Integer Programming

Wolfram Wiesemann

December 6, 2007

Contents of this Lecture

Revision: Mixed Integer Programming Problems

Mixed Integer Programming (MIP) Problem:

$$\min \; x_0 = c^{\mathrm{T}} x$$

subject to

$$\begin{aligned} Ax &= b \\ x_j &\geq 0 && \text{for } j \in N = \{1, \ldots, n\} \\ x_j &\in \mathbb{Z} && \text{for } j \in Z \subseteq N. \end{aligned}$$

Note: $x_j \in N \setminus Z$ are continuous, $x_j \in Z$ are integral.

Additional assumption this lecture: Finite bounds $\underline{x}_j$, $\overline{x}_j$ for $j \in Z$: $x_j \in \left\{\underline{x}_j, \underline{x}_j + 1, \ldots, \overline{x}_j\right\}$.

Contents of this Lecture

Solving MIP's: Complete Enumeration

Idea: Loop through all possible values of the integer variables (without loss of generality, $\{x_1, \ldots, x_z\}$ with $z \leq n$) and solve LP problems in the remaining (continuous) variables:

```
for x_1 ∈ {x̲_1, x̲_1 + 1, ..., x̄_1} do
    for x_2 ∈ {x̲_2, x̲_2 + 1, ..., x̄_2} do
        ...
            for x_z ∈ {x̲_z, x̲_z + 1, ..., x̄_z} do
                Solve LP in x_{k+1}, ..., x_n with x_1, ..., x_k fixed.
                Update tentative optimal solution if necessary.
            end for
        ...
    end for
end for
Print (final) optimal solution or report infeasibility.
```

Complexity?

Contents of this Lecture

Divide and Conquer Principle

Need to *structure* search so that we touch only few solutions.

One Approach: Divide and Conquer

- *Divide* a large problem into several smaller ones.
- *Conquer* by working on the smaller problems.

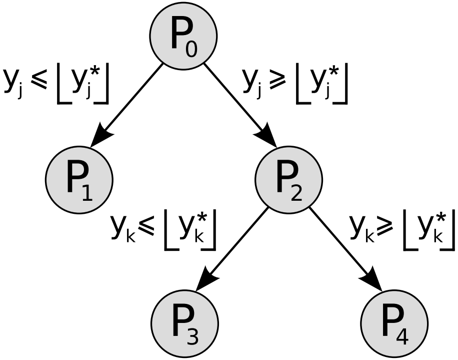
Branch & Bound:

- Solve continuous relaxation of original problem $P_0 \Rightarrow x^*(P_0)$.
- **Divide (Branch):** Choose $p \in Z$ with $x_p^* \notin \mathbb{Z}$. Create two subproblems, P_1 and P_2, with added constraints $x_p \leq \lfloor x_p^* \rfloor$ and $x_p \geq \lceil x_p^* \rceil$, respectively.
- **Conquer (Bound/Fathom):** If optimal solution of continous relaxation of P_i is worse than any known feasible solution for P_0, disregard P_i.

Note: Any solution to P_0 is also feasible for *either* P_1 *or* P_2. Hence, by solving P_1 *and* P_2, we solve P_0.

Divide and Conquer Principle

Recursive application of divide and conquer principle leads to binary tree:



Terminal nodes = problems that remain to be solved.

Contents of this Lecture

Branch & Bound Algorithm for MIP's

Preliminaries:

- P_0 denotes original problem:

$$\min \; x_0 = c^{\mathrm{T}} x$$

subject to

$$\begin{aligned} Ax &= b \\ x_j &\geq 0 && \text{for } j \in N = \{1, \ldots, n\} \\ x_j &\in \mathbb{Z} && \text{for } j \in Z \subseteq N. \end{aligned}$$

- For any problem P, $x^*(P)$ denotes optimal solution for continuous relaxation of P.
- OPT denotes objective function value of best *feasible* solution (for P_0) found so far. At beginning, $OPT = \infty$ or based on a priori knowledge (heuristic).

