67.612 SPRING 89

COMBINATORIAL OPTIMIZATION AND INTEGER PROGRAMMING

An optimization problem is a problem of the form

$$\min_{x \in S} f(x)$$

Here, $f(x)$ is <u>objective function</u>. eg $f(x) = \begin{cases} x & x \geq 0 \\ 5 & x < 0 \end{cases}$

S is feasible region eg $S = [-1, \infty)$

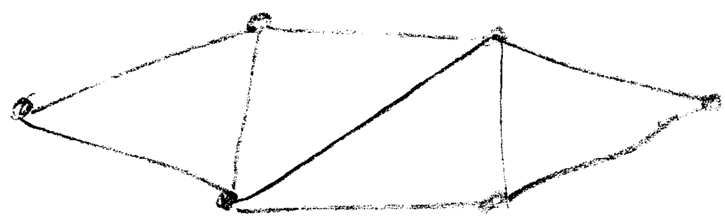
A <u>combinatorial</u> optimization problem is one where there are only a finite number of points in S.

A (mixed) integer programming problem is one where (some of) the variables are restricted to be integer:

eg
$$\min \; c^T x + d^T y$$
$$Ax + By \leq g$$
$$x, y \geq 0 \quad y \text{ integer.}$$

Graph: A collection of edges and vertices:

(More formal defn next time)



Max flow problem: ~~Directed~~ graph: ~~and~~ each edge only allows flow in one direction

Two special vertices s and t ← sink
source ↑

Capacities c_{ij} on edges ~~(i, j)~~ ~~edge connecting vertices i and j~~
(i,j) ← edge going from vertex i to vertex j

Want to maximize flow from s to t.

Introduce variables x_{ij} for each edge (i,j).

What are constraints?

$$\left.\begin{aligned} 0 &\le x_{ij} \\ x_{ij} &\le c_{ij} \end{aligned}\right\} \quad \forall (i,j).$$

Flow conservation:

For all vertices except s and t, flow out = flow ~~out~~ in

ie $\displaystyle\sum_j x_{ij} = \sum_k x_{ki} \quad \forall\, i \ne s, t.$

Objective: maximize net flow out of s

(= net flow into t)

ie

$$\max \sum_j x_{sj} - \sum_i x_{is}$$

Knapsack problem:

Have several ~~some~~ items, each of ~~valuables~~ which is indivisible

Each item j has a certain value b_j, and a certain weight w_j.

Can't carry more than a given weight W

Want to maximize value of goods carried.

Assign variable x_j to each item j

$$x_j = \begin{cases} 1 & \text{if take item } j \\ 0 & \text{o/w} \end{cases}$$

So

$$\max \sum b_j x_j$$

$$\text{s.t.} \quad \sum w_j x_j \leq W$$

$$0 \leq x_j \leq 1, \quad x_j \text{ integer}$$

TSP (Symmetric)

$$\min \sum_{e \in E} c_e x_e$$

$$\text{s.t.} \quad \sum_{e \in \delta(v)} x_e = 2 \quad \forall v \in V$$

$$x_e = 0 \text{ or } 1.$$

Also need subtour elimination constraints

Either: $$\sum_{e \in \delta(U)} x_e \geq 2 \quad \forall U \subseteq V$$

or $$\sum_{e \in E(U)} x_e \leq |U| - 1 \quad \forall U \subseteq V,\ U \neq \emptyset,\ U \neq V.$$

Held & Karp 1-tree relaxation:

Keep restriction that graph is connected.
Relax restriction that every vertex has degree 2.
Require that use $|V|$ edges and that vertex 1 has 2 incident edges.
Such a structure is called a <u>1-tree</u>.

$$\min \sum_{e \in E} c_e x_e + \sum_{i \in V} \lambda_i \left(\sum_{e \in \delta(v)} x_e - 2 \right)$$

$$\text{s.t.} \quad x \text{ is a 1-tree.}$$

Choose λ_i to encourage the solution x to correspond to a tour.

