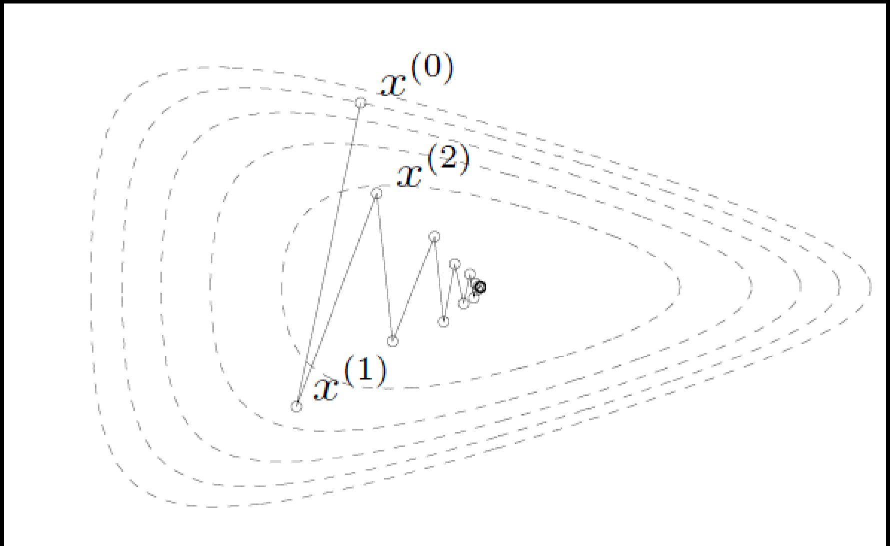
Constrained Optimization

This lecture is being recorded

Last time…

- We saw how to use local methods to solve unconstrained optimization problems



- What do we do if there are constraints on x?
- Example: Find a configuration of a robot arm with the end-effector at a target point while being collision-free

Outline

- Defining convex optimization problems
- Duality
- Linear programming
- Quadratic programming
- Solution methods

Defining Convex Optimization Problems with Constraints

Convex Optimization

"With only a bit of exaggeration, we can say that, if you formulate a practical problem as a convex optimization problem, then you have solved the original problem."

- Stephen Boyd

Definition of a general optimization problem

- The "standard form"

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

- $x \in \mathbf{R}^n$ is the optimization variable
- $f_0 : \mathbf{R}^n \to \mathbf{R}$ is the objective or cost function
- $f_i : \mathbf{R}^n \to \mathbf{R}$, $i = 1, \ldots, m$, are the inequality constraint functions
- $h_i : \mathbf{R}^n \to \mathbf{R}$ are the equality constraint functions

"inf" is a generalization of "min"

$$p^\star = \inf\{f_0(x) \mid f_i(x) \leq 0, \ i = 1, \ldots, m, \ h_i(x) = 0, \ i = 1, \ldots, p\}$$

- $p^\star = \infty$ if problem is infeasible (no x satisfies the constraints)
- $p^\star = -\infty$ if problem is unbounded below

What if we just want something feasible?

- If any solution will do:

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

- $p^\star = 0$ if constraints are feasible; any feasible x is optimal
- $p^\star = \infty$ if constraints are infeasible

Definition of a convex optimization problem

General Optimization Problem

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

Convex Optimization Problem

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

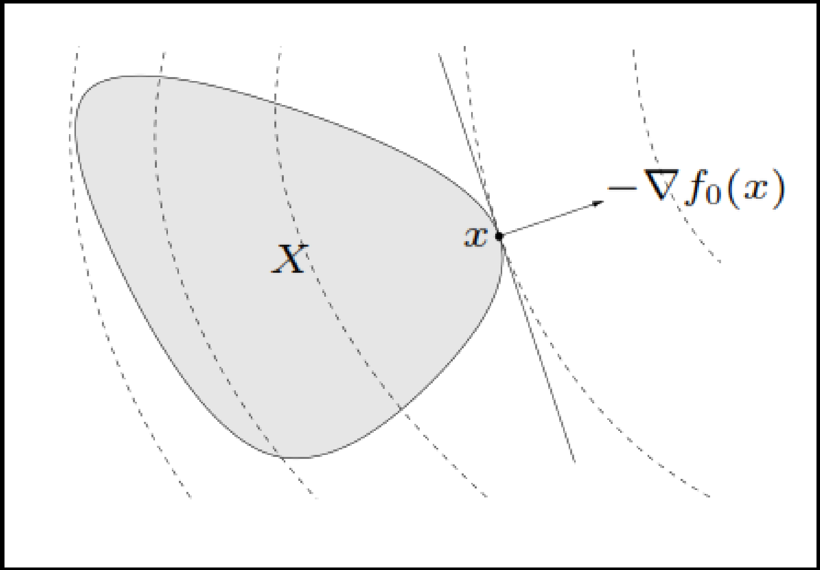
- $f_0, f_1, \ldots, f_m$ are convex

- The feasible set of solutions in a convex optimization problem must be convex
- Any locally-optimal point is globally-optimal!

