Compact Course on
Nonlinear Optimization and Inverse Problems

Prof. Dr. Volker Schulz
Universität Trier
FB IV - Mathematik

Stuttgart, March 9-11, 2005
1 Finite dimensional optimization

1.1 Introduction: Basic theory of nonlinear optimization

Figure 1: Mathematical tasks in engineering simulation & optimization
1.1.1 Optimization problems in practical applications:

**Definition 1.1.1 (terminology)**

For a set $X$ and a function $f : X \rightarrow \mathbb{R}$ define the notation

$$\hat{x} \in X \text{solves } \min_{x \in X} f(x) :\iff \hat{x} \in X \text{solves } \min f(x) \text{ s.t. } x \in X$$

$$\iff \hat{x} = \arg \min_{x \in X} f(x)$$

$$\iff f(\hat{x}) \leq f(x), \forall x \in X$$

1.1.2 Problem classes and practical examples

**Process optimization**:

typical situation: multiple objective criteria

$$\varphi_1(x), \ldots, \varphi_n(x)$$ (multi-criteria optimization)

approach: choose one criterion $\varphi_k$

$$\min_{x \in X} \varphi_k(x) \text{ s.t. } \varphi_i(x) \leq \bar{\varphi}_i, \forall i \neq k, \bar{\varphi}_i \text{ given}$$

usually, there arise additional restrictions in the form

$$c_E(x) = 0 \quad c_E : X \rightarrow Y \text{ (equality constraints)}$$

$$c_I(x) \geq 0 \quad c_I : X \rightarrow Z \text{ (inequality constraints)}$$

for $X, Y, Z$ Vector spaces

Examples:

- shape optimization for turbine blades
- optimal process control in chemistry
- optimal path planning for robots
Parameter estimation / Inverse problems

\[ X \text{ state space} \]
\[ P \text{ parameter space} \]

coupled by model equation \[ c(x, p) = 0 \]
\[ c : X \times P \rightarrow Y \]

Observation function \[ \psi : X \times p \rightarrow \Psi \]
Data in the form of real observations \( \bar{\psi} \)

Problem:
Recover "true" \( p \) corresponding to \( \psi(x, p) \approx \bar{\psi} \)
Typical approach: output-least-squares optimization
\[
\min \| \psi(x, p) - \bar{\psi} \|^2 \\
\text{s.t. } c(x, p) = 0
\]

there may also be additional restrictions

Typical distinction in terminology (not sharp!)

\[ \dim(P) < \infty : \text{"Parameter estimation"} \]
\[ \text{else : "Inverse problem"} \]

Examples:

- Parameter estimation in multiphase flow
- Inverse modeling in groundwater flow
Optimum experimental design

Process model \[ c(x, p; q) = 0 \]

\[ q \in Q \text{ additional design parameter} \]

Objective function: identifiability \( \Phi \)

\[
\begin{align*}
\min \quad & \Phi(x, p; q) \\
\text{s.t.} \quad & c(x, p; q) = 0 \quad \text{(model)}
\end{align*}
\]

& additional restriction

\( p \) is assumed as given

Example:
shape optimization for Bingham measurement devices

1.1.3 Focus

Discrete optimization: \( X \subset \mathbb{Z}^n \)

\( \rightarrow \) authors: Wolsey, Grötschel etc.

here we assume mostly \( X \subset V \) vector space (Banach-/Hilbert-)

discrete optimization typically require the solution of continuous subproblems

Global optimization:
here only considered in the following section in connection with stochastic optimization techniques

\( \rightarrow \) authors for global optimization: Floudas, Pardalos, Horst etc.

we investigate mainly local optimization techniques (typically, this is a subtask of global optimization approaches)
Definition 1.1.2 (local solution)
\( \hat{x} \) is called a (strict) local solution to \( \min_{x \in X} f(x) \), if there exists an open neighbourhood \( U \subset X \) with
\[
x^* \in U \text{ and } f(\hat{x}) \leq f(x) \quad \forall x \in U \\
(f(\hat{x}) < f(x) \quad \forall x \in U \setminus \{\hat{x}\})
\]

Remark:
In convex optimization problems, all local solutions are also global solutions.
\[
\left( \text{convex opt. probl. } : \iff f \text{ is convex} \quad \forall x \text{ is convex} \right)
\]

Stochastic optimization
Often, input variables are not known precisely or vary stochastically.
\[\rightarrow\] szenario optimization
\[\rightarrow\] optimization of expected value
authors: Birge, Kall, Henrion etc.

Here we assume all variables being deterministic; only exception: observations/measurements

1.1.4 Local theory

Definition 1.1.3 Let \( X \) be a Hilbert space with scalar product \((\cdot,\cdot)\) and \( f : X \to \mathbb{R} \) be Fréchet-differentiable.

Then, we define \( \nabla_xf(x) \in X \) as the Riesz-representation of \( f'(x) \in X^* \), where \( X^* \) denotes the dual space of \( X \), i.e.
\[
(\nabla_xf(x), v) = f'(x)v, \quad \forall v \in X
\]
Example: $X = \mathbb{R}^n$, $(x, y) := x^\top Ay$, $A > 0$ sym.

Then

$$f'(x)v = \begin{pmatrix} f'_1(x) \\ \vdots \\ f'_n(x) \end{pmatrix}^\top v = \begin{pmatrix} f'_1(x) \\ \vdots \\ f'_n(x) \end{pmatrix}^\top A^{-1} Av = \begin{bmatrix} A^{-1} f'(x)^\top \end{bmatrix}^\top Av$$

$$\Rightarrow \nabla_x f(x) = A^{-1} f'(x)^\top$$

- usually $A = I$ in $\mathbb{R}^n$, but varying metric will be important later.

**NOTICE:** The gradient is only defined after a scalar product is specified!

Analogously $\text{Hess}_x f := \nabla^2_x f$ is defined by

$$(\nabla^2_x f(x)v, w) = f''(x)(v, w), \forall v, w \in X$$

and thus in $(\mathbb{R}^n, (., .)_A)$:

$$\nabla^2_x f(x) = A^{-1} \left( \frac{\partial^2 f}{\partial x_i \partial x_j} (x) \right)_{ij}$$

Geometric interpretation:

consider the level set $X_f := \{x \in U | f(x) = f(\hat{x}) \ \forall \ x \in U \subset X \}$

The tangent space of that set in $\hat{x}$ is

$$TX_f = \{v \in X | f'(\hat{x})v = 0 \}$$

Now, for $v \in TX_f$ we have

$$(\nabla_x f(\hat{x}), v) = f'(\hat{x})v = 0$$

i.e. $\nabla_x f(\hat{x}) \perp TX_f$

**Theorem 1.1.4** (necessary conditions for unbounded optimization)

$(X, (., .))$ Hilbert space, $f : X \rightarrow \mathbb{R}$ Fréchet-differentiable; if $\hat{x}$ is a local solution to $\min_{x \in X} f(x)$ then

a) $\nabla_x f(\hat{x}) = 0$ ("necessary cond. of 1st order")
b) if $f''(\hat{x})$ exists, then $f''(\hat{x})$ is positive semidefinite, i.e.

$$(v, \nabla^2_x f(\hat{x})v) \geq 0 \quad \forall \ v \in X$$

("necessary cond. of 2nd order")

Proof: Define $\varphi : [-\varepsilon, \varepsilon] \to \mathbb{R}$

$$t \mapsto f(\hat{x} + t \cdot v), \ v \in X$$

From elementary calculus we know that if $\varphi$ is minimal at $t = 0$, then $\varphi'(0) = 0, \varphi''(0) \geq 0$

$$0 = \varphi'(0) = \frac{d}{dt} \bigg|_{t=0} f(\hat{x} + tv) = f'(\hat{x})v = (\nabla_x f(\hat{x}), v)$$

$$0 \leq \varphi''(0) = \frac{d^2}{dt^2} \bigg|_{t=0} f(\hat{x} + tv) = f''(\hat{x})(v, v) = (v, \nabla^2_x f(\hat{x})v)$$

$v$ is arbitrary $\Rightarrow$ proposition

**Theorem 1.1.5** *(sufficient condition)*

If the assumptions of Th. 1.1.4 a) and b) are satisfied and $\nabla_x, f(\hat{x}) = 0$ and $f''(\hat{x})$ is bounded away from zero, i.e.

$$f''(\hat{x})(v,v) = (v, \nabla^2_x f(\hat{x})v) \geq m \|v\|^2 \forall v \in X \text{ for some } m = 0$$

then $x^*$ is a strict local solution. \(\|v\|^2 := (v,v)\)

**Example:** $\mathbb{R}^n, (x,y) = x^\top A y$

$$0 = \nabla_x f(\hat{x}) = A^{-1} f'(\hat{x}) \Rightarrow f_i(\hat{x}) = 0, \ \forall \ i = 1, \ldots, n$$

For second order sufficient condition, it is enough to know that Hess$f(\hat{x})$ is positive definite; then we can use $m :=$ smallest eigenvalue of Hess $f(\hat{x})$

2D-Example: $f(x_1, x_2) = x_1^2 + 4x_2^2$

Usage of the scalar product \((x,y) := x^\top \begin{bmatrix} 1 & 0 \\ 0 & 1/4 \end{bmatrix} y\) corresponds to a rescaling $\bar{x} := \begin{bmatrix} 1 & 0 \\ 0 & 1/2 \end{bmatrix} x$
Theorem 1.1.6 (equality constrained problems)

Consider the optimization problem

\[ \min f(x) \]
\[ \text{s.t. } c(x) = 0 \]

where \( X \) is a Hilbert space, \( Y \) a Banach space and \( f : x \to \mathbb{R}, c : X \to Y \) are twice Fréchet-differentiable.

