EE5138 OPTIMIZATION FOR ELECTRICAL ENGINEERING/ EE6138 OPTIMIZATION FOR ELECTRICAL ENGINEERING (ADVANCED)

Lecture 0: Introduction

About the instructor

- Professor, ECE Department, NUS
- E-mail: elezhang@nus.edu.sg
- Ph.D., EE Department, Stanford University, USA
- Current Research Interests:
    1. Optimization Methods (for applications in Signal Processing, Communications, Information Theory, and Networks)
    2. Wireless Communications (MIMO, UAV, 6G)
    3. Wireless Power Transfer
- More information available at website: http://www.ece.nus.edu.sg/stfpage/elezhang/

Course information

- Course website: LumiNUS (lecture slide, practice problem & solution, assignment, announcement, etc.)
- Textbook: S. Boyd and L. Vandenberghe, “Convex Optimization” (available at NUS bookstore, and online at http://www.stanford.edu/~boyd/cvxbook/)
- Lecture slides on convex optimization based mainly on the slides of Boyd’s class at Stanford: http://www.stanford.edu/class/ee364a/
- Midterm exam (20%, 1 hour, open-book and online)
- One programming assignment (10%, take-home and online submission)
- Final exam (70%, 2 hours; if onsite, then closed-book, 1 A4-size help sheet allowed; if online, then open-book; we will decide the exam mode later)
- EE5138 and EE6138 share the same lectures, but differ in the programming assignment and final exam

Course topics and assessment

- Introduction (Lecture 0)
  - textbook: chapter 1, appendix A (mathematical preliminary, please read it yourself)
- Convex sets (Lecture 1)
  - textbook: chapter 2
- Convex functions (Lecture 2)
  - textbook: chapter 3
- Convex optimization problems (Lecture 3)
  - textbook: chapter 4
- Duality and KKT conditions (Lecture 4)
  - textbook: chapter 5
- Numerical algorithms (Lecture 5)
  - textbook: chapter 9, 10, 11

Theory (Midterm Exam covers Lecture 1 and 2)

Algorithm (Programing Assignment covers Lecture 5)

