Introduction to combinatorial optimization

Cours - 4BIM

Carole FRINDEL[1]

[1]Laboratoire CREATIS
Bâtiment L. De Vinci, Campus INSA
carole.frindel@creatis.insa-lyon.fr

September 2019

Outline

Outline

Definitions

- *Optimum* : State, degree of development of a system judged the most favorable according to given criteria
- If the system to be optimized can be modeled (put in the form of equations), the optimization tools can be used
- The optimization process includes several identification steps
  - the decision variables (parameters)
  - the influence of variables on the system (model)
  - the cost function
  - the constraints

Definitions

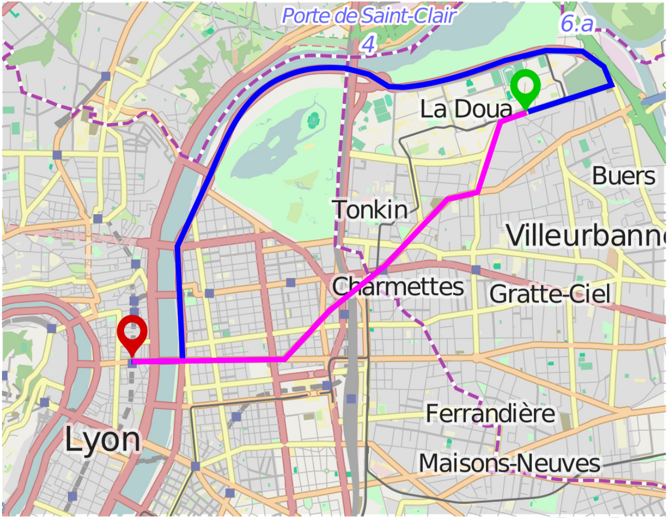
- *Optimum* : State, degree of development of a system judged the most favorable according to given criteria
- If the system to be optimized can be modeled (put in the form of equations), the optimization tools can be used
- The optimization process includes several identification steps
  - the decision variables (parameters)
  - the influence of variables on the system (model)
  - the cost function
  - the constraints

Definitions

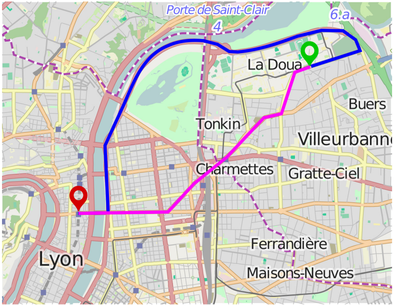
- *Optimum* : State, degree of development of a system judged the most favorable according to given criteria
- If the system to be optimized can be modeled (put in the form of equations), the optimization tools can be used
- The optimization process includes several identification steps
  - the decision variables (parameters)
  - the influence of variables on the system (model)
  - the cost function
  - the constraints

Trip calculator

Trip calculator



Trip calculator



- What are the decision parameters?
- The possible cost functions?
- The constraints ?

Decision variables

- System components on which it is possible to act to improve its state
- Generally put in the form of a vector noted $x = (x_1, x_2, \ldots, x_n)^T$

The objective/cost function

- Function that evaluates the state of the system for a given set of parameters
- Generally noted $f(x) : \mathbb{R}^n \rightarrow \mathbb{R}$
- **Parameters are optimal when f is minimal**

The constraints

- Constraints specifying the values that variables can take
- They define the domain of definition of the cost function

Outline

Some examples

Applications

- Autopilot
- MRI sequence
- Finance - Stock Market Investments

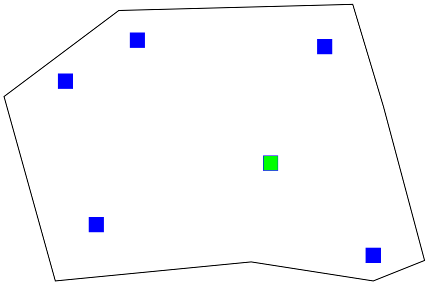
Model identification

- Parameters ?
- Cost function ?
- Constraints ?

Some examples

- The *commercial traveller* is a typical problem of complex optimization
- He must visit N given cities passing through each city exactly once
- In which order should he chain cities to minimize the distance traveled?

