Euler-Lagrange Formalism

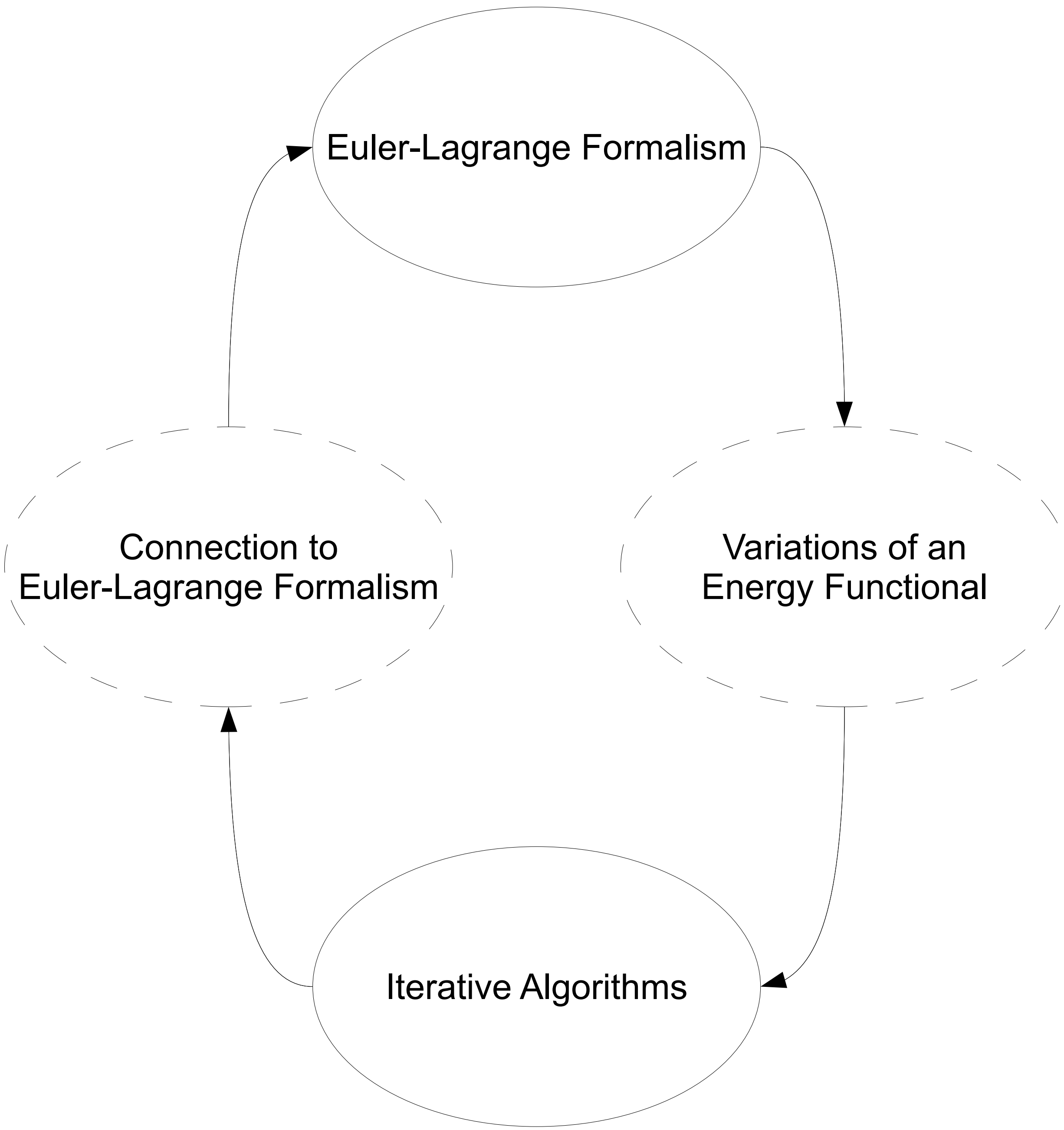
Jan Hendrik Dithmar
jadi5003@stud.uni-saarland.de

July 18th, 2009

Motivation

- Euler-Lagrange equations appear in
  - (classical) mechanics,
  - functional minimisation,
  - field theory,
  - ...
- Connections to iterative algorithms (Gradient Descent, Newton's method) to solve optimisation problems.

Outline


Euler-Lagrange Formalism
Connection to
Euler-Lagrange Formalism
Variations of an
Energy Functional
Iterative Algorithms

Basic observations

Standard calculus

- Considers real-valued functions $f : \mathbb{R}^N \to \mathbb{R}$, $(\xi_1, ..., \xi_N) \in \mathbb{R}^N$.
- If f has minimum in ξ, then $f'(\xi) = 0$.
- If f strictly convex and $f'(\xi) = 0$, then ξ is unique minimum of f.

Calculus of variations

- Considers real-valued functionals $\mathcal{E} : \mathcal{F} \to \mathbb{R}$, where $\mathcal{F}$ is a function space.
- If $\mathcal{E}$ minimised by function u, then u has to satisfy so-called *Euler-Lagrange equations*.
- If $\mathcal{E}$ strictly convex and satisfies Euler-Lagrange equations, then u is unique minimiser of $\mathcal{E}$.

