DM872

Math Optimization at Work

Dantzig-Wolfe Decomposition and Delayed Column Generation

Marco Chiarandini

Department of Mathematics & Computer Science
University of Southern Denmark

[*Partly based on slides by David Pisinger, DIKU (now DTU)*]

Outline

Outline

1. Dantzig-Wolfe Decomposition
2. Solving the LP Master Problem

Dantzig-Wolfe Decomposition

Motivation: Large difficult IP models

$\Longrightarrow$ split them up into smaller pieces

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

Leads to methods also known as:

- Branch-and-price (column generation + branch and bound)
- Branch-and-cut-and-price (column generation + branch and bound + cutting planes)

Dantzig-Wolfe Decomposition

From an orginal or compact formulation to an extensive formulation made of 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.

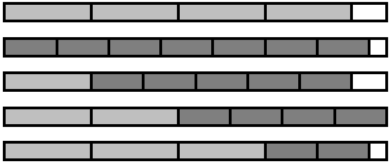
Motivation: Cutting stock problem

- 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

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

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:

	x[0, 0]	x[0, 1]	u[0]	x[1, 0]	x[1, 1]	u[1]	x[2, 0]	x[2, 1]	u[2]	x[3, 0]	x[3, 1]	u[3]	x[4, 0]	x[4, 1]	u[4]		
Minimize			u[0]			+u[1]			+u[2]			+u[3]			+u[4]		
stock[0]:	5x[0, 0]	+3x[0, 1]	+22u[0]													≤	0
stock[1]:				5x[1, 0]	+3x[1, 1]	−22u[1]										≤	0
stock[2]:							5x[2, 0]	+3x[2, 1]	−22u[2]							≤	0
stock[3]:										5x[3, 0]	+3x[3, 1]	−22u[3]				≤	0
stock[4]:													5x[4, 0]	+3x[4, 1]	−22u[4]	≤	0
type[0]:	x[0, 0]			+x[1, 0]			+x[2, 0]			+x[3, 0]			+x[4, 0]			≥	7
type[1]:		x[0, 1]			+x[1, 1]			+x[2, 1]			+x[3, 1]			+x[4, 1]		≥	3

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

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

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 Approach: Lagrangian Approach

Integer Programming Problem with block structure:

$$
\begin{array}{rlllllll}
z_{IP} = \max & c^1x^1 & + & c^2x^2 & + & \ldots + & c^Kx^K & \\
& A^1x^1 & + & A^2x^2 & + & \ldots + & A^Kx^K & = b \\
& D^1x^1 & & & & & & \leq d_1 \\
& & & D^2x^2 & & & & \leq d_2 \\
& & & & & \ldots & & \leq \vdots \\
& & & & & & D^Kx^K & \leq d_K \\
& x^1 \in \mathbb{Z}_+^{n_1}, & & x^2 \in \mathbb{Z}_+^{n_2}, & & \ldots, & x^K \in \mathbb{Z}_+^{n_K} &
\end{array}
$$

Lagrangian relaxation, multipliers $\lambda \in \mathbb{R}^K$
Objective becomes: $\max c^1x^1 + c^2x^2 + \ldots + c^Kx^K - \lambda(A^1x^1 + A^2x^2 + \ldots + A^Kx^K - b)$

$$
\begin{array}{rlllllll}
z_{LR}(\lambda) = \max & c^1x^1 - \lambda A^1x^1 & + & c^2x^2 - \lambda A^2x^2 & + & \ldots + & c^Kx^K - \lambda A^Kx^K & +b \\
& D^1x^1 & & & & & & \leq d_1 \\
& & & D^2x^2 & & & & \leq d_2 \\
& & & & & \ldots & & \leq \vdots \\
& & & & & & D^Kx^K & \leq d_K \\
& x^1 \in \mathbb{Z}_+^{n_1}, & & x^2 \in \mathbb{Z}_+^{n_2}, & & \ldots, & x^K \in \mathbb{Z}_+^{n_K} &
\end{array}
$$

model is separable

Strength of the Lagrangian Relaxation

General result

Integer Programming Problem:

$$
\begin{aligned}
z_{IP} = \max \ & cx \\
& Ax \leq b \\
& Dx \leq d \\
& x_j \in \mathbb{Z}_+ \ i = 1, \ldots, n
\end{aligned}
$$

Lagrangian relaxation, multipliers $\lambda \geq 0$

$$
\begin{aligned}
z_{LR}(\lambda) = \max \ & cx - \lambda(Ax - b) \\
& Dx \leq d \\
& x_j \in \mathbb{Z}_+ \ i = 1, \ldots, n
\end{aligned}
$$

for the best multiplier λ (from the Lagrangian Dual problem)

$$\boxed{z_{LD} = \max \{cx \mid Ax \leq b, x \in \text{conv}(Dx \leq d, x \in \mathbb{Z}_+)\}}$$

$z_{IP} \leq z_{LD} \leq z_{LP}$ hence z_{LD} is a better bound than z_{LP} from the linear relaxation of IP.

