Module 03 — Part 01
Introduction to System Optimization

Dr. Ahmad F. Taha

CE 4240/5240 — Intro to Infrastructure Systems Engineering

Email: ahmad.taha@vanderbilt.edu

Webpage: http://lab.vanderbilt.edu/taha

VANDERBILT UNIVERSITY

September 18, 2024

Module 02 — Part 01 Outline

Part I — Introduction to Optimization

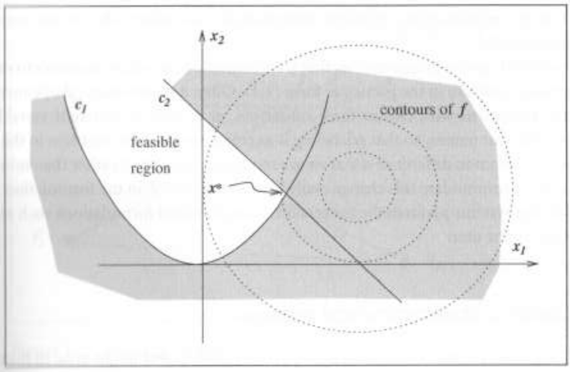
Optimization in words

Minimize a cost function or maximize a profit function

Subject to some constraints on the decision variables

A simple example

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



Picture from Nocedal-Wright, Numerical Optimization

What is an optimization problem?

Minimization of a function subject to constraints on its variables.

$$\begin{array}{lll} \text{minimize} & f_0(x) & \\ \text{subject to} & g_i(x) \leq 0, \quad i = 1, \ldots, m & (\textbf{inequality constraints}) \\ & h_i(x) = 0, \quad i = 1, \ldots, p & (\textbf{equality constraints}) \end{array}$$

- $x = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$ is a vector of **unknowns**, also called **variables**
- $f_0(x) : \mathbb{R}^n \rightarrow \mathbb{R}$ is the **objective function**, also called **cost function**
- $g_i(x) : \mathbb{R}^n \rightarrow \mathbb{R}$ and $h_i(x) : \mathbb{R}^n \rightarrow \mathbb{R}$ are the **constraint functions**

Solution of the optimization problem

$$
\begin{array}{ll}
\text{minimize} & f_0(x) \\
\text{subject to} & g_i(x) \leq 0, \quad i = 1, \ldots, m \\
& h_i(x) = 0, \quad i = 1, \ldots, p
\end{array}
$$

We want to find a point x^* that minimizes $f_0(x)$ and *at the same time* satisfies the constraints $g_i(x) \leq 0$ for all $i = 1, \ldots, m$, and $h_i(x) = 0$, for all $i = 1, \ldots, p$.

The set of points that satisfy the constraints is called the **feasible set**.

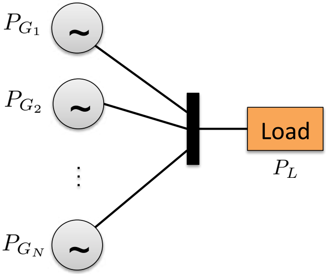
$\mathcal{F} = \{x \in \mathbb{R}^n$ such that $g_i(x) \leq 0 \; (i = 1, \ldots, m)$ and $h_i(x) = 0 \; (i = 1, \ldots, p)\}$

So the solution x^* of the problem must satisfy

$f_0(x^*) \leq f_0(x)$ for all x in the set $\mathcal{F}$.

Motivating example from power engineering

- Suppose N power generation units serve a given load P_L (e.g., a city)
- The power output of unit i is P_{G_i} MW
- The cost of operating unit i is $C_i(P_{G_i})$ \$/h

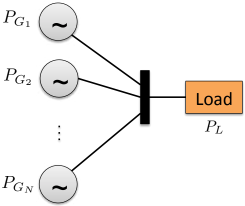


The economic dispatch problem

Given the cost functions $C_i(P_{G_i})$ and the load P_L, find the most economically generated power output.

$$\begin{aligned} \min \quad & \sum_{i=1}^{N} C_i(P_{G_i}) \\ \text{subj. to} \quad & \sum_{i=1}^{N} P_{G_i} = P_L \\ & P_{G_i} \geq 0, \; i = 1, \ldots, N \end{aligned}$$

Economic dispatch is solved every 5-15 minutes in modern power grids.



Numerical example

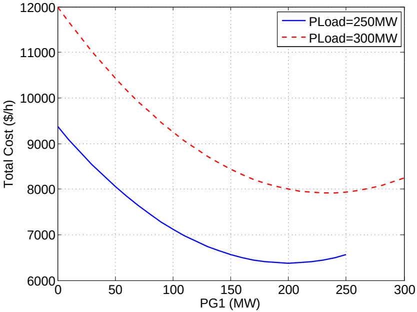
Two units: $C_1(P_{G_1}) = 0.025P_{G_1}^2 + 20P_{G_1}$ and $C_2(P_{G_2}) = 0.05P_{G_2}^2 + 25P_{G_2}$.

Comparison: Optimal solution vs. sharing the load equally between the two generators

P_L (MW)	$P_{G_1}^*$	$P_{G_2}^*$	Cost ($/h)	$P_{G_1} = \frac{P_L}{2}$	$P_{G_2} = \frac{P_L}{2}$	Cost ($/h)
250	200	50	6,375	125	125	6,797
300	233.3	66.7	7,917	150	150	8,438

The need for algorithms

In this example, we can read the solution by plotting $C_1(P_{G_1}) + C_2(P_L - P_{G_1})$ versus P_{G_1}.



What if we had hundreds of units?

We need **systematic methods** to solve optimization problems.