The commercial traveller



Determining the optimal path

- Number of possible permutations : $\frac{(N-1)!}{2}$

N	Permutations	Calculation time
5	12	12 μs
10	181 440	0.18 s
15	$43x10^9$	12 h
20	$60e^{15}$	1928 ans

- Primordial to find methods (algorithms) that converge to the optimal solution without evaluating all combinations

Mathematical modeling

- Let the matrix D_{ij} of size $N \times N$ representing the distance between cities i and j
- Find the permutation vector $p \in \mathbb{N}^N$ which minimizes the following C cost function:

$$
\begin{aligned}
C &: \mathbb{N}^N \rightarrow \mathbb{N} \\
C &: p \mapsto C(p)
\end{aligned}
$$

$$C(p) = \sum_{i=1}^{i=N-1} D_{p(i)p(i+1)}$$

Example for $N = 6$

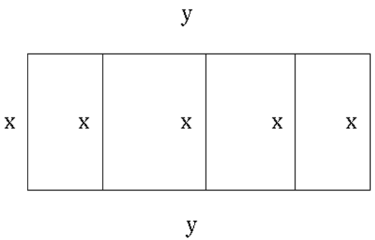
- Take $p = [6, 2, 4, 1, 5, 3]$
- $C(p) = D_{6,2} + D_{2,4} + D_{4,1} + D_{1,5} + D_{5,3}$
- The sequence of cities (p) is optimal if $C(p)$ is minimal ($\forall p$ in the eligible p space)

Exercises

- **Problem 1**: We want to build a rectangular enclosure with three parallel walls using 500m of fence. What are the dimensions that will maximize the total area of the enclosure?
- **Problem 2**: An open rectangular box with a square base must be made from $48cm^2$ of material. What are the dimensions that will maximize the total volume of the box?

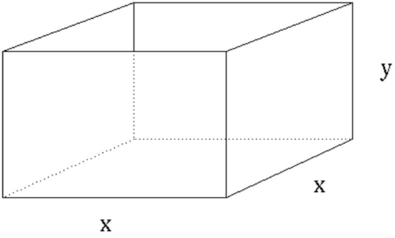
Model the problem as a function to minimize or maximize.

Problem 1



- The amount of fence needed: $500 = 5x + 2y$
- Hence $y = 250 - (5/2)x$
- Maximize total area: $A = xy = 250x - (5/2)x^2$

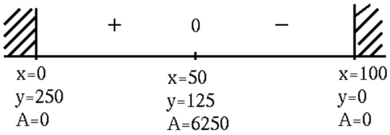
Problem 2



- The total area of the box: $48 = x^2 + 4(xy)$
- Hence y = $\frac{12}{x} - \frac{x}{4}$
- Maximize the total volume of the box: $V = xxy = 12x - \frac{1}{4}x^3$

Problem 1

- Maximize total area: $f(x) = 250x - (5/2)x^2$
- Derivative: $f'(x) = 250 - 5x = 5(50 - x)$
- Hence $f'(x) = 0 \Leftrightarrow x = 50$ and $y = 125$
- Note that since there are 5 lengths of x in this construction and 500m of closing, it follows that $0 \leq x \leq 100$



Outline

Modeling, in short ...

- Good modeling is fundamental to optimize a system (variables, cost function, constraints)
- In this course, we will study optimization problems
  - without constraints
  - with differentiable cost functions
- **Optimize a system means finding the state that minimizes the cost function**

Modeling, in short ...

- Good modeling is fundamental to optimize a system (variables, cost function, constraints)
- In this course, we will study optimization problems
  - without constraints
  - with differentiable cost functions
- **Optimize a system means finding the state that minimizes the cost function**

Minimization of f

- Minimizing a cost function may not be trivial!
- Some (non-convex) functions have multiple minima
- Potentially undifferentiable
- Generally impossible to calculate the cost on the whole f definition set

What is good optimization?

