Geometric Modeling

Summer Semester 2010

Mathematical Tools (3)

Recap: Numerics

UNIVERSITÄT DES SAARLANDES

CLUSTER OF EXCELLENCE

max planck institut informatik

Today...

Topics:

- Mathematical Background
  - Linear algebra
  - Analysis & differential geometry
  - Numerical techniques ←

Overview

Numerics & Applied Analysis:

- Cancellation Effects
- Multi-dimensional optimization
- Non-Linear optimization

Recap: Numerics

Floating Point Numbers & Cancellation

Floating Point Numbers

Floating Point Numbers

- Usually, computations are in floating point representation
- Means: Fixed number of digits for the number, exponent and sign:

$$\text{number} = \pm\, 0.\text{XXXXXXX} \cdot 10^{\text{XXX}}$$

- Criterion: *relative error w.r.t to the result*
- Operations:
  - *Multiplication*, *Division*: no problem
  - *Addition*: no problem (large error w.r.t. the input possible)
  - *Subtraction*: might go wrong; relative precision with respect to the result might be arbitrarily bad

Example

Example:

- Computing

$$\begin{array}{r} 0.12345????\\ -0.12344????\\ \hline =\ 0.00001???? \end{array}$$

- Lost 4 digits of precision (relative error in the result)

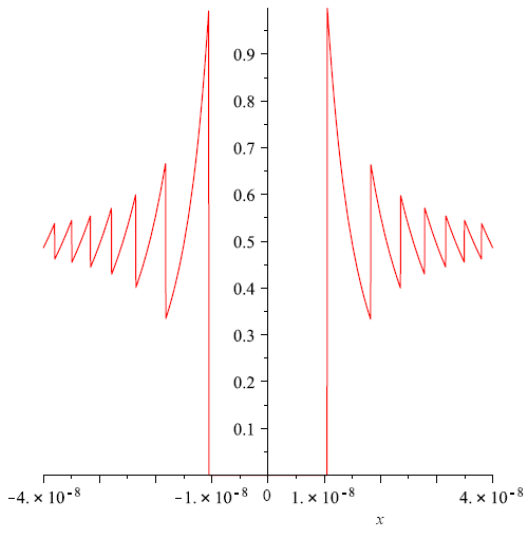
Conclusion:

- Subtracting numbers of similar magnitude is problematic
- Unavoidable flaw in floating point numbers
- The phenomenon is called *cancellation*.