Optimality for differentiable objective functions in convex optimization

- x is optimal if and only if it is feasible and

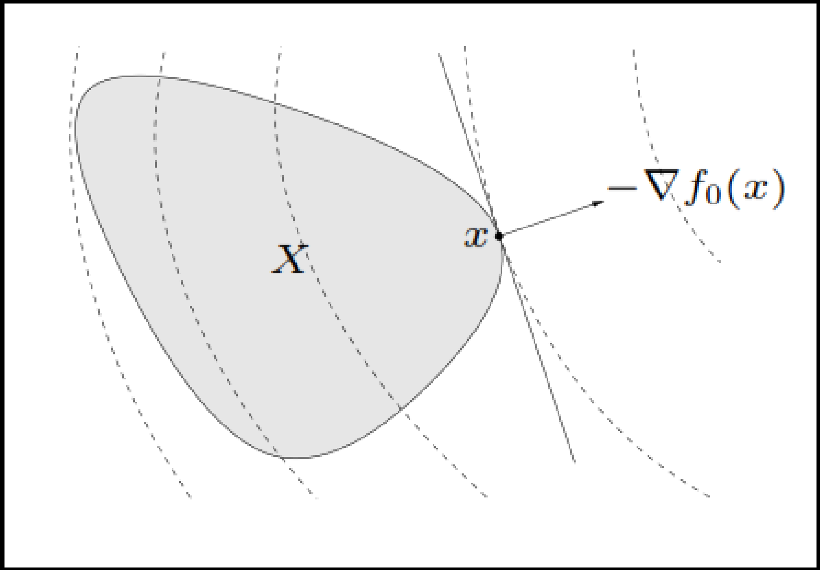
$$\nabla f_0(x)^T(y-x) \geq 0 \quad \text{for all feasible } y$$



- If non-zero, $\nabla f_0(x)$ defines a supporting hyperplane to feasible set X at x

Local methods yield global optimum

- If X is convex and we follow the gradient, we are guaranteed to reach the global optimum



Useful Definitions

- Generalized Inequality $\preceq$ – A generalization of $\leq$
  - Note that generalized inequalities do not necessarily give a linear ordering on elements
  - For our purposes, assume $\preceq$ and $\leq$ are the same
- Infimum (inf) – A generalization of minimum: The greatest lower bound
  - For our purposes, think of this as “min”
- Supremum (sup) – A generalization of maximum: The smallest upper bound
  - For our purposes, think of this as “max”

Duality

Duality Motivation

- We will transform the original (primal) problem into a different (dual) problem

- Analyzing the dual problem allows us to
  - Get a lower bound on the primal problem's optimal value
  - Formulate conditions that must be satisfied for a solution to be optimal

- Many modern optimizers (e.g. MOSEK) use *primal-dual methods*, which try to solve both the primal and dual problems
  - Convex Optimization book Chapter 11.7 shows a method like this, but we won't cover it

- Understanding duality can help you understand what is wrong when an optimizer fails (example)

The standard form *primal* optimization problem

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

- $x \in \mathbf{R}^n$ is the optimization variable
- $f_0 : \mathbf{R}^n \to \mathbf{R}$ is the objective or cost function
- $f_i : \mathbf{R}^n \to \mathbf{R}$, $i = 1, \dots, m$, are the inequality constraint functions
- $h_i : \mathbf{R}^n \to \mathbf{R}$ are the equality constraint functions

$$
p^\star = \inf\{f_0(x) \mid f_i(x) \leq 0,\ i = 1, \dots, m,\ h_i(x) = 0,\ i = 1, \dots, p\}
$$

- $p^\star = \infty$ if problem is infeasible (no x satisfies the constraints)
- $p^\star = -\infty$ if problem is unbounded below

Lagrangian Formulation

- Let's transform the problem

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

Lagrangian: $L : \mathbf{R}^n \times \mathbf{R}^m \times \mathbf{R}^p \to \mathbf{R}$, with $\mathbf{dom}\, L = \mathcal{D} \times \mathbf{R}^m \times \mathbf{R}^p$,

$$
L(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{p} \nu_i h_i(x)
$$

- weighted sum of objective and constraint functions
- λ_i is Lagrange multiplier associated with $f_i(x) \leq 0$
- ν_i is Lagrange multiplier associated with $h_i(x) = 0$

The Lagrange dual function

$$L(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{p} \nu_i h_i(x)$$

$$\begin{aligned} g(\lambda, \nu) &= \inf_{x \in \mathcal{D}} L(x, \lambda, \nu) \\ &= \inf_{x \in \mathcal{D}} \left(f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{p} \nu_i h_i(x) \right) \end{aligned}$$

remember
inf = min
for us

$$g : \mathbf{R}^m \times \mathbf{R}^p \to \mathbf{R}$$

- $g(\lambda,\nu)$ is called the Lagrange Dual Function
- g is **concave**, can be $-\infty$ for some λ,ν

