

Options in Branch-and-Bound

Branching strategy: How do we decide which nodes to examine next?

Having a good feasible solution is important - can prune more nodes by bounds.

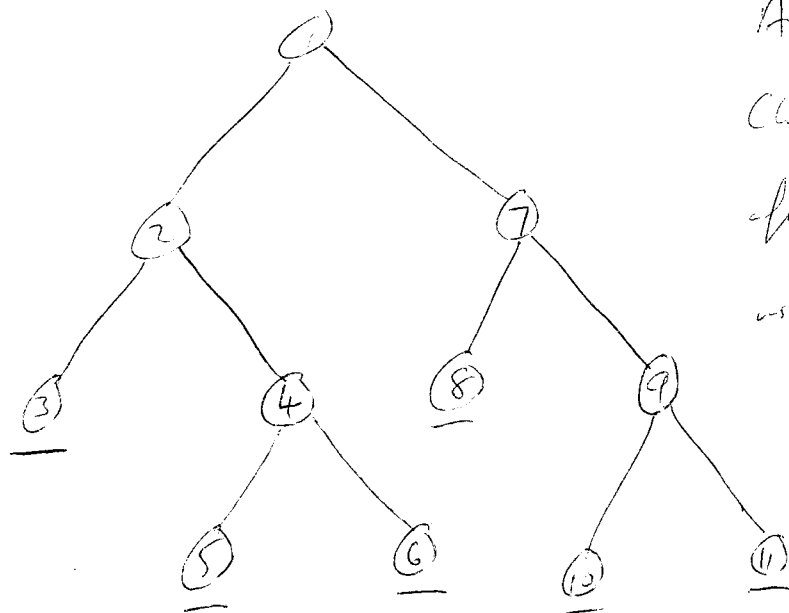
From experience, it seems that the best way to get a feasible solution is to use depth-first-search:

If the current node is not pruned, ~~one of its children~~ is the next node considered is one of its children.

After pruning, use backtracking to get next node:

Go back on path from current node to root until find the first node which has a child which has not been pruned. The next node considered is that child.

Eg



Another advantage of df: child is considered immediately after its parent, so ~~easy~~ if using simplex, can easily recognize w/o having to re-solve basis.

Alternatives to depth-first-search with backtracking:

Could use bounds:

$\bar{z}^i =$ ~~lower~~ ^{upper} bound on $z(P^i)$ obtained from parent of P^i

Could choose P^i with largest \bar{z}^i .

Strong branching: solve several candidate relaxations relative to P^i

Relaxation strategy

LP-relaxation - drop integrality requirements

Other possibilities:

i) if x binary, omit $Ax \leq b$.

Relaxation usually solved (Balas)

Try to futher by checking if infeasible, examining each constraint $a_i^T x \leq b_i$ in turn

ii) Lagrangian relaxation (Glover, Fisher, Held-Karp)

Get (P^i) in of form

$$\begin{aligned} \max \quad & c^T x \\ & Ax \leq b \quad \leftarrow \text{nice constraints (eg, network constraints)} \\ & Dx \leq e \quad \leftarrow \text{dirty constraints} \\ & x \in T \quad \leftarrow \text{includes integrality \& bounds} \end{aligned}$$

Omitting $Dx \leq e$ gives easy problem

Consider "Lagrangian relaxation" for given $\lambda \geq 0$, consider

(iii) Relax for λ

$$\max_{(P^i(\lambda))} \quad c^T x + \lambda^T (e - Dx) = \psi(\lambda)$$

$$\begin{aligned} & Ax \leq b \\ & x \in T \end{aligned}$$

(Choose λ to minimize $\psi(\lambda)$)

Separation strategies

Take $S = S^1 \cup S^2$ with

$$S^1 = S \cap \{x \in \mathbb{R}^n : dx \leq d_0\},$$

$$S^2 = S \cap \{x \in \mathbb{R}^n : dx \geq d_0 + 1\}.$$

where d and d_0 are integers.

If x^* is soln to $\max \{c^T x : x \in \mathbb{R}^n, Ax \leq b\}$,

choose (d, d_0) so that $d_0 < d^T x^* < d_0 + 1$.

Then x^* is not feasible in either of the subsequent relaxations.

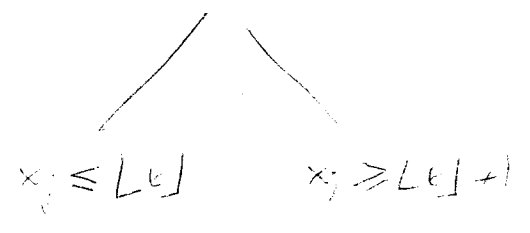
Most common choice of (d, d_0) is:

Variable dichotomy:

Choose noninteger $x_j^* = t$, separate

which variable do we choose?