- Optimization methods converge to the optimal parameter set without calculating f over its entire definition domain
- A good optimization:
  - Converge to the global minimum of f
  - Converge quickly
  - Robust (noise, model errors)

Outline

n-D tools

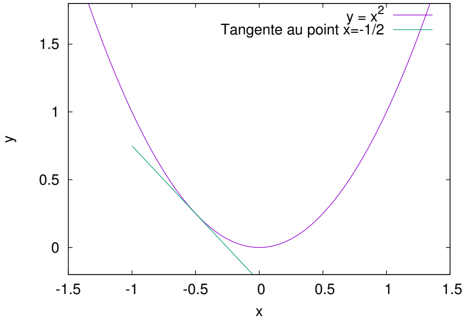
Gradient

Let f be a continuous function. The function denoted $\nabla f(x)$: $\mathbb{R}^n \to \mathbb{R}^n$ is called the gradient of f and is defined by:

$$\nabla f(x) = \left(\frac{\partial f(x)}{\partial x_1}, \ldots, \frac{\partial f(x)}{\partial x_n} \right)^T \tag{1}$$

n-D tools

- In 1D, the gradient is equivalent to the derivative
- The gradient value corresponds to the slope of the tangent



n-D tools

Directional derivative

Let f: $\mathbb{R}^n \to \mathbb{R}$ admitting a gradient $\nabla f(x)$, and $d \in \mathbb{R}^n$. The directional derivative is the dot product between the gradient of f and the direction d:

$$\nabla f(x)^T d \tag{2}$$

- As in 1D, the directional derivative gives indications on the slope of f in the direction d:
  - $\nabla f(x)^T d < 0 \Leftrightarrow$ f is decreasing in the direction d
  - $\nabla f(x)^T d > 0 \Leftrightarrow$ f is increasing in the direction d

n-D tools

Directional derivative

Let f: $\mathbb{R}^n \to \mathbb{R}$ admitting a gradient $\nabla f(x)$, and $d \in \mathbb{R}^n$. The directional derivative is the dot product between the gradient of f and the direction d:

$$\nabla f(x)^T d \tag{2}$$

- As in 1D, the directional derivative gives indications on the slope of f in the direction d:
  - $\nabla f(x)^T d < 0 \Leftrightarrow$ f is decreasing in the direction d
  - $\nabla f(x)^T d > 0 \Leftrightarrow$ f is increasing in the direction d

Direction of descent

Direction of descent

Let f: $\mathbb{R}^n \to \mathbb{R}$ a differentiable function. Let x, $d \in \mathbb{R}^n$. The d direction is a direction of descent in x if:

$$\nabla f(x)^T d < 0 \tag{3}$$

Direction of stronger descent (CS inequality)

Let $f : \mathbb{R}^n \to \mathbb{R}$ a differentiable function. Let $x \in \mathbb{R}^n$ and $d^* = -\nabla f(x)$. Then, $\forall d \in \mathbb{R}^n$ such that $\|d\| = \|\nabla f(x)\|$, we have:

$$d^T \nabla f(x) \geq d^{*T} \nabla f(x) = -\nabla f(x)^T \nabla f(x) \tag{4}$$

Direction of descent

Direction of descent

Let f: $\mathbb{R}^n \to \mathbb{R}$ a differentiable function. Let x, $d \in \mathbb{R}^n$. The d direction is a direction of descent in x if:

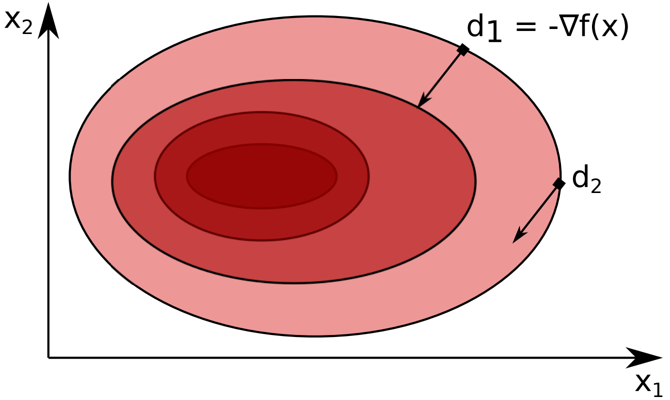
$$\nabla f(x)^T d < 0 \tag{3}$$

Direction of stronger descent (CS inequality)

