Nonlinear Programming

Ekaterina A. Kostina

UNIVERSITÄT HEIDELBERG
ZUKUNFT SEIT 1386

IWR School 2018 “Advances in Mathematical Optimization”
October 8-12, 2018, Heidelberg

Outline

1. Basic definitions, optimality conditions
2. Algorithms for unconstrained problems
3. Algorithms for constrained problems

Nonlinear Programming Problem

- General problem formulation:

$$
\begin{aligned}
\min f(x) \qquad & f : D \in \mathbb{R}^n \to \mathbb{R} \\
\text{s.t. } g(x) = 0 \qquad & g : D \in \mathbb{R}^n \to \mathbb{R}^m \\
h(x) \geq 0 \qquad & h : D \in \mathbb{R}^n \to \mathbb{R}^k
\end{aligned}
$$

  - x variables
  - f objective function/ cost function/ $\min -f(x) \equiv -\max f(x)$
  - g equality constraints
  - h inequality constraints
  - f, g, h shall be sufficiently smooth (e.g. twice differentiable) functions

Derivatives

- First and second derivatives of the objective function or the constraints play an important role in optimization
- The first order derivatives are called the gradient (of the resp. fct)

$$\nabla f(x) = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \dots, \frac{\partial f}{\partial x_n} \right)^T$$

- and the second order derivatives are called the Hessian matrix

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

Local and Global Solutions

Local and Global Solutions

- Feasible set: $S = \{x \in \mathbb{R}^n : g(x) = 0, h(x) \geq 0\}$

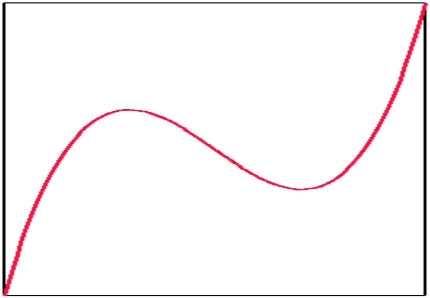
Local and Global Solutions

- Feasible set: $S = \{x \in \mathbb{R}^n : g(x) = 0, h(x) \geq 0\}$
- x^* global minimizer of f over S $\iff$ $x^* \in S$ and $f(x) \geq f(x^*)$, $\forall x \in S$

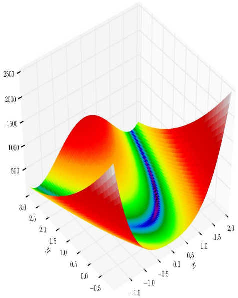
Local and Global Solutions

- Feasible set: $S = \{x \in \mathbb{R}^n : g(x) = 0, h(x) \geq 0\}$
- x^* global minimizer of f over $S \iff x^* \in S$ and $f(x) \geq f(x^*)$, $\forall x \in S$
- x^* local minimizer of f over $S \iff x^* \in S$ and there exists $\mathcal{N}(x^*, \delta)$ such that $f(x) \geq f(x^*)$, $\forall x \in S \cap \mathcal{N}(x^*, \delta)$ where $\mathcal{N}(x^*, \delta) := \{x \in \mathbb{R}^n : ||x - x^*||_2 \leq \delta\}$

Local and Global Solutions



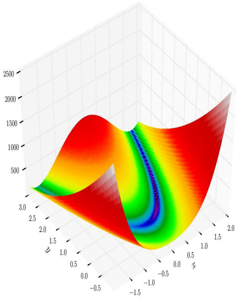
Local and Global Solutions

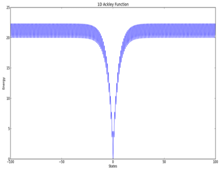


Rosenbrock's test function

see Wikipedia

Local and Global Solutions



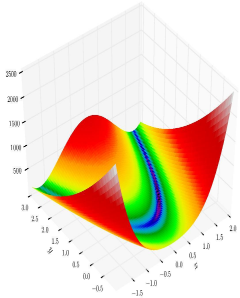


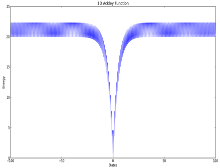
Rosenbrock's test function

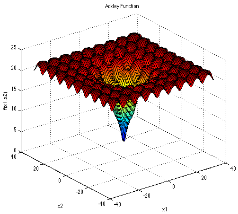
Ackeley's test function

see Wikipedia

Local and Global Solutions







Rosenbrock's test function

Ackeley's test function

see Wikipedia

Main Classes of Continuous Optimization Problems

Main Classes of Continuous Optimization Problems

- Linear programming: linear objective, linear constraints in the variables

$$\min_{x\in\mathbb{R}^n} \quad c^T x \qquad \text{subject to } a_i^T x = b_i, i \in E, a_i^T x \geq b_i, i \in I,$$

where $c, a_i \in \mathbb{R}^n$, for all i, E and I are finite index sets,

Main Classes of Continuous Optimization Problems

Quadratic programming: quadratic objective, linear constraints in the variables

$$\min_{x\in\mathbb{R}^n} \quad c^T x + \frac{1}{2} x^T H x \quad \text{subject to } a_i^T x = b_i, i \in E, a_i^T x \geq b_i, i \in I,$$

where $c, a_i \in \mathbb{R}^n$, for all i, E and I are finite index sets, $H \in \mathbb{R}^{n\times n}$ symmetric.

Main Classes of Continuous Optimization Problems

Quadratic programming: quadratic objective, linear constraints in the variables

$$\min_{x \in \mathbb{R}^n} \quad c^T x + \frac{1}{2} x^T H x \quad \text{subject to } a_i^T x = b_i, i \in E, a_i^T x \geq b_i, i \in I,$$

where $c, a_i \in \mathbb{R}^n$, for all i, E and I are finite index sets, $H \in \mathbb{R}^{n \times n}$ symmetric.

- Unconstrained nonlinear programming

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

Main Classes of Continuous Optimization Problems

Quadratic programming: quadratic objective, linear constraints in the variables

$$\min_{x \in \mathbb{R}^n} \quad c^T x + \frac{1}{2} x^T H x \quad \text{subject to } a_i^T x = b_i, i \in E, a_i^T x \geq b_i, i \in I,$$

where $c, a_i \in \mathbb{R}^n$, for all i, E and I are finite index sets, $H \in \mathbb{R}^{n \times n}$ symmetric.

- Unconstrained nonlinear programming

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

- Constrained nonlinear programming

$$\min_{x \in \mathbb{R}^n} \quad f(x) \quad \text{subject to } g(x) = 0, h(x) \geq 0.$$

Optimality Conditions for Unconstrained Optimization

$$\min f(x), \quad x \in \mathbb{R}^n$$

- Optimality conditions:
  - give algebraic characteriszations of solutions, suitable for computations
  - provide a way to guarantee that a candidate point is optimal (sufficient conditions)
  - indicate when a point is not optimal (necessary conditions)

Optimality Conditions for Unconstrained Optimization

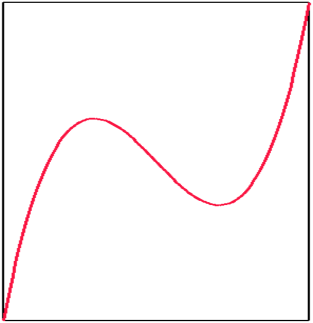
$$\min f(x), \quad x \in \mathbb{R}^n, \quad f \in \mathcal{C}^1$$

Optimality Conditions for Unconstrained Optimization

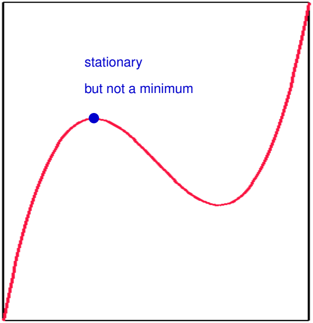
$$\min f(x), \quad x \in \mathbb{R}^n, \quad f \in \mathcal{C}^1$$

- Necessary conditions:
  x^* is a local minimizer of $f \Rightarrow \nabla f(x^*) = 0$ (stationarity)

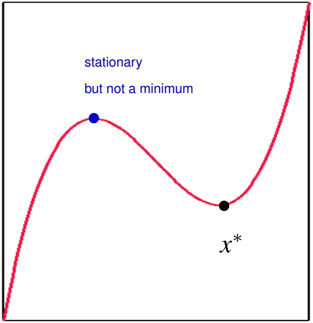
Optimality Conditions for Unconstrained Optimization



Optimality Conditions for Unconstrained Optimization



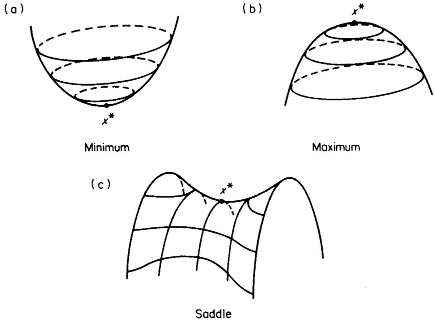
Optimality Conditions for Unconstrained Optimization



