DMP204
SCHEDULING,
TIMETABLING AND ROUTING

Lecture 12

Single Machine Models, Column Generation

Marco Chiarandini
Slides from David Pisinger's lectures at DIKU

Outline

Outline

Relaxation

In branch and bound we find upper bounds by relaxing the problem

Relaxation

$$\max_{s \in P} g(s) \geq \left\{ \begin{array}{c} \max_{s \in P} f(s) \\ \max_{s \in S} g(s) \end{array} \right\} \geq \max_{s \in S} f(s)$$

- P: candidate solutions;
- $S \subseteq P$ feasible solutions;
- $g(x) \geq f(x)$

Which constraints should be relaxed?

- Quality of bound (tightness of relaxation)
- Remaining problem can be solved efficiently
- Proper multipliers can be found efficiently
- Constraints difficult to formulate mathematically
- Constraints which are too expensive to write up

Different relaxations

- LP-relaxation
- Deleting constraint
- Lagrange relaxation
- Surrogate relaxation
- Semidefinite relaxation

Relaxations are often used in combination.

Tightness of relaxation

$$\begin{aligned} &\max cx \\ &\text{s.t. } Ax \le b \\ &\quad\ Dx \le d \\ &\quad\ x \in \mathbb{Z}^n_+ \end{aligned}$$

LP-relaxation:

$$\max\{cx : x \in \text{conv}(Ax \le b, Dx \le d, x \in \mathbb{Z}_+)\}$$

$\rightsquigarrow$ Lagrangian Relaxation:

$$\begin{aligned} &\max z_{LR}(\lambda) = cx - \lambda(Dx - d) \\ &\text{s.t. } Ax \le b \\ &\quad\ x \in \mathbb{Z}^n_+ \end{aligned}$$

LP-relaxation:

$$\max\{cx : Dx \le d, x \in \text{conv}(Ax \le b, x \in \mathbb{Z}_+)\}$$

Relaxation strategies

Which constraints should be relaxed

- "the complicating ones"
- remaining problem is polynomially solvable
  (e.g. min spanning tree, assignment problem, linear programming)
- remaining problem is totally unimodular
  (e.g. network problems)
- remaining problem is NP-hard but good techniques exist
  (e.g. knapsack)
- constraints which cannot be expressed in MIP terms
  (e.g. cutting)
- constraints which are too extensive to express
  (e.g. subtour elimination in TSP)

Subgradient optimization Lagrange multipliers

$$\begin{aligned} &\max\ z = cx \\ &\text{s.t. } Ax \le b \\ &\quad\ Dx \le d \\ &\quad\ x \in \mathbb{Z}^n_+ \end{aligned}$$

Lagrange Relaxation, multipliers $\lambda \ge 0$

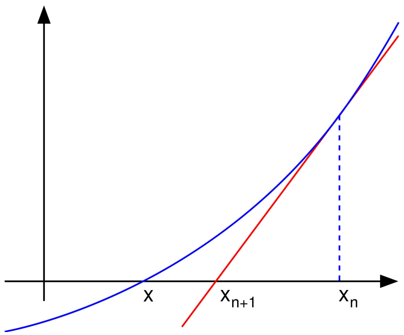
$$\begin{aligned} &\max\ z_{LR}(\lambda) = cx - \lambda(Dx - d) \\ &\text{s.t. } Ax \le b \\ &\quad\ x \in \mathbb{Z}^n_+ \end{aligned}$$

Lagrange Dual Problem

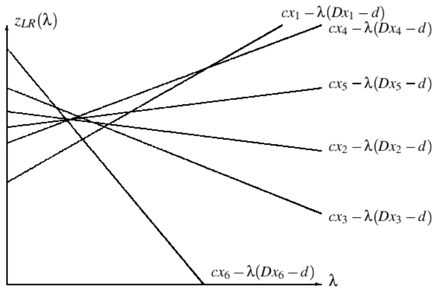
$$z_{LD} = \min_{\lambda \ge 0} z_{LR}(\lambda)$$

- We do not need best multipliers in B&B algorithm
- Subgradient optimization fast method
- Works well due to convexity
- Roots in nonlinear programming, Held and Karp (1971)

Subgradient optimization, motivation



Netwon-like method to minimize a function in one variable



Lagrange function $z_{LR}(\lambda)$ is piecewise linear and convex

Subgradient
Generalization of gradients to non-differentiable functions.