Let $f : \mathbb{R}^n \to \mathbb{R}$ a differentiable function. Let $x \in \mathbb{R}^n$ and $d^* = -\nabla f(x)$. Then, $\forall d \in \mathbb{R}^n$ such that $\|d\| = \|\nabla f(x)\|$, we have:

$$d^T \nabla f(x) \geq d^{*T} \nabla f(x) = -\nabla f(x)^T \nabla f(x) \tag{4}$$

Illustration of 2 directions of descent



Outline

Hessian

Hessian matrix

Let $f : \mathbb{R}^n \to \mathbb{R}$ a function that is twice differentiable. The function denoted $\nabla^2 f(x) : \mathbb{R}^n \to \mathbb{R}^{n\times n}$ is called Hessian matrix of f, is always symmetric, and defined by:

$$\nabla^2 f(x) = \begin{pmatrix} \frac{\partial^2 f(x)}{\partial^2 x_1^2} & \frac{\partial^2 f(x)}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f(x)}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f(x)}{\partial x_2 \partial x_1} & \frac{\partial^2 f(x)}{\partial x_2^2} & \cdots & \frac{\partial^2 f(x)}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f(x)}{\partial x_n \partial x_1} & \frac{\partial^2 f(x)}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f(x)}{\partial x_n^2} \end{pmatrix}$$

Convexity

- The Hessian gives information on the convexity of f
- If $\nabla^2 f(x)$ is positive definite (eigenvalues > 0) then f is strictly convex
- In this case, f has a unique minimum
- Example: $f(x, y) = x^2 + y^2$ is it convex?
- And $f(x, y) = x^2 - y^2$?

Convexity

- The function $f(x, y) = x^2 + y^2$ has for eigenvalues 2 and 2:
$$\nabla^2 f(x) = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}$$
- It is convex $\rightarrow$ single minimum
- The function $f(x, y) = x^2 - y^2$ has for eigenvalues -2 and 2:
$$\nabla^2 f(x) = \begin{pmatrix} 2 & 0 \\ 0 & -2 \end{pmatrix}$$
- It is not convex $\rightarrow$ no single minimum

Illustration of the function $f(x, y) = x^2 + y^2$

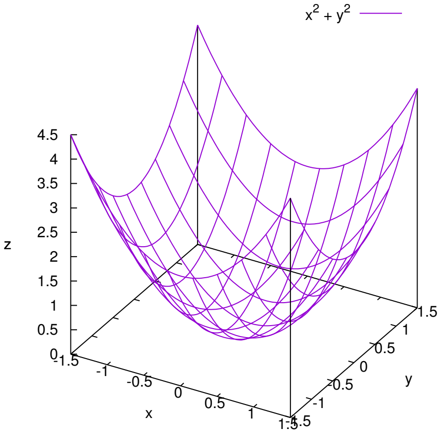
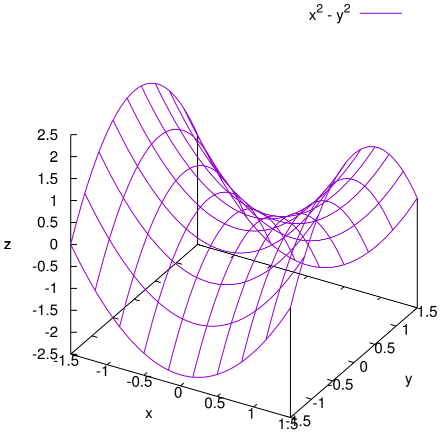


Illustration of a saddle point for $f(x, y) = x^2 - y^2$



Curvature

Curvature

Let $f : \mathbb{R}^n \to \mathbb{R}$ twice differentiable. Let x and $d \in \mathbb{R}^n$. The quantity :

$$\frac{d^T \nabla^2 f(x) d}{d^T d}$$

represents the curvature of the function f in x in the direction d.

