Optimization and Simulation

Introduction and unconstrained optimization

Michel Bierlaire

Transport and Mobility Laboratory
School of Architecture, Civil and Environmental Engineering
Ecole Polytechnique Fédérale de Lausanne

Outline

Introduction

- Management of complex systems
  - Transportation systems
  - Environmental systems
  - Process systems
  - Structural systems

  en.wikipedia.org/wiki/List_of_types_of_systems_engineering
- The whole may be different from the sum of the parts
- Need for methods to deal with the complexity

- *To optimize*: to find the best configuration
- *To simulate*: to act like.

Outline

Optimization

The problem

$$\min_{x \in \mathbb{R}^n} f(x)$$

subject to

$$\begin{aligned} h(x) &= 0 \\ g(x) &\leq 0 \\ x &\in X \subseteq \mathbb{R}^n \end{aligned}$$

Optimization

Many variants

- Linear vs non linear
- Convex vs non convex
- Continuous vs discrete
- Differentiable vs non differentiable
- Deterministic vs stochastic

In this course

- Non linear
- Non convex
- Continuous
- Differentiable
- Deterministic

Optimization

Optimization without constraint

- Assumed to be known
- Quick review today

Optimization with constraints

- Four different algorithms will be investigated
- Each group will implement one of them and present it

Outline

Optimality conditions

$$\min_{x \in \mathbb{R}^n} f(x).$$

Necessary optimality conditions

- Let x^* be a local minimum of $f : \mathbb{R}^n \to \mathbb{R}$.
- (first order condition) If f is differentiable in an open neighborhood of x^*, then
$$\nabla f(x^*) = 0.$$
- (second order condition) If f is twice differentiable in an open neighborhood of x^*, then
$$\nabla^2 f(x^*) \geq 0,$$
meaning that $\nabla^2 f(x^*)$ is *positive semidefinite*.

Optimality conditions

Sufficient optimality conditions

- Let $f : \mathbb{R}^n \to \mathbb{R}$ be twice differentiable in an open set $V \subseteq \mathbb{R}^n$.
- Let $x^* \in V$ such that
  - (first order condition)
  $$\nabla f(x^*) = 0.$$
  - (second order condition)
  $$\nabla^2 f(x^*) > 0,$$
  meaning that $\nabla^2 f(x^*)$ is *positive definite*.
- Then x^* is a local minimum of f.

Optimality conditions

Sufficient conditions for global optimality

- Let $f : \mathbb{R}^n \to \mathbb{R}$ be a continuous function
- Let $x^* \in \mathbb{R}^n$ be a local minimum of f
- If f is convex, then x^* is a global minimum of f.
- If f is strictly convex, then x^* is the unique global minimum of f.

Outline

System of equations

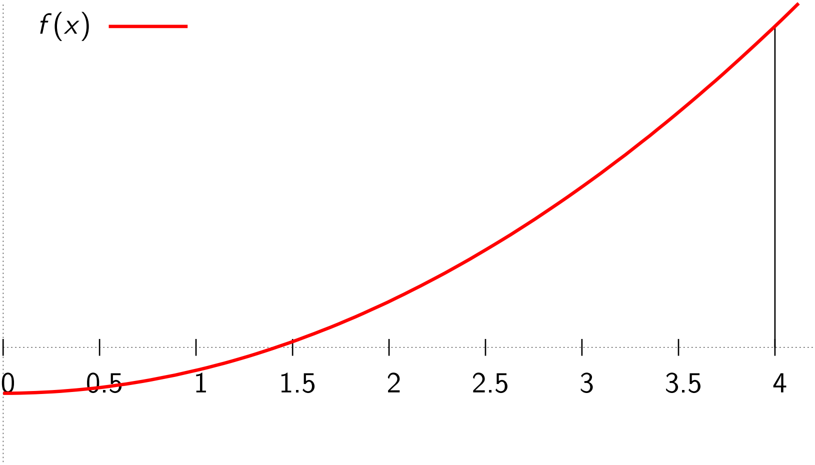
Motivation

- Optimality condition: $\nabla f(x) = 0$.
- In constrained optimization, it is also a system of equations.
- Algorithms to solve systems of equations play an important role.