Stationary Points

(a)-(c) x^* is stationary: $\nabla f(x^*) = 0$

(a) $\nabla^2 f(x^*)$ positive definite: local minimum



(b) $\nabla^2 f(x^*)$ negative definite: local maximum

(c) $\nabla^2 f(x^*)$ indefinite: saddle point

Optimality Conditions for Unconstrained Convex Optimization

$$\min f(x), \quad x \in \mathbb{R}^n$$

Optimality Conditions for Unconstrained Convex Optimization

$$\min f(x), \quad x \in \mathbb{R}^n$$

- f convex

Optimality Conditions for Unconstrained Convex Optimization

$$\min f(x), \quad x \in \mathbb{R}^n$$

- f convex
  - x^* is a local minimizer of $f \Rightarrow x^*$ is a global minimizer of f

Optimality Conditions for Unconstrained Convex Optimization

$$\min f(x), \quad x \in \mathbb{R}^n$$

- f convex
  - x^* is a local minimizer of $f \Rightarrow x^*$ is a global minimizer of f
  - x^* is stationary $\Rightarrow x^*$ is a global minimizer of f

Optimality Conditions for Unconstrained Convex Optimization

$$\min f(x), \quad x \in \mathbb{R}^n$$

- f convex
  - x^* is a local minimizer of $f \Rightarrow x^*$ is a global minimizer of f
  - x^* is stationary $\Rightarrow x^*$ is a global minimizer of f
- f nonconvex

Optimality Conditions for Unconstrained Convex Optimization

$$\min f(x), \quad x \in \mathbb{R}^n$$

- f convex
  - x^* is a local minimizer of $f \Rightarrow x^*$ is a global minimizer of f
  - x^* is stationary $\Rightarrow x^*$ is a global minimizer of f
- f nonconvex $\rightarrow$ look at higher order derivatives

Second-Order Optimality Conditions for Unconstrained Optimization

$$\min f(x), \quad x \in \mathbb{R}^n, \quad f \in \mathcal{C}^2$$

Second-Order Optimality Conditions for Unconstrained Optimization

$$\min f(x), \quad x \in \mathbb{R}^n, \quad f \in \mathcal{C}^2$$

- Necessary second-order conditions:
  x^* is a local minimizer of $f \Rightarrow \nabla^2 f(x^*)$ positive semidefinite
  (f locally convex)

Second-Order Optimality Conditions for Unconstrained Optimization

$$\min f(x), \quad x \in \mathbb{R}^n, \quad f \in \mathcal{C}^2$$

- Necessary second-order conditions:
  x^* is a local minimizer of f $\Rightarrow$ $\nabla^2 f(x^*)$ positive semidefinite
  (f locally convex)

- Sufficient conditions:
  x^* stationary and $\nabla^2 f(x^*)$ positive definite $\Rightarrow$ x^* is a (strict) local minimizer of f

Stability

Stability

Let ε be a perturbation of the problem, then the solution $x(\varepsilon)$ should be a small perturbation of the exact solution x^*:

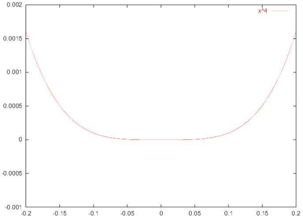
$$||x(\varepsilon) - x^*|| \leq c||\varepsilon||$$

Stability

Let ε be a perturbation of the problem, then the solution $x(\varepsilon)$ should be a small perturbation of the exact solution x^*:

$$||x(\varepsilon) - x^*|| \leq c||\varepsilon||$$

$f(x) = x^4$: Minimum at $x^* = 0$

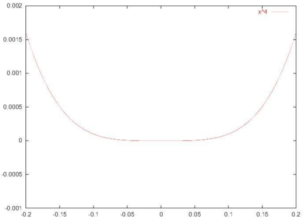


Stability

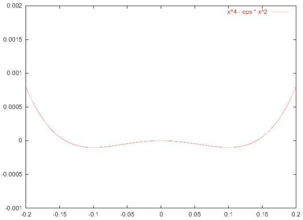
Let ε be a perturbation of the problem, then the solution $x(\varepsilon)$ should be a small perturbation of the exact solution x^*:

$$||x(\varepsilon) - x^*|| \leq c||\varepsilon||$$

$f(x) = x^4$: Minimum at $x^* = 0$



$f(x) = x^4 - \varepsilon x^2$: Maximum at $x = 0$,
Minima at $x = \pm\sqrt{\varepsilon}$

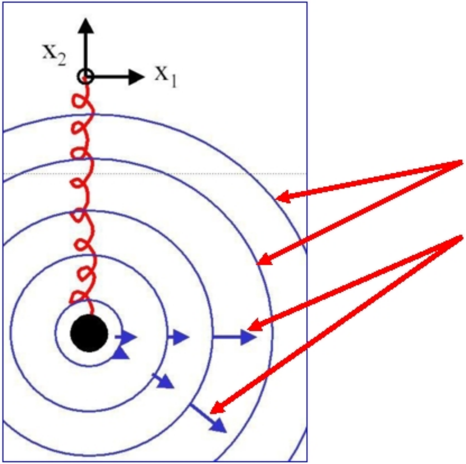


Stability

- In the example problem the sufficient optimality conditions were not satisfied ($\nabla^2 f(x^*)$ is not positive definite)
- One can show:

> Optima that satisfy the sufficient optimality conditions are stable against perturbations

Ball on a spring without constraints



$$\min_{x \in \mathbb{R}^2} \quad x_1^2 + x_2^2 + m x_2$$

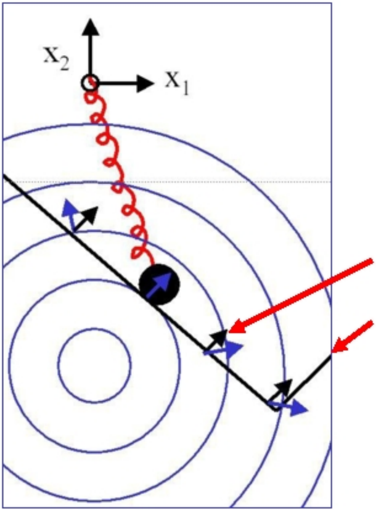
contour lines of $f(x)$

gradient vector
$\nabla f(x) = (2x_1, 2x_2 + m)$

unconstrained minimum:

$$0 = \nabla f(x^*) \Leftrightarrow (x_1^*, x_2^*) = (0, -\frac{m}{2})$$

Ball on a spring with constraints



$$\min_{x \in \mathbb{R}^2} \quad x_1^2 + x_2^2 + m x_2$$

$$h_1(x) = 1 + x_1 + x_2 \geq 0$$

$$h_2(x) = 3 - x_1 + x_2 \geq 0$$

gradient ∇h_1 of active constraint

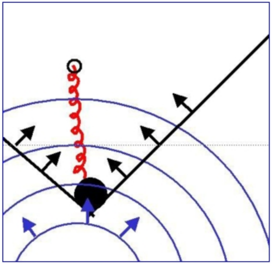
inactive constraint h_2

constrained minimum:

$$\nabla f(x^*) = \mu_1 \nabla h_1(x^*)$$

μ_1 is Lagrange multiplier

Ball on a spring with active constraints



$$\min_{x\in\mathbb{R}^2} \quad x_1^2 + x_2^2 + m x_2$$

$$h_1(x) = 1 + x_1 + x_2 \geq 0$$

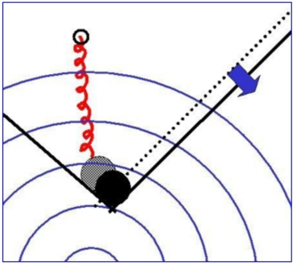
$$h_2(x) = 3 - x_1 + x_2 \geq 0$$

"equilibrium of forces"

$$\nabla f(x^*) \quad = \quad \mu_1 \nabla h_1(x^*) + \mu_2 \nabla h_2(x^*), \quad \mu_1 \geq 0, \mu_2 \geq 0$$

μ_1, μ_2, are Lagrange multipliers

Multipliers as “shadow prices”



old constraint: $h_2(x) \geq 0$

new constraint: $h_2(x) + \varepsilon \geq 0$

- What happens if we relax a constraint?
- Feasible set becomes larger, so new minimum $f(x_\varepsilon^*)$ becomes smaller.
- How much would we gain?

$$f(x_\varepsilon^*) \approx f(x^*) - \varepsilon\mu_2$$

- Multipliers show the hidden cost of constraints.

