Module 04 — Optimization Problems
KKT Conditions & Solvers

Ahmad F. Taha

EE 5243: Introduction to Cyber-Physical Systems

Email: ahmad.taha@utsa.edu

Webpage: http://engineering.utsa.edu/~taha/index.html

September 14, 2015

Module 04 Outline

In this module, we present basic mathematical optimization principles. The following topics are covered[1]:

- General introduction to optimization
- Convex optimization
- Linear programming, SDP
- Mixed-integer programming
- Relaxations
- KKT optimality conditions
- Optimization problems solvers

[1]Much of the material presented in this module can be found in [Taylor, 2015; Boyd & Vandenberghe, 2004]

Optimization — 1

Given a function to be minimized, $f(x)$**,** $x \in \mathbb{R}^n$

- x_0 is a global minimum of $f(x) \Rightarrow f(x_0) \leq f(x)$ for all x
- x_0 is a local minimum of $f(x) \Rightarrow f(x_0) \leq f(x)$ for $\{x \in \mathbb{R}^n; \|x - x_0\| \leq \epsilon, \epsilon > 0\}$

Convexity:

- *Function* — $f(x)$ is convex if:
$$f(\alpha x + (1-\alpha)y) \leq \alpha f(x) + (1-\alpha) f(y)$$
for all $0 \leq \alpha \leq 1$
- *Set* — $\mathcal{X}$ is convex if $x, y \in \mathcal{X} \Rightarrow \alpha x + (1-\alpha)y \in \mathcal{X}$
- If $g(x)$ is convex, then $\mathcal{X} = \{x \mid g(x) \leq 0\}$ is convex

Convex Optimization Problem

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & f(x) \\ \text{s.t.} \quad & g_i(x) \leq 0, \; i = 1, ..., m \end{aligned}$$

$f(x), g_i(x)$ are all convex $\Rightarrow$ any local minimum is a global minimum

Examples of Convex Functions

Functions mapping $\mathbb{R} \to \mathbb{R}$*:*

- Affine $(ax + b)$, exponential (e^{ax})
- Powers $(x^a; a \geq 1, a \leq 0)$, powers of absolute value $(|x|^p; p \geq 1)$

Functions mapping $\mathbb{R}^n \to \mathbb{R}$*:*

- Affine $(a^\top x + b)$
- Vector norms $\left(\|x\|_p = \left(\sum_{i=1}^n |x|^p\right)^{\frac{1}{p}} \;, p \geq 1 \right)$

Functions mapping $\mathbb{R}^{n \times m} \to \mathbb{R}$*:*

- Affine $(f(X) = trace(A^\top X) + b)$
- Matrix norms $(\|X\|_2 = \sigma_{\max}(X) = \sqrt{\lambda_{\max}(X^\top X)})$

Tractability & Linear Programs

Computational tractability:

- Convex optimization: easy to solve, polynomial-time
- Nonconvex optimization: difficult to solve, NP-hard[2]

Linear programming:

- $f(x) = c^T x$
- Affine constraints: $g_i(x) = a_i^T x - b_i$ (usually as vector: $Ax \leq b$)
- Easiest type of optimization
- Solvable in polynomial time
- Quadratic programming with $f(x) = x^T Qx$ is also easy if Q is symmetric positive semi-definite
- *Is it convex?* See next slide

[2] NP-hard: no polynomial-time (efficient) algorithm can exist

Convexity of a Quadratic Function

Solution:

- Given that $f(x) = x^\top Qx$, we apply the definition of convex function.

$$\alpha f(x) + (1-\alpha)f(y) - f(\alpha x + (1-\alpha)y) \geq 0.$$

- Substituting for $f(x)$ into the LHS of the previous equation yields:

$$\alpha x^\top Qx + (1-\alpha)y^\top Qy - (\alpha x + (1-\alpha)y)^\top Q(\alpha x + (1-\alpha)y)$$

$$= \alpha(1-\alpha)x^\top Qx - 2\alpha(1-\alpha)x^\top Qy + \alpha(1-\alpha)y^\top Qy = \alpha(1-\alpha)(x-y)^\top Q(x-y)$$

- Define $z = x - y \Rightarrow$

$$\alpha(1-\alpha)z^\top Qz$$

- Since $0 \leq \alpha \leq 1$, $Q = Q^\top \succeq$ and $\forall\, z \Rightarrow$

$$\alpha(1-\alpha)z^\top Qz \geq 0$$

Mixed-Integer Programming

MIP

$$\begin{aligned} \min_{x,y} \quad & f(x,y) \\ \text{s.t.} \quad & g_i(x,y) \le 0, \; i = 1, ..., m \\ & y_i \in \mathbb{Z} \quad \text{(the integers)} \end{aligned}$$

- NP-hard even when f and g_i are all linear
- $y_i \in \mathbb{Z}$ is nononvex
- Very common in CPS planning problems
- Even more relevant in smart grids: unit commitment problem, expansion models, ...