Duality

The Primal Problem

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

The Dual Problem

$$\begin{array}{ll} \text{maximize}_{\lambda, \nu} & g(\lambda, \nu) \\ \text{subject to} & \lambda \succeq 0 \end{array}$$

- Solution to dual problem is a lower-bound on solution to primal problem p^* (more on this soon)
- *The Dual problem is always convex*, regardless of convexity of primal
  - We can use local methods to solve it!
- λ, ν are *dual feasible* if $\lambda \geq 0, (\lambda, \nu) \in \mathbf{dom}\ g$

Duality

- The Dual problem is always convex!
- The Dual problem is always convex!
- The Dual problem is always convex!
- The Dual problem is always convex!
- The Dual problem is always convex!

Proof of Lower Bound Property of Dual

- If $\tilde{x}$ is feasible and $\lambda \geq 0$ then,

$$f_i(\tilde{x}) \leq 0 \text{ and } h_i(\tilde{x}) = 0$$

$$L(\tilde{x}, \lambda, \nu) = f_0(\tilde{x}) + \underbrace{\sum_{i=1}^{m} \lambda_i f_i(\tilde{x})}_{\leq 0} + \underbrace{\sum_{i=1}^{p} \nu_i h_i(\tilde{x})}_{= 0} \leq f_0(\tilde{x}).$$

$$f_0(\tilde{x}) \geq L(\tilde{x}, \lambda, \nu) \geq \inf_{x \in \mathcal{D}} L(x, \lambda, \nu) = g(\lambda, \nu)$$

- Minimizing over all feasible $\tilde{x}$ gives $g(\lambda, \nu) \leq p^*$
- Lower bound property: If $\lambda \geq 0$, then $g(\lambda, \nu) \leq p^*$

How good is the lower bound?

- Let d^* be the optimal value of $g(\lambda, \nu)$, i.e. the solution to the dual problem
- Weak duality: $d^* \leq p^*$
  - Always true (for *convex or non-convex* primal problems)
- Strong duality: $d^* = p^*$
  - Does not hold in general
  - Usually holds for *convex* primal problems
  - Conditions that guarantee strong duality in convex problems are called constraint qualifications (we won't discuss these)

Duality gap

- You can estimate how far you are away from the optimal solution by looking at the Duality Gap: $p_i^* - d_i^*$
  - For your current best solution (at iteration i) to the dual problem and primal problem

- If strong duality holds $p_i^* - d_i^* = 0$ at the optimum, and you know you're done!

- If strong duality doesn't hold, duality gap says how far you are from the lower bound

How can you certify that a solution is optimal?

- If **strong duality holds** and x, λ, ν are optimal, then they must satisfy The Karush-Kuhn-Tucker (KKT) conditions:
  1. Primal constraints are met: $f_i(x) \leq 0,\ i = 1, \ldots, m,\ h_i(x) = 0,\ i = 1, \ldots, p$
  2. Dual constraints are met: $\lambda \geq 0$
  3. Complementarity slackness: $\lambda_i f_i(x) = 0,\ i = 1, \ldots, m$
  4. Gradient of Lagrangian with respect to x is 0