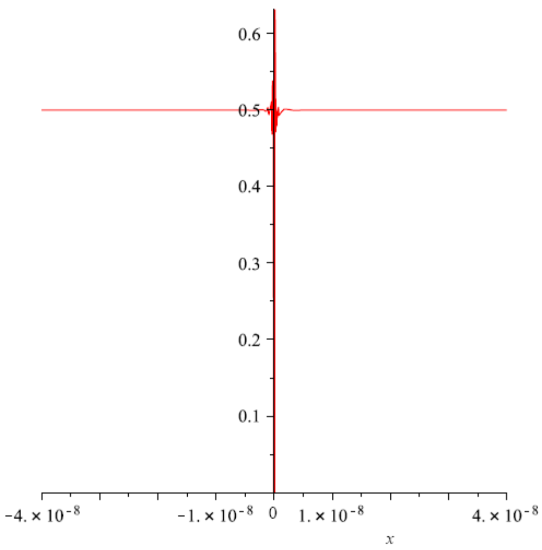
Cancellation example

$$f(x) = \frac{1 - \cos x}{x^2}$$

$Digits := 5$

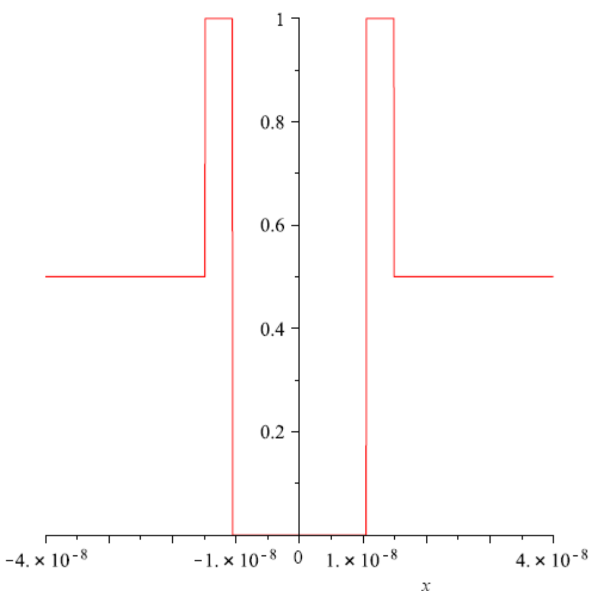


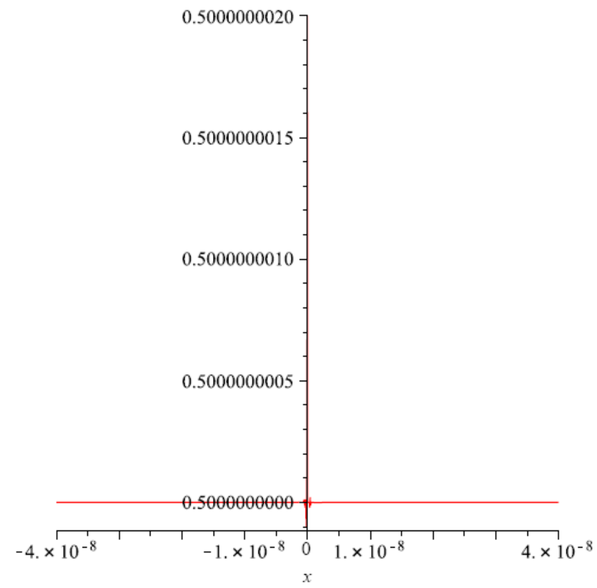
$Digits := 20$



Cancellation example (cont.)

$$f(x) = \frac{1}{x^2} - \frac{\cos(x)}{x^2}$$

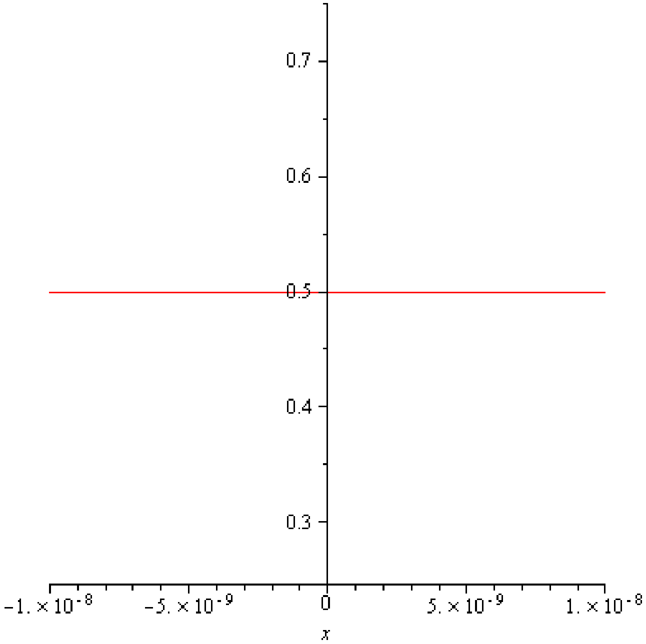




Solution

$$f(x) = \frac{1 - \cos x}{x^2}$$

Taylor: $\frac{1}{2} - \frac{1}{24}x^2 + \frac{1}{720}x^4 - \frac{1}{40320}x^6 + \frac{1}{3628800}x^8 + O(x^9)$



Numerical Problems

Two different phenomena:

- Cancellation: Representation of numbers as truncated floating point sequence cannot perform the operation with adequate precision
  - Problem of numerical resolution
  - Typical solution: Different numerical algorithm
- Ill posed problems: Solving the problem exactly amplifies noise in the input (scaling by large numbers necessary)
  - Typical solution: Do not solve exactly; ignore ill-conditioned subspaces, use additional assumptions
  - Problem independent of numerical representation

Recap: Analysis/Numerics

Multi-Dimensional Optimization

Optimization Problems

Optimization Problem:

- Given a C^1 function $f: \mathbb{R}^n \to \mathbb{R}$ (general heightfield)
- We are looking for a local extremum (minimum / maximum) of this function

Theorem:

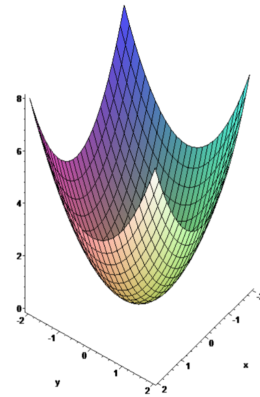
- $\mathbf{x}$ is a local extremum $\Rightarrow \nabla f(\mathbf{x}) = \mathbf{0}$

Sketch of a proof: If $\nabla f(\mathbf{x}) \neq 0$, we can walk a small step in gradient direction to improve the score further (in case of a maximum, minimum similar).

