MECH 6327: Convex Optimization in Systems and Control

Professor Tyler Summers

January 20, 2021

today

- course mechanics
- course overview

learning goals:

- appreciate ubiquity of optimization
- understand fundamental difficulty of general optimization
- introduce convex optimization as a subclass of problems that can be solved globally, efficiently, reliably

today

- course mechanics
- course overview

learning goals:

- appreciate ubiquity of optimization
- understand fundamental difficulty of general optimization
- introduce convex optimization as a subclass of problems that can be solved globally, efficiently, reliably

- please interrupt frequently with any questions or comments!

course info

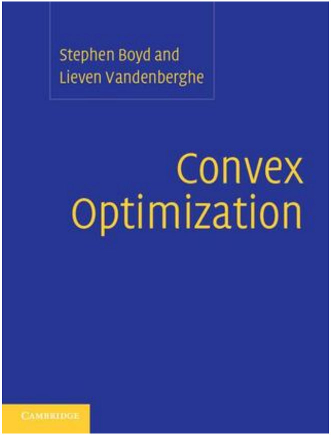
- mondays + wednesdays, 4:00-5:15pm
- (virtual) classroom: MS Teams
- prerequisites
  - MECH 6300 + "sufficient mathematical and programming maturity", solid understanding of linear algebra, calculus, probability, and at least one programming language
  - not dependencies, but may increase appreciation: other courses in control, optimization, and dynamical systems
- office hours
  - by appointment via MS Teams
- grade: 5-6 homeworks (35%), class participation/scribing (5%), short presentation (20%), final exam/course project (40%)

textbook and other references

required textbook:

Convex Optimization, by Stephen Boyd and Lieven Vandenberghe

available free on Prof. Boyd's website!



required software: Matlab (+ CVX/YALMIP modeling package)

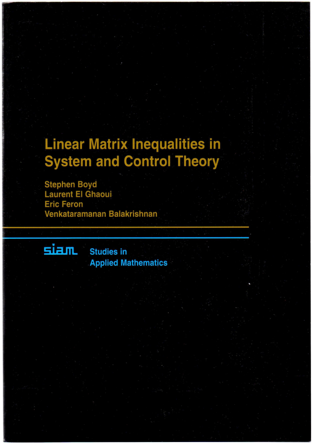
(other languages acceptable, e.g., Python, Julia)

textbook and other references

another great reference:

Linear Matrix Inequalities in System and Control Theory

by Stephen Boyd, Laurent El Ghaoui, Eric Feron, Venkataramanan Balakrishnan (also free on Prof. Boyd's website!)



textbook and other references

convex optimization map

mathematical optimization problems

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & x \in \mathcal{X} \end{array}$$

- $x = [x_1, ..., x_n]^T \in \mathbf{R}^n$: optimization/decision variables
- $f : \mathbf{R}^n \to \mathbf{R}$: objective/cost function
- $\mathcal{X} \subseteq \mathbf{R}^n$: constraint/feasible set

mathematical optimization problems

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & x \in \mathcal{X} \end{array}$$

- $x = [x_1, ..., x_n]^T \in \mathbf{R}^n$: optimization/decision variables
- $f : \mathbf{R}^n \to \mathbf{R}$: objective/cost function
- $\mathcal{X} \subseteq \mathbf{R}^n$: constraint/feasible set
  - usually have an explicit *algebraic* description of $\mathcal{X}$:

    $$\mathcal{X} = \{x \in \mathbf{R}^n \mid g_i(x) \leq 0,\ i = 1, ..., m, \quad h_i(x) = 0,\ i = 1, ..., p\}$$

    where $g_i : \mathbf{R}^n \to \mathbf{R}$: inequality constraint functions
    $h_i : \mathbf{R}^n \to \mathbf{R}$: equality constraint functions

mathematical optimization problems

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & x \in \mathcal{X} \end{array}$$