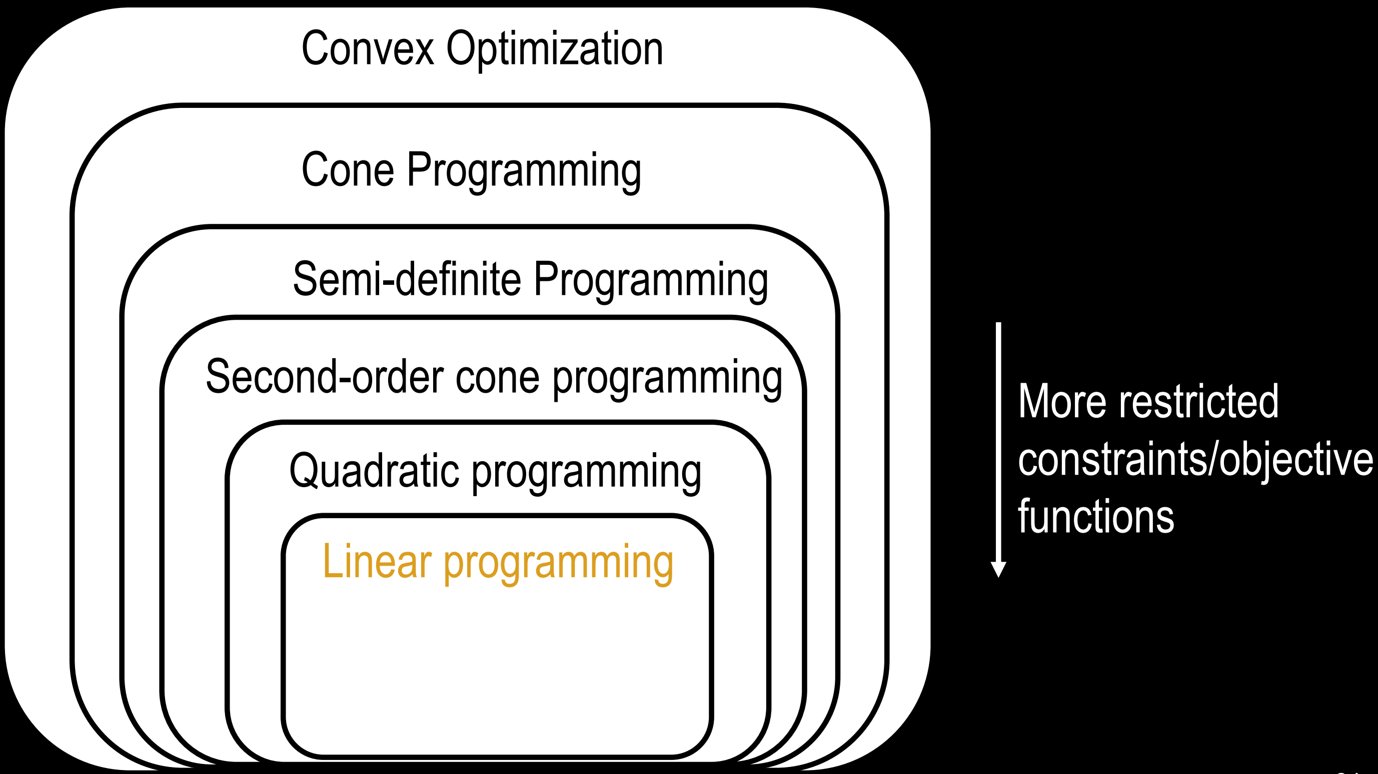
$$\nabla f_0(x) + \sum_{i=1}^{m} \lambda_i \nabla f_i(x) + \sum_{i=1}^{p} \nu_i \nabla h_i(x) = 0$$

- Note: This assumes f_i and h_i are differentiable

Linear Programming

Linear Programming

- Most common form of Convex optimization is linear programming
- A “technology” rather than a research field



Linear Programming

- Standard form Linear Program (LP)

General Optimization Problem

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

Convex Optimization Problem

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \le 0, \quad \cancel{i = 1, \ldots, m} \\ & Ax = b \end{array}$$

- $f_0, f_1, \ldots, f_m$ are convex

LP

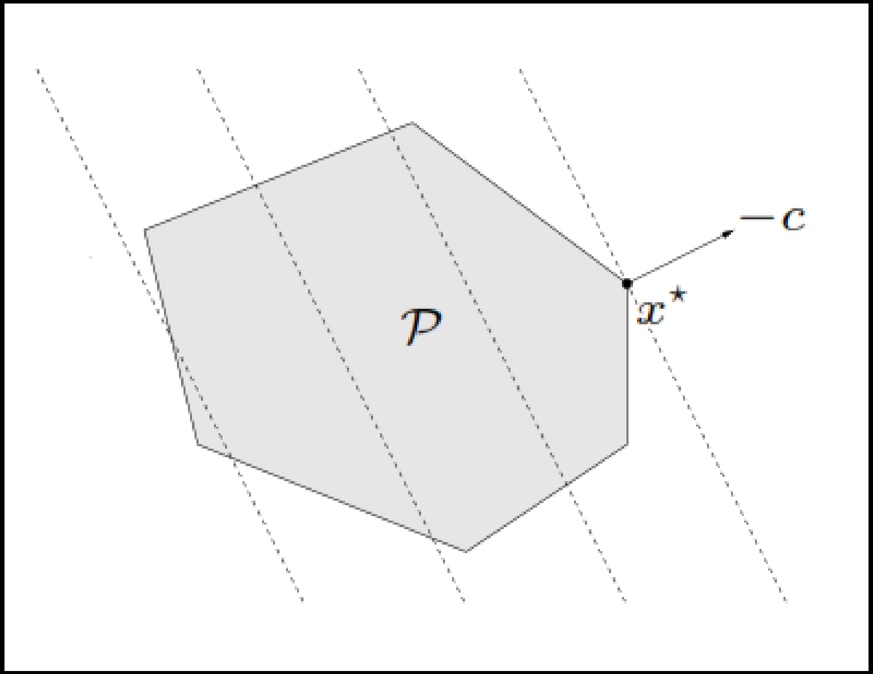
$$\begin{array}{ll} \cancel{\text{minimize}} & c^T x + d \\ \cancel{\text{subject to}} & Gx \preceq h \\ & Ax = b \end{array}$$