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27

Functionals

- Structure (with regularised smoothness term)

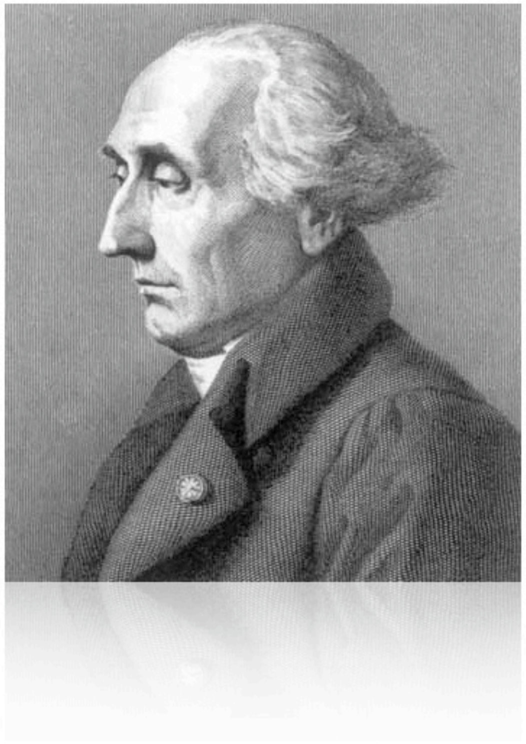
$$\mathcal{E}(u) = \int_a^b \underbrace{(u-f)^2}_{\text{data term}} + \alpha \underbrace{\Psi_S(|\nabla u|^2)}_{\text{smoothness term}} \, \mathrm{d}\mathbf{x}$$

  - u: (unknown) function
  - α: smoothness weight
  - Ψ_S: regulariser, penalises deviations from smoothness

- *Remark:* Data term can be robustified by using a penaliser Ψ_D.

  ($\rightarrow$ Talk by Laurent Hoeltgen about "Approximation and Fitting")

Euler-Lagrange equations



Left: Leonhard Euler (1707–1783). **Right:** Joseph-Louis Lagrange (1736–1813). **Source:** Wikipedia.

Theorem (1-D case)

A *smooth function* $u(x)$ with $x \in [a, b]$ that minimises the 1-D energy functional

$$\mathcal{E}(u) = \int_a^b \mathcal{L}(x, u, u') \, \mathrm{d}x$$

satisfies necessarily the *Euler-Lagrange equation*

$$\mathcal{L}_u - \frac{\mathrm{d}}{\mathrm{d}x} \mathcal{L}_{u'} = 0$$

and the so-called *natural boundary conditions*

$$\mathcal{L}_{u'} = 0$$

for the boundary ($x = a$ and $x = b$).

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27

Theorem (2-D case)

A minimiser of the 2-D energy functional

$$\mathcal{E}(u) = \int_\Omega \mathcal{L}(x_1, x_2, u, u_{x_1}, u_{x_2}) \, \mathrm{d}\mathbf{x}$$

satisfies necessarily the *Euler-Lagrange equation*

$$\mathcal{L}_u - \frac{\partial}{\partial x_1}\mathcal{L}_{u_{x_1}} - \frac{\partial}{\partial x_2}\mathcal{L}_{u_{x_2}} = 0$$

with the *natural boundary conditions*

$$\mathbf{n}^\top \begin{pmatrix} \mathcal{L}_{u_{x_1}} \\ \mathcal{L}_{u_{x_2}} \end{pmatrix} = 0$$

at the image boundaries $\partial\Omega$ with the normal vector $\mathbf{n}$.

Example for a Variational Problem

Problem: Find the shortest plane curve joining two points A and B.

Example for a Variational Problem

Problem: Find the shortest plane curve joining two points A and B.

This means: Find the curve $u(x)$ for which the functional

$$\mathcal{E}(u(x)) = \int_a^b \sqrt{1 + \left(\frac{\mathrm{d}}{\mathrm{d}x} u(x)\right)^2} \, \mathrm{d}x$$

achieves its minimum where $a \leq x \leq b$.

Solution: Straight line segment joining A and B. Why?

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27

Example for a Variational Problem

Problem: Find the shortest plane curve joining two points A and B.

This means: Find the curve $u(x)$ for which the functional

$$\mathcal{E}(u(x)) = \int_a^b \sqrt{1 + \left(\frac{\mathrm{d}}{\mathrm{d}x} u(x)\right)^2} \,\mathrm{d}x$$

achieves its minimum where $a \leq x \leq b$.

Solution: Straight line segment joining A and B. Why?

The Euler-Lagrange equation yields the differential equation

$$\frac{\mathrm{d}}{\mathrm{d}u} \sqrt{1 + \left(\frac{\mathrm{d}}{\mathrm{d}x} u(x)\right)^2} = 0 \quad \Longrightarrow \quad \frac{\mathrm{d}}{\mathrm{d}x} u(x) = C,$$

in other words a straight line.

Variations of an Energy Functional

Notation

- Continuous
  - u is a function
  - $\mathcal{E}$ is a functional (takes functions as arguments)
- Discrete
  - $\mathbf{u}$ is a vector representing function values
  - E is a function (takes vectors as arguments)

What is it about?

- *Problem:* How to compute a derivative w.r.t. a function?
- *Remedy:* Construct *something* that we can differentiate.

Variations of an Energy Functional

- *Problem:* $\mathcal{E} : \mathcal{F} \to \mathbb{R}$.
- *Remedy:* $\mathcal{P} : \mathbb{R} \to \mathbb{R}$.

Variations of an Energy Functional

- *Problem:* $\mathcal{E} : \mathcal{F} \to \mathbb{R}$.
- *Remedy:* $\mathcal{P} : \mathbb{R} \to \mathbb{R}$.
- *We do:* Construct $\mathcal{P}$, s.t. $\mathcal{P} : \mathbb{R} \to \mathbb{R}$
  - $\mathcal{S} : \mathbb{R} \to \mathcal{F}$; $\varepsilon \mapsto u + \varepsilon v$ brings us to function space $\mathcal{F}$.
  - $\mathcal{E} : \mathcal{F} \to \mathbb{R}$ brings us back to $\mathbb{R}$.

  $\Rightarrow$ $\mathcal{P} : \mathbb{R} \to \mathbb{R}$; $\varepsilon \mapsto \mathcal{E}(u + \varepsilon v)$.

