Optimization methods

Preparation:

- The student blindly draws one topic from the pool of the eight different topics described above.
- The student then has 15 minutes preparation time before the exam.
- Notice that this preparation is open book - all study material is allowed in the preparation room .

Exam:

- First, the examiner will ask a couple of questions on the miniproject.
- Then, the examiner will ask a couple of questions on the topic that was drawn.
- We expect that the discussion will take place at the blackboard.
- During the exam books/slides are not allowed but you may bring your own notes.
- It is not expected that you can do all the intermediate derivations of the theory/algorithms.
- It is expected that you understand the conditions and can interpret the results of the theory/algorithms covered in this course.
- The total time for the oral examination (excluding preparation time) is about 15 - 20 minutes including evaluation.

Topics:

Constrained Optimization

Linear Programming and The Simplex Method

Duality and Iterative Methods

Interpoint methods

Alternating Direction Methods of Multipliers (ADMM)

Semidefinite Relaxation

Simulated Annealing

Genetic algorithms

Constrained Optimization

Chapter 10 covers most of this subject.

First-Order Necessary Conditions for a Minimum

(a) If $f(x) \in C^1$ and x^* is a local minimizer, then

$$g(x^*)^T d \geq 0 \tag{1}$$

for every feasible direction d at x^*.

(b) If x^* is located in the interior of $\mathbf{R}$, then

$$g(x^*) = 0 \tag{2}$$

Second-Order Necessary Conditions for a Minimum

(a) If $f(x) \in C^2$ and x^* is a local minimizer, then for every feasible direction d a x^*

  (i) $g(x^*)^T d \geq 0$

  (ii) If $g(x^*)^T d = 0$, then $d^T H(x^*) d \geq 0$

(b) If x^* is a local minimizer in the interior of $\mathcal{R}$, then

  (i) $g(x^*) = 0$

  (ii) $d^T H(x^*) d \geq 0$ for all $d \neq 0$

Second-Order Sufficient Conditions for a Minimum
If $f(x) \in C^2$ and x^* is located in the interior of $\mathcal{R}$, then the conditions

(a) $g(x^*) = 0$

(b) $H(x^*)$ is positive semi definite

are sufficient for x^* to be a strong local minimizer.

See section 10.3, page 273 for classification of constrained optimization problems.

- Linear programmming
- Quadratic programmming

- Convex programming
- General constrained optimization problem

Consider the problem of nonlinear equality constraints and variable transformations.

Lagrange multipliers
At a local minimizer of a constrained optimization problem, the gradient of the objective function is a linear combination of the gradients of the constraints.

$$\nabla f(x^*) = \sum_{i=1}^{p} \lambda_i^* \nabla a_i(x^*) \tag{3}$$

First-Order Necessary Conditions for a Minimum, Equality Constraints
If x^* is a constrained minimizer of an equality contrained problem and is a regular point of the contraints, then

(a) $a_i(x^*) = 0$ for $i = 1, 2, \ldots, p$

(b) There exist Lagrange multipliers λ_i^* for $i = 1, 2, \ldots, p$ such that

$$\nabla f(x^*) = \sum_{i=1}^{p} \lambda_i^* \nabla a_i(x^*) \tag{4}$$

First-Order Necessary Conditions for a Minimum, Inequality Constraints
If x^* is a constrained minimizer of an inequality constrained problem and is a regular point of the constraints, then

$$\nabla f(x^*) = \sum_{i=1}^{p} \lambda_i^* \nabla a_i(x^*) + \sum_{k=1}^{K} \mu_{j_k}^* \nabla c_{j_k}(c^*) \tag{5}$$

where K is the number of active inequality constraints.

Karush-Kuhn-Tucker conditions
Build upon Lagrange multipliers for constrained problems. See theorem 10.2, page 298 in Practical Optimization for full theorem.

Second-Order Necessary Conditions for a Minimum, Equality Constraints
See theorem 10.3, page 303, and forward.

Necessary and Sufficient Conditions for a Minimum in Alternative-Form LP Problem
See theorem 11.1 on page 325.

