1. Theory and algorithms for nonlinear programming

Armin Nurkanović

Systems Control and Optimization Laboratory, University of Freiburg, Germany
(slides create jointly with **Moritz Diehl**)

Winter School on Numerical Methods for Optimal Control of Nonsmooth Systems
École des Mines de Paris
February 3-5, 2025, Paris, France

universität freiburg

Outline of the lecture

What is an optimization problem?

Optimization is a powerful tool used in all quantitative sciences.

Minimize (or maximize) an objective function $F(w)$ depending on decision variables w subject to equality and/or inequality constrains

What is an optimization problem?

Optimization is a powerful tool used in all quantitative sciences.

Minimize (or maximize) an objective function $F(w)$ depending on decision variables w subject to equality and/or inequality constrains

An optimization problem

$$\min_{w \in \mathbb{R}^n} \; F(w) \tag{1a}$$

$$\text{s.t.} \; G(w) = 0 \tag{1b}$$

$$H(w) \geq 0 \tag{1c}$$

Terminology

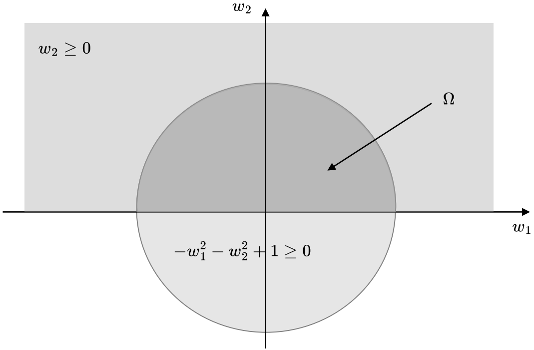
- $w \in \mathbb{R}^n$ - decision variable
- $F : \mathbb{R}^n \to \mathbb{R}$ - objective
- $G : \mathbb{R}^n \to \mathbb{R}^{n_G}$ - equality constraints
- $H : \mathbb{R}^n \to \mathbb{R}^{n_H}$ - inequality constraints

- If F, G, H are nonlinear and smooth, we speak of a *nonlinear programming problem* (NLP).
- Only in few special cases a closed form solution exists.
- Use an iterative algorithm to find an approximate solution.
- Problem may be parametric, and some (or all) functions depend on a fixed parameter $p \in \mathbb{R}^p$, e.g. model predictive control.

Basic definitions: the feasible set

Definition

The feasible set of the optimization problem (1) is defined as
$\Omega = \{w \in \mathbb{R}^n \mid G(w) = 0, H(w) \geq 0\}$. A point $w \in \Omega$ is is called a feasible point.



In the example, the feasible set is the intersection of the two grey areas (halfspace and circle).

Basic definitions: local and global minimizer

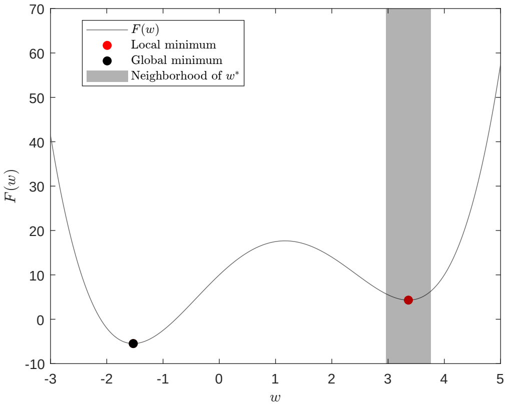
Definition (Local minimizer)

A point $w^* \in \Omega$ is called a **local minimizer** of the optimization problem (1) if there exists an open ball $\mathcal{B}_\epsilon(w^*)$ with $\epsilon > 0$, such that for all $w \in \mathcal{B}_\epsilon(w^*) \cap \Omega$ it holds that $F(w) \geq F(w^*)$.

Definition (Global minimizer)

A point $w^* \in \Omega$ is called a **global minimizer** of (1) if for all $w \in \Omega$ it holds that $F(w) \geq F(w^*)$.

- The value $F(w^*)$ at a local/global **minimizer** w^* is called local/global **minimum**.



$$F(w) = \frac{1}{2}w^4 - 2w^3 - 3w^2 + 12w + 10$$

Convex sets

A key concept in optimization is convexity.

A set Ω is said to be convex if for any w_1, w_2 and any $\theta \in [0,1]$ it holds $\theta w_1 + (1-\theta) w_2 \in \Omega$

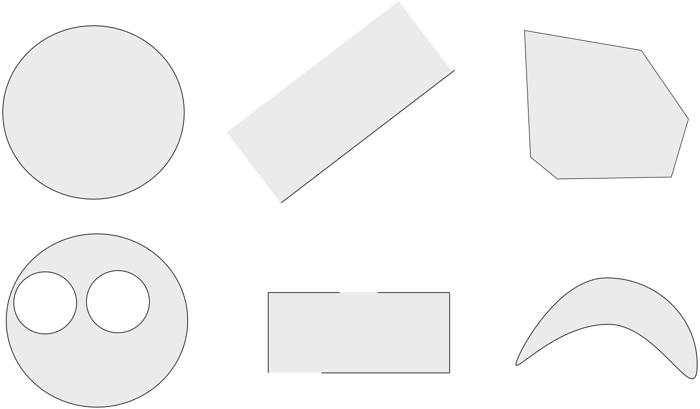


Figure inspired by Figure 2.2 in S. Boyd and L. Vandenberghe. Convex optimization. Cambridge university press, 2004.

Convex functions

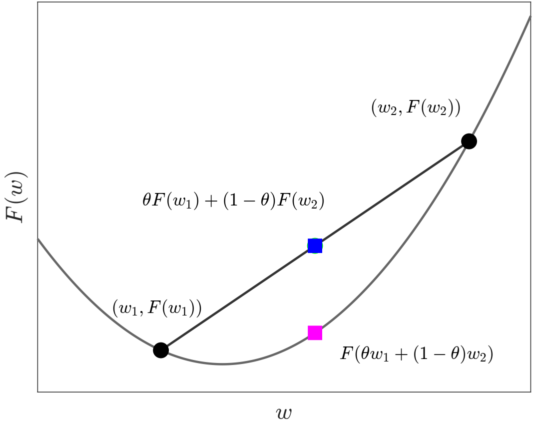
- A function F is convex if for every $w_1, w_2 \in \mathbb{R}^n$ and $\theta \in [0,1]$ it holds that

$$F(\theta w_1 + (1-\theta) w_2) \leq \theta F(w_1) + (1-\theta) F(w_2)$$

- F is concave if and only if $-F$ is convex
- F is convex if and only if the epigraph

$$\text{epi} F = \{(w,t) \in \mathbb{R}^{n_w+1} \mid F(w) \leq t\}$$

is a convex set



Convex optimization problems

A convex optimization problem

$$\begin{aligned} \min_{w \in \mathbb{R}^n} \; & F(w) \\ \text{s.t.} \; & G(w) = 0 \\ & H(w) \geq 0 \end{aligned}$$

An optimization problem is convex if the objective function F is convex and the feasible set Ω is convex.

- For convex problems, every locally optimal solution is globally optimal.
- First-order optimality conditions are necessary and sufficient.
- Many iterative algorithms for nonconvex optimization solve a sequence of convex optimization problems.

Convex optimization problems

A convex optimization problem

$$\begin{aligned} \min_{w \in \mathbb{R}^n} \ & F(w) \\ \text{s.t.} \ & G(w) = 0 \\ & H(w) \geq 0 \end{aligned}$$

An optimization problem is convex if the objective function F is convex and the feasible set Ω is convex.

- For convex problems, every locally optimal solution is globally optimal.
- First-order optimality conditions are necessary and sufficient.
- Many iterative algorithms for nonconvex optimization solve a sequence of convex optimization problems.

"...in fact, the great watershed in optimization isn't between linearity and nonlinearity, but convexity and nonconvexity." R. T. Rockafellar, SIAM Review, 1993

Outline of the lecture

Some classifications of optimization problems

Optimization problems can be:

- unconstrained ($\Omega = \mathbb{R}^n$) or constrained ($\Omega \subset \mathbb{R}^n$)
- convex or nonconvex
- linear or nonlinear
- differentiable or nonsmooth
- continuous or (mixed-)integer
- finite or infinite dimensional

Some classifications of optimization problems

Optimization problems can be:

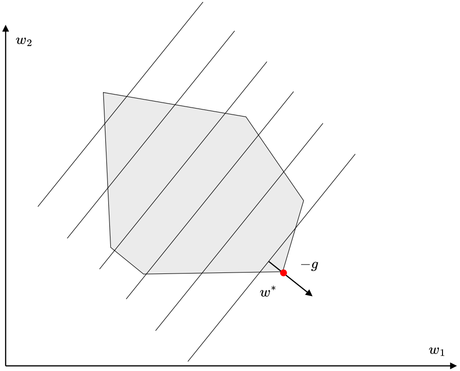
- unconstrained ($\Omega = \mathbb{R}^n$) or constrained ($\Omega \subset \mathbb{R}^n$)
- convex or nonconvex
- linear or nonlinear
- differentiable or nonsmooth
- continuous or (mixed-)integer
- finite or infinite dimensional

"... the main fact, which should be known to any person dealing with optimization models, is that in general, optimization problems are unsolvable."
Yurii Nesterov, Lectures on Convex Optimization, 2018.
("solvable" refers to finding a global minimizer)

Class 1: Linear Programming (LP)

Linear program

$$\min_{w \in \mathbb{R}^n} \; g^\top w$$
$$\text{s.t.} \quad Aw - b = 0$$
$$Cw - d \geq 0$$

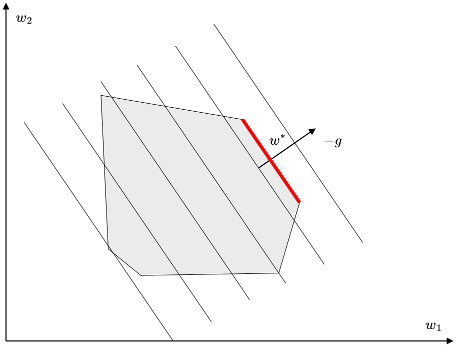


- convex optimization problem
- 1947: simplex method by Dantzig, 1984: polynomial time interior-point method by Karmarkar
- a solution is always at a vertex of the feasible set (possibly a whole facet if nonunique)
- very mature and reliable

Class 1: Linear Programming (LP)

Linear program

$$\min_{w \in \mathbb{R}^n} \; g^\top w$$
$$\text{s.t.} \quad Aw - b = 0$$
$$\qquad Cw - d \geq 0$$