Summary

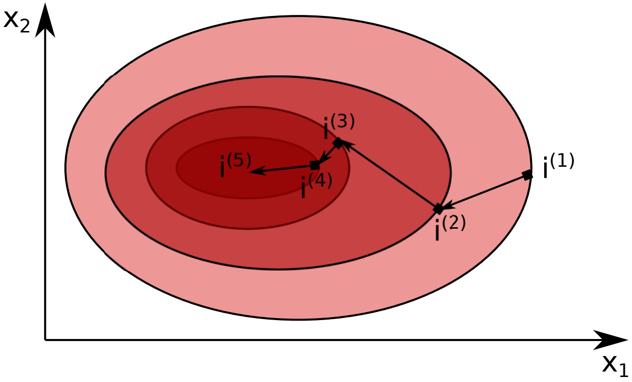
- 1st-order differentiability (gradient) gives information on the direction of descent
- 2nd order differentiability (Hessian) gives information on curvature and convexity
- These notions are essential in the search for the minimum of the cost function of an optimization problem

Outline

Overview

- A numerical optimization method seeks to minimize the cost function by following a certain convergence scheme
- Each optimization problem is unique $\rightarrow$ no ideal method
- General iterative principle:
  - We start from a point x_0
  - We look for the points $x_1, x_2, \ldots, x_n$ such that $f(x_{k+1}) < f(x_k)$, $k = 1, \ldots, n$
  - Stop when convergence

Illustration of a sequence of iterations



Key points

Speed

- Calculation time of the descent direction at each iteration
- Total number of iterations

Convergence threshold

- We rarely reach the exact minimum $\rightarrow$ definition of a convergence tolerance based on:
  - max number of iterations
  - the gradient norm
  - the convergence step norm

Stopping criteria

- Max number of iterations
  - + Simple, avoids calculation time too long
  - – Not connected to the real convergence of the algo
- Gradient norm
  - + Quite reliable
  - – Can fail with low decreasing functions
- Convergence step norm
  - + Quite reliable
  - – Depends on the step calculation method
- In practice: combination of several stopping criteria

Outline

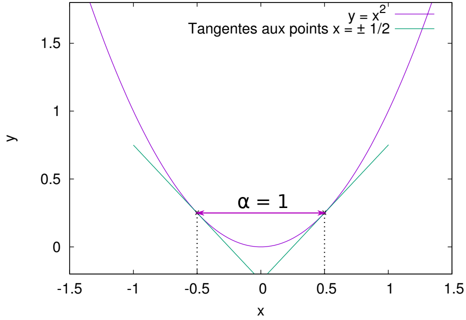
1. Introduction
   - Optimization concept
   - Applications examples
   - In short...
2. Differential tools
   - First order
   - Second order
3. Downhill direction optimization methods
   - Principle
   - First order methods
   - Second order methods

Method of the steepest slope

- Consists of choosing d as the direction of the steepest slope (orthogonal to the level lines)
- Let $d^\star = -\nabla f(x)$ (cf Eq. 4)
- We obtain the following iteration scheme: $x_{k+1} = x_k - \alpha_k \nabla f(x)$
- The step α_k is to be determined

Choice of convergence step

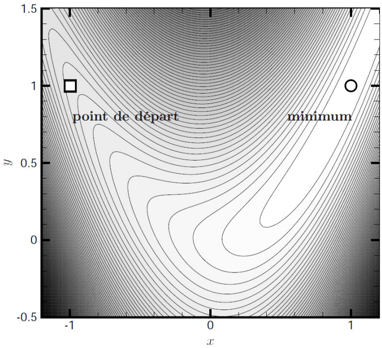
- The choice of convergence step α_k is crucial
- The simplest: $\alpha_k = \alpha =$cst $\rightarrow$ risky choice !

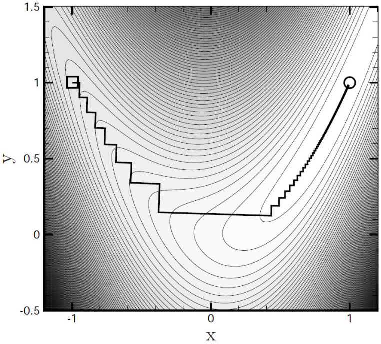


Choice of convergence step

- Several optimization methods exist to choose it
- Principle of the "line search" method:
  - At iteration k
  - Set a given descent direction d_k
  - Choose $\alpha > 0$ to minimize $h(\alpha) = f(x_k + \alpha d_k)$
  - Calculation of $h'(\alpha) = 0 \rightarrow$ new optimization problem !
- Compromise needed between optimal step and computation time