Modeling languages and solvers

Modeling language

- Accept the problem in a format similar to its mathematical form (minimize/subj. to)
- Then convert the problem to a format acceptable for a solver, and call the solver
- Easier and shorter code, but extra time is required to parse the problem

Solver

- Solves a specific type of problem (e.g., linear program) given the problem data
- Problem must be brought into a standard form (or in general, a form that the solver accepts)

We'll look at solving the economic dispatch using CVX (modeling language) and Matlab's quadprog (solver)

$$
\begin{array}{ll}
\text{minimize} & 0.025P_{G_1}^2 + 20P_{G_1} + 0.05P_{G_2}^2 + 25P_{G_2} \\
\text{subject to} & P_{G_1} + P_{G_2} = 250 \\
& P_{G_1} \geq 0, \; P_{G_2} \geq 0
\end{array}
$$

Solvers

- Examples: MATLAB's linprog, quadprog
  https://www.mathworks.com/help/optim/ug/linprog.html
  https://www.mathworks.com/help/optim/ug/quadprog.html
- SeDuMi (http://sedumi.ie.lehigh.edu/)
- SDPT3 (http://www.math.nus.edu.sg/~mattohkc/sdpt3.html)
- MOSEK (https://www.mosek.com/)
- GUROBI (https://www.gurobi.com/)
- CPLEX (https://www.ibm.com/analytics/cplex-optimizer)

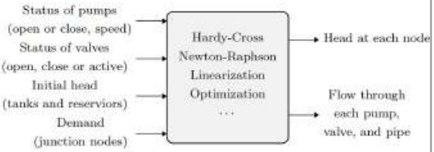
Modeling languages

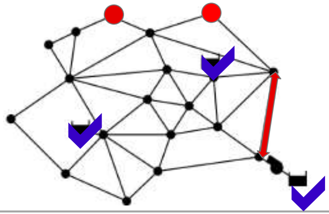
- CVX (http://cvxr.com/cvx/)
- YALMIP (https://yalmip.github.io/)
  - See all solvers that YALMIP can call: https://yalmip.github.io/allsolvers/
- GAMS (https://www.gams.com/)
- AIMMS, AMPL, CPLEX Optimization Studio, Google OR-Tools

Another example: Water flow problem in drinking networks

Water Flow Problem

Solving for water flow and head (pressure) given water demand forecasts

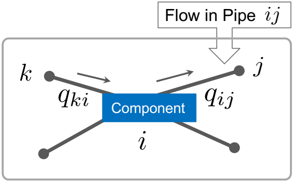


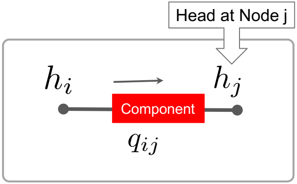


WFP — 1

Components and principles in WDNs

Typical Topology





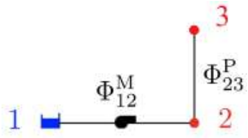
Components	Junctions, tanks, or reservoirs	Pipes, pumps, or valves
Principles	Conservation of mass	Conservation of energy
Effects	Linear equations	Nonlinear equations

All linear equations are in light blue, and all nonlinear equations are in red.

WFP — 2

Compo-nent	Hydraulic Modeling	Description
Junction	$\sum q^{\text{in}}(k) - \sum q^{\text{out}}(k) = d(k)$	Inflow – Outflow=Demand
Tank	$h^{\text{TK}}(k+1) = h^{\text{TK}}(k) + \alpha(\text{net flow})$	Tank Volume/Head Dynamics
Pipe	$\Delta h^{\text{P}}(k) = Rq(k)\,\lvert q(k)\rvert^{\mu-1}$	Head loss in pipes due to friction
Pump	$\Delta h^{\text{M}}(k) = -s^2(k)\left(h_0 - rq^{\nu}(k)s^{-\nu}(k)\right)$	Head gain through pump control
Valve	$\Delta h^{\text{W}}(k) = oRq(k)\lvert q(k)\rvert^{\mu-1}$	Head loss when water flows through a valve

WFP as a feasibility problem



$$\textbf{DAE} \;\Rightarrow \Phi_{12}(q_{12}, s) = \Delta h_{12}^{\mathrm{M}}(q_{12}, s) = s^2(h_{\mathrm{R}} r(q_{12}/s)^{\nu}),$$

$$\Phi_{23}(q_{23}) = \Delta h_{23}^{\mathrm{P}}(q_{23}) = R q_{23}|q_{23}|^{\mu-1},$$

$$h_2 = h_{\mathrm{R}} + \Phi_{12}(q_{12}, s),$$

$$h_3 = h_2 - \Phi_{23}(q_{23}).$$

$$q(k) := \{q_{12}(k), q_{23}(k)\}$$

$$h(k) := \{h^{\mathrm{J}}(k), h^{\mathrm{R}}(k)\}$$

$$x(k) := \{h(k), q(k)\}$$

Feasibility Prob. $\Rightarrow$ minimize$_x$ $\;0$

subject to DAE

$$x \in [x^{\min}, x^{\max}]$$

Applications

Communications and Networking

1. Power control in cellular networks
2. Capacity of parallel Gaussian channels and waterfilling
3. Optimal flow control and optimal routing

Energy, water, transportation

1. Economic dispatch,
2. Optimal power flow, optimal water flow, electricity markets, ramp metering
3. State estimation, water quality, traffic control

Signal Processing and Machine Learning

1. Least-squares, regression models, sparsity regularizations
2. Linear classification
3. Recommender systems: How does Netflix recommend movies?
4. Neural network training