Definition

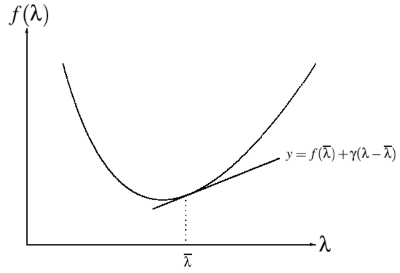
An m-vector γ is subgradient of $f(\lambda)$ at $\lambda - \bar{\lambda}$ if

$$f(\lambda) \geq f(\bar{\lambda}) + \gamma(\lambda - \bar{\lambda})$$

The inequality says that the hyperplane

$$y = f(\bar{\lambda}) + \gamma(\lambda - \bar{\lambda})$$

is tangent to $y = f(\lambda)$ at $\lambda - \bar{\lambda}$ and supports $f(\lambda)$ from below



Proposition Given a choice of nonnegative multipliers $\bar{\lambda}$. If x' is an optimal solution to $z_{LR}(\lambda)$ then

$$\gamma = d - Dx'$$

is a subgradient of $z_{LR}(\lambda)$ at $\lambda = \bar{\lambda}$.

Proof We wish to prove that from the subgradient definition:

$$\max_{Ax \leq b} (cx = \lambda(Dx - d)) \geq \gamma(\lambda - \bar{\lambda}) + \max_{Ax \leq b} \left(cx - \bar{\lambda}(Dx - d)\right)$$

where x' is an opt. solution to the right-most subproblem.
Inserting γ we get:

$$\begin{aligned} \max_{Ax \leq b} (cx - \lambda(Dx - d)) &\geq (d - Dx')(\lambda - \bar{\lambda}) + (cx' - \bar{\lambda}(Dx' - d)) \\ &= cx' - \lambda(Dx' - d) \end{aligned}$$

Intuition
Lagrange relaxation

$$\begin{aligned} &\max z_{LR}(\lambda) = cx - \lambda(Dx - d) \\ &\text{s.t. } Ax \leq b \\ &\quad x \in \mathbb{Z}^n_+ \end{aligned}$$

Gradient in x' is

$$\gamma = d - Dx'$$

Subgradient Iteration
Recursion

$$\lambda^{k+1} = \max\left\{\lambda^k - \theta\gamma^k, 0\right\}$$

where $\theta > 0$ is step-size

If $\gamma > 0$ and θ is sufficiently small $z_{LR}(\lambda)$ will decrease.

- Small θ slow convergence
- Large θ unstable

Held and Karp

Initially
$$\lambda^{(0)} = \{0, \ldots, 0\}$$
compute the new multipliers by recursion
$$\lambda_i^{(k+1)} := \begin{cases} \lambda_i^{(k)} & \text{if } |\gamma_i| \leq \varepsilon \\ \max(\lambda_i^{(k)} - \theta\gamma_i, 0) & \text{if } |\gamma_i| > \varepsilon \end{cases}$$
where γ is subgradient.

The step size θ is defined by
$$\theta = \mu \frac{\overline{z} - \underline{z}}{\sum_i \gamma_i^2}$$
where μ is an appropriate constant.
E.g. $\mu = 1$ and halved if upper bound not decreased in 20 iterations

Lagrange relaxation and LP

For an LP-problem where we Lagrange relax all constraints

- Dual variables are best choice of Lagrange multipliers
- Lagrange relaxation and LP "relaxation" give same bound

Gives a clue to solve LP-problems without Simplex

- Iterative algorithms
- Polynomial algorithms

Outline

Dantzig-Wolfe Decomposition

Motivation

- split it up into smaller pieces a large or difficult problem

Applications

- Cutting Stock problems
- Multicommodity Flow problems
- Facility Location problems
- Capacitated Multi-item Lot-sizing problem
- Air-crew and Manpower Scheduling
- Vehicle Routing Problems
- Scheduling (current research)

Two currently most promising directions for MIP:

- Branch-and-price
- Branch-and-cut

Dantzig-Wolfe Decomposition

The problem is split into a master problem and a subproblem

- \+ Tighter bounds
- \+ Better control of subproblem
- – Model may become (very) large

Delayed column generation

Write up the decomposed model gradually as needed

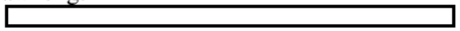
- Generate a few solutions to the subproblems
- Solve the master problem to LP-optimality
- Use the dual information to find most promising solutions to the subproblem
- Extend the master problem with the new subproblem solutions.

