Introduction to Convex Optimization

Slides adapted from presentations by:
Matt Deyo
Enrique Fernandez

Most figures and equations from “Convex Optimization” by S. Boyd and L. Vandenberghe (http://stanford.edu/~boyd/cvxbook/)

Nikhil Bhargava
16.410/413
December 5th, 2018

Assignments

- Assignment:
  - Due Today: Assignment #11
- Reading:
  - Today: Convex Programming
    - Optional: [AIMA] Chapter 14; re-read Chapter 13.
  - Next: Risk-Bounded Planning
    - See mers.csail.mit.edu/publications

Today:

- General Optimization Problems
- Finding a solution
  - Local vs. Global Optima [considering **convexity**]
  - Handling Nonlinear Constraints
- Path Planning with Obstacles and Dynamics using Convex Optimization

Today:

- **General Optimization Problems**
- Finding a solution
  - Local vs. Global Optima [considering **convexity**]
  - Handling Nonlinear Constraints
- Path Planning with Obstacles and Dynamics using Convex Optimization

Mathematical Programming (Optimization)

- Widely used across different fields
  - Operations Research, Logistics, etc
  - Design
  - Machine Learning
  - Robotics: trajectory optimization, etc.

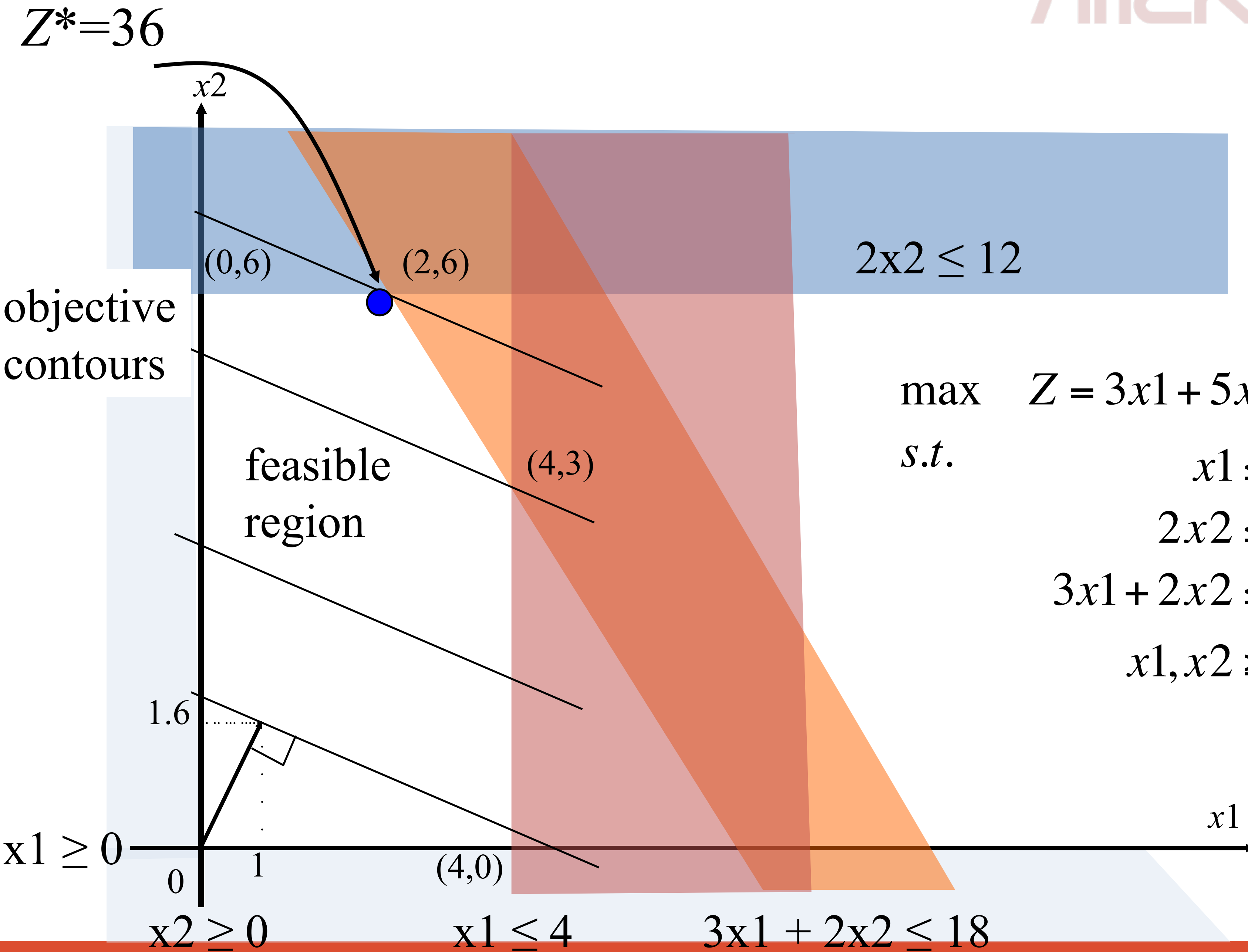
$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq b_i, \quad i = 1, \ldots, m. \end{array}$$

Resource Allocation

An aircraft company makes two aircraft engines.

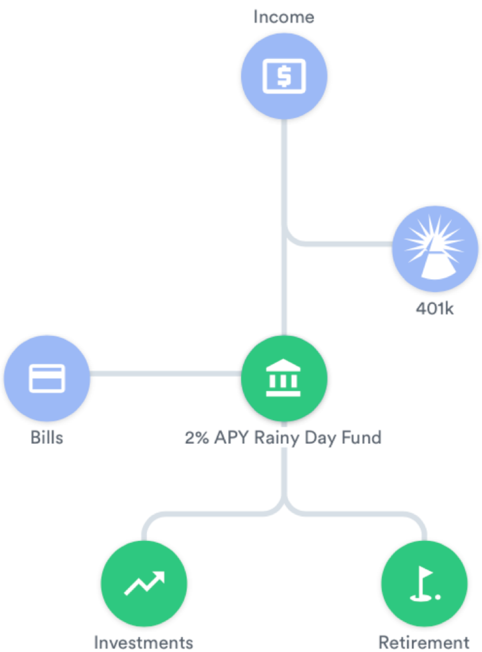
Problem: Decide how to allocate production of Engine's 1 and 2 to maximize profit.

- Engine parts are made at three plants: A, B, and C.
  - Engine 1 uses parts produced by A and C.
  - Engine 2 uses parts produced by B and C.
- Plants A, B, C have limited throughput.



$$
\begin{aligned}
\max \quad & Z = 3x1 + 5x2 \\
s.t. \quad & x1 \leq 4 \\
& 2x2 \leq 12 \\
& 3x1 + 2x2 \leq 18 \\
& x1, x2 \geq 0
\end{aligned}
$$

Saving for Retirement



haven

courtesy of havenmoney.com

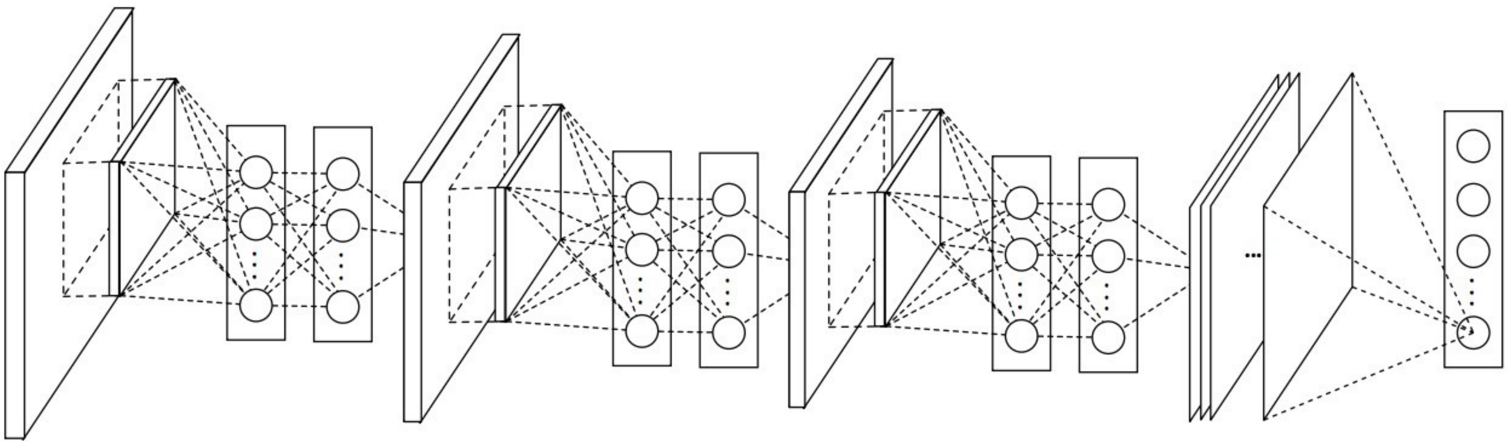
- When should you retire? Assuming:
  - You save 10% of your salary per year (starting at \$50k)
  - You get a 2.5% raise per year
  - Assume 6% annual ROI on your investments
  - You can live on \$40k annually in retirement
  - You don't want to run out of money before you die (at 90)

$$\min a$$

$$50{,}000\,\frac{\left((1.06\cdot 1.025)^{a+1-22}-1\right)}{(1.06\cdot 1.025-1)}=s$$

$$40{,}000(90-a)\leq s$$

State of the Art Technology



$$= N(\cdot,\cdot)$$

Images courtesy of the Not Hotdog app (https://itunes.apple.com/us/app/not-hotdog/id1212457521) and TowardsDataScience (https://towardsdatascience.com/neural-network-architectures-156e5bad51ba)

With training data and labels x, b:

$$\min ||N(p,x) - b||_2$$

Today:

- General Optimization Problems
- **Finding a solution**
  - Local vs. Global Optima [considering **convexity**]
  - Handling Nonlinear Constraints
- Path Planning with Obstacles and Dynamics using Convex Optimization

General Approaches

1. Solve Directly

2. Search Exhaustively

3. Sample the Search Space

General Approaches

1. ~~Solve Directly~~

don't know how???