MATCHING

MAXCUT

Eg: Ising spin glass.

LINEAR ORDERING

Eg: taste comparisons.

CHOOSING SONET RINGS

Have graph. Know demands d_{ij} for flow from i to j.
Divide vertices into equally sized clusters - each cluster has a SONET ring.
Synchronous Optical Network.

Choose clusters to maximize flow inside the clusters.

NETWORK DESIGN

Laying cable so get 2-connected graph.

CREW SCHEDULING

Facility location

Given a set $N = \{1, \ldots, n\}$ of potential facility locations

~~set~~ set $I = \{1, \ldots, m\}$ of clients.

Facility placed at j costs c_j for $j \in N$.

Total cost of satisfying demand of client i from a facility at

j is h_{ij}.

Problem: choose a subset of the locations at which to place facilities and then to assign the clients to these facilities so as to minimize total cost.

Binary variables $x_j = \begin{cases} 1 & \text{if a facility is placed at } j \\ 0 & \text{o/w} \end{cases}$

Continuous var y_{ij} = fraction of demand of client i satisfied by facility at j.

Condition that each client's demand must be satisfied is given by

$$\sum_{j \in N} y_{ij} = 1 \quad \text{for } i \in I$$

In addition, client i cannot be served from j unless ~~j~~ a facility is placed at j, so have constraints

$$y_{ij} - x_j \le 0 \quad \text{for } i \in I,\ j \in N$$

So uncapacitated facility location problem is the MIP

$$\min \sum_{j \in N} c_j x_j + \sum_{i \in I} \sum_{j \in N} h_{ij} y_{ij}$$

s.t. $\sum_{j \in N} y_{ij} = 1 \quad \forall i \in I$

$y_{ij} - x_j \leq 0 \quad$ for $i \in I, \; j \in N$

$y_{ij} \geq 0, \quad x_j$ binary.

Solution to LP relaxation may not be integer:

Eg: $c_j = \begin{bmatrix} 4 & 4 & 4 & 12 \end{bmatrix}$

$h_{ij} = \begin{bmatrix} 10 & 1 & 1 & 7 \\ 1 & 10 & 1 & 7 \\ 1 & 1 & 10 & 7 \end{bmatrix}$ 3 facilities, four locations.

$x = [1 \;\; 0 \;\; 0 \;\; 0]$: $\; y_{i1} = 1 \quad i = 1, 2, 3$. Value $= 4 + 10 + 1 + 1 =$ [illegible]

$x = [1 \;\; 1 \;\; 0 \;\; 0]$: $\; y_{12} = 1, \; y_{21} = y_{31} = 1$. Value $= 4 + 4 + 1 + 1 + 1 =$ [illegible]

$x = [1 \;\; 1 \;\; 1 \;\; 0]$: $\; y_{12} = y_{23} = y_{31} = 1$. Value $= 4 + 4 + 4 + 1 + 1 + 1 =$ [illegible]

$x = [\tfrac{1}{2} \;\; \tfrac{1}{2} \;\; \tfrac{1}{2} \;\; 0]$, $\; y_{12} = y_{13} = y_{21} = y_{23} = y_{31} = y_{32} = \tfrac{1}{2}$. Value $= 2 + 2 + 2 + 6(\tfrac{1}{2}) = 9$.

So fractional solution solves the LP relaxation.

If replace $y_{ij} - x_j \leq 0$ by $\sum_{i} y_{ij} - m x_j \leq 0$ $\; j \in N$:

$x = [\tfrac{1}{2} \;\; \tfrac{1}{2} \;\; 0 \;\; 0]$, $\; y_{21} = 1, \; y_{12} = 1, \; y_{31} = y_{32} = \tfrac{1}{2}$ is feasible.

Value $= 2 + 2$ [illegible] $= 7$

Review of Linear Programming (Algebraic characterization).

