252-0538-00L, Spring 2018

Shape Modeling and Geometry Processing

Mass-Spring and FEM Simulation

CRL

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Questions

If you have questions about the assignment:

moritzge@inf.ethz.ch

Overview

Today:

- Review relevant theory
  - Optimization: gradient descent (Ex 1.1) and Newton's method (Ex. 1.3)
  - Mass-Spring system: point masses and springs (Ex 1.2 & 1.4)
- Introduction to the code

Next Friday:

- FEM: deformation gradient, Neo-hookean material model
- Introduction to Automatic/Symbolic Differentiation

Optimization: Gradient Descent

Task: find minimizer $\boldsymbol{x}^*$ of function $E(\boldsymbol{x})$,

$$\boldsymbol{x}^* = \mathrm{argmin}_x\, E(\boldsymbol{x}) \;\rightarrow E(\boldsymbol{x}^*) \leq E(\boldsymbol{x})\; \forall\, \boldsymbol{x}$$

In general:

- $E(\boldsymbol{x})$ is a nonlinear function
- $E(\boldsymbol{x})$ is multivariate, i.e., $\boldsymbol{x} \in R^n$ with $n \geq 2$
- $E(\boldsymbol{x})$ may have local minima and maxima (numerical artifacts or expected behavior of physical system)?

Optimization: Gradient Descent

Given a point $\boldsymbol{x}_0$ how do we get to a minimum?

- "Walk" into a direction that decreases $E(\boldsymbol{x})$
- Which direction would you choose?

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n - \alpha \nabla E(\boldsymbol{x}_n)$$

Optimization: Gradient Descent

Taylor-Series expansion of

$$f(\mathbf{x}) \approx f(\mathbf{x}_0) + \nabla f(x_0)^{\mathrm{T}}\mathbf{dx} + \frac{1}{2}\mathbf{dx}\nabla^2 f(\mathbf{x}_0)\mathbf{dx} + \ldots$$