KKT Conditions for Constrained Optimization

$$\min_{x \in \mathbb{R}^n} f(x), \quad \text{s.t. } g(x) = 0, h(x) \geq 0.$$

KKT Conditions for Constrained Optimization

$$\min_{x \in \mathbb{R}^n} f(x), \quad \text{s.t. } g(x) = 0, h(x) \geq 0.$$

- Lagrangian function $\mathcal{L} : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^k \to \mathbb{R}$

$$\mathcal{L}(x, \lambda, \mu) := f(x) - \sum_i \lambda_i g_i(x) - \sum_i \mu_i h_i(x)$$

KKT Conditions for Constrained Optimization

$$\min_{x \in \mathbb{R}^n} f(x), \quad \text{s.t. } g(x) = 0, h(x) \geq 0.$$

- Lagrangian function $\mathcal{L} : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^k \to \mathbb{R}$

$$\mathcal{L}(x, \lambda, \mu) := f(x) - \sum_i \lambda_i g_i(x) - \sum_i \mu_i h_i(x)$$

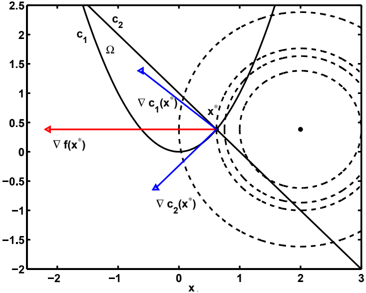
- Karush-Kuhn-Tucker (KKT) point:
  x is a KKT point if there exist $\lambda \in \mathbb{R}^m$ and $\mu \in \mathbb{R}^k$ such that (x, λ, μ) satisfies

$$g(x) = 0, h(x) \geq 0$$

$$\nabla_x \mathcal{L}(x, \lambda, \mu) = 0 \Leftrightarrow \nabla f(x) = \sum_i \lambda_i \nabla g_i(x) + \sum_i \mu_i \nabla h_i(x)$$

$$\mu \geq 0, \mu_i h_i(x) = 0, i = 1, ..., k \Leftrightarrow \mu_i = 0 \text{ or } h_i(x) = 0$$

KKT Conditions: Illustration



Optimality Conditions for Constrained Optimization

Optimality Conditions for Constrained Optimization

- In general in order to derive optimality conditions we need constraints/feasible set to satisfy regularity assumptions called constraint qualification (CQ)

Optimality Conditions for Constrained Optimization

- In general in order to derive optimality conditions we need constraints/feasible set to satisfy regularity assumptions called constraint qualification (CQ)
- CQ: linearized approximation of constraint functions covers the essential geometry of the feasible set

Optimality Conditions for Constrained Optimization

- In general in order to derive optimality conditions we need constraints/feasible set to satisfy regularity assumptions called constraint qualification (CQ)
- CQ: linearized approximation of constraint functions covers the essential geometry of the feasible set
- Examples

Optimality Conditions for Constrained Optimization

- In general in order to derive optimality conditions we need constraints/feasible set to satisfy regularity assumptions called constraint qualification (CQ)
- CQ: linearized approximation of constraint functions covers the essential geometry of the feasible set
- Examples
  - The Linear Independence Constraint Qualification (LICQ) at $x \Rightarrow$ $\nabla g_i(x), i = 1, ..., m$, $\nabla h_i(x)$, $i \in I(x)$, are linearly independent, where $I(x) := \{i : 1 \leq i \leq k, h_i(x) = 0\}$ “active set”

Optimality Conditions for Constrained Optimization

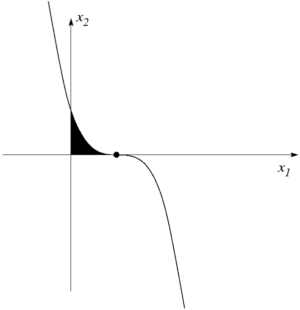
- In general in order to derive optimality conditions we need constraints/feasible set to satisfy regularity assumptions called constraint qualification (CQ)
- CQ: linearized approximation of constraint functions covers the essential geometry of the feasible set
- Examples
  - The Linear Independence Constraint Qualification (LICQ) at $x \Rightarrow$ $\nabla g_i(x), i = 1, ..., m, \nabla h_i(x), i \in I(x)$, are linearly independent, where $I(x) := \{i : 1 \leq i \leq k, h_i(x) = 0\}$ "active set"
  - All active constraints (equalities and active inequalities) are linear

Optimality Conditions for Constrained Optimization

- In general in order to derive optimality conditions we need constraints/feasible set to satisfy regularity assumptions called constraint qualification (CQ)
- CQ: linearized approximation of constraint functions covers the essential geometry of the feasible set
- Examples
  - The Linear Independence Constraint Qualification (LICQ) at $x \Rightarrow$ $\nabla g_i(x), i = 1, ..., m, \ \nabla h_i(x), \ i \in I(x)$, are linearly independent, where $I(x) := \{i : 1 \leq i \leq k, h_i(x) = 0\}$ "active set"
  - All active constraints (equalities and active inequalities) are linear
  - Mangansarian-Fromovitz CQ at $x \Rightarrow \nabla g_i(x), i = 1, ..., m$ linearly independent / or linear and $\exists p \in \mathbb{R}^n$ such that $\nabla g(x)^T p = 0, \ \nabla h_i(x)^T p > 0, \ i \in I(x)$

No CQ: Example from Fiacco, McCormick

$$
\begin{aligned}
h_1(x) &:= (1-x_1)^3 - x_2 &\geq& \quad 0 \quad x_2 \leq (1-x_1)^3 \\
h_2(x) &:= x_1 &\geq& \quad 0 \\
h_3(x) &:= x_2 &\geq& \quad 0
\end{aligned}
$$



$$
\nabla h_1 = \begin{pmatrix} -3(1-x_1) \\ -1 \end{pmatrix}
$$

$$
\nabla h_2 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}
$$

$$
\nabla h_3 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}
$$

Active inequalities in $(1,0)^T$:

$$h_3(x) = 0 \text{ and } h_1(x) = 0$$

Optimality Conditions for Constrained Optimization

Optimality Conditions for Constrained Optimization

- First-order necessary optimality conditions:
  Let x^* be optimal and CQ are satisfied in x^*, then x^* is a KKT point.

Optimality Conditions for Constrained Optimization

- First-order necessary optimality conditions:
  Let x^* be optimal and CQ are satisfied in x^*, then x^* is a KKT point.
- Second-order necessary optimality conditions:

Optimality Conditions for Constrained Optimization

- First-order necessary optimality conditions:
  Let x^* be optimal and CQ are satisfied in x^*, then x^* is a KKT point.
- Second-order necessary optimality conditions:
- Let x^* be optimal and CQ are satisfied in x^*, then x^* is a KKT point and the Hessian $\nabla^2 \mathcal{L}(x^*, \lambda^*, \mu^*)$ of the Lagrange function is positive semidefinite at the tangent set $T(x^*)$:

$$p^T \nabla^2 \mathcal{L}(x^*, \lambda^*, \mu^*) p \geq 0, \forall p \in T(x^*)$$
$$T(x^*) := \{s : s^T \nabla g_i(x^*) = 0,\ i = 1, ..., m,$$
$$s^T \nabla h_i(x^*) \geq 0,\ i \in I(x^*)\}$$

Optimality Conditions for Constrained Optimization

- First-order necessary optimality conditions:
  Let x^* be optimal and CQ are satisfied in x^*, then x^* is a KKT point.
- Second-order necessary optimality conditions:
- Let x^* be optimal and CQ are satisfied in x^*, then x^* is a KKT point and the Hessian $\nabla^2\mathcal{L}(x^*, \lambda^*, \mu^*)$ of the Lagrange function is positive semidefinite at the tangent set $T(x^*)$:

$$p^T \nabla^2 \mathcal{L}(x^*, \lambda^*, \mu^*) p \geq 0, \forall p \in T(x^*)$$
$$T(x^*) := \{s : s^T \nabla g_i(x^*) = 0,\ i = 1, ..., m,$$
$$s^T \nabla h_i(x^*) \geq 0,\ i \in I(x^*)\}$$

- Sufficient optimality conditions: if KKT conditions hold and $\nabla^2\mathcal{L}(x^*, \lambda^*, \mu^*)$ is positive definite at $\tilde{T}(x^*, \lambda^*)$, then x^* is optimal, $\tilde{T}(x^*, \lambda^*) := \{s \in T(x^*) : s^T \nabla h_i(x^*) = 0,\ i \in I(x^*)$ with $\mu_i > 0\}$

Algorithms for Unconstrained Optimization

Algorithms for Unconstrained Optimization

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

Algorithms for Unconstrained Optimization

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

- Find a local minimizer x^* of $f(x)$, i.e. a point satisfying
  - $\nabla f(x^*) = 0$ (stationarity)
  - and $\nabla^2 f(x^*)$ positive definite

Algorithms for Unconstrained Optimization

- Basic structure of most algorithms:
  - choose start value x^0
  - for $k = 1, \ldots,$
    - determine search (descent) direction p^k
    - determine steplenght α^k
    - new iterate $x^{k+1} = x^k + \alpha^k p^k$
    - check for convergence
- Optimization algorithms differ in the choice of p^k and α^k