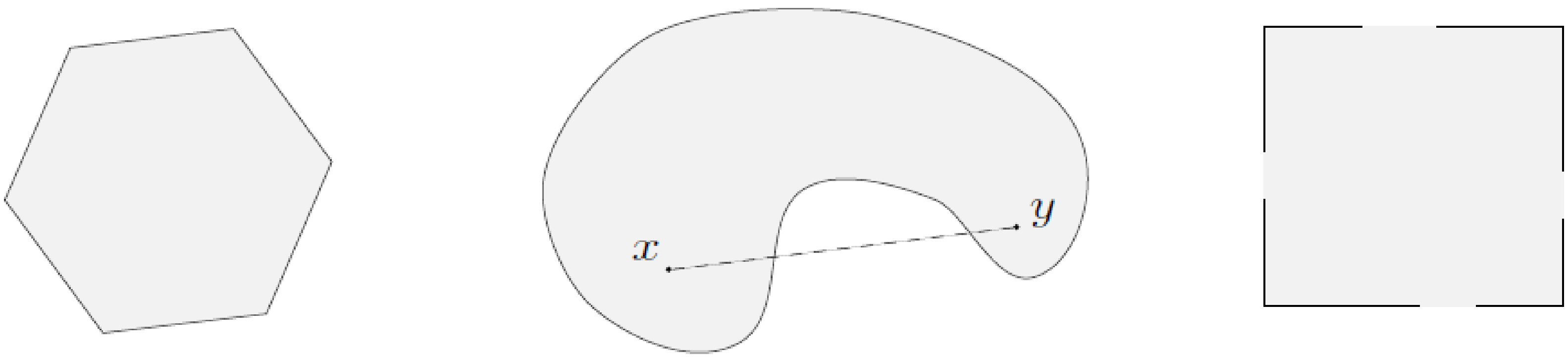
Final Exam covers Lecture 1-5

Convex Set

$C \subseteq \mathbf{R}^n$ is convex if

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

convex not convex

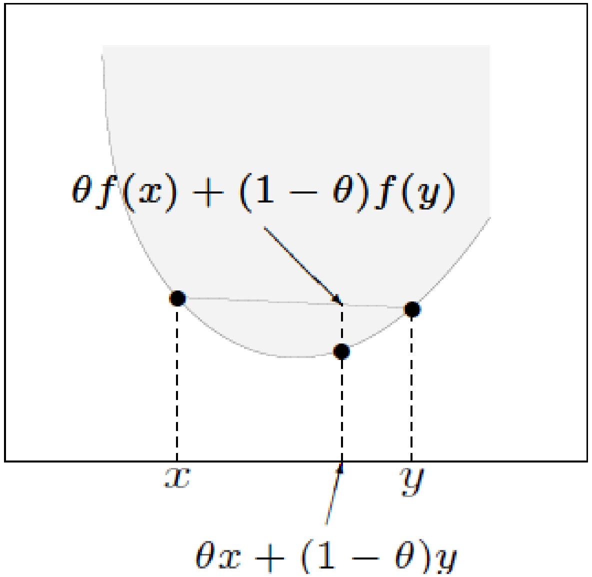


(more later!)

Convex Function

$f : \mathbf{R}^n \rightarrow \mathbf{R}$ is convex if

$$x, y \in \mathbf{R}^n, \quad \theta \in [0,1] \quad \Longrightarrow \quad f(\theta x + (1-\theta)y) \leq \theta f(x) + (1-\theta) f(y)$$



(more later!)

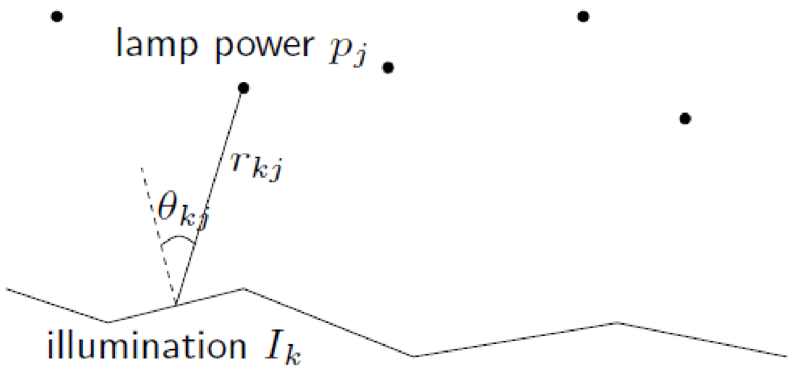
Convex Optimization Problem

minimize $f(x)$ subject to $x \in C$, with f convex, C convex

- can be solved numerically with great efficiency
- have extensive, useful theory
- occur often in engineering problems
- often go unrecognized
- **tractable** in theory and practice: there exist algorithms such that
  - computation time small, grows gracefully with problem size
  - global solutions attained
  - non heuristic stopping criteria; provable lower bounds
  - handle nondifferentiable as well as smooth problems
- **duality theory**:
  - necessary and sufficient conditions for global optimality
  - certificates that **prove** infeasibility or lower bounds on objective
  - sensitivity analysis w.r.t. changes in f, C

Example

m lamps illuminating n (small, flat) patches



intensity I_k at patch k depends linearly on lamp powers p_j:

$$I_k = \sum_{j=1}^{m} a_{kj} p_j, \qquad a_{kj} = r_{kj}^{-2} \max\{\cos\theta_{kj}, 0\}$$

problem: achieve desired illumination I_{des} with bounded lamp powers

$$\begin{array}{ll} \text{minimize} & \max_{k=1,\ldots,n} |\log I_k - \log I_{\text{des}}| \\ \text{subject to} & 0 \le p_j \le p_{\text{max}}, \quad j = 1, \ldots, m \end{array}$$

how to solve?

1. use uniform power: $p_j = p$, vary p
2. use least-squares:

$$\text{minimize} \quad \sum_{k=1}^{n} (I_k - I_{\text{des}})^2$$

   round p_j if $p_j > p_{\text{max}}$ or $p_j < 0$

3. use weighted least-squares:

$$\text{minimize} \quad \sum_{k=1}^{n} (I_k - I_{\text{des}})^2 + \sum_{j=1}^{m} w_j (p_j - p_{\text{max}}/2)^2$$

   iteratively adjust weights w_j until $0 \le p_j \le p_{\text{max}}$

4. use linear programming:

$$\begin{array}{ll} \text{minimize} & \max_{k=1,\ldots,n} |I_k - I_{\text{des}}| \\ \text{subject to} & 0 \le p_j \le p_{\text{max}}, \quad j = 1, \ldots, m \end{array}$$