Control

1. Optimal control of discrete-time linear systems
2. Numerous controllers are designed using Linear Matrix Inequalities

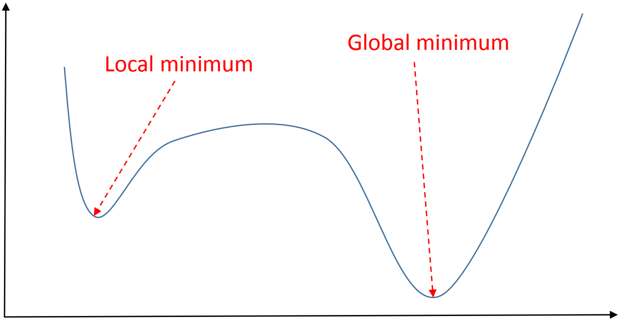
Aerospace and Navigation

1. Position, velocity, and time estimation by GPS

Part II — Complexity of Optimization Problems Convexity Definition

Solving optimization problems

Iterative algorithm: Start with an initial guess $x^{(0)}$, then find the next iterate $x^{(1)}$, etc.
Is the sequence $x^{(0)}, x^{(1)}, x^{(2)}, \ldots$ going to converge to x^*?
Answer is provided by *convergence analysis*.



Difficult versus easy problems

General optimization problems are very difficult to solve

- May need very long computation time
- Find a global minimum or another point?

Convex optimization problems can be solved efficiently and reliably

- A local minimum is always a global minimum for a convex optimization problem (proof later in the module)
- The computational effort to solve a problem (e.g., number of iterations, or number of additions and multiplications required) versus the size of the problem scales better in the case of convex problems than in general (nonconvex) problems
- It is possible to have a *certificate of optimality* in convex optimization problems, that is, we can have a criterion to check whether a given point is indeed the optimal solution and get a yes or no answer

Convex *versus* ***nonconvex*** *is the dividing line between easy and difficult*

- Convex optimization problems may not be easy to recognize
- Some engineering problems are nonconvex; convex optimization methods can provide approximate solution

Convex vs. nonconvex optimization problems

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

Is this problem convex? Nonconvex?
What about integer variables? Mixed-integer formulations?

Convex vs. nonconvex optimization problems

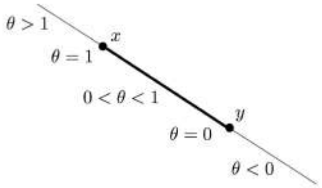
$$\begin{array}{lll} \text{minimize} & f_0(x) & \Rightarrow \textbf{Functions} \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m & \\ & h_i(x) = 0, \quad i = 1, \ldots, p & \Rightarrow \textbf{Sets} \end{array}$$

Is this problem convex? Nonconvex?

Convexity definition?!

Lines and line segments

Suppose x, y are two points in $\mathbb{R}^n$
Line through x and y: Points of the form $\theta x + (1-\theta)y$ with $\theta \in \mathbb{R}$
Line segment between x and y: Points of the form $\theta x + (1-\theta)y$ with $\theta \in [0, 1]$

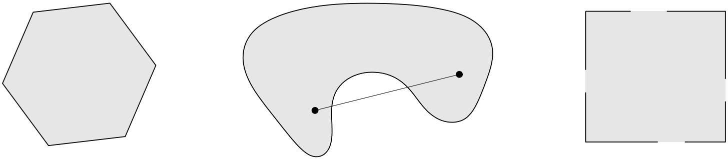


Convex set

Set $S \subset \mathbb{R}^n$ is **convex** if

$$x, y \in S,\ \theta \in [0,1] \quad \Longrightarrow \quad \theta x + (1-\theta) y \in S$$

Geometrically: $x, y \in S \Longrightarrow$ line segment between $x, y \subset S$
Examples (one convex set and two nonconvex sets)



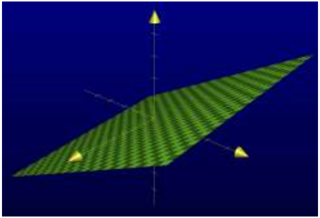
- If the border was entirely missing from the last example, we would have a convex set

Subspaces

Set $S \subset \mathbb{R}^n$ is a **subspace** if

$$x, y \in S,\ \lambda, \mu \in \mathbb{R} \quad \Longrightarrow \quad \lambda x + \mu y \in S$$

Geometrically: $x, y \in S \Longrightarrow$ plane through $0, x, y \subset S$



Example of a subspace

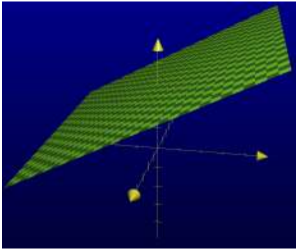
- If a set is subspace, then the origin must belong to the set
- This follows from the definition for $\lambda = \mu = 0$
- A line through the origin is also a subspace

Affine sets

Set $S \subset \mathbb{R}^n$ is **affine** if

$$x, y \in S, \quad \lambda \in \mathbb{R} \quad \Longrightarrow \quad \lambda x + (1-\lambda) y \in S$$

Geometrically: $x, y \in S \Longrightarrow$ line through $x, y \subset S$



- An affine set is like a subspace with an offset
- A subspace is also an affine set

Subspaces and affine sets are convex.