Properties of Optimization Algorithms

Properties of Optimization Algorithms

- Optimization algorithms are iterative, i.e. they create an infinite (in practice finite) sequence of points $\{x^k\}$ converging to the optimum x^*

Properties of Optimization Algorithms

- Optimization algorithms are iterative, i.e. they create an infinite (in practice finite) sequence of points $\{x^k\}$ converging to the optimum x^*

- Two types of convergence:

Properties of Optimization Algorithms

- Optimization algorithms are iterative, i.e. they create an infinite (in practice finite) sequence of points $\{x^k\}$ converging to the optimum x^*
- Two types of convergence:
  - Local convergence: convergence of the full-step ($\alpha_k \equiv 1$) algorithm near the solution

Properties of Optimization Algorithms

- Optimization algorithms are iterative, i.e. they create an infinite (in practice finite) sequence of points $\{x^k\}$ converging to the optimum x^*

- Two types of convergence:
  - Local convergence: convergence of the full-step ($\alpha_k \equiv 1$) algorithm near the solution
  - Global convergence: convergence of an algorithm starting from an any arbitrary point x^0

Rates of Convergence

Rates of Convergence

- $\{x^k\} \subset \mathbb{R}^n$, $x^* \in \mathbb{R}^n$, $\{x^k\} \to x^*$ as $k \to \infty$

Rates of Convergence

- $\{x^k\} \subset \mathbb{R}^n$, $x^* \in \mathbb{R}^n$, $\{x^k\} \to x^*$ as $k \to \infty$
- $\{x^k\} \to x^*$ with rate r if

$$\frac{||x^{k+1} - x^*||}{||x^k - x^*||^r} = c < \infty, \text{ for sufficiently large } k$$

Rates of Convergence

- $\{x^k\} \subset \mathbb{R}^n$, $x^* \in \mathbb{R}^n$, $\{x^k\} \to x^*$ as $k \to \infty$
- $\{x^k\} \to x^*$ with rate r if

$$\frac{||x^{k+1} - x^*||}{||x^k - x^*||^r} = c < \infty, \text{ for sufficiently large } k$$

  - $r = 1$: linear convergence $(c < 1)$

Rates of Convergence

- $\{x^k\} \subset \mathbb{R}^n$, $x^* \in \mathbb{R}^n$, $\{x^k\} \to x^*$ as $k \to \infty$
- $\{x^k\} \to x^*$ with rate r if

$$\frac{||x^{k+1} - x^*||}{||x^k - x^*||^r} = c < \infty, \text{ for sufficiently large } k$$

  - $r = 1$: linear convergence $(c < 1)$
  - $r = 2$: quadratic convergence

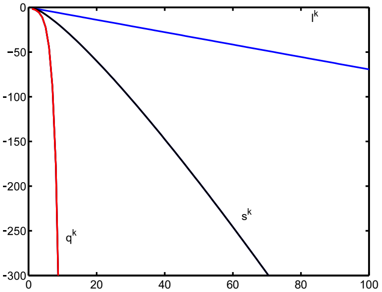
Rates of Convergence

- $\{x^k\} \subset \mathbb{R}^n$, $x^* \in \mathbb{R}^n$, $\{x^k\} \to x^*$ as $k \to \infty$
- $\{x^k\} \to x^*$ with rate r if

$$\frac{||x^{k+1} - x^*||}{||x^k - x^*||^r} = c < \infty, \text{ for sufficiently large } k$$

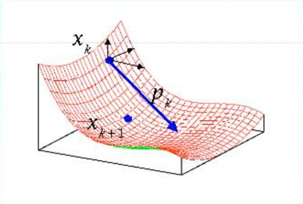
  - $r = 1$: linear convergence $(c < 1)$
  - $r = 2$: quadratic convergence
  - superlinear convergence: $\frac{||x^{k+1} - x^*||}{||x^k - x^*||} \to 0$ as $k \to \infty$

Rates of Convergence



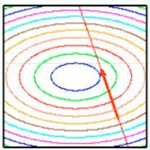
Generic Linesearch Algorithm

Search direction p^k:
f must decrease
along the direction p^k
$\nabla f(x^k)p^k < 0$



Steplength α^k
to guarantee global convergence:
solve $1D$ minimization problem
(exact or inexact):

$$\alpha^k = \text{arg}\min_{\alpha} f(x^k + \alpha p^k)$$



Computation of Steplength

- Ideal: Move to (global) minimum on the selected line (univariate optimization, exact line search)

$$\alpha^k = \text{arg} \min_{\alpha} f(x^k + \alpha p^k)$$

- In practice: approximate solution may guarantee global convergence, perform only inexact line search

$$\alpha^k \approx \text{arg} \min_{\alpha} f(x^k + \alpha p^k)$$

- Problem: how to guarantee sufficient decrease?
  - Answer: Check e.g. if α^k satisfies Armijo-Wolfe conditions

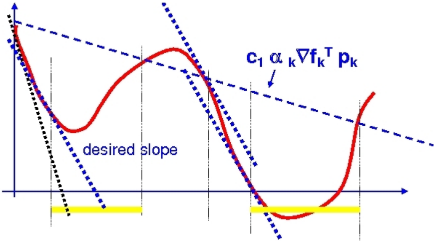
Armijo-Wolfe Conditions for Inexact Line Search

- Armijo condition (sufficent decrease condition):

$$f(x^k + \alpha p^k) \leq f(x^k) + c_1 \alpha^k \nabla^T f(x^k) p^k, \quad c_1 \in (0, 1)$$

- Curvature condition:

$$\nabla^T f(x^k + \alpha p^k) p^k \geq c_2 \nabla^T f(x^k) p^k, \quad c_2 \in (c_1, 1)$$



Global Convergence of Generic Line Search Method

Global Convergence of Generic Line Search Method

- $f \in \mathcal{C}^1$ bounded from below
- ∇f Lipschitz continuous
- Apply Armijo-Wolfe inexact line search

Global Convergence of Generic Line Search Method

- $f \in \mathcal{C}^1$ bounded from below
- ∇f Lipschitz continuous
- Apply Armijo-Wolfe inexact line search
- Then

Global Convergence of Generic Line Search Method

- $f \in \mathcal{C}^1$ bounded from below
- ∇f Lipschitz continuous
- Apply Armijo-Wolfe inexact line search
- Then
  - either there exist $l \geq 0$ such that $\nabla f(x^l) = 0$

Global Convergence of Generic Line Search Method

- $f \in \mathcal{C}^1$ bounded from below
- ∇f Lipschitz continuous
- Apply Armijo-Wolfe inexact line search
- Then
  - either there exist $l \geq 0$ such that $\nabla f(x^l) = 0$
  - or $\min\{\frac{|\nabla f(x^k)^T p^k|}{||s^k||}, |\nabla f(x^k)^T p^k|\} \to 0$ as $k \to \infty$

Global Convergence of Generic Line Search Method

Global Convergence of Generic Line Search Method

- Global convergence theorem:

Global Convergence of Generic Line Search Method

- Global convergence theorem:
  - if $\nabla f(x^k) \neq 0$ for all k then

    $$\lim_{k\to\infty} ||\nabla f(x^k)|| \cos\theta^k \min\{1, ||p^k||\} = 0,$$

    where θ is an angle between p and $-\nabla f(x)$

Global Convergence of Generic Line Search Method

- Global convergence theorem:
  - if $\nabla f(x^k) \neq 0$ for all k then

    $$\lim_{k\to\infty} ||\nabla f(x^k)|| \cos\theta^k \min\{1, ||p^k||\} = 0,$$

    where θ is an angle between p and $-\nabla f(x)$
- For global convergence (i.e.$||\nabla f(x^k)|| \to 0$ as $k \to \infty$) we need
  - not only p^k to be descent direction

Global Convergence of Generic Line Search Method

- Global convergence theorem:
  - if $\nabla f(x^k) \neq 0$ for all k then

    $$\lim_{k \to \infty} ||\nabla f(x^k)|| \cos \theta^k \min\{1, ||p^k||\} = 0,$$

    where θ is an angle between p and $-\nabla f(x)$
- For global convergence (i.e.$||\nabla f(x^k)|| \to 0$ as $k \to \infty$) we need
  - not only p^k to be descent direction
  - but also $\cos \theta^k \geq \delta > 0$ for all k
    (i.e.p^k and $\nabla f(x^k)$) should not become nearly orthogonal!