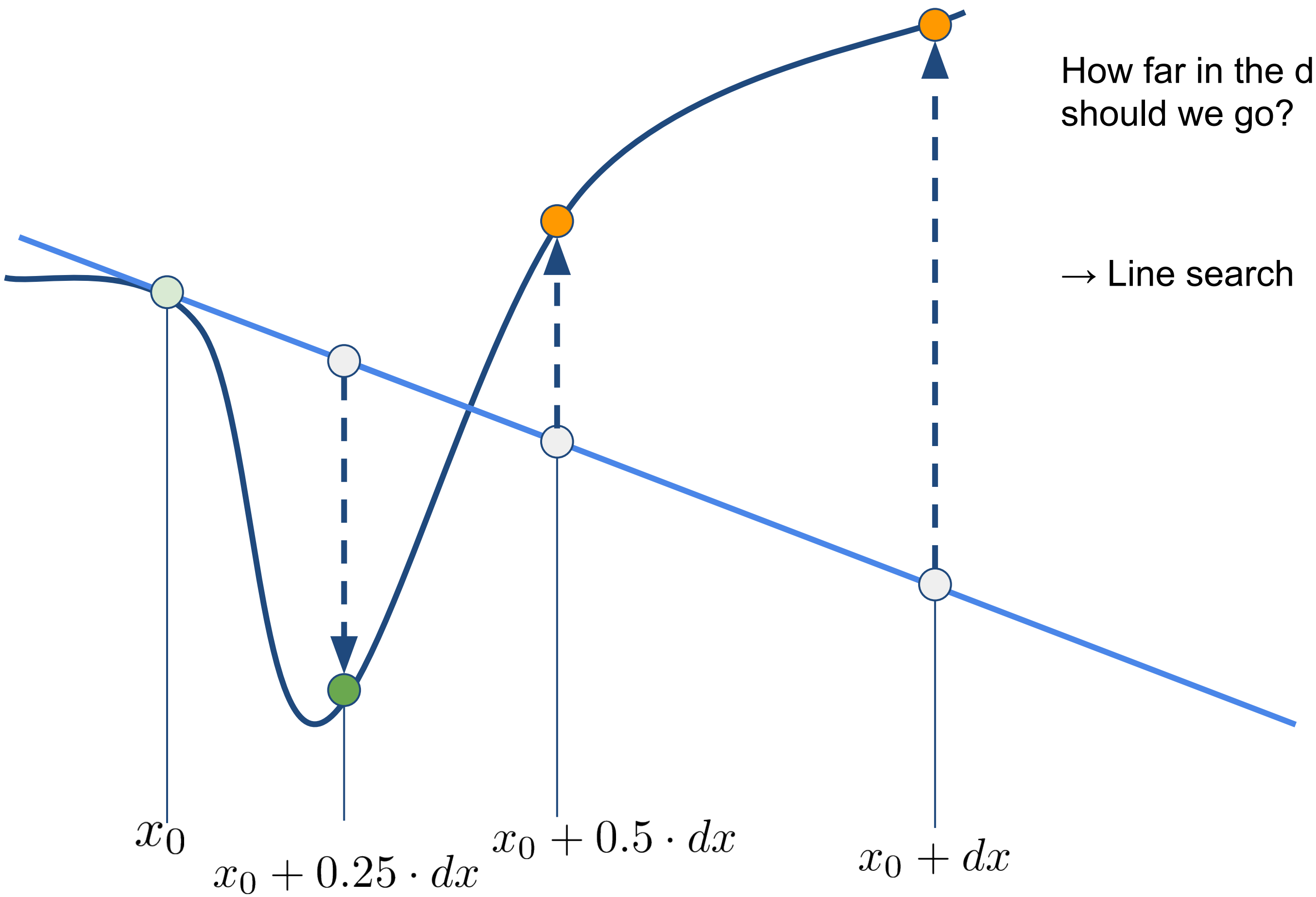
$$O(||\mathbf{dx}||^2)$$

$$f(\mathbf{x}) - f(\mathbf{x}_0) = \nabla f(\mathbf{x}_0)^T\mathbf{dx}$$

want this to be < 0

$$\Rightarrow \mathbf{dx} = -\nabla f(\mathbf{x}_0)$$

Optimization: Line Search



How far in the direction of the gradient should we go?

→ Line search

Optimization: Gradient Descent + Line Search

Algorithm: line_search
Input: x, dx, α, β
while $E(x - \alpha * dx) > E(x)$ **do**
$\quad \alpha = \alpha * \beta;$
end do;

Algorithm: steepest_descent
Input: $x, dx, \alpha, \beta, \varepsilon$
while $\text{abs}(\nabla E(x)) > \varepsilon$ **do**
$\quad dx = \nabla E(x);$
$\quad \alpha = \text{line\_search}(x, dx, \alpha, \beta);$
$\quad x = x - \alpha\, dx;$
end do;

x	current state
dx	search direction
α	initial step length
$0 < \beta < 1$	scaling factor

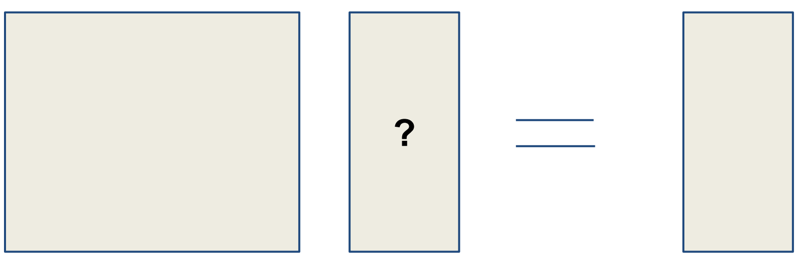
Optimization: Newton's Method

Taylor-Series expansion of the **gradient**

$$\nabla f(\mathbf{x}) = \nabla f(x_0) + \nabla^2 f(\mathbf{x}_0)^T \mathbf{dx} + \ldots$$

$$\nabla f(\mathbf{x}) = 0$$

$$\nabla^2 f(\mathbf{x}_0)^T \mathbf{dx} = -\nabla f(x_0)$$



Optimization: Newton's Method

Taylor-Series expansion of

$$f(\mathbf{x}) \approx f(\mathbf{x}_0) + \nabla f(x_0)^{\mathrm{T}}\mathbf{dx} + \frac{1}{2}\mathbf{dx}\nabla^2 f(\mathbf{x}_0)\mathbf{dx} + \ldots$$