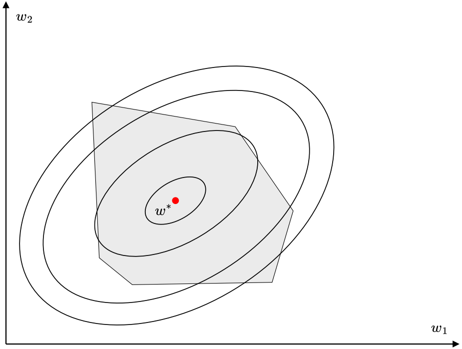


- convex optimization problem
- 1947: simplex method by Dantzig, 1984: polynomial time interior-point method by Karmarkar
- a solution is always at a vertex of the feasible set (possibly a whole facet if nonunique)
- very mature and reliable

Class 2: Quadratic Programming (QP)

Quadratic program

$$\begin{aligned} \min_{w \in \mathbb{R}^n} \quad & \frac{1}{2} w^\top Q w + g^\top w \\ \text{s.t.} \quad & Aw - b = 0 \\ & Cw - d \geq 0 \end{aligned}$$

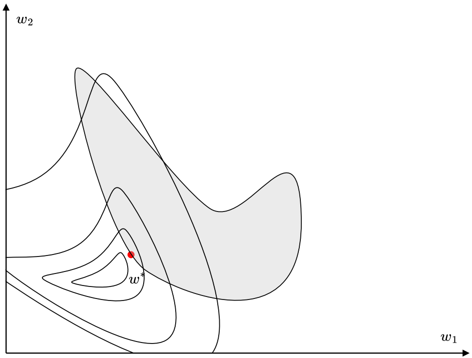


- depending on Q, can be convex and nonconvex
- many good solvers: `OSQP`, `HPIPM`, `qpOASES`, `Gurobi`, `Clarabel`, `DAQP`, `OOQP`, `MOSEK`, ...
- solved online in linear model predictive control
- subsproblems in nonlinear optimization

Class 3: Nonlinear Program (NLP)

Nonlinear programming problem

$$\begin{aligned} \min_{w \in \mathbb{R}^n} \quad & F(w) \\ \text{s.t.} \quad & G(w) = 0 \\ & H(w) \geq 0 \end{aligned}$$



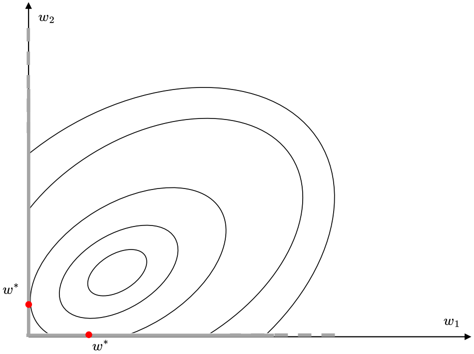
- F, G, H smooth functions, can be convex and nonconvex
- solved with iterative Newton-type algorithms
- solved in nonlinear model predictive control

Class 4: Mathematical programs with Complementarity Constraints (MPCC)

MPCC

$$\begin{aligned} \min_{w \in \mathbb{R}^n} \quad & F(w) \\ \text{s.t.} \quad & G(w) = 0 \\ & H(w) \geq 0 \\ & 0 \leq w_1 \perp w_2 \geq 0 \end{aligned}$$

$$w = [w_0^\top, w_1^\top, w_2^\top]^\top \in \mathbb{R}^n$$



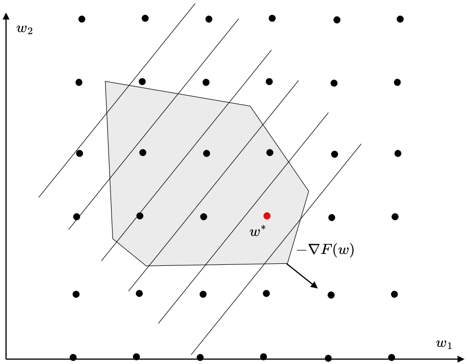
- more difficult than standard nonlinear programming
- feasible set is inherently nonsmooth and nonconvex
- powerful modeling concept
- requires specialized theory and algorithms (Lectures 5 and 6 on Wednesday)

Class 5: Mixed-integer programming

Mixed-integer nonlinear program (MINLP)

$$\begin{aligned} \min_{w_0 \in \mathbb{R}^p, w_1 \in \mathbb{Z}^q} \quad & F(w) \\ \text{s.t.} \quad & G(w) = 0 \\ & H(w) \geq 0 \end{aligned}$$

$w = [w_0^\top, w_1^\top]^\top, n = p + q$



- inherently nonconvex feasible set
- due to combinatorial nature, NP-hard even for linear F, G, H
- branch and bound, branch and cut algorithms based on iterative solution of relaxed continuous problems

Class 6: Continuous time optimal control problems (OCP)

Continuous-time Optimal Control Problem

$$
\begin{aligned}
\min_{x(\cdot),u(\cdot)} \quad & \int_0^T L_{\mathrm{c}}(x(t),u(t))\,\mathrm{d}t + E(x(T)) \\
\text{s.t.} \quad & x(0) = \bar{x}_0 \\
& \dot{x}(t) = f_{\mathrm{c}}(x(t),u(t)) \\
& 0 \geq h(x(t),u(t)), \; t \in [0,T] \\
& 0 \geq r(x(T))
\end{aligned}
$$

- decision variables $x(\cdot)$, $u(\cdot)$ in infinite dimensional function space
- infinitely many constraints for $t \in [0,T]$
- smooth ordinary differential equations (ODE)

$$\dot{x}(t) = f_{\mathrm{c}}(x(t),u(t))$$

- more generally, dynamic model can be based on
  - differential algebraic equations (DAE)
  - partial differential equations (PDE)
  - stochastic ODE
  - **nonsmooth ODE** - (treated on Tuesday)
- OCP can be convex or nonconvex
- all or some components of $u(t)$ may take integer values (mixed-integer OCP)

Direct optimal control methods solve Nonlinear Programs (NLP)

Treated in detail in the 2nd lecture.

Continuous time OCP

$$
\begin{aligned}
\min_{x(\cdot),u(\cdot)} \quad & \int_0^T L_{\mathrm{c}}(x(t),u(t))\,\mathrm{d}t + E(x(T)) \\
\text{s.t.} \quad & x(0) = \bar{x}_0 \\
& \dot{x}(t) = f_{\mathrm{c}}(x(t),u(t)) \\
& 0 \geq h(x(t),u(t)),\ t \in [0,T] \\
& 0 \geq r(x(T))
\end{aligned}
$$

Direct methods like direct collocation, multiple shooting first discretize, then optimize.

Direct optimal control methods solve Nonlinear Programs (NLP)

Treated in detail in the 2nd lecture.

Continuous time OCP

$$
\begin{aligned}
\min_{x(\cdot),u(\cdot)} \quad & \int_0^T L_\mathrm{c}(x(t),u(t))\,\mathrm{d}t + E(x(T)) \\
\text{s.t.} \quad & x(0) = \bar{x}_0 \\
& \dot{x}(t) = f_\mathrm{c}(x(t),u(t)) \\
& 0 \geq h(x(t),u(t)),\ t \in [0,T] \\
& 0 \geq r(x(T))
\end{aligned}
$$

Direct methods like direct collocation, multiple shooting first discretize, then optimize.

Discrete time OCP (an NLP)

$$
\begin{aligned}
\min_{x,u} \quad & \sum_{k=0}^{N-1} \ell(x_k,u_k) + E(x_N) \\
\text{s.t.} \quad & x_0 = \bar{x}_0 \\
& x_{k+1} = f(x_k,u_k) \\
& 0 \geq h(x_k,u_k),\ k = 0,\dots,N-1 \\
& 0 \geq r(x_N)
\end{aligned}
$$

Variables $x = (x_0,\dots,x_N)$ and $u = (u_0,\dots,u_{N-1})$ can be summarized in vector $w = (x,u) \in \mathbb{R}^n$.

Direct optimal control methods solve Nonlinear Programs (NLP)

Treated in detail in the 2nd lecture.

Discrete time OCP (an NLP)

$$
\begin{aligned}
\min_{x,u} \ & \sum_{k=0}^{N-1} \ell(x_k, u_k) + E(x_N) \\
\text{s.t.} \quad & x_0 = \bar{x}_0 \\
& x_{k+1} = f(x_k, u_k) \\
& 0 \geq h(x_k, u_k), \ k = 0, \dots, N-1 \\
& 0 \geq r(x_N)
\end{aligned}
$$

Variables $x = (x_0, \dots, x_N)$ and $u = (u_0, \dots, u_{N-1})$ can be summarized in vector $w = (x, u) \in \mathbb{R}^n$.

Nonlinear MPC solves Nonlinear Programs (NLP)

Discrete time NMPC Problem (an NLP)

$$\begin{aligned}\min_{x,u} \quad & \textstyle\sum_{k=0}^{N-1} \ell(x_k, u_k) + E(x_N) \\ \text{s.t.} \quad & x_0 = \bar{x}_0 \\ & x_{k+1} = f(x_k, u_k) \\ & 0 \geq h(x_k, u_k), \; k = 0, \dots, N-1 \\ & 0 \geq r(x_N)\end{aligned}$$

Variables $x = (x_0, \dots, x_N)$ and $u = (u_0, \dots, u_{N-1})$ can be summarized in vector $w = (x, u) \in \mathbb{R}^n$.

Nonlinear MPC solves Nonlinear Programs (NLP)

Discrete time NMPC Problem (an NLP)

$$\begin{aligned}\min_{x,u}\ & \textstyle\sum_{k=0}^{N-1} \ell(x_k, u_k) + E(x_N)\\ \text{s.t.}\quad & x_0 = \bar{x}_0\\ & x_{k+1} = f(x_k, u_k)\\ & 0 \geq h(x_k, u_k),\ k = 0, \ldots, N-1\\ & 0 \geq r(x_N)\end{aligned}$$

Variables $x = (x_0, \ldots, x_N)$ and $u = (u_0, \ldots, u_{N-1})$ can be summarized in vector $w = (x, u) \in \mathbb{R}^n$.