Computation of the Search Direction

- For the determination of p^k frequently first and second order derivatives of $f(x^k)$ are used
- We discuss:
  - Steepest descent method
  - Newton's method
  - Quasi-Newton methods
- Left out: conjugate gradients

Algorithm 1: Steepest Descent Method

- Based on first order Taylor series approximation of objective function

$$f(x^k + p^k) = f(x^k) + \underbrace{\nabla^T f(x^k) p^k}_{} + \ldots$$

- maximum descent, if

$$\frac{\nabla^T f(x^k) p^k}{||p^k||} \to \min!$$

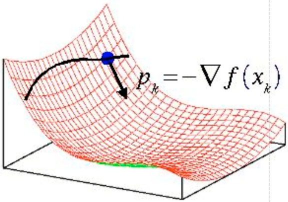
$$\Rightarrow p^k = -\nabla f(x^k)$$

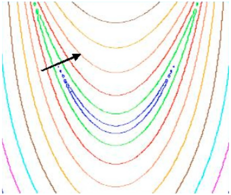
Algorithm 1: Steepest Descent Method

- Choose steepest descent direction, perform (exact) line search:

$$p^k = -\nabla f(x^k) \quad x^{k+1} = x^k - \alpha^k \nabla f(x^k)$$

- search direction is perpendicular to level sets of $f(x)$





Convergence of Steepest Descent Method

Convergence of Steepest Descent Method

- Excellent global convergence properties under weak assumptions

Convergence of Steepest Descent Method

- Excellent global convergence properties under weak assumptions
- Asymptotically, convergence rate is linear

i.e. $|f(x^{k+1}) - f(x^*)| \leq C|f(x^k) - f(x^*)|$

  - with $C < 1$

Convergence of Steepest Descent Method

- Excellent global convergence properties under weak assumptions
- Asymptotically, convergence rate is linear

  i.e. $|f(x^{k+1}) - f(x^*)| \leq C|f(x^k) - f(x^*)|$

  - with $C < 1$
  - Convergence can be very slow if C close to 1

Convergence of Steepest Descent Method

- Excellent global convergence properties under weak assumptions
- Asymptotically, convergence rate is linear

  i.e. $|f(x^{k+1}) - f(x^*)| \leq C|f(x^k) - f(x^*)|$

  - with $C < 1$
  - Convergence can be very slow if C close to 1

- If $f(x) = x^T Ax$, A positive definite (quadratic convex) λ_i are eigenvalues of A, one can show that

$$C \approx \frac{\lambda_{max} - \lambda_{min}}{\lambda_{max} + \lambda_{min}}$$

Convergence of Steepest Descent Method

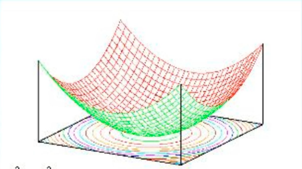
- Excellent global convergence properties under weak assumptions
- Asymptotically, convergence rate is linear

  i.e. $|f(x^{k+1}) - f(x^*)| \leq C|f(x^k) - f(x^*)|$

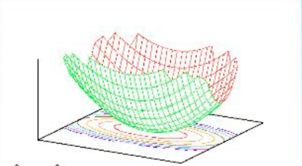
  - with $C < 1$
  - Convergence can be very slow if C close to 1

- If $f(x) = x^T Ax$, A positive definite (quadratic convex) λ_i are eigenvalues of A, one can show that

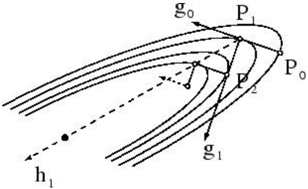
$$C \approx \frac{\lambda_{max} - \lambda_{min}}{\lambda_{max} + \lambda_{min}}$$



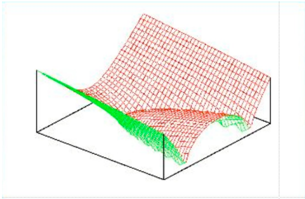
$x^2+y^2 \rightarrow C=0$



$x^2+5y^2 \rightarrow C\approx 0.6$



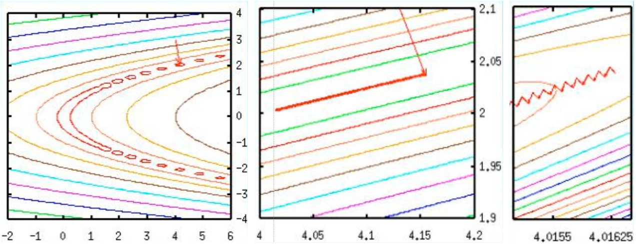
Example - Steepest Descent Method



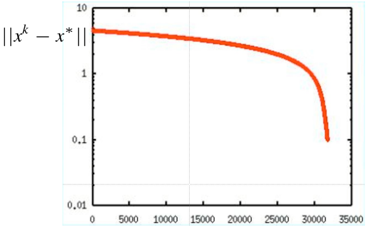
$$f(x,y) = \left((x-y^2)^2 + \frac{1}{100} \right)^{\frac{1}{4}} + \frac{1}{100} y^2$$

banana valley function

global minimum at $x = y = 0$



Example - Steepest Descent Method



Convergence of steepest descent method:

- needs almost 35.000 iterations to come closer than 0.1 to the solution
- mean value of convergence rate $C \approx 0.99995$
- it holds at $(x = 4, y = 2)$

$$\lambda_{min} = 0.1, \lambda_{max} = 268, C \approx \frac{268 - 0.1}{268 - 0.1} \approx 0.9993$$

Summary: Steepest Descent Methods

Summary: Steepest Descent Methods

- first-order method (inexpensive)

Summary: Steepest Descent Methods

- first-order method (inexpensive)
- global convergence under weak assumptions, but no second-order optimality guarantees for the generated solution

Summary: Steepest Descent Methods

- first-order method (inexpensive)
- global convergence under weak assumptions, but no second-order optimality guarantees for the generated solution
- scale-dependent: when the objective poorly scaled, very slow convergence, cumulation of round-off errors and break-down

Summary: Steepest Descent Methods

- first-order method (inexpensive)
- global convergence under weak assumptions, but no second-order optimality guarantees for the generated solution
- scale-dependent: when the objective poorly scaled, very slow convergence, cumulation of round-off errors and break-down
- useful for some special applications (e.g. in data analysis)

Algorithm 2: Newton's Method

Algorithm 2: Newton's Method

- Based on second order Taylor series approximation of objective function

$$f(x^k + p^k) = f(x^k) + \underbrace{\nabla^T f(x^k) p^k + \frac{1}{2}(p^k)^T \nabla^2 f(x^k) p^k}_{} + \ldots$$

Algorithm 2: Newton's Method

- Based on second order Taylor series approximation of objective function

$$f(x^k + p^k) = f(x^k) + \underbrace{\nabla^T f(x^k) p^k + \frac{1}{2}(p^k)^T \nabla^2 f(x^k) p^k}_{} + \ldots$$

- maximum descent, if

$$\nabla^T f(x^k) p^k + \frac{1}{2}(p^k)^T \nabla^2 f(x^k) p^k \to \min!$$
$$\to p^k = -(\nabla^2 f(x^k))^{-1} \nabla f(x^k)$$

Algorithm 2: Newton's Method

- Based on second order Taylor series approximation of objective function

$$f(x^k + p^k) = f(x^k) + \underbrace{\nabla^T f(x^k)p^k + \frac{1}{2}(p^k)^T \nabla^2 f(x^k)p^k}+ \ldots$$

- maximum descent, if

$$\nabla^T f(x^k)p^k + \frac{1}{2}(p^k)^T \nabla^2 f(x^k)p^k \to \text{min!}$$
$$\to p^k = -(\nabla^2 f(x^k))^{-1} \nabla f(x^k)$$

- p^k is "Newton Direction"

Algorithm 2: Newton's Method

- Based on second order Taylor series approximation of objective function

$$f(x^k + p^k) = f(x^k) + \underbrace{\nabla^T f(x^k) p^k + \frac{1}{2}(p^k)^T \nabla^2 f(x^k) p^k}_{} + \ldots$$

- maximum descent, if

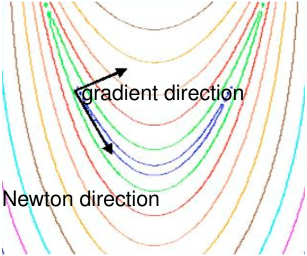
$$\nabla^T f(x^k) p^k + \frac{1}{2}(p^k)^T \nabla^2 f(x^k) p^k \to \min!$$
$$\to p^k = -(\nabla^2 f(x^k))^{-1} \nabla f(x^k)$$

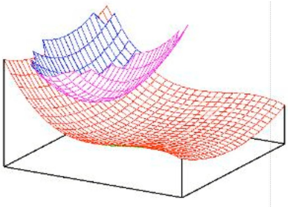
- p^k is "Newton Direction"
- p^k is decrease direction if the hessian $\nabla^2 f(x^k)$ is positive definite!

Visualization of Newton's method

- p^k minimizes quadratic approximation of the objective

$$Q(p^k) = f(x^k) + \nabla^T f(x^k) p^k + \frac{1}{2} (p^k)^T \nabla^2 f(x^k) p^k$$





Why is it called Newton's method?