Semidefinite Programming

- **Hermitian**: $X = X^*$ (conjugate transpose), $X \in \mathbb{C}^{n \times n}$
- **Definition**: $z^* X z \geq 0$ for all $z \in \mathbb{R}^n$
- **Equivalent**: all eigs. of X nonnegative, all principal minors nonnegative
- **Notation**: $X \succeq 0$
- $X \succeq 0$ **is convex constraint**

  Proof: Suppose $X, Y \succeq 0$. Then

$$z^*(\alpha X + (1-\alpha)Y)z = \alpha z^* X z + (1-\alpha) z^* Y z \geq 0$$

Semidefinite Programming — 2

Semidefinite Program (SDP)

$$\begin{aligned} \min_{X} \quad & \text{trace}(CX) \\ \text{s.t.} \quad & \text{trace}(A_i X) = b_i \ , \ X \succeq 0 \end{aligned}$$

Semidefinite Program (SDP) — Form 2

$$\begin{aligned} \min_{x} \quad & c^\top x \\ \text{s.t.} \quad & \underbrace{F(x) = F_0 + \sum_{i=1}^{n} F_i x_i \succeq 0}_{\text{Linear Matrix Inequality (LMI)},\ F_i = F_i^\top} \end{aligned}$$

- SDP: linear cost function, LMI constraints — Convex, 1 minimum
- Generalization of LP (don't solve LP as SDP)
- SDP's can be solved in polynomial-time using interior point methods

LMIs

- A system of LMIs $F_1(x), F_2(x), \ldots, \succeq 0$ can be represented as a single LMI:

$$F(x) = \begin{bmatrix} F_1(x) & & & \\ & F_2(x) & & \\ & & \ddots & \\ & & & F_m(x) \end{bmatrix} \succeq 0$$

- For an $\mathbb{R}^{m \times n}$ matrix A, the inequality $Ax \leq b$ can be represented as m LMIs:

$$b_i - a_i^\top x \geq 0, \quad i = 1, 2, \ldots, m$$

- Most optimization solvers cannot handle " $\succ$ " $\Rightarrow$ replace it with " $\succeq$ "
- Example: Lyapunov's $A^\top P + PA \prec 0$ is an LMI

LMI Example

Lyapunov's Theorem

Real parts of eig(A) are negative iff there exists a real symmetric positive definite matrix $P = P^\top \succ 0$ such that:

$$A^\top P + PA \prec 0.$$

- **Can we write Lyapunov's inequality as an LMI?**
- Define: $P = \begin{bmatrix} x_1 & x_2 & \dots & x_n \\ x_2 & x_{n+1} & & \\ \vdots & & & \\ x_n & x_{2n-1} & \dots & x_m \end{bmatrix}$, $m = \frac{(n+1)n}{2}$: # of Variables

$$P_1 = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 0 & & \\ \vdots & & & \\ 0 & 0 & \dots & 0 \end{bmatrix}, P_2 = \begin{bmatrix} 0 & 1 & \dots & 0 \\ 1 & 0 & & \\ \vdots & & & \\ 0 & 0 & \dots & 0 \end{bmatrix}, \dots, P_m = \begin{bmatrix} 0 & 0 & \dots & 0 \\ 0 & 0 & & \\ \vdots & & & \\ 0 & 0 & \dots & 1 \end{bmatrix}$$

$$\Rightarrow A^\top P + PA = \sum_{i=1}^m x_i(A^\top P_i + P_i A) = -x_1 F_1 - x_2 F_2 - \dots x_m F_m \prec 0$$

$$\text{IS AN LMI}$$

Example — Convex Quadratic Functions

Is the quadratic function

$$f(x) = x^\top \begin{bmatrix} 1 & 2 & 2 \\ 0 & 1 & 2 \\ 0 & 0 & 0 \end{bmatrix} x$$

indefinite, positive definite, negative definite, positive semidefinite, or negative semidefinite?