For \( \lambda^* \in Y^* \) (dual space of \( Y \)) we define the Lagrangian

\[ \mathcal{L}(x, \lambda^*) := f(x) - \lambda^* c(x) \]

If \( \hat{x} \) is a local optimal solution and \( c'(\hat{x}) \) is surjective (regular point) then there exists \( \hat{\lambda}^* \in Y^* \) such that

a) \( \nabla_x \mathcal{L}(\hat{x}, \hat{\lambda}^*) = 0 \)

b) \( \mathcal{L}''(\hat{x}, \hat{\lambda}^*)(v, v) = (v, \nabla_x^2 \mathcal{L}(\hat{x}, \hat{\lambda}^*)v) \geq 0, \forall v \in \mathcal{N}(c'(\hat{x})) \)

where \( \mathcal{N}(c'(\hat{x})) := \{ w \in x | c'(\hat{x})w = 0 \} \)

c) if conversely a) is satisfied and

\( \mathcal{L}''(\hat{x}, \hat{\lambda}^*)(v, v) \geq m \| v \|^2, \forall v \in \mathcal{N}(c'(\hat{x})) \) for some \( m > 0 \)

then \( \hat{x} \) is a strict local minimum

Remark:

If \( v \in \mathcal{N}(c'(\hat{x})) \) then from a) we conclude

\( (\nabla_x f(\hat{x}), v) = (\nabla_x (\lambda^* c(\hat{x})), v) = \lambda^* c'(\hat{x}) v = 0 \)

I.e. \( \nabla_x f(\hat{x}) \perp \mathcal{N}(c'(\hat{x})) \)

If we are in the situation of the so-called separability framework, i.e. \( x = (z, p) \in Z \times P \) with

\[ \min f(z, p) \]
\[ \text{s.t. } c(z, p) = 0, \frac{\partial c}{\partial z} \text{ invertible} \]

then we see that \( \mathcal{N}(c'(z, p)) = \)

\[ = \left\{ \begin{bmatrix} -c^{-1}_z c_p \\ I \end{bmatrix} p \mid p \in P \right\} \]

and the first order condition is with the definition \( T(z, p) := \begin{bmatrix} -c^{-1}_z c_p \\ I \end{bmatrix} : P \to Z \times P \)
equivalent to \( g(\hat{z}, \hat{p}) := T(\hat{z}, \hat{p})^* f'(\hat{z}, \hat{p}) = 0 \) where \( g(\hat{z}, \hat{p}) \) denotes the reduced derivative and \( T^* \) the adjoint mapping to \( T \).

Again, by use of a scalar product in \( P, (.,.)_P \) we can define the Riesz-representation \( \gamma(z, p) \) of \( g(z, p) \) and call it the reduced gradient.

In the separability framework, also the 2nd order conditions get a simpler formulation, if we define the reduced Hessian \( B := T^* \mathcal{L}'' T \)

necess. cond.: \( B \) pos. semidef.
suffic. cond. : \( B(p, p) > m \| p \|_P^2, \quad \forall \ p \in P, \exists m > 0 \)

It can be shown that the reduced gradient \( \gamma \) and the reduced Hessian \( B \) are exactly the gradient and the Hessian of the black-box reformulation

\[
\min_p f(z(p), p)
\]

where \( z(p) \) is defined via implicit function theorem by

\[
c(z, p) = 0
\]

Finally, we formulate now conditions for problems with inequality constraints. However, the theory for infinite-dimensional spaces there, is too complicated to be covered by this compact course (refer to the term ”optimal control with state and control constraints”). Here we restrict the presentation to the finite-dimensional case.

Consider for \( x = \mathbb{R}^n \) the optimization-problem

\[
\min_{x \in x} f(x) \quad \text{s.t.} \quad c(x) = 0 \quad c : \mathbb{R}^n \to \mathbb{R}^m, m < n \quad \text{\( d(x) \geq 0 \) } d : \mathbb{R}^n \to \mathbb{R}^l
\]

Define: \( I(x) := \{ i \in \{1, \ldots, l\} \mid d_i(x) = 0 \} \) ”active set”

\( I^\perp(x) := \{1, \ldots, l\} \setminus I(x) \) ”inactive set”

\( \text{LICQ: ( Linear Independence Constraint Qualification)} \)

\( x \) is called regular according to LICQ, if the set \( \{c_1'(x), \ldots, c_m'(x), d_1'(x), \ldots, d_s'(x)\} \) is linear independent \( \{\{i_1, \ldots, i_s\} = I(x)\} \)

Remark: there are many more constraint qualifications ...
Theorem 1.1.7 If LICQ is satisfied for the problem above, and \( \hat{x} \) is a local solution, then there is 
\[ \lambda \in \mathbb{R}^m \text{ and } 0 \leq \mu \in \mathbb{R}^l \text{ such that} \]

\begin{align*}
a) \quad & f'(\hat{x}) - \lambda^\top c'(\hat{x}) - \mu^\top d'(\hat{x}) = 0 \quad (1\text{st order nec.}) \\
b) \quad & \left( f''(\hat{x}) - \lambda^\top g''(\hat{x}) - \mu^\top d''(\hat{x}) \right)(\xi, \xi) \geq 0 \quad (2\text{nd order nec.}) \\
c) \quad & \mu^\top d(\hat{x}) = 0 ("\text{complementarity}") \\
c) \quad & \text{sufficient condition of 2nd order with } " > 0" \text{ in } b) 
\end{align*}

Stability issues:

If the problem data are perturbed by \( \varepsilon \), then there should exist a continuous mapping 
\[ \varepsilon \mapsto \hat{x}(\varepsilon). \]

Otherwise we would not trust whatever we compute numerically.

It can be shown, that stability in that sense is satisfied, if the necessary conditions of 2nd order are satisfied. Furthermore, the active set can be shown to be constant in that case for \( \varepsilon \) small enough.

Literature

Nocedal/Wright: Numerical optimization , Springer 1999
Wouk: A course of applied functional Analysis, Wiley, 1979
1.2 Computational methods for unconstrained optimization

In this section, we discuss optimization strategies, which require only the evaluation of the cost functional. Therefore, they are highly flexible and easily implemented. The price that one typically pays for that feature is the large computational effort compared to algorithms which use higher order information. These methods are also favorable in the presence of multiple discontinuities. There is an abundance of such methods with names like direct search, simplex, polytope algorithm, pattern search, response surface fitting, tabu search, genetic algorithms, evolution algorithms, simulated annealing, ...

Here, we consider only the most popular 3 of them. For the rest we refer to the literature.

1.2.1 Direct search methods

These methods are characterized as progressing through the variable space by the usage of function values only.

For an overview, c.f. Lewis/Torczon/Trasset 2000.

Simplest method: method of local variation

0 select $\delta_1, \ldots, \delta_n \in \mathbb{R}^n$, starting point $\bar{x} \in \mathbb{R}^n$

1 For $i = 1, \ldots, n$ do
   if $f(\bar{x} \pm \delta_i e_i) < f(\bar{x})$, $\bar{x} := \bar{x} \pm \delta_i e_i$
   and do

2 if $\bar{x}$ not changed: $\delta_i := \frac{1}{2} \cdot \delta_i$, $\forall i = 1, \ldots, n$
   continue with step 1 until convergence.

Theorem
(Polak 1971) Any accumulation point of the algorithm above satisfies $\nabla f(x^*) = 0$

Remarks:

a) This is an example of a pattern search method

b) The continuous variant, coordinate descent method, where in each coordinate direction an almost exact line-search is performed, is known to lead to infinite cycles for certain example problems! (Powell 1973)
Most popular: Polytope algorithm (also called "simplex" method) by Nelder/Mead 1965

For the problem

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

we sketch the basic idea in 2D

The general algorithm works in the following way

0. Start with an initial polytope \( \{x_1, \ldots, x_n\} \) of points which do not lie on a common affine hyperplane

1. Compute the centroid \( c = \frac{1}{n} \sum_{i=1}^{n} x_i \)

2. Determine \( x_w \) with \( f(x_w) \geq f(x_i), \forall i \neq w \)
   and \( x_{sw} \) with \( f(x_{sw}) \geq f(x_i) \forall i \neq w, i \neq sw \) (second worst)
   and \( x_b \) with \( f(x_b) \geq f(x_i), \forall i \neq b \)

3. Compute reflection \( x_r := c + \alpha (c - x_w) (\alpha > 0) \)

4. consider 3 cases:
   a) if \( f(x_b) \leq f(x_r) \leq f(x_{sw}) \)
      replace \( x_w \) by \( x_r \) (some improvement is obtained)
   b) if \( f(x_r) < f(x_b) \) conclude "good" direction and expand the polytope
      \( x_e := c + \beta (x_r - c) (\beta > 1) \)
      Now if \( f(x_e) < f(x_r) \) replace \( x_w \) by \( x_e \) otherwise replace \( x_w \) by \( x_r \)
   c) if \( f(x_r) > f(x_{sw}) \), polytope is too large \(\Rightarrow\)
      shrinking:
      \[
      x_c = \begin{cases} 
        c + \gamma(x_w - c) & \text{if } f(x_r) \geq f(x_w) \\
        c + \gamma(x_r - c) & \text{if } f(x_r) < f(x_w) 
      \end{cases}
      \]
      \(0 < \gamma < 1\)
      if \( f(x_c) < \min \{ f(x_r), f(x_w) \} \) contraction successful: \( x_w \) is replaced by \( x_c \)
      else perform another contraction-step