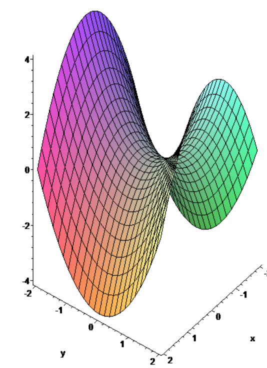
Critical Points

Critical points:

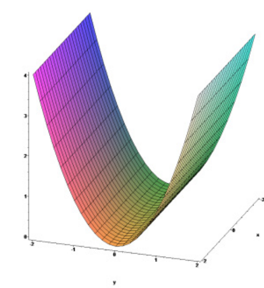
- $\nabla f(\mathbf{x}) = \mathbf{0}$ does not guarantee an extremum (saddle points)
- Points with $\nabla f(\mathbf{x}) = \mathbf{0}$ are called *critical points*.
- Final decision via *Hessian matrix*:
  - All eigenvalues > 0: local minimum
  - All eigenvalues < 0: local maximum
  - Mixed eigenvalues: saddle point
  - Some zero eigenvalues: critical line



$\lambda_i > 0$



$\lambda_0 > 0, \lambda_1 < 0$



$\lambda_0 = 0, \lambda_1 > 0$

Quadratic Optimization

Quadratic Case:

- $f: \mathbb{R}^n \to \mathbb{R}$
- Objective function: $f(\mathbf{x}) = \mathbf{x}^T\mathbf{A}\mathbf{x} + \mathbf{b}^T\mathbf{x} + c$
  - symmetric $n \times n$ matrix $\mathbf{A}$
  - n-dim. vector $\mathbf{b}$
  - constant c
- Gradient: $\nabla f(\mathbf{x}) = 2\mathbf{A}\mathbf{x} + \mathbf{b}$
- Critical points: solution to $2\mathbf{A}\mathbf{x} = -\mathbf{b}$
- Solution: Solve system of linear equations

Example

Gradient computation example:

$$[x, y]\begin{pmatrix} a \\ b \end{pmatrix} = ax + by \rightarrow \nabla = \begin{pmatrix} a \\ b \end{pmatrix}$$

$$[x, y]\begin{pmatrix} a & b \\ b & c \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix} = ax^2 + 2bxy + cy^2$$

$$\left.\begin{matrix} \partial_x \rightarrow 2ax + 2by \\ \partial_y \rightarrow 2bx + 2cy \end{matrix}\right\} \nabla \rightarrow 2\mathbf{A}\begin{pmatrix} x \\ y \end{pmatrix}$$

Global Extrema of Quadratic Funcs.

Three cases:

- Eigenvalues of **A** ≥ 0: critical points are *global minima*
- Eigenvalues of **A** ≤ 0: critical points are *global maxima*
- Mixed eigenvalues: no global minimum/maximum exists (minimum and maximum at infinity)

Structure:

- Critical points form an affine subspace of $\mathbb{R}^n$.
- I.e.: Point, line, plane...

Recap: Numerics

Non-Linear Optimization

Non-Quadratic Optimization

Optimization Problems:

- General formulation: Find the global minimum of a function $E: \mathbb{R}^n \supseteq \Omega \to \mathbb{R}$.
  (E for "energy", analogous to physics problems)
- Occur frequently in geometric modeling and geometry processing.
- We saw earlier: Quadratic problems can be solved exactly by a linear system of equations.
- What to do if E is non-quadratic?

$\Rightarrow$ **General non-linear optimization techniques**

Gradient Descent

Gradient Descent:

- Gradient ∇E points into direction of steepest ascent.
- Walking a small step in direction $-\nabla E$ will decrease the energy.
- When $\nabla E = \mathbf{0}$, a critical point is found.

