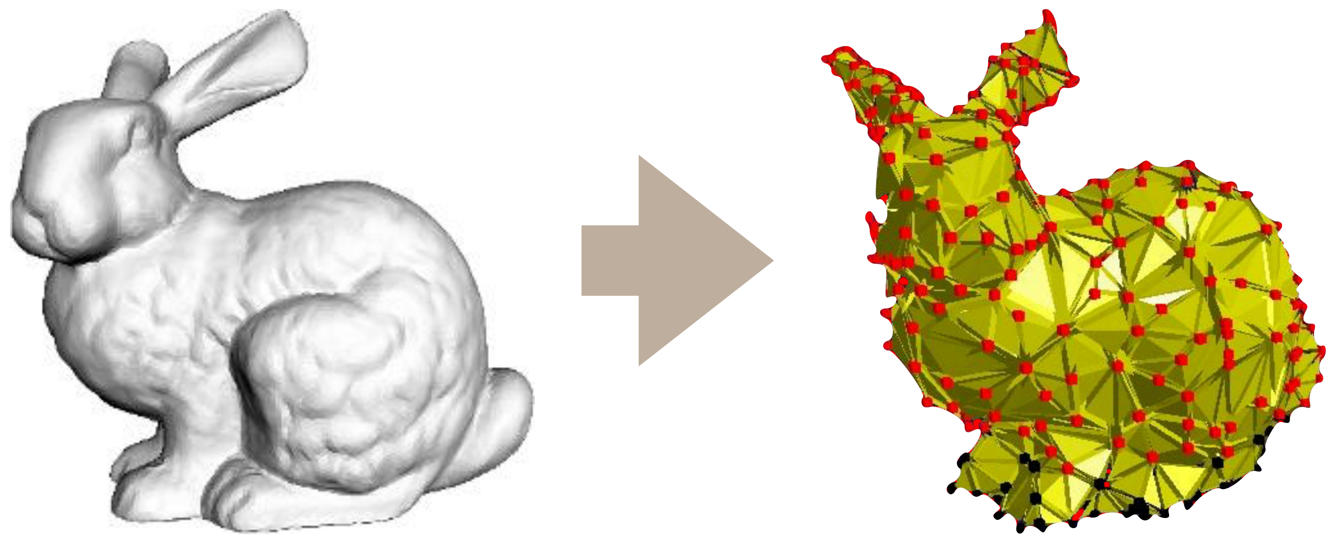
- Deformation gradient

$$\mathbf{F} = \frac{\partial \mathbf{x}}{\partial \boldsymbol{\xi}} \left(\frac{\partial \mathbf{X}}{\partial \boldsymbol{\xi}} \right)^{-1}$$

$$\frac{\partial \mathbf{X}}{\partial \boldsymbol{\xi}} = \left[\; \mathbf{X}_1 - \mathbf{X}_4 \quad \mathbf{X}_2 - \mathbf{X}_4 \quad \mathbf{X}_3 - \mathbf{X}_4 \; \right]$$

$$\frac{\partial \mathbf{x}}{\partial \boldsymbol{\xi}} = \left[\; \mathbf{x}_1 - \mathbf{x}_4 \quad \mathbf{x}_2 - \mathbf{x}_4 \quad \mathbf{x}_3 - \mathbf{x}_4 \; \right]$$

Linear Elasticity



1. Divide input model input *tetrahedra* e

2. Form per-element deformation gradient, Cauchy strain, and strain energy density

deformation gradient
$$\mathbf{F}^e = \frac{\partial \mathbf{x}^e}{\partial \boldsymbol{\xi}} \left(\frac{\partial \mathbf{X}^e}{\partial \boldsymbol{\xi}} \right)^{-1}$$

Cauchy strain
$$\boldsymbol{\varepsilon}^e = \frac{1}{2}\left(\mathbf{F}^e + (\mathbf{F}^e)^T\right) - \mathbf{I}$$

strain energy density
$$\Psi^e = \frac{1}{2}\lambda \mathrm{tr}(\boldsymbol{\varepsilon}^e)^2 + \mu \mathrm{tr}((\boldsymbol{\varepsilon}^e)^2)$$

$$\frac{\partial \mathbf{X}^e}{\partial \boldsymbol{\xi}} = \begin{bmatrix} \mathbf{X}_1^e - \mathbf{X}_4^e & \mathbf{X}_2^e - \mathbf{X}_4^e & \mathbf{X}_3^e - \mathbf{X}_4^e \end{bmatrix} \qquad \frac{\partial \mathbf{x}^e}{\partial \boldsymbol{\xi}} = \begin{bmatrix} \mathbf{x}_1^e - \mathbf{x}_4^e & \mathbf{x}_2^e - \mathbf{x}_4^e & \mathbf{x}_3^e - \mathbf{x}_4^e \end{bmatrix}$$