2. Search Exhaustively

3. Sample the Search Space

General Approaches

1. ~~Solve Directly~~ don’t know how???

2. ~~Search Exhaustively~~ search space too big :(

3. Sample the Search Space

General Approaches

1. ~~Solve Directly~~ — don't know how???

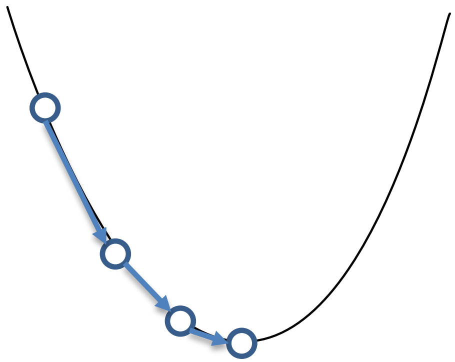
2. ~~Search Exhaustively~~ — search space too big :(

3. **Sample the Search Space**

Gradient Descent

$$\begin{array}{ll} \underset{\mathbf{x}}{\text{minimize}} & f(\mathbf{x}) \\ \text{subject to} & \mathbf{x} \in \mathbb{R}^n. \end{array}$$

Natural idea is to search in direction of negative gradient

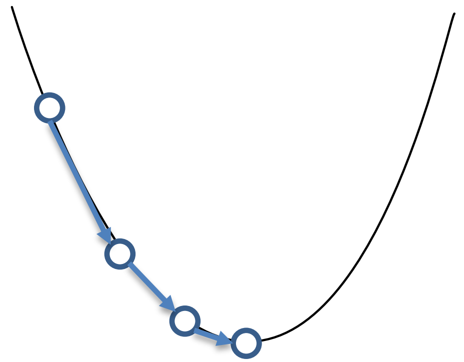


1. Choose arbitrary starting point x
2. Compute direction of steepest descent: $-\nabla f(\mathbf{x})$
3. Take a small step in this direction $\mathbf{x}_{new} = \mathbf{x} - \sigma \nabla f(\mathbf{x})$
4. Go to 2

Gradient Descent

$$\begin{aligned} \underset{\mathbf{x}}{\text{minimize}} \quad & f(\mathbf{x}) \\ \text{subject to} \quad & \mathbf{x} \in \mathbb{R}^n. \end{aligned}$$

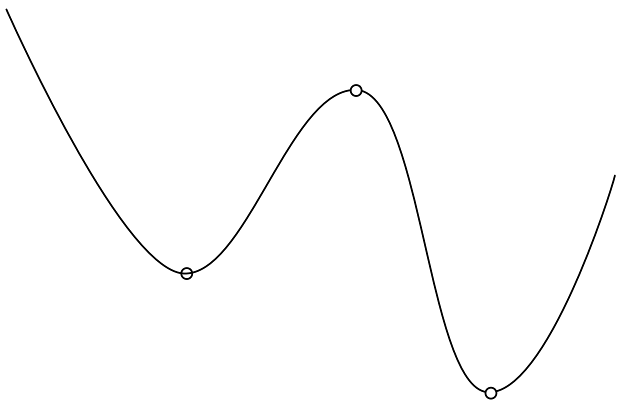
Natural idea is to search in direction of negative gradient



1. Choose arbitrary starting point x
2. Compute direction of steepest descent: $-\nabla f(\mathbf{x})$
3. Take a small step in this direction $\mathbf{x}_{new} = \mathbf{x} - \sigma \nabla f(\mathbf{x})$
4. Go to 2

Questions:

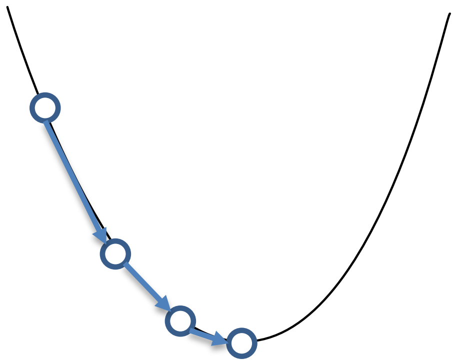
1. When to stop?
2. How big should steps be?
3. Will it find the minimum?



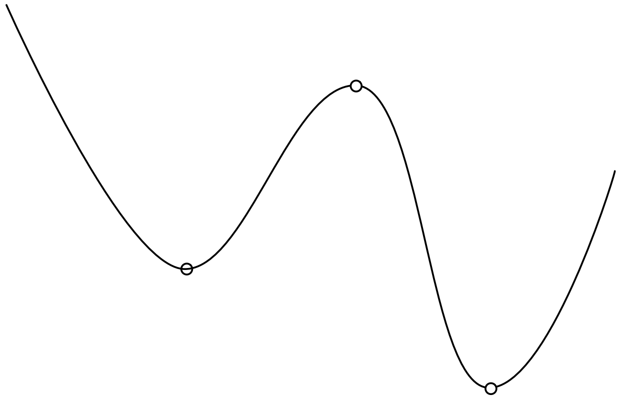
Gradient Descent

$$\begin{array}{ll} \underset{\mathbf{x}}{\text{minimize}} & f(\mathbf{x}) \\ \text{subject to} & \mathbf{x} \in \mathbb{R}^n. \end{array}$$

Natural idea is to search in direction of negative gradient



1. Choose arbitrary starting point x
2. Compute direction of steepest descent: $-\nabla f(\mathbf{x})$
3. Take a small step in this direction $\mathbf{x}_{new} = \mathbf{x} - \sigma \nabla f(\mathbf{x})$
4. Go to 2



Questions:

1. When to stop?
2. How big should steps be?
3. Will it find the minimum?
4. …Globally?
   In general? No
   For Convex Problems? Yes! Gradient Descent will find global min

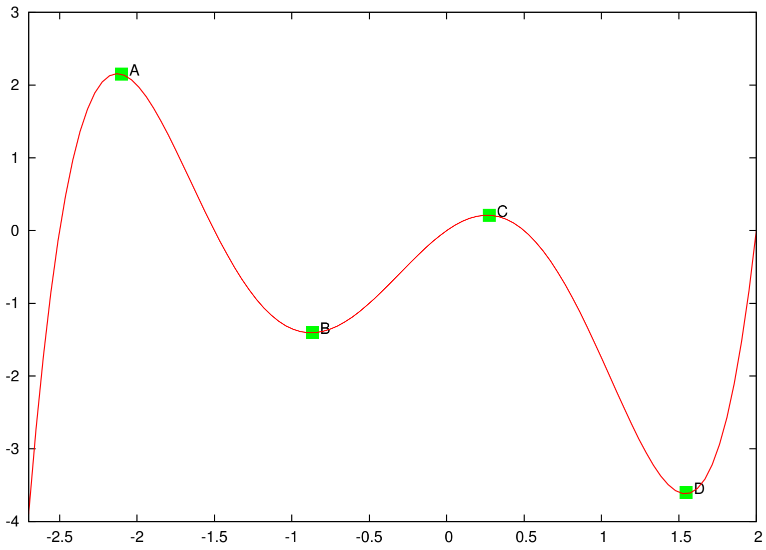
Today:

- General Optimization Problems
- Finding a solution
  - **Local vs. Global Optima [considering convexity]**
  - Handling Nonlinear Constraints
- Path Planning with Obstacles and Dynamics using Convex Optimization

Local vs Global Minima

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

Find minimum x^*



- Global minimum

$$f(x^*) \leq f(x) \quad \forall x \in X$$

- Local minimum

$$\exists \epsilon > 0$$
$$f(\mathbf{x}^*) \leq f(\mathbf{x}) \quad \forall \mathbf{x}, \quad \|\mathbf{x}^* - \mathbf{x}\| \leq \epsilon$$

Ideally we'd like to find the global minimum

How do we find the minimum?

Optimality conditions

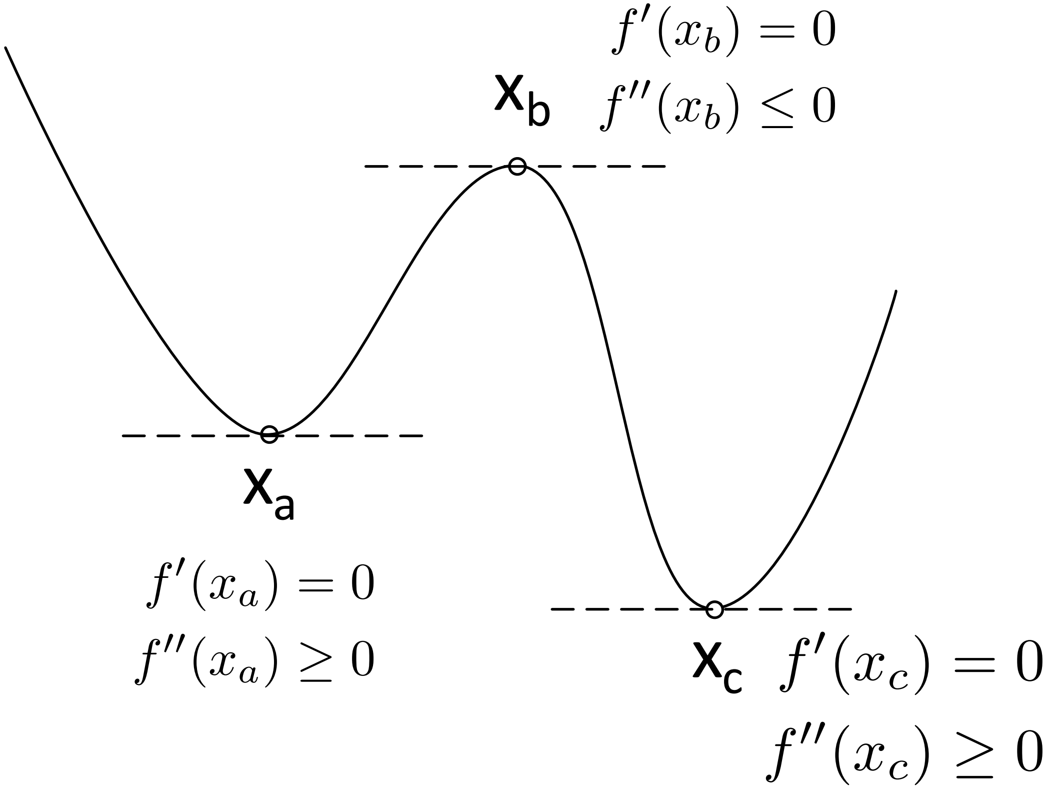
(for unconstrained optimization)