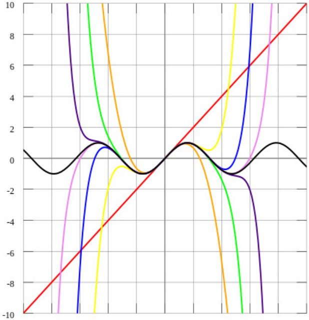
Method of the steepest slope with optimal step





Taylor Series

- In analysis, the Taylor series of a function *f* (at point *a*) is an integer series constructed from *f* and its successive derivatives at point *a*
- The series results in a polynomial: the higher its **degree**, the better the approximation of f to point a



Analogy with the Taylor series

- The Taylor series of a function $f(x)$ infinitely differentiable at point a is expressed thanks to its successive derivatives:

$$f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!}(x-a)^n$$

- At order n=1, we get :

$$f(x) = f(a) + (x-a)f'(a) + O(x^2)$$

- A the iteration $x^{(k)}$, in the descent direction $d = x^{(k+1)} - x^{(k)}$, we find :

$$f(x^{(k+1)}) = f(x^{(k)} + d) = f(x^{(k)}) + d.f'(x^{(k)}) + O(x^2)$$

- Gradient descent tends to minimize **1-order approximation** of the cost function

Analogy with the Taylor series

- The Taylor series of a function $f(x)$ infinitely differentiable at point a is expressed thanks to its successive derivatives:
$$f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!}(x-a)^n$$
- At order n=1, we get :
$$f(x) = f(a) + (x-a)f'(a) + O(x^2)$$
- A the iteration $x^{(k)}$, in the descent direction $d = x^{(k+1)} - x^{(k)}$, we find :
$$f(x^{(k+1)}) = f(x^{(k)} + d) = f(x^{(k)}) + d.f'(x^{(k)}) + O(x^2)$$
- Gradient descent tends to minimize **1-order approximation** of the cost function

Summary on order 1

- Simple method to implement
- Minimize the cost function at each iteration
- Few calculations by iteration
- Not very effective for slow decreasing functions

Outline

Newton's method

- Use of the second derivative for calculating the d direction
- Taking curvature information into account
- It is based on a quadratic approximation of f in x

Quadratic model

Quadratic model of a function

Let $f : \mathbb{R}^n \to \mathbb{R}$ twice differentiable. The quadratic model of f at point $\hat{x}$ is a function $m_{\hat{x}} : \mathbb{R}^n \to \mathbb{R}$ defined by :

$$m_{\hat{x}}(x) = f(\hat{x}) + (x - \hat{x})^T \nabla f(\hat{x}) + \frac{1}{2}(x - \hat{x})^T \nabla^2 f(\hat{x})(x - \hat{x})$$

where ∇f is the gradient of f and $\nabla^2(f)$ the hessian matrix. By assuming $d = x - \hat{x}$, we obtain the equivalent formula:

$$m_{\hat{x}}(\hat{x} + d) = f(\hat{x}) + d^T \nabla f(\hat{x}) + \frac{1}{2} d^T \nabla^2 f(\hat{x}) d$$

Quadratic model

- The quadratic model is an approximation (second order) of the f function at $\hat{x}$ (Taylor series)
- Let's find the minimum of $m_{\hat{x}}$:

$$\min_{d \in \mathbb{R}^n} m_{\hat{x}}(\hat{x} + d) = f(\hat{x}) + d^T \nabla_x f(\hat{x}) + \frac{1}{2} d^T \nabla_x^2 f(\hat{x}) d$$

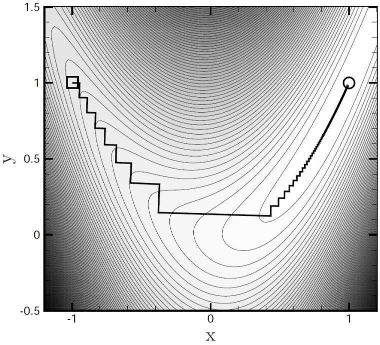
$$\Leftrightarrow \nabla_d m_{\hat{x}}(\hat{x} + d) = \nabla_x f(\hat{x}) + \nabla_x^2 f(\hat{x}) d = 0$$