See Butt, Crowder, Johnson, Padberg,
OR 31, 1983, p. 803-834.



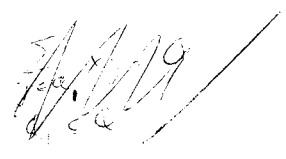
An alternative

or ZOOM: Singhal, Marsten, Morin, ORSA J. of. C. 1(1), 1980, pp. 44-54
- fixed order branching.

If we have generalized upper bound (GUB) constraint

$$\sum_{j \in Q} x_j = 1$$

for some $Q \subseteq N$ (multiple-choice constraint), then can split by



$$\sum_{j \in Q_1} x_j = 0$$

$Q_1 \subseteq Q$

$$\sum_{j \in Q_2} x_j = 0$$

$Q_2 \subseteq Q$

x^* is infeasible in relaxation.

provided $0 < \sum_{j \in Q} x_j^* < 1$.

Implicit Enumeration. (18A, p. 175)

Schemes for computing bounds.

Use simple feasibility and cost comparisons, obtained by direct inspection of a problem's constraints and objective function.

Eg:

$$\begin{aligned} \max \quad & x_1 + 2x_2 - 3x_3 - x_4 \\ \text{s.t.} \quad & -3x_1 - 3x_2 - 3x_3 + x_4 \leq -5 \\ & 2x_1 + 4x_2 + 5x_3 + 4x_4 \leq 5. \\ & x_i \text{ binary, } i=1, \dots, 4. \end{aligned}$$

First constraint implies we need at least two of the variables equal to 1.

Second constraint implies we can have at most one of the variables equal to 1.

Thus, this ~~is~~ problem is infeasible.

Note: LP relaxation is feasible: $x = (\frac{5}{6}, \frac{5}{6}, 0, 0)$.

Can use other procedures. Also useful when deciding what nodes to examine next.

~~Strong branching.~~

46a

Pseudocosts: (Parker & Radner, p. 200) (The IP is a MINIMIZATION problem)

Estimate ^{value} ~~cost~~ of child

Have ~~child~~ problem $P(S_k)$, relaxed $P(\bar{S}_k)$ solved to optimality, so know $v(P(S_k))$.

Should we branch on 0 or 1?

So need to forecast $v(P(S_k: x_p=0))$ and $v(P(S_k: x_p=1))$.

Assign penalties for variables that are currently fractional.

$$\text{Eg: } v(P(S_k: x_p=0)) \approx v(P(\bar{S}_k)) + \mu \bar{x}_p^k + \mu \sum_{\substack{j \in I \\ j \neq p}} \min \{ \bar{x}_j^k, 1 - \bar{x}_j^k \}$$

$$v(P(S_k: x_p=1)) \approx v(P(\bar{S}_k)) + \mu (1 - \bar{x}_p^k) + \mu \sum_{\substack{j \in I \\ j \neq p}} \min \{ \bar{x}_j^k, 1 - \bar{x}_j^k \}$$

More sophisticated:

$$v(P(S_k: x_p=0)) \approx v(P(\bar{S}_k)) + \mu_p^- \bar{x}_p^k + \sum_{\substack{j \in I \\ j \neq p}} \min \{ \mu_j^- \bar{x}_j^k, \mu_j^+ (1 - \bar{x}_j^k) \}$$

separate μ for each variable and also for increasing or decreasing the variable.

See overhead.

When setting up S_8 and S_9 :

How do we get μ ?

Use earlier information. I.e., what happened before when we branched?

$$\text{W.M. sample } \mu = \frac{\sum_{\text{all previous branches}} \text{Increase in value of relaxation}}{\text{Change in fixed variable}} \Bigg/ \# \text{ previous branches.}$$

$$= \frac{1}{6} \left(\sum_{\text{six numbers}} \frac{4.2-3.6}{8/9} + \frac{5.0-3.6}{1/9} + \frac{5.3-4.2}{8/9} + \frac{5.1-4.2}{1/9} + \frac{5.2-5.0}{1/9} + \frac{5.8-5.0}{1/9} \right) \approx 4.22$$

Estimate of pseudo-cost for nodes ⑧, ⑨:

$$\textcircled{8}: v(P(S_8)) \approx 5.1 + 4.22 \left(\frac{2}{5}\right) = 6.79 \leftarrow$$

$$\textcircled{9}: v(P(S_9)) \approx 5.1 + 4.22 \left(\frac{3}{5}\right) = 7.63.$$

Can also estimate:

μ_2^+ = per unit cost of changing x_1 from fractional to 1

$$\approx \frac{5.8-5.0}{8/9} \approx 0.90$$

μ_2^- = per unit cost of changing x_2 from fractional to 0

$$\approx \frac{5.2-5.0}{1/9} \approx 1.80$$

Pseudo cost for nodes ⑧, ⑨:

$$v(P(S_8)) \approx 5.1 + \frac{2}{5}(1.80) = 5.82$$

$$v(P(S_9)) \approx 5.1 + \frac{3}{5}(0.90) = 5.64 \leftarrow$$

Parker & Harding, p. 200 →

(MINIMIZATION problem).