Motivation: Cutting stock problem

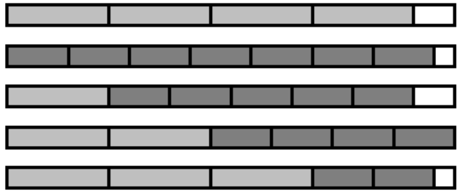
- Infinite number of raw stocks, having length L.
- Cut m piece types i, each having width w_i and demand b_i.
- Satisfy demands using least possible raw stocks.

Example:

- $w_1 = 5, b_1 = 7$ 
- $w_2 = 3, b_2 = 3$ 
- Raw length $L = 22$



Some possible cuts



Formulation 1

$$
\begin{aligned}
\text{minimize} \quad & u_1 + u_2 + u_3 + u_4 + u_5 \\
\text{subject to} \quad & 5x_{11} + 3x_{12} \leq 22u_1 \\
& 5x_{21} + 3x_{22} \leq 22u_2 \\
& 5x_{31} + 3x_{32} \leq 22u_3 \\
& 5x_{41} + 3x_{42} \leq 22u_4 \\
& 5x_{51} + 3x_{52} \leq 22u_5 \\
& x_{11} + x_{21} + x_{31} + x_{41} + x_{51} \geq 7 \\
& x_{12} + x_{22} + x_{32} + x_{42} + x_{52} \geq 3 \\
& u_j \in \{0, 1\} \\
& x_{ij} \in \mathbb{Z}_+
\end{aligned}
$$

LP-relaxation gives solution value $z = 2$ with

$$u_1 = u_2 = 1, x_{11} = 2.6, x_{12} = 3, x_{21} = 4.4$$

Block structure

$$
\begin{array}{lcccccc}
\text{min} & u_1 & +u_2 & +u_3 & +u_4 & +u_5 & \\
\text{s.t} & x_{11} & +x_{21} & +x_{31} & +x_{41} & +x_{51} & \geq 7 \\
 & x_{12} & +x_{22} & +x_{32} & +x_{42} & +x_{51} & \geq 3 \\
 & 5x_{11} + 3x_{12} - 22u_1 & & & & & \leq 0 \\
 & & 5x_{21} + 3x_{22} - 22u_2 & & & & \leq 0 \\
 & & & 5x_{31} + 3x_{32} - 22u_3 & & & \leq 0 \\
 & & & & 5x_{41} + 3x_{42} - 22u_4 & & \leq 0 \\
 & & & & & 5x_{51} + 3x_{52} - 22u_5 & \leq 0
\end{array}
$$

Formulation 2

The matrix A contains all different cutting patterns
All (undominated) patterns:

$$A = \begin{pmatrix} 4 & 0 & 1 & 2 & 3 \\ 0 & 7 & 5 & 4 & 2 \end{pmatrix}$$

Problem

$$
\begin{aligned}
\text{minimize} \quad & \lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 + \lambda_5 \\
\text{subject to} \quad & 4\lambda_1 + 0\lambda_2 + 1\lambda_3 + 2\lambda_4 + 3\lambda_5 \geq 7 \\
& 0\lambda_1 + 7\lambda_2 + 5\lambda_3 + 4\lambda_4 + 2\lambda_5 \geq 3 \\
& \lambda_j \in \mathbb{Z}_+
\end{aligned}
$$

LP-relaxation gives solution value $z = 2.125$ with

$$\lambda_1 = 1.375, \lambda_4 = 0.75$$

Due to integer property a lower bound is $\lceil 2.125 \rceil = 3$.
Optimal solution value is $z^* = 3$.

Round up LP-solution getting heuristic solution $z_H = 3$.

