ELE539A: Optimization of Communication Systems
Lecture 16: Pareto Optimization and Nonconvex Optimization

Professor M. Chiang
Electrical Engineering Department, Princeton University

March 16, 2007

Lecture Outline

- Vector-valued optimization and Pareto optimality
- Application to detection problems
- Nonconvex optimization
- Signomial programming and successive GP approximation
- Integer programming and branch and bound

Vector-valued Convex Optimization

Minimize $f_0(x)$ with respect to K, where

$x \in \mathbf{R}^n$ is optimization variable

$f_0 : \mathbf{R}^n \to \mathbf{R}^q$ is objective function, convex with respect to K

K is proper cone in $\mathbf{R}^q$

x is optimal if it is feasible and $f_0(x) \preceq_K f_0(y)$ for all feasible y

This unambigious optimality is often not satisfied by any x

Pareto Optimality

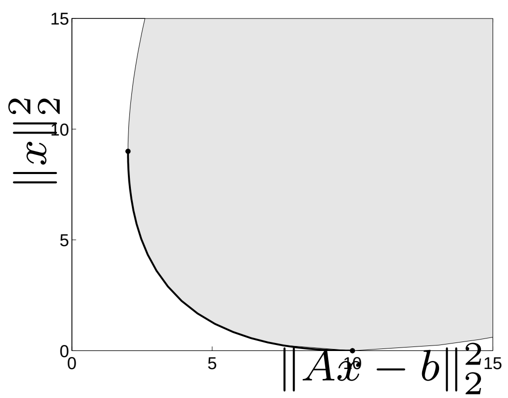
x is Pareto optimal if it is feasible and $f_0(y) \preceq_K f_0(x)$ implies $f_0(y) = f_0(x)$ for any feasible y

Important special case: $K = \mathbf{R}^q_+$. Multi-objective optimization

Tradeoff analysis: x, y both Pareto optimal

$$F_i(x) < F_i(y),\ i \in A,\ \ F_i(x) = F_i(y),\ i \in B,\ \ F_i(x) > F_i(y),\ i \in C,$$

Sets A and C must be either (both) empty or (both) nonempty



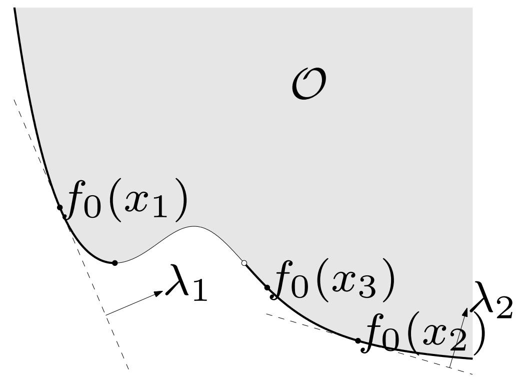
Scalarization

Choose any $\lambda \succ_{K_*} 0$ and solve scalar convex optimization problem:

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

Optimizer x is also Pareto optimal

Converse: for every Pareto optimal x, there exists some nonzero $\lambda \succeq_{K^*} 0$ such that x is the solution to the scalarized problem using λ



Optimal Detector

$X \in \{1, \dots, n\}$ is random variable

$\theta \in \{1, \dots, m\}$ is parameter

X distribution depends on θ through $P \in \mathbf{R}^{n \times m}$:

$$p_{kj} = \mathbf{Prob}(X = k | \theta = j)$$

Estimate θ based on observed X and form $\hat{\theta}$

Detector matrix $T \in \mathbf{R}^{m \times n}$:

$$t_{ik} = \mathbf{Prob}(\hat{\theta} = i | X = k)$$

Obvious constraints:

$$t_k \succeq 0, \;\; \mathbf{1}^T t_k = 1$$

Optimal Detector

Define $D = TP$:

$$D_{ij} = \mathbf{Prob}(\hat{\theta} = i | \theta = j)$$

Detection probabilities:

$$P_i^d = D_{ii}$$

Error probabilities:

$$P_i^e = 1 - D_{ii} = \sum_{j \neq i} D_{ji}$$

Scalar-valued objective functions:

1. Minimax detector: minimize $\max_j P_j^e$
2. Bayes detector: given a prior distribution q for θ, minimize $q^T P^e$
3. MMSE detector: minimize $\sum_i q_i \sum_j (\theta_j - \theta_i)^2 D_{ji}$

Multicriterion Formulation and Scalarization

Multicriterion LP in T:

$$\begin{array}{ll} \text{minimize with}\mathbf{R}_+^{m(m-1)} & D_{ij},\ \ i,j=1,\dots,m,\ \ i\neq j \\ \text{subject to} & t_k \succeq 0,\ \ \mathbf{1}^T t_k = 1,\ \ k=1,\dots,n \end{array}$$

Scalarized: minimize$\mathbf{trace}(W^T D)$ with weights $W_{ii}=0,\ W_{ij}>0, i\neq j$

Let c_k be kth column of WP^T. Since

$$\mathbf{trace}(W^T D) = \mathbf{trace}(W^T TP) = \mathbf{trace}(PW^T T) = \sum_{k=1}^{n} c_k^T t_k$$

Scalarized LP is separable, for each k:

$$\begin{array}{ll} \text{minimize} & c_k^T t_k \\ \text{subject to} & t_k \succeq 0,\ \ \mathbf{1}^T t_k = 1 \end{array}$$

Simple analytic solution: when $X=k$,

$$\hat{\theta} = \underset{j}{\operatorname{argmin}} (WP^T)_{jk}$$

MAP and ML Detector

Bayes optimal detector with a prior distribution q on θ

Weight matrix:

$$W_{ii} = 0, \quad W_{ij} = q_j, i \neq j$$

Since

$$(WP^T)_{jk} = \sum_{i=1}^{m} q_i p_{ki} - q_j p_{kj}$$

optimal detector is MAP detector: when $X = k$,

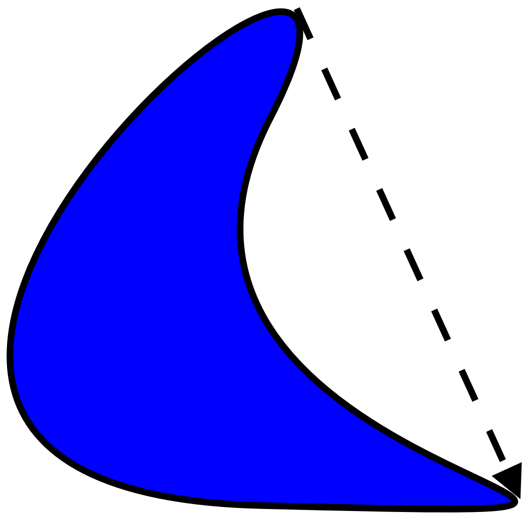
$$\hat{\theta} = \underset{j}{\operatorname{argmax}}(p_{kj} q_j)$$

If q is uniform distribution, reduces to ML detector:

