# Introduction to Convex Optimization

Prof. Daniel P. Palomar

The Hong Kong University of Science and Technology (HKUST)

ELEC5470/IEDA6100A - Convex Optimization
Fall 2020-21, HKUST, Hong Kong

# Outline

1. Optimization Problems
2. Examples
3. Solving Optimization Problems
4. More Examples
5. This Course
6. References

# Outline

1. Optimization Problems
2. Examples
3. Solving Optimization Problems
4. More Examples
5. This Course
6. References

# Optimization problem

- General optimization problem in standard form:

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

  where

  $x = (x_1, \ldots, x_n)$ is the optimization variable
  $f_0 : \mathbf{R}^n \longrightarrow \mathbf{R}$ is the objective function
  $f_i : \mathbf{R}^n \longrightarrow \mathbf{R}, \quad i = 1, \ldots, m$ are inequality constraint functions
  $h_i : \mathbf{R}^n \longrightarrow \mathbf{R}, \quad i = 1, \ldots, p$ are equality constraint functions.

- **Goal**: find an optimal solution $x^\star$ that minimizes $f_0$ while satisfying all the constraints.

# Outline

1. Optimization Problems
2. Examples
3. Solving Optimization Problems
4. More Examples
5. This Course
6. References

# Examples

Convex optimization is currently used in many different areas:

- circuit design (start-up named Barcelona in Silicon Valley)
- signal processing (e.g., filter design)
- communication systems (e.g., transceiver design, beamforming design, ML detection, power control in wireless)
- financial engineering (e.g., portfolio design, index tracking)
- image proc. (e.g., deblurring, compressive sensing, blind separation)
- machine learning
- biomedical applications (e.g., analysis of DNA)

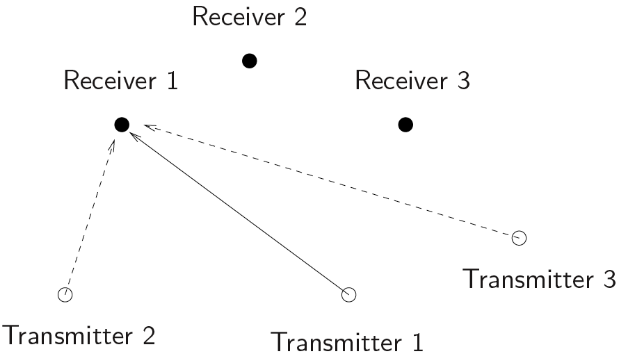
# Elements in the formulation

- An optimization problem has three basic elements:
  1. variables,
  2. constraints, and
  3. objective.

- Example: device sizing in electronic circuits:
  - variables: device widths and lengths
  - constraints: manufacturing limits, timing requirements, max area
  - objective: power consumption
- Example: portfolio optimization:
  - variables: amounts invested in different assets
  - constraints: budget, max investments per asset, min return
  - objective: overall risk or return variance.

# Example: Power control in wireless networks

- Consider a wireless network with *n* logical transmitter/receiver pairs:



- **Goal**: design the power allocation so that each receiver receives minimum interference from the other links.

# Example: Power control in wireless networks

- The signal-to-interference-plus-noise-ratio (SINR) at the $i$th receiver is

$$\mathsf{sinr}_i = \frac{p_i G_{ii}}{\sum_{j \neq i} p_j G_{ij} + \sigma_i^2}$$

where

$p_i$ is the power used by the $i$th transmitter<br>
$G_{ij}$ is the path gain from transmitter $j$ to receiver $i$<br>
$\sigma_i^2$ is the noise power at the $i$th receiver.

- **Problem**: maximize the weakest SINR subject to power constraints $0 \leq p_i \leq p_i^{\max}$:

$$\begin{array}{ll} \underset{\mathbf{p}}{\text{maximize}} & \min\limits_{i=1,\ldots,n} \frac{p_i G_{ii}}{\sum_{j \neq i} p_j G_{ij} + \sigma_i^2} \\ \text{subject to} & 0 \leq p_i \leq p_i^{\max} \qquad i = 1, \ldots, n. \end{array}$$

# Outline

1 Optimization Problems

2 Examples

3 Solving Optimization Problems

4 More Examples

5 This Course

6 References

# Solving optimization problems

- General optimization problems are very difficult to solve (either long computation time or not finding the best solution).
- Exceptions: least-squares problems, linear programming problems, and convex optimization problems.