- LP is always convex

Linear Programming

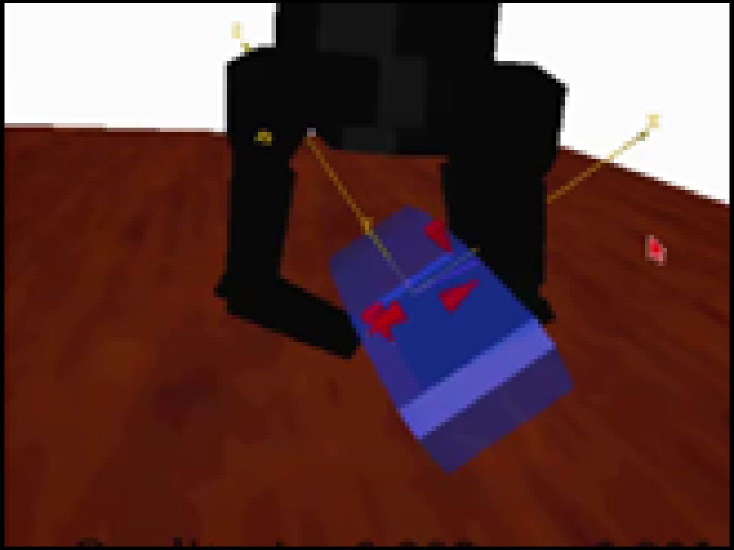
- The feasible set is a polyhedron

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



Example: Grasping

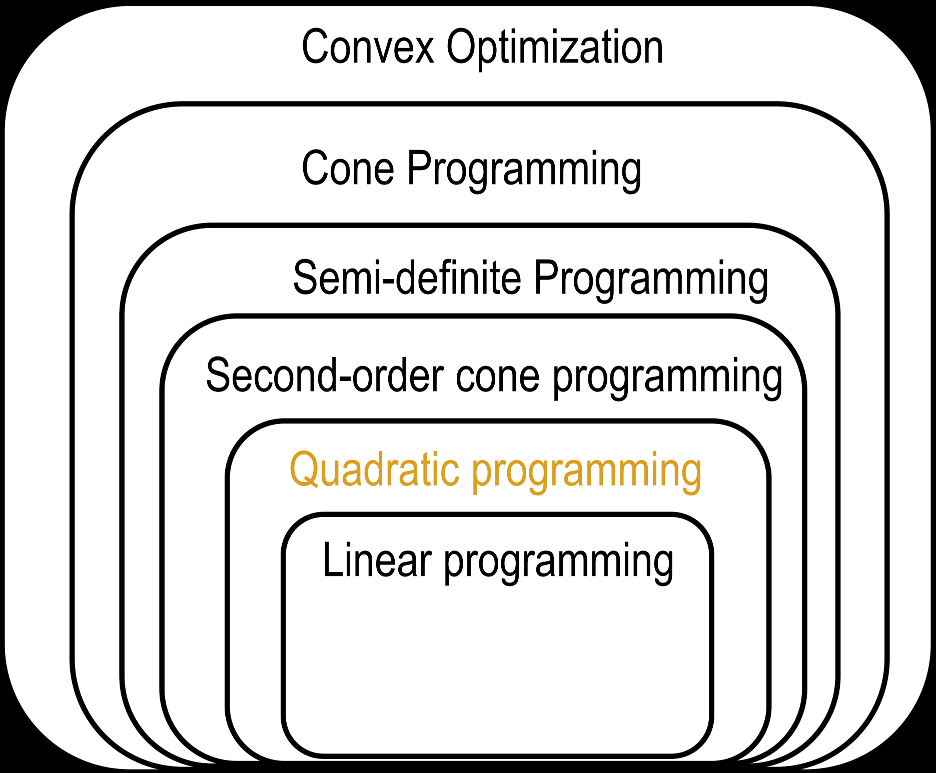
- Can use linear programming to check if a grasp immobilizes an object
  - We'll cover grasping later in the course



Quadratic Programming

Quadratic Programming

- Common form of Convex optimization used in control and robotics
- Many solvers available



More restricted constraints/objective functions

Quadratic Programming

- Standard form Quadratic Program (QP)