3. Integrate per-element strain energy density

$$f_{\mathrm{static}}(\mathbf{x}) = \sum_e U^e(\mathbf{x}) - \mathbf{f}_{\mathrm{ext}}^T(\mathbf{x} - \mathbf{X}) \qquad U^e(\mathbf{x}) = \int_{\Omega^e} \Psi^e(\mathbf{x})\mathrm{d}\boldsymbol{\xi} = \Psi^e(\mathbf{x})V_e$$

Linear Elasticity

constant stiffness matrix $\nabla^2 f_{\text{static}}(\mathbf{x}) = \mathbf{K}$

linear forces $\nabla f_{\text{static}}(\mathbf{x}) = \mathbf{K}\left(\mathbf{x} - \mathbf{X}\right) - \mathbf{f}_{\text{ext}}$

very fast

1. factorize stiffness matrix $\mathbf{K}$ (e.g., Cholesky decomposition)
2. compute displacement $\mathbf{u}^*$ for external forces $\mathbf{f}_{\text{ext}}$
3. compute deformation $\mathbf{x}^* = \mathbf{X} + \mathbf{u}^*$

- Problem: visible artifacts for large rotations (Cauchy strain)
- Solution: nonlinear elasticity

Nonlinear Elasticity

- Replace Cauchy strain with Green strain:

$$\Psi_{\mathrm{StVK}} = \frac{1}{2}\lambda \mathrm{tr}(\mathbf{E})^2 + \mu \mathrm{tr}(\mathbf{E}^2)$$

St. Venant-Kirchhoff material model

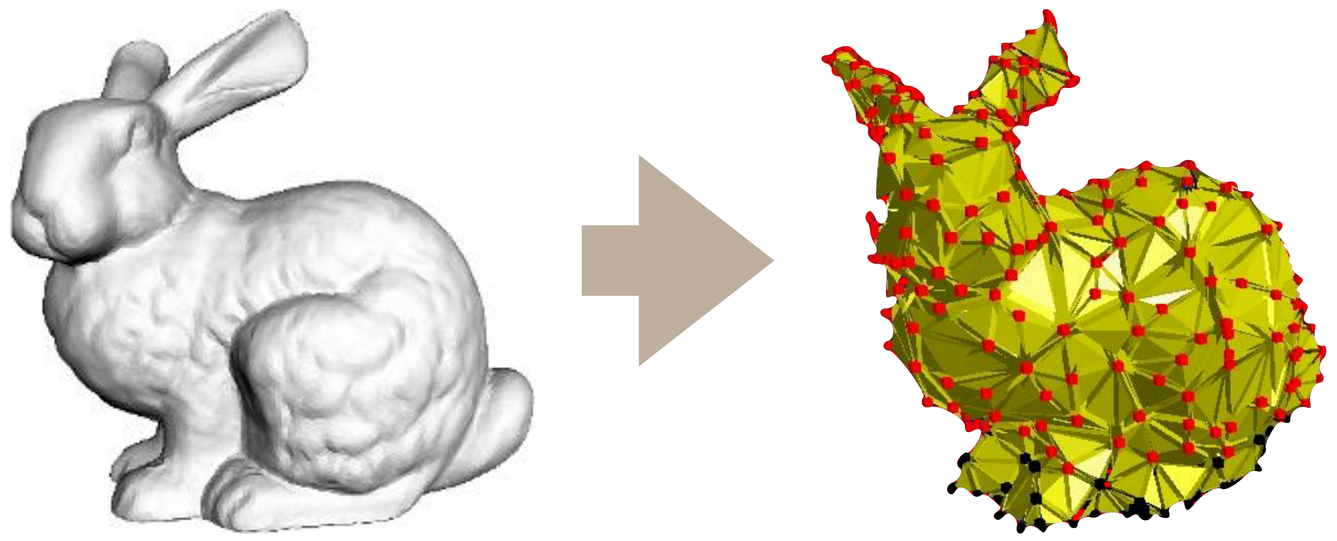
- Stiffness matrix: no longer constant

- Use Newton's method for minimization

$$\min_{\mathbf{x}} f_{\mathrm{static}}(\mathbf{x}) \qquad f_{\mathrm{static}}(\mathbf{x}) = \sum_e U^e(\mathbf{x}) - \mathbf{f}_{\mathrm{ext}}^T(\mathbf{x} - \mathbf{X})$$