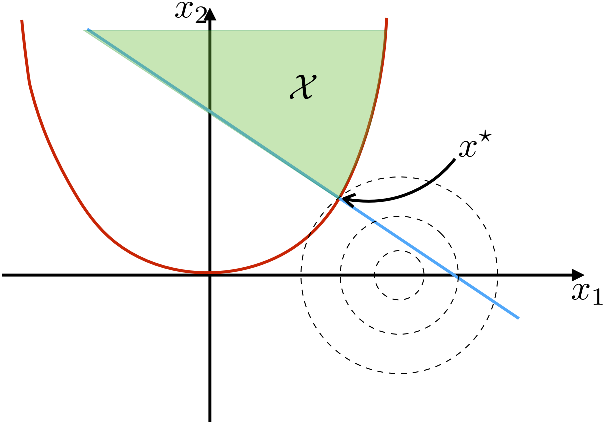
- $x = [x_1, ..., x_n]^T \in \mathbf{R}^n$: optimization/decision variables
- $f : \mathbf{R}^n \to \mathbf{R}$: objective/cost function
- $\mathcal{X} \subseteq \mathbf{R}^n$: constraint/feasible set
  - usually have an explicit *algebraic* description of $\mathcal{X}$:

    $$\mathcal{X} = \{x \in \mathbf{R}^n \mid g_i(x) \leq 0,\ i = 1, ..., m, \quad h_i(x) = 0,\ i = 1, ..., p\}$$

    where $g_i : \mathbf{R}^n \to \mathbf{R}$: inequality constraint functions
    $h_i : \mathbf{R}^n \to \mathbf{R}$: equality constraint functions
- goal: find an **optimal solution** $x^\star$ with smallest value of f among all vectors that satisfy the constraints
  - find any $x^\star \in \mathcal{X}$ such that $f(x^\star) \leq f(x), \quad \forall x \in \mathcal{X}$

a simple example to illustrate the geometry

$$\begin{aligned} \min \quad & (x_1 - 1.5)^2 + x_2^2 \\ \text{subject to} \quad & x_1^2 - x_2 \le 0 \\ & x_1 + x_2 \ge 2 \end{aligned}$$



motivation: core theory + applications

convex optimization occupies an increasingly important position in many disciplines, including systems and control

also there are tons of applications of optimization:

- automatic control
- robot path planning and control
- autonomous driving
- scheduling in transportation networks
- operation of a smart power grid
- operation of a smart building, data center, etc.
- machine learning and data analysis
- supply chain optimization
- medical treatment planning
- multi-period investment
- internet ad display
- revenue management

... and many, many others!

examples in engineering and finance

portfolio investment

- variables: amounts invested in different assets
- constraints: total budget, no more than 20% invested in one asset, risk limit
- objective: maximize expected return

smart grid operation

- variables: power setpoints of generators, controllable loads
- constraints: power balance, device limits, network limits
- objective: minimize total cost of delivered power

fitting a model to data (i.e., machine learning)

- variables: model parameters
- constraints: parameter limits, prior info
- objective: minimize data misfit or prediction error

activity

- describe other examples of optimization problems in engineering (and beyond)
- identify variables, constraints, objectives

examples in systems and control

system identification

- variables: dynamic model parameters
- constraints: parameter limits, prior info
- objective: minimize prediction error

state estimation

- variables: state estimates
- constraints: dynamic model, state limits
- objective: minimize estimation error

trajectory optimization

- variables: control input sequence
- constraints: actuator limits, state transfer, collision avoidance
- objective: minimize actuation energy

control design

- variables: control design parameters (e.g., PID gains)
- constraints: stability, overshoot, settling time
- objective: minimize tracking errors, disturbance sensitivity

examples in systems and control

system identification

- variables: dynamic model parameters
- constraints: parameter limits
- objective: minimize data misfit or prediction error

state estimation

- variables: state estimates
- constraints: dynamic model, state limits
- objective: minimize data misfit or estimation error

model predictive control

- variables: state and control input sequence
- constraints: actuator limits, state bounds
- objective: minimize state error, actuator energy

control system analysis and design

- variables: Lyapunov function and control design parameters
- constraints: stability, performance, robustness
- objective: minimize system $\mathcal{H}_2$ or $\mathcal{H}_\infty$ norm

wait a second...

wait a second...

EVERYTHING is an optimization problem, right!?

wait a second...

EVERYTHING is an optimization problem, right!?

- but unfortunately there's bad news...

wait a second...

EVERYTHING is an optimization problem, right!?

- but unfortunately there's bad news...
- the most important (meta) fact about optimization problems, often missed in standard courses:
  - **in general, optimization problems are unsolvable*!**

wait a second...

EVERYTHING is an optimization problem, right!?