Necessary conditions for **local** minimum

$$f : \mathbb{R} \to \mathbb{R}$$

$$f'(x) = 0$$

$$f''(x) \geq 0$$



Optimality conditions

(for unconstrained optimization)

Necessary conditions for **local** minimum

$$f : \mathbb{R} \to \mathbb{R} \qquad f : \mathbb{R}^n \to \mathbb{R}$$

First order necessary condition

$$f'(x) = 0 \qquad \nabla f(\mathbf{x}) = 0$$

Second order necessary condition

$$f''(x) \geq 0 \qquad H_f(\mathbf{x}^*) \succeq 0$$

$$H_f(\mathbf{x}^x) \in \mathbb{R}^{n \times n}$$

is positive semidefinite (PSD)

Hessian matrix

$$H_{f\,i,j} = \frac{\partial^2 f}{\partial x_i \partial x_j}$$

Non-linear **unconstrained** optimization

$$\begin{aligned} \underset{\mathbf{x}}{\text{minimize}} \quad & f(\mathbf{x}) \\ \text{subject to} \quad & \mathbf{x} \in \mathbb{R}^n. \end{aligned}$$

First order necessary condition

$$\nabla f(\mathbf{x}) = 0$$

Second order necessary condition

$$H_f(\mathbf{x}^*) \succeq 0$$

How to find **x***?

1. Find critical points (first order condition)
2. Check candidates

This can be hard in practice

With a convex function, all local minima are *global* minima.

Convex optimization

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

with f_0, f_i convex

- Convex objective function
- Convex feasible set

Convex optimization

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

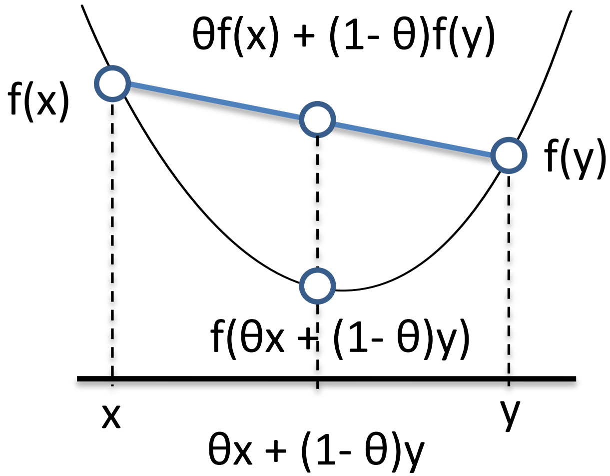
with f_0, f_i convex

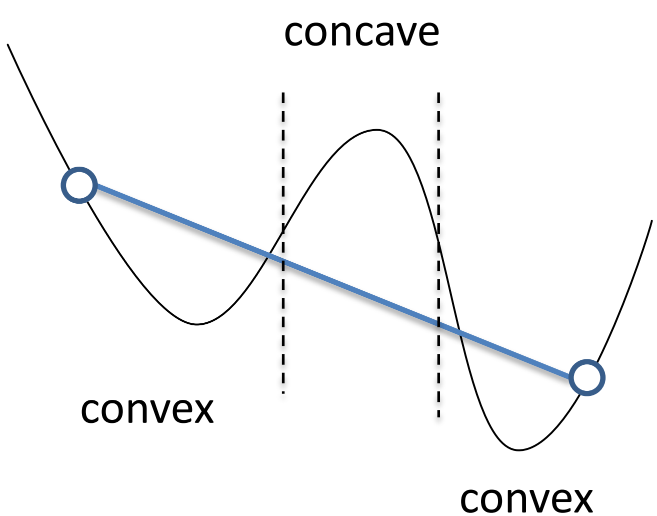
- **Convex objective function**
- Convex feasible set

Convex functions

f is convex if and only if: $$f(\theta x + (1-\theta)y) \le \theta f(x) + (1-\theta) f(y)$$

∀x, y and ∀ θ such that $0 \le \theta \le 1,$





A function f is **concave** if –f is convex

Convex functions

Checking convexity

Method 1: Definition

$$f(\theta x + (1-\theta)y) \leq \theta f(x) + (1-\theta) f(y)$$

∀x, y and ∀ θ such that $0 \leq \theta \leq 1,$

Method 2:

A function f with continuous second-order partial derivatives is convex if and only if its **Hessian** is **positive semidefinite** at **all points** in R^n

$$H_f(\mathbf{x}) \succeq 0,\ \forall \mathbf{x} \in \mathbb{R}^n$$

Hessian Matrix: $H_f \in \mathbb{R}^{n \times n}$

$$H_{f\,i,j} = \frac{\partial^2 f}{\partial x_i \partial x_j}$$

Convex functions

Example convex functions

- Affine functions are both convex and concave $f(\mathbf{x}) = A\mathbf{x} + \mathbf{b}$
- Exponential $f(x) = e^{ax} \quad \forall a \in \mathbb{R}$
- Powers $f(x) = x^a, \quad x \in \mathbb{R}_{++}, \forall a \in (-\infty, 0] \cup [1, \infty)$
- $f(x) = -\log x$
- Quadratic functions $f(\mathbf{x}) = (1/2)\mathbf{x}^T P \mathbf{x} + \mathbf{q}^T \mathbf{x} + r$ if and only if $P \succeq 0$
- Norms (every norm in R^n)

(some) **Operations that preserve convexity**

- Non-negative weighted sums of convex functions $f = w_1 f_1 + \cdots + w_m f_m$
- Maximum of convex functions
- Composition with affine function
- Composition with convex non-decreasing function

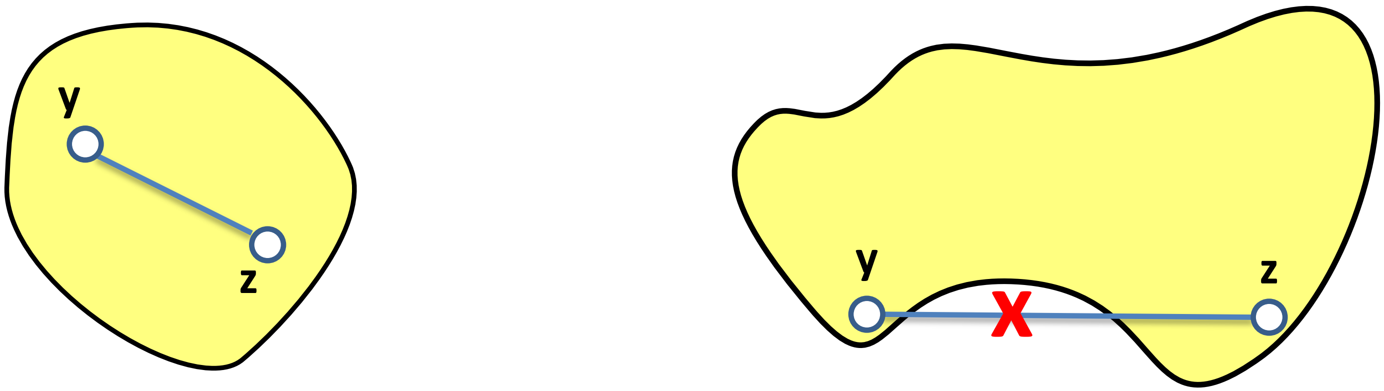
Convex optimization

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

with f_0, f_i convex

- Convex objective function
- **Convex feasible set**

Convex sets



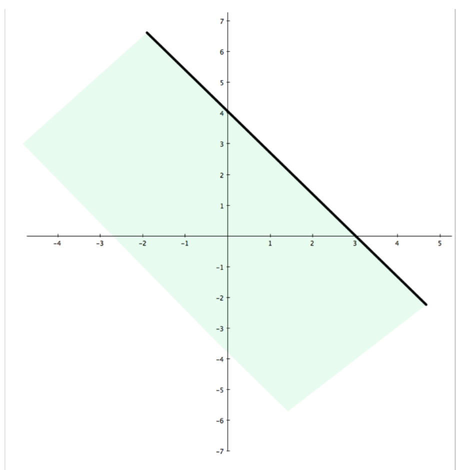
$S \subseteq \mathbb{R}^n$ is convex iff $\lambda \mathbf{y} + (1-\lambda)\mathbf{z} \in S, \quad \forall \mathbf{y}, \mathbf{z} \in S, \, \lambda \in [0,1]$