- *Remark:* If $\mathcal{F} = \mathbb{R}^N$, then $\mathbf{u} = (u_1, ..., u_N)$ and $\mathbf{v} = (v_1, ..., v_N)$.
- first variation $\approx$ first derivative
- second variation $\approx$ second derivative

Overview

Continuous	*Discrete*
First Variation	?
$\frac{\partial}{\partial \varepsilon} \mathcal{E}(u+\varepsilon v)\Big\vert_{\varepsilon=0}$	$\frac{\partial}{\partial \varepsilon} E(\mathbf{u}+\varepsilon \mathbf{v})\Big\vert_{\varepsilon=0} = \quad ?$
Second Variation	?
$\frac{\partial^2}{\partial \varepsilon^2} \mathcal{E}(u+\varepsilon v)\Big\vert_{\varepsilon=0}$	$\frac{\partial^2}{\partial \varepsilon^2} E(\mathbf{u}+\varepsilon \mathbf{v})\Big\vert_{\varepsilon=0} = \quad ?$

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27

First Derivative: Gradient

- First derivative in $\mathbf{u} \in \mathbb{R}^N$ in direction $\mathbf{v}$ is given by

$$\begin{aligned} \mathbf{D}E(\mathbf{u})\mathbf{v} &= \frac{\partial}{\partial \varepsilon} E(\mathbf{u} + \varepsilon \mathbf{v})\Big|_{\varepsilon=0} \\ &= \nabla E(\mathbf{u}) \cdot \mathbf{v} \end{aligned}$$

Second Derivative: Hessian

- Second derivative in $\mathbf{u} \in \mathbb{R}^N$ in direction $\mathbf{v}$ is given by

$$
\begin{aligned}
\mathbf{D}^2 E(\mathbf{u})\mathbf{v} &= \frac{\partial^2}{\partial \varepsilon^2} E(\mathbf{u} + \varepsilon \mathbf{v}) \Big|_{\varepsilon=0} \\
&= \frac{\partial}{\partial \varepsilon} (\nabla E(\mathbf{u} + \varepsilon \mathbf{v}) \cdot \mathbf{v}) \Big|_{\varepsilon=0} \\
&= \frac{\partial}{\partial \varepsilon} \left(\left(\frac{\partial}{\partial u_1} E(\mathbf{u} + \varepsilon \mathbf{v}), \dots, \frac{\partial}{\partial u_N} E(\mathbf{u} + \varepsilon \mathbf{v}) \right) \cdot \mathbf{v} \right) \Big|_{\varepsilon=0} \\
&= \left(\left(\left(\frac{\partial^2}{\partial u_1 \partial u_i} E(\mathbf{u}) \right), \dots, \left(\frac{\partial^2}{\partial u_N \partial u_i} E(\mathbf{u}) \right) \right)_{i=1,\dots,N} \cdot \mathbf{v} \right) \cdot \mathbf{v} \\
&= \left(\left(\underbrace{\left(\frac{\partial^2}{\partial u_j \partial u_i} E(\mathbf{u}) \right)}_{\mathcal{H}E(\mathbf{u})} \right)_{i,j=1,..,N} \cdot \mathbf{v} \right) \cdot \mathbf{v} \\
&= \mathbf{v}^\top \mathcal{H} E(\mathbf{u}) \mathbf{v}
\end{aligned}
$$

Variations of an Energy Functional

Overview

Continuous	*Discrete*
First Variation	**Gradient**
$\frac{\partial}{\partial \varepsilon} \mathcal{E}(u + \varepsilon v)\Big\vert_{\varepsilon=0}$	$\frac{\partial}{\partial \varepsilon} E(\mathbf{u} + \varepsilon \mathbf{v})\Big\vert_{\varepsilon=0} = \nabla E(\mathbf{u}) \cdot \mathbf{v}$
Second Variation	**Hessian Matrix**
$\frac{\partial^2}{\partial \varepsilon^2} \mathcal{E}(u + \varepsilon v)\Big\vert_{\varepsilon=0}$	$\frac{\partial^2}{\partial \varepsilon^2} E(\mathbf{u} + \varepsilon \mathbf{v})\Big\vert_{\varepsilon=0} = \mathbf{v}^\top \mathcal{H} E(\mathbf{u}) \mathbf{v}$

Appearance of the concept

- *Assume:* energy functional = cost function (discrete!)
- *Idea:* Minimise the cost
- *Ingredients:*
  - $\nabla E(\mathbf{u})$: gradient of the energy functional

$$\nabla E(\mathbf{u}) = \left(\frac{\partial}{\partial u_j} \left(E(\mathbf{u}) \right) \right)_{j=1,\ldots,N}$$

  - $\mathcal{H}E(\mathbf{u})$: Hessian matrix of the energy functional

$$\mathcal{H}E(\mathbf{u}) = \left(\frac{\partial^2}{\partial u_j \partial u_i} \left(E(\mathbf{u}) \right) \right)_{i,j=1,\ldots,N}$$

Gradient Descent Algorithm

- give a starting point x
- repeat
  1. $\mathbf{d} = -\nabla E(\mathbf{u})$
  2. Choose step size γ (e.g. by backtracking line search)
  3. $\mathbf{x} := \mathbf{x} + \gamma \cdot \mathbf{d}$

  until stopping criterion is satisfied.

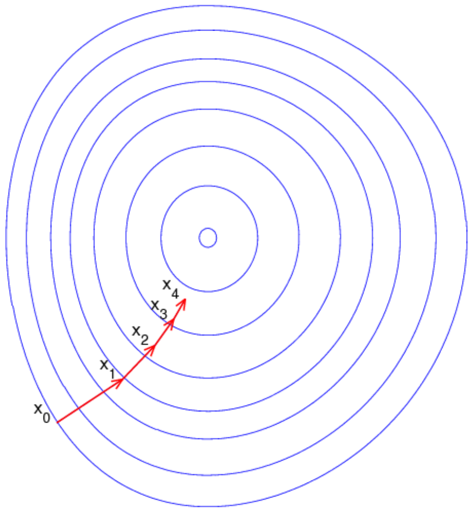


Illustration of gradient descent. **Source:** Wikipedia.

Newton's Method

- give a starting point x
- repeat
  1. $\mathbf{d} := -\left[\mathcal{H}E(\mathbf{u})\right]^{-1}\nabla E(\mathbf{u})$
  2. $\mathbf{x} := \mathbf{x} + \mathbf{d}$

  until stopping criterion is satisfied.