Linear Programming and the Simplex Method

LP Standard Form

$$\begin{aligned} \text{minimize } & f(x) = c^T x \\ \text{subject to: } & Ax = b \\ & x \geq 0 \end{aligned}$$

If the problem is on alternative form it can reformulated by introducing slack variables.

$$\begin{aligned} Ax - y &= b & \text{for } y \geq 0 \\ x &= x^+ - x^- & x^+ \geq 0 \text{ and } x^- \geq 0 \end{aligned}$$

The problem then becomes

$$\hat{x} = \begin{bmatrix} x^+ \\ x^- \\ y \end{bmatrix}, \hat{c} = \begin{bmatrix} c \\ -c \\ 0 \end{bmatrix}, \qquad \hat{A} = \begin{bmatrix} A & -A & -I_p \end{bmatrix}$$

with

$$\begin{aligned} \text{minimize } & f(x) = \hat{c}^T \hat{x} \\ \text{subject to } & \hat{A}\hat{x} = b \\ & \hat{x} \geq 0 \end{aligned}$$

KKT conditions are found in theorem 11.1 on page 324

Polytopes
The feasible region defined by $\mathcal{R} = \{x : Ax \geq b\}$ is in general convex. A set of points $\mathcal{F}$ is said to be a face of $\mathcal{R}$ if $p_1, p_2 \in \mathcal{F} \implies (p_1 + p_2)/2 \in \mathcal{F}$. If l is the dimension of a face $\mathcal{F}$, a facet is an $(l-1)$-dimensional face, an edge is a one-dimensional face and a vertex is a zero dimensional face.

See theorem 11.3 and 11.4 for sufficient and necessary conditions for minimum in LP problem.

Simplex Method
See page 344 in Pract ical Optimization.

Duality and Iterative Methods

Jacobi method
Solve a system of linear equations with the Jacobi method:

$$Ax = b \tag{6}$$

Decompose $A = D + R$ where D is diagonal. The solution x is then obtained by

$$\begin{aligned} x^{(k+1)} &= D^{-1}(b - Rx^{(k)}) \\ x_i^{(k+1)} &= \frac{1}{a_{ii}}\left(b_i - \sum_{j \neq i} a_{ij} x_k^{(k)}\right), \qquad i = 1, \ldots, n \end{aligned}$$

Gauss-Seidel method
Solves the system

$$Ax = b \tag{7}$$

by decomposing A into a lower triangular matrix L_* and a strictly upper triangular matrix U:

$$L_* x = b - Ux \tag{8}$$

The iterations are described by

$$\begin{aligned} x^{(k+1)} &= L_*^{-1}(b - Ux^{(k)} \\ x_i^{(k+1)} &= \frac{1}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)}\right), \qquad i = 1, \ldots, n \end{aligned}$$

Lagrangian Dual Function Given problem

$$\begin{aligned} \text{minimize } & f_0(x) \\ \text{subject to } & f_i(x) \leq 0, \qquad i = 1, \ldots, m \\ & h_i(x) = 0, \qquad i = 1, \ldots, p \end{aligned}$$

Lagrangian function is defined as

$$\mathcal{L}(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{p} \nu_i h_i(x) \tag{9}$$

Lagrangian dual function defined as

$$g(\lambda, \nu) = \inf_{x \in \mathcal{D}} \mathcal{L}(x, \lambda, \nu) = \inf_{x \in \mathcal{D}} \left(f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{p} \nu_i h_i(x)\right) \tag{10}$$

If p^* is optimal value for original (primal) problem, then

$$g(\lambda, \nu) \leq p^* \tag{11}$$

The Lagrange dual problem considers maximizing g for some $\lambda \succeq 0$:

$$\begin{aligned} &\text{maximize } g(\lambda, \nu) \\ &\text{subject to } \lambda \succeq 0 \end{aligned}$$

Weak duality is when the solution to the Lagrange dual problem is less than the solution to the primal problem:

$$d^* \leq p^* \tag{12}$$

Difference between d^* and p^* is called the optimal duality gap. If $d^* = p^*$ it is called strong duality.

- Exact line search
$$t = argmin_{s \geq 0} f(x + s\Delta x) \tag{13}$$

- Backtracking line search: $0 < \alpha < 0.5$, $0 < \beta < 1$.
$$f(x + t\Delta x) > f(x) + \alpha t \nabla f(x)^T \Delta x, \qquad t := \beta t \tag{14}$$

- Gradient descent method
$$\Delta x = -\nabla f(x) \tag{15}$$
Stopping criterion:
$$\|\nabla f(x)\|_2 \tag{16}$$

- Steepest descent
Use Taylor's:
$$f(x + v) \approx \hat{f}(x + v) = f(x) \nabla f(x)^T v \tag{17}$$
The second term is the directional derivative of f. The normalized steepest descent direction with respect to some norm $\|\cdot\|$ is defined as
$$\Delta x_{nsd} = argmin\{\nabla f(x)^T v \,|\, \|v\| = 1\}. \tag{18}$$
Use backtracking line search for distance travelled along the descent.