- **Least-squares (LS)** [Gauss, 1795]:

$$\underset{x}{\text{minimize}} \quad \|Ax - b\|_2^2$$

  - solving LS problems: closed-form solution $x^\star = \left(A^T A\right)^{-1} A^T b$ for which there are reliable and efficient algorithms; mature technology
  - using LS: easy to recognize

# Solving optimization problems

- **Linear Programming (LP)**:

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

  - solving LP problems: no closed-form solution, but reliable and efficient algorithms and software; mature technology
  - using LP: not as easy to recognize as LS problems, a few standard tricks to convert problems into LPs

- **Convex optimization**:

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

  - solving convex problems: no closed-form solution, but still reliable and efficient algorithms and software; almost a technology
  - using convex optimization: often difficult to recognize, many tricks for transforming problems into convex form.

# Nonconvex optimization problems

- Nonconvex optimization problems are generally very difficult to solve, although there are some rare exceptions.

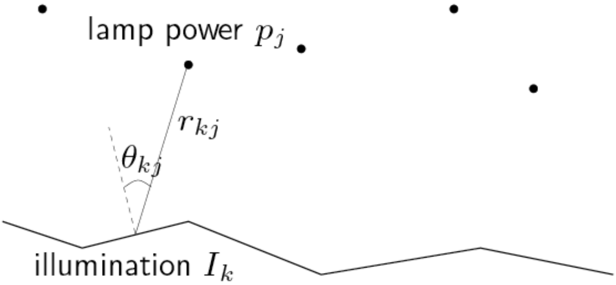
- In general, they require either a long computation time or the compromise of not always finding the optimal solution:
  - **local optimization**: fast algorithms, but no guarantee of global optimality, only local solution around the initial point
  - **global optimization**: worst-case complexity grows exponentially with problem size, but finds global solution.

# Outline

1. Optimization Problems
2. Examples
3. Solving Optimization Problems
4. More Examples
5. This Course
6. References

# Example: Lamp illumination problem

- Consider *m* lamps illuminating *n* small flat patches:



- **Goal**: achieve a desired illumination $I_{\text{des}}$ on all patches with bounded lamp powers.

# Example: Lamp illumination problem

- The intensity $I_k$ at patch $k$ depends linearly on the lamp powers $p_j$:

$$I_k = \sum_{j=1}^{m} a_{kj} p_j$$

where the coefficients $a_{kj}$ are given by $a_{kj} = \cos\theta_{kj} / r_{kj}^2$.

- Problem formulation: since the illumination is perceived logarithmically by the eye, a good formulation of the problem is

$$\begin{array}{ll} \underset{I_1,\ldots,I_n,p_1,\ldots,p_m}{\text{minimize}} & \max_k \left|\log I_k - \log I_{\text{des}}\right| \\ \text{subject to} & 0 \leq p_j \leq p_{\max}, \quad j = 1,\ldots,m \\ & I_k = \sum_{j=1}^{m} a_{kj} p_j, \quad k = 1,\ldots,n. \end{array}$$

- How to solve the problem?
- Answer: It depends on how much you know about optimization.

# Solving the lamp illumination problem

1. If you don't know anything, then you just take a heuristic guess like using a uniform power $p_j = p$, perhaps trying different values of $p$.

2. If you know about least-squares, then approximate the problem as
$$\begin{array}{ll} \underset{I_1,\ldots,I_n,p_1,\ldots,p_m}{\text{minimize}} & \sum_{k=1}^{n} \left(I_k - I_{\text{des}}\right)^2 \\ \text{subject to} & I_k = \sum_{j=1}^{m} a_{kj} p_j, \quad k = 1, \ldots, n. \end{array}$$
and then clip $p_j$ if $p_j > p_{\max}$ or $p_j < 0$.