1. Start with finding the leading principal minors? **NO!**
2. Need to symmetrize $f(x)$:

$$f(x) = \frac{1}{2} x^\top (Q + Q^\top) x = \frac{1}{2} x^\top \begin{bmatrix} 2 & 2 & 2 \\ 2 & 2 & 2 \\ 2 & 2 & 0 \end{bmatrix} x$$

3. From the principal minors, we conclude that the quadratic form is indefinite

Most Popular LMIs

- **LMIP** — find a feasible x such that $F(x) \succ 0$
  - *Example:* Lyapunov theorem
- **EVP/PDP** — eigenvalue problem (EVP) is to minimize the maximum eigenvalue of a matrix $A(x)$ that depends affinely on a variable subject to an LMI constraint
  - *Example:* Finding the best H_∞ robust controller: stabilization + good performance

Example — Eigenvalue Optimization

Suppose $A(x) \in \mathbb{C}^{n\times n}$ is a linear function of x

- **Objective:** minimize the maximum eigenvalue of $A(x)$:

$$\begin{array}{ll} \min\limits_{x,\lambda} & \lambda \\ \text{s.t.} & \lambda \text{ is the largest eig. of } A(x) \end{array}$$

- Eigenvalue:

$$\begin{aligned} A(x)v = \lambda v \quad &\Rightarrow \quad v^*A(x)v = \lambda v^* v \;\Rightarrow\; \frac{v^*A(x)v}{v^*v} = \lambda \\ &\Rightarrow \quad \max_{v\in\mathbb{C}^n} \frac{v^*A(x)v}{v^*v} = \lambda_{\max} \\ &\Rightarrow \quad \lambda_{\max} v^* I v \geq v^* A(x) v \quad \forall v \end{aligned}$$

- Hence, optimization problem can be equivalently written as:

$$\begin{array}{ll} \min\limits_{\lambda} & \lambda \\ \text{s.t.} & v^*(\lambda I - A(x))v \geq 0 \quad \forall v \end{array} \qquad \begin{array}{ll} \min\limits_{\lambda} & \lambda \\ \text{s.t.} & \lambda I - A(x) \succeq 0 \end{array}$$

Quadratic Optimization Problems

Quadratic Constrained Problem (QCP)

$$\begin{aligned} \min_{x} \quad & x^* C x \\ \text{s.t.} \quad & x^* A_i x \leq b_i \end{aligned}$$

Solvability:

- If $C \succeq 0$ and $A_i \succeq 0$, solvable in polynomial time
- If any are not positive semi-definite (PSD), problem becomes NP-hard

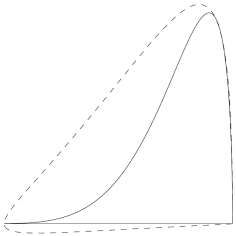
Applications:

- Binary constraints: $x \in \{0, 1\} \Leftrightarrow x^2 = x$
- AC power flow in power systems
- Both examples are nonconvex

Relaxations

- **Hard problem (exact):**

$$P_1 : \min_{x \in \mathcal{X}} f(x)$$



- **Easier, relaxed problem:**

$$P_2 : \min_{x \in \mathcal{Y}} f(x), \quad X \subset Y$$

- **Facts:**
  - (Obj. of P_2) $\leq$ (Obj. of P_1)
  - IF x IS OPTIMAL FOR RELAXATION AND FEASIBLE FOR EXACT, x IS OPTIMAL FOR EXACT

  * **Proof:** Suppose x is relaxed optimal and feasible suboptimal for exact problem $\Rightarrow \exists\, y$ s.t. $f(y) < f(x)$, $y \in X$. But by relaxation, $y \in Y$, and therefore x is not relaxed optimal — a contradiction

SDP Relaxations

- SDP can be written:

$$\begin{aligned} \min_{x} \quad & \text{trace}(xx^* C) \\ \text{s.t.} \quad & \text{trace}(xx^* A_i) \leq b_i \end{aligned}$$

- $X = xx^*$ equivalent to: $X \succeq 0$, $\text{rank}(X) = 1$

$$\begin{aligned} \min_{X} \quad & \text{trace}(XC) \\ \text{s.t.} \quad & \text{trace}(XA_i) \leq b_i \\ & X \succeq 0 \\ & \text{rank}(X) = 1 \end{aligned}$$

- Removing a constraint enlarges the feasible set, i.e. relaxation:

$$\begin{aligned} \min_{X} \quad & \text{trace}(XC) \\ \text{s.t.} \quad & \text{trace}(XA_i) \leq b_i \\ & X \succeq 0 \end{aligned}$$

- If solution X^* has rank 1, then ***relaxation is tight***
- Feasible, optimal exact solution is Cholesky: $X = xx^*$