Nonlinear Elasticity

1. Divide input model input *tetrahedra* e



2. Form per-element deformation gradient, Green strain, and strain energy density

deformation gradient

$$\mathbf{F}^e = \frac{\partial \mathbf{x}^e}{\partial \boldsymbol{\xi}} \left(\frac{\partial \mathbf{X}^e}{\partial \boldsymbol{\xi}} \right)^{-1}$$

Green strain

$$\mathbf{E}^e = \frac{1}{2} \left((\mathbf{F}^e)^T \mathbf{F}^e - \mathbf{I} \right)$$

strain energy density

$$\Psi^e_{\mathrm{StVK}} = \frac{1}{2} \lambda \mathrm{tr}(\mathbf{E}^e)^2 + \mu \mathrm{tr}((\mathbf{E}^e)^2)$$

$$\frac{\partial \mathbf{X}^e}{\partial \boldsymbol{\xi}} = \begin{bmatrix} \mathbf{X}^e_1 - \mathbf{X}^e_4 & \mathbf{X}^e_2 - \mathbf{X}^e_4 & \mathbf{X}^e_3 - \mathbf{X}^e_4 \end{bmatrix} \quad \frac{\partial \mathbf{x}^e}{\partial \boldsymbol{\xi}} = \begin{bmatrix} \mathbf{x}^e_1 - \mathbf{x}^e_4 & \mathbf{x}^e_2 - \mathbf{x}^e_4 & \mathbf{x}^e_3 - \mathbf{x}^e_4 \end{bmatrix}$$

3. Integrate per-element strain energy density

$$f_{\mathrm{static}}(\mathbf{x}) = \sum_e U^e(\mathbf{x}) - \mathbf{f}^T_{\mathrm{ext}}(\mathbf{x} - \mathbf{X}) \qquad U^e(\mathbf{x}) = \int_{\Omega^e} \Psi^e_{\mathrm{StVK}}(\mathbf{x}) \mathrm{d}\boldsymbol{\xi} = \Psi^e_{\mathrm{StVK}}(\mathbf{x}) V_e$$

Nonlinear Elasticity: Implementation

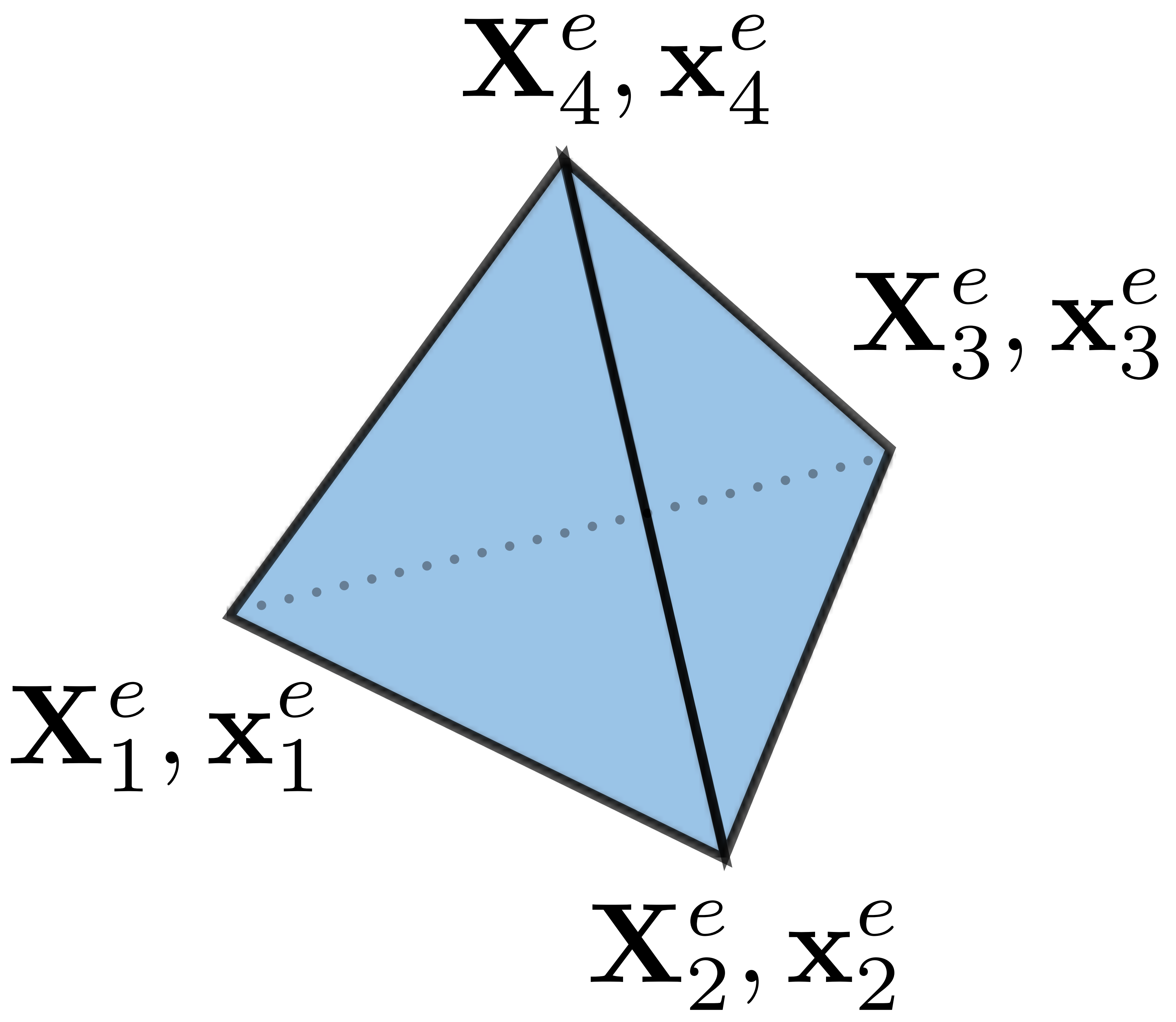
Use symbolic or automatic differentiation to generate code for a single element e :