5. continue with step 1 until convergence.
Remark:

- Occasionally, the most recent polytope is discarded and replaced by a regular polytope (“restarting”)
- various modifications are used
- implementations
  - E04CCF from NAG-Library
  - subplex from netlib
  - hooke.c from netlib
- theoretical analysis in Lagarias/Reeds/Wright\(^2\) 1998
- surprisingly robust method even for mildly discontinuous problems although no similar result as above can be shown
- problems with instable performance are known

Literature:

Nelder/Mead: A simplex method for function minimization, Comput. J. 7 (1965) 308-313
Lagarias/Reeds/Wright/Wright: Convergence properties of the Nelder-Mead simplex method in low dimensions, SIAM J. Optim. 9 (1988) 112-147
Lewis/Torczon/Trosset: Direct search methods: then and now, J. CAM 129 (2000) 191-207

A promising recent deterministic global optimization technique can be found in

1.2.2 Genetic algorithms

Heuristic approach towards global optimization

Basic idea: mimic biological evolution processes algorithmically

Biological paradigm: Individuals within a generation mutate and mate and generate new individuals

generic algorithm:

0 Initial population
1 Evaluation
2 Selection: take the best p% from population
3 Crossover: Randomly selected couples exchange parts of information and produce offspring
4 Mutation: Information Bits are randomly changed and monadic offspring is produced

* typical data type of variables encoding ”chromosomes” in a binary string
* also possible representation of chromosomes $\in \mathbb{R}^n$ as arrays of real numbers
  → mutation: choice of random number in i-th component

Properties:

* GA can be interpreted as Markov processes and as dynamical processes and analysed in these frameworks
* For a guarantee of convergence to the optimum one has to apply special strategies like reducing the mutation rate slowly while increasing the strength of selection
* The effect of crossover is not yet completely understood but it typically improves performance
* There is a large variety in implementations with lots of tuning parameters
* Good implementations are always problem specific
* Typical performance numbers: 500 Generations with 30 individuals
* GA software: www.geneticprogramming.com/ga/GAsoftware.html
Literature:
the GA archive: http://www.aic.nrl.navy.mil/galist/# archive

(Theory in the framework of stochastic dynamical systems)
Chambers: The practical handbook of genetic algorithms applications, Chapman & Hall 2001
Goldberg: Genetic algorithms in search, optimization and machine learning, Addisson-Wesley, 1989
Reeves/Rowe: Genetic algorithms - principles and perspectives, a guide to GA theory, Kluwer 2003

On GA and SA:

Pham/Karaboga: Intelligent optimization techniques, Springer 1952
Ansari/Hou: Computational intelligence for optimization, Kluwer 1997
Rayward-Smith/Osman/
Reeves/Smith: Modern heuristic search methods, Wiley 1996
(contains also a chapter on integrating local search deterministic search into genetic algorithms)
Paradalos/Romeign/
Tuy: Recent developments and trends in global optimization, J. CAM 124 (2000) 209-228
1.2.3 Simulated Annealing

Physical analogy:
cooling of material in a heat bath ("annealing"); if solid material is heated past its melting point and then cooled back into a solid state, the structural properties of the cooled solid depend on the rate of cooling.

Basic algorithm:

0 initial iterate
1 change current iterate randomly $x^{new}$
2 if $f(x^{new}) \leq f(\bar{x})$ accept $x^{new}$
   else chose current temperature $T$ in a decreasing way
   and accept $x^{new}$, if
   
   $$\text{Boltzman-factor} := \exp\left(\frac{[f(\bar{x}) - f(x^{new})]}{T}\right) < \text{random} \in [0, 1]$$

Boltzman-factor is the probability of acceptance of the corresponding iterate, it is monotonically decreasing in $T$.

Typical data representation as binary string

Theoretical result: $P(x_n = x_{opt}) \underset{n \to \infty}{\to} 1$

Theoretical analysis is based on Markov chains and can be found e.g. in

Literature:

van Laarhoven/Aarts: Simulted annealing, theory and applications,
Kluwer 1988
1.2.4 Additional contraints

Typical optimization problems are accompanied by constraints in the form
\[
\min f(x) \\
\text{s.t. } c(x) = 0 \\
d(x) \geq 0
\]

For the application of algorithms from 1.2.1-1.2.3 one has to reformulate this in the form of (a sequence of) unbounded optimization problems
\[
\min \varphi(x)
\]

Approaches:

(1) Quadratic penalty method: solve successively the problems
\[
QPM_k \min_x f(x) + \frac{1}{2\mu_k} c(x)^\top c(x) + \frac{1}{2\mu_k} \sum_{i=1}^l \left( \min\{0, d_i(x)\} \right)^2
\]
with \( \mu_k \to 0 (k \to \infty) \)
use the solution from problem \( QPM_{(k-1)} \) as inital data for problem \( QPM_k \)

Properties:

* \( x_k \to \hat{x} (k \to \infty) \), every limit point is a local solution
* For \( \mu_k \) decreasing the problems become more and more difficult - the valleys are getting deeper and deeper, the local models are becoming ill-conditioned. Therefore, ever better initial guesses are necessary
* Typical \( \mu \)-Sequence: \( \mu_k = \beta \cdot \mu_{k-1}, \beta \in (0,1) \)

(2) Logarithmic barrier method:
Problem
\[
\min f(x) \\
\text{s.t. } c(x) \geq 0
\]
strict interior \( \mathcal{F}^0 := \{ x | d_i(x) > 0, \forall i \} \)
logarithmic barrier function
\[
P(x; \mu) = f(x) - \mu \sum_i \log d_i(x)
\]
Properties:
- is infinite everywhere except $F^o$
- smooth inside $F^o$ (if $f$ and $c$ are smooth)
- approaches $\infty$ as $x$ approaches the boundary of $F^o$
- $\mu$ is called ”barrier parameter”

Method:
$\mu$ is decreased towards zero $\mu \downarrow 0$

* analogously as above, unconstrained subproblems are becoming ill-conditioned, since typically the iterates approach the boundary
* this is the starting point for interior-point-methods for linear and nonlinear programming, which are mainly used for convex problems

(3) $l_1$ exact penalty function:
\[
\varphi_1(x; \mu) = f(x) + \frac{1}{\mu} \sum_i |c_i(x)| - \frac{1}{\mu} \sum_j \min\{0, d_i(x)\}
\]

Properties:
* there exists $\mu_0 > 0$ such that the solution of $\min_x \varphi_1(x; \mu)$ is the solution of the original problem for all $0 < \mu < \mu_0$ ("exact")
* $\mu_0$ is difficult to determine
* again subproblems are becoming difficult for decreasing $\mu$
* $\varphi_1(x; \mu)$ is nonsmooth

(3b) Han-Powell variant
\[
\Phi^{HP}(x) = f(x) + \sum_i \nu_i |c_i(x)| - \sum_j \nu_j \min\{0, d_i(x)\}
\]
with $\nu_i > |\hat{\lambda}_i|$, $\hat{\lambda}_i$ adjoint variables at the solution
* also exact penalty functions
* estimations for $\nu_i$ are hard to obtain without higher order information
(4) Augmented Lagrangian method

Problem

\[ \min_{x} f(x) \]
\[ \text{s.t. } c(x) = 0 \]

\[ \mathcal{L}_A(x, \lambda, \mu) := f(x) - \lambda^\top c(x) + \frac{1}{2\mu} c(x)^\top c(x) \]

\[ \mu \searrow 0 \quad , \lambda \text{ adjoint variables at solution} \]

Properties:

* exact penalty function: \( \mu \) needs not being decreased below a certain value \( \bar{\mu} > 0 \)
* estimations for \( \lambda \) from necessary conditions for fixed \( \mu \)
\[ \nabla_x \mathcal{L}(x, \lambda, \mu) = \nabla f(x) - c_x^\top \left( \lambda - \frac{1}{\mu} c(x) \right) = 0 \]
\[ \Rightarrow \lambda - \frac{1}{\mu} c(x) \longrightarrow \hat{\lambda} (\mu \to 0) \]
\[ \Rightarrow \text{update for } \lambda : \lambda^{k+1} := \lambda^k - \frac{1}{\mu} c(x_k) \]
* that is the basis for the code LANCELOT (Toint)
* Inequality constraints \( d_i(x) \geq 0 \) are reformulated by slack variables with bound constraints
\[ d_i(x) - s_i = 0 \]
\[ s_i \geq 0 \]

bound constraints are treated explicitly in LANCELOT
* ill-conditioning is a far less harmful problem compared with the other approaches

(5) Specialties for GA

* Penalty approach as above is often difficult to implement
* Repair strategy:
  - accept infeasible solution, but ”repair” it before inserting it in the population
  - this strategy is also applied when using GA offspring as the springboard for a deterministic local search.
  - should the original offspring inserted or the repaired one ?

  experimental evidence is reported, that one should decide that stochastically and introduce the repaired version only with a probability of 5-10%.
* use infeasibilities as secondary objective in a multiple objective GA
  (e.g. selection for reproduction based on feasibility
   selection for deletion based on domination w.r.t. two objectives)

**Literature**

Nocedal/Wright: Numerical optimization, Springer 1999
Reeves/Rowe: Genetic algorithms - principles and perspectives,
a guide to GA theory, Kluwer 2003
1.3 Gradient based (constrained) optimization

1.3.1 The steepest descent method

**Theorem 1.3.1** Let $X$ be a Hilbert space with scalar product $(\cdot, \cdot)$ and $f : X \to \mathbb{R}$ be Fréchet-differentiable. Then $-\nabla_x f(x_0)/\|\nabla_x f(x_0)\|$ is the direction of steepest descent.