Linear Relaxation

- Consider the following optimizaton problem:

$$\begin{aligned} \min_{x} \quad & x_1(x_2-1) \\ \text{s.t.} \quad & x_1 \geq 1 \\ & x_2 \geq 2 \\ & x_1x_2 \leq 3 \end{aligned}$$

- Clearly, this problem is not convex (objective & a constraint)
- **Relaxation:** let $y = x_1x_2$, OP becomes:

$$\begin{aligned} \min_{x,y} \quad & y - x_1 \\ \text{s.t.} \quad & x_1 \geq 1 \\ & x_2 \geq 2 \\ & y \leq 3 \\ & \underbrace{y - 2x_1 - x_2 + 2}_{=(x_1-1)(x_2-2)} \geq 0 \end{aligned}$$

- Last constraint guarantees that $y \neq -\infty$

Solving Unconstrained OPs

Objective:

$$\underset{x \in \mathbb{R}^n}{\text{minimize}}\; f(x)$$

Necessary & Sufficinet Conditions for Optimality

x^* is a local minimum of $f(x)$ iff:

1. Zero gradient at x^*:
$$\nabla_x f(x^*) = 0$$
2. Hessian at x^* is positive semi-definite:
$$\nabla_x^2 f(x^*) \succeq 0$$

- For maximization, Hessian is negative semi-definite

Solving Constrained OPs

- **Main objective:** find/compute minimum or a maximum of an objective function subject to equality and inequality constraints
- Formally, problem defined as finding the optimal x^*:

$$\begin{aligned} \min_{x} \quad & f(x) \\ \text{subject to} \quad & g(x) \leq 0 \\ & h(x) = 0 \end{aligned}$$

  - $x \in \mathbb{R}^n$
  - $f(x)$ is scalar function, possibly nonlinear
  - $g(x) \in \mathbb{R}^m, h(x) \in \mathbb{R}^l$ are vectors of constraints

Main Principle

To solve constrained optimization problems: transform constrained problems to unconstrained ones.

How?

Augment the constraints to the cost function.

KKT Conditions

$$\begin{aligned} \min_{x} \quad & f(x) \\ \text{subject to} \quad & g(x) \leq 0 \\ & h(x) = 0 \end{aligned}$$

- Define the Lagrangian: $\mathcal{L}(x, \lambda, \mu) = f(x) + \lambda^T h(x) + \mu^T g(x)$

Optimality Conditions

The constrained optimization problem (above) has a local minimizer x^* iff there exists a unique μ^* such that:

1. $\nabla_x \mathcal{L}(x^*, \lambda^*, \mu^*) = \nabla_x f(x) + \lambda^{*T} \nabla_x h(x^*) + \mu^{*T} \nabla_x g(x^*) = 0$
2. $\mu_j^* \geq 0$ for $j = 1, \ldots, m$
3. $\mu_j^* g_j(x^*) = 0$ for $j = 1, \ldots, m$
4. $g_j(x^*) \leq 0$ for $j = 1, \ldots, m$
5. $h_i(x^*) = 0$ for $i = 1, \ldots, l$ (if x^*, μ^*, λ^* satisfy 1–5, they are candidates)
6. Second order necessary conditions (SONC): $\nabla_x^2 \mathcal{L}(x^*, \lambda^*, \mu^*) \succeq 0$

KKT Conditions — Example[3]

Find the minimizer of the following optimization problem:

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f(x) = (x_1 - 1)^2 + x_2 - 2 \\ \text{subject to} \quad & g(x) = x_1 + x_2 - 2 \leq 0 \\ & h(x) = x_2 - x_1 - 1 = 0 \end{aligned}$$

- First, find the Lagrangian function:

$$\mathcal{L}(x, \lambda, \mu) = (x_1 - 1)^2 + x_2 - 2 + \lambda(x_2 - x_1 - 1) + \mu(x_1 + x_2 - 2)$$

- Second, find the conditions of optimality (from previous slide):

1. $\nabla_x \mathcal{L}(x^*, \lambda^*, \mu^*) = \begin{bmatrix} 2x_1^* - 2 - \lambda^* + \mu^* & 1 + \lambda^* + \mu^* \end{bmatrix}^\top = \begin{bmatrix} 0 & 0 \end{bmatrix}^\top$
2. $\mu^*(x_1^* + x_2^* - 2) = 0$
3. $\mu^* \geq 0$
4. $x_1^* + x_2^* - 2 \leq 0$
5. $x_2^* - x_1^* - 1 = 0$
6. $\nabla_x^2 \mathcal{L}(x^*, \lambda^*, \mu^*) = \nabla_x^2 f(x^*) + \lambda^* \nabla_x^2 h(x^*) + \mu^* \nabla_x^2 g(x^*) \succeq 0$

$$= \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix} + \lambda^* \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} + \mu^* \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \succeq 0$$