- Newton Step
$$\Delta x_{nt} = -\nabla^2 f(x)^{-1} \nabla f(x) \quad (19)$$
The Newton step is a descent or optimal from the fact of positive definiteness og the Hessian:
$$\nabla f(x)^T \Delta x_{nt} = -\nabla f(x)^T \nabla^2 f(x)^{-1} \nabla f(x) < 0 \quad (20)$$
Newton step is the steepest descent in the Hessian norm. The Newton step is optimal close to the minimizing point. Use
$$\lambda^2 = \nabla f(x)^T \nabla^2 f(x)^{-1} \nabla f(x) \quad (21)$$
for quitting when $\lambda^2/2 \leq \epsilon$. Use backtracking line search.

Interior-Point Methods

Barrier method
Used for a problem with inequality constraints:

$$\begin{aligned} \text{minimize } & f_0(x) \\ \text{subject to } & f_i(x) \leq 0, \qquad\qquad i = 1, \dots, m \\ & Ax = b \end{aligned}$$

Assume that the problem is strictly feasible meaning there exists $x \in \mathcal{D}$ such that $f_i(x) < 0 \; \forall i$. With Slater's condition dual optimal λ^* and ν^* exist and KKT conditions are fulfilled:

$$\begin{aligned} Ax^* = b, \qquad f_i(x^*) &\leq 0, \qquad\qquad i = 1, \dots m \\ \lambda^* &\succeq 0 \\ \nabla f_0(x^*) + \sum_{i=1}^{m} \lambda_i^* \nabla f_i(x^*) + A^T \nu^* &= 0 \\ \lambda_i^* f_i(x^*) &= 0, \qquad\qquad i = 1, \dots m \end{aligned}$$

Example:

$$\begin{aligned} \text{minimize } & f_0(x) + \sum_{i=1}^{m} I_- f_i(x)) \\ \text{subject to } & Ax = b \end{aligned}$$

where $I_-(u) = 0$ for $u \leq 0$ and $I_-(u) = \infty$ otherwise. $I_-(u)$ is approximated by a logarithmic function:

$$\hat{I}_-(u) = -(1/t) \log(-u), \qquad \mathbf{dom} \hat{I}_- = -\mathbb{R}_{>0} \tag{22}$$

Use Newton's method to solve. where $t > 0$ sets accuracy. $\hat{I}$ is convex, differentiable and increasing and therefore easier to handle. The central path is the path for x towards the optimal value of the original problem, when $t \to \infty$. The problem described by

$$\begin{aligned} \text{minimize } & t f_0(x) + \phi(x) \\ \text{subject to } & Ax = b \end{aligned}$$

has central points $x^*(t)$ on the central path. $x^*(t)$ is found by

$$t \nabla f_0(x^*(t)) + \sum_{i=1}^{m} \frac{1}{-f_i(x^*(t))} \nabla f_i(x^*(t)) + A^T \hat{\nu} \tag{23}$$

Dual points and convergence
Define

$$\lambda_i^*(t) = \frac{1}{t f_i(x^*(t))} \qquad \nu^*(t) = \hat{\nu}/t \tag{24}$$

These are dual feasible points and therefore yield a lower bound from the dual function:

$$\begin{aligned} p^* = g(\lambda^*, \nu^*) &= f_0(x^*(t)) + \sum_{i=1}^{m} \lambda_i^*(t) f_i(x^*(t)) + \nu^*(t)^T (A x^*(t) - b) \\ &= f_0(x^*(t)) - m/t \end{aligned}$$

As such the duality gap is

$$f_0(x^*(t)) - p^* = m/t \tag{25}$$

Alternating Direction Methods of Multipliers (ADMM)

Solves problems of the form

$$\begin{aligned} &\text{minimize } f(x) + g(z) \\ &\text{subject to } Ax + Bz = c \end{aligned}$$

Utilizes the splitting of functions (dual ascent property) and superior convergence (method of multipliers). optimal value denoted by

$$p^* = \inf\{f(x) + g(z) | Ax + Bz = c\} \tag{26}$$

Form augmented Lagrangian and update steps:

$$\begin{aligned} \mathcal{L}_\rho(x, z, y) &= f(x) + g(x) + y^T(Ax + Bz - c) + \frac{\rho}{2}\|Ax + Bz - c\|_2^2 \\ x^{k+1} &= argmin \mathcal{L}_\rho(x, z^k, y^k) \\ z^{k+1} &= argmin \mathcal{L}_\rho(x^{k+1}, z, y^k) \\ y^{k+1} &= y^k + \rho(Ax^{k+1} + Bz^{k+1} - c) \end{aligned}$$