$$O(||\mathbf{dx}||^3)$$

$$\nabla_{dx}(\ldots) = 0$$

$$\nabla^2 f(\mathbf{x}_0)^T\mathbf{dx} = -\nabla f(x_0)$$

Optimization: Newton's Method

Algorithm: steepest_descent

Input: $x, \alpha, \beta, \varepsilon$

while abs($\nabla E(x)$) $> \varepsilon$ **do**

$dx = -\nabla E(x);$

$\alpha = \text{line\_search}(x, dx, \alpha, \beta);$

$x = x + \alpha\, dx;$

end do;

Algorithm: newton

Input: $x, \alpha, \beta, \varepsilon$

while abs($\nabla E(x)$) $> \varepsilon$ **do**

$dx = -\nabla^2 E^{-1}(\mathrm{x}) * \nabla E(x);$

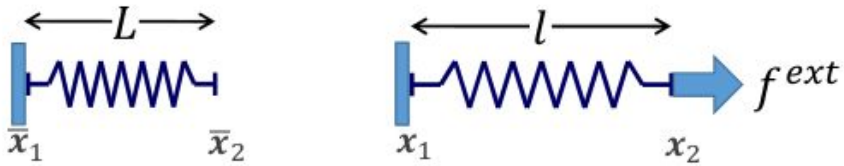
$\alpha = \text{line\_search}(x, dx, \alpha, \beta);$

$x = x + \alpha dx;$

end do;

Mass-Spring System

- 1 mass point, 1 spring



Deformation Measure

$$\varepsilon = \frac{l}{L} - 1$$

Elastic Energy

$$E = \frac{1}{2}\tilde{k}\varepsilon^2 L$$

Forces

$$\boldsymbol{f}_{int} = -\frac{\partial E(\boldsymbol{x})}{\partial \boldsymbol{x}}$$

Working it out…

$$l = |\boldsymbol{e}|_2 = (\boldsymbol{e}^T\boldsymbol{e})^{\frac{1}{2}}$$

with $\boldsymbol{e} = \boldsymbol{x}_2 - \boldsymbol{x}_1$

$$\boldsymbol{f}_1 = -\frac{\partial E(\boldsymbol{x}_1, \boldsymbol{x}_2)}{\partial \boldsymbol{x}_1} = -\frac{\partial E(\boldsymbol{x}_1, \boldsymbol{x}_2)}{\partial l}\frac{\partial l}{\partial \boldsymbol{x}_1}$$

$$\frac{\partial E}{\partial l} = \tilde{k}\varepsilon$$

$$\frac{\partial l}{\partial \boldsymbol{x}_1} = \frac{1}{2}(\boldsymbol{e}^T\boldsymbol{e})^{-\frac{1}{2}}\frac{\partial(\boldsymbol{e}^T\boldsymbol{e})}{\partial \boldsymbol{x}_1}$$

$$\boldsymbol{f}_1 = -\tilde{k}(\frac{l}{L} - 1)\frac{\boldsymbol{x}_2 - \boldsymbol{x}_1}{|\boldsymbol{x}_2 - \boldsymbol{x}_1|}$$

$$\boldsymbol{f}_1 = -\boldsymbol{f}_2$$

10

Mass-Spring System

Non-zero Length Spring

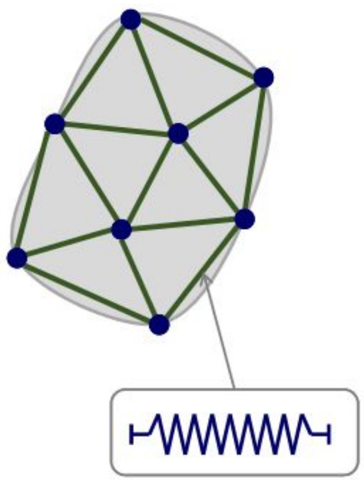
$$F = -k \underbrace{\left(\frac{|x_i - x_j|}{L} - 1 \right)}_{\varepsilon} \underbrace{\frac{x_i - x_j}{|x_i - x_j|}}_{u} = -k\varepsilon \frac{u}{|u|}$$

$$\frac{\partial F}{\partial x_i} = -\frac{\partial F}{\partial x_j} = -k\left(\frac{1}{L}\frac{uu^T}{u^T u} + \frac{\varepsilon}{|u|}\left(I - \frac{uu^T}{u^T u} \right) \right)$$

$$\mathbf{H} = \frac{\partial^2 E}{\partial \boldsymbol{x}^2} = -\frac{\partial F}{\partial \boldsymbol{x}}$$