- but unfortunately there's bad news...
- the most important (meta) fact about optimization problems, often missed in standard courses:
  - **in general, optimization problems are unsolvable*!**
  - *solvable: algorithm guaranteed to find (near) globally optimal solution within reasonable time for every possible instance
- very important for setting expectations and understanding the past, present, and future of optimization theory

activity

- imagine we want to solve a $n = 1$ optimization problem
- feasible set is the unit interval $\mathcal{X} = \{x \in \mathbf{R} \mid 0 \leq x \leq 1\}$
- you're allowed to evaluate the function and its derivatives (if they exist) at a sequence test points in the unit interval
- in practice, finding an *exact* solution is usually impossible, so we'll settle for an approximate solution with accuracy $\epsilon > 0$
  - find any $\bar{x} \in \mathcal{X}$ such that $f(\bar{x}) \leq f(x^\star) + \epsilon$
- when is this possible? can you come up with an algorithm? or maybe a nasty function that can't easily be optimized?

activity

- what if we restrict the function class?

activity

- what if we restrict the function class?
- e.g., would it help if f were continuous and smooth?

activity

- what if we restrict the function class?
- e.g., would it help if f were continuous and smooth?
- other ideas?

activity

- what if we restrict the function class?
- e.g., would it help if f were continuous and smooth?
- other ideas?
- let's assume the objective f is *Lipschitz continuous* on $\mathcal{X}$:

$$|f(x) - f(y)| \leq L|x - y|, \quad \forall x, y \in \mathcal{X}$$

  for some constant L (called the *Lipschitz constant*)
- ideas for a simple algorithm using only function evaluations? how many do you need to guarantee a near optimal solution?
  - what about for $n = 2$ or higher?

algorithmic performance bounds for global optimization

- turns out no zeroth order algorithm can always obtain an ϵ approximate solution with fewer than

$$\left(\left\lfloor \frac{L}{2\epsilon} \right\rfloor\right)^n \qquad \text{(lower bound)}$$

function evaluations

algorithmic performance bounds for global optimization

- turns out no zeroth order algorithm can always obtain an ϵ approximate solution with fewer than

$$\left(\left\lfloor \frac{L}{2\epsilon} \right\rfloor\right)^n \qquad \text{(lower bound)}$$

  function evaluations (and the bounds for higher order algorithms are not much better)

algorithmic performance bounds for global optimization

- turns out no zeroth order algorithm can always obtain an ϵ approximate solution with fewer than

$$\left(\left\lfloor \frac{L}{2\epsilon} \right\rfloor\right)^n \qquad \text{(lower bound)}$$

  function evaluations (and the bounds for higher order algorithms are not much better)

- consider $L = 2$, $n = 12$, $\epsilon = 0.01$ and suppose our computer can evaluate the function in 1 nanosecond

algorithmic performance bounds for global optimization

- turns out no zeroth order algorithm can always obtain an ϵ approximate solution with fewer than

$$\left(\left\lfloor \frac{L}{2\epsilon} \right\rfloor\right)^n \qquad \text{(lower bound)}$$

  function evaluations (and the bounds for higher order algorithms are not much better)

- consider $L = 2$, $n = 12$, $\epsilon = 0.01$ and suppose our computer can evaluate the function in 1 nanosecond
  - we need $\sim$ **32,000,000 years!**
  - a manifestation of the "curse of dimensionality"

solving optimization problems

- question is **not** which problems are optimization problems (answer: everything), but rather which optimization problems can we actually solve?

- general optimization problems difficult, involve compromise
  - (extremely) long computation times
  - not always finding a (global) solution

- roughly speaking[1] **convex optimization problems** are a broad class of problems we can solve efficiently and reliably
  - also, methods for solving or analyzing non-convex optimization problems often involve solving convex subproblems

[1] there is fine print, which we will discuss!

tentative course outline

- part 1: convex optimization
  - convex sets, convex functions
  - convex optimization problems: linear, quadratic, second-order cone, semidefinite programs
  - duality theory and optimality conditions
  - algorithms and software
- part 2: applications in systems and control
  - trajectory optimization
  - model predictive control
  - control design for stability and performance using linear matrix inequalities (LMIs)
  - advanced topics: sum of squares programming, robust and stochastic optimization, algorithm convergence, etc.