Need: strict convexity $\Longleftrightarrow$ Hessian positive definite

Given: Hessian $\mathcal{H}E(u)$ not (always) positive definite

Solution: Perturb it to be positive definite! $\rightarrow$ Trust-Region Method

Connection to the Euler-Lagrange Formalism

- Assume the energy functional

$$\mathcal{E}(u) = \int_a^b (u-f)^2 + \alpha \Psi_S\big((u')^2\big)\,\mathrm{d}x$$

- Two approaches:
  - First employ necessary optimality condition, then discretise. (OD)
  - First discretise, then employ necessary optimality condition. (DO)
- What's better? OD or DO?

Necessary optimality condition → Discretise: OD

The Euler-Lagrange equation

$$
\begin{aligned}
0 &= 2(u-f) - \frac{\mathrm{d}}{\mathrm{d}x}\Big[2\alpha \Psi_S'\big((u')^2\big)\cdot u'\Big] \\
&= u - f - \alpha \frac{\mathrm{d}}{\mathrm{d}x}\Big[\Psi_S'\big((u')^2\big)\cdot u'\Big]
\end{aligned}
$$

After discretisation

$$
\begin{aligned}
0 &= u_j - f_j - \alpha \frac{\mathrm{d}}{\mathrm{d}x}\Big[\Psi_S'\big((u')^2\big)\cdot u'\Big]\Big|_{x=x_j} \\
&= u_j - f_j - \alpha \Psi_S'\left(\frac{(u_{j+1}-u_j)^2}{\Delta x^2}\right)\cdot \frac{u_{j+1}-u_j}{\Delta x^2} \\
&\qquad + \alpha \Psi_S'\left(\frac{(u_j-u_{j-1})^2}{\Delta x^2}\right)\cdot \frac{u_j-u_{j-1}}{\Delta x^2} \qquad , j = 1, ..., N.
\end{aligned}
$$

Discretise → Necessary optimality condition: DO

Discretised energy functional

$$E(\mathbf{u}) = \sum_{j=1}^{N} (u_j - f_j)^2 + \alpha \Psi_S\Big((\mathbf{u}')^2\Big)\Big|_{x=x_j}.$$

Necessary optimality condition

$$\begin{aligned} 0 &= \frac{\partial}{\partial u_j}\left(E(\mathbf{u})\right) \\ &= u_j - f_j - \alpha \Psi_S'\left(\frac{(u_{j+1}-u_j)^2}{\Delta x^2}\right) \cdot \frac{u_{j+1}-u_j}{\Delta x^2} \\ &\qquad + \alpha \Psi_S'\left(\frac{(u_j-u_{j-1})^2}{\Delta x^2}\right) \cdot \frac{u_j-u_{j-1}}{\Delta x^2} \qquad, j = 1, ..., N. \end{aligned}$$

Example in Image Processing: Gaussian Noise

Consider

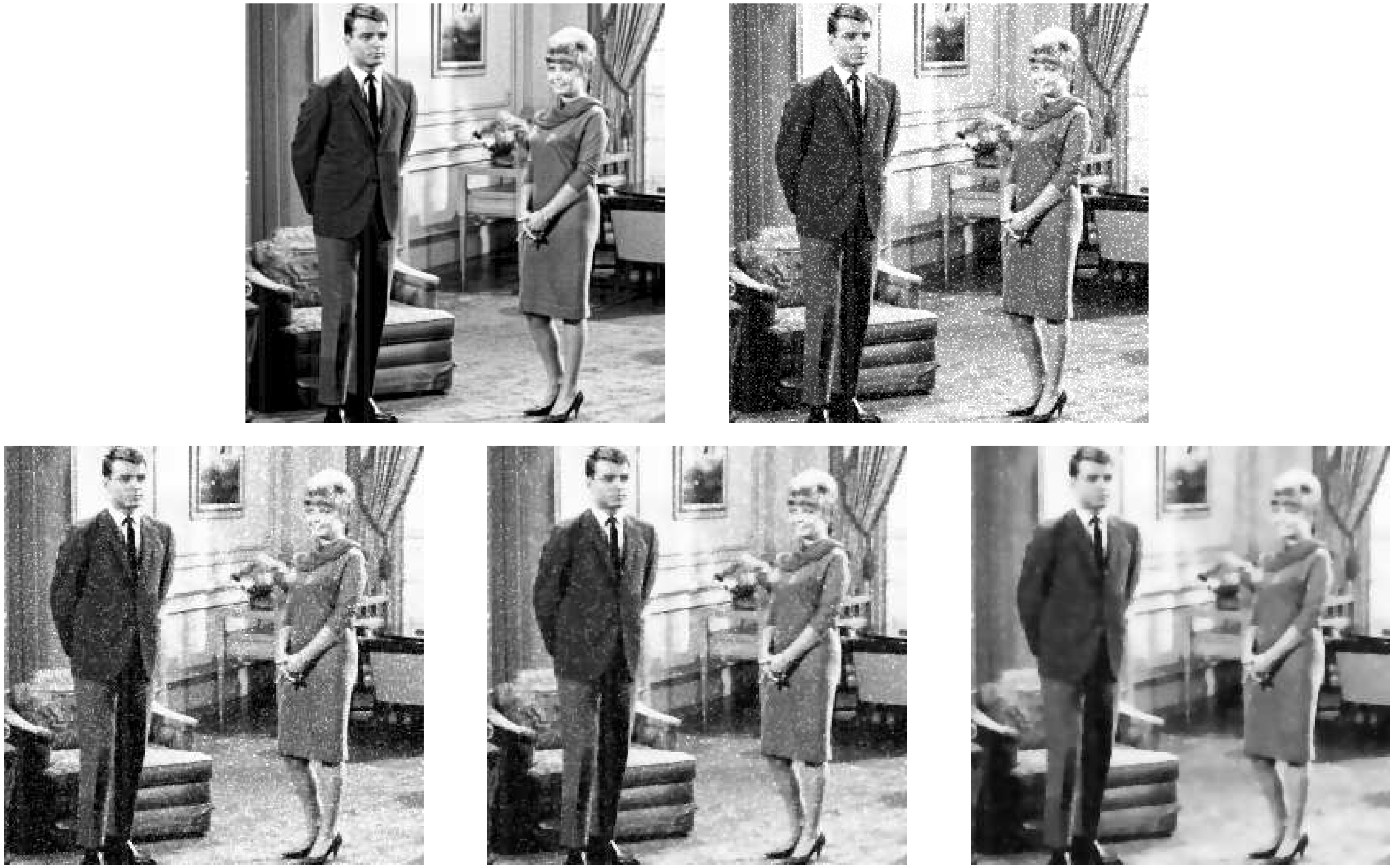
$$\mathcal{E}(u) = \int_a^b (u-f)^2 + \alpha \Psi_S(|\nabla u|^2)\,\mathrm{d}x$$