Branch & Bound Algorithm for MIP's

Algorithm:

1. **Initialization.**
   - Set list of problems to $\{P_0\}$. Initialize OPT.
   - Solve LP relaxation of $P_0 \Rightarrow x^*(P_0)$.
   - If $x^*(P_0)$ feasible for P_0, $OPT = c^{\mathrm{T}} x^*(P_0)$ and stop.
2. **Problem Selection.** Choose a problem P from list whose $x^*(P)$ has $c^{\mathrm{T}} x^*(P) < OPT$. If no such P exists, stop.
3. **Variable Selection.** Choose $x_p \in Z$ with $x_p^*(P) \notin \mathbb{Z}$.
4. **Branching.**
   - Create two new problems P' and P'' with $x_p \leq \lfloor x_p^*(P) \rfloor$ and $x_p \geq \lceil x_p^*(P) \rceil$, respectively.
   - Solve continuous relaxations of P' and $P'' \Rightarrow x^*(P')$, $x^*(P'')$.
   - **Update OPT:** If P' feasible, $x^*(P')$ feasible for P_0 and $c^{\mathrm{T}} x^*(P') < OPT \Rightarrow OPT = c^{\mathrm{T}} x^*(P')$. Same for P''.
   - **Further Inspection:** If P' feasible and $c^{\mathrm{T}} x^*(P') < OPT \Rightarrow$ add P' to list of problems. Same for P''.

   Afterwards, go back to (2).

Branch & Bound Algorithm for MIP's

Output:

- $OPT = \infty$: P_0 is infeasible.
- $OPT < \infty$: P_0 is feasible. $OPT =$ optimal objective value.

Optimal Solution: Obtained via slight modification

- Store vector $\widehat{x}$ for best feasible solution (for P_0) found so far.
- Whenever OPT is updated (Steps 1+4), also update $\widehat{x}$.

Termination: Under assumption of finite bounds $\underline{x}_j$, $\overline{x}_j$ for $j \in Z$, algorithm terminates in finitely many steps.

Contents of this Lecture

Example

Assume the following problem is given:

$$\max\ 2x_1 + 3x_2 + x_3 + 2x_4$$

subject to

$$\begin{aligned} 5x_1 + 2x_2 + x_3 + x_4 &\le 15 \\ 2x_1 + 6x_2 + 10x_3 + 8x_4 &\le 60 \\ x_1 + x_2 + x_3 + x_4 &\le 8 \\ 2x_1 + 2x_2 + 3x_3 + 3x_4 &\le 16. \end{aligned}$$

The bounds are $x_1 \in [0, 3]$, $x_2 \in [0, 7]$, $x_3 \in [0, 5]$ and $x_4 \in [0, 5]$. Furthermore, $x_j \in \mathbb{Z}$ for all $j = 1, \ldots, 4$.

Example

Change to minimization objective (not necessary!):

$$\min \; -2x_1 - 3x_2 - x_3 - 2x_4$$

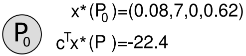
subject to

$$\begin{aligned} 5x_1 + 2x_2 + x_3 + x_4 &\leq 15 \\ 2x_1 + 6x_2 + 10x_3 + 8x_4 &\leq 60 \\ x_1 + x_2 + x_3 + x_4 &\leq 8 \\ 2x_1 + 2x_2 + 3x_3 + 3x_4 &\leq 16. \end{aligned}$$

$x_1 \in [0,3]$, $x_2 \in [0,7]$, $x_3 \in [0,5]$ and $x_4 \in [0,5]$. $x_j \in \mathbb{Z}$ for all $j = 1, \ldots, 4$.

Example

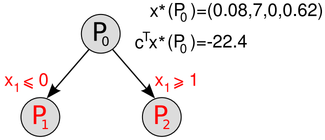
1. **Initialization.**
   - Set list of problems to $\{P_0\}$. Initialize OPT.
   - Solve LP relaxation of $P_0 \Rightarrow x^*(P_0)$.
   - If $x^*(P_0)$ feasible for P_0, $OPT = c^{\mathrm{T}} x^*(P_0)$ and stop.



Problem list: $\{P_0\}$, $OPT = \infty$.