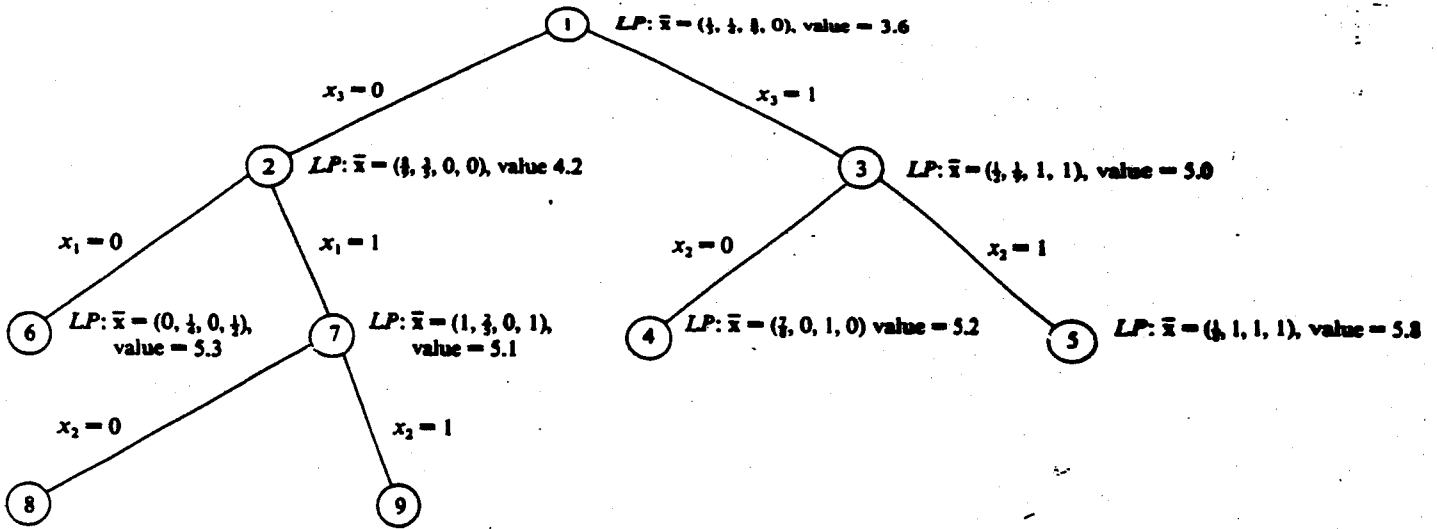


Fig. 5.6. Pseudo-cost Example 5.9.

(48a)

Pseudocosts: (Parker & Radtke, p. 200) (The IP is a minimization problem)

Estimate ^{value} ~~cost~~ of child

Have ~~cost~~ problem $P(S_k)$ relaxed to $P(\bar{S}_k)$ solved to optimality, so know $v(P(\bar{S}_k))$.

Should we branch on 0 or 1?

So need to forecast $v(P(S_k: x_p = 0))$ and $v(P(S_k: x_p = 1))$

Assign penalties for variables that are currently fractional.

Eg:
$$v(P(S_k: x_p = 0)) \approx v(P(\bar{S}_k)) + \mu \bar{x}_p^k + \mu \sum_{\substack{j \in I \\ j \neq p}} \min \{ \bar{x}_j^k, 1 - \bar{x}_j^k \}$$

$$v(P(S_k: x_p = 1)) \approx v(P(\bar{S}_k)) + \mu (1 - \bar{x}_p^k) + \mu \sum_{\substack{j \in I \\ j \neq p}} \min \{ \bar{x}_j^k, 1 - \bar{x}_j^k \}.$$

More sophisticated:

$$v(P(S_k: x_p = 0)) \approx v(P(\bar{S}_k)) + \mu_p^- \bar{x}_p^k + \sum_{\substack{j \in I \\ j \neq p}} \min \{ \mu_j^- \bar{x}_j^k, \mu_j^+ (1 - \bar{x}_j^k) \}$$

separate μ for each variable and also for increasing or decreasing the variable.