$$\hat{\theta} = \underset{j}{\operatorname{argmax}}(p_{kj})$$

Tackling Nonconvexity

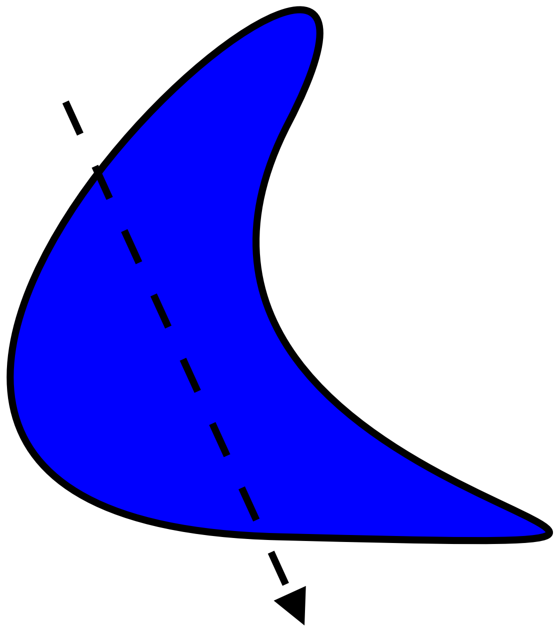
Option 1: Go around nonconvexity



- Geometric Programming, change of variable
- Sufficient condition under which the problem is convex
- Sufficient conditions for uniqueness of KKT points

Tackling Nonconvexity

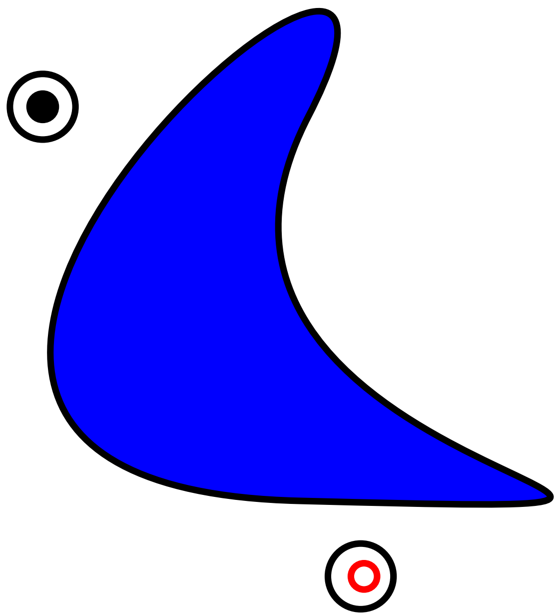
Option 2: Go through nonconvexity



- SOS, Signomial programming, successive convex approximation
- Special structure (e.g., DC, generalized quasiconcavity)
- Smart branch and bound

Tackling Nonconvexity

Option 3: Go above nonconvexity: Design for Optimizability



Change difficult optimization problem, rather than solve it

- Redraw architecture or protocol to make the problem easy to solve
- Need to balance with the cost of making changes to protocols

Optimization as a flag to design issues

Convexity and SIR Regime

Constrained optimization formulations (more general than concave utility maximization over convex feasible region):

Objective: Network-wide metric (*e.g.*, system throughput, min max fairness)

Constraints: Individual user QoS requirements (*e.g.*, rate, delay, outage)

SIR regime matters:

- High SIR: pseudo-nonconvexity
- Medium to Low SIR: real nonconvexity

Notation

Signal Interference Ratio:

$$\mathsf{SIR}_i(\mathbf{P}) = \frac{P_i h_{ii}}{\sum_{j \neq i}^{N} P_j h_{ij} + \eta_i}.$$

Attainable data rate at high SIR:

$$c_i(\mathbf{P}) = \frac{1}{T} \log_2(1 + K\mathsf{SIR}_i(\mathbf{P})).$$

Outage probability on a wireless link:

$$P_{o,i}(\mathbf{P}) = \mathbf{Prob}\{\mathsf{SIR}_i(\mathbf{P}) \leq \mathsf{SIR}_{th}\}$$