Nonlinear Program (NLP)

$$\begin{aligned}\min_{w \in \mathbb{R}^n}\ & F(w)\\ \text{s.t.}\ & G(w) = 0\\ & H(w) \geq 0\end{aligned}$$

Outline of the lecture

Algebraic characterization of **unconstrained** local optima

Consider the unconstrained problem: $\min_{w \in \mathbb{R}^n} \quad F(w)$

First-Order Necessary Condition of Optimality (FONC) (in convex case also sufficient)

w^* local optimizer $\quad \Rightarrow \quad \nabla F(w^*) = 0,\ w^*$ stationary point

Second-Order Necessary Condition of Optimality (SONC)

w^* local minimizer $\quad \Rightarrow \quad \nabla^2 F(w^*) \succeq 0$

Algebraic characterization of **unconstrained** local optima

Consider the unconstrained problem: $\min_{w \in \mathbb{R}^n} \quad F(w)$

First-Order **Necessary** Condition of Optimality (FONC) (in convex case also sufficient)

w^* local optimizer $\quad \Rightarrow \quad \nabla F(w^*) = 0,\ w^*$ stationary point

Second-Order **Necessary** Condition of Optimality (SONC)

w^* local minimizer $\quad \Rightarrow \quad \nabla^2 F(w^*) \succeq 0$

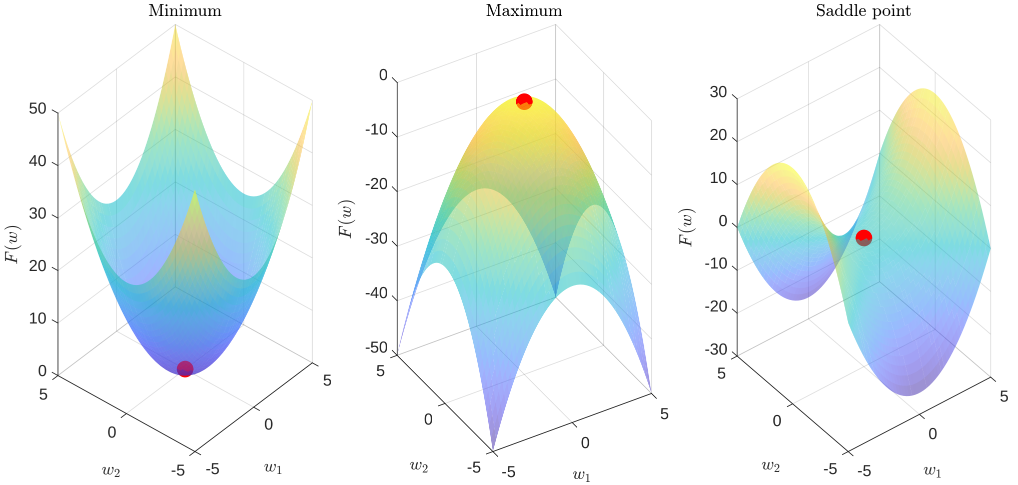
Second-Order **Sufficient** Conditions of Optimality (SOSC)

$\nabla F(w^*) = 0$ and $\nabla^2 F(w^*) \succ 0 \quad \Rightarrow \quad x^*$ strict local minimizer

$\nabla F(w^*) = 0$ and $\nabla^2 F(w^*) \prec 0 \quad \Rightarrow \quad x^*$ strict local maximizer

no conclusion can be drawn in the case $\nabla^2 F(w^*)$ is indefinite.

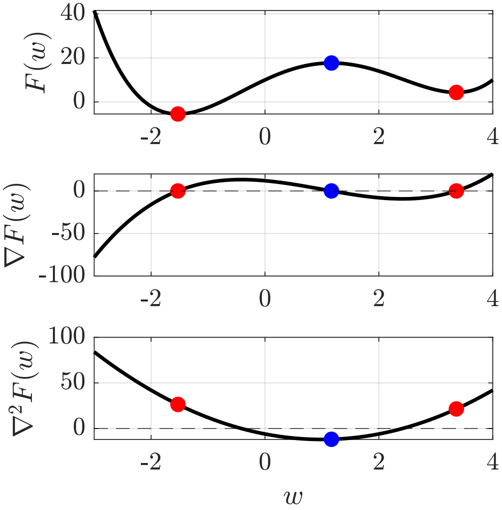
Type of stationary points



a stationary point w^* with $\nabla F(w^*) = 0$ can be a minimizer, a maximizer, or a saddle point

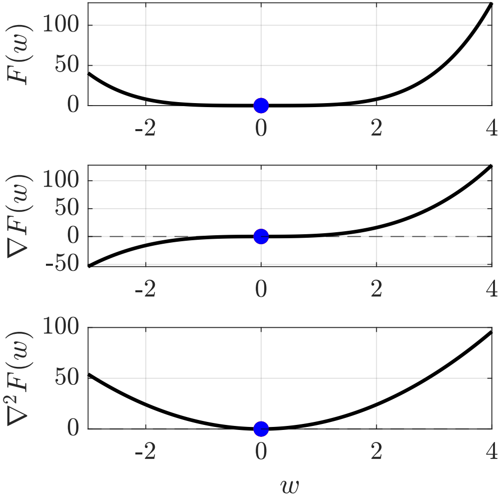
Optimality conditions - unconstrained

- Necessary conditions: find a candidate point (or to exclude points)
- Sufficient conditions: verify optimality of a candidate point



Optimality conditions - unconstrained

- Necessary conditions: find a candidate point (or to exclude points)
- Sufficient conditions: verify optimality of a candidate point
- A minimizer must satisfy SONC, but does not have to satisfy SOSC



First-order necessary conditions for equality constrained optimization

Nonlinear Program (NLP)

$$\min_{w \in \mathbb{R}^n} \; F(w)$$
$$\text{s.t.} \; G(w) = 0$$

Lagrangian function: $\mathcal{L}(w, \lambda) = F(w) - \lambda^\top G(w)$

First-order necessary conditions for equality constrained optimization

Nonlinear Program (NLP)

$$\min_{w \in \mathbb{R}^n} \; F(w)$$
$$\text{s.t. } G(w) = 0$$

Lagrangian function: $\mathcal{L}(w, \lambda) = F(w) - \lambda^\top G(w)$

Definition (LICQ)

A point w satisfies Linear Independence Constraint Qualification (LICQ) if $\nabla G(w) := \frac{\partial G(w)}{\partial w}^\top$ is full column rank.

First-order necessary conditions for equality constrained optimization

Nonlinear Program (NLP)

$$\min_{w \in \mathbb{R}^n} F(w)$$
$$\text{s.t. } G(w) = 0$$

Lagrangian function: $\mathcal{L}(w, \lambda) = F(w) - \lambda^\top G(w)$

Definition (LICQ)

A point w satisfies Linear Independence Constraint Qualification (LICQ) if $\nabla G(w) := \frac{\partial G(w)}{\partial w}^\top$ is full column rank.

First-order necessary conditions (in convex case also sufficient)

Let F, G in $\mathcal{C}^1$. If w^* is a (local) minimizer, **and** w^* satisfies LICQ, then there is a unique vector λ such that:

$$\nabla_w \mathcal{L}(w^*, \lambda^*) = \nabla F(w^*) - \nabla G(w^*)\lambda = 0 \qquad \text{dual feasibility}$$
$$\nabla_\lambda \mathcal{L}(w^*, \lambda^*) = G(w^*) = 0 \qquad \text{primal feasibility}$$

The Karush-Kuhn-Tucker (KKT) conditions

Nonlinear Program (NLP)

$$\begin{aligned} \min_{w \in \mathbb{R}^n} \; & F(w) \\ \text{s.t.} \; & G(w) = 0 \\ & H(w) \geq 0 \end{aligned}$$

$$\mathcal{L}(w, \lambda, \mu) = F(w) - \lambda^\top G(w) - \mu^\top H(w)$$

The Karush-Kuhn-Tucker (KKT) conditions

Nonlinear Program (NLP)

$$\begin{aligned}\min_{w \in \mathbb{R}^n} \; & F(w) \\ \text{s.t. } & G(w) = 0 \\ & H(w) \geq 0\end{aligned}$$

$$\mathcal{L}(w, \lambda, \mu) = F(w) - \lambda^\top G(w) - \mu^\top H(w)$$

Definition (LICQ)

A point w satisfies LICQ if

$$[\nabla G(w), \quad \nabla H_{\mathcal{A}}(w)]$$

is full column rank.

Active set $\mathcal{A} = \{i \mid H_i(w) = 0\}$

The Karush-Kuhn-Tucker (KKT) conditions

Nonlinear Program (NLP)

$$\min_{w \in \mathbb{R}^n} \; F(w)$$
$$\text{s.t.} \; G(w) = 0$$
$$H(w) \geq 0$$

$$\mathcal{L}(w, \lambda, \mu) = F(w) - \lambda^\top G(w) - \mu^\top H(w)$$

Definition (LICQ)

A point w satisfies LICQ if

$$[\nabla G(w), \quad \nabla H_{\mathcal{A}}(w)]$$

is full column rank.

Active set $\mathcal{A} = \{i \mid H_i(w) = 0\}$

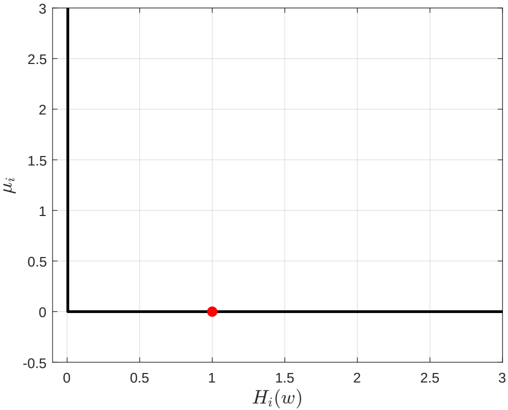
Theorem (KKT conditions - FONC for constrained optimization)

Let F, G, H be $\mathcal{C}^1$. If w^* is a (local) minimizer ***and*** *satisfies LICQ, then there are unique vectors λ^* and μ^* such that (w^*, λ^*, μ^*) satisfies:*

$\nabla_w \mathcal{L}(w^*, \mu^*, \lambda^*) = 0, \quad \mu^* \geq 0,$	*dual feasibility*
$G(w^*) = 0, \quad H(w^*) \geq 0$	*primal feasibility*
$\mu_i^* H_i(w^*) = 0, \quad \forall\, i$	*complementary slackness*

The complementary slackness condition

- complementarity conditions
  $0 \leq \mu^* \perp H(w^*) \geq 0$
- same as $\min(\mu^*, H(w^*)) = 0$
- zero level set of $\min$ is an L-shaped set