Convex cone

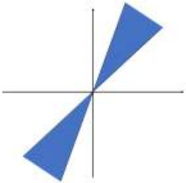
Set $S \subset \mathbb{R}^n$ is a **cone** if

$$x \in S,\ \lambda \geq 0 \quad \Longrightarrow \quad \lambda x \in S$$

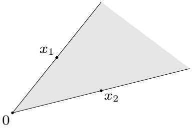
Geometrically: $x \in S \Longrightarrow$ ray through $0, x \subset S$
Set $S \subset \mathbb{R}^n$ is a **convex cone** if

$$x, y \in S,\ \lambda, \mu \geq 0 \quad \Longrightarrow \quad \lambda x + \mu y \in S$$

Geometrically: $x, y \in S \Longrightarrow$ "pie slice" between $0, x, y \subset S$



(Nonconvex) cone



Convex cone

Some notations

- The notation $a \preceq b$, where a and b are **vectors**, means *componentwise inequality*:

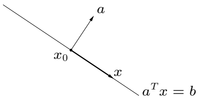
$$\begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix} \preceq \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix} \Longleftrightarrow a_i \leq b_i, \ i = 1, \ldots, n$$

- The notation $a \prec b$ means componentwise strict inequality
- For a **matrix** A, the notation $A \succ 0$ means A is positive definite (sometimes abbreviated as p.d.)
- For a matrix A, $A \succeq 0$ means A is positive semidefinite
- When we say that a matrix is positive/negative (semi-)definite or indefinite, we always imply that the matrix is **symmetric**

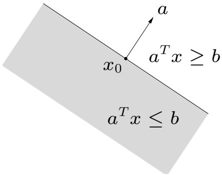
Positive/negative (semi-)definite and indefinite?

Hyperplanes and halfspaces

hyperplane is set of the form $\{x \mid a^T x = b\}$
a hyperplane is an affine set; it is a subspace if $b = 0$
a is called *normal vector*



halfspace is set of the form $\{x \mid a^T x \leq b\}$
a halfspace is a convex set; it is a convex cone if $b = 0$
a is called *(outward) normal vector*



Another representation of halfspaces

Consider any point x_0 on the hyperplane defined by $a^T x = b$ (that is, $a^T x_0 = b$).
The halfspace can be represented in two ways

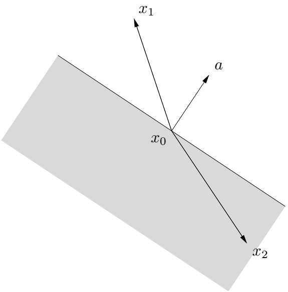
$$\{x \mid a^T x \leq b\} = \{x \mid a^T (x - x_0) \leq 0\}$$

- Vector $x_1 - x_0$ forms an acute angle with a, so x_1 is not in the halfspace

$$a^T (x_1 - x_0) > 0$$

- Vector $x_2 - x_0$ forms an obtuse angle with a, so x_2 is in the halfspace

$$a^T (x_2 - x_0) < 0$$

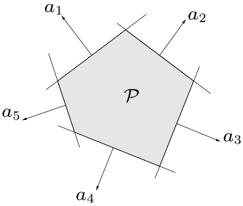


Polyhedra

Polyhedron is the solution set of finitely many linear inequalities and equalities

$$\begin{aligned}\mathcal{P} &= \{x \mid a_j^T x \leq b_j,\ j=1,\ldots,m; \quad c_j^T x = d_j,\ j=1,\ldots,p\} \\ &= \{x \mid Ax \preceq b,\ Cx = d\}\end{aligned}$$

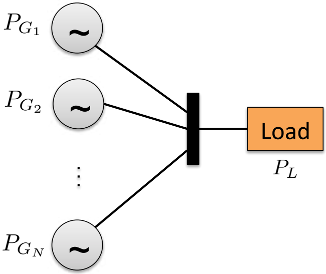
where A has the vectors a_j^T as rows, and similarly for C; and vector b has entries b_j, and similarly for d
A polyhedron is a convex set.



Example: The nonnegative orthant
$\mathbb{R}_+^n = \{x \in \mathbb{R}^n \mid x_i \geq 0,\ i = 1, \ldots, n\} = \{x \in \mathbb{R}^n \mid x \succeq 0\}$ is a polyhedron and a convex cone.

Motivating example from power engineering

- Suppose N power generation units serve a given load P_L (e.g., a city)
- The power output of unit i is P_{G_i} MW
- The cost of operating unit i is $C_i(P_{G_i})$ \$/h



The economic dispatch problem

Given the cost functions $C_i(P_{G_i})$ and the load P_L, find the most economically generated power output.

$$\begin{aligned} \min \quad & \sum_{i=1}^{N} C_i(P_{G_i}) \\ \text{subj. to} \quad & \sum_{i=1}^{N} P_{G_i} = P_L \\ & 0 \leq P_{G_i},\ i = 1, \ldots, N \end{aligned}$$

Is the feasible set convex? Polyhedron?
What if each unit has a limit on how much it can produce, i.e., $P_{G_i} \leq P_{G_i}^{\max}$?

Set operations that preserve convexity

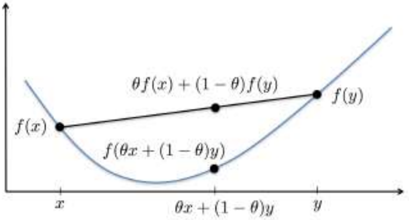
Let $a \in \mathbb{R}$, $b \in \mathbb{R}^n$ and suppose that sets $S \subset \mathbb{R}^n$, S_1, and S_2 are convex. The following sets are convex.

1. *Scaling:* $aS = \{ax \,|\, x \in S\}$.
2. *Translation:* $S + b = \{x + b \,|\, x \in S\}$
3. *Sum:* $S_1 + S_2 = \{x_1 + x_2 \,|\, x_1 \in S_1, x_2 \in S_2\}$
4. *Direct or Cartesian Product:* $S_1 \times S_2 = \{(x_1, x_2) \,|\, x_1 \in S_1, x_2 \in S_2\}$

Functions convexity

A function $f : \mathbb{R}^n \to \mathbb{R}$ is convex if 1) its domain is a convex set; and 2) it satisfies the following property:

$$f(\theta x + (1-\theta)y) \leq \theta f(x) + (1-\theta)f(y), \quad \text{for all } x, y \in \mathrm{dom}(f),\ \theta \in [0,1].$$