Average (Markovian) queuing delay with Poisson(Λ_i) arrival:

$$\bar{D}_i(\mathbf{P}) = \frac{1}{c_i(\mathbf{P}) - \Lambda_i}$$

GP Formulations

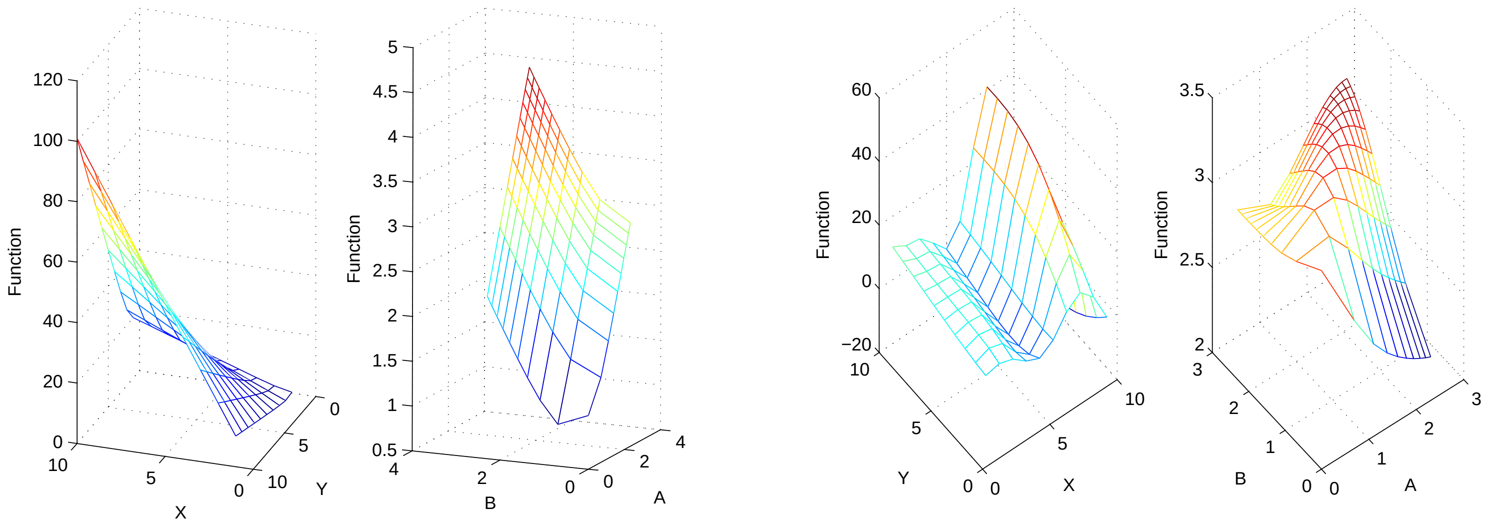
Any combination is GP in high SIR

Any combination without (C,D,c) is GP in any SIR

Objective Function	*Constraints*
(A) Maximize R_{i^*}	(a) $R_i \geq R_{i,min}$
(B) Maximize $\min_i R_i$	(b) $P_{i1}G_{i1} = P_{i2}G_{i2}$
(C) Maximize $\sum_i R_i$	(c) $\sum_i R_i \geq R_{system,min}$
(D) Maximize $\sum_i w_i R_i$	(d) $P_{o,i} \leq P_{o,i,max}$
(E) Minimize $\sum_i P_i$	(e) $0 \leq P_i \leq P_{i,max}$