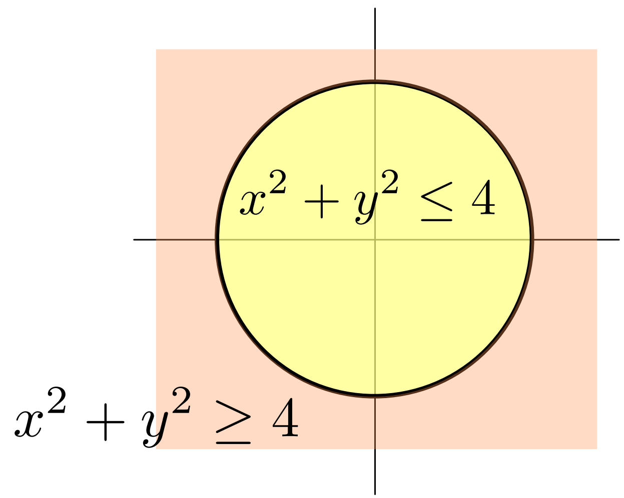
Sublevel sets of convex functions

Property: Sublevel sets of convex functions are convex

if f(x) is convex, then $S = \{\mathbf{x} \in \mathbb{R}^n : f(\mathbf{x}) \leq b\}$ is convex

$$4x + 3y - 12 \leq 0$$

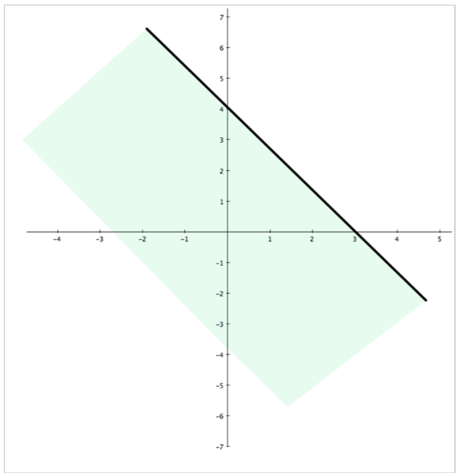




Operations that preserve convexity

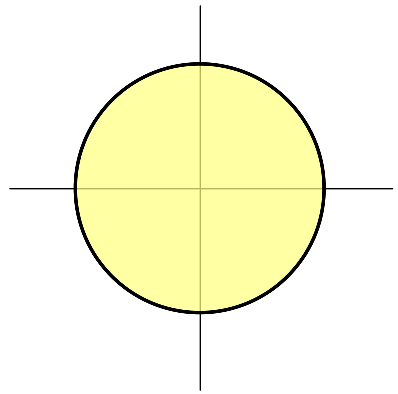
Intersection? Yes. Intersection of convex sets is convex

$$4x + 3y - 12 \leq 0$$



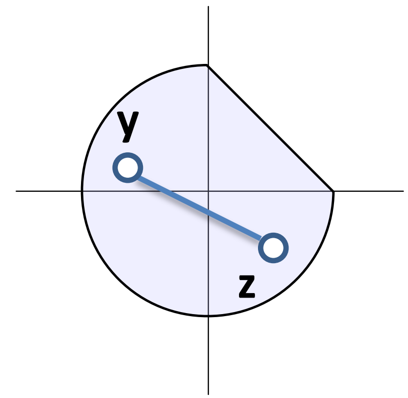
$\cap$

$$x^2 + y^2 \leq 16$$

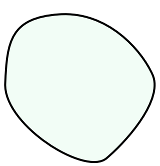


=

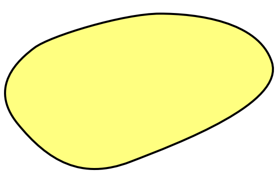
$$4x + 3y - 12 \leq 0$$
$$x^2 + y^2 \leq 16$$



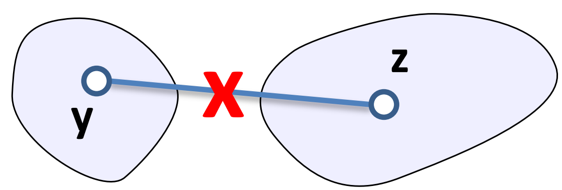
Union? No. Union of convex sets is not necessarily convex



$\cup$



=



Affine transformations Rotation, scaling, translation? Preserve convexity

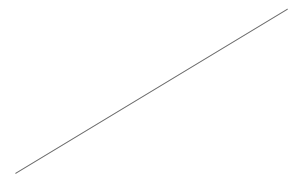
Examples of convex sets

$\varnothing$

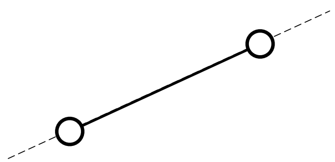
The empty set

$\mathbb{R}^n$

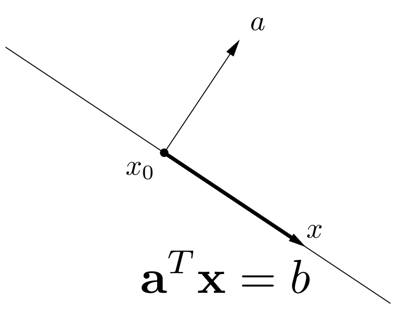
The whole space



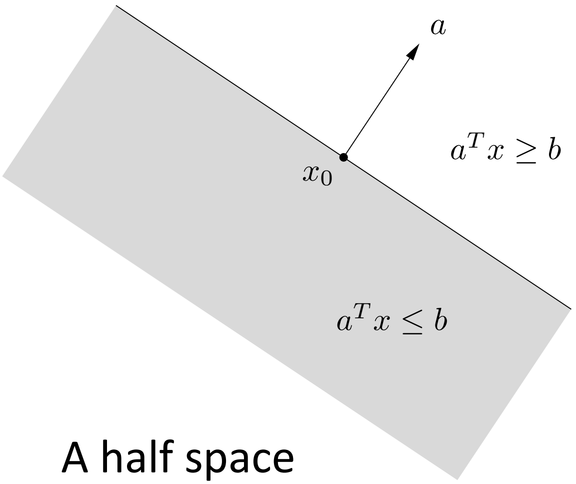
A line



A line segment



A hyper plane



A half space

Convex optimization

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

with f_0, f_i convex

convex objective f_0

convex feasible set
(f_i convex)

Properties of convex optimization

- Local minima? Any locally optimal point is also globally optimal
- No analytical formula to solve convex optimization problems, but very effective methods for solving them
  - Interior point methods in polynomial time
  - Can easily solve problems with hundreds of variables and thousands of constraints on a current desktop computer in a few tens of seconds

Today:

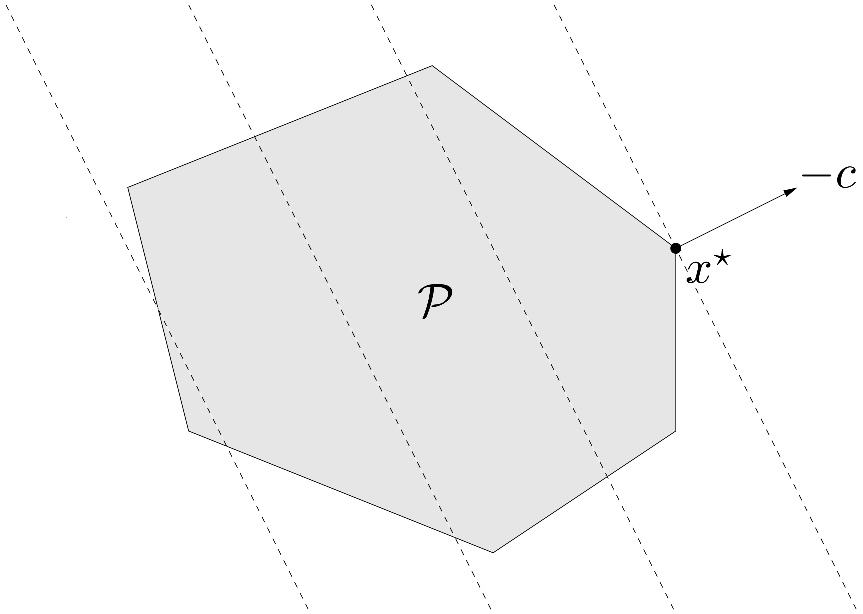
- General Optimization Problems
- Finding a solution
  - Local vs. Global Optima [considering **convexity**]
  - **Handling Nonlinear Constraints**
- Path Planning with Obstacles and Dynamics using Convex Optimization

Linear Programs (LP)

$$\begin{array}{ll} \text{minimize} & c^T x + d \\ \text{subject to} & Gx \preceq h \\ & Ax = b, \end{array}$$

Affine objective (convex)

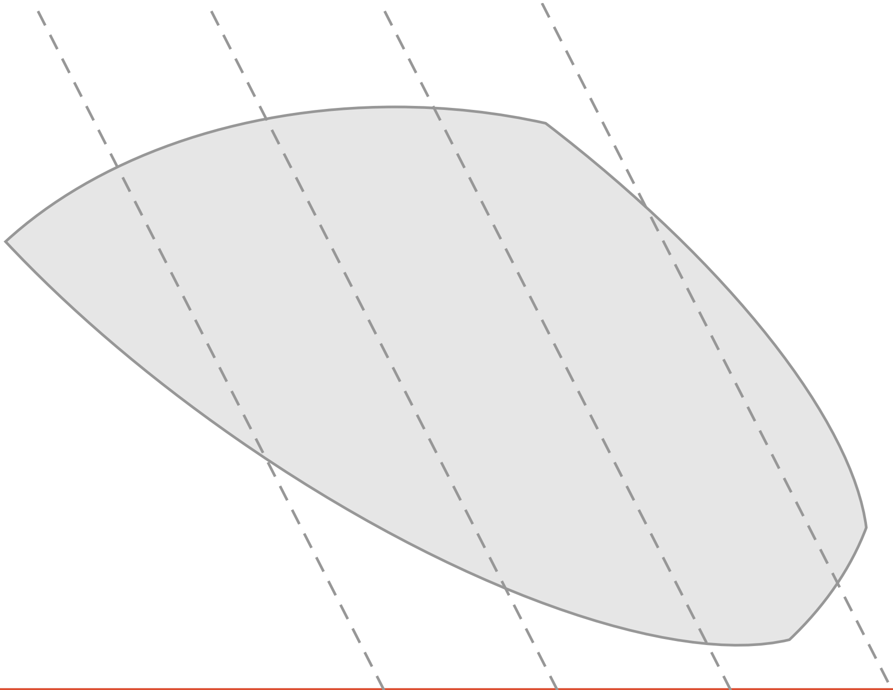
Intersection of half-spaces and hyperplanes