**Proof:** "steepest descent" means direction in which $f$ decreased most rapidly. The size of decrease in an arbitrary direction $d$ is

$$\frac{d}{d\alpha} f(x_0 + \alpha d) = f'(x_0) d$$

Now we normalize all candidates, i.e. require $\|d\| = 1$ and look for the minimum of

$$\min_d f'(x_0) d \quad \text{s.t.} \quad \|d\| = (d, d) = 1$$

We apply the necessary conditions from section 1.1 and formulate the Langrangian $L(d, \lambda) = f'(x_0) d - \lambda \cdot (d, d), \lambda \in \mathbb{R}$

Thus we obtain the condition

$$\nabla_d L(d, \lambda) = \nabla_x f(x_0) - 2 \cdot \lambda \cdot d = 0 \Rightarrow d = \frac{1}{2\lambda} \nabla_x f(x_0)$$

From the constraint we conclude $\lambda = \pm 2\|\nabla_x f(x_0)\|$

and therefore $d = -\frac{1}{\|\nabla_x f(x_0)\|} \cdot \nabla_x f(x_0)$, since the Hessian of the Lagrangian is only positive for $\lambda < 0$.

**Remark:**
Note that the gradient and therefore the direction of steepest descent depend on the choice of the scalar product in $X$. But for a fixed scalar product, the negative normalized gradient is always the direction of steepest descent.
Graphical interpretation:

We can only profit from the steepest descent direction as long as the objective is decreased.

Ideal steepest descent algorithm (≡ gradient method):

1. At iterate \( x_k \) compute \( d_k := -\nabla_x f(x_k) \)
2. If \( \|d\| < \varepsilon \) → STOP
3. Solve the 1D minimization problem
   \[ \alpha_k := \arg \min_{\alpha} f(x_k + \alpha d_k) \]
4. set \( x_{k+1} := x_k + \alpha d_k , ~ k := k + 1 \), continue with step 1

Lemma 1.3.2 (Zig-Zagging)

For succeeding iterations of the algorithm above, the following condition holds

\[ \nabla_x f(x_{k+1}) \perp \nabla_x f(x_k) \]

Proof:

\[ \alpha_k = \arg \min_{\alpha} f(x_k - \alpha \nabla_x f(x_k)) \]

\[ \Rightarrow 0 = \frac{d}{d\alpha} \bigg|_{\alpha = \alpha_k} f(x_k - \alpha \nabla_x f(x_k)) = \]

\[ = f'(x_k - \alpha_k \nabla_x f(x_k)) \nabla_x f(x_k) = \]

\[ = \left( \nabla_x f(x_{k+1}) , \nabla_x f(x_k) \right) \]

\[ \square \]

Theorem 1.3.3 (local convergence in \( \mathbb{R}^n \))
a) if for $x_0 \in D \subset \mathbb{R}^n$ the level set $N(x_0) := \{ x \mid f(x) \leq f(x_0) \}$ is compact, then either $\nabla_x f(x_k) = 0$ for some $k$ or there exists a cumulation point $\hat{x}$ of the series $\{x_k\}$ with $\nabla_x f(\hat{x}) = 0$

b) the convergence is asymptotically linear, i.e.

$$
\begin{align*}
f(x_{k+1}) & \leq \left( \frac{\lambda_{\text{max}} - \lambda_{\text{min}}}{\lambda_{\text{max}} + \lambda_{\text{min}}} \right)^2 f(x_k) + O(\|x^k - \hat{x}\|^2) \\
\|x_{k+1} - \hat{x}\|_Q & \leq \frac{\lambda_{\text{max}} - \lambda_{\text{min}}}{\lambda_{\text{max}} + \lambda_{\text{min}}} \|x_k - \hat{x}\|_Q + O(\|x^k - \hat{x}\|^2)
\end{align*}
$$

where $\lambda_{\text{max}} = \lambda_{\text{max}}(\nabla^2_x f(\hat{x}))$, $\lambda_{\text{min}} = \lambda_{\text{min}}(\nabla^2_x f(\hat{x}))$

$$
Q := \nabla^2_x f(\hat{x})
\quad \|v\|_q := (v, Qv)
$$

and $Q := \nabla^2_x (f(\hat{x}))$ is assumed to be positive definite

How can we improve the convergence rate?

Note that $\nabla^2_x f(x) = A^{-1}\left(\frac{\partial^2 f(x)}{\partial x_i \partial x_j}(x)\right)_{ij}$, if $A$ defines the local metric, $(x, y) = x^\top Ay$.

We should have $\lambda_{\text{max}}(A^{-1}\left(\frac{\partial^2 f(x)}{\partial x_i \partial x_j}(x)\right)_{ij}) \approx \lambda_{\text{min}}(A^{-1}\left(\frac{\partial^2 f(x)}{\partial x_i \partial x_j}(x)\right)_{ij})$.

$\Rightarrow$ ideal, if we choose $A \approx \left(\frac{\partial^2 f(x)}{\partial x_i \partial x_j}(x)\right)_{ij}$

Then $\nabla_x f(x_k) \approx -\left(\frac{\partial^2 f(x_k)}{\partial x_i \partial x_j}(x_k)\right)^{-1}_{ij} (f(x_k))' \to "\text{variable metric methods}"$ (Powell)

**Lemma 1.3.4 (Newton-interpretation)**

The direction $d_k = -(f''(x_k))^{-1}(f(x_k))'$ is the Newton direction for the necessary condition of first order.

**Proof:** general Problem $\varphi(x) = 0$

Newton-dir. $d^N_k = -(-\varphi_k')^{-1}\varphi_k$
solves the equation
\[ \varphi(x_k) + \varphi'(x_k) d_k = 0 \]

We apply that to the necessary condition \( f'(x) = 0 \)
we have to solve \( f'(x_k) + f''(x_k) d_k = 0 \)
\[ \Rightarrow d_k = -\left( f''(x_k) \right)^{-1} f'(x_k) \]

Remarks:

a) The Newton-method for optimization can now be considered as a steepest descent method with a special choice of a varying metric

b) Under the assumption \( \nabla^2 f \geq c > 0 \) one can show local quadratic convergence of the Newton-iteration for \( f' = 0 \).

c) There exist nonlinear variants (Fletcher-Reeves/Polak-Ribiere) of the conjugate gradient method. They can be considered as halfway between gradient and Newton method.

Globalization techniques:
so far, we have only convergence for exact line-search and positive definite Hessian everywhere

Approximate line-searches: for \( \varphi : \mathbb{R} \rightarrow \mathbb{R} \)

**Goldstein:** choose \( 0 < \mu_1 \leq 1/2 \leq \mu_2 < 1 \) (e.g. \( \mu_1 = 0.1, \mu_2 = 0.9 \))
and determine \( \tau \) such that
\[ \varphi(0) + \tau \mu_2 \varphi'(0) \leq \varphi(\tau) \leq \varphi(0) + \tau \mu_1 \varphi'(0) \]

**Armijo:** determine \( \tau \) as the largest number in the sequence \( \{q^i\} \)
\[ \varphi(q^i) \leq \varphi(0) + q^i \mu \varphi'(0) , \mu < 1 \) (e.g. \( \mu = 0.8 \))
\[ q \text{ e.g. } = 1/10 \text{ or } 1/2 \in (0,1) \]
Theorem 1.3.5 (Schittkowski)
consider the algorithm

\begin{enumerate}
\item start: \(x_0\) no local minimum, choose \(\varepsilon > 0\)
\item choose descent direction \(d_k\) with
\[\varphi(t) := f(x_k - t d_k) < f(x_k) \forall t \in [0, \varepsilon]\]
\item determine \(\min_t \varphi(t)\) approximately according to Armijo or Goldstein
\item \(x_{k+1} := x_k + \tau d_k, \ k := k + 1\)
\end{enumerate}

If \(N(x_0)\) is compact and
\[\left(\nabla_x f(x_k), d_k\right) \leq -\varepsilon \|d_k\| \|\nabla_x f(x_k)\| \text{ for } \varepsilon > 0\]
then, each accumulation point \(\hat{x}\) satisfies \(\nabla_x f(\hat{x}) = 0\)

Remarks:

a) approximate line-search can be algorithmically improved by quadratic approximation of the level function \(\varphi\).

b) all gradient variants derived by the variable metric approach are obviously descent direction

c) still the problem of intermediate indefinite Hessian is not resolved

remedy: trust region strategies

\begin{itemize}
\item limit step length a priori \(\|d_k\| \leq \delta_k\)
\item minimize local model
\[\min_{d} m(x_k, d) \quad \text{s.t. } \|d\| \leq \delta_k\]
then \(x_{k+1} = x_k + d_k\)
* typically local model is derived from Taylor expansion

\[ m(x_k, d) = f(x_k) + f'(x_k) d + \frac{1}{2}(d, Qd) \]

with \( Q \approx \nabla_x^2 f(x_k) \)

sufficient for convergence: fractional Cauchy decrease

Cauchy point: \( \tau_c = \arg \min m((x_k, -\tau \nabla_x f(x_k))) \)

\[ \text{s.t. } \| \tau \nabla_x f(x_k) \| \leq \delta_k \]

\[ p_c = x_k + \tau_c \nabla_x f(x_k) \]

fractional Cauchy decrease

\[ f(x_n) - f(x_{k+1}) \geq \alpha (f(x_k) - f(p_c)) , \alpha \in \mathbb{R} \]

* for \( \delta_k \) large: \( d_k \approx \text{Newton step} \)

* for \( \delta_k \) small: \( \delta_k \approx \text{steepest descent direction} \)

(can be derived from necessary conditions)

How to solve efficiently the trust-region subproblem?