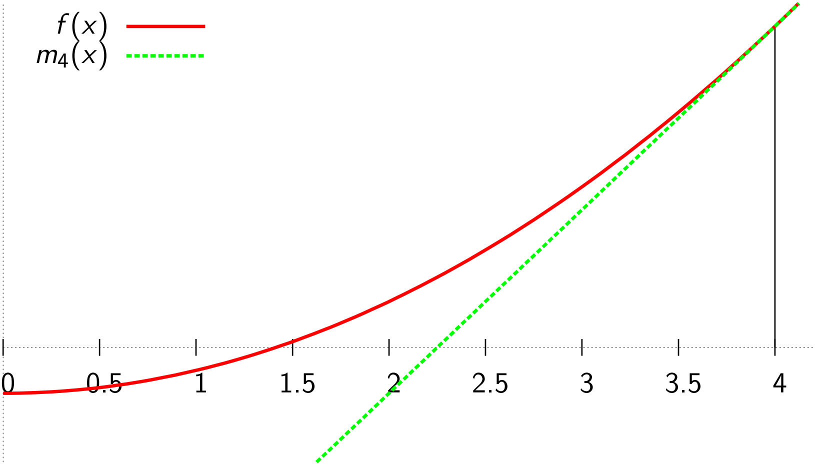
Newton's method

- Fast method.
- Work only under specific conditions.