Mass-Spring System

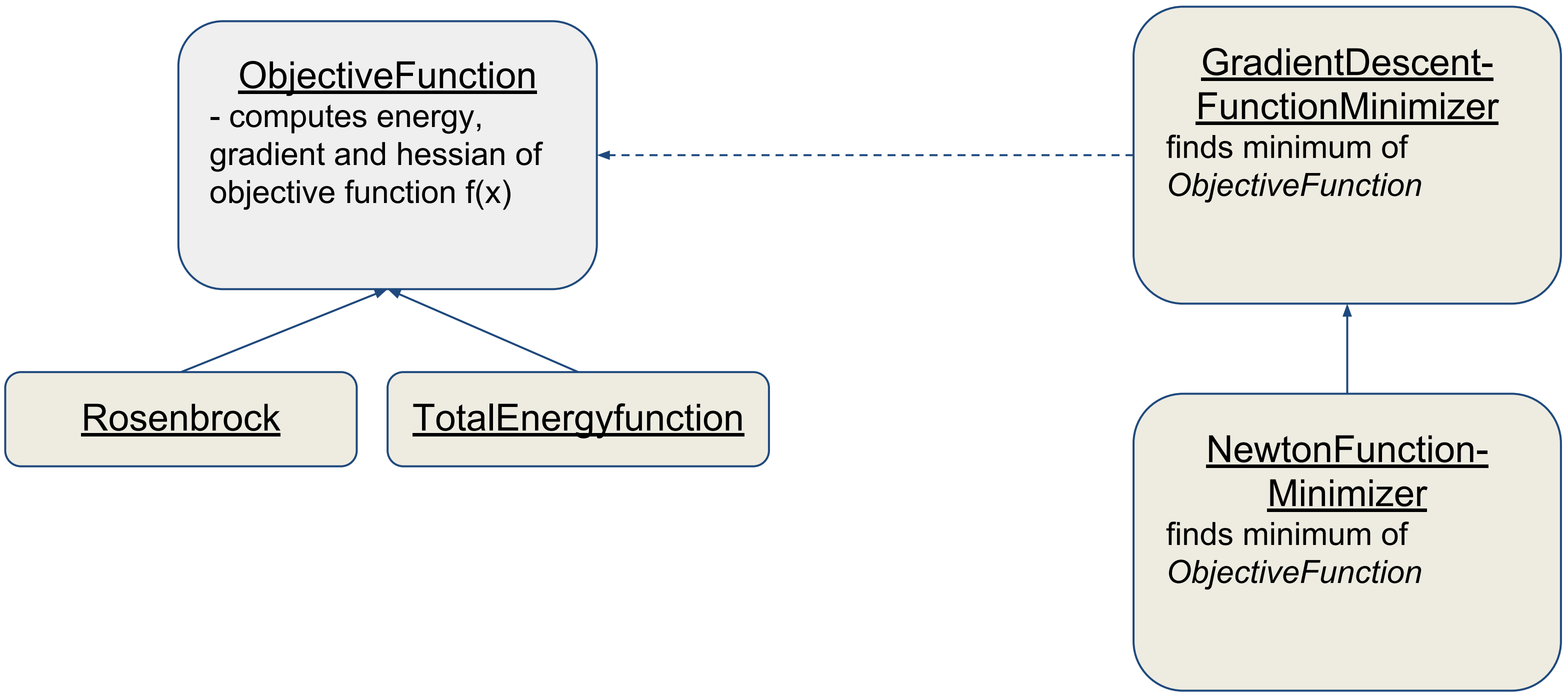
Straightforward concept: sample object with mass points, connect them with springs…