GP and Convexity

Convexity is not invariant under nonlinear change of coordinates



GP and SOS

Minimize a sum of monomials with upper bound constraints on other sums

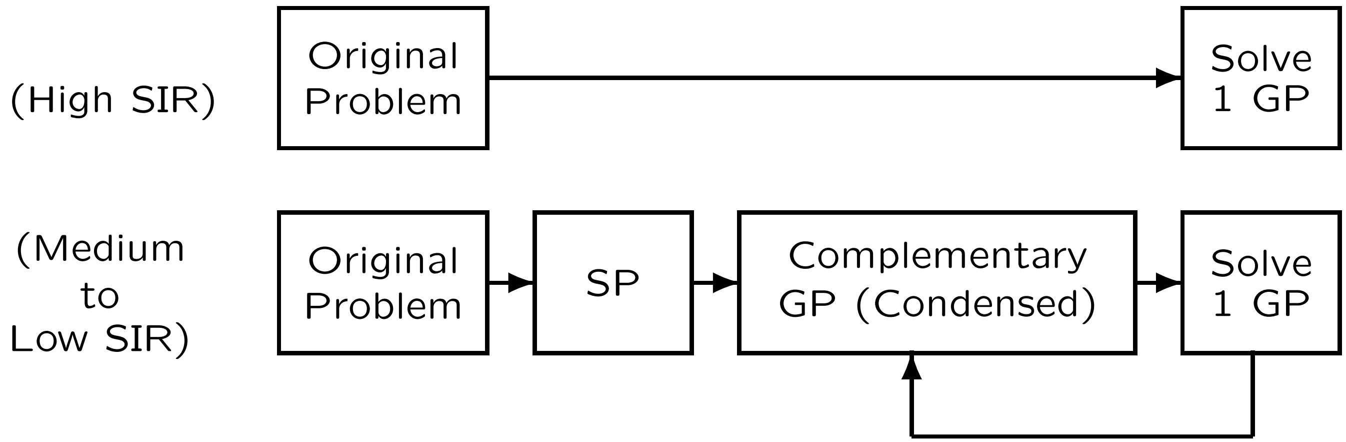
Different definitions of monomial: $c\prod_j x_j^{a^{(j)}}$

	GP	*PMoP*	*SP*
c	$\mathbf{R}_+$	$\mathbf{R}$	$\mathbf{R}$
$a^{(j)}$	$\mathbf{R}$	$\mathcal{Z}_+$	$\mathbf{R}$
x_j	$\mathbf{R}_{++}$	$\mathbf{R}_{++}$	$\mathbf{R}_{++}$

- GP (Polynomial time)
- PMoP: constrained polynomial minimization over the positive quadrant (SOS)
- SP: Signomial Programming (Condensation)

1 GP or Many GPs

SIR is an inverted posynomial in $\mathbf{P}$ but $1 +$ SIR is not



SOS: successive SDP approximations

- Finite time convergence to global optimum

Condensation: successive GP approximations

- Asymptotic convergence to local optimum

Condensation Method

Lemma: $g(\mathbf{x}) = \sum_i u_i(\mathbf{x})$ is a posynomial

$$g(\mathbf{x}) \geq \tilde{g}(\mathbf{x}) = \prod_i \left(\frac{u_i(\mathbf{x})}{\alpha_i}\right)^{\alpha_i}$$

If $\alpha_i = u_i(\mathbf{x}_0)/g(\mathbf{x}_0), \;\; \forall i$, then $\tilde{g}(\mathbf{x}_0)$ is the best monomial approximation

Algorithm:

1) Evaluate the denominator posynomial of signomials with the given $\mathbf{P}$.

2) Compute for each term i in this posynomial,

$$\alpha_i = \frac{\text{value of } i\text{th term in posynomial}}{\text{value of posynomial}}.$$

3) Condense the denominator posynomial into a monomial.

4) Solve the resulting GP using an interior point method.

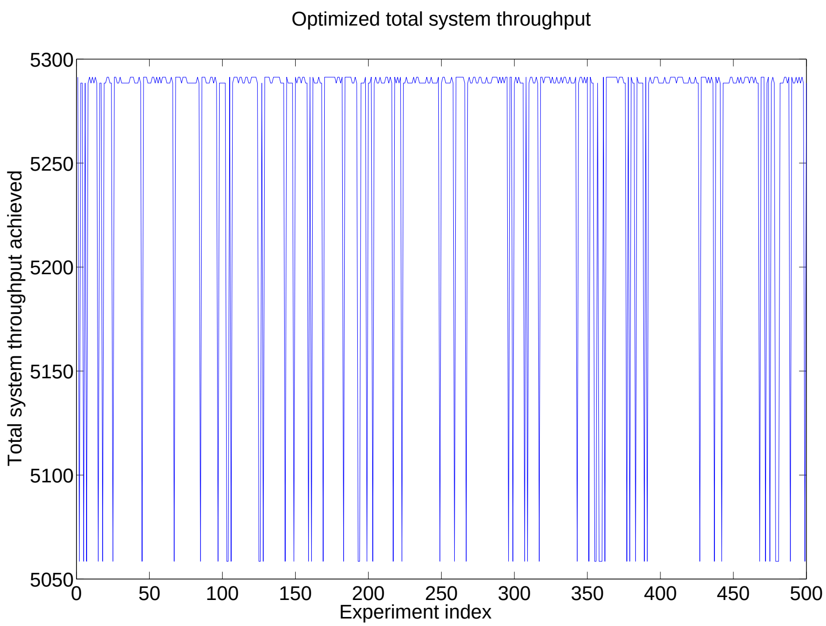
5) Go to step 1 using $\mathbf{P}$ of step 4.