3. If you know about linear programming, then approximate the problem as
$$\begin{array}{ll} \underset{I_1,\ldots,I_n,p_1,\ldots,p_m}{\text{minimize}} & \max_k \left|I_k - I_{\text{des}}\right| \\ \text{subject to} & 0 \leq p_j \leq p_{\max}, \quad j = 1, \ldots, m \\ & I_k = \sum_{j=1}^{m} a_{kj} p_j, \quad k = 1, \ldots, n, \end{array}$$
which may not look as an LP but it is!

# Solving the lamp illumination problem

1 If you don't know anything, then you just take a heuristic guess like using a uniform power $p_j = p$, perhaps trying different values of $p$.

2 If you know about least-squares, then approximate the problem as

$$\begin{array}{ll} \underset{I_1,\ldots,I_n,p_1,\ldots,p_m}{\text{minimize}} & \sum_{k=1}^{n} \left(I_k - I_{\text{des}}\right)^2 \\ \text{subject to} & I_k = \sum_{j=1}^{m} a_{kj} p_j, \quad k = 1, \ldots, n. \end{array}$$

and then clip $p_j$ if $p_j > p_{\max}$ or $p_j < 0$.

3 If you know about linear programming, then approximate the problem as

$$\begin{array}{ll} \underset{I_1,\ldots,I_n,p_1,\ldots,p_m}{\text{minimize}} & \max_k \left|I_k - I_{\text{des}}\right| \\ \text{subject to} & 0 \leq p_j \leq p_{\max}, \quad j = 1, \ldots, m \\ & I_k = \sum_{j=1}^{m} a_{kj} p_j, \quad k = 1, \ldots, n, \end{array}$$

which may not look as an LP but it is!

# Solving the lamp illumination problem

1. If you don't know anything, then you just take a heuristic guess like using a uniform power $p_j = p$, perhaps trying different values of $p$.

2. If you know about least-squares, then approximate the problem as

$$\begin{array}{ll} \underset{I_1,\ldots,I_n,p_1,\ldots,p_m}{\text{minimize}} & \sum_{k=1}^{n} \left(I_k - I_{\text{des}}\right)^2 \\ \text{subject to} & I_k = \sum_{j=1}^{m} a_{kj} p_j, \quad k = 1, \ldots, n. \end{array}$$

and then clip $p_j$ if $p_j > p_{\max}$ or $p_j < 0$.

3. If you know about linear programming, then approximate the problem as

$$\begin{array}{ll} \underset{I_1,\ldots,I_n,p_1,\ldots,p_m}{\text{minimize}} & \max_k \left|I_k - I_{\text{des}}\right| \\ \text{subject to} & 0 \leq p_j \leq p_{\max}, \quad j = 1, \ldots, m \\ & I_k = \sum_{j=1}^{m} a_{kj} p_j, \quad k = 1, \ldots, n, \end{array}$$

which may not look as an LP but it is!

# Solving the lamp illumination problem

4 If you know about convex optimization, after staring at the problem long enough, you may realize that you can actually reformulate the original problem in convex form and then find the global solution:

$$\begin{array}{ll} \underset{I_1,\ldots,I_n,p_1,\ldots,p_m}{\text{minimize}} & \max_k\, h\left(I_k/I_{\text{des}}\right) \\ \text{subject to} & 0 \le p_j \le p_{\max}, \quad j = 1,\ldots,m \\ & I_k = \sum_{j=1}^{m} a_{kj}p_j, \quad k = 1,\ldots,n, \end{array}$$

where $h\left(u\right) = \max\left\{u, 1/u\right\}$.

# Additional constraints

- **Additional constraints**: does adding the constraints below complicate the problem?

  (a) no more than half of total power is in any 10 lamps

  (b) no more than half of the lamps are on ($p_j > 0$).

- Answer:

  (a) does not complicate the problem, whereas

  (b) makes the problem extremely difficult.