- various approaches exist

- simplest and still viable one: Steihaug's trick:
  try to solve the unconstrained problem

\[ \min_d m(x_k, d) \]

by conjugate gradient. As soon as the CG-step surpasses the trust-region, stop - solution found. Other two cases trivial.
How to update the trust-region?

Define ratio: \( \rho_k := \frac{f(x_k) - f(x_{k+1})}{m_k(0) - m_k(d_k)} \)

Algorithm (Trust Region)

Given \( \tilde{\Delta} > 0, \Delta_0 \in (0, \tilde{\Delta}) \), and \( \eta \in [0, \frac{1}{4}) \):

\[ \text{for} \ k = 0, 1, 2, \ldots \]

Obtain \( d_k \) by (approximately) solving TR-problem

Evaluate \( \rho_k \)

- if \( \rho_k < \frac{1}{2} \)
  \[ \Delta_{k+1} = \frac{1}{4} \| d_k \| \]
- else
  - if \( \rho_k > \frac{3}{4} \) and \( \| d_k \| = \Delta_k \)
    \[ \Delta_{k+1} = \min(2\Delta_k, \tilde{\Delta}) \]
  - else
    \[ \Delta_{k+1} = \Delta_k ; \]

- if \( \rho_k > \eta \)
  \[ x_{k+1} = x_k + d_k \]
- else
  \[ x_{k+1} = x_k ; \]

end (for)

How to obtain good Hessian approximations

Popular: Broyden-updates (BFGS, DFP, ...)

\[ Q_{k+1} = Q_k + U(x_{k+1} - x_k, \nabla f_{k+1} - \nabla f_k) \]

- "Quasi-Newton-methods"
- Typically local superlinear convergence can be shown, if \( Q_k \| d_k \| \xrightarrow{k \to \infty} Hess_k \| d_k \| \).

- \( \Delta \) for line-search globalization, the updates need to be positive def. always!
- Limited memory update variants exist
  \( \Rightarrow \) advantages in problems with many local minima
- Filtering techniques by Kelley:
  - step reduction is omitted as long as possible
    \( \Rightarrow \) advantages in problems with many local minima
- Special case: nonlinear least-squares
  \( \Rightarrow \) Gauss-Newton technique
1.3.2 Sequential quadratic programming (SQP)

Recall unconstrained case
\[ \min f(x) \quad \text{use step} \quad \min_{d} f(x_k) + f'(x_k) d + (d, Q_k d) \]
\[ Q_k \approx Hess f \]

Analogously in the constrained case:
\[ \min f(x) \quad \text{s.t. } c(x) = 0 \quad \text{use step} \quad \min_{d} f'(x_k) d + (d, Q_k d) \]
\[ c(x_k) + c'(x_k) d = 0 \quad (QP) \]
\[ d(x_k) + d'(x_k) d \geq 0 \]
\[ Q_k \approx Hess \mathcal{L} \]

the adjoints of the linearized constraints are used as approximations of the adjoints to the nonlinear constraints

Globalization by use of merit functions for line-search and trust-region

Software examples: MINOS, LANCELOT, SNOPT

Same questions as before:

→ how to obtain good Hessian approximations?
  - Approximation needs only be good on the kernel of the constraints
    → superlinear convergence
  - Particular problem with lack of positive definiteness in Quasi-Newton-updates.
  - Can be overcome by augmented Lagrangian SQP-variant
  - Particularly interesting variant: (partially) reduced SQP (→ later)

→ how to solve quadratic subproblems?
  - In finite dimensions basic difference between range space and null space decompositions
  - Inequalities are treated by active set or interior point strategies

Literature:

Nocedal/Wright: Numerical Optimization, Springer 1999

Decision-tree for optimization software: http://plato.la.asu.edu/guide.html
CUTE testing environment: http://hsl.rl.ac.uk/cuter-www/
NEOS optimization server: http://www-neos.mcs.nal.gov/neos/
2 Optimization for differential equation models

2.1 Optimal control in ODE

Control for elliptic/Stationary problems → section 2.3
Here, instationary problems with a time dimension are treated.
Optimal control comes in two flavors:

1. off-line control
2. feedback control

2.1.1 Off-line control

A dynamical model
\[ \dot{y} = f(y, u), \quad \dot{y}(t_0) = y_0 \in Y \]
is considered on a fixed time frame \( t \in [t_0, T] \). The control \( u \in U = \{ [t_0, T] \to Z, Z \text{ Hilbertspace} \} \) is a time dependent function.

More complicated process models, like implicit DAE with interior point constraints, can be considered, but not in this introduction.

The dynamical system is to be controled so that an objective - typically in integral form - is minimized
\[ \varphi(y, u) = \int_{t_0}^{T} L(y(t), u(t)) \, dt \]

The whole optimization problem now is written as
\[
\begin{align*}
\min_{(y, u)} & \quad \int_{t_0}^{T} L(y(t), u(t)) \, dt \\
\text{s.t.} & \quad \dot{y} = f(y, u) \\
& \quad r(y(t_0), y(T)) = 0 \\
& \quad d(y, u) \geq 0
\end{align*}
\]

Additional restrictions, \( d \), may be present, as well.
Calculus of Variations, the Pontryagin Maximum principle

Necessary conditions in function space lead to the following characterization of the optimal solution in the form of a two-point boundary value problem:

Hamiltonian $\mathcal{H}(y, u, \lambda, t) := -L(y, u, t) + \lambda(t)^\top f(y, u, t)$ is maximized locally for determining the control

\[
\begin{align*}
\dot{y} &= f(y, u), \quad r(y(t_0), y(T)) = 0 \\
\dot{\lambda} &= -\frac{\partial}{\partial y} \mathcal{H}(y, u, \lambda, t), \quad \exists \alpha \in \mathbb{R}^n_r \\
\lambda(0) &= \nabla_{y_0} \alpha^\top r(y_0, y_T) \\
\lambda(T) &= -\nabla_{y_T} \alpha^\top r(y_0, y_T) \\
u(t) &= \arg \max_{u \in Z(t)} \mathcal{H}(y(t), u, \lambda(t), t)
\end{align*}
\]

Remarks:

- If the local maximization of the Hamiltonian does not yield a solution, we are in a situation of singular control and have to use higher order information.

- The TPBVP is usually highly nonlinear and this even more in the presence of state constraints or model switches (bang-bang control). Good initial guesses are necessary.

- gives highly accurate solutions

- examples: rocket on wheels, elbow robot
Direct approach

1. Discretization of the control function, e.g. piecewise constant on a grid

2. Choice of a state parameterization fitting to the control grid by multiple shooting (or collocation)

This results in a large finite dimensional optimization problem

\[
\min_{s,u} \sum_i L_i(y, u_i) \\
\text{s.t.} \quad s_{i+1} - y(\tau_{i+1}, u_i; s_i, \tau_i) = 0 \quad \forall i = 1, \ldots, N - 1, \quad s_0 = y_0
\]

Again, this problem is highly structured and the exploitation of these structures is very rewarding, but basically, this problem can now be tackled by any nonlinear SQP-solver, if the gradient generation is performed by appropriate recursions (Variational ODE!). The QP-subproblems are related to the Riccati-equation.

Software realizations:
- MUSCOD (Bock group)
- OCPRSQP (Schulz)
- SOCS (Boeing, Betts)
The case, when $\dot{y} = f$ is not just a few scalar ODE

In principle, one can apply an SQP-technique in Hilbert spaces to this optimization problem. However, if the space $Y$ needs a fine space-discretization, a full SQP-approach necessitates the storage of all time-steps in $[t_0, t]$, which is often not possible.

In this situation, one can only apply techniques from suboptimal control (e.g. instantaneous), which essentially shrink the optimization horizon to one time-step of a method of lines $[t_0, T] = [t_k, t_{k+1}]$. This brings the situation back to elliptic control problems and section 2.3.

Reduced order modeling based on proper orthogonal decomposition (POD)

Basic idea:

a) choose finite set of vectors $y_1, \ldots, y_l \in Y$ which are orthonormal and represent the dynamics of the problem

b) Galerkin approach: $y(t) : \sum_{i=1}^{l} \eta_i(t) y_i$  
   $\Rightarrow$ dynamics is finite dimensional

   $\dot{\eta}(t) = (f(\sum_{i=1}^{l} \eta_i(t)y_i, u), y_j), i = 1, \ldots, l$

c) do the same for the control

d) Result: finite-dimensional instationary optimal control problem

Remarks

- how to generate the Galerkin basis:
  - snapshots (numerical or photograhical), which give information on states and first derivatives
  - SVD (or Lanczos iteration) leads to orthonormal basis with maximum information

- snapshots might not carry crucial information of the process $\rightarrow$ trust region approach with convergence result exists in theory (Fahl/Sachs)

- good result in some practical examples

High dimensional optimal control methods are still not yet in a state of finished research.