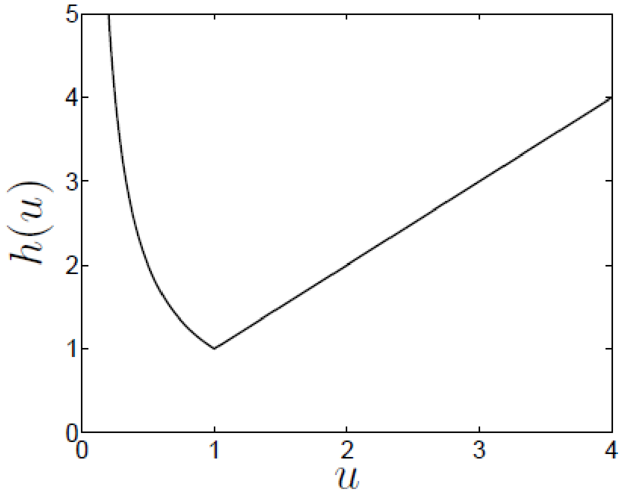
   which can be solved via linear programming (Why?)

of course these are approximate (suboptimal) 'solutions'

5. use convex optimization: problem is equivalent to (Why?)

$$\begin{array}{ll} \text{minimize} & f_0(p) = \max_{k=1,\ldots,n} h(I_k/I_{\text{des}}) \\ \text{subject to} & 0 \leq p_j \leq p_{\max}, \quad j = 1, \ldots, m \end{array}$$

with $h(u) = \max\{u, 1/u\}$



f_0 is convex because maximum of convex functions is convex

exact solution obtained with effort $\approx$ modest factor $\times$ least-squares effort

Variants: two additional constraints

1. no more than half total power is in any 10 lamps
2. no more than half of the lamps are on ($p_i > 0$)

does adding (1) or (2) complicate the problem?

- with (1), still easy to solve (Why?)
- with (2), **extremely difficult** to solve (Why?)

Moral:

- without the proper background (*i.e.*, this course) very easy problems can appear quite similar to very difficult problems
- (untrained) intuition doesn't always work

Application of duality

1. feasibility problem: find $x \in C$

- convex optimization methods

$$\text{either find } x \in C\text{, or yield proof that } C = \emptyset$$

c.f. conventional case: algorithms either

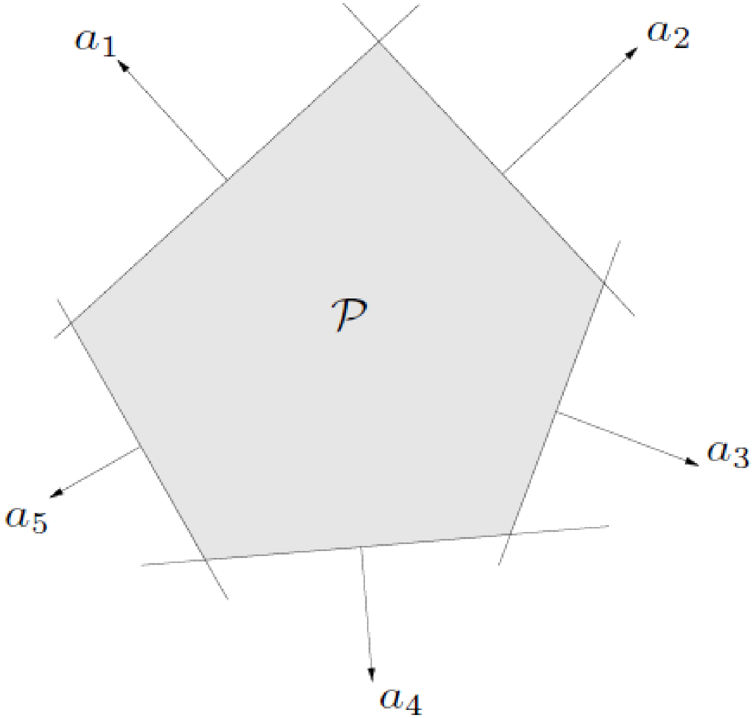
$$\text{find } x \in C\text{, or do not find } x \in C$$

- convex case: feasibility algs. return **yes** or **no**

  general case: feasibility algs. return **yes** or **maybe**

example:

$$\mathcal{P} = \left\{ x \mid a_k^T x \le b_k,\, k = 1, \ldots, m \right\}$$



how could you know $\mathcal{P} = \emptyset$?