Standard form LP:

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & Ax = b \\ & x \geq 0 \end{aligned} \qquad (P) \qquad A \in \mathbb{R}^{m \times n},\ b \in \mathbb{R}^m,\ x, c \in \mathbb{R}^n.$$

Consider constraints $Ax = b$.

Pick any m lin indep columns B of A, so $A = [\overbrace{B}^{m} \,\vdots\, \overbrace{N}^{n-m}]\}m$

so $Bx_B + Nx_N = b$ (may rearrange cols).

The basic solution associated with these columns is

$$\underbrace{x_B = B^{-1}b}_{\text{basic variables}}, \quad \underbrace{x_N = 0}_{\text{nonbasic variables}}.$$

Say this solution uses the columns B of A.

If $B^{-1}b \geq 0$, then this is a ~~feasit~~ basic feasible solution, or bfs.

Thm If $\exists$ optimal soln to (P), then there exists a bfs which is optimal

Thm If x is a bfs then $|x_j| \leq m!\,\alpha^{m-1}\beta$,

where $\alpha := \max_{i,j} \{|a_{ij}|\}$

and $\beta := \max_{j=1,\dots,m} \{|b_j|\}$.

Note that $\log(x_j)$ is $O(\dots)$ so these can be stored efficiently.

(PapSte Lemma 2.1).

(see also N&W, prop 3.1, Chapter I.5.3, p. 123)

~~Two bfs are adjacent if the~~

A bfs $\hat{x}$ is adjacent to the bfs $\bar{x}$ if the set of columns used by $\hat{x}$ contains exactly one column not used by $\bar{x}$.

Simplex algorithm:

Moves from a bfs to an adjacent bfs, which is no worse, until ~~act~~ optimality is achieved.

$$\# \text{bfs} \leq \binom{n}{m}$$

So, provided algorithm does not cycle, it converges.

Eg:
$$\begin{aligned} \min \quad & x_1 + x_2 + x_3 + x_4 \\ \text{s.t.} \quad & x_1 + 2x_2 - x_3 - x_4 = 1 \\ & -x_1 - 5x_2 + 2x_3 + 3x_4 = 1 \\ & x_i \geq 0. \end{aligned}$$

BFS: Using cols 1 & 3: $x = [3 \ \ 0 \ \ 2 \ \ 0]^T$
Using cols 1 & 4: $x = [2 \ \ 0 \ \ 0 \ \ 1]^T$ } adjacent bfs.

Cols 1 & 2: basic solution is $x = [\frac{7}{3} \ \ -\frac{2}{3} \ \ 0 \ \ 0]^T$, not bfs.

Cols 2 & 4: $x = [0 \ \ 4 \ \ 0 \ \ 7]^T$ not adj to $[3 \ \ 0 \ \ 2 \ \ 0]^T$.

Cols 2 & 3 : not feas.
3 & 4 : not feas.

LP Duality

Every LP has a dual LP

Dual of

$$\min\ c^T x \qquad Ax = b \qquad x \geq 0 \qquad (P)$$

is

$$\max\ b^T y \qquad A^T y \leq c \qquad (D).$$

In general:

min problem	$\longleftrightarrow$	max problem
obj fn		rhs
A		A^T
rhs		obj fn
constraints $\begin{pmatrix} = \\ \geq \\ \leq \end{pmatrix}$		$\begin{pmatrix} \text{unres} \\ \geq 0 \\ \leq 0 \end{pmatrix}$ variables
$\begin{pmatrix} \text{unres} \\ \geq 0 \\ \leq 0 \end{pmatrix}$ variables		$\begin{pmatrix} = \\ \leq \\ \geq \end{pmatrix}$ ~~variab~~ constraints

Eg: Dual of

$$\begin{aligned} \min\ & c^T x + d^T w \\ & Ax + Bw = b \\ & Cx + Dw \leq h \\ & x \geq 0 \end{aligned}$$

is

$$\begin{aligned} \max\ & b^T y + h^T v \\ & A^T y + C^T v \leq c \\ & B^T y + D^T v = d \\ & v \leq 0 \end{aligned}$$

Weak duality Theorem

Consider the standard pair $\min c^T x$, $Ax = b$, $x \geq 0$ (P) $\quad \max b^T y$, $A^T y \leq c$ (D)

If $\bar{x}$ is feasible in (P) and $\bar{y}$ is feasible in (D) then
$c^T \bar{x} \geq b^T \bar{y}$.