Properties:

- For sufficiently small steps, this algorithm is guaranteed to converge
- Generally slow convergence
- Does not work in practice for ill-conditioned problems

Newton Optimization

Newton Optimization

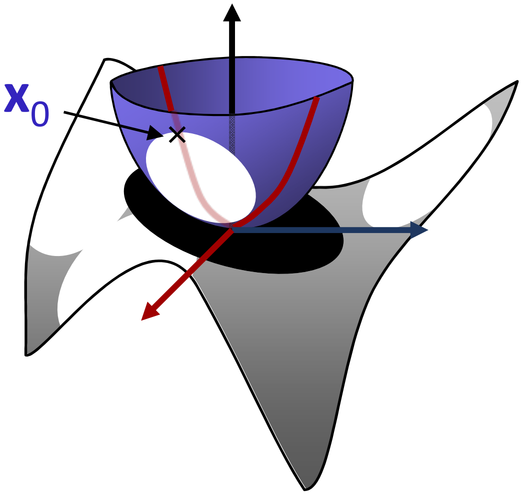
- Basic idea: Local quadratic approximation of E:

$$E(\mathbf{x}) \approx E(\mathbf{x}_0) + \nabla E(\mathbf{x}_0) \cdot (\mathbf{x} - \mathbf{x}_0) + \frac{1}{2}(\mathbf{x} - \mathbf{x}_0)^{\mathrm{T}} \cdot H_E(\mathbf{x}_0) \cdot (\mathbf{x} - \mathbf{x}_0)$$

- Solve for vertex (critical point) of the fitted parabola
- Iterate until a minimum is found ($\nabla E = 0$)

Properties:

- Typically much faster convergence, more stable
- No convergence guarantee



Newton Optimization - Divergence

Problem:

- If the Hessian matrix has negative eigenvalues, steps might point uphill (not towards the minimum)
- (Near-) zero eigenvalues make problem ill-conditioned.
- Simple solution: Add $\lambda \mathbf{I}$ to the Hessian for a small λ.
- Sum of two quadrics: $\lambda \mathbf{I}$ keeps solution at $\mathbf{x}_0$.
- This is an example of regularization