Decomposition

If model has "block" structure

$$
\begin{array}{llllllll}
\max & c^1x^1 & + & c^2x^2 & +\ldots+ & c^Kx^K & & \\
\text{s.t.} & A^1x^1 & + & A^2x^2 & +\ldots+ & A^Kx^K & = & b \\
 & D^1x^1 & + & & & & \le & d_1 \\
 & & + & D^2x^2 & & & \le & d_2 \\
 & & & & \ldots & & \le & \vdots \\
 & & & & & D^Kx^K & \le & d_K \\
 & x^1 \in \mathbb{Z}_+^{n_1} & & x^2 \in \mathbb{Z}_+^{n_2} & \ldots & x^K \in \mathbb{Z}_+^{n_K} & &
\end{array}
$$

Lagrangian relaxation

Objective becomes

$$
\begin{array}{l}
c^1x^1 + c^2x^2 + \ldots + c^Kx^K \\
-\lambda\left(A^1x^1 + A^2x^2 + \ldots + A^Kx^K - b\right)
\end{array}
$$

Decomposed into

$$
\begin{array}{llllllll}
\max & c^1x^1 - \lambda A^1x^1 + c^2x^2 - \lambda A^2x^2 + \ldots + c^Kx^K - \lambda A^Kx^K + b & & & & & & \\
\text{s.t.} & D^1x^1 & + & & & & \le & d_1 \\
 & & + & D^2x^2 & & & \le & d_2 \\
 & & & & \ldots & & \le & \vdots \\
 & & & & & D^Kx^K & \le & d_K \\
 & x^1 \in \mathbb{Z}_+^{n_1} & & x^2 \in \mathbb{Z}_+^{n_2} & \ldots & x^K \in \mathbb{Z}_+^{n_K} & &
\end{array}
$$

Model is separable

Dantzig-Wolfe decomposition

If model has "block" structure

$$
\begin{array}{llllllll}
\max & c^1x^1 & + & c^2x^2 & +\ldots+ & c^Kx^K & & \\
\text{s.t.} & A^1x^1 & + & A^2x^2 & +\ldots+ & A^Kx^K & = & b \\
 & D^1x^1 & + & & & & \le & d_1 \\
 & & + & D^2x^2 & & & \le & d_2 \\
 & & & & \ldots & & \le & \vdots \\
 & & & & & D^Kx^K & \le & d_K \\
 & x^1 \in \mathbb{Z}_+^{n_1} & & x^2 \in \mathbb{Z}_+^{n_2} & \ldots & x^K \in \mathbb{Z}_+^{n_K} & &
\end{array}
$$

Describe each set X^k, $k = 1,\ldots,K$

$$
\begin{array}{llllllll}
\max & c^1x^1 & + & c^2x^2 & +\ldots+ & c^Kx^K & & \\
\text{s.t.} & A^1x^1 & + & A^2x^2 & +\ldots+ & A^Kx^K & = & b \\
 & x^1 \in X^1 & & x^2 \in X^2 & \ldots & x^K \in X^K & &
\end{array}
$$

where $X^k = \{x^k \in \mathbb{Z}_+^{n_k} : D^kx^k \le d_k\}$

Assuming that X^k has finite number of points $\{x^{k,t}\}$ $t \in T_k$

$$
X^k = \left\{ \begin{array}{ll}
x^k \in \mathbb{R}^{n_k}: & x^k = \sum_{t\in T_k} \lambda_{k,t} x^{k,t}, \\
 & \sum_{t\in T_k} \lambda_{k,t} = 1, \\
 & \lambda_{k,t} \in \{0,1\}, t \in T_k
\end{array} \right\}
$$

Dantzig-Wolfe decomposition

Substituting X^k in original model getting *Master Problem*

$$
\begin{array}{lll}
\max c^1\Big(\sum_{t\in T_1} \lambda_{1,t}x^{1,t}\Big) + c^2\Big(\sum_{t\in T_2} \lambda_{2,t}x^{2,t}\Big) + \ldots + c^K\Big(\sum_{t\in T_K} \lambda_{K,t}x^{K,t}\Big) & & \\
\text{s.t.} \;\; A^1\Big(\sum_{t\in T_1} \lambda_{1,t}x^{1,t}\Big) + A^2\Big(\sum_{t\in T_2} \lambda_{2,t}x^{2,t}\Big) + \ldots + A^K\Big(\sum_{t\in T_K} \lambda_{K,t}x^{K,t}\Big) = b & & \\
\quad \sum_{t\in T_k} \lambda_{k,t} = 1 & & k = 1,\ldots,K \\
\quad \lambda_{k,t} \in \{0,1\}, & t \in T_k & k = 1,\ldots,K
\end{array}
$$