1. Internal energy $U^e(\mathbf{x}^e) = V^e\Psi(\mathbf{x}^e)$

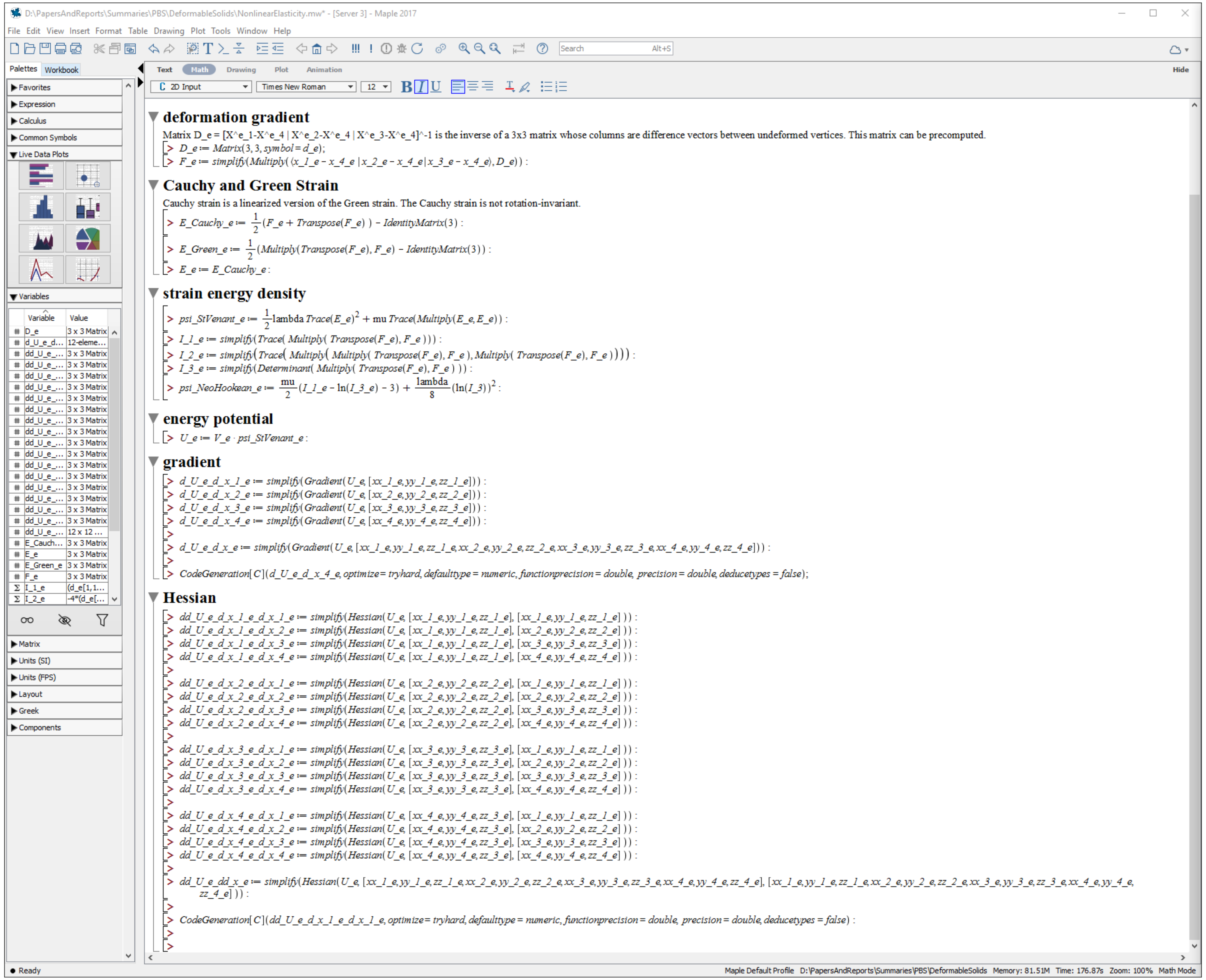
2. Energy gradient $\nabla_{\mathbf{x}^e} U(\mathbf{x}^e)$

