



LP-based Branch and Bound

Jesper Larsen

jla@imm.dtu.dk

Department of Management Engineering
Technical University of Denmark



Divide and conquer

In an effort to solve

$$z = \max\{cx : x \in S\}$$

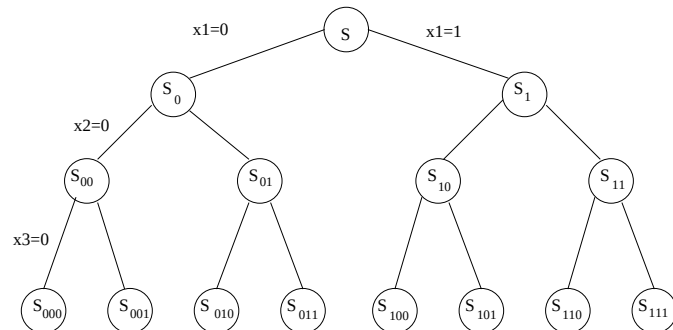
an idea would be to divide the problem into smaller problems that are easier to solve.

- $S = S_1 \cup S_2 \cup \dots \cup S_K$ be a **decomposition** of S .
- $z^k = \max\{cx : x \in S_k\}$ for $k = 1, 2, \dots, K$.
- $z = \max_k z^k$.

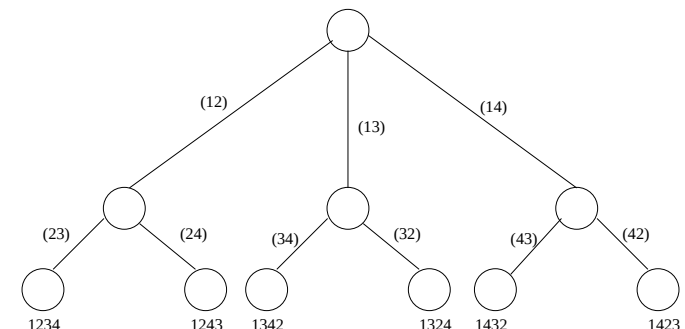


Enumeration tree

A way to represent the decomposition approach is via an enumeration tree.



Enumeration tree for the TSP





Enumeration

In order to overcome large problems we need to do more than just divide and solve leaf nodes.

- How can we put together bound information?
- How can we use some bounds on the values of $\{z^k\}$ intelligently?



Bounds

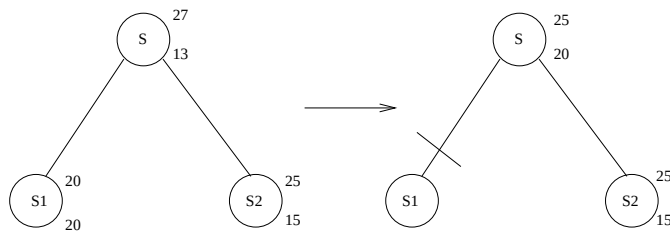
Considering a maximization problem. Let $S = S_1 \cup S_2 \cup \dots \cup S_K$ be a **decomposition** of S , and let $\bar{z}^k$ be an upper bound on z^k and $\underline{z}^k$ be a lower bound on z^k . Then

- $\bar{z} = \max_k \bar{z}^k$ is an upper bound on z .
 - Only $\bar{z}$ is definitely an upper bound.
- $\underline{z} = \max_k \underline{z}^k$ is a lower bound on z .
 - In fact all $\underline{z}^k$'s are lower bounds but $\underline{z}$ is the tightest one.



Pruned by optimality

MAX

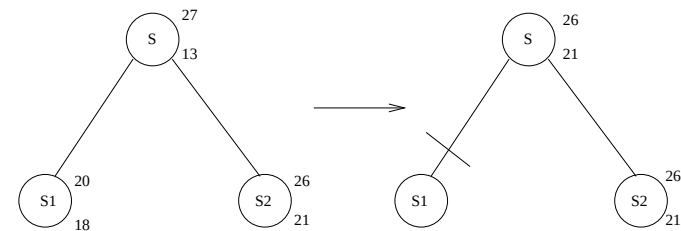


- $\bar{z} = \max_k \bar{z}^k = \max\{20, 25\} = 25$.
- $\underline{z} = \max_k \underline{z}^k = \max\{15, 20\} = 20$.
- $\bar{z}_1 = \underline{z}_1 = 20 = z_1$ – no reason to examine S1.