Strength of linear master model

Solving LP-relaxation of master problem, is equivalent to (Wolsey Prop 11.1)

$$
\begin{array}{llllllll}
\max & c^1x^1 & + & c^2x^2 & +\ldots+ & c^kx^k & & \\
\text{s.t.} & A^1x^1 & + & A^2x^2 & +\ldots+ & A^kx^k & = & b \\
 & x^1 \in \text{conv}(X^1) & & x^2 \in \text{conv}(X^2) & \ldots & x^k \in \text{conv}(X^k) & &
\end{array}
$$

Proof: Consider LP-relaxation

$$
\begin{array}{lll}
\max c^1\Big(\sum_{t\in T_1} \lambda_{1,t}x^{1,t}\Big) + c^2\Big(\sum_{t\in T_2} \lambda_{2,t}x^{2,t}\Big) + \ldots + c^K\Big(\sum_{t\in T_K} \lambda_{K,t}x^{K,t}\Big) & & \\
\text{s.t.} \;\; A^1\Big(\sum_{t\in T_1} \lambda_{1,t}x^{1,t}\Big) + A^2\Big(\sum_{t\in T_2} \lambda_{2,t}x^{2,t}\Big) + \ldots + A^K\Big(\sum_{t\in T_K} \lambda_{K,t}x^{K,t}\Big) = b & & \\
\quad \sum_{t\in T_k} \lambda_{k,t} = 1 & & k = 1,\ldots,K \\
\quad \lambda_{k,t} \ge 0, & t \in T_k & k = 1,\ldots,K
\end{array}
$$

Informally speaking we have

- joint constraint is solved to LP-optimality
- block constraints are solved to IP-optimality

Strength of Lagrangian relaxation

- z^{LPM} be LP-solution value of master problem
- z^{LD} be solution value of lagrangian dual problem

(Theorem 11.2)
$$z^{LPM} = z^{LD}$$

Proof: Lagrangian relaxing joint constraint in

$$\begin{array}{llllllll}
\max & c^1x^1 & + & c^2x^2 & + \dots + & c^Kx^K & & \\
\text{s.t.} & A^1x^1 & + & A^2x^2 & + \dots + & A^Kx^K & = & b \\
 & D^1x^1 & + & & & & \leq & d_1 \\
 & & + & D^2x^2 & & & \leq & d_2 \\
 & & & & \dots & & \leq & \vdots \\
 & & & & & D^Kx^K & \leq & d_K \\
 & x^1 \in \mathbb{Z}_+^{n_1} & & x^2 \in \mathbb{Z}_+^{n_2} & \dots & x^K \in \mathbb{Z}_+^{n_K} & &
\end{array}$$

Using result next page

$$\begin{array}{lllllll}
\max & c^1x^1 & + & c^2x^2 & + \dots + & c^kx^k & \\
\text{s.t.} & A^1x^1 & + & A^2x^2 & + \dots + & A^kx^k & = b \\
 & x^1 \in \operatorname{conv}(X^1) & & x^2 \in \operatorname{conv}(X^2) & \dots & x^k \in \operatorname{conv}(X^k) &
\end{array}$$

Strength of Lagrangian Relaxation (section 10.2)

Integer Programming Problem

$$\begin{array}{ll}
\text{maximize} & cx \\
\text{subject to} & Ax \leq b \\
 & Dx \leq d \\
 & x_j \in \mathbb{Z}_+, \quad j = 1, \dots, n
\end{array}$$

Lagrange Relaxation, multipliers $\lambda \geq 0$

$$\begin{array}{ll}
\text{maximize} & z_{LR}(\lambda) = cx - \lambda(Dx - d) \\
\text{subject to} & Ax \leq b \\
 & x_j \in \mathbb{Z}_+, \quad j = 1, \dots, n
\end{array}$$

for best multiplier $\lambda \geq 0$

$$\boxed{\max \left\{ cx : Dx \leq d, x \in \operatorname{conv}(Ax \leq b, x \in \mathbb{Z}_+) \right\}}$$

Delayed Column Generation

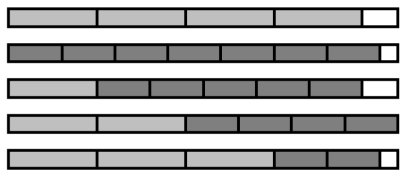
Delayed column generation, linear master