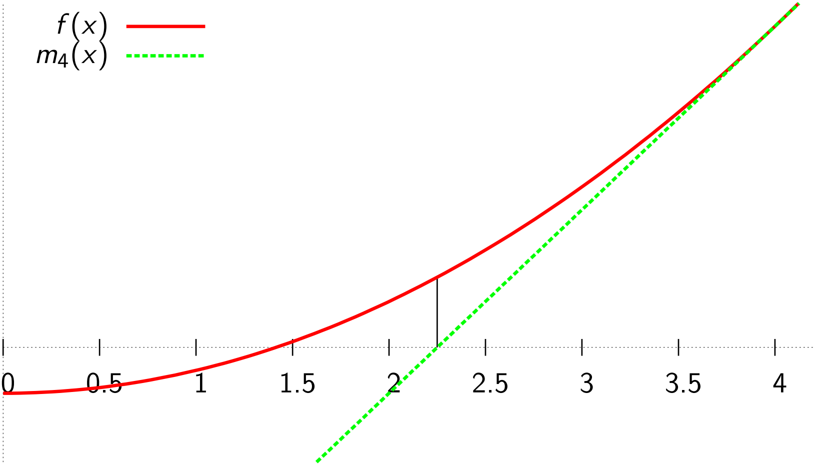
Newton's method



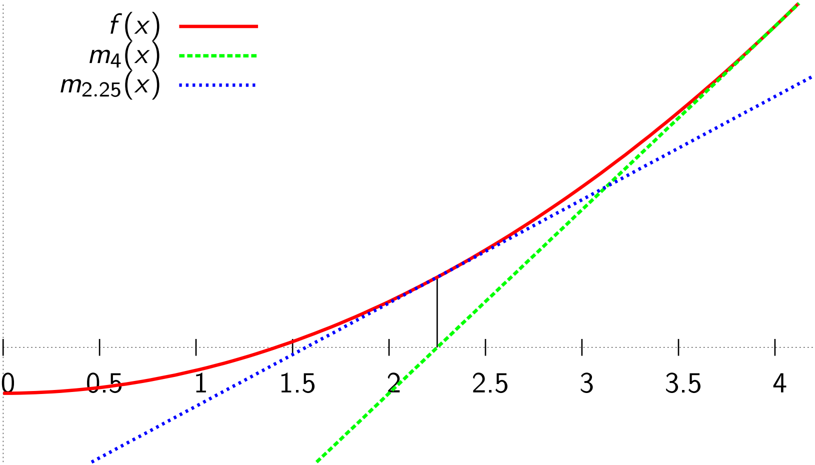
Newton's method



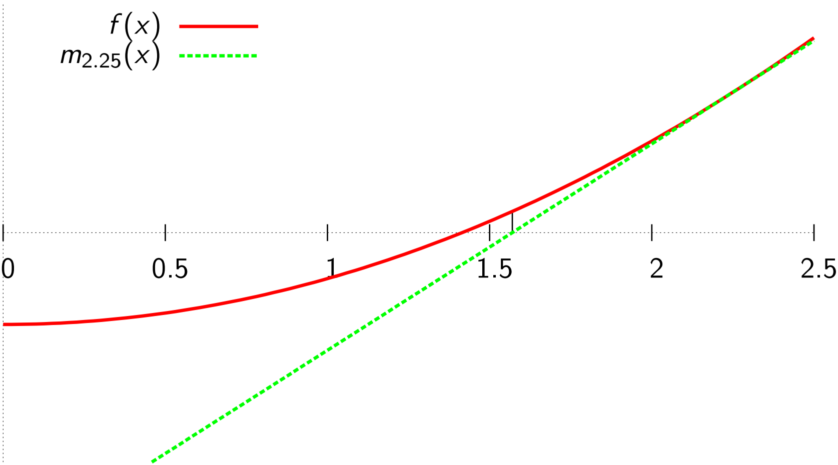
Newton's method



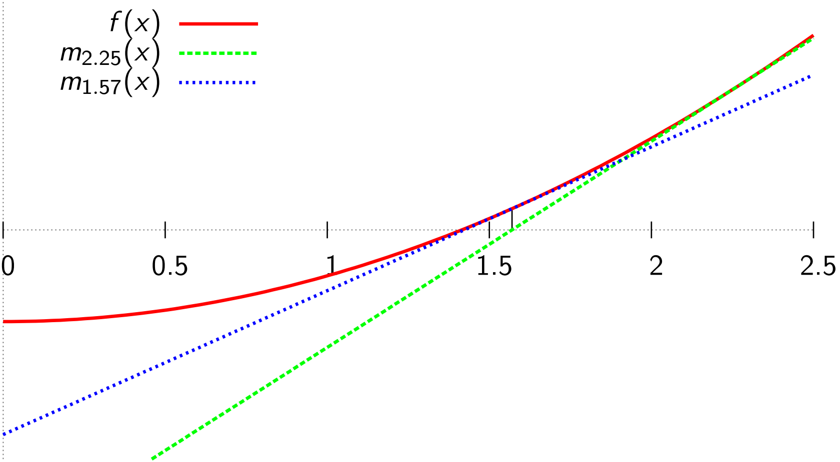
Newton's method



Newton's method



Newton's method



Solving systems of equations

The problem

Find x^* such $F(x^*) = 0$, where $F : \mathbb{R}^n \to \mathbb{R}^n$.

Newton's method

- Start at an arbitrary iterate $x_0 \in \mathbb{R}^n$
- At each iteration k, linearize F around x_k
- Find the root of the linear system and defines it as the next iterate

Key analytical tool

The gradient matrix, or the Jacobian matrix.

- For a function $F : \mathbb{R}^n \to \mathbb{R}^m$, the gradient and the Jacobian matrices are defined as follows.
- Note: for systems of equations, $n = m$.

Solving systems of equations

Gradient matrix

$$\nabla F(x) = \begin{pmatrix} | & & | \\ \nabla F_1(x) & \cdots & \nabla F_m(x) \\ | & & | \end{pmatrix}$$

$$= \begin{pmatrix} \frac{\partial F_1}{\partial x_1} & \frac{\partial F_2}{\partial x_1} & \cdots & \frac{\partial F_m}{\partial x_1} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial F_1}{\partial x_n} & \frac{\partial F_2}{\partial x_n} & \cdots & \frac{\partial F_m}{\partial x_n} \end{pmatrix}.$$

Solving systems of equations

Jacobian matrix

$$J(x) = \nabla F(x)^T = \begin{pmatrix} \text{———} & \nabla F_1(x)^T & \text{———} \\ & \vdots & \\ \text{———} & \nabla F_m(x)^T & \text{———} \end{pmatrix}.$$

Newton's method

Objective

Find (an approximation of) a solution of the systems of equations:

$$F(x) = 0.$$

Inputs

- The function $F : \mathbb{R}^n \to \mathbb{R}^n$;
- The Jacobian matrix: $J : \mathbb{R}^n \to \mathbb{R}^{n \times n}$;
- A first approximation of the solution: $x_0 \in \mathbb{R}^n$;
- The requested precision: $\varepsilon \in \mathbb{R}$, $\varepsilon > 0$.