3. Energy Hessian $\nabla^2_{\mathbf{x}^e} U(\mathbf{x}^e)$

$$\mathbf{x}^e = \begin{bmatrix} \mathbf{x}_1^e \\ \mathbf{x}_2^e \\ \mathbf{x}_3^e \\ \mathbf{x}_4^e \end{bmatrix}$$



Nonlinear Elasticity: Implementation



Nonlinear Elasticity: Implementation

Apply Newton's method to objective $f_{\text{static}}(\mathbf{x})$:

1. Write function to evaluate f_{static} at $\mathbf{x}$
   - For each element e, compute $U^e(\mathbf{x}^e)$
   - Sum up per-element contributions and subtract external work

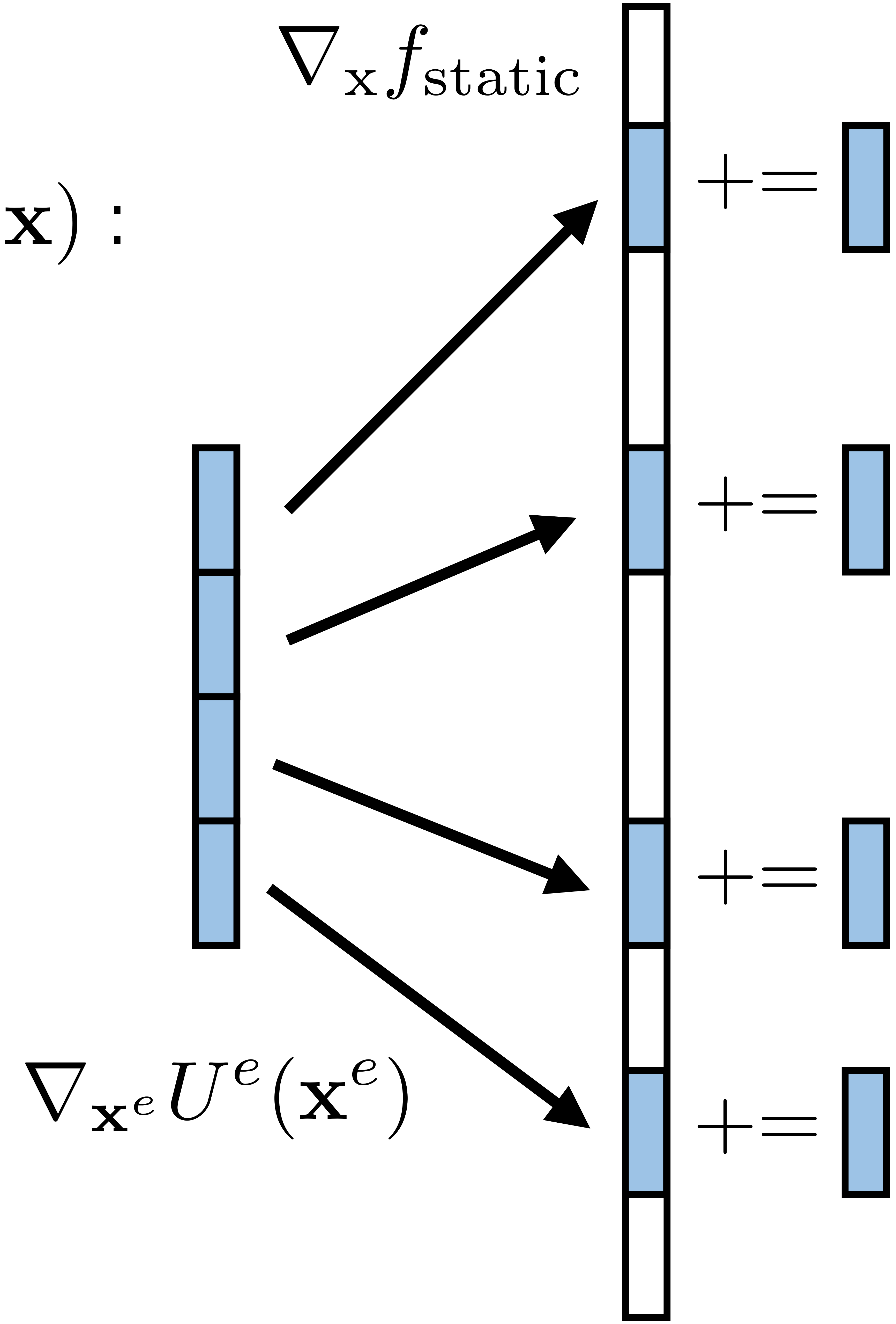
$$\sum_e U^e(\mathbf{x}^e) - \mathbf{f}_{\text{ext}}^T(\mathbf{x} - \mathbf{X})$$