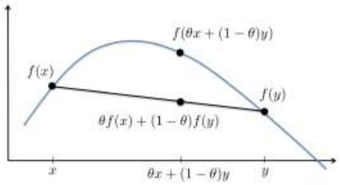
A function $f : \mathbb{R}^n \to \mathbb{R}$ is *strictly convex* if 1) its domain is a convex set; and 2) it satisfies the following property:

$$f(\theta x+(1-\theta)y) < \theta f(x)+(1-\theta)f(y), \quad \text{for all } x, y \in \mathrm{dom}(f),\ x \neq y,\ \theta \in (0,1).$$

Concavity

A function $f : \mathbb{R}^n \to \mathbb{R}$ is concave if $-f$ is convex.
Equivalently, a function $f : \mathbb{R}^n \to \mathbb{R}$ is concave if 1) its domain is a convex set; and 2) it satisfies the following property:

$$f(\theta x + (1-\theta)y) \geq \theta f(x) + (1-\theta) f(y), \quad \text{for all } x, y \in \mathrm{dom}(f),\ \theta \in [0,1].$$



A function $f : \mathbb{R}^n \to \mathbb{R}$ is *strictly concave* if 1) its domain is a convex set; and 2) it satisfies the following property:

$$f(\theta x+(1-\theta)y) > \theta f(x)+(1-\theta) f(y), \quad \text{for all } x, y \in \mathrm{dom}(f),\ x \neq y,\ \theta \in (0,1).$$

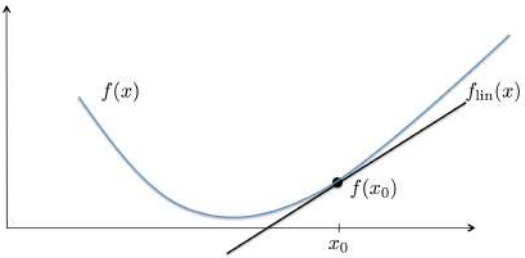
Notes

- For affine functions, we always have equality in the definition of convex/concave functions
- Hence: all affine (and also linear) functions are convex and concave
- Conversely, any function that is convex and concave is affine
- A convex function is continuous on the interior of its domain—it can have discontinuities only on its boundary

Convexity criterion based on gradient vector (1st-order condition)

A differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ is convex if and only if 1) its domain is a convex set; and 2) its linear approximation (1st-order Taylor approximation) at any point is a global underestimator of the function. That is,

$$f(x) \geq f(x_0) + \nabla f(x_0)^T (x - x_0) \quad \text{for all } x_0, x \in \mathrm{dom}(f)$$



Additional 1st-order conditions

A differentiable function[1] $f : \mathbb{R}^n \to \mathbb{R}$ is concave if and only if 1) its domain is a convex set; and 2) its linear approximation (1st-order Taylor approximation) at any point is a global overestimator of the function. That is,

$$f(x) \leq f(x_0) + \nabla f(x_0)^T (x - x_0) \quad \text{for all } x_0, x \in \mathrm{dom}(f)$$

A differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ is *strictly convex* if and only if 1) its domain is a convex set; and 2) the following condition holds:

$$f(x) > f(x_0) + \nabla f(x_0)^T (x - x_0) \quad \text{for all } x_0, x \in \mathrm{dom}(f), \; x \neq x_0$$

A differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ is *strictly concave* if and only if 1) its domain is a convex set; and 2) the following condition holds:

$$f(x) < f(x_0) + \nabla f(x_0)^T (x - x_0) \quad \text{for all } x_0, x \in \mathrm{dom}(f), \; x \neq x_0$$

[1] This means that its gradient ∇f exists at each point in $\mathrm{dom}(f)$.

Criterion based on Hessian matrix (2nd-order condition)

For a twice differentiable function, the **Hessian matrix** is defined as

$$\nabla^2 f(x) = \left[\frac{\partial^2 f(x)}{\partial x_i \partial x_j}\right]$$

A twice differentiable function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is convex if and only if 1) its domain is a convex set; and 2) its Hessian matrix $\nabla^2 f(x)$ is positive semidefinite for all $x \in \text{dom}(f)$.

A twice differentiable function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is concave if and only if 1) its domain is a convex set; and 2) its Hessian matrix $\nabla^2 f(x)$ is negative semidefinite for all $x \in \text{dom}(f)$.

A twice differentiable function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is strictly convex if 1) its domain is a convex set; and 2) its Hessian matrix $\nabla^2 f(x)$ is positive definite for all $x \in \text{dom}(f)$.
Converse is not true (ex: $f(x) = 1/x^2$ with domain $x \in \{\mathbb{R} - 0\}$).

Examples on $\mathbb{R}^n$

- Quadratic function $f(x) = \frac{1}{2}x^T Px + q^T x + r$
  - Convex if $P \succeq 0$; strictly convex if $P \succ 0$
  - Concave if $P \preceq 0$; strictly concave if $P \prec 0$;
  - Not convex and not concave if P is indefinite
  - Proof: The Hessian is $\nabla^2 f(x) = P$
- Negative entropy $f(x) = x \log x$ for positive x is convex (domain is convex, take first then second derivative, show it's positive for all positive x)

Operations that preserve convexity

1. *Nonnegative weighted sums:* If $f_1, \ldots, f_m$ are convex, and $w_1 \geq 0, \ldots, w_m \geq 0$, then $f = w_1 f_1 + \cdots + w_m f_m$ is convex
2. *Pointwise maximum:* If $f_1, \ldots, f_m$ are convex, then $f(x) = \max\{f_1(x), \ldots, f_m(x)\}$ is convex
3. Affine transformation of domain: If $f(z)$ is convex, then $f(Ax + b)$ is convex

Convex vs. nonconvex optimization problems

Remember this standard optimization problem?