with Charbonnier regulariser

$$\Psi_S(|\nabla u|^2) = 2(\lambda^2 \sqrt{1+|\nabla u|^2/\lambda^2} - \lambda^2)$$

Left: Original image. **Right:** Noisy version with Gaussian noise ($\sigma = 40$).

Denoised version after 20 iterations



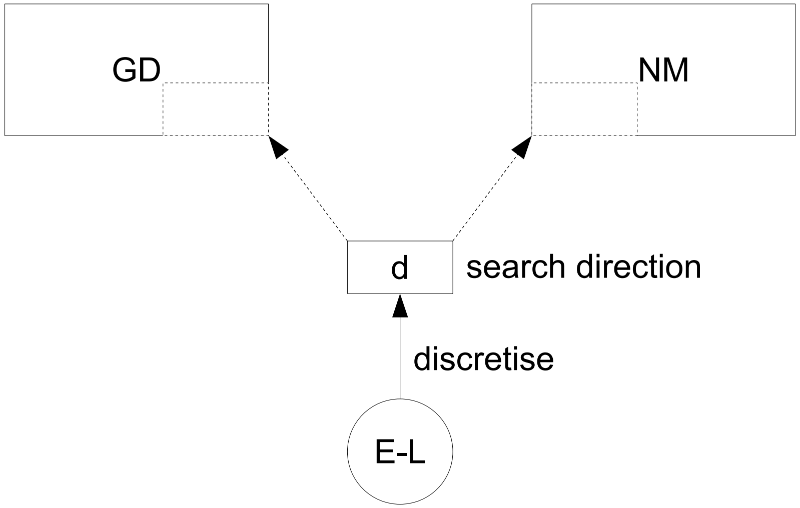
Top Left: Original image. **Top Right:** Noisy version with Gaussian noise ($\sigma = 40$). **Bottom row:** Denoised version with $\lambda = 1, 2, 5$.

Example in Image Processing: Salt-and-Pepper Noise

Left: Noisy version with salt-and-pepper noise. **Right:** Denoised version using the regularised data term $\Psi_D(s^2) = 2(\sqrt{s^2+\varepsilon^2} - \varepsilon)$ and a Charbonnier regulariser as smoothness term $\Psi_S(s^2) = 2(\lambda^2\sqrt{1+s^2/\lambda^2} - \lambda^2)$. **Author:** J. Weickert.

Summary

- Euler-Lagrange equation: necessary condition for a minimiser of an energy functional.
- Several approaches use the Euler-Lagrange theory. Good for solving optimisation problems.
- Gradient Descent Method and Newton's Method are algorithms to find a local extremum. The Euler-Lagrange formalism is related to these methods:

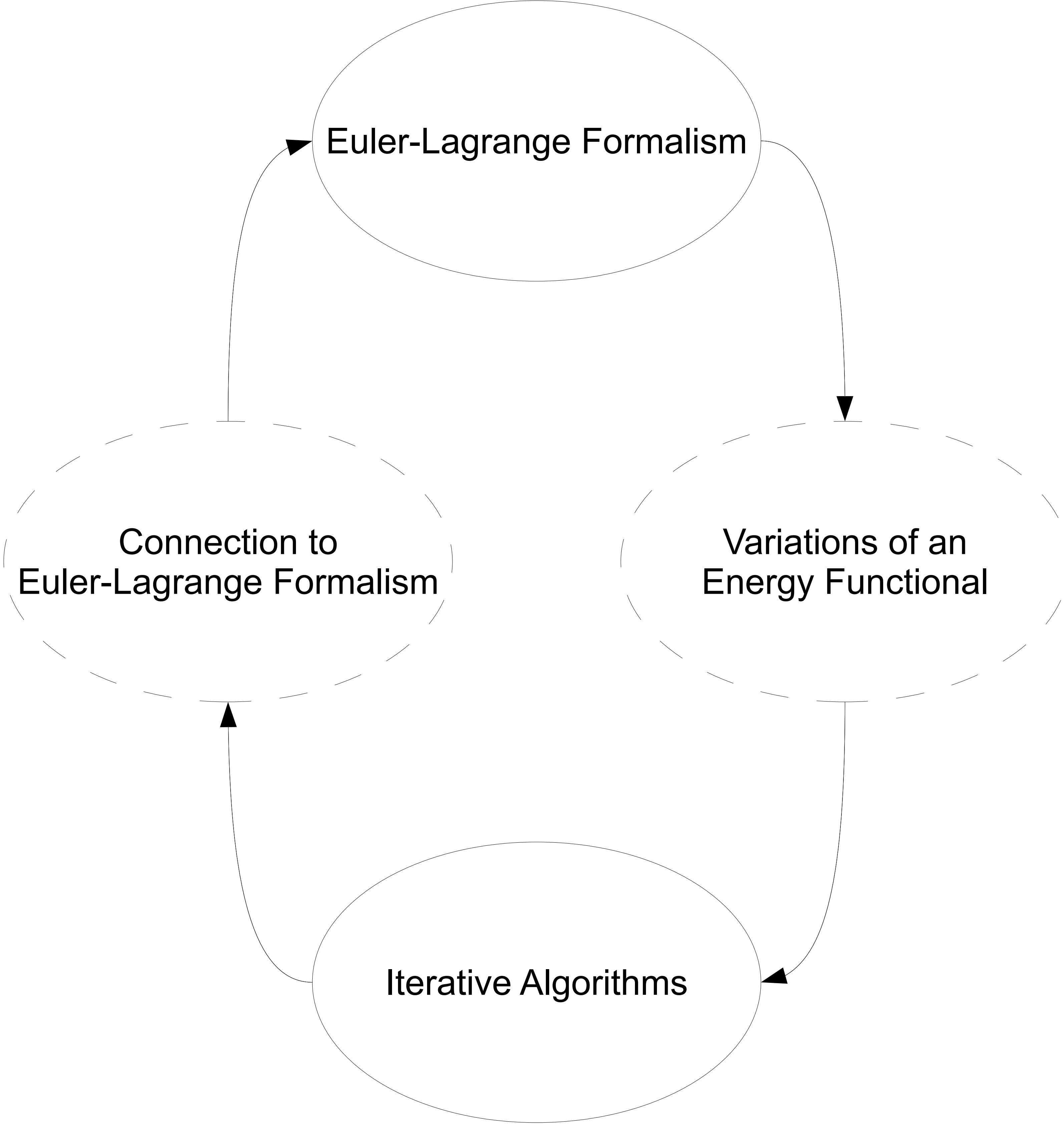


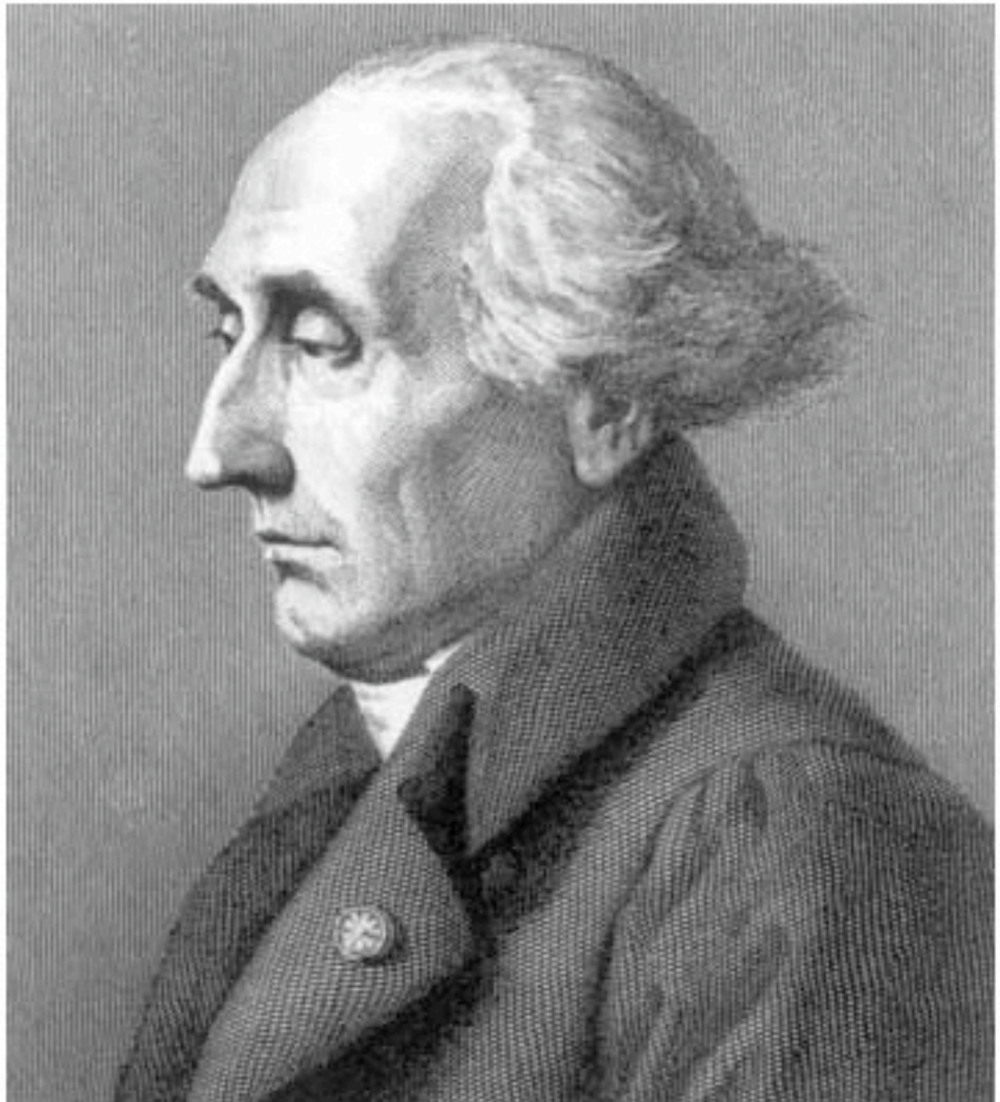
- OD and DO lead to the same result.
- Energy functional minimisation is used for image denoising.