Example

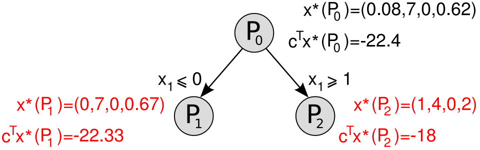
1. **Problem Selection.** Choose a problem P from list whose $x^*(P)$ has $c^{\mathrm{T}}x^*(P) < OPT$. If no such P exists, stop.
2. **Variable Selection.** Choose $x_p \in Z$ with $x_p^*(P) \notin \mathbb{Z}$.
3. **Branching.**
   - Create two new problems P' and P'' with $x_p \leq \lfloor x_p^*(P) \rfloor$ and $x_p \geq \lceil x_p^*(P) \rceil$, respectively.



Problem list: $\{P_1, P_2\}$, $OPT = \infty$.

Example

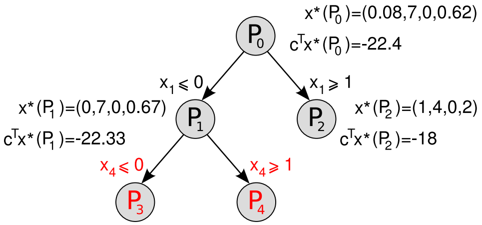
1. **Branching.**
   - Solve continuous relaxations of P' and $P'' \Rightarrow x^*(P')$, $x^*(P'')$.
   - **Update** OPT**:** If P' feasible, $x^*(P')$ feasible for P_0 and $c^{\mathrm{T}}x^*(P') < OPT \Rightarrow OPT = c^{\mathrm{T}}x^*(P')$. Same for P''.
   - **Further Inspection:** If P' feasible and $c^{\mathrm{T}}x^*(P') < OPT \Rightarrow$ add P' to list of problems. Same for P''.



Problem list: $\{P_1\}$, $OPT = -18$.

Example

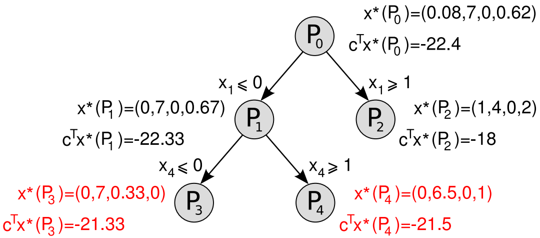
1. **Problem Selection.** Choose a problem P from list whose $x^*(P)$ has $c^{\mathrm{T}}x^*(P) < OPT$. If no such P exists, stop.
2. **Variable Selection.** Choose $x_p \in Z$ with $x_p^*(P) \notin \mathbb{Z}$.
3. **Branching.**
   - Create two new problems P' and P'' with $x_p \leq \lfloor x_p^*(P) \rfloor$ and $x_p \geq \lceil x_p^*(P) \rceil$, respectively.



Problem list: $\{P_3, P_4\}$, $OPT = -18$.

Example

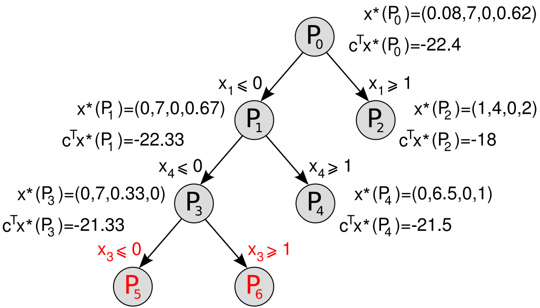
1. **Branching.**
   - Solve continuous relaxations of P' and $P'' \Rightarrow x^*(P'), x^*(P'')$.
   - **Update** OPT**:** If P' feasible, $x^*(P')$ feasible for P_0 and $c^{\mathrm{T}}x^*(P') < OPT \Rightarrow OPT = c^{\mathrm{T}}x^*(P')$. Same for P''.
   - **Further Inspection:** If P' feasible and $c^{\mathrm{T}}x^*(P') < OPT \Rightarrow$ add P' to list of problems. Same for P''.



Problem list: $\{P_3, P_4\}$, $OPT = -18$.