Now, we assume $Y = \mathbb{R}^{n_y}$ and $Z = \mathbb{R}^{n_u}$. 
2.1.2 Feedback and nonlinear model predictive control

Block diagram:

Typical controllers:

(P) proportional controller \( u = K_p e \)
(D) derivative controller \( u = K_D \dot{e} \)
(I) integral controller \( u = K_I \int e(t) \, dt \)

combinations (i.e. sums of that) are called \( PI, DP, DI, PID \) controllers
design of these control laws \( \rightarrow \) control systems

Here, we want to exploit the fact that we can compute off-line controls in a fast way.

Linearized neighbouring feedback control (perturbation feedback control)
For small disturbances, use Taylor series expansion

\[
u^{fb}(t) = u^n(t) + K(t)(e^d(t) - e^n)\]

\( fb^n \) : feedback
\( n^n \) : nominal
\( d^n \) : disturbed

where \( K(t) = \frac{\partial u}{\partial e} (e^n(t)) \)

This is based on the fact that the mapping \( \varepsilon \mapsto u \), where \( u(\varepsilon) \) solves \( \min_u f(u, \varepsilon) \) is differentiable and

\[
\frac{\partial u}{\partial \varepsilon} = -[\text{Hess}_u f]^{-1} f_u \varepsilon
\]
Proof: implicit function theorem applied to $\nabla_x f(u(\varepsilon), \varepsilon) = 0$

Analogous results are valid for equality constrained optimization-problems. For inequality constraints, a little more sophistication has to be applied (→ Bock/Krämmer-Eis)

Remark: Second order information has to be available for this approach.

linear model predictive control

at $t^n$ one performs a backward system identification (Kalman-filter, extended Kalman-filter or nonlinear parameter estimation)

For the control horizon, an optimal control problem is solved as described above.

Within a real-time environment, there might not be enough time to compute the full control history.
A remedy for that case is to use in the first control subinterval the control update after one SQP and so on. (cf. Diehl-thesis)
2.2 Parameter estimation and optimum experimental design

2.2.1 (Generalized) Gauß-Newton theory

The process: We consider a finite dimensional parameter vector $p \in \mathbb{R}^n_p$, which is mapped to an observation vector $\psi \in \mathbb{R}^{n_\psi}$ (measurements). This observation vector is perturbed by measurement errors $\varepsilon \in \mathbb{R}^{n_\varepsilon}$

$$p \xrightarrow{\Psi} \bar{\psi} = \psi(p) + \varepsilon$$

Usually the measurements are assumed to be multivariat normally distributed around 0, $\varepsilon \sim N(0,W)$, $W \in \mathbb{R}^{n_\varepsilon \times n_\varepsilon}$ covariance of measurements

Then, a Maximum-Likelihood estimator for the parameter corresponding to a given observation $\bar{\psi}$ is the solution of

$$\min_p \frac{1}{2} \| \psi(p) - \bar{\psi} \|_W^2 = \frac{1}{2} (\psi(p) - \bar{\psi})^\top W (\psi(p) - \bar{\psi})$$

Other assumptions on the statistical properties of $\varepsilon$ (like equi-distribution) lead to different objective functions (like $\| \psi(p) - \bar{\psi} \|_1$). Also the robustness of the estimation with regard to outliers can be improved by modifications of the objective. ($\rightarrow$ see "robust parameter estimation")

Here, we stick to the $l_2$-output-least-squares objective as formulated above.

The necessary condition for optimality of the output-least-squares problem is

$$0 = \frac{1}{2} \frac{d}{dp} (\psi(p) - \bar{\psi}, \psi(p) - \bar{\psi})_W = (\psi_p, \psi(p) - \bar{\psi})_W$$

after introducing a scalar product in $P$, $(p, q)_A = p^\top A q$ we can write

$$0 = \nabla_p (\psi(p) - \bar{\psi}, \psi(p) - \bar{\psi})_W = A^{-1} \psi_p^\top W (\psi(p) - \bar{\psi}) = (\nabla_p \psi)^\ast (\psi(p) - \bar{\psi})$$

Literature:


Compact Course: Nonlinear Optimization & Inverse Problems
Stuttgart, March 9-11, 2005
University of Trier
Now, we try to apply a Newton-SQP method to find the minimum: we need the Hessian

\[ \text{Hess}_p \left( \frac{1}{2} (\psi(p) - \bar{\psi} \!, \! \psi(p) - \bar{\psi})_W \right) = \psi_p^T W \psi_p + \psi_{pp}^T W (\psi(p) - \bar{\psi}) \]

Linear statistical theory gives us the result, that

\[ H_1 := \psi_p^T W \psi_p \text{ is the inverse of the variance-covariance matrix of the linear model } \Delta p \mapsto \psi_p \Delta p \text{ evaluated at } p. \]

Since \( H_1 \) is positive definite (if \( \psi_p \) is injective) it is a good and now statistically motivated candidate for a (risk-)metric in P-space.

(according to the motto ”a step in an unsure direction is riskier than in a sure direction”)

If we choose \( A = H_1 = \psi_p^T W \psi_p \), then the gradient step

\[ \Delta p = -\nabla_p (\psi(p) - \bar{\psi} \!, \! \psi(p) - \bar{\psi}) = -H_1^{-1} \psi_p^T W (\psi(p) - \bar{\psi}) \]

is also the solution of the problem

\[ \min_{\Delta p} \frac{1}{2} \| \psi(p) + \psi_p \Delta p - \bar{\psi} \|_W^2 \] "Linearization under the norm"

The resulting method is called the Gauß-Newton-method.

Properties

- local linear convergence

\[ \| p_{k+1} - \hat{p} \| \leq c \| p_k - \hat{p} \| , \quad c < \infty \]

but \( c \to 0 \) if \( \| \psi(\hat{p}) - \bar{\psi} \| \to 0 \)

- there exist structured Hessian techniques, which recover the missing part of the Hessian,

\[ \psi_{pp}^T W (\psi(p) - \bar{\psi}) \]

by Boyden-updates;

these methods allow local superlinear convergence and the solution of ”large residual problems”

- the algorithm ”feels” well- or ill-conditioning of the problem as reflected in \( H_1 \)
The idea of linearization under the norm can also be applied to constrained optimization problems with a process as constraint

\[
\begin{align*}
\min_{x,p} & \quad \frac{1}{2} \left\| \psi(x, p) - \bar{\psi} \right\|^2_W \\
\text{s.t.} & \quad c(x, p) = 0 \\
& \quad d(x, p) \geq 0
\end{align*}
\]

is solved iteratively by computing the increments

\[
\begin{align*}
\min_{\Delta x, \Delta p} & \quad \frac{1}{2} \left\| \psi(x, p) + \psi_x \Delta x + \psi_p \Delta p - \bar{\psi} \right\|^2_W \\
\text{s.t.} & \quad c_x \Delta x + c_p \Delta p = -c(x, p) \\
& \quad d_x \Delta x + d_p \Delta p \geq -d(x, p)
\end{align*}
\]

This amounts to the choice of

\[
H_1 = \begin{bmatrix} \psi^\top_x W \psi_x & 0 \\ 0 & \psi^\top_p W \psi_p \end{bmatrix}
\]

as an approximation to the Hessian of the Lagrangian \( Hess(x, p) \mathcal{L}(x, p, \lambda) \)

The method is called Generalized Gauß-Newton method and is due to Bock (1985).

Properties:

- again local linear convergence with small convergence rate for good measurements
- Bock (1985) also has introduced highly efficient line-search globalization techniques for this problem class based on so-called ”natural level functions”.
- reduced variant uses the matrix

\[
\hat{H}_1 = \begin{bmatrix} 0 & 0 \\ 0 & B_1 \end{bmatrix}
\]

with \( B_1 = T^\top H_1 T \) where \( T = \begin{bmatrix} -c_x^{-1} c_p \\ I \end{bmatrix} \)

as approximation for the Hessian of Lagrangian
- Covariance matrix is the same as in the unconstrained black-box reformulation

\[
Cov(p) = [T^\top H^{-1} T]^{-1}
\]
2.2.2 Multiple shooting implementation for instationary models

Process model \( \dot{y} = f(y, p) \) e.g. chemical engineering or groundwater flow

\( y(0) = y_0 \)

Observations \( \psi(y(t_i)) \), \( i = 1, \ldots, N \)

Output-least-squares problem:

\[
\min \frac{1}{2} \sum_{i=1}^{N} \frac{1}{\sigma_i^2} \| \psi(y(t_i)) - \bar{\psi} \|_2^2 \\
\text{s.t. } \dot{y} = f(y, p), \ y(0) = y_0
\]

Why multiple shooting?