Convergence

- *Residual Convergence*: $r^k \to 0$ as $k \to \infty$
- *Objective convergence*: $f(x^k) + g(z^k) \to p^*$ as $k \to \infty$
- *Dual variable convergence*: $y^k \to y^*$ as $k \to \infty$

Optimality conditions Necessary and sufficient conditions include primal feasibility

$$Ax^* + Bz^* - c = 0 \tag{27}$$

and dual feasibility

$$\begin{aligned} 0 &\in \partial f(x^*) + A^T y^* \\ 0 &\in \partial g(x^*) + B^T y^* \end{aligned}$$

Stopping criteria

$$\begin{aligned} \|r^k\|_2 &= \|Ax + Bz - c\|_2 \leq \epsilon^{pri} \\ \|s^k\|_2 &= \|\rho A^T B(z^{k+1} - z^k)\|_2 \leq \epsilon^{dual} \end{aligned}$$

Semidefinite Relaxation

Quadratically constrained quadratic problem
Objective function and constraints are quadratic functions:

$$
\begin{aligned}
&\text{minimize } \tfrac{1}{2}x^T P_0 x + q_0^T x \\
&\text{subject to } \tfrac{1}{2}x^T P_i x + q_i^T x + r_i \leq 0 \qquad i = 1, \dots, m \\
&\qquad Ax = b
\end{aligned}
$$

where P_i are positive semidefinite and symmetric matrices. For problems of the form

$$
\begin{aligned}
&\text{minimize } x^T C x \\
&\text{subject to } x^T F_i x \geq g_i, \qquad i = 1, \dots, p \\
&\qquad x^T H_i x, \qquad i = 1, \dots, q
\end{aligned}
$$

semidefinite relaxation is used. Use

$$
\begin{aligned}
x^T C x &= \mathrm{Tr}(x^T C x) = \mathrm{Tr}(C x x^T) \\
x^T A_i x &= \mathrm{Tr}(x^T A_i x) = \mathrm{Tr}(A_i x x^T)
\end{aligned}
$$

such that

$$
\begin{aligned}
&\text{minimize } \mathrm{Tr}(CX) \\
&\text{subject to } \mathrm{Tr}(A_i X) \leq \ \textbf{or} \ \geq \ \textbf{or} \ = b_i, \qquad i = 1, \dots, m \\
&\qquad X \succeq 0 \\
&\qquad \mathrm{rank}(X) = 1
\end{aligned}
$$

When the assumption of $\mathrm{rank}(X) = 1$ is discarded it is called semidefinite or rank-1 relaxation. Finding the optimal value X^* by eigenvalue decomposition:

$$
\begin{aligned}
X^* &= \sum_{i=1}^{r} \lambda_i q_i q_i^T \\
x &\approx \sqrt{\lambda_1} q_1
\end{aligned}
$$

Randomization
By using $xx^T = X$ as the covariance matrix for a randoom vector $\xi \sim \mathcal{N}(0, X)$ we get the stochastic QCQP

$$
\begin{aligned}
&\text{minimize } \mathbb{E}[\xi^T C \xi] \\
&\text{subject to } \mathbb{E}[\xi^T A_i \xi] \leq \ \textbf{or} \ \geq \ \textbf{or} \ = b_i, \qquad i = 1, \dots, m
\end{aligned}
$$

Use the trace:

$$\mathrm{Tr}(\mathbb{E}[\xi^T C\xi]) = \mathbb{E}[\mathrm{Tr}(\xi^T C\xi)] = \mathbb{E}[\mathrm{Tr}(Cxx^T)] = \mathrm{Tr}(Cxx^T) = \mathrm{Tr}(CX) \quad (28)$$

As such the problem can be expressed as

$$\begin{aligned} &\text{minimize } \mathrm{Tr}(CX) \\ &\text{subject to } \mathrm{Tr}(A_i X) \leq \ \textbf{or} \ \geq \ \textbf{or} \ = b_i, \qquad i = 1, \ldots, m \end{aligned}$$

If this is solve for X then $\xi \sim \mathcal{N}(0, X)$ can be generated as solution.

Simulated Annealing

Hillclimbing
If the trajectory searched is not monotonically decreasing we need to use hillclimbing.

Metropolis Criterion Used in simulated annealing.

$$P(\text{accept } s_j) = \exp\left(\frac{f(s_i) - (s_j)}{k_B T}\right) \tag{29}$$

where k_B Bolzmann's contant and T is temperature. Simplify to

$$\frac{f(s_i) - f(s_j)}{c_l} \tag{30}$$

where $c_l > 0$ is the temperature at count l. For every temperature L transitions are generated and evaluated.