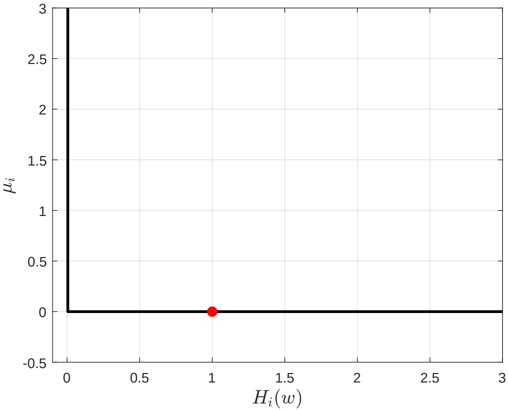


The complementary slackness condition

- complementarity conditions
  $0 \leq \mu^* \perp H(w^*) \geq 0$
- same as $\min(\mu^*, H(w^*)) = 0$
- zero level set of $\min$ is an L-shaped set

Cases:

- $H_i(w^*) > 0$ then $\mu_i^* = 0$, and $H_i(w)$ is inactive

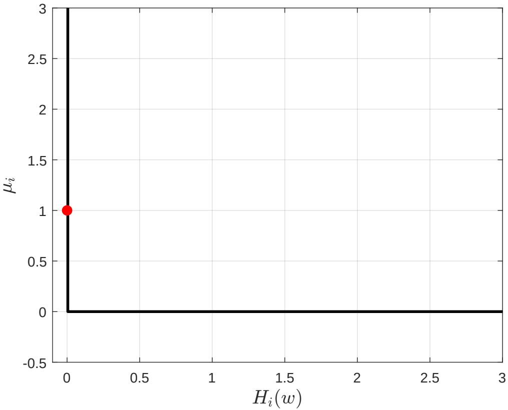


The complementary slackness condition

- complementarity conditions
  $0 \leq \mu^* \perp H(w^*) \geq 0$
- same as $\min(\mu^*, H(w^*)) = 0$
- zero level set of $\min$ is an L-shaped set

Cases:

- $H_i(w^*) > 0$ then $\mu_i^* = 0$, and $H_i(w)$ is inactive
- $\mu_i^* > 0$ and $H_i(w) = 0$ then $H_i(w)$ is strictly active

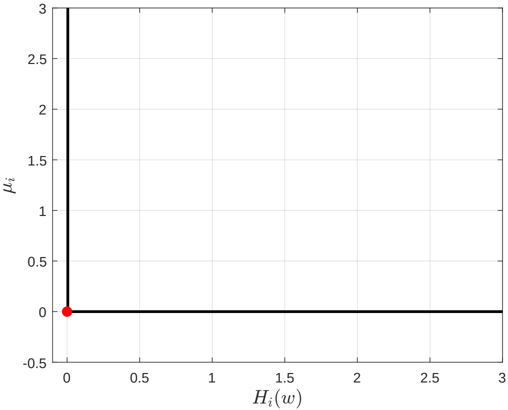


The complementary slackness condition

- complementarity conditions
  $0 \leq \mu^* \perp H(w^*) \geq 0$
- same as $\min(\mu^*, H(w^*)) = 0$
- zero level set of $\min$ is an L-shaped set

Cases:

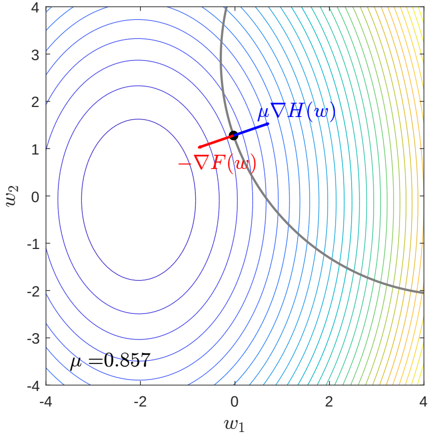
- $H_i(w^*) > 0$ then $\mu_i^* = 0$, and $H_i(w)$ is inactive
- $\mu_i^* > 0$ and $H_i(w) = 0$ then $H_i(w)$ is strictly active
- $\mu_i^* = 0$ and $H_i(w) = 0$ then then $H_i(w)$ is weakly active



Some intuitions on the KKT conditions

Ball rolling down a valley blocked by a fence - test problem with two variables and one inequality constraint

$$\min_{w \in \mathbb{R}^n} \; F(w)$$
$$\text{s.t.} \; H(w) \geq 0$$



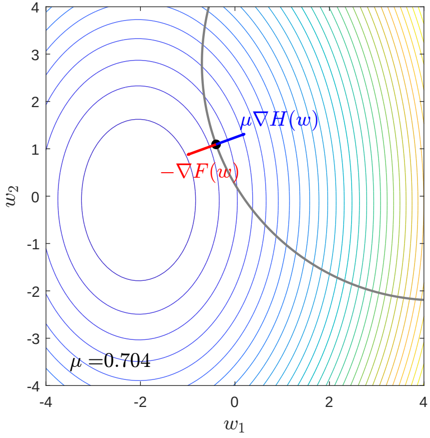
Animation inspired by Lecture 2 of the Winter School on Numerical Optimal Control with Differential Algebraic Equations by S. Gros and M. Diehl, Freiburg, 2016

Some intuitions on the KKT conditions

Ball rolling down a valley blocked by a fence - test problem with two variables and one inequality constraint

$$\min_{w \in \mathbb{R}^n} \; F(w)$$
$$\text{s.t. } H(w) \geq 0$$

- $-\nabla F$ is the gravity



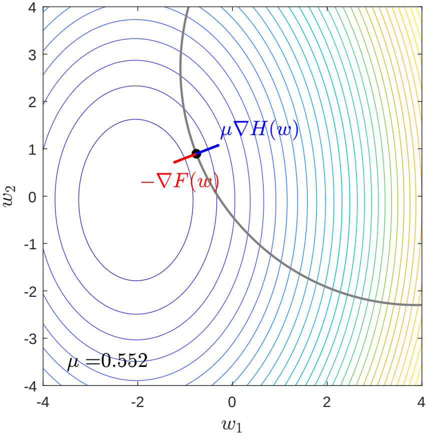
Animation inspired by Lecture 2 of the Winter School on Numerical Optimal Control with Differential Algebraic Equations by S. Gros and M. Diehl, Freiburg, 2016

Some intuitions on the KKT conditions

Ball rolling down a valley blocked by a fence - test problem with two variables and one inequality constraint

$$\min_{w\in\mathbb{R}^n} \; F(w)$$
$$\text{s.t. } H(w) \geq 0$$

- $-\nabla F$ is the gravity



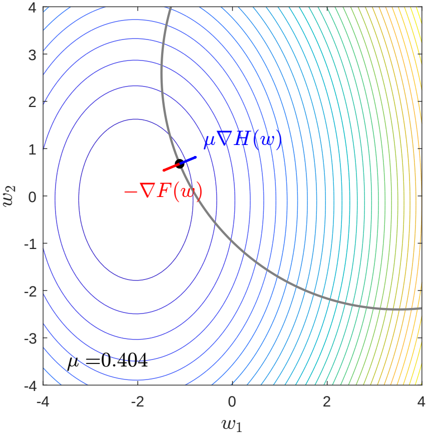
Animation inspired by Lecture 2 of the Winter School on Numerical Optimal Control with Differential Algebraic Equations by S. Gros and M. Diehl, Freiburg, 2016

Some intuitions on the KKT conditions

Ball rolling down a valley blocked by a fence - test problem with two variables and one inequality constraint

$$\min_{w \in \mathbb{R}^n} \; F(w)$$
$$\text{s.t. } H(w) \geq 0$$

- $-\nabla F$ is the gravity
- $\mu \nabla H$ is the force of the fence. Sign $\mu \geq 0$ means the fence can only "push" the ball



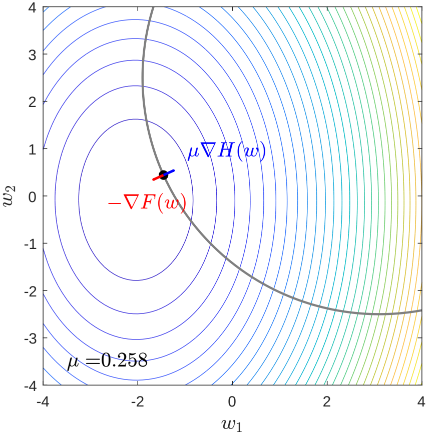
Animation inspired by Lecture 2 of the Winter School on Numerical Optimal Control with Differential Algebraic Equations by S. Gros and M. Diehl, Freiburg, 2016

Some intuitions on the KKT conditions

Ball rolling down a valley blocked by a fence - test problem with two variables and one inequality constraint

$$\min_{w \in \mathbb{R}^n} \; F(w)$$
$$\text{s.t. } H(w) \geq 0$$

- $-\nabla F$ is the gravity
- $\mu \nabla H$ is the force of the fence. Sign $\mu \geq 0$ means the fence can only "push" the ball
- ∇H gives the direction of the force and μ adjusts the magnitude.

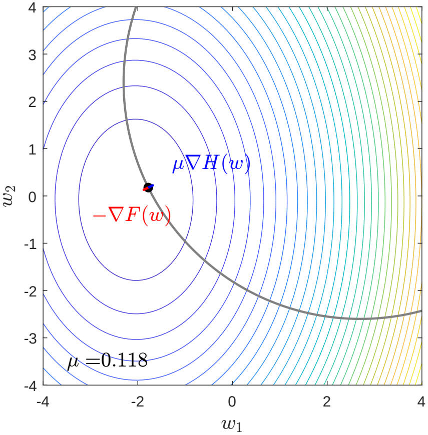


Animation inspired by Lecture 2 of the Winter School on Numerical Optimal Control with Differential Algebraic Equations by S. Gros and M. Diehl, Freiburg, 2016

Ball rolling down a valley blocked by a fence - test problem with two variables and one inequality constraint

$$\min_{w \in \mathbb{R}^n} \; F(w)$$
$$\text{s.t. } H(w) \geq 0$$

- $-\nabla F$ is the gravity
- $\mu \nabla H$ is the force of the fence. Sign $\mu \geq 0$ means the fence can only "push" the ball
- ∇H gives the direction of the force and μ adjusts the magnitude.
- active constraint: $H(w) = 0, \; \mu > 0$

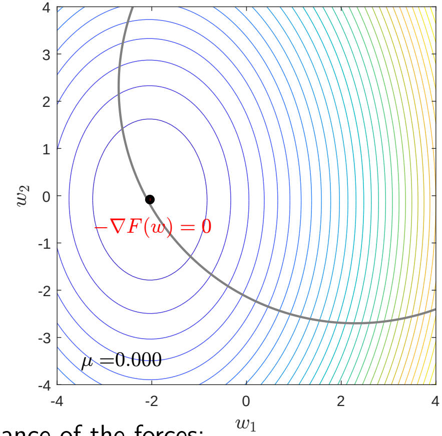


Animation inspired by Lecture 2 of the Winter School on Numerical Optimal Control with Differential Algebraic Equations by S. Gros and M. Diehl, Freiburg, 2016