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

Now, we can say for the problem to be ***convex***:

- Functions $f_i(x)$, $i = 0, 1 \dots, m$, are convex

$$f_i(\alpha x + (1-\alpha)y) \leq \alpha f_i(x) + (1-\alpha) f_i(y) \text{ for all } x, y, \text{and } \alpha \in (0,1)$$

- Functions $h_i(x)$, $i = 1 \dots, p$, are linear

$$h_i(x) = a_{i1}x_1 + a_{i2}x_2 + \ldots + a_{in}x_n + b_i$$

Different classes of convex optimization problems

- Linear programs
- Linear-fractional programs
- Quadratic programs
- Second-order cone programs
- Geometric programs
- Semidefinite programs

Linear program (LP)

$$\begin{aligned} \text{minimize} \quad & c^T x \\ \text{subject to} \quad & Gx \preceq h \\ & Ax = b \end{aligned}$$

- $x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$ is a vector of optimization variables
- Objective is linear; feasible set is a polyhedron
- A linear program is a convex optimization problem
- Maximizing $c^T x$ over a polyhedron is also a linear program

Standard and inequality form linear programs

Standard form LP:

$$\begin{aligned} \text{minimize} \quad & c^T x \\ \text{subject to} \quad & Ax = b \\ & x \succeq 0 \end{aligned}$$

- All variables are constrained to be nonnegative. They are the only inequality constraints allowed.
- Widely used in LP software
- Every LP can be converted to standard form

Inequality form LP:

$$\begin{aligned} \text{minimize} \quad & c^T x \\ \text{subject to} \quad & Ax \preceq b \end{aligned}$$

- No equality constraints

Converting LPs to standard form (1)

1. Introduce a **slack variable** $s_i \geq 0$ for each inequality constraint $(i = 1, \dots, m)$
$$g_i^T x \leq h_i \Rightarrow g_i^T x + s_i = h_i$$
2. Express x_i as $x_i = x_i^+ - x_i^-$ with $x_i^+ \geq 0$ and $x_i^- \geq 0$

- The new optimization variables are $s = [s_1, \dots, s_m]^T$, $x^+ = [x_1^+, \dots, x_n^+]^T$, $x^- = [x_1^-, \dots, x_n^-]^T$
- The objective becomes $c^T x = c^T x^+ - c^T x^-$
- One row of the inequality constraint is
$$g_i^T x + s_i = h_i \Leftrightarrow g_i^T x^+ - g_i^T x^- + s_i = h_i$$
- Organize in matrix form

$$\begin{bmatrix} g_1^T \\ \dots \\ g_m^T \end{bmatrix} x^+ - \begin{bmatrix} g_1^T \\ \dots \\ g_m^T \end{bmatrix} x^- + \begin{bmatrix} s_1 \\ \vdots \\ s_m \end{bmatrix} = \begin{bmatrix} h_1 \\ \vdots \\ h_m \end{bmatrix} \Leftrightarrow Gx^+ - Gx^- + s = h$$

$$\begin{aligned} \text{minimize} \quad & c^T x^+ - c^T x^- \\ \text{subject to} \quad & Gx^+ - Gx^- + s = h \\ & Ax^+ - Ax^- = b, \;\; x^+ \succeq 0, \; x^- \succeq 0, \; s \succeq 0 \end{aligned}$$

Converting LPs to standard form (2)

- The problem is written as an inequality form LP as follows

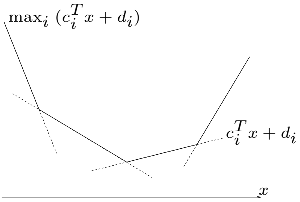
$$\begin{aligned}\text{minimize} \quad & c^T x^+ - c^T x^- \\ \text{subject to} \quad & Gx^+ - Gx^- + s = h \\ & Ax^+ - Ax^- = b \\ & x^+ \succeq 0, \; x^- \succeq 0, \; s \succeq 0\end{aligned}$$

- To see it more clearly, define the variable $y = [x_1^+, \ldots, x_n^+, x_1^-, \ldots, x_n^-, s_1, \ldots, s_m]^T$

$$\begin{aligned}\text{minimize} \quad & \begin{bmatrix} c \\ -c \\ 0_{m\times 1}\end{bmatrix}^T y \\ \text{subject to} \quad & \begin{bmatrix} G & -G & I_{m\times m} \\ A & -A & 0_{p\times n}\end{bmatrix} y = \begin{bmatrix} h \\ b \end{bmatrix} \\ & y \succeq 0\end{aligned} \quad \Leftrightarrow \quad \begin{aligned}\text{minimize} \quad & \tilde{c}^T y \\ \text{subject to} \quad & \tilde{A} y = \tilde{b} \\ & y \succeq 0\end{aligned}$$

Piecewise-linear minimization

$$\begin{aligned} \text{minimize} \quad & \max_{i=1,\ldots,K} \{c_i^T x + d_i\} \\ \text{subject to} \quad & Gx \preceq h \\ & Ax = b \end{aligned}$$



is equivalent to an LP with variables $x \in \mathbb{R}^n$ and $t \in \mathbb{R}$

$$\begin{aligned} \text{minimize} \quad & t \\ \text{subject to} \quad & c_i^T x + d_i \le t,\ i = 1, \ldots, K \\ & Gx \preceq h \\ & Ax = b \end{aligned}$$

Quadratic functions and quadratic forms

- quadratic function
$$f(x) = x^T P x + q^T x + r$$
convex if and only if $P \succeq 0$
- quadratic form
$$f(x) = x^T P x$$
convex if and only if $P \succeq 0$

(P is always symmetric: $P \in \mathbb{S}^n$)

Quadratic programs

Quadratic program (QP):

$$\begin{aligned} \text{minimize} \quad & (1/2)x^T P_0 x + q_0^T x + r_0 \\ \text{subject to} \quad & Gx \preceq h \\ & Ax = b \end{aligned}$$