Output

An approximation of the solution $x^* \in \mathbb{R}^n$.

Newton's method (ctd)

Initialization

$k = 0$.

Iterations

1. Compute d_{k+1} solution of
$$J(x_k)d_{k+1} = -F(x_k).$$
2. $x_{k+1} = x_k + d_{k+1}$.
3. $k = k + 1$.

Stopping criterion

If $\|F(x_k)\| \leq \varepsilon$, then $x^* = x_k$.

Convergence

Fast

- Quadratic convergence...
- ... under conditions.

Conditions

- Start close enough from the solution.
- Derivative stays away from 0.
- Function not too nonlinear (Lipschitz continuity of the Jacobian).

Secant method

Idea

Replace the derivative by a secant approximation.

Motivation

- does not use the derivatives J,
- while keeping good convergence properties (superlinear).

Names

Secant, Broyden, or quasi-Newton method.

Secant method

Objective

Find (an approximation of) the solution of the system

$$F(x) = 0.$$

Inputs

- $F : \mathbb{R}^n \to \mathbb{R}^n$
- A first approximation of the solution $x_0 \in \mathbb{R}^n$;
- A first approximation of the Jacobian matrix A_0 (by default $A_0 = I$);
- Required precision $\varepsilon \in \mathbb{R}$, $\varepsilon > 0$.

Secant method (ctd)

Output

An approximation of the solution $x^* \in \mathbb{R}^n$.

Initialization

1. $x_1 = x_0 - A_0^{-1} f(x_0)$.
2. $d_0 = x_1 - x_0$.
3. $y_0 = f(x_1) - f(x_0)$.
4. $k = 1$.

Secant method (ctd)

Iterations

1. Broyden's update:

$$A_k = A_{k-1} + \frac{(y_{k-1} - A_{k-1} d_{k-1}) d_{k-1}^T}{d_{k-1}^T d_{k-1}}.$$

2. Compute d_k solution of $A_k d_k = -F(x_k)$.
3. $x_{k+1} = x_k + d_k$.
4. Compute $y_k = F(x_{k+1}) - F(x_k)$.
5. $k = k + 1$.

Stopping criterion

If $\|F(x_k)\| \leq \varepsilon$, then $x^* = x_k$.

Outline

Unconstrained optimization

$$\min_{x \in \mathbb{R}^n} f(x)$$

Local Newton method

Apply Newton's method to solve $\nabla f(x^*) = 0$

$$\begin{array}{rcl} F(x) & \to & \nabla f(x) \\ J(x) & \to & \nabla^2 f(x) \end{array}$$

Problems

- not guaranteed to converge
- $\nabla^2 f(x_k)^{-1}$ may not exist
- may converge to a point which is not a minimum

Local Newton: geometric interpretation

Quadratic approximation of f

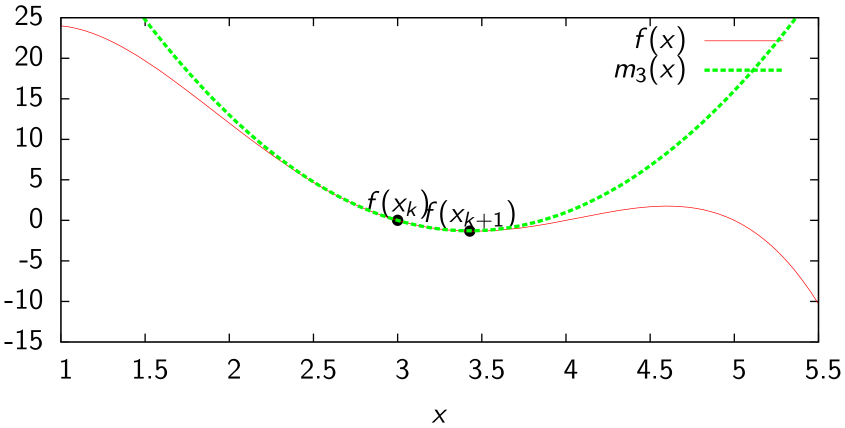
$$m_{x_k}(x) = f(x_k) + (x - x_k)^T \nabla f(x_k) + \frac{1}{2}(x - x_k)^T \nabla^2 f(x_k)(x - x_k).$$