Ball rolling down a valley blocked by a fence - test problem with two variables and one inequality constraint

$$\min_{w \in \mathbb{R}^n} \; F(w)$$
$$\text{s.t. } H(w) \geq 0$$

- $-\nabla F$ is the gravity
- $\mu \nabla H$ is the force of the fence. Sign $\mu \geq 0$ means the fence can only "push" the ball
- ∇H gives the direction of the force and μ adjusts the magnitude.
- active constraint: $H(w) = 0, \;\; \mu > 0$
- weakly active constraint: $H(w) = 0, \;\; \mu = 0$ the ball touches the fence but no force is needed



Balance of the forces:

$$\nabla \mathcal{L}(w, \mu) = \nabla F(w) - \nabla H(w)\mu = 0$$

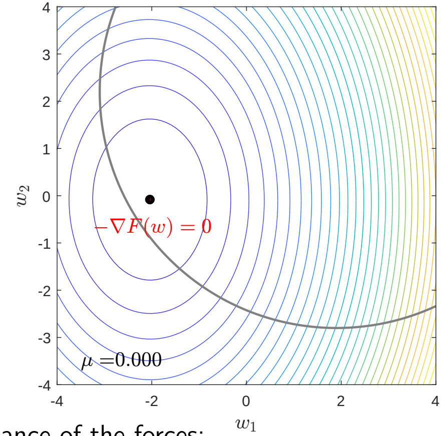
Animation inspired by Lecture 2 of the Winter School on Numerical Optimal Control with Differential Algebraic Equations by S. Gros and M. Diehl, Freiburg, 2016

Some intuitions on the KKT conditions

Ball rolling down a valley blocked by a fence - test problem with two variables and one inequality constraint

$$\min_{w \in \mathbb{R}^n} \; F(w)$$
$$\text{s.t. } H(w) \geq 0$$

- $-\nabla F$ is the gravity
- $\mu \nabla H$ is the force of the fence. Sign $\mu \geq 0$ means the fence can only "push" the ball
- ∇H gives the direction of the force and μ adjusts the magnitude.
- active constraint: $H(w) = 0, \;\; \mu > 0$
- weakly active constraint: $H(w) = 0, \;\; \mu = 0$ the ball touches the fence but no force is needed
- inactive constraint: $H(w) > 0, \;\; \mu = 0$



Balance of the forces:

$$\nabla \mathcal{L}(w, \mu) = \nabla F(w) - \nabla H(w)\mu = 0$$

Animation inspired by Lecture 2 of the Winter School on Numerical Optimal Control with Differential Algebraic Equations by S. Gros and M. Diehl, Freiburg, 2016

Outline of the lecture

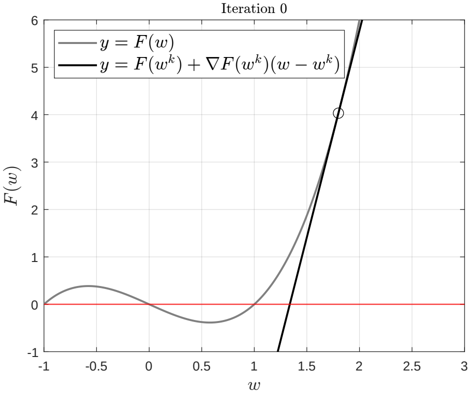
Linearization of F at linearization point $\bar{w}$

equals

First-order Taylor series at $\bar{w}$

equals

$$F_{\mathrm{L}}(w;\bar{w}) := F(\bar{w}) + \frac{\partial F}{\partial w}(\bar{w}) \quad (w - \bar{w})$$



Newton's method

To solve a nonlinear system, solve a sequence of linear systems

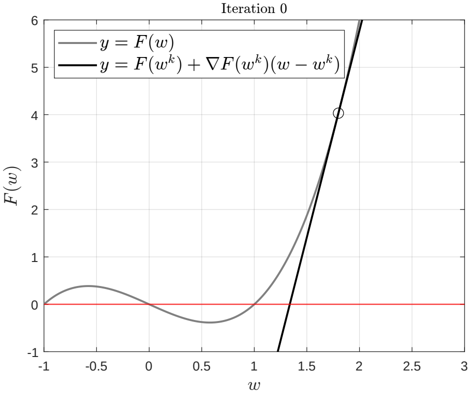
Linearization of F at linearization point $\bar{w}$

equals

First-order Taylor series at $\bar{w}$

equals

$$F_{\mathrm{L}}(w;\bar{w}) := F(\bar{w}) + \nabla_w F(\bar{w})^\top (w - \bar{w})$$



Newton's method

To solve a nonlinear system, solve a sequence of linear systems

Linearization of F at linearization point $\bar{w}$

equals

First-order Taylor series at $\bar{w}$

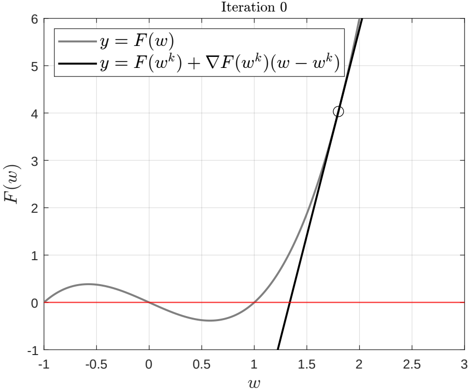
equals

$$F_{\mathrm{L}}(w;\bar{w}) := F(\bar{w}) + \nabla_w F(\bar{w})^\top (w - \bar{w})$$

Newton's methods, solve sequence of:

$$F(w^k) + \nabla F(w^k)^\top \Delta w = 0,$$

update $w^{k+1} = w^k + \Delta w$.
(for continuously differentiable $F : \mathbb{R}^n \to \mathbb{R}^n$)



Newton's method

To solve a nonlinear system, solve a sequence of linear systems

Linearization of F at linearization point $\bar{w}$

equals

First-order Taylor series at $\bar{w}$

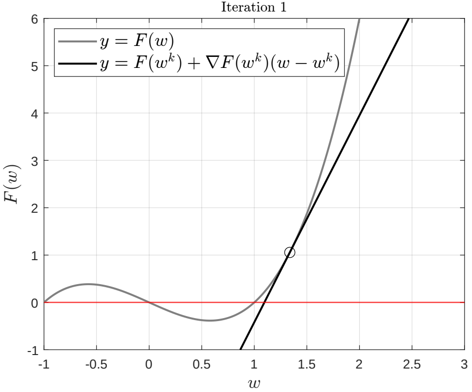
equals

$$F_{\mathrm{L}}(w;\bar{w}) := F(\bar{w}) + \nabla_w F(\bar{w})^\top (w - \bar{w})$$

Newton's methods, solve sequence of:

$$F(w^k) + \nabla F(w^k)^\top \Delta w = 0,$$

update $w^{k+1} = w^k + \Delta w$.
(for continuously differentiable $F : \mathbb{R}^n \to \mathbb{R}^n$)



Newton's method

To solve a nonlinear system, solve a sequence of linear systems

Linearization of F at linearization point $\bar{w}$

equals

First-order Taylor series at $\bar{w}$

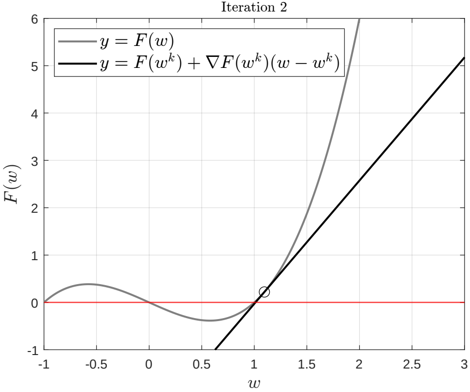
equals

$$F_{\mathrm{L}}(w;\bar{w}) := F(\bar{w}) + \nabla_w F(\bar{w})^\top (w - \bar{w})$$

Newton's methods, solve sequence of:

$$F(w^k) + \nabla F(w^k)^\top \Delta w = 0,$$

update $w^{k+1} = w^k + \Delta w$.
(for continuously differentiable $F : \mathbb{R}^n \to \mathbb{R}^n$)



Newton's method

To solve a nonlinear system, solve a sequence of linear systems

Linearization of F at linearization point $\bar{w}$

equals

First-order Taylor series at $\bar{w}$

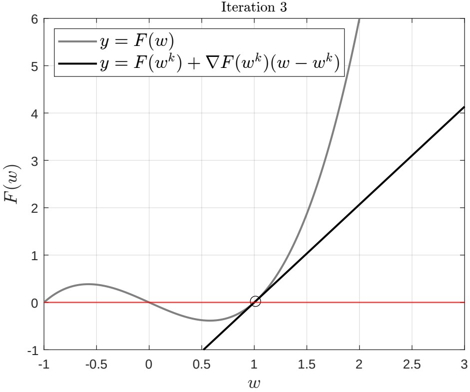
equals

$$F_{\mathrm{L}}(w;\bar{w}) := F(\bar{w}) + \nabla_w F(\bar{w})^\top (w - \bar{w})$$

Newton's methods, solve sequence of:

$$F(w^k) + \nabla F(w^k)^\top \Delta w = 0,$$

update $w^{k+1} = w^k + \Delta w$.
(for continuously differentiable $F : \mathbb{R}^n \to \mathbb{R}^n$)



General Nonlinear Program (NLP)

In direct methods, we have to solve the discretized optimal control problem, which is a Nonlinear Program (NLP)

General Nonlinear Program (NLP)

$$\min_{w \in \mathbb{R}^n} F(w) \quad \text{s.t.} \quad \begin{cases} G(w) & = & 0 \\ H(w) & \geq & 0 \end{cases}$$

We first treat the case without inequality constraints

NLP only with equality constraints

$$\min_{w \in \mathbb{R}^n} F(w) \quad \text{s.t.} \quad G(w) = 0$$

Lagrange function and optimality conditions

Lagrange function

$$\mathcal{L}(w, \lambda) = F(w) - \lambda^\top G(w)$$

For an optimal solution w^* there exist multipliers λ^* such that

Nonlinear root-finding problem

$$\begin{aligned} \nabla_w \mathcal{L}(w^*, \lambda^*) &= 0 \\ G(w^*) &= 0 \end{aligned}$$

Newton's method on optimality conditions

Use Newton's method to solve:

$$\begin{aligned} \nabla_w \mathcal{L}(w^*, \lambda^*) &= 0 \\ G(w^*) &= 0 \quad ? \end{aligned}$$