Proof $\quad b^T \bar{y} = (A\bar{x})^T \bar{y} = \bar{x}^T A^T \bar{y} = (A^T \bar{y})^T \bar{x} \leq c^T \bar{x}$
since $\bar{x} \geq 0$ and $A^T \bar{y} \leq c$.

Consequence: If $\bar{x}$ is feasible in (P) and $\bar{y}$ is feasible in (D) and $c^T \bar{x} = b^T \bar{y}$ then $\bar{x}$ and $\bar{y}$ solve (P) and (D) respectively.

Provides check for optimality:

If someone provides you with an $\bar{x}$ and $\bar{y}$, claiming they solve (P), (D), you can verify simply by checking feasibility and making sure $c^T \bar{x} = b^T \bar{y}$.

Strong duality theorem

Consider the standard pair

$$\min c^T x \quad Ax = b \quad x \ge 0 \quad (P) \qquad\qquad \max b^T y \quad A^T y \le c \quad (D).$$

Then one of the following happens:

(1) (P) & (D) are both infeasible

(2) (P) is infeasible and (D) has unbounded ~~optimal~~ objective function value.

(3) (D) is infeasible and (P) has unbounded objective function value.

(4) (P) both have finite optimal value, and they have the same optimal values.

Consequence: The check for optimality is <u>always</u> available.

Simplex gives a dual solution:

$$\begin{aligned} \min\ & c_B^T x_B + c_N^T x_N \\ \text{s.t. } & B x_B + N x_N = b \\ & x_B, x_N \ge 0 \end{aligned} \qquad\qquad \begin{aligned} \max\ & b^T y \\ \text{s.t. } & B^T y \le c_B \\ & N^T y \le c_N \end{aligned}$$

Choose $y = B^{-T} c_B$.

Feasible provided $c_N - N^T B^{-T} c_B \ge 0$,
i.e., red costs ≥ 0.

Obj fn value:
$$b^T B^{-T} c_B = c_B^T B^{-1} b$$
= primal value.

Complementary Slackness.

$\bar{x}$ and $\bar{y}$ are (P) and (D) optimal respectively

$\iff$

$$A\bar{x} = b,\ \bar{x} \geq 0, \quad A^T\bar{y} \leq c, \qquad \text{(feasibility)}$$

$$\bar{x}^T(A^T\bar{y} - c) = 0. \qquad \text{(Comp slackness)}$$

NB: Since $\bar{x} \geq 0$ and $A^T\bar{y} - c \leq 0$,
we have $\bar{x}_i (A^T\bar{y} - c)_i = 0$ for each component i,
so either $\bar{x}_i = 0$ or $(A^T\bar{y})_i = c_i$

The dual solution given on the previous page satisfies complementary slackness.

Optimal iff:
(1) Primal feasible
(2) Dual feasible
(3) Satisfy complementary slackness.

Return to simplex motivation:

$$\begin{aligned} \min \quad & c_B^T x_B + c_N^T x_N \\ & B x_B + N x_N = b \\ & x_B, x_N \geq 0. \end{aligned}$$

So must have $x_B = B^{-1}b - B^{-1}N x_N$

Substitute in objective function:

$$c_B^T x_B + c_N^T x_N = c_B^T B^{-1} b + (c_N^T - c_B^T B^{-1} N) x_N.$$

Assume $B^{-1}b \geq 0$:

If B gives a basic feasible solution (i.e. if $B^{-1}b \geq 0$),

then $c_N^T - c_B^T B^{-1} N$ is vector of reduced costs

If $c_N^T - c_B^T B^{-1} N \geq 0$ then $x_B = B^{-1}b$, $x_N = 0$

is optimal:

Dual can be written

$$\begin{aligned} \max \quad & b^T y \\ & B^T y \leq c_B \\ & N^T y \leq c_N \end{aligned}$$

Consider $y = B^{-T} c_B$:

Then $B^T y = c_B$ $\quad b^T y = c_B{}^T B^{-1} b = c_B{}^T x_B$ $(+ c_N{}^T x_N)$

$c_N - BN^T y = c_N - BN^T B^{-1} c_B$

$=$ vector of reduced costs.