Quadratically constrained quadratic program (QCQP):

$$\begin{aligned} \text{minimize} \quad & (1/2)x^T P_0 x + q_0^T x + r_0 \\ \text{subject to} \quad & (1/2)x^T P_i x + q_i^T x + r_i \leq 0, \quad i = 1, \ldots, m \\ & Ax = b \end{aligned}$$

- Convex problems if $P_0, P_1, \ldots, P_m$ are positive semidefinite
- Very hard to solve if nonconvex
- LP $\subset$ QP (set $P_0 = 0$) $\subset$ QCQP (set $P_i = 0$, $i = 1, \ldots, m$)

Nonconvex extensions of LP and QP

Boolean LP or **zero-one LP**:

$$\begin{array}{rl} \text{minimize} & c^T x \\ \text{subject to} & Gx \preceq h, \ Ax = b \\ & x_i \in \{0,1\}, \quad i = 1, \ldots, n \end{array}$$

mixed-integer LP (some variables take integer values, some others continuous):

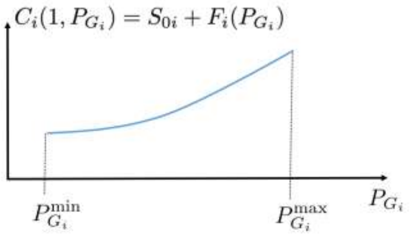
$$\begin{array}{rl} \text{minimize} & c^T x \\ \text{subject to} & Gx \preceq h, \ Ax = b \\ & x_i \in \mathbb{Z}, \, i = 1, \ldots, \tilde{n}; \quad x_i \in \mathbb{R}, \, i = \tilde{n}+1, \ldots, n \end{array}$$

These are in general very hard to solve.

Application: Static unit commitment

- Extend the economic dispatch to include on/off scheduling of units
- Binary variable: $u_i = 1$ if unit i will be scheduled to be on; 0 otherwise
- If unit i is scheduled to be on, it has to produce at least $P_{G_i}^{\min}$
- ... and incur nonzero *startup cost* $S_{0i} > 0$
- Cost function $C_i(u_i, P_{G_i}) = u_i S_{0i} + F_i(P_{G_i})$, where $F_i(P_{G_i})$ is convex in P_{G_i}
- Optimization problem with variables P_{G_i} and u_i, $i = 1, \ldots, n$

$$
\begin{aligned}
\min \quad & \sum_{i=1}^{N} C_i(u_i, P_{G_i}) \\
\text{subj. to} \quad & \sum_{i=1}^{N} P_{G_i} = P_L \\
& u_i P_{G_i}^{\min} \le P_{G_i} \le u_i P_{G_i}^{\max}, \\
& u_i \in \{0,1\},\ i = 1, \ldots, N
\end{aligned}
$$



Second-order cone programs (SOCP)

$$
\begin{aligned}
\min \quad & f^T x \\
\text{subj. to} \quad & \|A_i x + b_i\|_2 \le c_i^T x + d_i, \quad i = 1, \ldots, m \\
& Fx = g
\end{aligned}
$$

- What is the constraint $\|A_i^T x + b_i\|_2 \le c_i^T x + d_i$?
- When is an SOCP an LP? Or a QCQP?
- SOCP includes LP with $A_i = 0$, $i = 1, \ldots, m$
- SOCP includes convex QCQP with $c_i = 0$, $i = 1, \ldots, m$

Facility location (Weber problem)

- Given M points $z_i \in \mathbb{R}^d$ $(i = 1, \ldots, M)$, find the location p so that the sum of distances of p from these points is minimized

$$\min \sum_{i=1}^{M} \|p - z_i\|_2$$

- Using the epigraph trick: Introduce variable t_i such that $\|p - z_i\|_2 \leq t_i$
- The problem becomes SOCP with variables $t_i \in \mathbb{R}$ $(i = 1, \ldots, M)$ and $p \in \mathbb{R}^d$

$$\begin{aligned} \min \quad & \sum_{i=1}^{M} t_i \\ \text{subject to} \quad & \|p - z_i\|_2 \leq t_i, \quad i = 1, \ldots, M \end{aligned}$$

Geometric programming

monomial function with $c \geq 0$, $\alpha_j \in \mathbb{R}$, $\mathrm{dom}(f) = \{x \,|\, x \succ 0\}$

$$f(x) = c x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n}$$

posynomial function: nonnegative sum of monomials

$$f(x) = \sum_{k=1}^{K} c_k x_1^{\alpha_{1k}} x_2^{\alpha_{2k}} \cdots x_n^{\alpha_{nk}}$$

geometric program (this is not a convex problem)

$$\begin{array}{ll} \min & f_0(x) \\ \text{subj. to} & f_i(x) \leq 1, \quad i = 1, \ldots, n \\ & h_i(x) = 1, \quad i = 1, \ldots, p \end{array}$$

with $f_0, \ldots, f_n$ posynomials and $h_1, \ldots, h_p$ monomials

GP in convex form

Change of variables: $y_i = \log x_i \Leftrightarrow x_i = e^{y_i}$
If f(x) is monomial, then $\log f(e^{y_1}, \ldots, e^{y_n})$ is *affine*

$$f(x) = cx_1^{\alpha_1}x_2^{\alpha_2}\cdots x_n^{\alpha_n} \Rightarrow \log f(e^{y_1}, \ldots, e^{y_n}) = \alpha_1 y_1 + \ldots + \alpha_n y_n + \log c$$