6) Terminate the kth loop if $\| \mathbf{P}^{(k)} - \mathbf{P}^{(k-1)} \| \leq \epsilon$

Example

Problem formulation: System throughput maximization under individual user's rate and outage constraints

96% of the time: attains global optimum



Other Extensions

- Effective heuristics for jumping out of local optimum
- Signomial constraints in constraints

Distributed GP Power Control

Generally applicable to coupled but additive objective function:

$$\text{minimize} \quad \sum_i f_i(x_i, \{x_j\}_{j \in I(i)})$$

Log change of variables for convexity:

$$\text{minimize} \quad \sum_i f_i(e^{y_i}, \{e^{y_j}\}_{j \in I(i)})$$

Introduce auxiliary variables (local copies) and consistency constraints:

$$\begin{array}{ll} \text{minimize} & \sum_i f_i(e^{y_i}, \{e^{y_{ij}}\}_{j \in I(i)}) \\ \text{subject to} & y_{ij} = y_j, \forall j \in I(i), \forall i. \end{array}$$

Lagrangian relaxation for dual decomposition with consistency pricing:

$$\begin{aligned} L(\{y_i\}, \{y_{ij}\}; \{\gamma_{ij}\}) &= \sum_i f_i(e^{y_i}, \{e^{y_{ij}}\}_{j \in I(i)}) + \sum_i \sum_{j \in I(i)} \gamma_{ij}(y_j - y_{ij}) \\ &= \sum_i L_i(y_i, \{y_{ij}\}; \{\gamma_{ij}\}) \end{aligned}$$

Distributed GP Power Control

Lagrange dual problem:

$$\max_{\{\gamma_{ij}\}} \quad g(\{\gamma_{ij}\}) = \sum_i \min_{y_i, \{y_{ij}\}} L_i(y_i, \{y_{ij}\}; \{\gamma_{ij}\})$$

Dual ascent algorithm with consistency pricing update:

$$\gamma_{ij}(t+1) = \gamma_{ij}(t) + \delta(t)(y_j(t) - y_{ij}(t))$$

Message passing needed:

- $\mathcal{O}(N^2)$ as above
- $\mathcal{O}(N)$ by utilizing interference terms

Integer Programming

Some variables can only assume discrete values:

- Boolean constrained
- Integer constraints

Useful for many applications:

- Some resources cannot be infinitesimally divided
- Only discrete levels of control

In general, integer programming is extremely difficult

- Ad hoc solution: *e.g.*, branch and bound (this lecture)
- Systematic solution: *e.g.*, SOS method
- Understand problem structure (this lecture)

MILP

A special case: mixed integer linear programming

LP with some of the variables constrained to be integers:

$$\begin{array}{ll} \text{minimize} & c^T x + d^T y \\ \text{subject to} & Ax + By = b \\ & x, y \succeq 0 \\ & x \in \mathcal{Z}_+^n \end{array}$$

LP relaxation:

$$\begin{array}{ll} \text{minimize} & c^T x + d^T y \\ \text{subject to} & Ax + By = b \\ & x, y \succeq 0 \end{array}$$