<u>General</u> Optimization Problem

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

<u>Convex</u> Optimization Problem

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

- $f_0, f_1, \ldots, f_m$ are convex

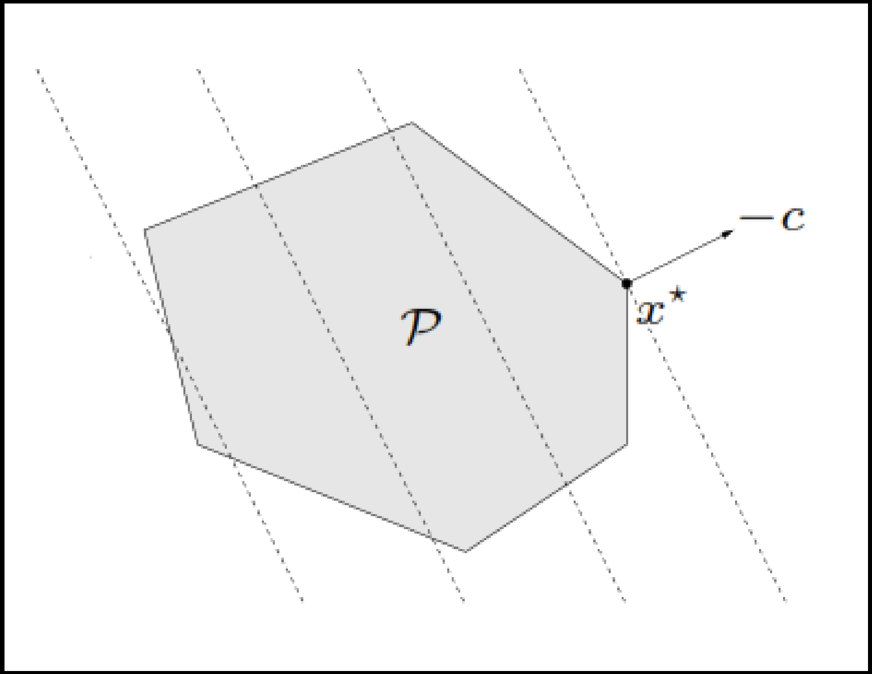
QP

$$\begin{array}{ll} \text{minimize} & (1/2)x^T P x + q^T x + r \\ \text{subject to} & Gx \preceq h \\ & Ax = b \end{array}$$

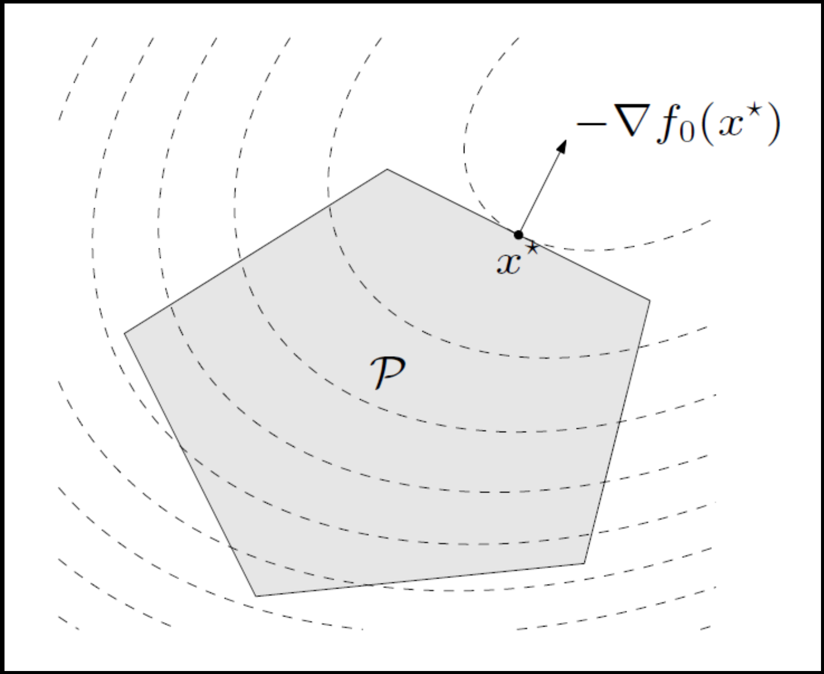
- P must be a symmetric semi-definite matrix
  - $z^T P z \geq 0$ for any z
- Constraints are same as LP
- Objective function is quadratic

Quadratic Programming

- The feasible set is a polyhedron (same as LP)
- Objective function is more expressive than LP



LP



QP

Example: Optimal Control with a QP

- Assume we have a robot with linear dynamics:

$$x_{t+1} = Ax_t + Bu_t$$

where x is the state and u is the control input

- We also have some constraints on each dimension of u

$$u_{i,min} \leq u_i \leq u_{i,max}$$

- We start at state x_0 and want to reach a goal x_{goal}
- Problem: Find a control input u^* that gets the robot as close to the goal as possible.