\( \rightarrow \) robustness

\( \rightarrow \) exploitation of a priori knowledge

\( \rightarrow \) cost equivalent to single shooting

\( \rightarrow \) usually not enough storage available for adjoint techniques

Multiple shooting reformulation:

\[
\min \frac{1}{2} \sum_{i=1}^{N} \frac{1}{\sigma_i^2} \| \psi(y(t_i, p; s_{j(t_i)}, \tau_{j(t_i)})) - \bar{\psi} \|_2^2 \\
\text{s.t. } s_{j+1} - y(\tau_{j+1}, p; s_j, \tau_j) = 0, j = 0, \ldots, M - 1 \\
\quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad s_0 = y_0
\]

where \( y(t, p; s, \tau) \) is the solution of the initial value problem \( \dot{y} = f(y, t) \), \( y(\tau) = s \) and \( j(t_i) = \max\{ j | \tau_j \leq t_i \} \).
We envision a multiple shooting grid of the form

![Multiple Shooting Grid Diagram](image)

Possible problem: Generalized Gauß-Newton needs the construction of Wronskian Matrices

\[ G_j = \frac{\partial y(\tau_{j+1}, p; s_j, \tau_j)}{\partial s_j} \]

Remedy:

* in reduced generalized Gauß-Newton one only needs matrix-vector products with Wronskian matrices, \( G_j \Delta p \)

* with implicit integration schemes, the computation of \( G_j \Delta p \) means \( \#p \) additional linear solves in each time-step after the convergence of a Newton method. No new system matrix has to be built up or factorized.

Implementations: PARFIT (Bock et.al) COLFIT (Schulz)
2.2.3 Optimum experiment design

In principle, the mapping from measurement distributions to parameter values is nonlinear.

However, locally in the solution of the output-least-squares problem, it makes sense to investigate an affine linear process model defined by first order Taylor expansion:

$$\psi(p) \approx \psi(\hat{p}) + \psi_p(p - \hat{p})$$

From that we can, by applying linear sensitivity theory (e.g. Pukelsheim 1993), obtain the so-called moment or information matrix

$$M = \psi_p^T W \psi_p$$

which is exactly the Gauß-Newton-Hessian approximation. Its inverse is the variance-covariance matrix of the parameters

$$Cov(p) = (\psi_p^T W \psi_p)^{-1}$$

Now, ellipsoids in measurement space are mapped to ellipsoids in parameter space.
The problem of experimental design is to influence the measurement conditions (i.e. further process variables which have tacitly been fixed to a certain value up to that point) in such a way that the confidence regions in parameter space are as small as possible.

That leads to objective criteria on the Covariance matrix.

$$\min_q \varphi\left(\text{Cov}(\hat{p}; q)\right)$$

where $q$ is some adjustable "screw" in the process.

Examples for $q$ are: Location and number of measurements in space and time, geometry of measurement device, process conditions like temperature or concentration history, ...

Note, that $\hat{p}$ is not among the degrees of freedom for optimization but rather a fixed parameter.

Candidates for optimization criteria are

- $\varphi_A(C) = \frac{1}{n} \text{trace}(C)$ "A-criterion"
- $\varphi_D(C) = \sqrt{\text{det}(C)}$ "D-criterion"
- $\varphi_E(C) = \max \sigma(C)$ "E-criterion" (Minimize the maximal eigenvalue)
- $\varphi_M(C) = \max\{\sqrt{C_{ii}}, i = 1, \ldots, n\}$ "M-criterion" (Minimize the maximal length of confidence intervals)

These criteria depend strongly on the scaling of the parameters.

$\varphi_E$ leads to eigenvalue optimization similarly to robust controller design.

Numerical solution of the experimental design problems:

- SQP techniques have been developed in particular in the group of Bock (Heidelberg) for chemical engineering problems. There, one has to generate 2nd order derivatives for which special recursions have been implemented in Heidelberg.

- There are reasonably good experiences with the application of Nelder/Mead, if computation time is not a crucial issue (still 1st order derivatives are necessary to evaluate the cost function)
Sequential experiment design:

Robust experiment design:

tries to avoid too many loops in experiment design by reducing the nonlinearity in $p$.

This can be motivated by considering a linearized worst case scenario and leads essentially to considering a change of the objective function

$$\varphi(Cov(\hat{p}; q)) := \varphi(Cov(\hat{p}; q)) + \| \nabla_p \varphi(Cov(\hat{p}; q)) \|_p$$

where the norm $\| \cdot \|_p$ comes from an a priori given metric in parameter space e.g. as before for some a priori known information metric $M$

$$\begin{align*}
(p_1, p_2)_p &= p_1^T M p_2 \\
\Rightarrow \| \nabla_p \varphi \|_p &= (\varphi_p M^{-T} M M^{-1} \varphi_p^T)^{1/2} = (\varphi_p M^{-1} \varphi_p^T)^{1/2} \\
&= (\varphi_p \overline{Cov} \varphi_p^T)^{1/2}
\end{align*}$$

Literature:

Pukelsheim: Optimal design of experiments,
Wiley & Sons, 1993

Körkel: Numerische Methoden für optimale Versuchplanungsprobleme bei nichtlinearen DAE-Modellen,
PhD-thesis, IWR Heidelberg, 2002
2.3 Shape optimization and inverse problems

Problem class

\[
\begin{align*}
\min & \quad f(x, p) \\
\text{s.t.} & \quad c(x, p) = 0, \quad c_x\ \text{non singular} \\
& \quad d(x, p) \geq 0
\end{align*}
\]

\(x \in X\) state variables from some Hilbert space
\(p \in P\) design variables from some Hilbert space
\(c\) system equation: flow, elasticity, ...
\(d\) restrictions

often no time dimension is involved, but that is not a must.

shape optimization: objective function is of highest importance
\[\rightarrow \text{stopping criterion based on change of objective.}\]

inverse problems: typical objective =
\[\text{output least squares + regularization}\]
\[\text{design variable is of highest importance}\]
\[\rightarrow \text{stopping criterion based on norm of increment.}\]

often shape optimization and inverse problems are formulated so that the unknown design variable should be chosen in a way that certain observations
\[\bar{\psi} = \psi(x, p)\]
are met.

Direct inverse modeling uses \(\bar{\psi}\) as boundary condition in \(c(x, p) = 0\), thus determining \(p\)

problems: \(\bar{\psi}\) is prone to measurement errors which can lead to severe perturbations of the solution

better approach: reformulation as an optimization problem
\[
\begin{align*}
\min & \quad \| \bar{\psi} - \psi(x, p') \|^2 \quad (+ \text{regularization}) \\
\text{s.t.} & \quad c(x, p) = 0
\end{align*}
\]

that is the Output-least-squares approach.

The latter has also some statistical implications as we have seen above.
2.3.1 The concept of ill-posedness

**Definition 2.3.1** Let $K : X \to Y, X, Y$ Hilbert spaces. An operator equation

$$K(x) = y$$

is said to be well-posed provided

(i) for each $y \in Y$ there exists $x \in X$, called a solution, for which $K(x) = y$ holds

(ii) the solution $x$ is unique; and

(iii) the solution is stable with respect to perturbations in $y$. This means that if $K\hat{x} = \hat{y}$

and $Kx = y$, then $x \to \hat{x}$ whenever $y \to \hat{y}$

otherwise the problem is said to be ill-posed

Now we consider the mapping

$T : p \mapsto x(p) = \text{unique solution of } c(x, p) = 0$

e.g. $x(p) = -\Delta^{-1} p$, if $c(x, p) = -\Delta x - p = 0$

Typically $T$ is a compact operator, i.e. the closure of the image of a bounded set $\subset P$ is

a compact subset $\subset X$.

(i.e. there are values $\psi \in X$, for which one cannot find a reasonable $p$ for which $\psi = x(p)$)

In Hilbert spaces, there exists a singular value decomposition of compact operators such that

$$U^*KV = \begin{bmatrix} s_1 & s_2 & 0 \\ & \ddots & \vdots \\ & & s_\infty \\ 0 & 0 & 0 \end{bmatrix}$$

with $U, V$ orthogonal operators, $s_i \geq 0 \ \forall i$, and

$$\lim_{i \to \infty} s_i = 0$$

Therefore, even if $K$ is bijective and we have the representation
\[ U^* K U = \begin{bmatrix} s_1 & 0 \\ \vdots & \ddots \\ 0 & \cdots & s_\infty \end{bmatrix} \quad \text{with } \lim_{i \to \infty} s_i = 0 \]

we cannot determine a solution of \( K p = \psi \) in a numerically stable way. Numerically we will always see a nontrivial nullspace of this operator.

In Hilbert spaces one can, exactly as in the finite dimensional case, define a Moore-Penrose-pseudoinverse as

\[ K^+ := U \begin{bmatrix} 1/s_1 & 0 \\ \vdots & \ddots \\ 0 & \cdots & 1/s_\infty \end{bmatrix} V^* \]

which is equivalent to the solution of

\[
\begin{aligned}
\min_p & \quad \|p\|^2 \\
\text{s.t.} & \quad \|K p - \psi\|^2 = \min
\end{aligned}
\]

A variation of that approach is the formulation of

Tikhonov-regularization

\[
\min_p \quad \|K p - \psi\|^2 + \mu \|p\|^2 , \quad \mu > 0
\]

or equivalently and more general

\[
\begin{aligned}
\min_{(x,p)} & \quad f(x, p) + \mu \|p\|^2 \\
\text{s.t.} & \quad c(x, p) = 0 \\
& \quad d(x, p) \geq 0
\end{aligned}
\]

This problem is now well-posed.
2.3.2 Regularization techniques

Regularization:
transforms an ill-posed, and therefore numerically unsolvable, problem into a well-posed, and therefore solvable problem.

But: The problem is always changed - we will never get the "true" solution.

**Tikhonov-regularization:** \( f(x, p) + \mu \| p \|^2 \)
The choice of the norm \( \| \cdot \| \) is crucial for the meaningfulness of the solution and the performance of numerical solution approaches.