Example: $f(x) = -x^4 + 12x^3 - 47x^2 + 60x$

1. $x_k = 3$. Quadratic model: $m_3(x) = 7x^2 - 48x + 81$
2. $x_k = 4$. Quadratic model: $m_4(x) = x^2 - 4x$
3. $x_k = 5$. Quadratic model: $m_5(x) = -17x^2 + 160x - 375$.

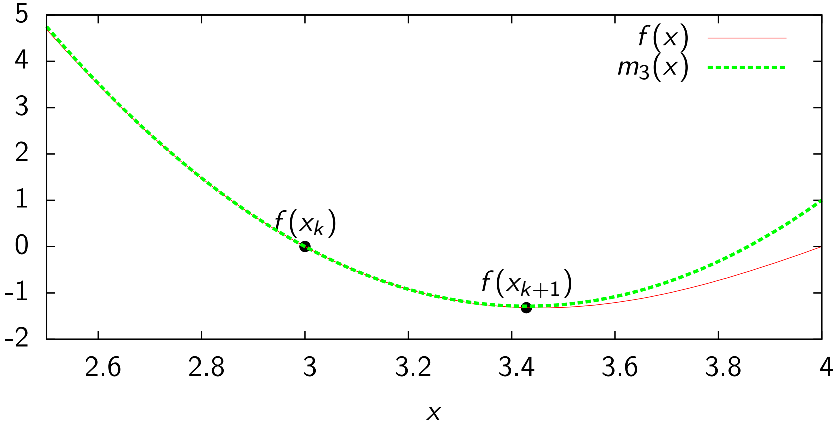
Local Newton: geometric interpretation

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



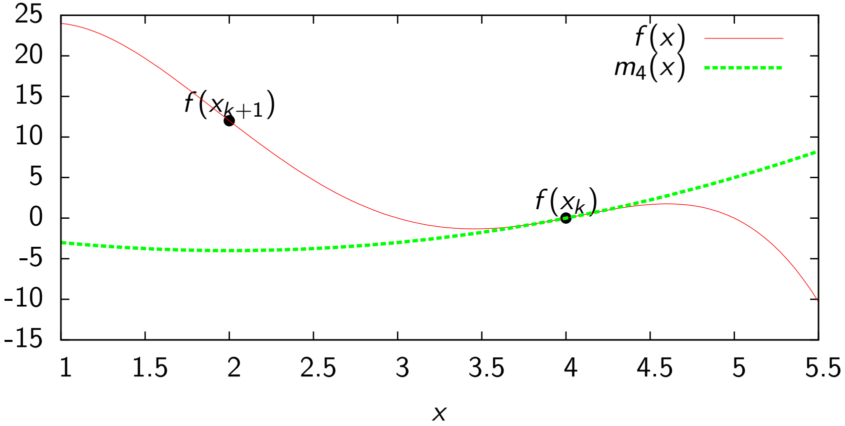
Local Newton: geometric interpretation

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



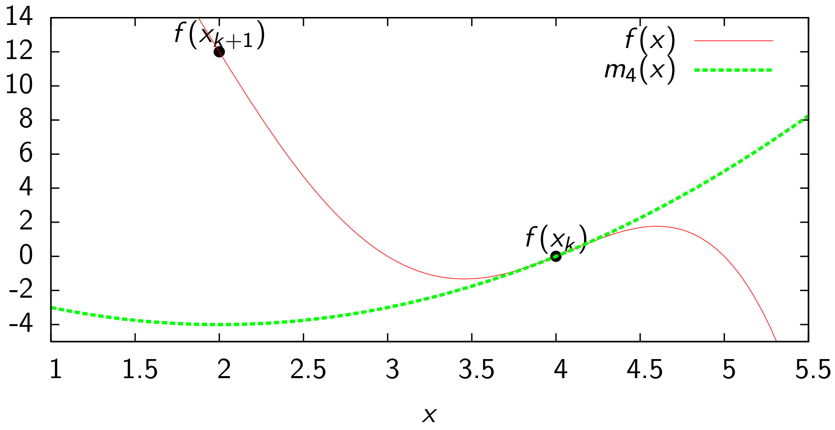
Local Newton: geometric interpretation

$m_4(x) = x^2 - 4x$: bad predictor



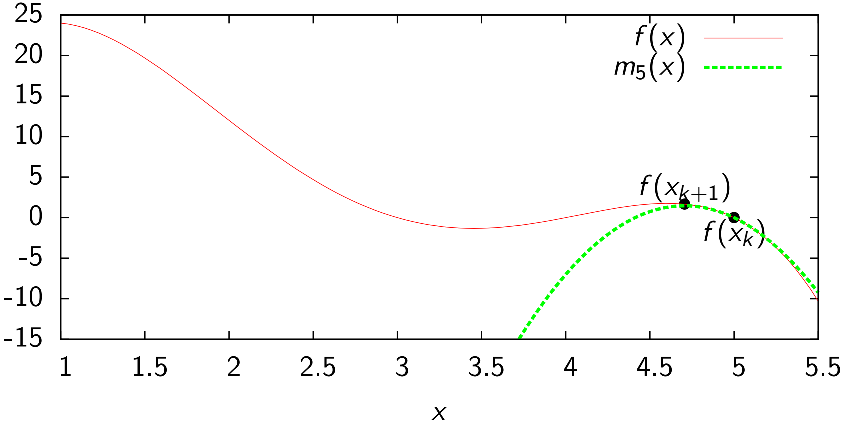
Local Newton: geometric interpretation