Example: Optimal Control with a QP

- Let's write this as a QP:

This will make the math simpler/faster without changing u^*

$$\underset{u}{minimize} \quad \left\| x_{goal} - (Ax_0 + Bu) \right\|^2$$

$$Subject\ to \qquad u_{i,min} \le u_i \le u_{i,max} \quad \text{for } i = 1 \ldots |u|$$

Can convert to standard form

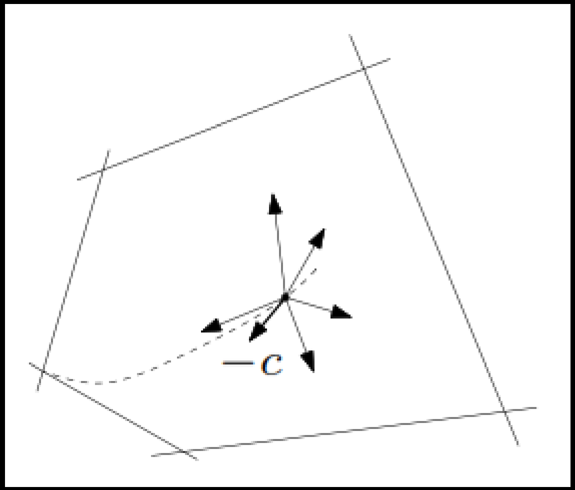
$$\begin{array}{ll} \text{minimize} & (1/2)x^TPx + q^Tx + r \\ \text{subject to} & Gx \preceq h \\ & Ax = b \end{array}$$

Break

Solving an LP

Interior Point Methods (AKA Barrier Methods)

- Solve general Convex Optimization problems, not just linear programming



- Main idea: Start at a point inside the feasible set, pull the point in the direction of decreasing cost while pushing away from constraints

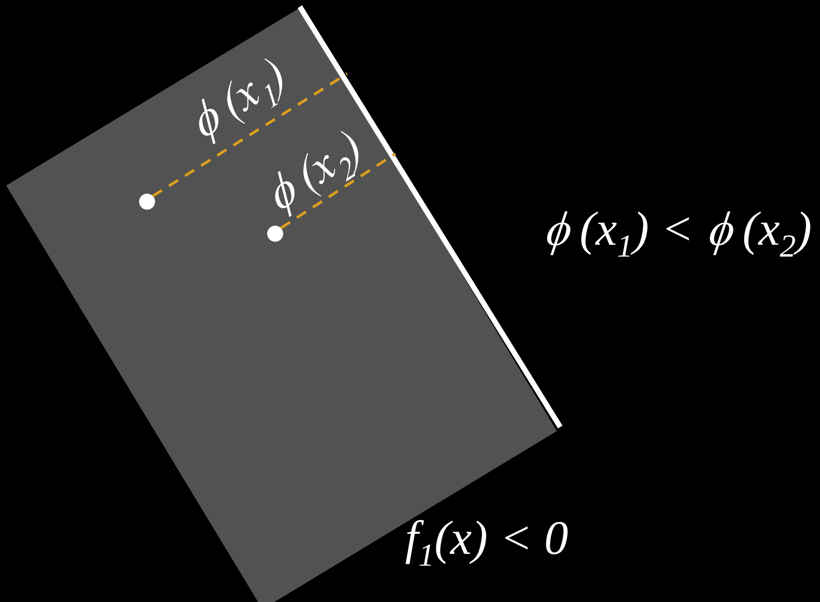
- Interior point methods can get very complicated (we will only discuss the most basic algorithm)

Barriers

- Logarithmic barrier function:

$$\phi(x) = -\sum_{i=1}^{m} \log(-f_i(x)), \quad \mathbf{dom}\,\phi = \{x \mid f_1(x) < 0, \ldots, f_m(x) < 0\}$$

- The closer the point is to the constraint boundary, the bigger $\phi(x)$.



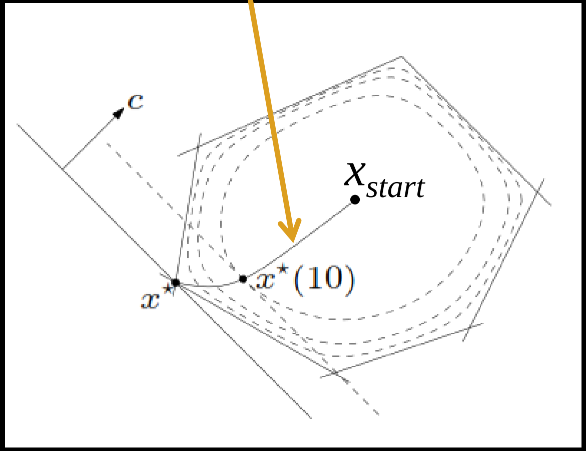
Central Path

- For $t > 0$ define $x^*(t)$ as the solution of

$$\begin{array}{ll} \text{minimize}_x & t f_0(x) + \phi(x) \\ \text{subject to} & Ax = b \end{array}$$

- Solve this optimization problem for increasing t until you get to the optimum
- The sequence of $x^*(t)$ you get is called the central path
- Example: Central path for an LP

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & a_i^T x \le b_i, \quad i = 1, \ldots, 6 \end{array}$$



Force field interpretation

不考虑equality constraints

- The centering problem (drop the equality constraints):