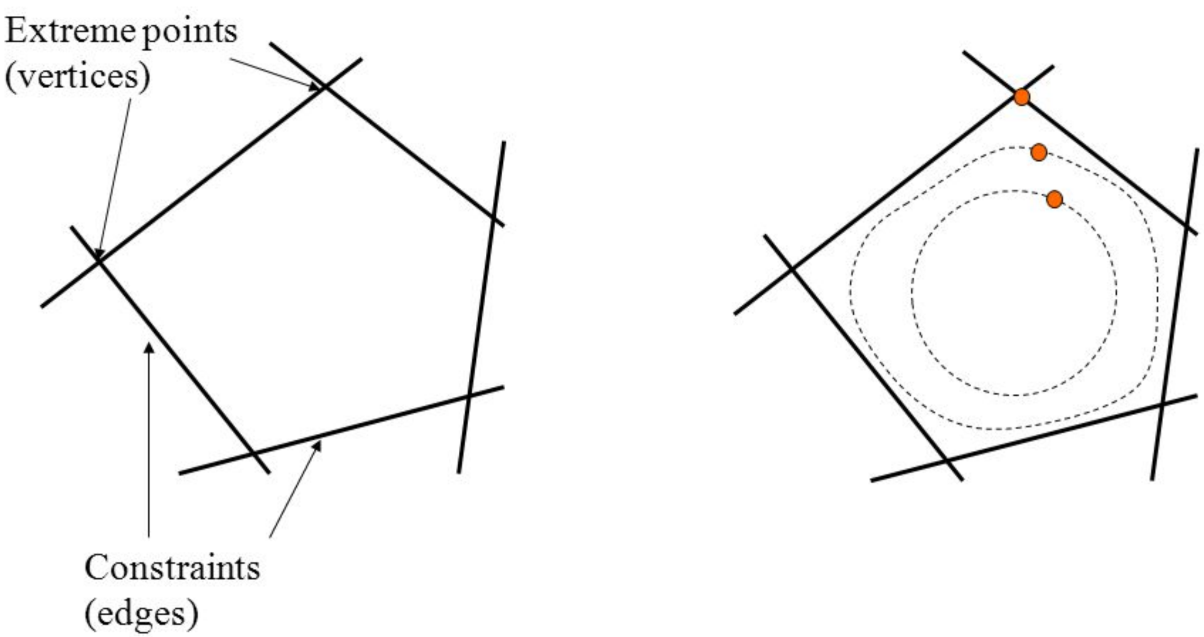
Nonlinear Programs

$$
\begin{array}{ll}
\text{minimize} & c^T x + d \\
\text{subject to} & Gx \preceq h \\
& Ax = b,
\end{array}
$$

Solution not guaranteed to lie at a "corner", even with linear objectives.



Interior Point Methods



Simplex: search from vertex to vertex along the edges

Interior-point methods: go through the inside of the feasible space

Interior Point Methods

Also known as Barrier Methods

Based on Karmarkar's algorithm - Developed for solving linear programs in linear time

More powerful and faster in practice than simplex

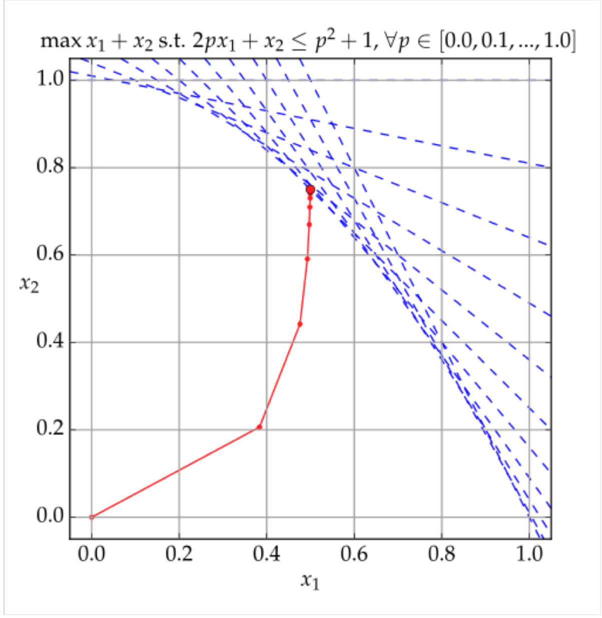
Primal-dual interior-point outline:

1. Replace constraints with functions in the objective Ex: Logarithmic barrier functions for inequalities

$$B(x, \mu) = f(x) - \mu \sum_{i=1}^{m} \log(c_i(x))$$

2. Compute primal-dual search direction using Newton's method

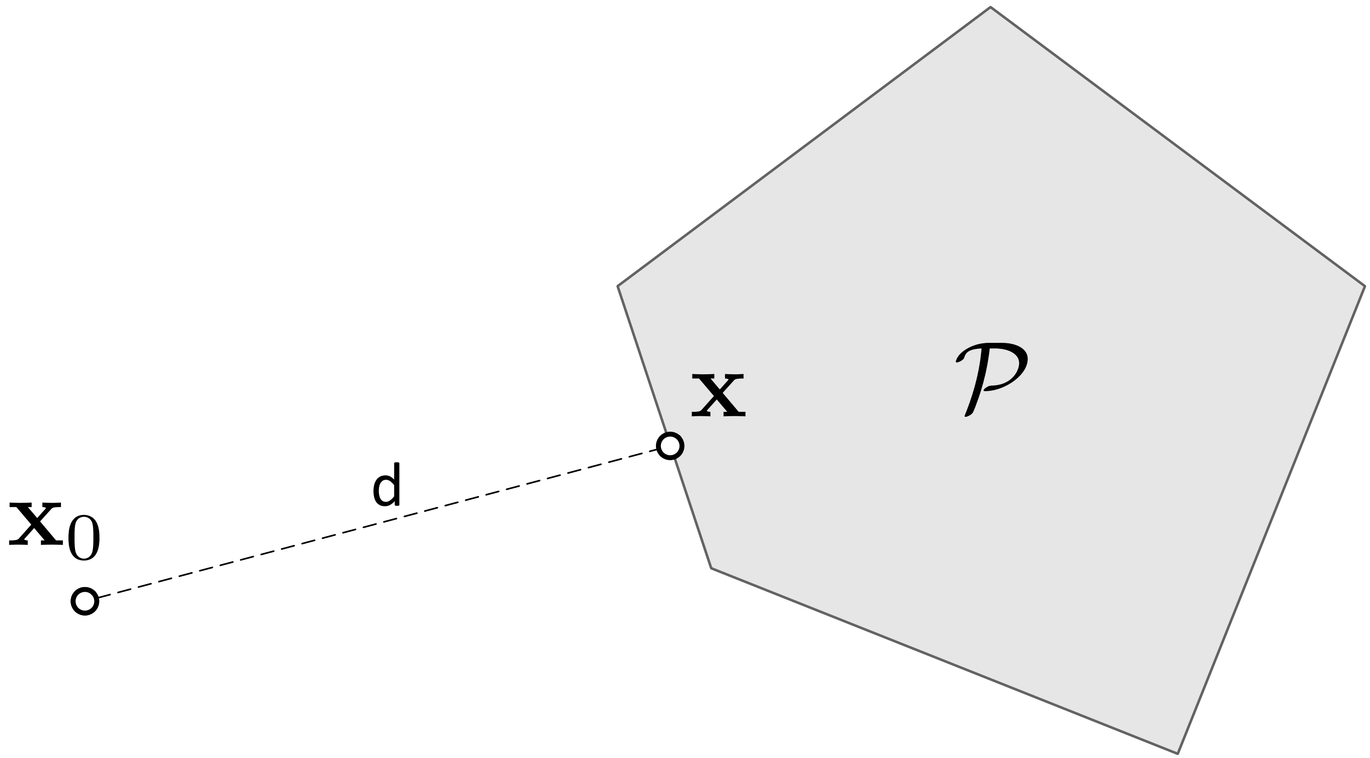
3. Iterate search and update until surrogate duality gap converges



Today:

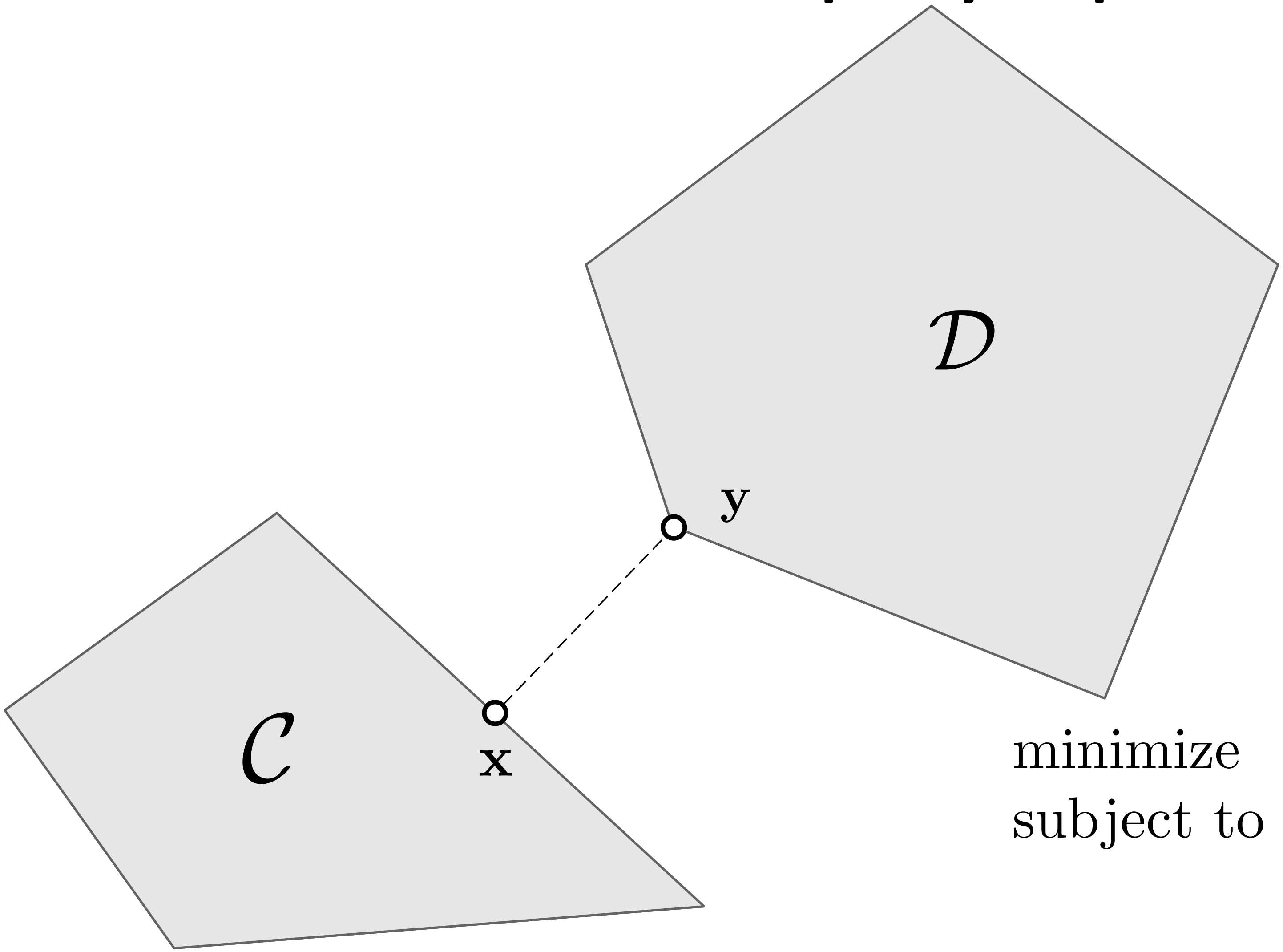
- General Optimization Problems
- Finding a solution
  - Local vs. Global Optima [considering **convexity**]
  - Interior Point Method
- **Path Planning with Obstacles and Dynamics using Convex Optimization**