If f(x) is posynomial, then $\log f(e^{y_1}, \ldots, e^{y_n})$ is *convex*

$$f(x) = \sum_{k=1}^{K} c_k x_1^{\alpha_{1k}} x_2^{\alpha_{2k}} \cdots x_n^{\alpha_{nk}} \Rightarrow \log f(e^{y_1}, \ldots, e^{y_n}) = \log \sum_{k=1}^{K} e^{\alpha_{1k}y_1 + \ldots + \alpha_{nk}y_n + \log c_k}$$

Geometric program in equivalent convex form with variables $y_1, \ldots, y_n$

$$\begin{array}{ll} \min & \log f_0(e^{y_1}, \ldots, e^{y_n}) \\ \text{subj. to} & \log f_i(e^{y_1}, \ldots, e^{y_n}) \leq 0, \quad i = 1, \ldots, n \\ & \log h_i(e^{y_1}, \ldots, e^{y_n}) = 0, \quad i = 1, \ldots, p \end{array}$$

Semidefinite programming

Semidefinite program (SDP) with variable $x \in \mathbb{R}^n$

$$\begin{aligned} \min \quad & c^T x \\ \text{subj. to} \quad & \sum_{i=1}^{n} x_i F_i + G \preceq 0 \\ & Ax = b \end{aligned}$$

where $F_1, \ldots, F_n, G \in \mathbb{S}^m$ (i.e., $F_1, \ldots, F_n, G$ are **symmetric** matrices)

- Constaint $\sum_{i=1}^{n} x_i F_i + G \preceq 0$ is called a linear matrix inequality (LMI)

Standard and inequality form SDP

Standard SDP with variable $X \in \mathbb{S}^n$ (symmetric matrix)

$$\begin{aligned} \min \quad & \operatorname{tr}(CX) \\ \text{subj. to} \quad & \operatorname{tr}(A_i X) = b_i, \quad i = 1, \ldots, m \\ & X \succeq 0 \end{aligned}$$

where $C, A_i \in \mathbb{S}^n$
Inequality SDP with variable $x \in \mathbb{R}^n$

$$\begin{aligned} \min \quad & c^T x \\ \text{subj. to} \quad & \sum_{i=1}^{n} x_i A_i \preceq B \end{aligned}$$

where $A_i, B \in \mathbb{S}^m$

LP vs SDP

General LP

$$\begin{aligned} &\min\ c^T x \\ \text{s. to }\ & Gx \preceq h \\ & Ax = b \end{aligned}$$

Standard LP

$$\begin{aligned} &\min\ c^T x \\ \text{s. to }\ & Ax = b \\ & x \succeq 0 \end{aligned}$$

Inequality LP

$$\begin{aligned} &\min\ c^T x \\ \text{s. to }\ & Ax \preceq b \end{aligned}$$

General SDP

$$\begin{aligned} &\min\ c^T x \\ \text{s. to }\ & \sum_{i=1}^{n} x_i F_i + G \preceq 0 \\ & Ax = b \end{aligned}$$

$$G, F_i \in \mathbb{S}^m$$

Standard SDP

$$\begin{aligned} &\min\ \mathrm{tr}(CX) \\ \text{s. to }\ & \mathrm{tr}(A_i X) = b_i \\ & (i = 1, \ldots, m) \\ & X \succeq 0 \end{aligned}$$

$$C, A_i \in \mathbb{S}^n$$

Inequality SDP

$$\begin{aligned} &\min\ c^T x \\ \text{s. to }\ & \sum_{i=1}^{n} x_i A_i \preceq B \end{aligned}$$

$$A_i, B \in \mathbb{S}^m$$

- SDP becomes LP if the respective matrices are diagonal

LMIs with matrix variable

- Example: Given $A \in \mathbb{R}^{m \times m}$, find $P \succ 0$ such that $A^T P + PA \prec 0$
- In control theory, the previous problem amounts to solving a Lyapunov inequality
- It can be cast as an SDP feasibility problem
- First, we convert strict inequalities to "greater than or equal," i.e., $P \succeq \epsilon_1 I$ and $A^T P + PA \preceq \epsilon_2 I$, where $\epsilon_1 > 0$ and $\epsilon_2 > 0$ are chosen by us and are small

$$\begin{bmatrix} -P & 0 \\ 0 & A^T P + PA \end{bmatrix} + \begin{bmatrix} \epsilon_1 I & 0 \\ 0 & -\epsilon_2 I \end{bmatrix} \preceq 0$$

- How can we convert to LMI of the form $\sum_{i=1}^{n} x_i F_i + G \preceq 0$?

Bottomline: SDPs must contain linear forms

A problem is an SDP if it includes any combination of the following:

- An objective linear in a vector variable, e.g., $c^T x$
- An objective linear in a matrix variable, that is, $\mathrm{Tr}(BY)$ (where B and Y are symmetric; and Y is positive or negative semidefinite)
- Multiple LMIs of the form $\sum_i^N x_i F_i + G \preceq 0$
- Linear inequalities and equalities
- LMIs with $f^T x$ as matrix entry
- LMIs that include a matrix variable (e.g., $A^T P + PA \preceq -\epsilon I$)
- Linear equalities or linear inequalities that involve the trace of a matrix variable

Convex optimization modeling languages such as CVX or YALMIP accept any combination of the previous objectives or constraints.

Hierarchy of convex optimization problems

LP $\subset$ (convex) QP $\subset$ (convex) QCQP $\subset$ SOCP $\subset$ SDP

- SDP includes SOCP
- As SOCP includes QCQP, QP, and LP, we conclude that SDP is the most general class
- GP is outside of this hierarchy
- Recall that there are also nonconvex QPs and QCQPs (very hard to solve in general)
- A more restricted problem class is more computationally efficient (faster to solve)
    - Better to have LP with a million variables than SDP with a million variables

Questions And Suggestions?

Thank You!

Please visit

https://lab.vanderbilt.edu/taha/

IFF you want to know more ☺