- Newton's method finds zeros of nonlinear equations. Here: find solution of the equation

$$\nabla f(x) = 0$$

- Newton's idea: use Taylor series of ∇f at x^k:

$$\nabla f(x^k + p^k) \approx \nabla f(x^k) + \nabla^2 f(x^k) p^k = 0!$$

- and to make this zero, set:

$$p^k = \underbrace{-(\nabla^2 f(x^k))^{-1} \nabla f(x^k)}_{\textit{Newton direction}}$$

- (Full step) Newton's method: iterate

$$x^{k+1} = x^k + p^k$$

Convergence of Newton's method

Convergence of Newton's method

- Newton's method has quadratic rate of local convergence

  i.e. $||x^{k+1} - x^*|| \leq C||x^k - x^*||^2, C < \infty$

Convergence of Newton's method

- Newton's method has quadratic rate of local convergence

  i.e. $||x^{k+1} - x^*|| \leq C||x^k - x^*||^2, C < \infty$

- This is very fast if we are close to a solution:
- Doubles the correct digits in each iteration!

Convergence of Newton's method

- Newton's method has quadratic rate of local convergence

  i.e. $||x^{k+1} - x^*|| \leq C||x^k - x^*||^2, C < \infty$

- This is very fast if we are close to a solution:
- Doubles the correct digits in each iteration!
- Problem:
  - If the start value x^0 of the iteration is near to a saddle point or a maximum, the full step method converges to this saddle point or maximum.

Convergence of Newton's method

- Newton's method has quadratic rate of local convergence

  i.e. $||x^{k+1} - x^*|| \leq C||x^k - x^*||^2, C < \infty$

- This is very fast if we are close to a solution:
- Doubles the correct digits in each iteration!
- Problem:
  - If the start value x^0 of the iteration is near to a saddle point or a maximum, the full step method converges to this saddle point or maximum.
  - Line search helps, but is only possible if p is descent direction, i.e. if $\nabla^2 f$ positive definite.

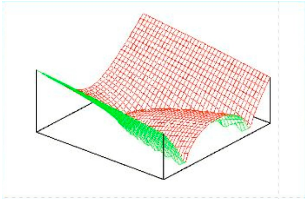
Convergence of Newton's method

- Newton's method has quadratic rate of local convergence

$$\text{i.e. } ||x^{k+1} - x^*|| \leq C||x^k - x^*||^2, C < \infty$$

- This is very fast if we are close to a solution:
- Doubles the correct digits in each iteration!
- Problem:
  - If the start value x^0 of the iteration is near to a saddle point or a maximum, the full step method converges to this saddle point or maximum.
  - Line search helps, but is only possible if p is descent direction, i.e. if $\nabla^2 f$ positive definite.
  - Ensure this by: Levenberg-Marquardt, or trust-region methods

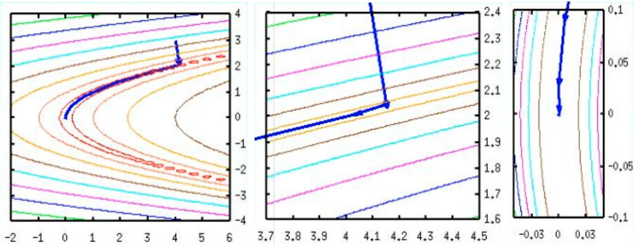
Example - Newton's method



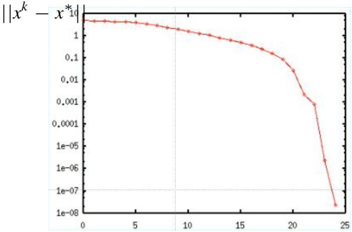
$$f(x,y) = \left((x - y^2)^2 + \frac{1}{100} \right)^{\frac{1}{4}} + \frac{1}{100} y^2$$

banana valley function

global minimum at $x = y = 0$

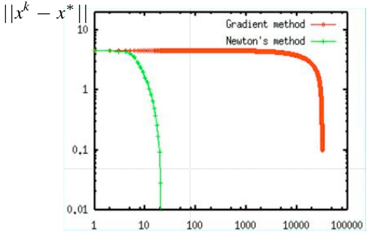


Example - Newton's method



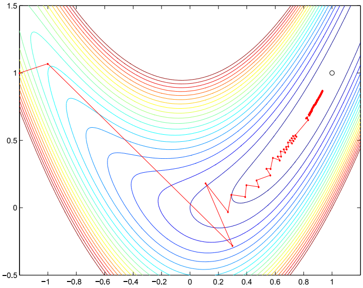
- Convergence of Newton's method:
  - less than 25 iterations for an accuracy of better than 10^{-7}!
  - convergence roughly linear for first 15-20 iterations since step length $\alpha^k \neq 1$
  - convergence roughly quadratic for last iterations with step length $\alpha^k = 1$

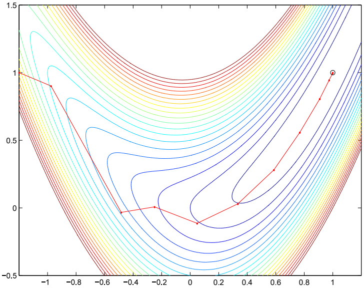
Comparison of the Steepest Descent and Newton's Methods



- For banana valley example:
  - Newton's method much faster than steepest descent method (factor 1000)
  - Newton's method superior due to higher order of convergence
  - steepest descent method converges too slowly for practical applications

Comparison of the Steepest Descent and Newton's Methods





Summary: Newton's Methods

Summary: Newton's Methods

- Fast (i.e. quadratic) local rate of convergence

Summary: Newton's Methods

- Fast (i.e. quadratic) local rate of convergence
- Scale-invariant w.r.t. linear transformations of the variables

Summary: Newton's Methods

- Fast (i.e. quadratic) local rate of convergence
- Scale-invariant w.r.t. linear transformations of the variables
- p^k is not well-defined if $\nabla^2 f(x^k)$ singular, p^k is not a descent if $\nabla^2 f(x^k)$ is not positive definite

Summary: Newton's Methods

- Fast (i.e. quadratic) local rate of convergence
- Scale-invariant w.r.t. linear transformations of the variables
- p^k is not well-defined if $\nabla^2 f(x^k)$ singular, p^k is not a descent if $\nabla^2 f(x^k)$ is not positive definite
- x^k can be attracted to local maxima or saddle points of f

Summary: Newton's Methods

- Fast (i.e. quadratic) local rate of convergence
- Scale-invariant w.r.t. linear transformations of the variables
- p^k is not well-defined if $\nabla^2 f(x^k)$ singular, p^k is not a descent if $\nabla^2 f(x^k)$ is not positive definite
- x^k can be attracted to local maxima or saddle points of f
- Very small neighbourhood of local convergence, Newton's method is not globally convergent

Summary: Newton's Methods

- Fast (i.e. quadratic) local rate of convergence
- Scale-invariant w.r.t. linear transformations of the variables
- p^k is not well-defined if $\nabla^2 f(x^k)$ singular, p^k is not a descent if $\nabla^2 f(x^k)$ is not positive definite
- x^k can be attracted to local maxima or saddle points of f
- Very small neighbourhood of local convergence, Newton's method is not globally convergent
- Line search, trust region

Algorithm 3: Quasi-Newton Methods

Algorithm 3: Quasi-Newton Methods

- In practice, evaluation of second derivatives for the hessian is expensive!
- Idea: approximate Hessian matrix $\nabla^2 f(x^k)$
- also ensure that the approximation B^k is positive definite

$$
\begin{aligned}
x^{k+1} &= x^k - (B^k)^{-1} \nabla f(x^k) \\
B^k &\approx \nabla^2 f(x^k)
\end{aligned}
$$

Algorithm 3: Quasi-Newton Methods

- In practice, evaluation of second derivatives for the hessian is expensive!
- Idea: approximate Hessian matrix $\nabla^2 f(x^k)$
- also ensure that the approximation B^k is positive definite

$$
\begin{aligned}
x^{k+1} &= x^k - (B^k)^{-1}\nabla f(x^k) \\
B^k &\approx \nabla^2 f(x^k)
\end{aligned}
$$

- methods are known as Quasi-Newton methods

Algorithm 3: Quasi-Newton Methods

- In practice, evaluation of second derivatives for the hessian is expensive!
- Idea: approximate Hessian matrix $\nabla^2 f(x^k)$
- also ensure that the approximation B^k is positive definite

$$x^{k+1} = x^k - (B^k)^{-1} \nabla f(x^k)$$
$$B^k \approx \nabla^2 f(x^k)$$

- methods are known as Quasi-Newton methods
- special case: steepest descent method: $B = I$

Quasi-Newton Methods

Quasi-Newton Methods

- different Quasi-Newton methods:

Quasi-Newton Methods

- different Quasi-Newton methods:
  - simplified Newton method: keep Hessian approximation B constant, e.g.

$$B^k \equiv \nabla^2 f(x^0)$$

Quasi-Newton Methods

- different Quasi-Newton methods:
  - simplified Newton method: keep Hessian approximation B constant, e.g.

$$B^k \equiv \nabla^2 f(x^0)$$

  - or: use same matrix B for several iterations:

$$\text{if } \frac{||\Delta x^{k+1}||}{\Delta x^k||} \geq \delta_{max} \text{ then update } B^k \equiv \nabla^2 f(x^k)$$

Quasi-Newton Methods

- different Quasi-Newton methods:
  - simplified Newton method: keep Hessian approximation B constant, e.g.

$$B^k \equiv \nabla^2 f(x^0)$$

  - or: use same matrix B for several iterations:

$$\text{if } \frac{||\Delta x^{k+1}||}{\Delta x^k||} \geq \delta_{max} \text{ then update } B^k \equiv \nabla^2 f(x^k)$$

  - or, even cheaper: use update-formulas for Hessian...