course goals

- recognize and formulate convex optimization problems
- learn about extensive array of applications of convex optimization in systems and control
- develop code and understand underlying algorithms for moderately-sized problems
- be exposed to underlying theory, including convex analysis, duality, algorithm convergence properties, computational complexity, etc.
- enable application to wide range of novel research problems

homework

- read Syllabus carefully
- read BV Chapter 1 and Appendix A
- homework 1 posted soon, due Monday February 1 via email

convex optimization problems

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & x \in \mathcal{X} \end{array}$$

- objective and inequality constraint functions are convex

  $$f(\alpha x + \beta y) \leq \alpha f(x) + \beta f(y)$$

  for any α and β with $\alpha + \beta = 1$, $\alpha \geq 0$, $\beta \geq 0$
- equality constraints are affine: $h_i(x) = a_i^T x + b$
- includes least squares and linear programs as special cases, but also many other interesting problems

a canonical convex optimization problem: least squares

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

solving least squares problems

- analytical solution: $x^\star = (A^T A)^{-1} A^T b$
- reliable and efficient algorithms and software
- computation time proportional to $n^2 k$ ($A \in \mathbf{R}^{k \times n}$); less if structured
- ubiquitous in applications and easy to recognize

least squares in systems and control

optimal state regulation for linear dynamical systems

$$\text{minimize} \quad \sum_{t=0}^{N} \|z_t\|_2^2 + \|u_t\|_2^2$$

where $z_{t+1} = Az_t + Bu_t$ with z_0 given, $z_t \in \mathbf{R}^n$, $u_t \in \mathbf{R}^m$

- can you transform this to a least squares problem?

another canonical convex optimization problem: linear programming

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

solving linear programs

- no analytical formula for solution
- reliable and efficient algorithms and software
- computation time proportional to $n^2 m$ if $m \geq n$; less if structured
- very common in applications, though not as easy to recognize as least squares
- a few standard tricks to convert problems into linear programs (e.g., involving ℓ_1- or ℓ_∞-norms, piecewise-linear functions)

solving convex optimization problems

- no analytical formula for solution
- reliable and efficient algorithms and software
- computation time roughly proportional to $\max\{n^3, n^2m, n^2p, F\}$, where F is cost of evaluating objective and constraint functions and their first and second derivatives
- many tricks for transforming problems into convex form
- surprisingly common in applications, but often difficult to recognize
  - without trained intuition, extremely difficult problems and easy problems can appear quite similar

convex optimization in systems and control

example: optimal state regulation for linear dynamical systems

$$\begin{array}{ll} \text{minimize} & \displaystyle\sum_{t=0}^{N} \|z_t\|_1 + \|u_t\|_\infty \\ \text{subject to} & z_{t+1} = Az_t + Bu_t \\ & u_t \in \mathcal{U} = \{u_t \in \mathbf{R}^m \mid Fu_t \leq g\} \\ & z_t \in \mathcal{Z} = \{z_t \in \mathbf{R}^n \mid Gz_t \leq h\} \end{array}$$

- no longer analytical solution, but can be easily solved with convex optimization
- using custom solvers, can solve extremely fast ($\sim \mu$s)
- repeatedly solving leads to **model predictive control**, emerging as the most widely used advanced control method in an increasingly broad array of applications

static output feedback problem

consider a linear dynamical system

$$\dot{x} = Ax + Bu$$
$$y = Cx$$

find a static output feedback controller $u = Ky$, where $K \in \mathbf{R}^{m \times p}$, that stabilizes the system, or report that none exists

equivalently, is the following optimization problem feasible?

$$\begin{array}{ll} \text{minimize} & 0 \\ \text{subject to} & \mathbf{Re}[\text{eig}(A + BKC)] \leq 0 \end{array}$$

easy or hard?

static output feedback problem

consider a linear dynamical system

$$\dot{x} = Ax + Bu$$
$$y = Cx$$

find a static output feedback controller $u = Ky$, where $K \in \mathbf{R}^{m \times p}$, that stabilizes the system, or report that none exists

equivalently, is the following optimization problem feasible?

$$\begin{array}{ll} \text{minimize} & 0 \\ \text{subject to} & \mathbf{Re}[\text{eig}(A + BKC)] \leq 0 \end{array}$$

easy or hard?

- no one knows!
- lots of research (still) goes into distinguishing "tractable" problems from "intractable" ones