- Master problem can (and will) contain many columns
- To find bound, solve LP-relaxation of master
- Delayed column generation gradually writes up master

Delayed column generation, linear master

- $w_1 = 5, b_1 = 7$ 
- $w_2 = 3, b_2 = 3$ 
- Raw length $L = 22$

Some possible cuts



In matrix form
$$A = \begin{pmatrix} 4 & 0 & 1 & 2 & 3 & \cdots \\ 0 & 7 & 5 & 4 & 2 & \cdots \end{pmatrix}$$

LP-problem
$$\begin{array}{ll}
\min & cx \\
\text{s.t.} & Ax = b \\
 & x \geq 0
\end{array}$$

where

- $b = (7, 3)$,
- $x = (x_1, x_2, x_3, x_4, x_5, \cdots)$
- $c = (1, 1, 1, 1, 1, \cdots)$.

Reduced Costs

Simplex in matrix form

$$\min\{cx \mid Ax = b, x \geq\}$$

In matrix form:

$$\begin{bmatrix} 0 & A \\ -1 & c \end{bmatrix} \begin{bmatrix} z \\ x \end{bmatrix} = \begin{bmatrix} b \\ 0 \end{bmatrix}$$

- $\mathcal{B} = \{1, 2, \ldots, p\}$ basic variables
- $\mathcal{L} = \{1, 2, \ldots, q\}$ non-basis variables (will be set to lower bound = 0)
- $(\mathcal{B}, \mathcal{L})$ basis structure
- $x_\mathcal{B}, x_\mathcal{L}, c_\mathcal{B}, c_\mathcal{L}$,
- $B = [A_1, A_2, \ldots, A_p]$, $L = [A_{p+1}, A_{p+2}, \ldots, A_{p+q}]$

$$\begin{bmatrix} 0 & B & L \\ -1 & c_\mathcal{B} & c_\mathcal{L} \end{bmatrix} \begin{bmatrix} z \\ x_\mathcal{B} \\ x_\mathcal{L} \end{bmatrix} = \begin{bmatrix} b \\ 0 \end{bmatrix}$$

$$Bx_\mathcal{B} + Lx_\mathcal{L} = b \quad \Rightarrow \quad x_\mathcal{B} + B^{-1}Lx_\mathcal{L} = B^{-1}b \quad \Rightarrow \quad \left[\begin{array}{l} x_\mathcal{L} = 0 \\ x_\mathcal{B} = B^{-1}b \end{array}\right.$$

$$\begin{bmatrix} 0 & B & L \\ -1 & c_\mathcal{B} & c_\mathcal{L} \end{bmatrix} \begin{bmatrix} z \\ x_\mathcal{B} \\ x_\mathcal{L} \end{bmatrix} = \begin{bmatrix} b \\ 0 \end{bmatrix}$$

Simplex algorithm sets $x_\mathcal{L} = 0$ and $x_\mathcal{B} = B^{-1}b$
B invertible, hence rows linearly independent

The objective function is obtained by multiplying and subtracting constraints by means of multipliers π (the dual variables)

$$z = \sum_{j=1}^{p}\left[c_j - \sum_{i=1}^{p}\pi_i a_{ij}\right] + \sum_{j=1}^{q}\left[c_j - \sum_{i=1}^{p}\pi_i a_{ij}\right] + \sum_{i=1}^{p}\pi_i b_i$$

Each basic variable has cost null in the objective function

$$c_j - \sum_{i=1}^{p}\pi_i a_{ij} = 0 \qquad \Longrightarrow \qquad \pi = B^{-1}c_\mathcal{B}$$

Reduced costs of non-basic variables:

$$c_j - \sum_{i=1}^{p}\pi_i a_{ij}$$

Delayed column generation (example)

- $w_1 = 5, b_1 = 7$
- $w_2 = 3, b_2 = 3$
- Raw length $L = 22$

Initially we choose only the trivial cutting patterns

$$A = \begin{pmatrix} 4 & 0 \\ 0 & 7 \end{pmatrix}$$

Solve LP-problem

$$\begin{array}{ll} \min & cx \\ \text{s.t.} & Ax = b \\ & x \geq 0 \end{array}$$

i.e.