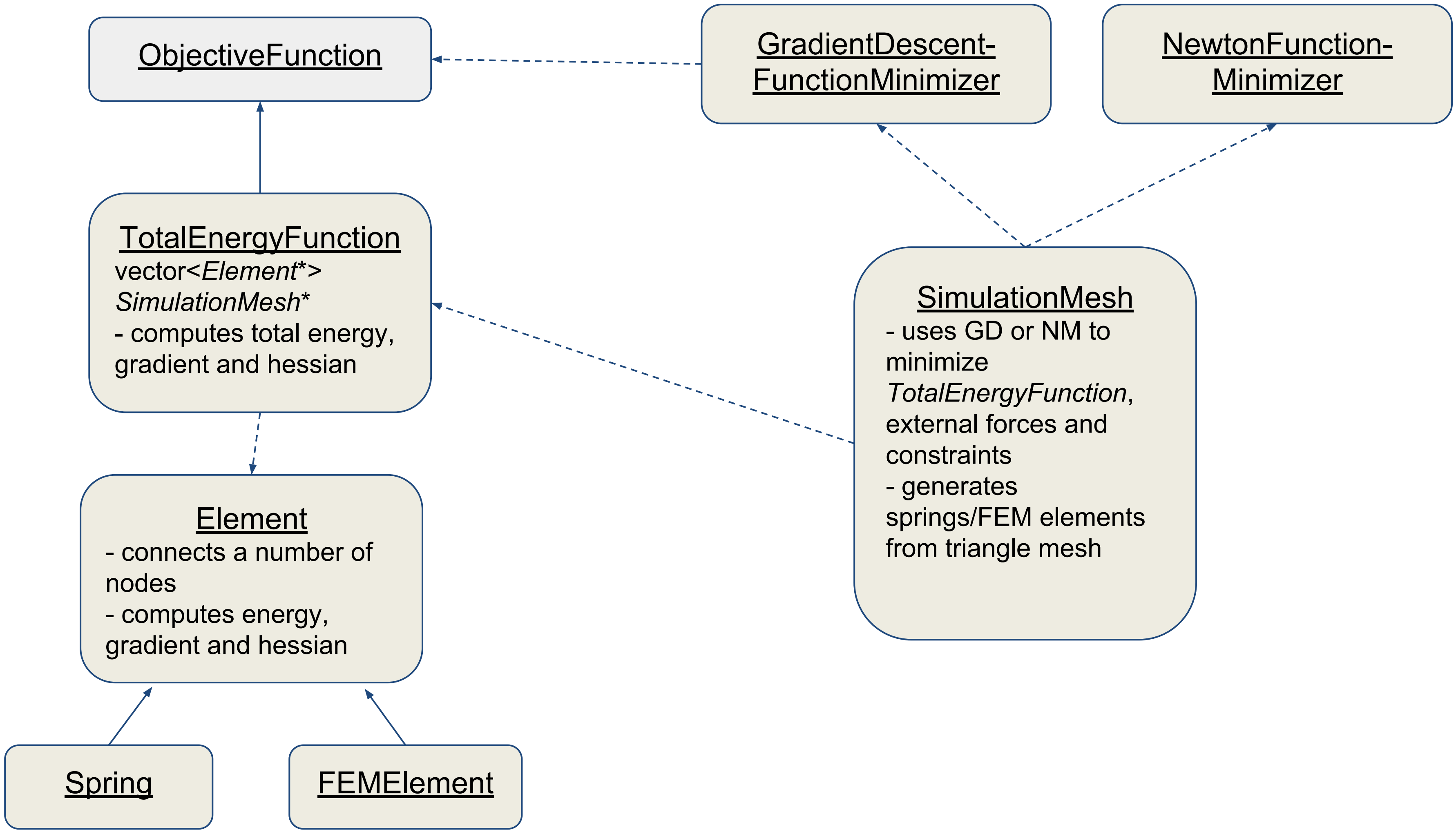
Representation: 2D triangle mesh (or 3D tetrahedral mesh, of course)

- Vertices $\mathbf{x}_i \in \mathbf{R}^2$
- Edges E_{ij} are springs connecting vertices $\mathbf{x}_i$ and $\mathbf{x}_j$

Code Structure: Optimization



Code Structure: Simulation



Next Week

- Exercise 2
- FEM
- Automatic Differentiation