- **Moral**: untrained intuition doesn't always work; one needs to obtain the proper background and develop the right intuition to discern between difficult and easy problems.

# Additional constraints

- **Additional constraints**: does adding the constraints below complicate the problem?

  (a) no more than half of total power is in any 10 lamps

  (b) no more than half of the lamps are on ($p_j > 0$).

- Answer:

  (a) does not complicate the problem, whereas

  (b) makes the problem extremely difficult.

- **Moral**: untrained intuition doesn't always work; one needs to obtain the proper background and develop the right intuition to discern between difficult and easy problems.

# Additional constraints

- **Additional constraints**: does adding the constraints below complicate the problem?
  - (a) no more than half of total power is in any 10 lamps
  - (b) no more than half of the lamps are on ($p_j > 0$).

- Answer:
  - (a) does not complicate the problem, whereas
  - (b) makes the problem extremely difficult.

- **Moral**: untrained intuition doesn't always work; one needs to obtain the proper background and develop the right intuition to discern between difficult and easy problems.

# Additional constraints

- **Additional constraints**: does adding the constraints below complicate the problem?
  - (a) no more than half of total power is in any 10 lamps
  - (b) no more than half of the lamps are on ($p_j > 0$).
- Answer:
  - (a) does not complicate the problem, whereas
  - (b) makes the problem extremely difficult.
- **Moral**: untrained intuition doesn't always work; one needs to obtain the proper background and develop the right intuition to discern between difficult and easy problems.

# Historical snapshot

- **Theory** (convex analysis): ca1900-1970 (e.g. Rockafellar)
- **Algorithms**:
  - 1947: simplex algorithm for linear programming (Dantzig)
  - 1960s: early interior-point methods (Fiacco & McCormick, Dikin)
  - 1970s: ellipsoid method and other subgradient methods
  - 1980s: polynomial-time interior-point methods for linear programming (Karmakar 1984)
  - late 1980s-now: polynomial-time interior-point methods for nonlinear convex optimization (Nesterov & Nemirovski 1994)
- **Applications**:
  - before 1990s: mostly in operations research; few in engineering
  - since 1990: many new applications in engineering and new problem classes (SDP, SOCP, robust optim.)

# Outline

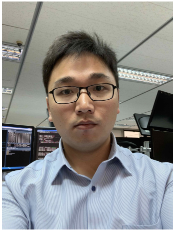
1. Optimization Problems
2. Examples
3. Solving Optimization Problems
4. More Examples
5. This Course
6. References

# Course goals and topics

- **Goal**: to introduce convex optimization theory and to illustrate its use with many recent applications with emphasis on
  - i) the art of unveiling the hidden convexity of problems
  - ii) a proper characterization of the solution either analytically or algorithmically.
- The course follows a case-study approach with applications in the areas of signal processing, finance, machine learning, and big data.
- Problems covered include: portfolio optimization in financial markets, filter design, beamforming design in wireless communications, classification in machine learning, circuit design, robust designs under uncertainty, sparse optimization, low-rank optimization, graph learning from data, discrete maximum likelihood decoding, network optimization, distributed algorithms, Internet protocol design, etc.

# Course team

- Fantastic TAs (Rui Zhou and Ze Vinicius):



- Professor Daniel P. Palomar:

# Outline

1. Optimization Problems

2. Examples

3. Solving Optimization Problems

4. More Examples

5. This Course

6. References

# References

- Stephen Boyd and Lieven Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge University Press, 2004.

  https://web.stanford.edu/~boyd/cvxbook/bv_cvxbook.pdf

- Daniel P. Palomar and Yonina C. Eldar, Eds., *Convex Optimization in Signal Processing and Communications*, Cambridge University Press, 2009.
- Ben Tal & Nemirovsky, *Lectures on Modern Convex Optimization*. SIAM 2001.
- Nesterov & Nemirovsky, *Interior-point Polynomial Algorithms in Convex Programming*. SIAM 1994.

# Thanks

For more information visit:

https://www.danielppalomar.com