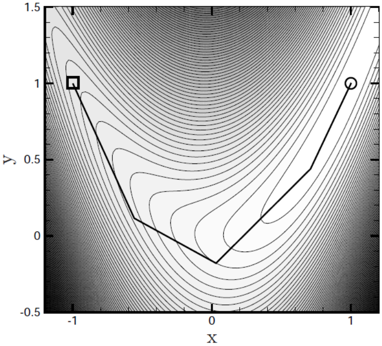
$$\Leftrightarrow d = -\frac{\nabla_x f(\hat{x})}{\nabla_x^2 f(\hat{x})}$$

Quadratic model: algorithm

- Sufficient condition of optimality: $\nabla^2 f(\hat{x})$ must be positive
- In this case, we minimize **in a single iteration** the quadratic model
- We can then define the following algorithm:
- Choose $k = 0$ and x_0. As long as not convergence:
  - Calculate the quadratic model at x_k : $m_{x_k}(x)$
  - Calculate $d_k = \min_d [m_{x_k}(x_k + d)] = -\frac{\nabla f(x_k)}{\nabla^2 f(x_k)}$
  - $x_{k+1} = x_k + d_k$
  - $k = k + 1$

Comparison of the two orders





Quadratic model: example

- Let the function $f(x) = -x^4 + 12x^3 - 47x^2 + 60x$
- Calculate $m_{x_k}(x)$ at $x_k = 3$
- $m_3(x) = f(3) + (x-3).f'(3) + \frac{(x-3)^2}{2}.f''(3)$
- Calculation of derivatives :

$$f'(x) = -4x^3 + 36x^2 - 94x + 60$$
$$f''(x) = -12x^2 + 78x - 94$$

- Hence $m_3(x) = 7x^2 - 48x + 81$

Quadratic model: example

- Let the function $f(x) = -x^4 + 12x^3 - 47x^2 + 60x$
- Calculate $m_{x_k}(x)$ at $x_k = 3$
- $m_3(x) = f(3) + (x-3).f'(3) + \frac{(x-3)^2}{2}.f''(3)$
- Calculation of derivatives :

$$f'(x) = -4x^3 + 36x^2 - 94x + 60$$
$$f''(x) = -12x^2 + 78x - 94$$

- Hence $m_3(x) = 7x^2 - 48x + 81$

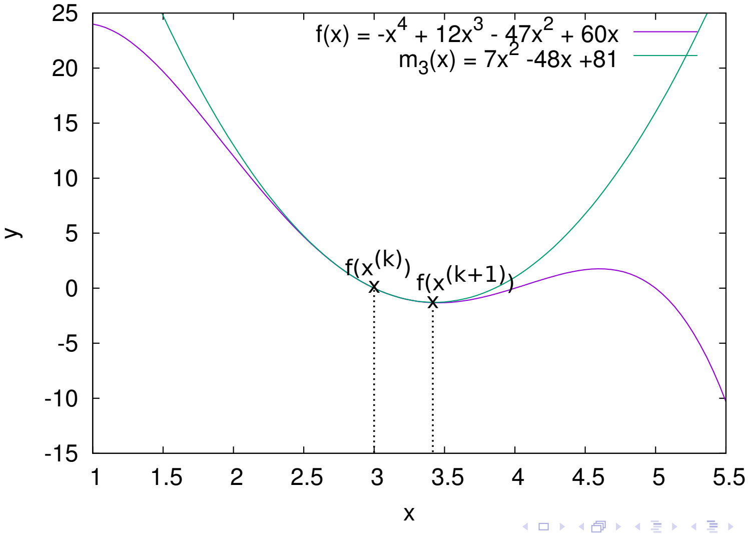
Quadratic model: example

- Let the function $f(x) = -x^4 + 12x^3 - 47x^2 + 60x$
- Calculate $m_{x_k}(x)$ at $x_k = 3$
- $m_3(x) = f(3) + (x-3).f'(3) + \frac{(x-3)^2}{2}.f''(3)$
- Calculation of derivatives :