So if ~~x_B is optim~~ $x_B = B^{-1} b$, $x_N = 0$ is (P)-optimal, then $y = B^{-T} c_B$ is (D)-~~optimal~~ ~~feasible~~ optimal.

Simplex:

If some component of $c_N^T - c_B^T B^{-1} N < 0$,

let that variable enter the basis.

Standard variant: choose the component with ~~largest component~~ smallest value.

See handout.

$K = \{ x \in \mathbb{R}^4 : x_1 + 2x_2 - x_3 - x_4 = 1,\ -x_1 - 5x_2 + 2x_3 + 3x_4 = 1,$
$x \geq 0 \}.$

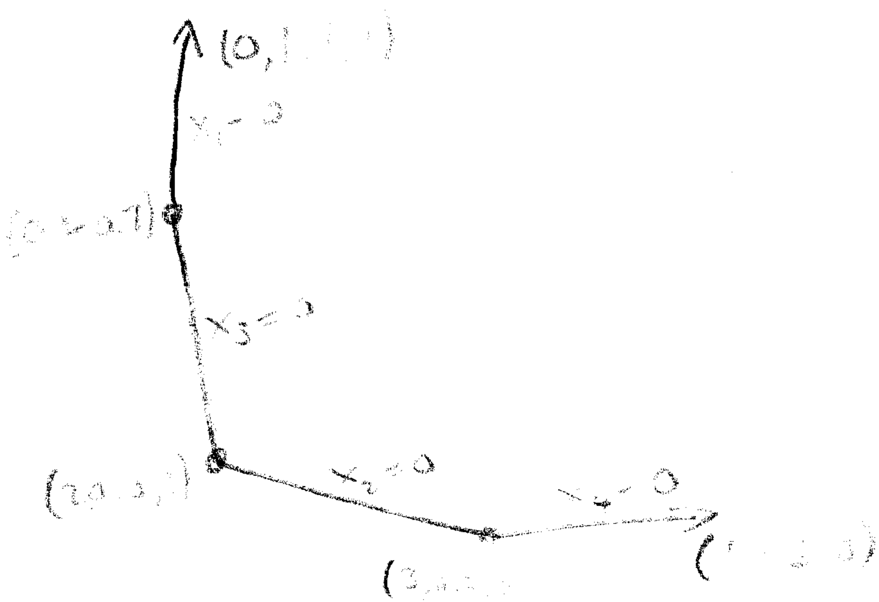
$A = \begin{bmatrix} 1 & 2 & -1 & -1 \\ -1 & -5 & 2 & 3 \end{bmatrix}$ Rank$(A) = 2$,

so $\dim(K) \leq 4 - 2 = 2$.

Also: The points $x^1 = [3\ 0\ 2\ 0]^T$, $x^2 = [2\ 0\ 0\ 1]^T$,
$x^3 = [0\ 4\ 0\ 7]^T$ are all in K.

These are affinely indep

$x^2 - x^1 = \begin{bmatrix} -1 \\ 0 \\ -2 \\ 1 \end{bmatrix}$ $x^3 - x^1 = \begin{bmatrix} -3 \\ 4 \\ -2 \\ 7 \end{bmatrix}$ These two are lin indep.



$x_1, x_3 \geq 0$ gives face of dim 0

equivalent to

$x_1 + 2x_2 - x_4 \geq 1$

$x_1 \geq 0$ gives face of dimension 1

equivalent to

$x_1 \ \ldots \leq$