[3] Example from [Chong & Zak, 2011]

Example — Cont'd

- To solve the system equations for the optimal x^*, λ^*, μ^*, we first try $\mu^* > 0$.
- Given that, we solve the following set of equations:
  1. $2x_1^* - 2 - \lambda^* + \mu^* = 0$
  2. $1 + \lambda^* + \mu^* = 0$
  3. $x_1^* + x_2^* - 2 = 0$
  4. $x_2^* - x_1^* - 1 = 0$

  $\Rightarrow$ $x_1^* = 0.5, x_2^* = 1.5, \lambda^* = -1, \mu^* = 0$
- But this solution contradicts the assumption that $\mu^* > 0$
- **Alternative:** assume $\mu^* = 0 \Rightarrow x_1^* = 0.5, x_2^* = 1.5, \lambda^* = -1, \mu^* = 0$
- This solution satisfies $g(x^*) \leq 0$ constraint, hence it's a candidate for being a minimizer
- We now verify the SONC: $L(x^*, \lambda^*, \mu^*) = \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix} \succeq 0$
- Thus, $x^* = \begin{bmatrix} 0.5 & 1.5 \end{bmatrix}^\top$ is a strict local minimizer

OPs Taxonomy

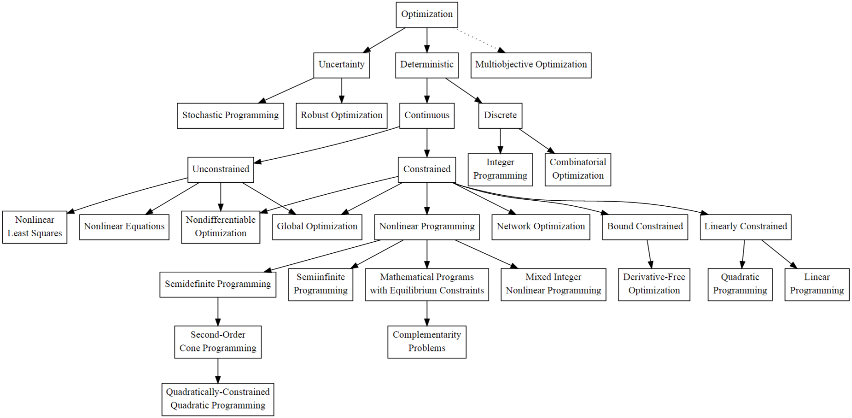


Figure from:
http://www.neos-guide.org/content/optimization-introduction

Solvers

- Solving optimization problems require few things
  1. Modeling the problem
  2. Translating the problem model (constraints and objectives) into a modeling language (`AMPL`, `GAMS`, `MATLAB`, `YALMIP`, `CVX`)
  3. Choosing optimization algorithms solvers (`Simplex`, `Interior-Point`, `Brand & Bound`, `Cutting Planes`,...)
  4. Specifying tolerance, exit flags, flexible constraints, bounds, ...
- Convex optimization problems: use `cvx` (super easy to install and code)
- MATLAB's `fmincon` is always handy too (too much overhead, often fails to converge for nonlinear optimization problems)
- Visit http://www.neos-server.org/neos/solvers/index.html
- Check http://www.neos-guide.org/ to learn more

Complexity

- Clearly, **complexity** of an OP depends on the solver used
- Example: most LMI solvers use interior-point methods
- **Complexity:** primal-dual interior-point has a worst-case complexity $\mathcal{O}(m^{2.75}L^{1.5})$
  - $m: \#of Variables$, $L: \#of Constraints$
  - Applies to a set of L Lyapunov inequalities
  - Typical performance: $\mathcal{O}(m^{2.1}L^{1.2})$

Questions And Suggestions?

Thank You!

Please visit

engineering.utsa.edu/~taha

IFF you want to know more ☺

References I

Boyd, S., & Vandenberghe, L. (2004). *Convex Optimization*. New York, NY, USA: Cambridge University Press.

Chong, E., & Zak, S. (2011). *An Introduction to Optimization*. Wiley Series in Discrete Mathematics and Optimization. Wiley.
URL https://books.google.com/books?id=THlxFmlEy_AC

Taylor, J. (2015). *Convex Optimization of Power Systems*. Cambridge University Press.