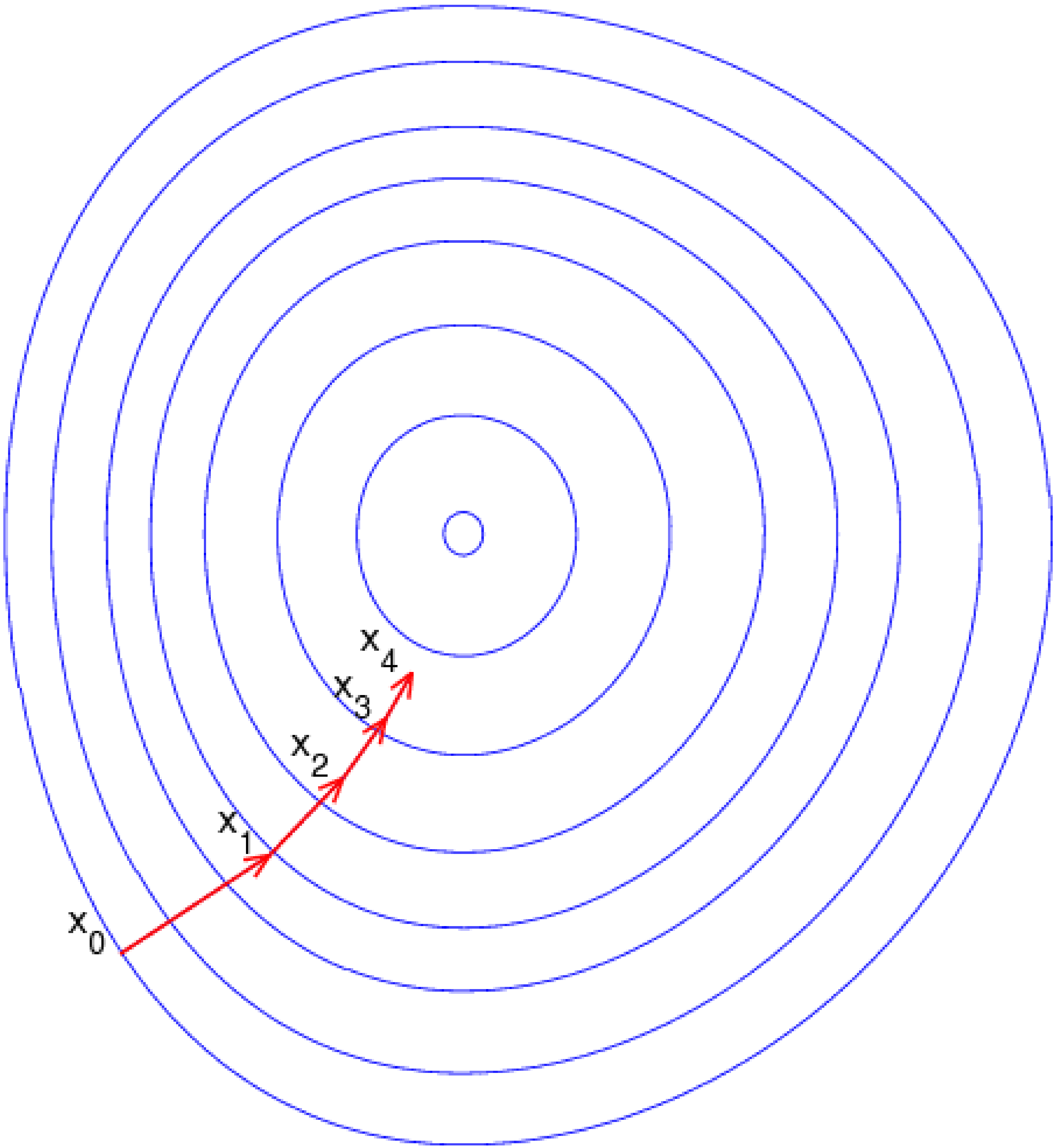
References

- Pablo Pedregal. Introduction to Optimization. Springer New York. 2004.
  (Chapter 5 deals with variational problems and the Euler-Lagrange equations.)
- I.M. Gelfand, S.V. Fomin. Calculus of Variations. Dover New York. 2000.
  (Section 4 gives an introduction to the Euler-Lagrange equations. Sections 16 ff. go into detail.)
- Euler-Lagrange equations (Wikipedia).
  `http://en.wikipedia.org/wiki/Euler-Lagrange_equation`
- Lecture "Differential Equations in Image Processing and Computer Vision". SS 2008.
  (Lecture 10 gives an introduction to Euler-Lagrange equations.)
- Lecture "Numerical Algorithms for Visual Computing III". SS 2009.
  (Covers all the material presented here in a more detailed way.)
- L. Bar, G. Sapiro. Generalized Newton-Type Methods for Energy Formulations in Image Processing. SIAM Journal on Imaging Sciences, 2009.
  `http://www.ece.umn.edu/users/barxx002/papers/Siam09.pdf`

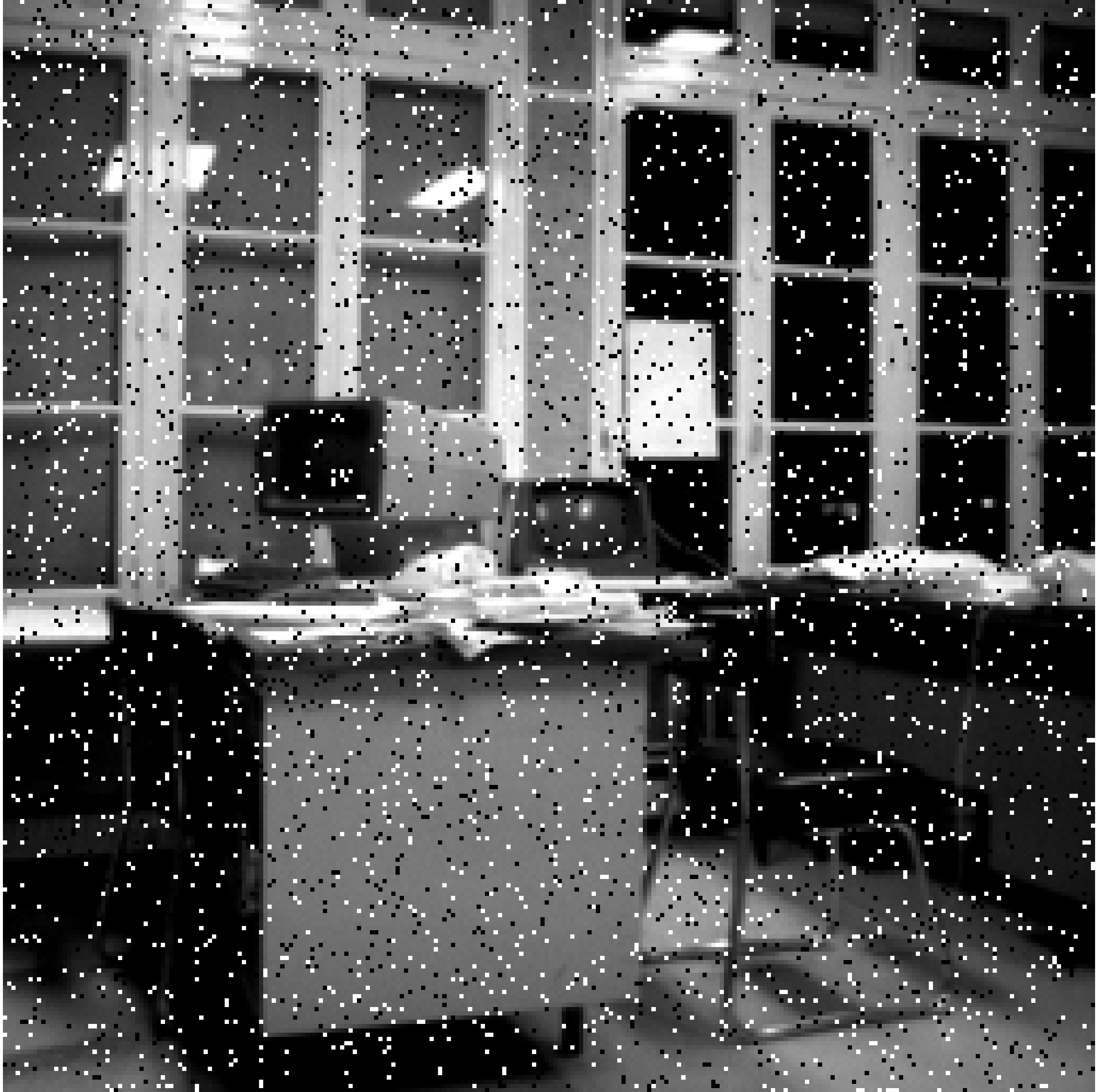
Thank you for your attention!