Given an iterate (w^k, λ^k), the linearization reads as:

$$\begin{array}{rlll} \nabla_w \mathcal{L}(w^k, \lambda^k) & +\nabla_w^2 \mathcal{L}(w^k, \lambda^k)\Delta w & -\nabla_w G(w^k)\Delta\lambda & = 0 \\ G(w^k) & +\nabla_w G(w^k)^\top \Delta w & & = 0 \end{array}$$

Due to $\nabla \mathcal{L}(w^k, \lambda^k) = \nabla F(w^k) - \nabla G(w^k)\lambda^k$, this is equivalent to:

$$\begin{array}{rlll} \nabla_w F(w^k) & +\nabla_w^2 \mathcal{L}(w^k, \lambda^k)\Delta w & -\nabla_w G(w^k)\lambda^+ & = 0 \\ G(w^k) & +\nabla_w G(w^k)^\top \Delta w & & = 0 \end{array}$$

with the shorthand $\lambda^+ = \lambda^k + \Delta\lambda$

Newton Step = Quadratic Program

Conditions

$$\begin{array}{cccccc} \nabla_w F(w^k) & +\nabla_w^2 \mathcal{L}(w^k, \lambda^k)\Delta w & -\nabla_w G(w^k)\lambda^+ & = & 0 \\ G(w^k) & +\nabla_w G(w^k)^\top \Delta w & & = & 0 \end{array}$$

are the KKT optimality conditions of a quadratic program (QP), namely:

Quadratic program

$$\begin{aligned} \min_{\Delta w \in \mathbb{R}^n} \quad & \nabla F(w^k)^\top \Delta w + \frac{1}{2} \Delta w^\top A^k \Delta w \\ \text{s.t.} \quad & G(w^k) + \nabla G(w^k)^\top \Delta w \quad = \quad 0, \end{aligned}$$

with $A^k = \nabla_w^2 \mathcal{L}(w^k, \lambda^k)$

Newton's method

The full step Newton's Method iterates by solving in each iteration the QP

Quadratic program in Sequential Quadratic Programming (SQP)

$$\begin{aligned} \min_{\Delta w \in \mathbb{R}^n} \quad & \nabla F(w^k)^\top \Delta w + \frac{1}{2} \Delta w^\top A^k \Delta w \\ \text{s.t.} \quad & G(w^k) + \nabla G(w^k)^\top \Delta w \quad = \quad 0, \end{aligned}$$

with $A^k = \nabla_w^2 \mathcal{L}(w^k, \lambda^k)$

This obtains as solution the step Δw^k and the new multiplier $\lambda_{\mathrm{QP}}^+ = \lambda^k + \Delta\lambda^k$

New iterate

$$\begin{aligned} w^{k+1} &= w^k + \Delta w^k \\ \lambda^{k+1} &= \lambda^k + \Delta\lambda^k = \lambda_{\mathrm{QP}}^+ \end{aligned}$$

This is the "full step, exact Hessian SQP method for equality constrained optimization".

NLP with inequality constraints

Regard again NLP with both, equality and inequality constraints:

NLP with equality and inequality constraints

$$\min_{w \in \mathbb{R}^n} F(w) \quad \text{s.t.} \quad \begin{cases} G(w) & = & 0 \\ H(w) & \geq & 0 \end{cases}$$

Lagrangian function for NLP with equality and inequality constraints

$$\mathcal{L}(w, \lambda, \mu) = F(w) - \lambda^\top G(w) - \mu^\top H(w)$$

Optimality conditions with inequalities

Theorem (Karush-Kuhn-Tucker (KKT) conditions)

Let F, G, H be $\mathcal{C}^2$. If w^* is a (local) minimizer and satisfies LICQ, then there are unique vectors λ^* and μ^* such that (w^*, λ^*, μ^*) satisfies:

$$
\begin{aligned}
&\nabla_w \mathcal{L}(w^*, \mu^*, \lambda^*) = 0 \\
&G(w^*) = 0 \\
&H(w^*) \geq 0 \\
&\mu^* \geq 0 \\
&H(w^*)^\top \mu^* = 0
\end{aligned}
$$

- Last tree *complementarity conditions* make the KKT conditions nonsmooth
- This system cannot be solved by plain Newton's method. But we can use SQP...

Sequential Quadratic Programming (SQP)

By Linearizing all functions within the KKT Conditions, and setting $\lambda^+ = \lambda^k + \Delta\lambda$ and $\mu^+ = \mu^k + \Delta\mu$, we obtain the KKT conditions of a Quadratic Program (QP)

QP with inequality constraints

$$\begin{aligned} \min_{\Delta w \in \mathbb{R}^n} \quad & \nabla F(w^k)^\top \Delta w + \frac{1}{2} \Delta w^\top A^k \Delta w \\ \text{s.t.} \quad & G(w^k) + \nabla G(w^k)^\top \Delta w = 0 \\ & H(w^k) + \nabla H(w^k)^\top \Delta w \geq 0 \end{aligned}$$

with $A^k = \nabla_w^2 \mathcal{L}(w^k, \lambda^k, \mu^k)$

Sequential Quadratic Programming (SQP)

By Linearizing all functions within the KKT Conditions, and setting $\lambda^+ = \lambda^k + \Delta\lambda$ and $\mu^+ = \mu^k + \Delta\mu$, we obtain the KKT conditions of a Quadratic Program (QP)

QP with inequality constraints

$$\begin{aligned} \min_{\Delta w \in \mathbb{R}^n} \quad & \nabla F(w^k)^\top \Delta w + \frac{1}{2} \Delta w^\top A^k \Delta w \\ \text{s.t.} \quad & G(w^k) + \nabla G(w^k)^\top \Delta w = 0 \\ & H(w^k) + \nabla H(w^k)^\top \Delta w \geq 0 \end{aligned}$$

with $A^k = \nabla_w^2 \mathcal{L}(w^k, \lambda^k, \mu^k)$

- QP solution: $\Delta w^k, \quad \lambda_{\mathrm{QP}}^+, \quad \mu_{\mathrm{QP}}^+$
- full step: $w^{k+1} = w^k + \Delta w^k$, $\lambda^{k+1} = \lambda_{\mathrm{QP}}^+$, $\mu^{k+1} = \mu_{\mathrm{QP}}^+$
- nonsmooth complementarity conditions resolved at QP level

Interior-point methods

(without equality constraint for lighter notation)

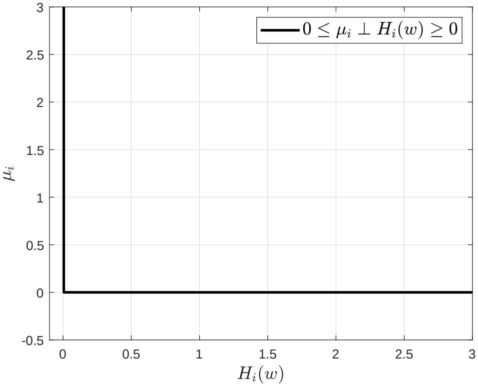
NLP with inequalites

$$\min_{w \in \mathbb{R}^n} \quad F(w)$$
$$\text{s.t.} \quad H(w) \geq 0$$

KKT conditions

$$\nabla F(w) - \nabla H(w)\mu = 0$$
$$0 \leq \mu \perp H(w) \geq 0$$

- Main difficulty: nonsmoothness of complementarity conditions
- 4th lecture (Tuesday) will show why Newton's method does not work for nonsmooth problems



Barrier problem in interior-point method

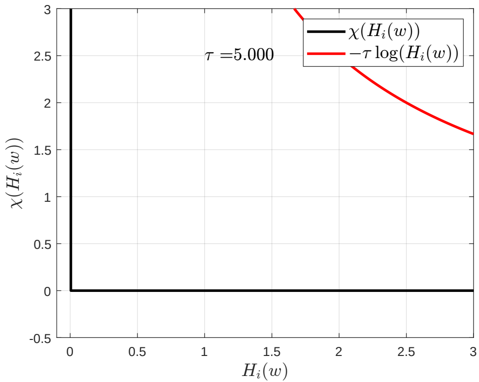
NLP with inequalites

$$\min_{w \in \mathbb{R}^n} \quad F(w)$$
$$\text{s.t.} \quad H(w) \geq 0$$

Idea: put inequality constraint into objective

Barrier problem

$$\min_{w \in \mathbb{R}^n} \; F(w) - \tau \sum_{i=1}^{m} \log(H_i(w)) =: F_\tau(w)$$



$\tau \log(H_i(w))$ approximates:

$$\chi(H_i(w)) = \begin{cases} 0 & \text{if } H_i(w) \geq 0 \\ \infty & \text{if } H_i(w) < 0 \end{cases}$$

Barrier problem in interior-point method

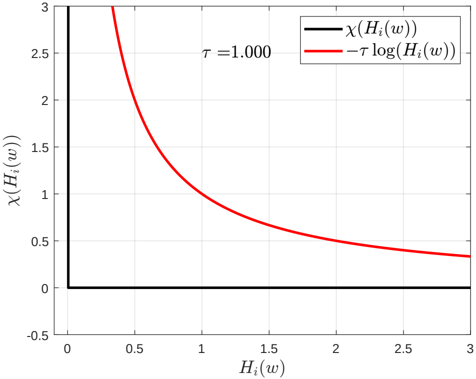
NLP with inequalites

$$\min_{w \in \mathbb{R}^n} \quad F(w)$$
$$\text{s.t.} \quad H(w) \geq 0$$

Idea: put inequality constraint into objective

Barrier problem

$$\min_{w \in \mathbb{R}^n} \; F(w) - \tau \sum_{i=1}^{m} \log(H_i(w)) =: F_\tau(w)$$



$\tau \log(H_i(w))$ approximates:

$$\chi(H_i(w)) = \begin{cases} 0 & \text{if } H_i(w) \geq 0 \\ \infty & \text{if } H_i(w) < 0 \end{cases}$$

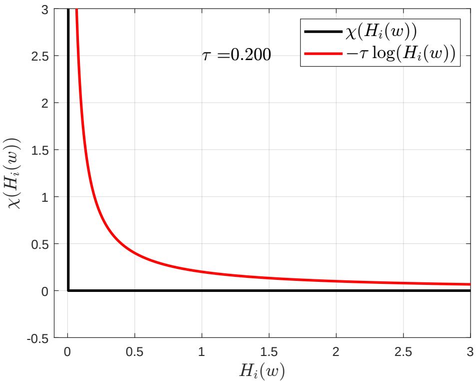
NLP with inequalites

$$\min_{w \in \mathbb{R}^n} \quad F(w)$$
$$\text{s.t.} \quad H(w) \geq 0$$

Idea: put inequality constraint into objective

Barrier problem

$$\min_{w \in \mathbb{R}^n} \; F(w) - \tau \sum_{i=1}^{m} \log(H_i(w)) =: F_\tau(w)$$