Here is how:

suppose $\lambda_i \geq 0$, $\sum \lambda_i a_i = 0$, $\sum \lambda_i b_i < 0$

then $a_i^T x \leq b_i$, $i = 1, \ldots, m$, implies

$$0 \leq \sum_i \lambda_i (b_i - a_i^T x) = \sum_i \lambda_i b_i < 0$$

$\longrightarrow$ **Contradiction!**

we conclude:

$$\exists \lambda_i \geq 0, \ \sum \lambda_i a_i = 0, \ \sum \lambda_i b_i < 0 \Longrightarrow \mathcal{P} = \emptyset$$

we say λ_i's are a *certificate* or *proof* of infeasibility

fact (convexity): if $a_i^T x \leq b_i$ is infeasible, then there exists a certificate proving it!

2. stopping criterion

convex optimization algorithms provide at iteration k

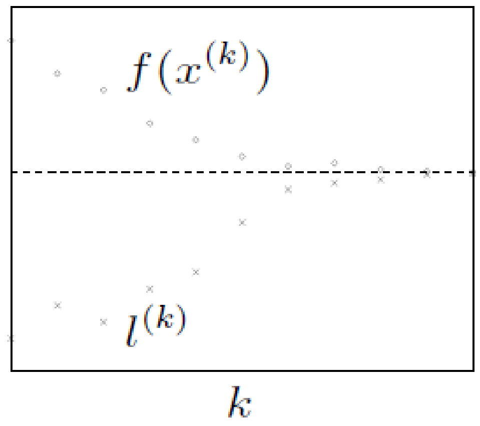
$$x^{(k)} \in C, \text{ a suboptimal point,}$$

with $f(x^{(k)}) \rightarrow f^\star = \inf_{x \in C} f(x)$ as $k \rightarrow \infty$

and a provable lower bound on optimal value, *i.e.*,

$$l^{(k)} \text{ s.t. } l^{(k)} \leq f^\star$$

with $l^{(k)} \rightarrow f^\star$ as $k \rightarrow \infty$



at iteration k we **know**

$$f^\star \in \left[l^{(k)}, f(x^{(k)})\right]$$

hence stopping criterion

$$\textbf{until } f(x^{(k)}) - l^{(k)} \leq \epsilon$$

guarantees on exit

$$\text{absolute error } = \left| f(x^{(k)}) - f^\star \right| \leq \epsilon$$

similarly, stopping criterion

$$\textbf{until } \left(l^{(k)} > 0 \ \ \& \ \ \frac{f(x^{(k)}) - l^{(k)}}{l^{(k)}} \leq \epsilon \right)$$

guarantees (for $f^\star > 0$) on exit

$$\text{relative error } = \frac{f(x^{(k)}) - f^\star}{f^\star} \leq \epsilon$$

What we will/won’t cover

what we will cover

- recognizing & exploiting convexity in engineering context
- ideas of convex optimization and duality
- a few example algorithms (e.g., Newton method, barrier method)
- convex relaxation method for non-convex problems

what we won’t do

- details of convex analysis
- details of optimization theory (regularity conditions, constraint qualifications,. . .)
- encyclopedia of algorithms (sub-gradient, decomposition methods,. . .)
- convergence analysis
- details of non-convex optimization (sequential convex programming, branch & bound,. . .)

What fraction of 'real' problems are convex?

- by no means all
- many more than are recognized

example: linear programming (LP)

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & a_i^T x \leq b_i, \;\; i = 1, \ldots, m \end{array}$$

- convex, but no "closed form" solution
- very large LPs solved very quickly in practice
- extensive, useful theory

But, how many problems are LPs?

- 1940s: "the real world is nonlinear, hence LP silly"
- many nonlinear (convex) optimization techniques developed to date (e.g., QP, SOCP, SDP, GP, . . .)

Why Convex Optimization?

convex optimization

- dividing line between 'easy' and 'hard' optimization problems
- no local min; always global optimal solution;
- no such headaches as stepsize selection, initialization, etc
- handles* some problems very well; highly efficient algorithms exist
- can say a lot about it

other *wildly* used methods: simulated annealing, genetic algorithms, neural networks, ...

- handle† many problems
- slow convergence (they are too general to be efficient)
- can say very little about it

* means a lot — global solutions, always works, worst case computation time, etc.

† means much less — local solutions (sometimes), no complexity theory, etc.