$$\begin{pmatrix} 4 & 0 \\ 0 & 7 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 7 \\ 3 \end{pmatrix}$$

with solution $x_1 = \frac{7}{4}$ and $x_2 = \frac{3}{7}$.
The dual variables are $y = c_B A_B^{-1}$ i.e.

$$\begin{pmatrix} 1 & 1 \end{pmatrix}\begin{pmatrix} \frac{1}{4} & 0 \\ 0 & \frac{1}{7} \end{pmatrix} = \begin{pmatrix} \frac{1}{4} \\ \frac{1}{7} \end{pmatrix}$$

Small example (continued)

Find entering variable

$$\begin{array}{lcl} A & = & \begin{pmatrix} 1 & 2 & 3 & \cdots \\ 5 & 4 & 2 & \cdots \end{pmatrix} \qquad \begin{array}{l} \frac{1}{4} \leftarrow y_1 \\ \frac{1}{7} \leftarrow y_2 \end{array} \\ c_N - yA_N & = & \begin{pmatrix} 1 - \frac{27}{28} & 1 - \frac{30}{28} & 1 - \frac{29}{28} & \cdots \end{pmatrix} \end{array}$$

We could also solve optimization problem

$$\begin{array}{ll} \min & 1 - \frac{1}{4}x_1 - \frac{1}{7}x_2 \\ \text{s.t.} & 5x_1 + 3x_2 \leq 22 \\ & x \geq 0, \text{integer} \end{array}$$

which is equivalent to knapsack problem

$$\begin{array}{ll} \max & \frac{1}{4}x_1 + \frac{1}{7}x_2 \\ \text{s.t.} & 5x_1 + 3x_2 \leq 22 \\ & x \geq 0, \text{integer} \end{array}$$

This problem has optimal solution $x_1 = 2, x_2 = 4$.
Reduced cost of entering variable

$$1 - 2\frac{1}{4} - 4\frac{1}{7} = 1 - \frac{30}{28} = -\frac{1}{14} < 0$$

Small example (continued)

Add new cutting pattern to A getting

$$A = \begin{pmatrix} 4 & 0 & 3 \\ 0 & 7 & 2 \end{pmatrix}$$

Solve problem to LP-optimality, getting primal solution

$$x_1 = \frac{5}{8}, x_3 = \frac{3}{2}$$

and dual variables

$$y_1 = \frac{1}{4}, y_2 = \frac{1}{8}$$

Note, we do not need to care about "leaving variable"
To find entering variable, solve

$$\begin{aligned} \max \ & \frac{1}{4}x_1 + \frac{1}{8}x_2 \\ \text{s.t.} \ & 5x_1 + 3x_2 \leq 22 \\ & x \geq 0, \text{integer} \end{aligned}$$

This problem has optimal solution $x_1 = 4$, $x_2 = 0$.
Reduced cost of entering variable

$$1 - 4\frac{1}{4} - 0\frac{1}{7} = 0$$

Terminate with $x_1 = \frac{5}{8}$, $x_3 = \frac{3}{2}$, and $z_{LP} = \frac{17}{8} = 2.125$.

Questions

- Will the process terminate?

  Always improving objective value. Only a finite number of basis solutions.

- Can we repeat the same pattern?

  No, since the objective functions is improved. We know the best solution among existing columns. If we generate an already existing column, then we will not improve the objective.

Outline

1. Lagrangian Relaxation
2. Dantzig-Wolfe Decomposition
   Dantzig-Wolfe Decomposition
   Delayed Column Generation
3. Single Machine Models

Scheduling

$1|prec|\sum w_j C_j$

Sequencing (linear ordering) variables

$$\begin{aligned} \min & \sum_{j=1}^{n}\sum_{k=1}^{n} w_j p_k x_{kj} + \sum_{j=1}^{n} w_j p_j \\ \text{s.t. } & x_{kj} + x_{lk} + x_{jl} \geq 1 \qquad j,k,l = 1,\dots,n j \neq k, k \neq l \\ & x_{kj} + x_{jk} = 1 \qquad \forall j,k = 1,\dots,n, j \neq k \\ & x_{jk} \in \{0,1\} \qquad j,k = 1,\dots,n \\ & x_{jj} = 0 \qquad \forall j = 1,\dots,n \end{aligned}$$