Cooling strategy
Initial value of c:

$$\chi(c) = \frac{\#\text{ accepted transitions}}{\#\text{ generated transitions}} \tag{31}$$

Find $\chi(x)$ such that proposed solutions are accepted 95-99% of the time.
Stopping value c_{stop} can be:

- Predefined – stops at a certain temperature.
- When cost has not improved over the last N iterations.

There exist different models for the cooling process (decrease of c). There has to be made a trade-off between runtime and quality of solution.

Transition function

- It is problem dependent
- It should be efficient
- Is should generate valid solutions s_j
- It should be able to generate $s_j \in N(s_i)$ – points in the neighbourhood of s_i

Genetic Algorithms

Pheno-types
Real world populations or genetics. Need to be translated in order to abstract from the physical system.

Geno-types
Encoded pheno-types, for example numbers in a string.

Genetic encoding
The process of encoding pheno-types to geno-types.

Chromosome
A chromosome is a string of genes and is a solution to the problem.

Gene
Part of a chromosome.

Genetic alphabet
The differeent feasible alleles for the problem. Contains the numbers to be found as genes in the chromosomes. **Cost function**
The cost function is typically sought minimzed while fitness is sought maximized. This needs to be taken into account.

The GA-cycle
See bottom figure.

Selection

- *Roulette wheel*

$$p_{select}(genotype_i) = \frac{\text{fitness}(genotype_i)}{\text{total fitness}}$$

$$\text{total fitness} = \sum_{i=1}^{POP} \text{fitness}(genotype_i)$$

  - Pros: Easy to understand and implement. Provides fair selection.
  - Cons: Cannot guarantee that the most fit geno-type is transferred to the mating pool.

- *Elite selection*
  Make sure that the best geno-types found so far are copied into the mating pool. If old geno-types have higher fitness values than the maximum fitness value of the current mating pool then copy them to the current mating pool.

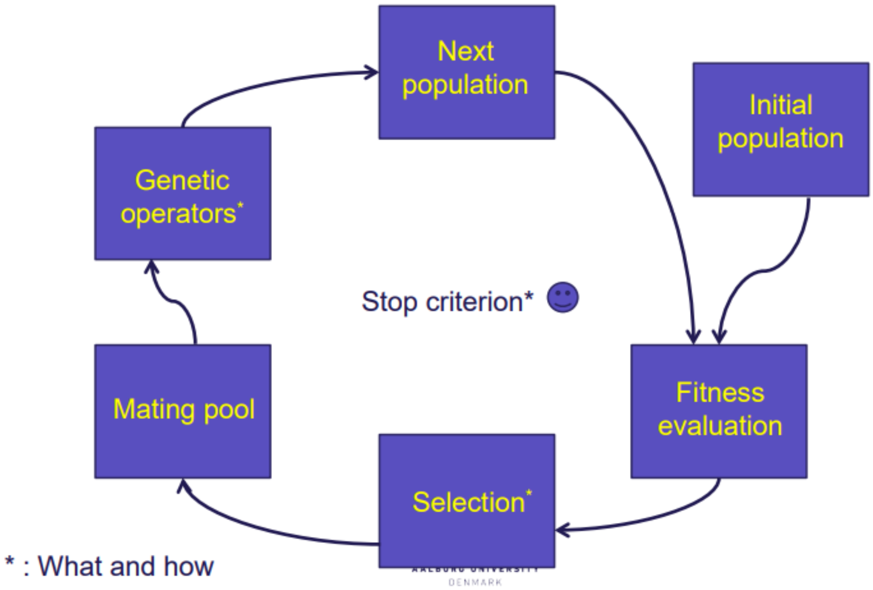
- *Tournament selection*
  Geno-types are selected by running tournaments of randomly selected geno-types from the population pool. The geno-type with the highest fitness is put into the mating pool. The number and size of the tournaments can be adjusted – larger tournaments give less fit geno-types less chance to be selected.

Genetic operators
See bottom figure.

- *Cross-over*
  Two parents are picked from the mating pool. A number of genes are swithced between them resulting in two new chromosomes – children. Any number and order of the children can be selected to proceed. even though new solutions are created no new genetic material is generated and we may get trapped in a local optimum – inbreeding.

- *Mutation*
  Change (mutate) any number of genes in order to introduce new genetic material.

- *Copying*
  Makes sure that all parameters are not changed all at once. Merely copies existing solutions into the new population.



Genetic Operators – overview