IE418: Integer Programming. Decomposition Techniques.

Jeff Linderoth

Department of Industrial and Systems Engineering
Lehigh University

25th April 2005

Today

- Homework questions?
- Final discussion
- Preprocessing
- Decompositions and fun with Mosel
- Eat some candy

"Objective Function" Preprocessing

max

$$3x_1 + 2x_2 - 5x_3$$

subject to

$$\begin{aligned} x_1 + x_2 + x_3 &\leq 1 \\ -x_1 + 2x_3 &\leq 4 \\ x &\in \mathbb{B}^3 \end{aligned}$$

min

$$\pi_1 + 4\pi_2 + \mu_1 + \mu_2 + \mu_3$$

subject to

$$\begin{aligned} \pi_1 - \pi_2 + \mu_1 &\geq 3 \\ \pi_1 + \mu_2 &\geq 2 \\ \pi_1 + 2\pi_2 + \mu_3 &\geq -5 \\ \pi &\in \Re^2_+ \\ \mu &\in \Re^3_+ \end{aligned}$$

Advanced Probing Techniques

- Consider $y_1 + 3y_2 \geq 15$
- $x_1 = 0 \Rightarrow$ constraint $y_1 + 3y_2 \geq 15$ is redundant.
- This implies we can improve the coefficient of x_1 (from 0) in the constraint.
- (From the paper) in "1.2—Improving Coefficients", Savelsbergh argues that the set of feasible solutions of the $\leq$ constraint is not changes if both the coefficient of the probing variable and the RHS (b) are reduced by
  - $z_k = -60$
  - $b_i = -15$
- $-45x_1 - y_1 - 3y_2 \leq -60$

Clique Inequalities

- I think the notation is a bit confusing
- $|C^o \cap C^c|$ means the set of variables that have both their nodes in the clique
- $T_C = \{k \in B | x_k, \bar{x}_k \in C\}$
- If $|T_C| = 1$, with $k \in T_C$:
  - $x_k + (1 - x_k) + \sum_{j \in C^o \setminus k} x_j + \sum_{j \in C^c \setminus k} (1 - x_j) \leq 1$
  - $\sum_{j \in C^o \setminus k} x_j + \sum_{j \in C^c \setminus k} (1 - x_j) \leq 0$
  - $x_j = 0 \; \forall j \in C^o \setminus k$
  - $x_j = 1 \; \forall j \in C^c \setminus k$
- A similar argument shows that if $\exists$ a clique such that $|T_C| > 1$, the problem is infeasible

Fall into the...

- Given machines M and jobs N, find a least cost assignment of jobs to machines not exceeding the machine capacities
- Each job $j \in N$ requires a_{ij} units of machine $i \in M$'s capacity b_i

$$
\begin{aligned}
\max z_{GAP} &\equiv \sum_{i \in M} \sum_{j \in N} c_{ij} x_{ij} \\
\text{s.t.} \sum_{j \in N} a_{ij} x_{ij} &\leq b_i \quad \forall i \in M \\
\sum_{i \in M} x_{ij} &= 1 \quad \forall j \in N \\
x_{ij} &\in \{0, 1\} \quad \forall i \in M, \forall j \in N
\end{aligned}
$$

Relaxation 1

Relax the Knapsacks

- Let $X_1 = \{x \in \mathbb{B}^{|M| \times |N|} \mid \sum_{i \in M} x_{ij} = 1 \; \forall j \in N\}$
- For $\mu \in \mathbb{R}_+^{|M|}$

$$LR1(\mu) = \max_{x \in X_1} \sum_{i \in M} \sum_{j \in N} c_{ij} x_{ij} + \sum_{i \in M} \mu_i \left(b_i - \sum_{j \in N} a_{ij} x_{ij} \right)$$

  provides an upper bound on z_{GAP}
- $z_{LD1} \stackrel{\text{def}}{=} \min_{\mu \geq 0} LR1(\mu)$ also provides an upper bound

Relaxation 2

Relax the Assignment Constraints

- Let $X_2 = \{x \in \mathbb{B}^{|M| \times |N|} \mid \sum_{J \in N} a_{ij} x_{ij} \leq b_i \; \forall i \in M\}$
- For any $\mu \in \mathbb{R}_+^{|N|}$

$$LR2(\mu) = \max_{x \in X_2} \sum_{i \in M} \sum_{j \in N} c_{ij} x_{ij} + \sum_{j \in N} \mu_j \left(\sum_{i \in M} x_{ij} - 1 \right)$$

  provides an upper bound on z_{GAP}
- $z_{LD2} \stackrel{\text{def}}{=} \min_{\mu} LR1(\mu)$ also provides an upper bound

The Tale of the Tape

Relax Knapsacks

- "Subproblem" for fixed μ can be solved in linear time.
- For each $j \in N$) find machine i such that $c_{ij} - \mu_i a_{ij}$ is largest. Call solution x^*
- The vector with components $s_i \stackrel{\text{def}}{=} b_i - \sum_{j \in N} a_{ij} x^*_{ij} \in \partial(LR1(\mu))$
- How small can z_{LD1} get?

Relax Assignment

- "Subproblem" for fixed μ is equivalent to the solution of $|M|$ independent knapsack problems.
- For each $i \in M$ find feasible assignment that maximizes with costs $c_{ij} - \mu_i a_{ij}$. Call solution x^*
- The vector with components $s_j \stackrel{\text{def}}{=} \sum_{i \in M} x^*_{ij} - 1 \in \partial(LR2(\mu))$
- How small can z_{LD2} get?

Preprocessing
Decomposition
Comparing Decomposition Schemes

Jeff Linderoth
IE418 Integer Programming