Distance between point and polytopes



$$
\begin{array}{ll}
\text{minimize} & \|x - x_0\|_2^2 \\
\text{subject to} & Ax \preceq b.
\end{array}
$$

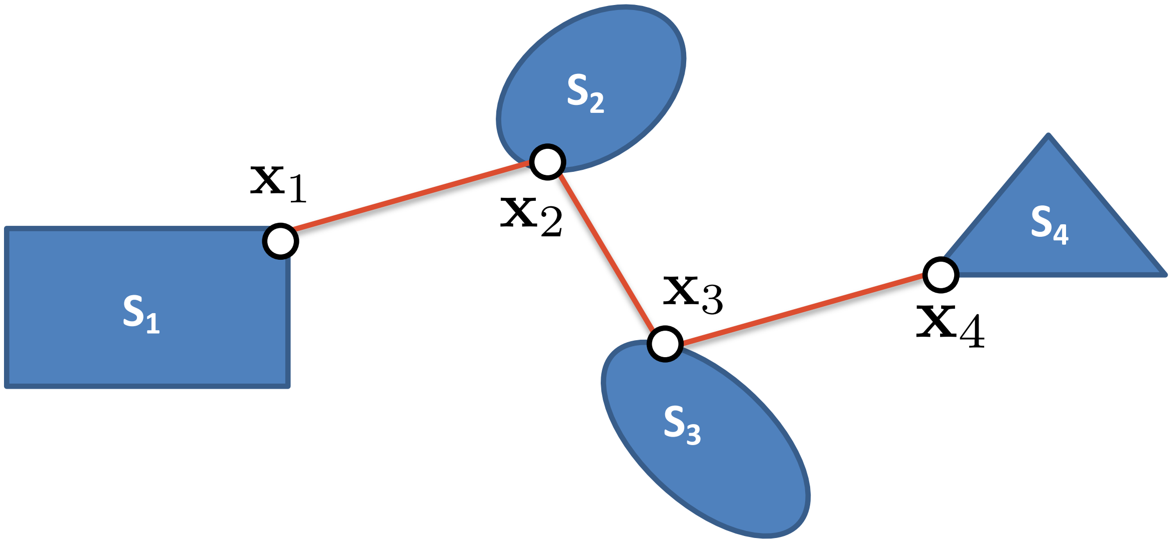
Distance between polytopes



$$
\begin{array}{ll}
\text{minimize} & \|x - y\|_2 \\
\text{subject to} & A_1 x \preceq b_1 \\
& A_2 y \preceq b_2
\end{array}
$$

Shortest path visiting regions

Shortest path going through S_1, S_2, S_3 and S_4



$$\begin{array}{ll} \underset{\mathbf{x}}{\text{minimize}} & \|\mathbf{x}_2 - \mathbf{x}_1\| + \|\mathbf{x}_3 - \mathbf{x}_2\| + \|\mathbf{x}_4 - \mathbf{x}_3\| \\ \text{subject to} & \mathbf{x}_1 \in \mathcal{S}_1,\ \mathbf{x}_2 \in \mathcal{S}_2,\ \mathbf{x}_3 \in \mathcal{S}_3,\ \mathbf{x}_4 \in \mathcal{S}_4 \end{array}$$

Encoding Dynamics and Actuation Constraints

- 2-D Omni-dimensional Holonomic Vehicle in a room

Dynamics

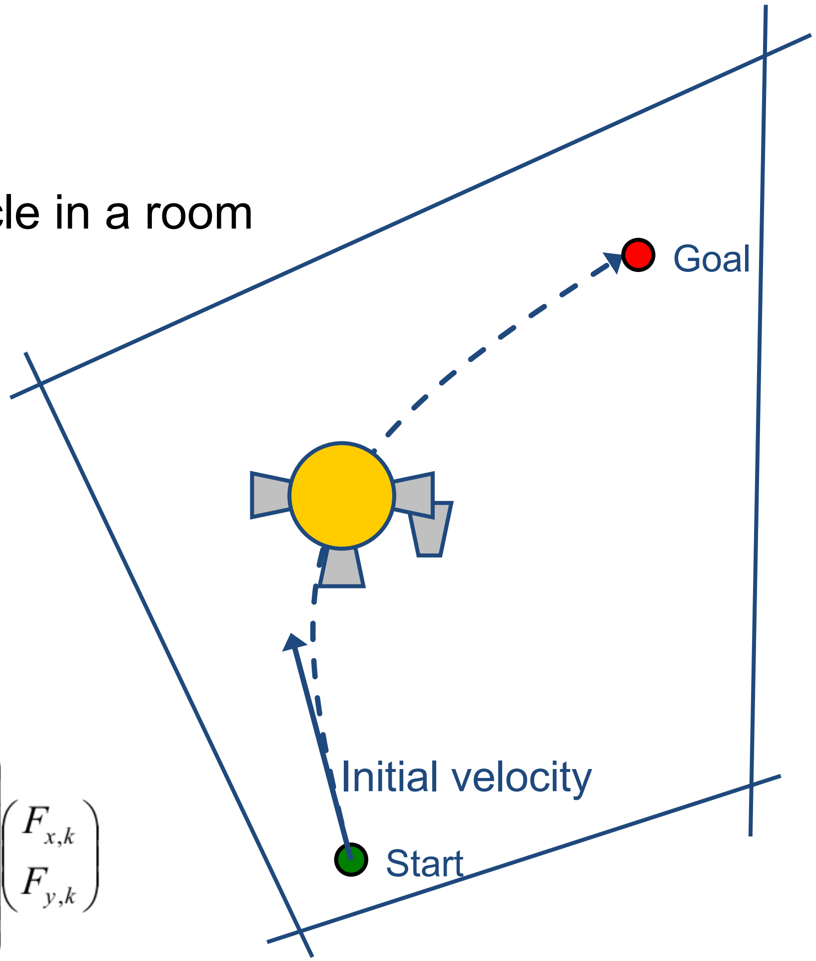
$$m\begin{pmatrix}\ddot{x}\\ \ddot{y}\end{pmatrix}=\begin{pmatrix}F_x\\ F_y\end{pmatrix}$$

$$|F_x|\le F_{\max},|F_y|\le F_{\max}$$ (Thrust limits)

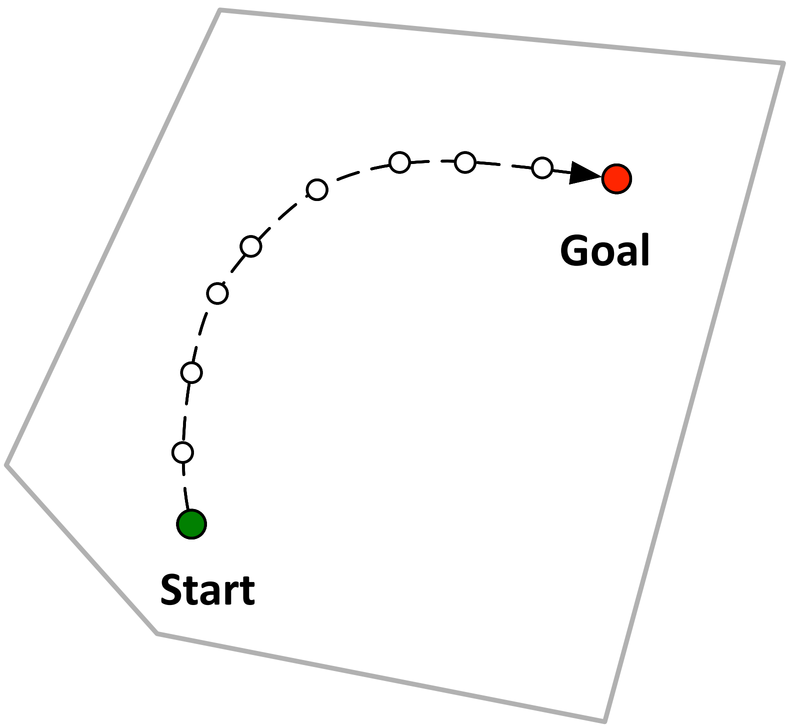
Discrete-time dynamics*

$$\begin{pmatrix}x_{k+1}\\ y_{k+1}\\ \dot{x}_{k+1}\\ \dot{y}_{k+1}\end{pmatrix}=\begin{pmatrix}1&0&\Delta t&0\\ 0&1&0&\Delta t\\ 0&0&1&0\\ 0&0&0&1\end{pmatrix}\begin{pmatrix}x_k\\ y_k\\ \dot{x}_k\\ \dot{y}_k\end{pmatrix}+\begin{pmatrix}0.5\Delta t^2/m&0\\ 0&0.5\Delta t^2/m\\ \Delta t/m&0\\ 0&\Delta t/m\end{pmatrix}\begin{pmatrix}F_{x,k}\\ F_{y,k}\end{pmatrix}$$

$$\boxed{\boldsymbol{x}_{t+1} = \boldsymbol{A}\boldsymbol{x}_t + \boldsymbol{B}\boldsymbol{u}_t}$$