Optimal value of LP relaxation provides a lower bound (since minimizing over a larger constraint set) that is readily computed

Boolean Constrained LP

LP with some of the variables constrained to be binary:

$$\begin{array}{ll} \text{minimize} & c^T x + d^T y \\ \text{subject to} & Ax + By = b \\ & x, y \succeq 0 \\ & x_i \in \{0, 1\} \end{array}$$

LP relaxation:

$$\begin{array}{ll} \text{minimize} & c^T x + d^T y \\ \text{subject to} & Ax + By = b \\ & x, y \succeq 0 \\ & x_i \in [0, 1] \end{array}$$

Modelling Techniques

- Binary choice:

$$\begin{aligned} \text{maximize} \quad & c^T x \\ \text{subject to} \quad & w^T x \le K \\ & x_i \in \{0,1\} \end{aligned}$$

where c is value vector and w is weight vector

- Forcing constraint:

If $x \le y$ and $x, y \in \{0,1\}$, then $y = 0$ implies $x = 0$

- Relationship between variables:

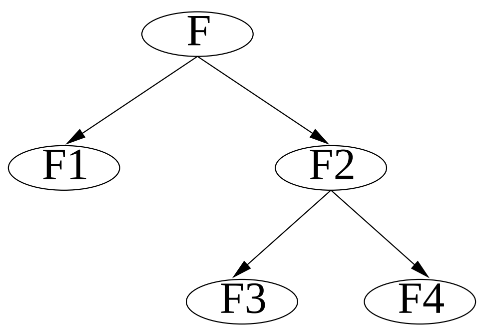
$\sum_{i=1}^n x_i \le 1$ and $x_i \in \{0,1\}$ means at most one of x_i can be 1

$y = \sum_{i=1}^n a_i x_i, \sum_{i=1}^n x_i \le 1$ and $x_i \in \{0,1\}$ means that y must take a value in $\{a_1, \ldots, a_m\}$

Branch and Bound

Instead of exploring the entire set of feasible integer solutions, which is exponential time, use bounds on optimal cost to avoid exploring certain parts of the set of feasible integer solutions

Worst case is still exponential time, but sometimes saves searching time



Split constraint set F into a finite collection of subsets $F_1, \ldots, F_k$ and solve separately each of the problems for $i = 1, \ldots, k$:

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & x \in F_i \end{array}$$

Branch and Bound

Assume we can quickly produce a lower bound on a subproblem:

$$b(F_i) = \min_{x \in F_i} c^T x$$

which is also a lower bound on the original problem

This lower bound is usually obtained by LP relaxation that takes away integer constraints

If a subproblem is solved to integral optimality, or its objective function is evaluated using feasible integral solution, an upper bound U is obtained

If a subproblem produces a lower bound $b(F_i) \geq U$, that branch can be ignored

Branch and bound: delete subproblems that are infeasible or produce a lower bound that is larger than an upper bound on the original problem. Subproblems not deleted either can be solved for optimality or be split into more subproblems

Example

$$\begin{aligned} \text{minimize} \quad & x_1 - 2x_2 \\ \text{subject to} \quad & -4x_1 + 6x_2 \le 9 \\ & x_1 + x_2 \le 4 \\ & x_1, x_2 \ge 0 \\ & x_1, x_2 \in \mathcal{Z} \end{aligned}$$

LP relaxation: $x = (1.5, 2.5), b(F) = -3.5$

Add $x_2 \ge 3$ to form F_1: infeasible, so delete

Add $x_2 \le 2$ to form F_2: $x = (3/4, 2), b(F_2) = -3.25$

Add $x_1 \ge 1$ to form F_3: $x = (1, 2), U = -3$

Add $x_1 \le 0$ to form F_4: $x = (0, 3/2), b(F_4) = -3 \ge U$, so delete

No more subproblem, optimal integer solution $x^* = (1, 2)$