$m_4(x) = x^2 - 4x$: bad predictor (zoom)



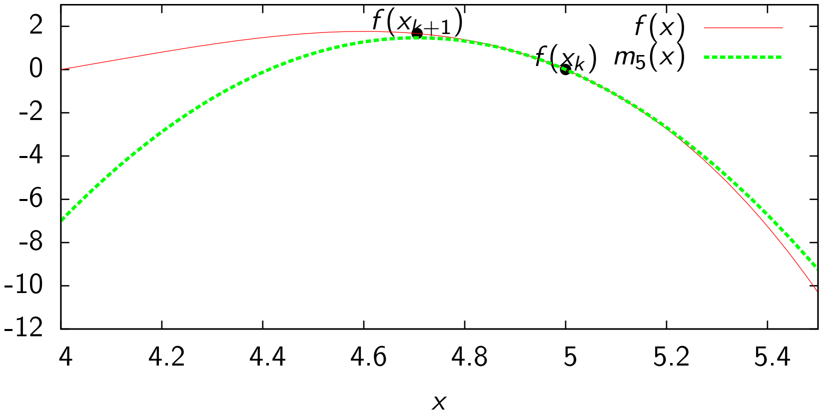
Local Newton: geometric interpretation

$m_5(x) = -17x^2 + 160x - 375$: concave



Local Newton: geometric interpretation

$m_5(x) = -17x^2 + 160x - 375$: concave (zoom)



Outline

Descent methods

Typical iteration

- Find a descent direction d_k such that $\nabla f(x_k)^T d_k < 0$.
- Find a step α_k such that $f(x_k + \alpha_k d_k) < f(x_k)$
- Compute $x_{k+1} = x_k + \alpha_k d_k$.

Descent methods: find a direction

Steepest descent

$$d_k = -\nabla f(x_k)$$

Exhibits slow to very slow convergence

Preconditioning

Change the metric

$$d_k = -D_k \nabla f(x_k)$$

where D_k is positive definite.