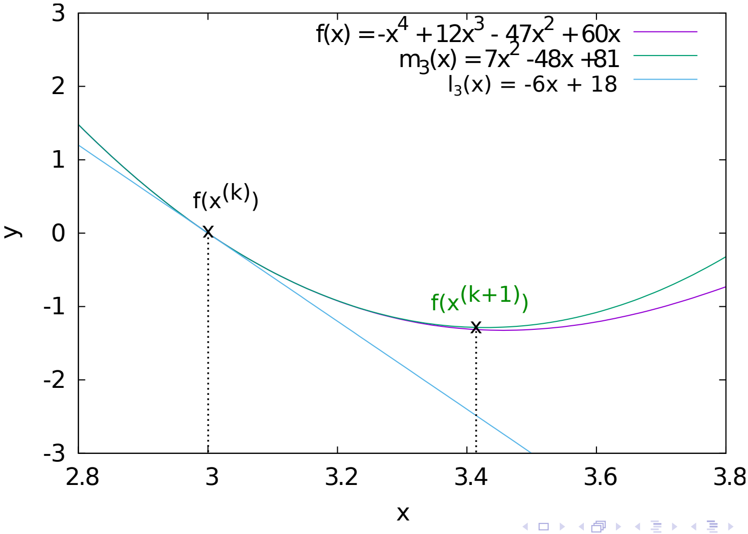
$$f'(x) = -4x^3 + 36x^2 - 94x + 60$$
$$f''(x) = -12x^2 + 78x - 94$$

- Hence $m_3(x) = 7x^2 - 48x + 81$

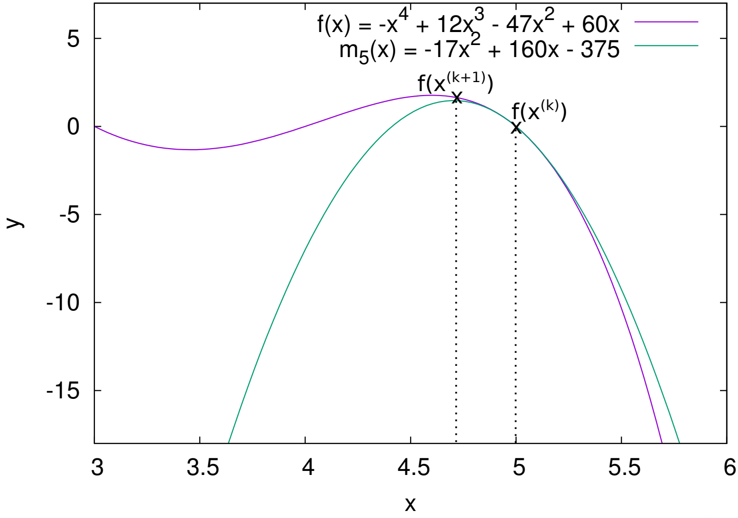
Quadratic model: example



Quadratic model: zoom



Quadratic model: critical case



Quadratic model: summary

- Advantages: :
  - Generally fewer iterations needed than the deepest slope method
  - Optimal for quadratic cost functions
- Limitations :
  - No distinctions between minimum, maxima and saddle point
  - Calculation of the second derivative at each iteration
  - The hessian must be set positive

Extensions of Newton's method

- Quasi-Newton methods
- They avoid the calculation (usually long) of the Hessian
- The following methods can be mentioned:
  - Davidon–Fletcher–Powell (DFP)
  - Symmetric rank-one (SR1)
  - Broyden–Fletcher–Goldfarb–Shanno (BFGS)

For further investigation...

- *Optimisation et analyse convexe*, Jean-Baptiste Hiriart-Urruty, 2012
- *A gentle introduction to optimization*, Guenin, B., Könemann, J., Tuncel, L., 2014
- *Conditioning of quasi-Newton methods for function minimization*, D. F. Shanno, Math. Comp. 24 (1970), pp. 647-656
- Librairie Python : *http://docs.scipy.org/doc/scipy/reference/tutorial/optimize.html*