Example

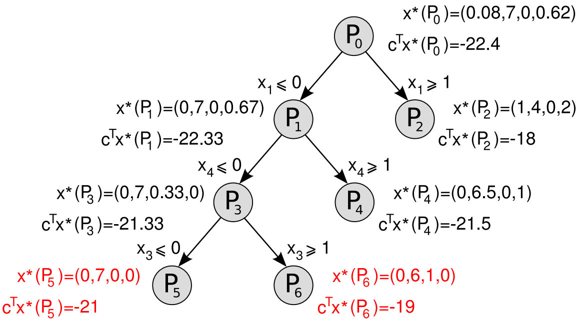
1. **Problem Selection.** Choose a problem P from list whose $x^*(P)$ has $c^{\mathrm{T}}x^*(P) < OPT$. If no such P exists, stop.
2. **Variable Selection.** Choose $x_p \in Z$ with $x_p^*(P) \notin \mathbb{Z}$.
3. **Branching.**
   - Create two new problems P' and P'' with $x_p \leq \lfloor x_p^*(P) \rfloor$ and $x_p \geq \lceil x_p^*(P) \rceil$, respectively.



Problem list: $\{P_4, P_5, P_6\}$, $OPT = -18$.

Example

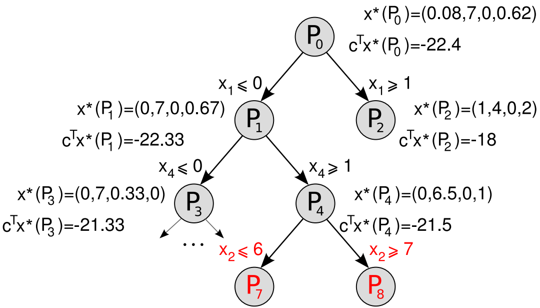
1. **Branching.**
   - Solve continuous relaxations of P' and $P'' \Rightarrow x^*(P'), x^*(P'')$.
   - **Update OPT:** If P' feasible, $x^*(P')$ feasible for P_0 and $c^{\mathrm{T}}x^*(P') < OPT \Rightarrow OPT = c^{\mathrm{T}}x^*(P')$. Same for P''.
   - **Further Inspection:** If P' feasible and $c^{\mathrm{T}}x^*(P') < OPT \Rightarrow$ add P' to list of problems. Same for P''.



Problem list: $\{P_4\}$, $OPT = -21$.

Example

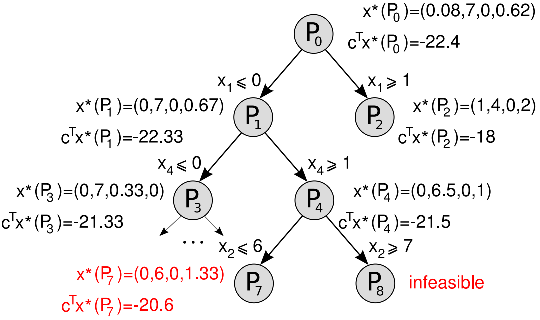
1. **Problem Selection.** Choose a problem P from list whose $x^*(P)$ has $c^{\mathrm{T}}x^*(P) < OPT$. If no such P exists, stop.
2. **Variable Selection.** Choose $x_p \in Z$ with $x_p^*(P) \notin \mathbb{Z}$.
3. **Branching.**
   - Create two new problems P' and P'' with $x_p \leq \lfloor x_p^*(P) \rfloor$ and $x_p \geq \lceil x_p^*(P) \rceil$, respectively.



Problem list: $\{P_7, P_8\}$, $OPT = -21$.

Example

1. **Branching.**
   - Solve continuous relaxations of P' and $P'' \Rightarrow x^*(P'), x^*(P'')$.
   - **Update** OPT**:** If P' feasible, $x^*(P')$ feasible for P_0 and $c^{\mathrm{T}}x^*(P') < OPT \Rightarrow OPT = c^{\mathrm{T}}x^*(P')$. Same for P''.
   - **Further Inspection:** If P' feasible and $c^{\mathrm{T}}x^*(P') < OPT \Rightarrow$ add P' to list of problems. Same for P''.



Problem list: $\{\}$, $OPT = -21$. Done; $\widehat{x} = (0, 7, 0, 0)$.