Newton

$$d_k = -(\nabla^2 f(x_k) + \lambda I)^{-1} \nabla f(x_k)$$

where λ is such that $\nabla f(x_k) + \lambda I$ is positive definite

Descent methods: find a step

Motivation

- Finding the optimal step is not cost effective
- Wolfe's conditions characterize steps guaranteeing convergence
- Use any step that verify these conditions

Conditions

- Avoid long steps, and request a sufficient decrease.
- Avoid short steps, and request a sufficient progress.

Descent methods: find a step

Wolfe 1: sufficient decrease

Consider

- Function $f : \mathbb{R}^n \to \mathbb{R}$
- Iterate $x_k \in \mathbb{R}^n$
- Direction $d_k \in \mathbb{R}^n$ such that $\nabla f(x_k)^T d_k < 0$
- Step $\alpha_k \in \mathbb{R}$, $\alpha_k > 0$

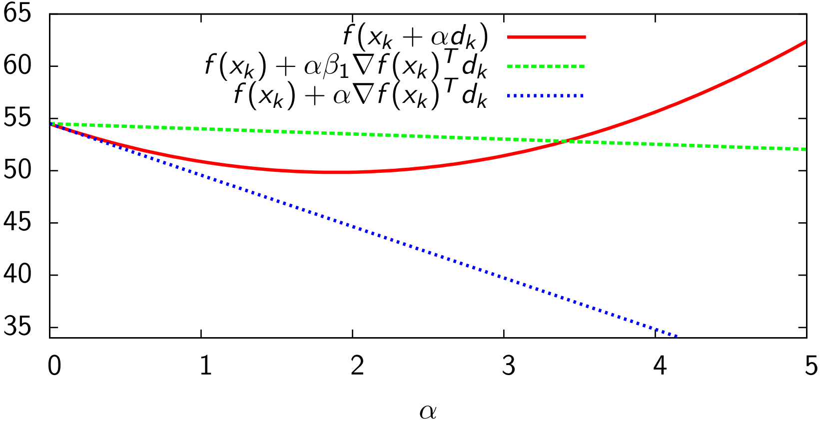
f decreases sufficiently at $x_k + \alpha_k d_k$ compared to x_k if

$$f(x_k + \alpha_k d_k) \leq f(x_k) + \alpha_k \beta_1 \nabla f(x_k)^T d_k,$$

where $0 < \beta_1 < 1$.

Descent methods: find a step

Wolfe 1: $\beta_1 = 0.1$



Descent methods: find a step

Wolfe 2: sufficient progress

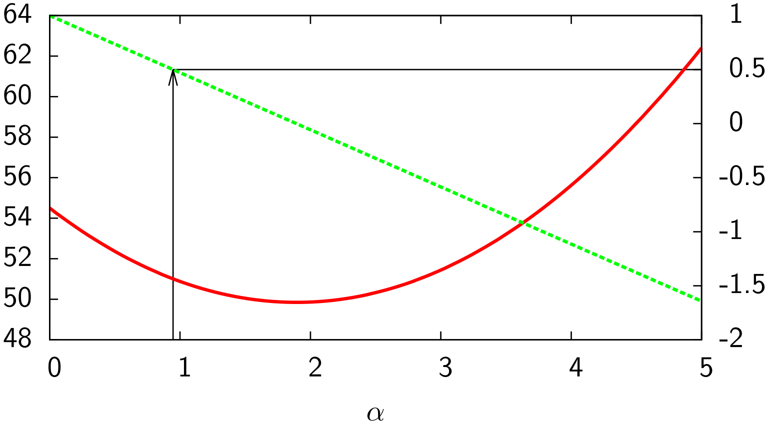
- Function $f : \mathbb{R}^n \to \mathbb{R}$
- Iterate $x_k \in \mathbb{R}^n$
- Direction $d_k \in \mathbb{R}^n$ such that $\nabla f(x_k)^T d_k < 0$
- Step $\alpha_k \in \mathbb{R}$, $\alpha_k > 0$

$x_k + \alpha_k d_k$ brings sufficient progress compared to x_k if

$$\nabla f(x_k + \alpha_k d_k)^T d_k \geq \beta_2 \nabla f(x_k)^T d_k,$$

with $0 < \beta_2 < 1$, $\beta_2 > \beta_1$.