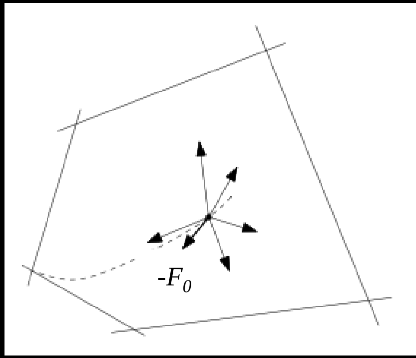
$$\text{minimize}_x \quad t f_0(x) - \sum_{i=1}^{m} \log(-f_i(x))$$

$\phi(x)$

- $t f_0(x)$ is potential of force field $F_0(x) = -t\nabla f_0(x)$
- $-\log(-f_i(x))$ is potential of force field $F_i(x) = (1/f_i(x))\nabla f_i(x)$

- The forces balance at $x^*(t)$ such that:

$$F_0(x^\star(t)) + \sum_{i=1}^{m} F_i(x^\star(t)) = 0$$

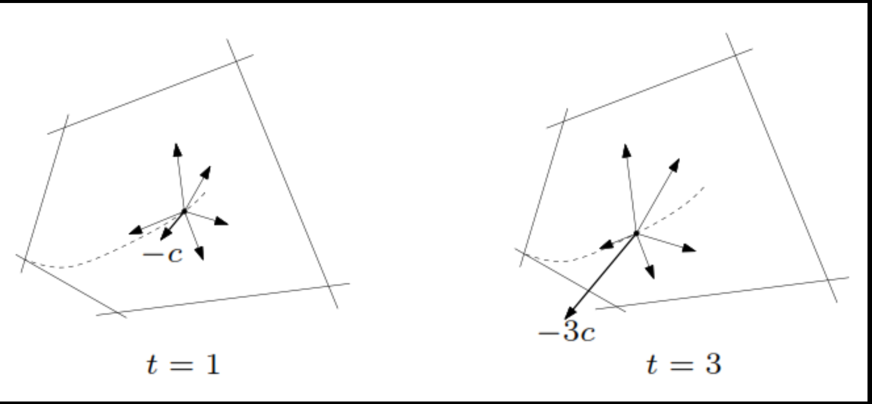


Barrier Method

- Idea: Trace out the central path by increasing t until you get to optimum
- Example for LP:

$$\begin{array}{ll} \text{minimize}_x & c^T x \\ \text{subject to} & a_i^T x \leq b_i, \quad i = 1, \ldots, 6 \end{array}$$

$$x^*(t) = \min_{feasible\ x} tc^T x - \sum_{i=1}^{m} \log(-a_i^T x + b_i)$$



The Barrier Method

We won't discuss barrier method with equality constraints, please ignore

---

given strictly feasible x, $t := t^{(0)} > 0$, $\mu > 1$, tolerance $\epsilon > 0$.

repeat

1. *Centering step.* Compute $x^\star(t)$ by minimizing $t f_0 + \phi$, subject to $Ax = b$.
2. *Update.* $x := x^\star(t)$.
3. *Stopping criterion.* **quit** if $m/t < \epsilon$.
4. *Increase* t. $t := \mu t$.

- terminates with $f_0(x) - p^\star \leq \epsilon$ (stopping criterion follows from $f_0(x^\star(t)) - p^\star \leq m/t$)
- centering usually done using Newton's method, starting at current x
- choice of μ involves a trade-off: large μ means fewer outer iterations, more inner (Newton) iterations; typical values: $\mu = 10$–20
- several heuristics for choice of $t^{(0)}$ (see book)

In practice

- Interior point methods are very complicated
  - We only looked at some basic principles and a basic algorithm
- Interior point methods are also used for solving other convex optimization problems (e.g. QPs)
- Tools for solving LPs and QPs
  - MATLAB's linprog, quadprog
  - CVX

CVX: Software for Disciplined Convex Programming

- A convex optimization solver (wow!), can be used to solve LPs, QPs

- http://cvxr.com/cvx/, Matlab toolbox available
- Python version: https://www.cvxpy.org/ (we will use this for homework)

- Another python solver: http://cvxopt.org/

Summary

- Saw how to define convex optimization problems <u>with constraints</u>
- Solving the dual problem gives us either
  - a lower bound on the optimal solution of the primal problem (weak duality)
  - the exact value of the optimal solution (strong duality)
- The duality gap tells us how far a current solution is
  - from lower bound (weak duality)
  - from optimal value (strong duality)
- Checking KKT conditions certifies a solution is optimal when strong duality holds
- Linear programming is a popular and powerful convex optimization problem class
- Quadratic programming can be used for many control problems in robotics
- Interior point methods are used to solve convex optimization problems
  - Intuition: Push a particle away from constraints and toward direction of decreasing cost

Homework

- AI book Ch. 4.1-4.2
- Graphs in computer science