Nonlinear Elasticity: Implementation

Apply Newton's method to objective $f_{\text{static}}(\mathbf{x})$:

2. Write function to evaluate $\nabla_{\mathbf{x}} f_{\text{static}}$ at $\mathbf{x}$

- Set gradient to zero $\nabla_{\mathbf{x}} f_{\text{static}} := \mathbf{0}$
- For each element e, compute $\nabla_{\mathbf{x}^e} U^e(\mathbf{x}^e)$ and add 4 3-vectors to gradient
- Subtract external forces $\mathbf{f}_{\text{ext}}$
- Set entries corresponding to constrained vertices to zero



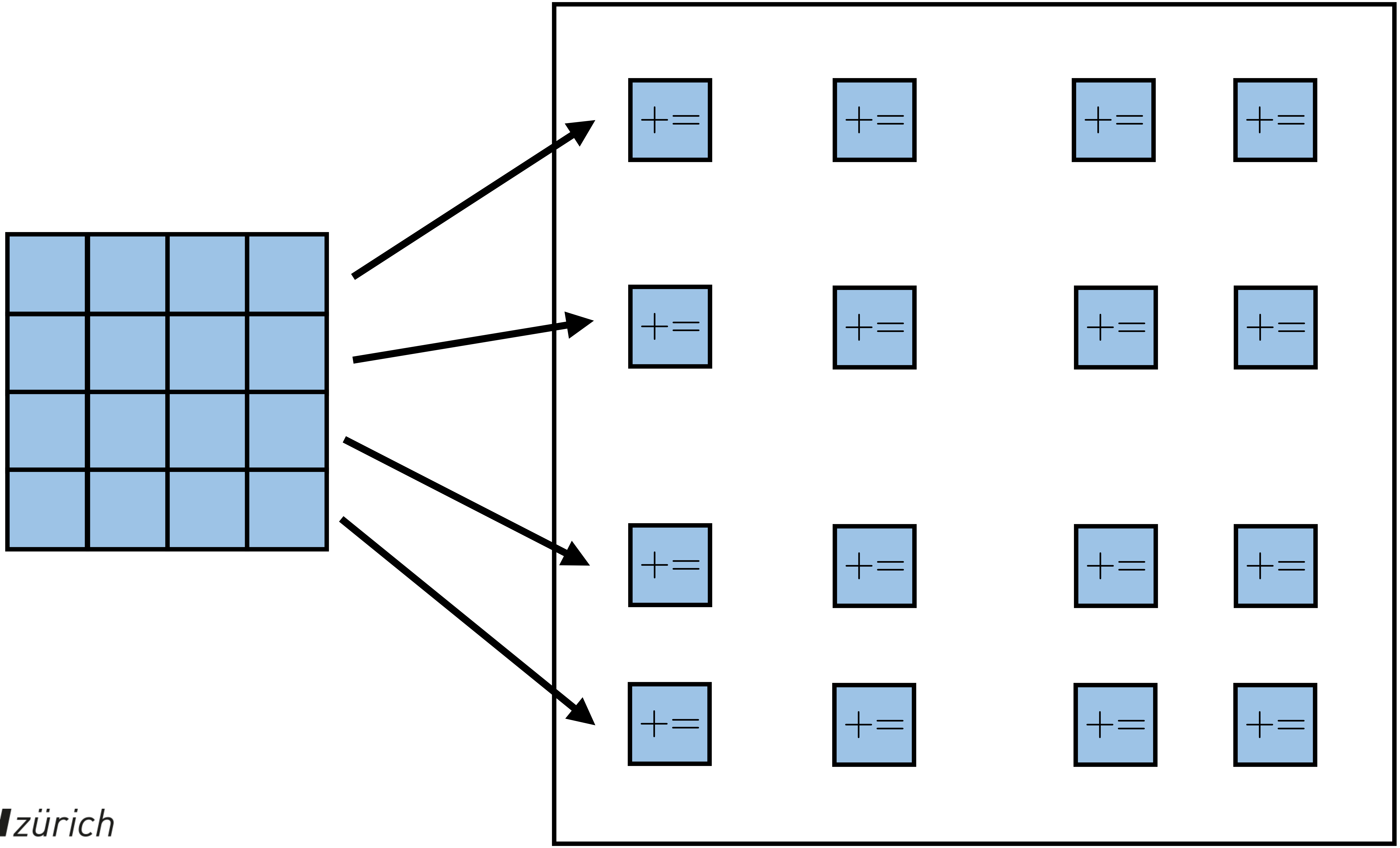
Nonlinear Elasticity: Implementation

Apply Newton's method to objective $f_{\text{static}}(\mathbf{x})$:

3. Write function to evaluate $\nabla^2_{\mathbf{x}} f_{\text{static}}$ at $\mathbf{x}$

- Set Hessian to zero $\nabla^2_{\mathbf{x}} f_{\text{static}} := \mathbf{O}$
- For each element e, compute $\nabla^2_{\mathbf{x}^e} U^e(\mathbf{x}^e)$ and add the 16 3x3-matrices to Hessian
- Set rows and columns corresponding to constrained vertices to zero, then corresponding diagonal elements to 1