$\tau \log(H_i(w))$ approximates:

$$\chi(H_i(w)) = \begin{cases} 0 & \text{if } H_i(w) \geq 0 \\ \infty & \text{if } H_i(w) < 0 \end{cases}$$

Barrier problem in interior-point method

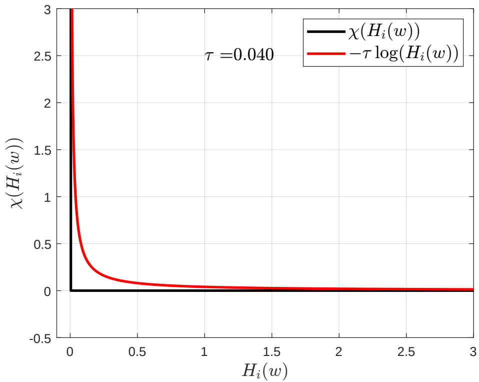
NLP with inequalites

$$\begin{aligned} \min_{w\in\mathbb{R}^n} \quad & F(w) \\ \text{s.t.} \quad & H(w) \geq 0 \end{aligned}$$

Idea: put inequality constraint into objective

Barrier problem

$$\min_{w\in\mathbb{R}^n} \; F(w) - \tau \sum_{i=1}^{m} \log(H_i(w)) =: F_\tau(w)$$



$\tau \log(H_i(w))$ approximates:

$$\chi(H_i(w)) = \begin{cases} 0 & \text{if } H_i(w) \geq 0 \\ \infty & \text{if } H_i(w) < 0 \end{cases}$$

Barrier problem in interior-point method

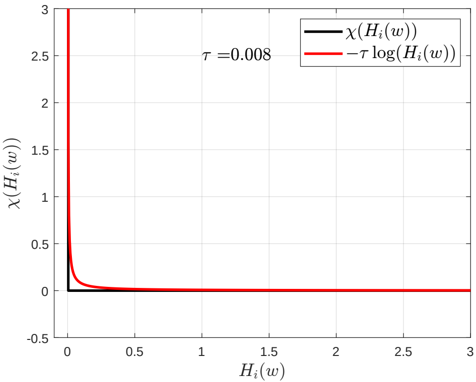
NLP with inequalites

$$\min_{w\in\mathbb{R}^n} \quad F(w)$$
$$\text{s.t.} \quad H(w) \geq 0$$

Idea: put inequality constraint into objective

Barrier problem

$$\min_{w\in\mathbb{R}^n} \; F(w) - \tau \sum_{i=1}^{m} \log(H_i(w)) =: F_\tau(w)$$



$\tau\log(H_i(w))$ approximates:

$$\chi(H_i(w)) = \begin{cases} 0 & \text{if } H_i(w) \geq 0 \\ \infty & \text{if } H_i(w) < 0 \end{cases}$$

Barrier problem in interior-point method

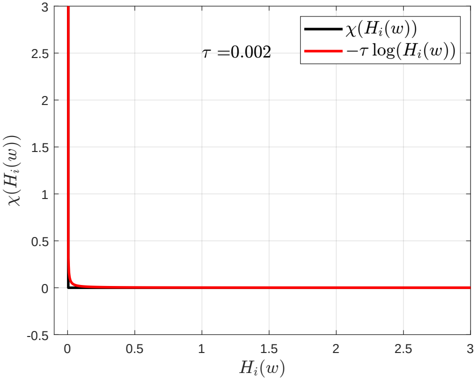
NLP with inequalites

$$\min_{w \in \mathbb{R}^n} \quad F(w)$$
$$\text{s.t.} \quad H(w) \geq 0$$

Idea: put inequality constraint into objective

Barrier problem

$$\min_{w \in \mathbb{R}^n} \ F(w) - \tau \sum_{i=1}^{m} \log(H_i(w)) =: F_\tau(w)$$



$\tau \log(H_i(w))$ approximates:

$$\chi(H_i(w)) = \begin{cases} 0 & \text{if } H_i(w) \geq 0 \\ \infty & \text{if } H_i(w) < 0 \end{cases}$$

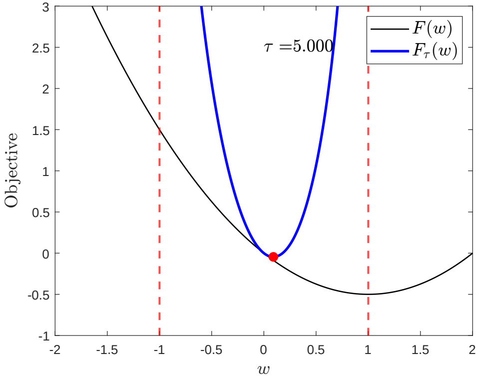
Example barrier problem

Example NLP

$$\min_{w \in \mathbb{R}^2} \quad 0.5w^2 - 2w$$
$$\text{s.t.} \quad -1 \leq w \leq 1$$

Barrier problem

$$\min_{w \in \mathbb{R}^2} \; 0.5w^2 - 2 - \tau \log(w+1) - \tau \log(1-w)$$



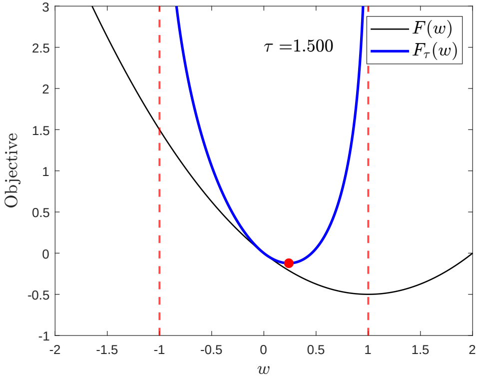
Example barrier problem

Example NLP

$$\min_{w \in \mathbb{R}^2} \quad 0.5w^2 - 2w$$
$$\text{s.t.} \quad -1 \leq w \leq 1$$

Barrier problem

$$\min_{w \in \mathbb{R}^2} \; 0.5w^2 - 2 - \tau \log(w+1) - \tau \log(1-w)$$



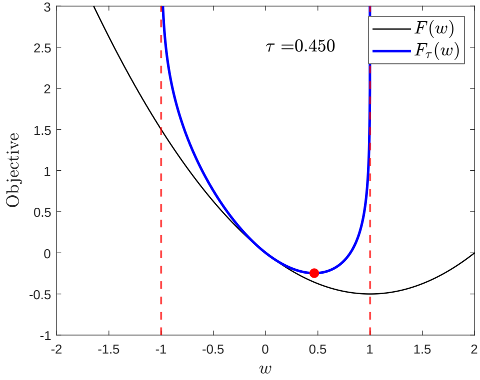
Example barrier problem

Example NLP

$$\min_{w \in \mathbb{R}^2} \quad 0.5w^2 - 2w$$
$$\text{s.t.} \quad -1 \le w \le 1$$

Barrier problem

$$\min_{w \in \mathbb{R}^2} \; 0.5w^2 - 2 - \tau \log(w+1) - \tau \log(1-w)$$



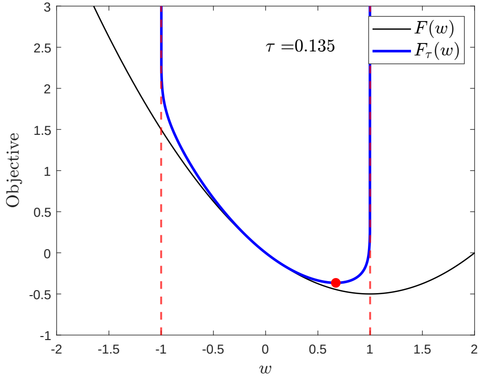
Example barrier problem

Example NLP

$$\min_{w \in \mathbb{R}^2} \quad 0.5w^2 - 2w$$
$$\text{s.t.} \quad -1 \leq w \leq 1$$

Barrier problem

$$\min_{w \in \mathbb{R}^2} \; 0.5w^2 - 2 - \tau \log(w+1) - \tau \log(1-w)$$



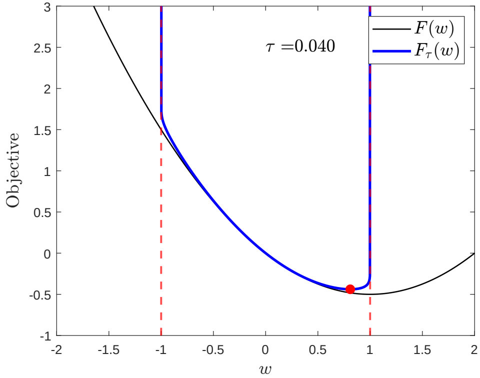
Example barrier problem

Example NLP

$$\min_{w\in\mathbb{R}^2} \quad 0.5w^2 - 2w$$
$$\text{s.t.} \quad -1 \leq w \leq 1$$

Barrier problem

$$\min_{w\in\mathbb{R}^2} \; 0.5w^2 - 2 - \tau \log(w+1) - \tau \log(1-w)$$



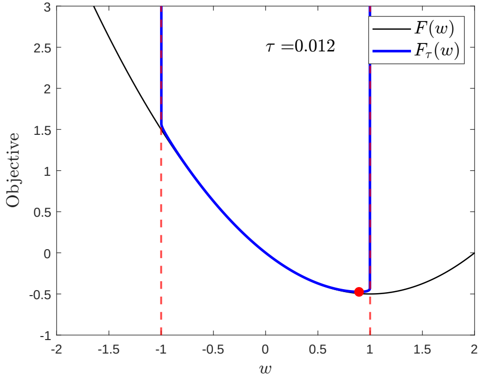
Example barrier problem

Example NLP

$$\min_{w\in\mathbb{R}^2} \quad 0.5w^2 - 2w$$
$$\text{s.t.} \quad -1 \leq w \leq 1$$

Barrier problem

$$\min_{w\in\mathbb{R}^2} \; 0.5w^2 - 2 - \tau\log(w+1) - \tau\log(1-w)$$



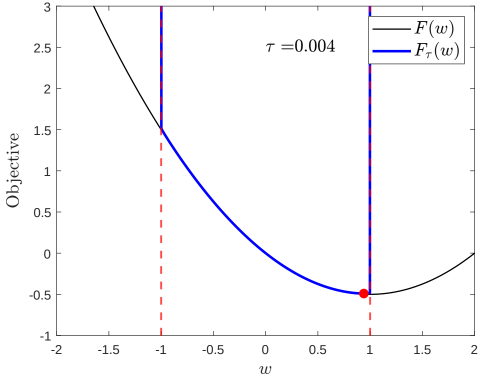
Example barrier problem

Example NLP

$$
\begin{aligned}
\min_{w \in \mathbb{R}^2} \quad & 0.5w^2 - 2w \\
\text{s.t.} \quad & -1 \leq w \leq 1
\end{aligned}
$$