Optimal Control (QP)



Variables

States $\mathbf{x}_k$

Actuation $\mathbf{u}_k$

$$N = \frac{T}{\Delta t}$$

pick fixed arrival time (FAT)

Constraints

Initial and final conditions

$$\mathbf{x}_0 = \mathbf{x}_{\text{start}}$$

$$\mathbf{x}_N = \mathbf{x}_{\text{goal}}$$

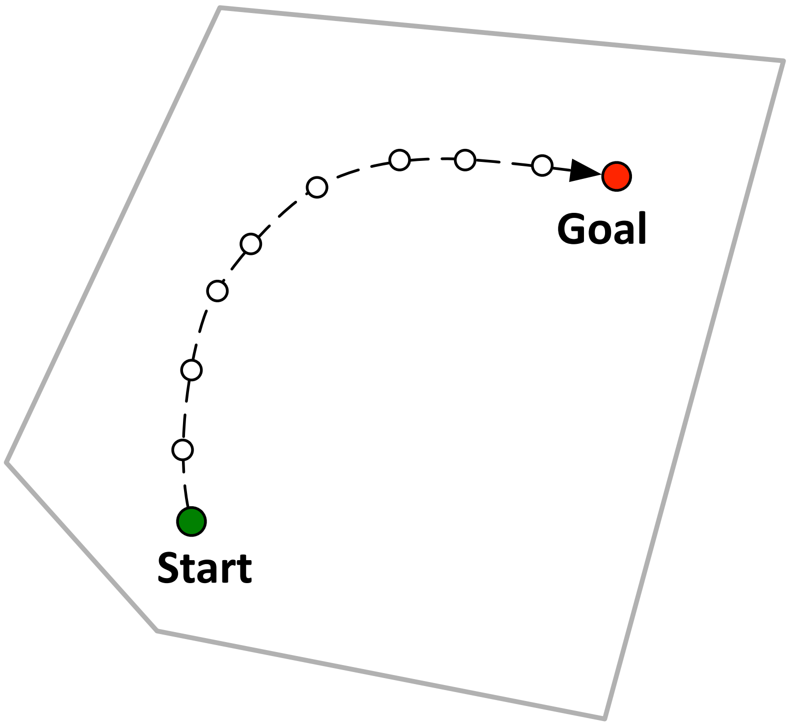
Linear Dynamics

$$\mathbf{x}_{k+1} = A\mathbf{x}_k + B\mathbf{u}_k$$

Stay within region

$$G\mathbf{x}_k \preceq \mathbf{b}$$

Optimal Control (QP)



Constraints

$$\mathbf{x}_0 = \mathbf{x}_{\text{start}}$$

$$\mathbf{x}_N = \mathbf{x}_{\text{goal}}$$

$$\mathbf{x}_{k+1} = A\mathbf{x}_k + B\mathbf{u}_k$$

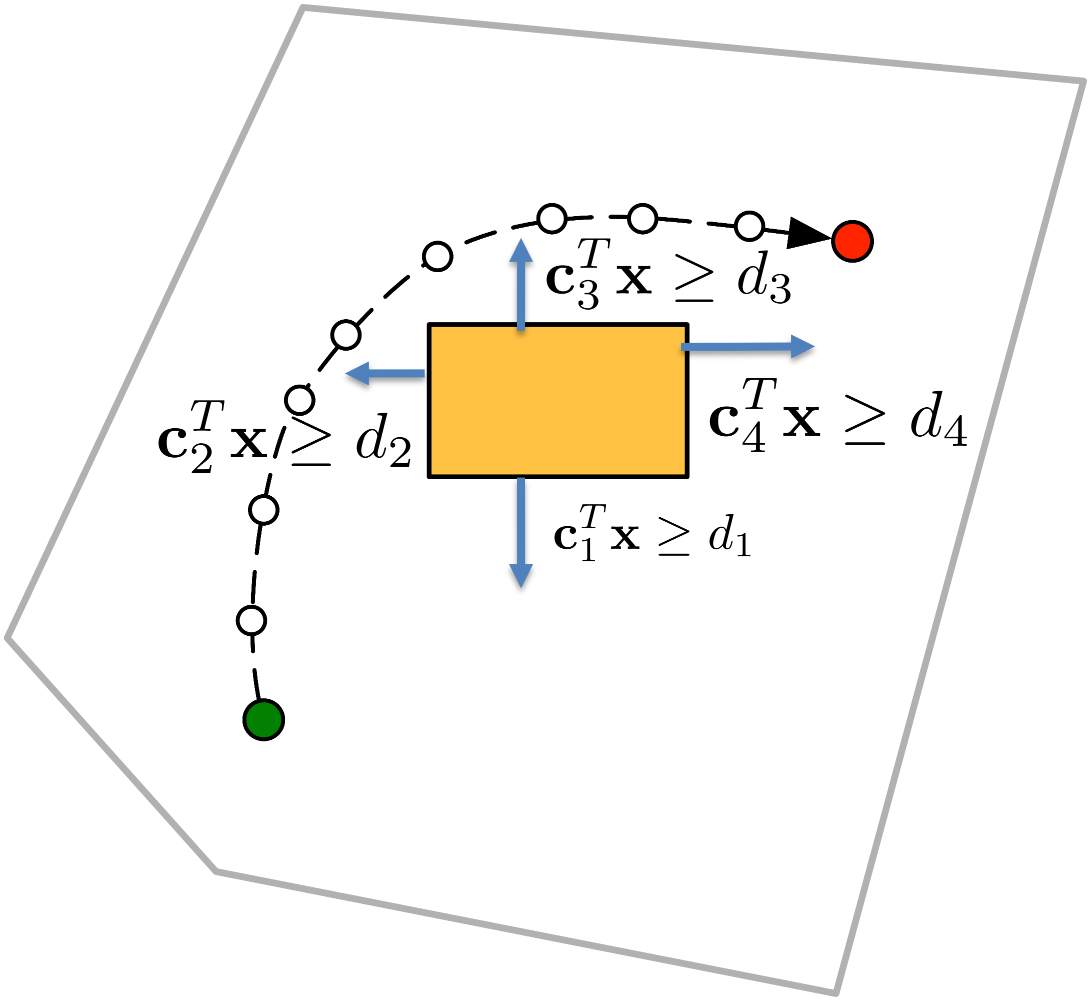
$$G\mathbf{x}_k \preceq \mathbf{b}$$

Objective

$$\text{minimize} \sum_{k=0}^{N} \|u_k\|^2$$

Linear constraints and convex quadratic objective → QP

Optimal Control

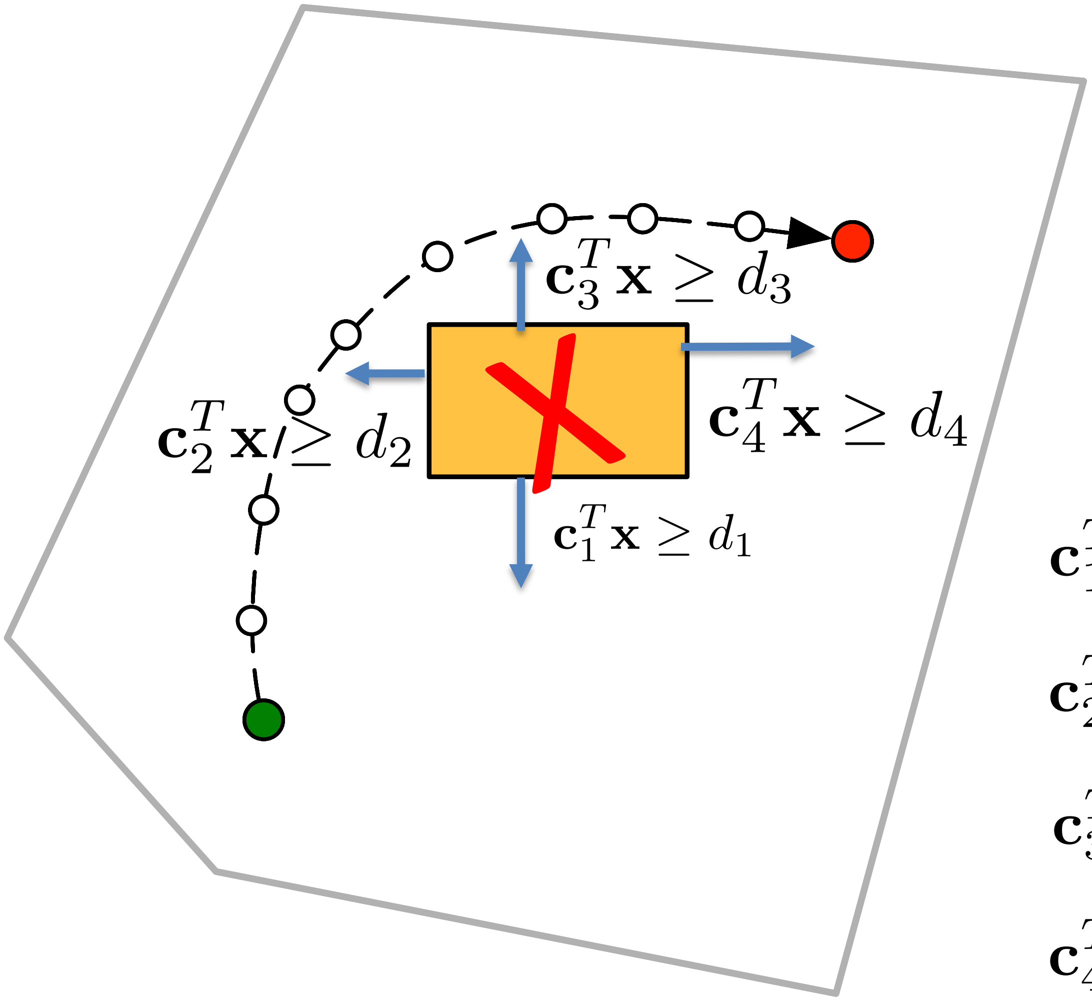


Stay inside gray region

$$G\mathbf{x}_k \preceq \mathbf{b}$$

Can we avoid obstacles?