Nonlinear Elasticity: Implementation



Dynamics

- Implicit Euler

$$\mathbf{M}\mathbf{a}_n(\mathbf{x}_n^*) + \nabla U(\mathbf{x}_n^*) - \mathbf{f}_{\text{ext}} \stackrel{!}{=} \mathbf{o}$$

Find $\mathbf{x}_n^*$ *that fulfills this "dynamic" equilibrium.*

$$\mathbf{v}_n(\mathbf{x}_n) = \frac{\mathbf{x}_n - \mathbf{x}_p}{h}$$

$$\mathbf{a}_n(\mathbf{x}_n) = \frac{\mathbf{v}_n(\mathbf{x}_n) - \mathbf{v}_p}{h} = \frac{\mathbf{v}_n(\mathbf{x}_n)}{h} - \frac{\mathbf{v}_p}{h} = \frac{\mathbf{x}_n - \mathbf{x}_p}{h^2} - \frac{\mathbf{v}_p}{h}$$

p previous, *known* $\quad$ n next, *unknown*

Dynamics

- Implicit Euler

$$\min_{\mathbf{x}_n} f_{\text{dynamic}}(\mathbf{x}_n)$$

$$\begin{aligned} f_{\text{dynamic}}(\mathbf{x}_n) = & \frac{h^2}{2}\left(\mathbf{a}_n(\mathbf{x}_n)\right)^T \mathbf{M}\mathbf{a}_n(\mathbf{x}_n) && \textit{“inertia”} \\ & + U(\mathbf{x}_n) && \textit{internal energy} \\ & - \mathbf{f}_{\text{ext}}^T(\mathbf{x}_n - \mathbf{X}) && \textit{external energy} \end{aligned}$$