**Bayesian regularization:** \( f(x, p) + \mu \| p - \hat{p} \|^2 \)
- \( \hat{p} \): rough a priori knowledge about solution
- algorithmically equivalent to Tikhonov-regularization
- related to Levenberg-Marquardt (\( l_2 \)-penalty) globalization technique

**Total variation regularization:** \( f(x, p) + \mu \| \text{grad} \, p \|_1 \)
used e.g. in image deblurring

**Discrepancy principle** (Morozov 1966):
Select (a posteriorily and iteratively) the largest value for \( \mu \) so that the solution satisfies \( \| KP - \psi \|^2 \leq \delta \), where \( \delta \) is a given admissible error level.

**Regularization by finite parameterization of the design space**
Often done by use of splines reflecting manufacturing conditions.
Result: essentially finite dimensional optimization problem, where no regularization is necessary.

**Regularization by truncated iteration**
- start with initial guess \( p \equiv 0 \)
- perform a variant of the steepest descent algorithm
  \( \Rightarrow \) iteration vector is becoming rougher and the objective first decreases and then increases after a certain point
- stop the iterations as soon as \( \| KP - \psi \|^2 \) starts rising or is below a discrepancy level \( \delta \)
- approach is only viable in black-box algorithmic framework.
2.3.3 Discretization aspects

Topology optimization via homogenization approach

optimum density distribution $\rho : \Omega \rightarrow [0, 1], \quad \text{vol}(\rho) \leq V_0$ is searched for.

minimal compliance in elasticity or minimum electrical resistance in electrical devices

Examples:

$$\min \sum_{\nu=1}^{N} \int_{\Gamma_\nu} I_\nu \cdot u \, ds$$

s.t. $\nabla \cdot (\rho \nabla u) = 0$

$\rho \cdot \vec{n} \cdot \nabla u = \begin{cases} I_\nu & \text{on } \Gamma_\nu \subset \partial \Omega \\ 0 & \text{elsewhere} \end{cases}$

$$\int_{\Omega} \rho \, dx = C$$

$\rho_{\text{min}} \leq \rho \leq \rho_{\text{max}}$

a) typical discretization: staggered, i.e.

$u: \text{bilinear}$

$\rho: \text{piecewise constant}$

- Various regularization techniques are discussed in the literature on structural optimization
- There has been developed a Babushka-Brezzi-style theory for stable discretizations by Joakim Petersson (†, Linköping)
b) Inverse problem for groundwater flow:

$$\min \sum_i (u(x_i) - \hat{u}_i)^2 + \int_{\Omega} \| \nabla p \|^2$$

s.t. $\nabla \cdot (p \nabla u) = 0$ (Darcy)

$u = u_0$ on $\Gamma = \partial \Omega$

Mesh movement:
Typical change of geometry for flow problems

Typical change of geometry in elasticity

The boundary geometry is the degree of freedom
Method of mapping approach:

Reference geometry is mapped to physical geometry

\[ \Phi : \text{diffeomorphism} \]

Thus, the boundary shape enters the differential equation in transformation coefficients on the reference frame.

This mapping can also be used to provide a smooth transition between successive meshes.

Other approach by fictition-domain methods → see publications by Haslinger, in particular:


problem: leads to nondifferentiable optimization problems and the necessity to use subgradients
2.3.4 SQP-approaches to the resulting optimization problems

An SQP approach leads to the following linear-quadratic subproblems

$$\min_{(\Delta x, \Delta p)} \frac{1}{2} \left( \begin{array}{c} \Delta x \\ \Delta p \end{array} \right)^* H \left( \begin{array}{c} \Delta x \\ \Delta p \end{array} \right) + \left( \begin{array}{c} \Delta x \ f \\ \Delta p \ f \end{array} \right)^* \left( \begin{array}{c} \Delta x \\ \Delta p \end{array} \right)$$

(QP)

s.t.  
$$C_x \Delta x + C_p \Delta p = -c$$  
$$D_x \Delta x + D_p \Delta p \geq -d$$

for the increments $$\left( \begin{array}{c} \Delta x \\ \Delta p \end{array} \right)$$, where $H$ denotes an appropriate approximation to the Hessian of the Lagrangian and $C_x, C_p, D_x, D_p$ denote derivatives. (We assume, that $f(x, p)$ already incorporates some regularization).

The notation $(,)^*$ denotes the action of the scalar product.

Several questions arise:

- What Hessian approximation should be used?
- How do we generate the required gradients?
- How should we solve the QP-subproblems?

However, these questions are tightly coupled!

Hessian approximations (for ease omit inequalities)

- Nonlinear least-squares $\rightarrow$ special form, later
- finite and small number of design variables:
  let’s look at the black-box-approach first

\[ \min_p f(x(p), p) \quad (x(p) \text{ is defined by } c(x, p) = 0) \]

An SQP approach to that problem necessitates the solution of

\[ (QP) \quad \min \frac{1}{2} \Delta p^* B \Delta p + \nabla_p f^* \Delta p \]

How do we obtain the (total) \( \nabla_p f \)? - from implicit function

theorem as \( \nabla_p f = \nabla_p f - C_p^* C_x^* \nabla_x f \)

this is the ”adjoint” computation of the gradient.
Since \( B \) is small, it can be approximated by Quasi-Newton updates.

Still, in each iteration, we have to solve \( c(x, p) = 0 \).

If we perform only one Newton step towards feasibility, the result is a reduced SQP method. (→ Schulz)

In each iteration, the linear system to be solved is

\[
\begin{pmatrix}
0 & 0 & C^* \\
0 & B & C_p^* \\
C_x & C_p & 0
\end{pmatrix}
\begin{pmatrix}
\Delta x \\
\Delta p \\
\lambda
\end{pmatrix}
= \begin{pmatrix}
-\nabla_x f \\
-\nabla_p f \\
-c
\end{pmatrix}
\]

If we contrast this with the necessary conditions for a full SQP approach (without inequalities) we see

\[
\begin{pmatrix}
H_{xx} & H_{xp} & C^* \\
H_{px} & H_{pp} & C_p^* \\
C_x & C_p & 0
\end{pmatrix}
\begin{pmatrix}
\Delta x \\
\Delta p \\
\lambda
\end{pmatrix}
= \begin{pmatrix}
-\nabla_x f \\
-\nabla_p f \\
\lambda
\end{pmatrix}
\]

Now, we recognize, that rSQP methods just use a different Hessian approximation, which makes the resulting \((rQP)\)–subproblems triangular and therefore easily solvable.

Now we should recognize, that we called earlier

\[
B : \text{ reduced Hessian} \\
\gamma = \nabla_p f - C_p^* C_x \nabla_x f : \text{ reduced gradient}
\]

This approach preserves high modularity in the implementation.
In principle, it is also possible to generate the reduced gradient by automatic differentiation (Bischof/Griewank)

Solution of the full QP

- Multigrid methods \( \rightarrow \) Borzi/Schulz/Ta 'Asan
- Preconditioned Krylov-subspace methods.
  Best preconditioner: matrix from (r QP)
  \( \rightarrow \) Bittermann/Heinkenschloss/Sachs

Treatment of inequality constraints

- Interior point approaches
- Active-set methods: these are especially good, when there are only design constraints
  (\( \rightarrow \) ”Semismooth Newton-methods” by Kunisch/Hintermüller)

Globalization strategies

- Often two merit functions are used
  \( \varphi_1 \): exact merit function
  \( \varphi_2 \): residual of necessary conditions

  in a ”watchdog” technique, allowing the increase of \( \varphi_2 \) for several iterations as long as \( \varphi_2 \) is decreased.

Pseudo-timestepping approach

Often, a solver for the model equation \( c(x, p) = 0 \) based on (preconditioned) pseudo-timestepping is available in the form of an ODE:

\[
\dot{x} = -W(x)c(x, p),
\]

where \( W(x) \) is a so-called ”preconditioning” matrix. This ODE is integrated towards ”infinity” so that \( \lim_{t \to \infty} x(t) \) gives the solution to the equation \( c(x, p) = 0 \) for a certain \( p \). In this situation, a continuous rSQP-approach as in Hazra/Schulz/Brezillon/Gauger can be profitably applied. This leads to an ODE of the form

\[
\begin{pmatrix}
\dot{x} \\
\dot{p} \\
\dot{\lambda}
\end{pmatrix} = -
\begin{bmatrix}
0 & 0 & W^* \\
0 & B & c_p^* \\
W & c_p & 0
\end{bmatrix}^{-1}
\begin{bmatrix}
\nabla_x L(x, p, \lambda) \\
\nabla_p L(x, p, \lambda) \\
c(x, p)
\end{bmatrix}
\]
Here, the limit at \( t = \infty \) gives the solution to the necessary conditions for optimality. Notice that in this approach no additional globalization strategy is necessary and that the supposedly pre-existing forward solver can be embedded in an implementation. One has to come up with an idea about the reduced Hessian \( B \), which can be obtain by analytical considerations or update-like techniques.

**Literature:**


M. Gunzburger: Perspectives in flow control and optimization, SIAM, 2003


F. Natterer: The mathematics of computerized tomography, Teubner, 1986

G. Rozvany (ed.): Shape and layout optimization of structural systems and optimality criteria methods, Springer 1992


C. Vogel: Computational methods for inverse problems, SIAM Frontiers in applied mathematics, 2002