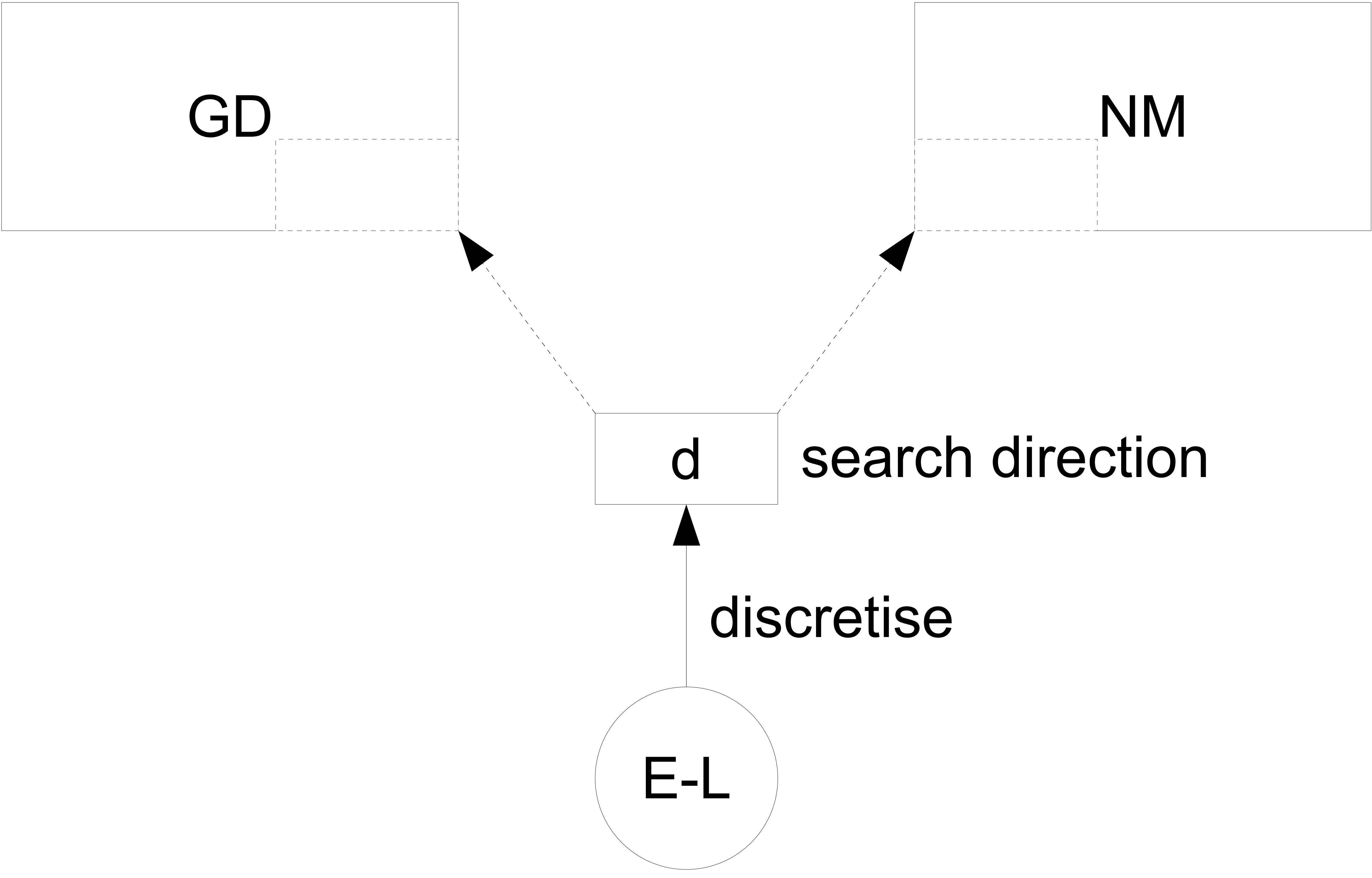

Euler-Lagrange Formalism
Variations of an Energy Functional
Iterative Algorithms
Connection to Euler-Lagrange Formalism





x0
x1
x2
x3
x4




GD
NM
d
search direction
discretise
E-L