Dynamics Lumped Masses

1. Initialize diagonal matrix $\mathbf{M} \in \mathbb{R}^{3n \times 3n}$ with zero elements

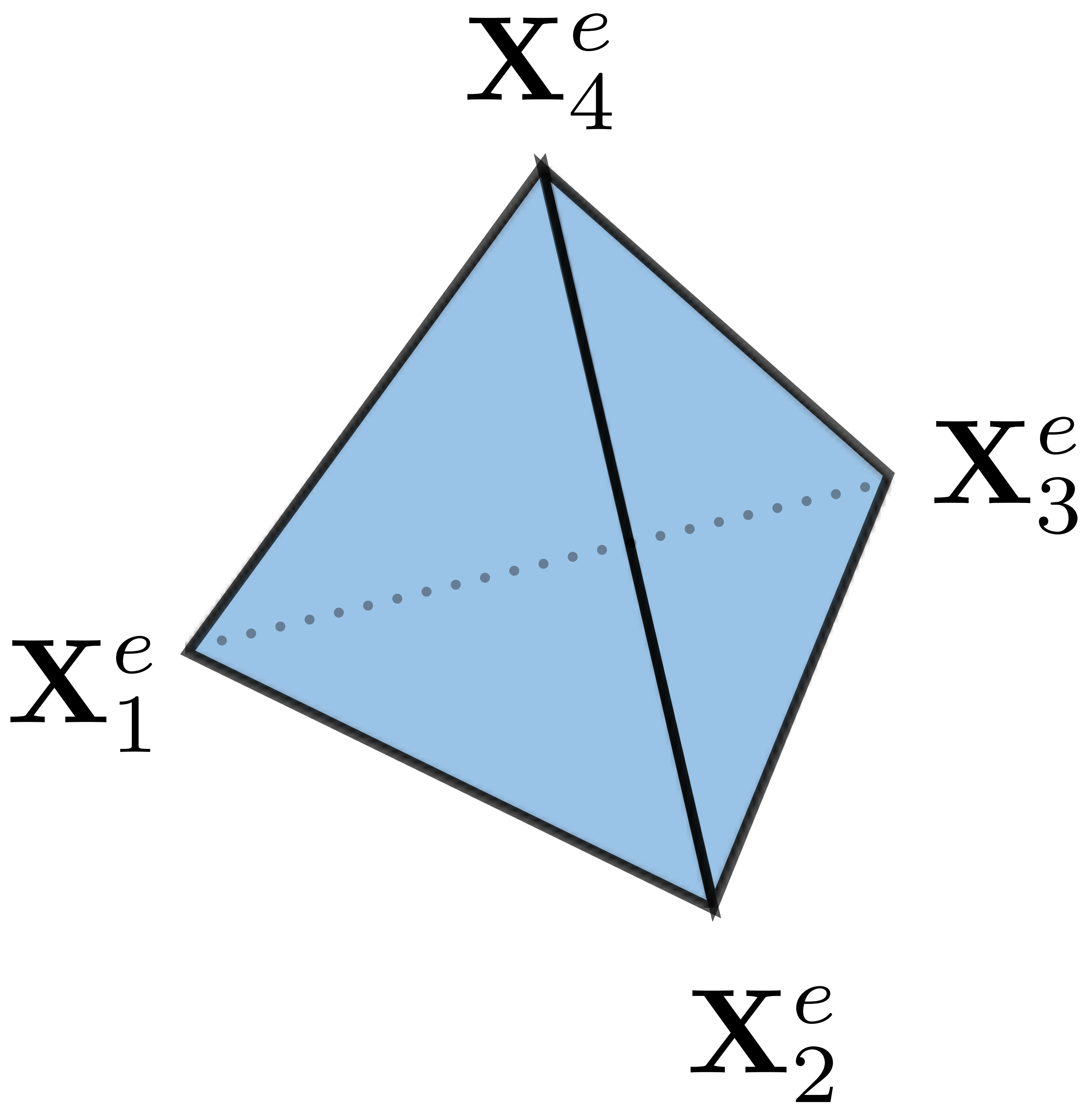
2. For each tetrahedron e :

   - compute the element volume and mass

     $$V^e = \frac{1}{6}\det\left(\begin{bmatrix} \mathbf{X}_1^e - \mathbf{X}_4^e & \mathbf{X}_2^e - \mathbf{X}_4^e & \mathbf{X}_3^e - \mathbf{X}_4^e \end{bmatrix}\right)$$

     $$m^e = V^e \rho \qquad \rho \text{ volumetric mass density}$$

   - add a fourth of m^e to the 12 diagonal elements of $\mathbf{M}$ corresponding to 4 element nodes



References

[Sifakis and Barbič 2012] ***FEM Simulation of 3D Deformable Solids: A Practitioner's Guide to Theory, Discretization and Model Reduction, Part 1,*** *SIGGRAPH Courses, 2012*
Eftychios Sifakis, Jernej Barbič
http://www.femdefo.org/

[Witkin and Baraff 1997] ***Physically Based Modeling: Principles and Practice,*** *SIGGRAPH Course, 1997*
Andrew Witkin, David Baraff
https://www.cs.cmu.edu/~baraff/sigcourse/

[Skouras et al. 2013] ***Computational Design of Actuated Deformable Characters,*** *SIGGRAPH 2013*
M. Skouras, B. Thomaszewski, S. Coros, B. Bickel, M. Gross

[Zehnder et al. 2017] ***MetaSilicone: Design and Fabrication of Composite Silicone with Desired Mechanical Properties,*** *SIGGRAPH 2017*
J. Zehnder, E. Knoop, M. Bächer, B. Thomaszewski

[Schumacher et al. 2018] ***Set-In-Stone: Worst-Case Optimization of Structures Weak in Tension,*** *SIGGRAPH 2018*
C. Schumacher, J. Zehnder, M. Bächer