Descent methods: find a step



Descent methods: find a step

Trial and errors

- Start with a first guess.
- If the step is too long (Wolfe 1 violated), make it shorter.
- If the step is too short (Wolfe 2 violated), make it longer.
- Until both conditions are verified.

Outline

Newton's method with linesearch

Objective

Find (an approximation of) a local minimum of

$$\min_{x \in \mathbb{R}^n} f(x).$$

Input

- Function $f : \mathbb{R}^n \to \mathbb{R}$ continuously differentiable;
- Gradient $\nabla f : \mathbb{R}^n \to \mathbb{R}^n$;
- Hessian $\nabla^2 f : \mathbb{R}^n \to \mathbb{R}^{n \times n}$;
- First approximation $x_0 \in \mathbb{R}^n$;
- Required precision $\varepsilon \in \mathbb{R}$, $\varepsilon > 0$.

Output

An approximation of the solution $x^* \in \mathbb{R}$.

Newton's method with linesearch

Iterations

1. Compute a lower triangular matrix and$\tau \geq 0$ such that
$$L_k L_k^T = \nabla^2 f(x_k) + \tau I,$$
using a modified Cholesky factorization.
2. Find z_k by solving the triangular system $L_k z_k = \nabla f(x_k)$.
3. Find d_k by solving the triangular system $L_k^T d_k = -z_k$.
4. Find α_k with line search starting with $\alpha_0 = 1$.
5. $x_{k+1} = x_k + \alpha_k d_k$, $k = k + 1$.

Stopping criterion

If $\|\nabla f(x_k)\| \leq \varepsilon$, then $x^* = x_k$.

Outline

Quasi-Newton

Ideas

- Adapt Broyden's (secant) method to optimization
- Additional constraint: the approximated matrix must be
  - symmetric
  - positive definite
- Update formula: BFGS (Broyden, Fletcher, Goldfarb and Shanno)

Quasi-Newton

Objective

Find (an approximation of) a local minimum of

$$\min_{x \in \mathbb{R}^n} f(x).$$

Input

- Function $f : \mathbb{R}^n \to \mathbb{R}$
- Gradient $\nabla f : \mathbb{R}^n \to \mathbb{R}^n$;
- First approximation of the solution $x_0 \in \mathbb{R}^n$;
- First approximation of the inverse of the hessian $H_0^{-1} \in \mathbb{R}^{n \times n}$ symmetric positive definite. Typically, $H_0^{-1} = I$.
- Required precision: $\varepsilon \in \mathbb{R}$, $\varepsilon > 0$.

Output

Quasi-Newton

Initialization

$k = 0$.

Iterations

1. Compute $d_k = -H_k^{-1} \nabla f(x_k)$.
2. Find α_k with linesearch starting with $\alpha_0 = 1$.
3. $x_{k+1} = x_k + \alpha_k d_k$.
4. $k = k + 1$.
5. Update H_k^{-1}

$$H_k^{-1} = \left(I - \frac{\bar{d}_{k-1} y_{k-1}^T}{\bar{d}_{k-1}^T y_{k-1}} \right) H_{k-1}^{-1} \left(I - \frac{\bar{y}_{k-1} d_{k-1}^T}{\bar{d}_{k-1}^T y_{k-1}} \right) + \frac{\bar{d}_{k-1} \bar{d}_{k-1}^T}{\bar{d}_{k-1}^T y_{k-1}}$$

with $\bar{d}_{k-1} = \alpha_{k-1} d_{k-1} = x_k - x_{k-1}$ and $y_{k-1} = \nabla f(x_k) - \nabla f(x_{k-1})$.

Quasi-Newton

Stopping criterion

If $\|\nabla f(x_k)\| \leq \varepsilon$, then $x^* = x_k$.

Summary

- Solving systems of equations: $\nabla f(x) = 0$
  - Newton
  - Quasi-Newton
- Unconstrained optimization
  - Local Newton
  - Linesearch
  - Quasi-Newton