Scheduling

$1|prec|C_{max}$

Completion time variables

$$
\begin{aligned}
\min & \sum_{j=1}^{n} w_j z_j \\
\text{s.t. } & z_k - z_j \geq p_k \quad \text{for } j \to k \in A \\
& z_j \geq p_j, \quad \text{for } j = 1, \ldots, n \\
& z_k - z_j \geq p_k \quad \text{or} \quad z_j - z_k \geq p_j, \quad \text{for } (i,j) \in I \\
& z_j \in \mathbf{R}, \quad j = 1, \ldots, n
\end{aligned}
$$

Scheduling

$1||\sum h_j(C_j)$

Time indexed variables

$$
\begin{aligned}
\min & \sum_{j=1}^{n} \sum_{t=1}^{T-p_j+1} h_j(t+p_j) x_{jt} \\
\text{s.t. } & \sum_{t=1}^{T-p_j+1} x_{jt} = 1, \quad \text{for all } j = 1, \ldots, n \\
& \sum_{j=1}^{n} \sum_{s=t-p_j+1}^{t} x_{js} \leq 1, \quad \text{for each } t = 1, \ldots, T \\
& x_{jt} \in \{0,1\}, \quad \text{for each } j = 1, \ldots, n; \ t = 1, \ldots, T - p_j + 1
\end{aligned}
$$

+ This formulation gives better bounds than the two preceding

− pseudo-polynomial number of variables

Dantzig-Wolfe decomposition

Reformulation:

$$
\begin{aligned}
\min & \sum_{j=1}^{n} \sum_{t=1}^{T-p_j+1} h_j(t+p_j) x_{jt} \\
\text{s.t. } & \sum_{t=1}^{T-p_j+1} x_{jt} = 1, \quad \text{for all } j = 1, \ldots, n \\
& x_{jt} \in X \quad \text{for each } j = 1, \ldots, n; \ t = 1, \ldots, T - p_j + 1
\end{aligned}
$$

where $X = \left\{ x \in \{0,1\} : \sum_{j=1}^{n} \sum_{s=t-p_j+1}^{t} x_{js} \leq 1, \text{ for each } t = 1, \ldots, T \right\}$

$x^l, l = 1, \ldots, L$ extreme points of X.

$$
X = \left\{ \begin{array}{lcl} x \in \{0,1\} & : & x = \sum_{l=1}^{L} \lambda_l x^l \\ & & \sum_{l=1}^{L} \lambda_l = 1, \\ & & \lambda_l \in \{0,1\} \end{array} \right\}
$$

matrix of X is interval matrix

extreme points are integral

they are pseudo-schedules

Dantzig-Wolfe decomposition

Substituting X in original model getting master problem

$$
\begin{aligned}
& \min \sum_{j=1}^{n} \sum_{t=1}^{T-p_j+1} h_j(t+p_j) \left(\sum_{l=1}^{L} \lambda_l x^l\right) \\
\pi \quad & \text{s.t. } \sum_{t=1}^{T-p_j+1} \sum_{l=1}^{L} \lambda_l x^l_{jt} = 1, \quad \text{for all } j = 1, \ldots, n \Longleftarrow \sum_{l=1}^{L} \lambda_l n^l_j = 1 \\
\alpha \quad & \sum_{l=1}^{L} \lambda_l = 1, \\
& \lambda_l \in \{0,1\} \Longleftarrow \lambda_l \geq 0 \text{ LP-relaxation}
\end{aligned}
$$

- solve LP-relaxation by column generation on pseudo-schedules x^l
- reduced cost of λ_k is $\bar{c}_k = \sum_{j=1}^{n} \sum_{t=1}^{T-p_j+1} (c_{jt} - \pi_j) x^k_{jt} - \alpha$

- The subproblem can be solved by finding shortest path in a network N with
  - $1, 2, \ldots, T+1$ nodes corresponding to time periods
  - process arcs, for all j, t, $t \rightarrow t + p_j$ and cost $c_{jt} - \pi_j$
  - idle time arcs, for all t, $t \rightarrow t + 1$ and cost 0
- a path in this network corrsponds to a pseudo-schedule in which a job may be started more than once or not processed.
- the lower bound on the master problem produced by the LP-relaxation of the restricted master problem can be tighten by inequalities

[Pessoa, Uchoa, Poggi de Aragão, Rodrigues, 2008], propose another time index formulation that dominates this one.
They can solve consistently instances up to 100 jobs.