Barrier problem

$$
\min_{w \in \mathbb{R}^2} \; 0.5w^2 - 2 - \tau \log(w+1) - \tau \log(1-w)
$$



Barrier problem

$$\min_{w \in \mathbb{R}^n} \; F(w) - \tau \sum_{i=1}^{m} \log(H_i(w)) =: F_\tau(w)$$

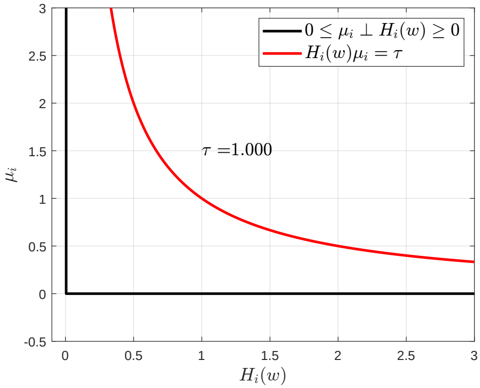
KKT conditions ($\nabla F_\tau(w) = 0$)

$$\nabla F(w) - \tau \sum_{i-1}^{m} \frac{1}{H_i(w)} \nabla H_i(w) = 0$$

Introduce variable $\mu_i = \frac{\tau}{H_i(w)}$

Smoothed KKT conditions

$$\nabla F(w) - \nabla H(w)^\top \mu = 0$$
$$H_i(w)\mu_i = \tau$$
$$(H_i(w) > 0, \mu_i > 0)$$



Barrier problem

$$\min_{w\in\mathbb{R}^n} \; F(w) - \tau \sum_{i=1}^{m} \log(H_i(w)) =: F_\tau(w)$$

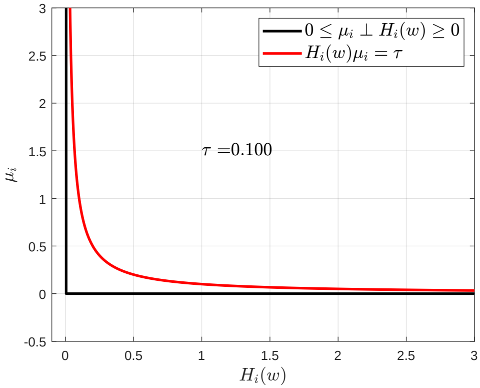
KKT conditions ($\nabla F_\tau(w) = 0$)

$$\nabla F(w) - \tau \sum_{i-1}^{m} \frac{1}{H_i(w)} \nabla H_i(w) = 0$$

Introduce variable $\mu_i = \frac{\tau}{H_i(w)}$

Smoothed KKT conditions

$$\begin{aligned} &\nabla F(w) - \nabla H(w)^\top \mu = 0 \\ &H_i(w)\mu_i = \tau \\ &(H_i(w) > 0, \mu_i > 0) \end{aligned}$$



Primal-dual interior-point method

Alternative interpretation

Barrier problem

$$\min_{w \in \mathbb{R}^n} \; F(w) - \tau \sum_{i=1}^{m} \log(H_i(w)) =: F_\tau(w)$$

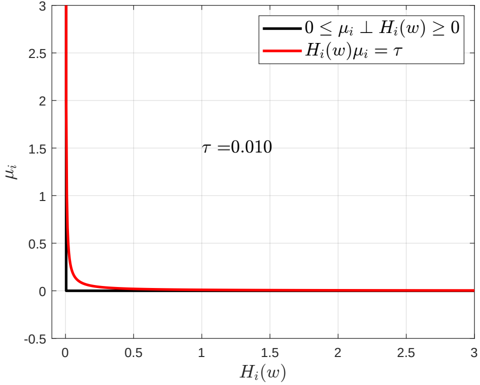
KKT conditions ($\nabla F_\tau(w) = 0$)

$$\nabla F(w) - \tau \sum_{i-1}^{m} \frac{1}{H_i(w)} \nabla H_i(w) = 0$$

Introduce variable $\mu_i = \frac{\tau}{H_i(w)}$

Smoothed KKT conditions

$$\nabla F(w) - \nabla H(w)^\top \mu = 0$$
$$H_i(w)\mu_i = \tau$$
$$(H_i(w) > 0, \mu_i > 0)$$



Primal-dual interior-point method

Alternative interpretation

Barrier problem

$$\min_{w \in \mathbb{R}^n} \; F(w) - \tau \sum_{i=1}^{m} \log(H_i(w)) =: F_\tau(w)$$

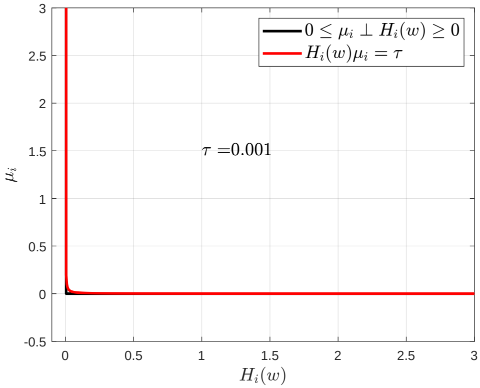
KKT conditions ($\nabla F_\tau(w) = 0$)

$$\nabla F(w) - \tau \sum_{i-1}^{m} \frac{1}{H_i(w)} \nabla H_i(w) = 0$$

Introduce variable $\mu_i = \frac{\tau}{H_i(w)}$

Smoothed KKT conditions

$$\nabla F(w) - \nabla H(w)^\top \mu = 0$$
$$H_i(w)\mu_i = \tau$$
$$(H_i(w) > 0, \mu_i > 0)$$



Primal-dual interior-point method

Nonlinear programming problem

$$\min_{\substack{w\in\mathbb{R}^n \\ s\in\mathbb{R}^{n_H}}} \quad F(w)$$
$$\text{s.t.} \quad G(w) = 0$$
$$H(w) - s = 0$$
$$s \geq 0$$

Smoothed KKT conditions

$$R_\tau(w,s,\lambda,\mu) = \begin{bmatrix} \nabla_w \mathcal{L}(w,\lambda,\mu) \\ G(w) \\ H(w) - s \\ \operatorname{diag}(s)\mu - \tau e \end{bmatrix} = 0$$

$(s, \mu > 0)$

$e = (1, \ldots, 1)$

Solve approximately with Newton's method for fixed τ

$$R_\tau(w,s,\lambda,\mu) + \nabla R_\tau(w,s,\lambda,\mu)\Delta z = 0$$

with $z = (w,s,\lambda,\mu)$

Line-serach

Find $\alpha \in (0,1)$

$$w^{k+1} = w^k + \alpha\Delta w$$
$$s^{k+1} = s^k + \alpha\Delta s$$
$$\lambda^{k+1} = \lambda^k + \alpha\Delta\lambda$$
$$\mu^{k+1} = \mu^k + \alpha\Delta\mu$$

such that $s^{k+1} > 0, \mu^{k+1} > 0$

and reduce τ...

Summary

- Optimization problem come in many variants (LP, QP, NLP, MPCC, MINLP, OCP, ...)
- Each problem class be addressed with suitable software.
- Lagrangian function, duality, and KKT conditions are important concepts
- For convex problems KKT conditions sufficient for global optimality.
- Newton-type optimization for NLP solves the nonsmooth KKT conditions via Sequential Quadratic Programming (SQP) or via the Interior-Point Method.
- NLP solvers need to evaluate
  first and second order derivatives (e.g. via `CasADi`).

Some interesting and important topics not covered today

- Duality for convex optimization problems.
- First-order methods (gradient descent, stochastic gradient descent, ...).
- Solution methods for linear and quadratic programs (active set, interior-point, simplex).
- Augmented Lagrangian methods for constrained optimization.
- Solution methods for mixed-integer problems (branch and bound, ...)
- Computing derivatives via automatic differentiation.
- Globalization strategies (linear search vs trust region, merit functions vs filter).
- Regularization (convexification of the Hessian, LICQ violation).

Optimization textbooks

Nonlinear optimization:

- Nocedal, Jorge, and Stephen J. Wright, eds. Numerical optimization. New York, NY: Springer New York, 2006.
- Biegler, Lorenz T. Nonlinear programming: concepts, algorithms, and applications to chemical processes. Society for Industrial and Applied Mathematics, 2010.

Convex optimization:

- Boyd, Stephen, and Lieven Vandenberghe. Convex Optimization. Cambridge University Press, 2004. online: `https://web.stanford.edu/~boyd/cvxbook/`
- Rockafellar, R. T., Fundamentals of optimization. Lecture Notes 2007. online: `https://sites.math.washington.edu/~rtr/fundamentals.pdf`

Free online video lectures

Numerical optimization video lectures by Moritz Diehl (**highly recommended!**):

- Videos: `https://www.syscop.de/teaching/ws2020/numerical-optimization`
- Lecture notes: `https://publications.syscop.de/Diehl2016.pdf`

Lecture notes/slides by Mario Zanon Sébastien Gros

- `https://mariozanon.wordpress.com/teaching/numerical-methods-for-optimal-control/`

Optimization software:

- `https://plato.asu.edu/guide.html`
- `https://www.syscop.de/research/software`

References for this lecture

- Moritz Diehl, Sébastien Gros. "Numerical optimal control (Draft)," Lecture notes, 2024. online: `https://www.syscop.de/files/2024ws/NOC/book-NOCSE.pdf`
- Karmarkar, Narendra. "A new polynomial-time algorithm for linear programming." In Proceedings of the sixteenth annual ACM symposium on Theory of computing, pp. 302-311. 1984.
- Dantzig, George B. "Origins of the simplex method." In A history of scientific computing, pp. 141-151. 1990.

Summary of optimality conditions

Optimality conditions for NLP with equality and/or inequality constraints:

- **First-Order Necessary Conditions**: A local optimizer of a (differentiable) NLP is a KKT point
- **Second-Order Sufficient Conditions** require positivity of the Hessian in all critical feasible directions

Nonconvex problem ⇒ a minimizer is not necessarily a global minimizer.
Note: some nonconvex problems may have a unique **minimum**

Some important practical consequences...

- A KKT point **may not** be a local (global) optimizer
  ... the lack of equivalence results from a lack of regularity and/or SOSC
- A local (global) optimizer **may not** be a KKT point
  ... due to violation of constraint qualifications, e.g. LICQ violated