Quasi-Newton Update Formulas

Quasi-Newton Update Formulas

- Idea: Given an Hessian approximation B^k
- find a new approximation B^{k+1} that is "close" to B^k and satisfies

$$\nabla f(x^k) + B^{k+1}(x^{k+1} - x^k) = \nabla f(x^{k+1})$$

Quasi-Newton Update Formulas

- Idea: Given an Hessian approximation B^k
- find a new approximation B^{k+1} that is “close” to B^k and satisfies

$$\nabla f(x^k) + B^{k+1}(x^{k+1} - x^k) = \nabla f(x^{k+1})$$

- Advantages:
  - needs only evaluation of gradient $\nabla f(x^k)$ (same cost as steepest descent), but incorporates second order information
  - additional advantage: can update the inverse $(B^k)^{-1}$ directly

Quasi-Newton Update Formulas

- Idea: Given an Hessian approximation B^k
- find a new approximation B^{k+1} that is "close" to B^k and satisfies

$$\nabla f(x^k) + B^{k+1}(x^{k+1} - x^k) = \nabla f(x^{k+1})$$

- Advantages:
  - needs only evaluation of gradient $\nabla f(x^k)$ (same cost as steepest descent), but incorporates second order information
  - additional advantage: can update the inverse $(B^k)^{-1}$ directly
- Examples:
  - Symmetric Broyden-update
  - DFP-update (Davidon, Fletcher, Powell)
  - BFGS-update (Broyden, Fletcher, Goldfarb, Shanno) (most widely used)

Convergence Properties

Convergence Properties

- Quasi-Newton update methods converge locally superlinearly

  i.e. $||x^{k+1} - x^*|| \leq C_k ||x^k - x^*||, C_k \to 0$

Convergence Properties

- Quasi-Newton update methods converge locally superlinearly

  i.e. $||x^{k+1} - x^*|| \leq C_k ||x^k - x^*||, C_k \to 0$

- Quasi-Newton methods converge globally (i.e. from arbitrary initial point), if B^k remain positive definite and line search is used

Convergence Properties

- Quasi-Newton update methods converge locally superlinearly

  i.e. $||x^{k+1} - x^*|| \leq C_k ||x^k - x^*||, C_k \to 0$

- Quasi-Newton methods converge globally (i.e. from arbitrary initial point), if B^k remain positive definite and line search is used
- Quasi-Newton methods most popular method for medium scale problems

Constrained Optimization: SQP-method

Constrained Optimization: SQP-method

- Constrained problem:

$$
\begin{aligned}
\min f(x) \qquad & f : D \in \mathbb{R}^n \to \mathbb{R} \\
\text{s.t. } g(x) = 0 \qquad & g : D \in \mathbb{R}^n \to \mathbb{R}^l \\
h(x) \geq 0 \qquad & h : D \in \mathbb{R}^n \to \mathbb{R}^k
\end{aligned}
$$

Constrained Optimization: SQP-method

- Constrained problem:

$$\begin{aligned} \min f(x) \qquad & f : D \in \mathbb{R}^n \to \mathbb{R} \\ \text{s.t. } g(x) = 0 \qquad & g : D \in \mathbb{R}^n \to \mathbb{R}^l \\ h(x) \geq 0 \qquad & h : D \in \mathbb{R}^n \to \mathbb{R}^k \end{aligned}$$

- Idea: Consider successively quadratic approximations of the problem:

$$\begin{aligned} \min_p \qquad & f(x^k) + \nabla^T f(x^k) p + \frac{1}{2} p^T H^k p \\ \text{s.t.} \qquad & g(x^k) + \nabla g(x^k) p = 0 \\ & h(x^k) + \nabla h(x^k) p \geq 0 \end{aligned}$$

- $H^k \approx \nabla^2 L(x, \lambda, \mu)$

Constrained Optimization: SQP-method

Constrained Optimization: SQP-method

- if we use the exact Hessian of the Lagrangian

$$H = \nabla^2 L(x, \lambda, \mu)$$

  this leads to a Newton-method for the optimality conditions and feasibility (KKT-conditions)

Constrained Optimization: SQP-method

- if we use the exact Hessian of the Lagrangian

$$H = \nabla^2 L(x, \lambda, \mu)$$

  this leads to a Newton-method for the optimality conditions and feasibility (KKT-conditions)
- with update-formulas for H^k, we obtain quasi-Newton SQP-methods

Constrained Optimization: SQP-method

- if we use the exact Hessian of the Lagrangian

$$H = \nabla^2 L(x, \lambda, \mu)$$

  this leads to a Newton-method for the optimality conditions and feasibility (KKT-conditions)

- with update-formulas for H^k, we obtain quasi-Newton SQP-methods
- if we use appropriate update-formulas, we can have superlinear convergence

Constrained Optimization: SQP-method

- if we use the exact Hessian of the Lagrangian

$$H = \nabla^2 L(x, \lambda, \mu)$$

  this leads to a Newton-method for the optimality conditions and feasibility (KKT-conditions)

- with update-formulas for H^k, we obtain quasi-Newton SQP-methods
- if we use appropriate update-formulas, we can have superlinear convergence
- global convergence can be achieved by using a stepsize strategy based on (inexact) 1D minimization of an appropriate merit function, e.g. exact merit function

$$T(x) = f(x) + \sum_{eq} \gamma_i |g_i(x)| + \sum_{ineq} \beta_i |\min\{0, h_i(x)\}|$$

  with sufficiently large γ_i, β_i

Constrained Optimization: SQP-method

- if we use the exact Hessian of the Lagrangian

$$H = \nabla^2 L(x, \lambda, \mu)$$

  this leads to a Newton-method for the optimality conditions and feasibility (KKT-conditions)
- with update-formulas for H^k, we obtain quasi-Newton SQP-methods
- if we use appropriate update-formulas, we can have superlinear convergence
- global convergence can be achieved by using a stepsize strategy based on (inexact) 1D minimization of an appropriate merit function, e.g. exact merit function

$$T(x) = f(x) + \sum_{eq} \gamma_i |g_i(x)| + \sum_{ineq} \beta_i |\min\{0, h_i(x)\}|$$

  with sufficiently large γ_i, β_i
- alternatively, global convergence by trust region

Constrained Optimization: SQP-method

1. Start with $k = 0$, start value x^0 and $H^0 = I$
2. Compute $f(x^k), g(x^k), h(x^k), \nabla f(x^k), \nabla g(x^k), \nabla h(x^k)$
3. If x^k feasible and $||\nabla \mathcal{L}(x, \lambda, \mu)|| < \varepsilon$ then stop $\rightarrow$ convergence achieved
4. Solve quadratic problem (QP) and get p^k
5. Perform line search and get stepsize α^k
6. Iterate $x^{k+1} = x^k + \alpha^k p^k$
7. Update Hessian of the Lagrange function
8. $k = k + 1$, goto step 2

Solution of the Quadratic Program

- Unconstrained case:

$$\min_p g^T p + \frac{1}{2} p^T H p$$

- H must be positive definite, otherwise the optimization problem has no solution
- necessary optimality condition:

$$Hp + g = 0$$

- $=>$ use cholesky-method or cg-method to solve

Solution of the Quadratic Program

- equality constrained case:

$$\min_p g^T p + \frac{1}{2} p^T H p$$
$$Ap + a = 0$$

- necessary optimality condition (KKT-system): $\exists \lambda$ such that

$$\begin{pmatrix} H & A^T \\ A & 0 \end{pmatrix} = - \begin{pmatrix} g \\ a \end{pmatrix}$$

- matrix is indefinite, use range- or nullspace-method to solve

Solution of the Quadratic Program

- equality and inequality constrained case:

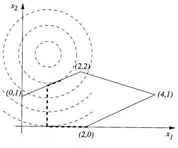
$$\min_p g^T p + \frac{1}{2} p^T H p$$
$$Ap + a = 0$$
$$Bp + b \geq 0$$