Pruned by bound

MAX



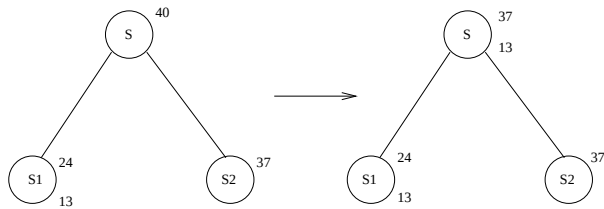
- $\bar{z} = \max_k \bar{z}^k = \max\{20, 26\} = 26$.
- $\underline{z} = \max_k \underline{z}^k = \max\{18, 21\} = 21$.
- $\bar{z}_1 = 20, \underline{z}_2 = 21$ – no reason to examine S1.





No pruning possible

MAX



- $\bar{z} = \max_k \bar{z}^k = \max\{24, 37\} = 37$.
- $\underline{z} = \max_k \underline{z}^k = \max\{13, -\} = 13$.
- No conclusions can be drawn – we have to examine S1 and S2.



Pruning possibilities

Based on the three cases we have:

- Pruning by optimality: $z^t = \max\{cx : x \in S_t\}$ has been solved.
- Pruning by bound: $\bar{z}^t \leq \underline{z}$
- Pruning by infeasibility $S_t = \emptyset$
- Branching



Example

Let us use branch and bound to solve the following integer programming problem:

$$\begin{aligned} z = \max \quad & 4x_1 - x_2 \\ & 7x_1 - 2x_2 \leq 14 \\ & x_2 \leq 3 \\ & 2x_1 - 2x_2 \leq 3 \\ & x \in \mathbb{Z}_+^2 \end{aligned}$$

For generating upper bounds we will use the **LP relaxation**.



Branching I

- As $\underline{z} < \bar{z}$ we need to branch and bound.
- To branch identify an integer variable that is fractional. Define two subproblems:
 - ▶ $S_1 = S \cap \{x : x_j \leq \lfloor \bar{x}_j \rfloor\}$.
 - ▶ $S_2 = S \cap \{x : x_j \geq \lceil \bar{x}_j \rceil\}$.





Branching II

- Note that: $S = S_1 \cup S_2, S_1 \cap S_2 = \emptyset$
- Furthermore note:
 - ▶ $\bar{x}$ of $LP(S)$ is not feasible in either $LP(S_1)$ or $LP(S_2)$.
 - ▶ This implies that if there is no degeneracy so $\max\{\bar{z}_1, \bar{z}_2\} < \bar{z}$



Practical aspects

- **Storing the tree:** List of *active* nodes, best known dual bound, variable lower and upper bounds, optimal/near-optimal basis.
- **How to bound:** LP-relaxation and LP-solver.
- **How to branch:**
 - ▶ Branch on most *fractional* variable.
 - ▶ Branch on least *fractional* variable.
 - ▶ “Estimate the cost of forcing x_j to become integer.”
- **How to choose a node:** Next time!!



Preprocessing

Idea: Detect and eliminate redundant constraints and variables, and tighten bounds where possible.

- Tightening bounds: use known bounds on some variables to tighten bounds on others.
- Redundant constraints
- Variable fixing (by duality)



Tightening bounds

Consider the IP problem:

$$\begin{aligned}
 \max \quad & 2x_1 + x_2 - x_3 \\
 \text{s.t.} \quad & 5x_1 - 2x_2 + 8x_3 \leq 15 \\
 & 8x_1 + 3x_2 - x_3 \geq 9 \\
 & x_1 + x_2 + x_3 \leq 6 \\
 & 0 \leq x_1 \leq 3 \\
 & 0 \leq x_2 \leq 1 \\
 & 1 \leq x_3
 \end{aligned}$$





Preprocessing - for BIPs

- Generating logical inequalities
- Combining pairs of logical inequalities
- Simplifying



Logical inequalities

Consider the constraints of a BIP problem:

$$\begin{aligned}7x_1 + 3x_2 - 4x_3 - 2x_4 &\leq 1 \\ -2x_1 + 7x_2 + 3x_3 + x_4 &\leq 6 \\ -2x_2 - 3x_3 - 6x_4 &\leq -5 \\ 3x_1 - 2x_3 &\geq -1 \\ x &\in B^4\end{aligned}$$