Optimal Control



Stay inside gray region

$$G\mathbf{x}_k \preceq \mathbf{b}$$

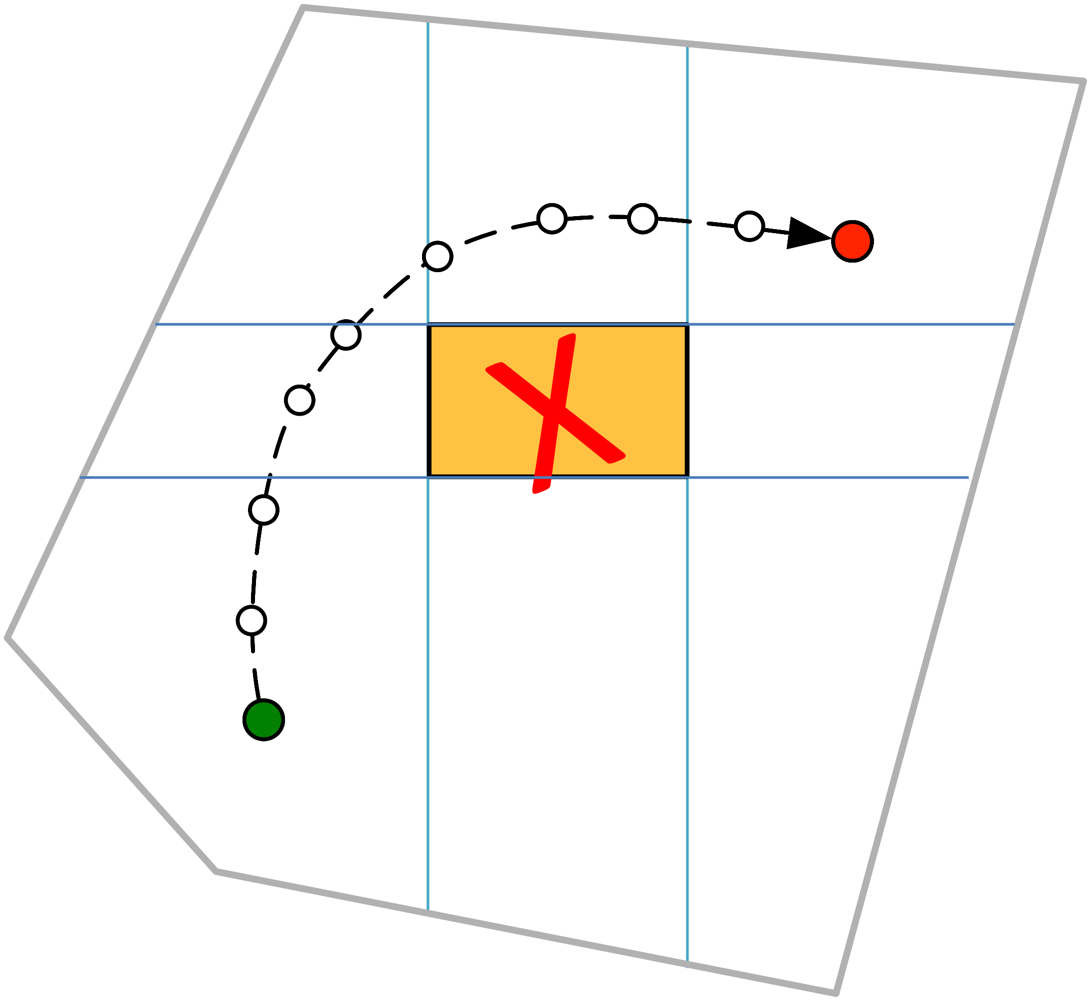
Can we avoid obstacles?

$\mathbf{c}_1^T\mathbf{x} \geq d_1$
or
$\mathbf{c}_2^T\mathbf{x} \geq d_2$
or
$\mathbf{c}_3^T\mathbf{x} \geq d_3$
or
$\mathbf{c}_4^T\mathbf{x} \geq d_4$

Need to satisfy at least one of these

Feasible set is **not convex!**
Can't use convex optimization

Optimal Control



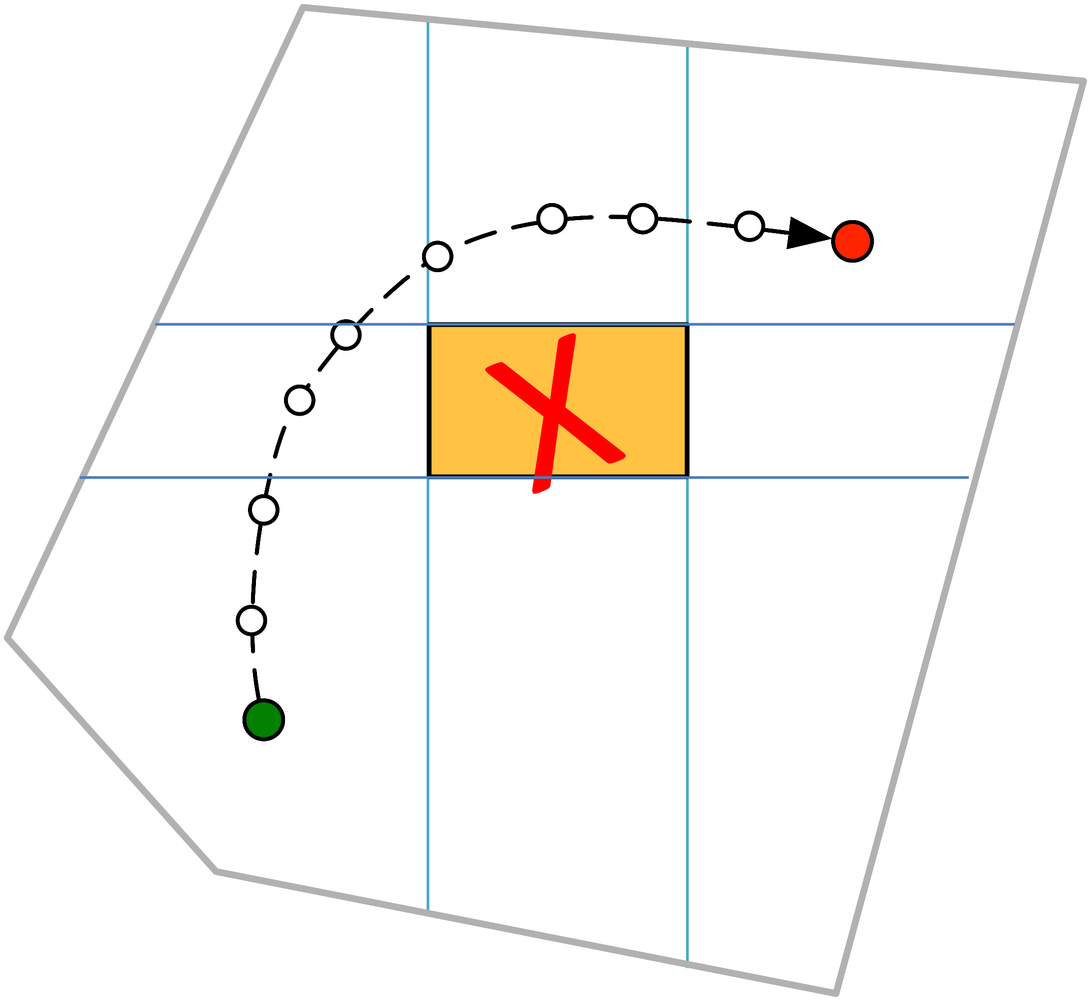
Given an obstacle, we can discretize our space around that obstacle, e.g.

$$G\mathbf{x}_k \preceq \mathbf{b}$$

$$\mathbf{c}_1^T \mathbf{x} \geq d_1$$

$$\mathbf{c}_2^T \mathbf{x} \geq d_2$$

Optimal Control



Given an obstacle, we can discretize our space around that obstacle, e.g.

$$G\mathbf{x}_k \preceq \mathbf{b}$$
$$\mathbf{c}_1^T\mathbf{x} \geq d_1$$
$$\mathbf{c}_2^T\mathbf{x} \geq d_2$$

We can use conventional search to pick our macro-level path.

Fernández-González, Enrique, Erez Karpas, and Brian Charles Williams. "Mixed Discrete-Continuous Planning with Convex Optimization." *AAAI*. 2017.

Today:

- General Optimization Problems
- Finding a solution
  - Local vs. Global Optima [considering **convexity**]
  - Handling Nonlinear Constraints
- Path Planning with Obstacles and Dynamics using Convex Optimization

Reference solvers and frameworks

Frameworks

- yalmip, cvx (MATLAB)
- cvxpy, picos, cvxopt (Python)
- Convex.jl (Julia)

Solvers

- LPSolve (LP) – Opensource
- Gurobi (LP, SOCP, MISOCP) – Commercial, free for Academia
- CPLEX (LP, SOCP, MISOCP) – commercial
- ECOS (LP, SOCP, MISOCP) – OpenSource
- SeDuMi (LP, SOCP, MISOCP, SDP) – OpenSource
- MOSEK (LP, SOCP, MISOCP, SDP) – Commercial, free for Academia