- use active-set-strategy
- aim: find out which inequalities are active at the solution and which not
- idea: solve a sequence of equality constrained QPs

$$\begin{aligned} \min_p \quad & g^T p + \frac{1}{2} p^T H p \\ & Ap + a = 0 \\ & B_i p + b_i = 0, \quad i \in W^k \end{aligned}$$

where W^k is a “guess” for an optimal active set

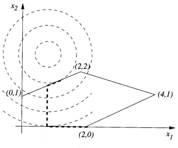
Active-Set Strategy: Example



- $p^0 = (2,0)^T$, $W^0 = \{3,5\}$, negative multiplier with respect to constraint 3, remove constraint 3

$$
\begin{aligned}
\min (p_1 - 1)^2 + (p_2 - 2.5)^2 \\
p_1 + 2p_2 + 2 &\geq 0 \\
-p_1 - 2p_2 + 6 &\geq 0 \\
-p_1 - 2p_2 + 2 &\geq 0 \\
p_1 &\geq 0 \\
p_2 &\geq 0
\end{aligned}
$$

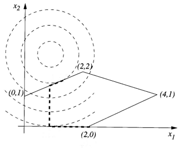
Active-Set Strategy: Example



- $p^0 = (2,0)^T$, $W^0 = \{3,5\}$, negative multiplier with respect to constraint 3, remove constraint 3
- $p^1 = (2,0)$, $W^1 = \{5\}$, no negative multipliers, solve QP, step length $\theta = 1$

$$\min(p_1 - 1)^2 + (p_2 - 2.5)^2$$
$$p_1 + 2p_2 + 2 \geq 0$$
$$-p_1 - 2p_2 + 6 \geq 0$$
$$-p_1 - 2p_2 + 2 \geq 0$$
$$p_1 \geq 0$$
$$p_2 \geq 0$$

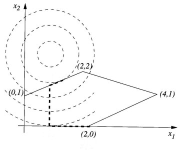
Active-Set Strategy: Example



$$\min (p_1 - 1)^2 + (p_2 - 2.5)^2$$
$$p_1 + 2p_2 + 2 \geq 0$$
$$-p_1 - 2p_2 + 6 \geq 0$$
$$-p_1 - 2p_2 + 2 \geq 0$$
$$p_1 \geq 0$$
$$p_2 \geq 0$$

- $p^0 = (2,0)^T$, $W^0 = \{3,5\}$, negative multiplier with respect to constraint 3, remove constraint 3
- $p^1 = (2,0)$, $W^1 = \{5\}$, no negative multipliers, solve QP, step length $\theta = 1$
- $p^2 = (1,0)$, $W^2 = \{5\}$, negative multiplier respect to constraint 5, remove constraint 5

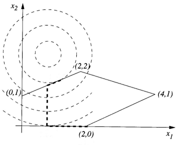
Active-Set Strategy: Example



$$\min (p_1 - 1)^2 + (p_2 - 2.5)^2$$
$$p_1 + 2p_2 + 2 \geq 0$$
$$-p_1 - 2p_2 + 6 \geq 0$$
$$-p_1 - 2p_2 + 2 \geq 0$$
$$p_1 \geq 0$$
$$p_2 \geq 0$$

- $p^0 = (2,0)^T$, $W^0 = \{3,5\}$, negative multiplier with respect to constraint 3, remove constraint 3
- $p^1 = (2,0)$, $W^1 = \{5\}$, no negative multipliers, solve QP, step length $\theta = 1$
- $p^2 = (1,0)$, $W^2 = \{5\}$, negative multiplier respect to constraint 5, remove constraint 5
- $p^3 = (1,0)$, $W^3 = \{\}$, no negative multipliers, solve QP, step length $\theta < 1$, constraint 1 gets active

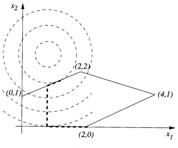
Active-Set Strategy: Example



$$\min (p_1 - 1)^2 + (p_2 - 2.5)^2$$
$$p_1 + 2p_2 + 2 \geq 0$$
$$-p_1 - 2p_2 + 6 \geq 0$$
$$-p_1 - 2p_2 + 2 \geq 0$$
$$p_1 \geq 0$$
$$p_2 \geq 0$$

- $p^0 = (2,0)^T$, $W^0 = \{3,5\}$, negative multiplier with respect to constraint 3, remove constraint 3
- $p^1 = (2,0)$, $W^1 = \{5\}$, no negative multipliers, solve QP, step length $\theta = 1$
- $p^2 = (1,0)$, $W^2 = \{5\}$, negative multiplier respect to constraint 5, remove constraint 5
- $p^3 = (1,0)$, $W^3 = \{\}$, no negative multipliers, solve QP, step length $\theta < 1$, constraint 1 gets active
- $p^4 = (1,1.5)$, $W^4 = \{1\}$, no negative multipliers, solve QP, step length $\theta = 1$

Active-Set Strategy: Example



$$\min (p_1 - 1)^2 + (p_2 - 2.5)^2$$
$$p_1 + 2p_2 + 2 \geq 0$$
$$-p_1 - 2p_2 + 6 \geq 0$$
$$-p_1 - 2p_2 + 2 \geq 0$$
$$p_1 \geq 0$$
$$p_2 \geq 0$$

- $p^0 = (2,0)^T$, $W^0 = \{3,5\}$, negative multiplier with respect to constraint 3, remove constraint 3
- $p^1 = (2,0)$, $W^1 = \{5\}$, no negative multipliers, solve QP, step length $\theta = 1$
- $p^2 = (1,0)$, $W^2 = \{5\}$, negative multiplier respect to constraint 5, remove constraint 5
- $p^3 = (1,0)$, $W^3 = \{\}$, no negative multipliers, solve QP, step length $\theta < 1$, constraint 1 gets active
- $p^4 = (1,1.5)$, $W^4 = \{1\}$, no negative multipliers, solve QP, step length $\theta = 1$
- $p^5 = (1.4,1.7)$, $W^5 = \{3,5\}$ all multipliers positive $\rightarrow$ solution

Alternative Methods for Constrained Optimization

Alternative Methods for Constrained Optimization

- Penalty and barrier methods

Alternative Methods for Constrained Optimization

- Penalty and barrier methods
- Augmented lagrangian methods

Alternative Methods for Constrained Optimization

- Penalty and barrier methods
- Augmented lagrangian methods
- Interior point methods for inequality constrained problems

Summary: Optimization Methods Overview

Summary: Optimization Methods Overview

- Optimization problems can be (un)constrained, (non)convex, (non)linear, (non)smooth, continuous/integer,(in)finite dimensional, ...

Summary: Optimization Methods Overview

- Optimization problems can be (un)constrained, (non)convex, (non)linear, (non)smooth, continuous/integer,(in)finite dimensional, ...
- Here: try to find local minima of smooth nonlinear problems: $\nabla f(x) = 0$ (resp. $\nabla \mathcal{L}(x, \lambda, \mu) = 0, g(x) = 0, h_{active} = 0$)
- Starting at an initial guess x^0 , most methods iterate $x^{k+1} = x^k + \alpha^k p^k$ with search direction p^k and step length α^k

Summary: Optimization Methods Overview

- Optimization problems can be (un)constrained, (non)convex, (non)linear, (non)smooth, continuous/integer,(in)finite dimensional, ...
- Here: try to find local minima of smooth nonlinear problems: $\nabla f(x) = 0$ (resp. $\nabla \mathcal{L}(x, \lambda, \mu) = 0, g(x) = 0, h_{active} = 0$)
- Starting at an initial guess x^0 , most methods iterate $x^{k+1} = x^k + \alpha^k p^k$ with search direction p^k and step length α^k
- Search direction can be chosen differently
  - steepest descent (simple, but slow and rarely used in practice)
  - Newton's method (very fast if Hessian cheaply available)
  - Quasi-Newton methods (cheap, fast, and popular, e.g. BFGS)
  - SQP methods for constrained optimization
  - CG method (good for very large scale problems)

Summary: Optimization Methods Overview

- Optimization problems can be (un)constrained, (non)convex, (non)linear, (non)smooth, continuous/integer,(in)finite dimensional, ...
- Here: try to find local minima of smooth nonlinear problems: $\nabla f(x) = 0$ (resp. $\nabla \mathcal{L}(x, \lambda, \mu) = 0, g(x) = 0, h_{active} = 0$)
- Starting at an initial guess x^0 , most methods iterate $x^{k+1} = x^k + \alpha^k p^k$ with search direction p^k and step length α^k
- Search direction can be chosen differently
  - steepest descent (simple, but slow and rarely used in practice)
  - Newton's method (very fast if Hessian cheaply available)
  - Quasi-Newton methods (cheap, fast, and popular, e.g. BFGS)
  - SQP methods for constrained optimization
  - CG method (good for very large scale problems)
- Other methods: direct search, simulated annealing, genetic algorithms, ... useful for special optimization problems

References

- J. Nocedal, S. Wright: Numerical Optimization, Springer, 1999
- P. E. Gill, W. Murray, M. H. Wright: Practical Optimization, Academic Press, 1981
- R. Fletcher, Practical Methods of Optimization, Wiley, 1987
- D. E. Luenberger: Linear and Nonlinear Programming, Addison Wesley, 1984

Thank you for your attention!