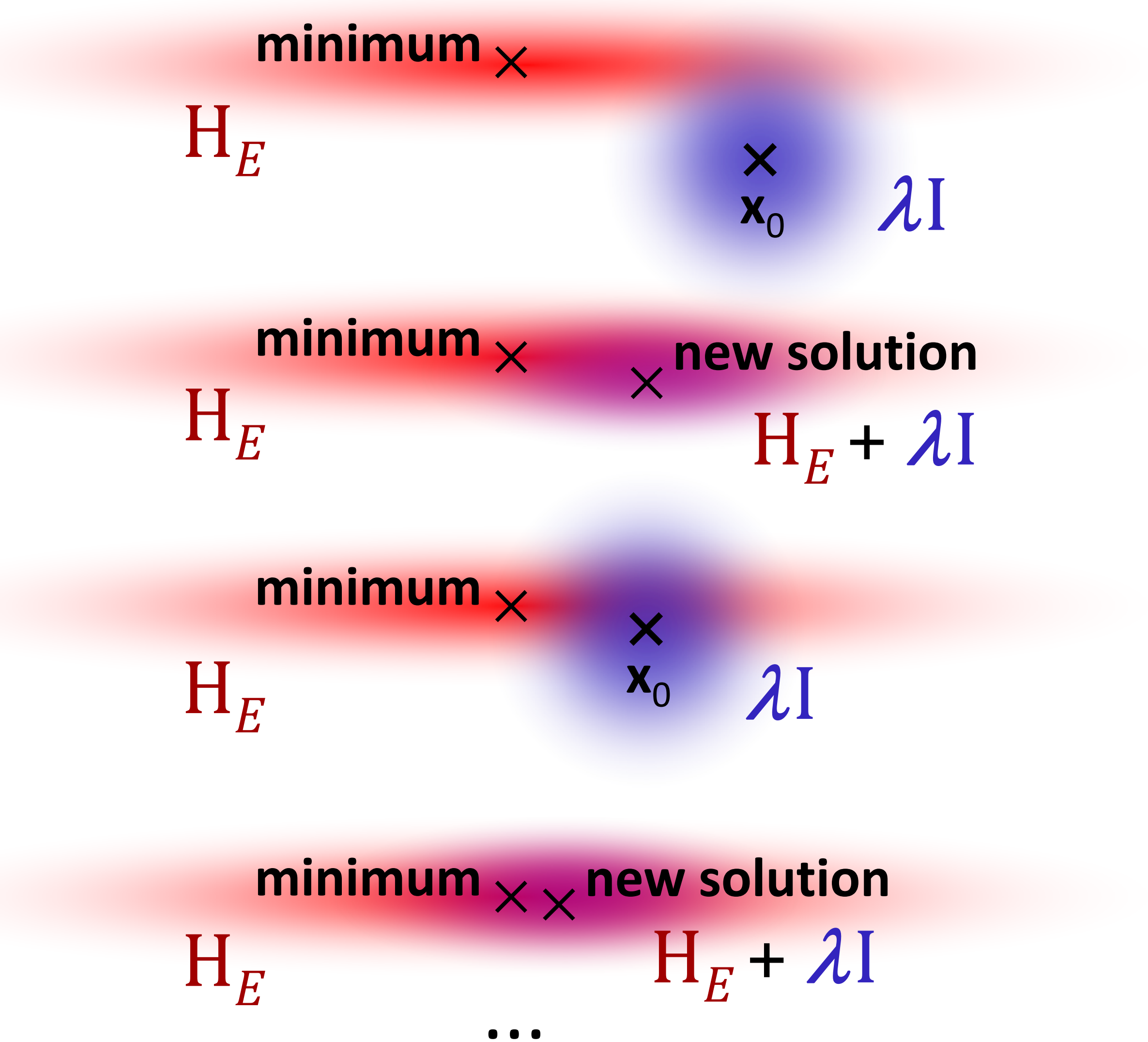
Handling Indefinite Situations

Initial state:

First Iteration:

New state:

Second Iteration:



…

Further Algorithms

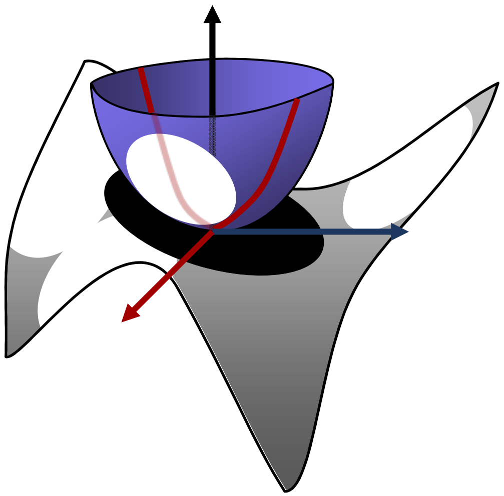
Gradient descent line search:

- Optimize step size for gradient descent
  - Fit 1D parabola to E in gradient direction
  - Perform 1D Newton search
  - If E does not decrease at the new position:
    - Try to half step width (say up to 10-20 times).
    - If this still does not decrease E, stop and output local minimum.

Further Algorithms

Line search for Newton-optimization:

- Following the quadratic fit might overshoot
- Line search:
  - Test value of E at new position
  - Half step width until error decreases
    (say 10-20 iterations)
  - Switch to gradient descent, if this does not work



Convex Problems

General Classification:

- Non-linear optimization problems can be hard to solve.
- What is definitely "easy"?

Convex Problems:

- *Convex functions* on a *convex domain* can be optimized "easily" using a generic algorithm.
- Other problems *might* be hard to solve.

Convex Problems

Convex Function:

- A C^2 function E is convex, if $H_E > 0$ (all eigenvalues of the Hessian are strictly positive everywhere)
- A set Ω is convex if every line connecting two points from Ω is also contained in Ω.
- A convex function has at most one local minimum

Problem Properties:

- Assume a global minimum exists
- Will be the only local minimum
- Can be reached on a straight line from any point in Ω

Convex Problems

Generic Optimization Algorithm (Sketch):

- Gradient descent
- Start at any point $\mathbf{p} \in \Omega$
- Perform gradient descent in "small enough" steps
- In case of hitting the domain boundary, project on boundary surface (follow the wall)
- When the gradient becomes zero, the minimum is found

There are more efficient algorithms...