Dantzig-Wolfe decomposition

If model has "block" structure

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

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

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

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

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*

$$\max c^1(\sum_{t\in T_1}\lambda_{1,t}x^{1,t}) + c^2(\sum_{t\in T_2}\lambda_{2,t}x^{2,t}) + \ldots + c^K(\sum_{t\in T_K}\lambda_{K,t}x^{K,t})$$

$$\text{s.t.}\quad A^1(\sum_{t\in T_1}\lambda_{1,t}x^{1,t}) + A^2(\sum_{t\in T_2}\lambda_{2,t}x^{2,t}) + \ldots + A^K(\sum_{t\in T_K}\lambda_{K,t}x^{K,t}) = b$$

$$\sum_{t\in T_k}\lambda_{k,t} = 1 \qquad k = 1,\ldots,K$$

$$\lambda_{k,t} \in \{0,1\}, \qquad t \in T_k \;\; k = 1,\ldots,K$$

Strength of linear master model

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

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

Proof: Consider LP-relaxation

$$\max c^1(\sum_{t\in T_1}\lambda_{1,t}x^{1,t})+c^2(\sum_{t\in T_2}\lambda_{2,t}x^{2,t})+\ldots+c^K(\sum_{t\in T_K}\lambda_{K,t}x^{K,t})$$

$$\text{s.t.}\ \ A^1(\sum_{t\in T_1}\lambda_{1,t}x^{1,t})+A^2(\sum_{t\in T_2}\lambda_{2,t}x^{2,t})+\ldots+A^K(\sum_{t\in T_K}\lambda_{K,t}x^{K,t})=b$$

$$\sum_{t\in T_k}\lambda_{k,t}=1 \qquad\qquad k=1,\ldots,K$$

$$\lambda_{k,t}\geq 0, \qquad t\in T_k \qquad k=1,\ldots,K$$

Informally speaking we have

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

Theorem

- z_{LMP} *be the LP-solution value of the master problem*
- z_{LD} *be solution value of Lagrangian dual problem*

$$z_{LMP} = z_{LD}$$

Proof: as a consequence of the previous five slides the linear relaxation of the master problem and the Lagrangian dual correspond to solving the following problem:

$$\begin{array}{llllllll} \max & c^1x^1 & + & c^2x^2 & + & \dots + & c^Kx^K & \\ & A^1x^1 & + & A^2x^2 & + & \dots + & A^Kx^K & = b \\ & \multicolumn{7}{l}{x^1 \in \text{conv}(X^1), x^2 \in \text{conv}(X^2), \dots, x^K \in \text{conv}(X^K)} \end{array}$$

Hence, also the DW decomposition leads to a better dual bound than the linear relaxation of the original problem

$$z_{IP} \leq z_{LMP} = z_{LD} \leq z_{LP} \qquad \text{(for a maximization problem)}$$

Outline

1. Dantzig-Wolfe Decomposition

2. Solving the LP Master Problem

Delayed Column Generation

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

Solve the linear relaxation of the master problem by delayed column generation

Consider the general linear program

$$\begin{array}{lrcl} \text{minimize} & c^T x & & \\ \text{subject to} & Ax & = & b, \\ & x & \geq & 0, \end{array} \tag{3}$$

with $A \in \Re^{m \times n}$, $c \in \Re^n$, $b \in \Re^m$. The dual of (3) is

$$\begin{array}{lrcl} \text{maximize} & b^T y & & \\ \text{subject to} & A^T y & \leq & c. \end{array} \tag{4}$$

The sifting procedure begins by taking a "working set" of columns $\mathcal{W} \subset \{1, \ldots, n\}$ such that

$$\begin{array}{lrcl} \text{minimize} & c_{\mathcal{W}}^T x_{\mathcal{W}} & & \\ \text{subject to} & A_{\mathcal{W}} x_{\mathcal{W}} & = & b, \\ & x_{\mathcal{W}} & \geq & 0, \end{array} \tag{5}$$

is feasible. (This assumption is not essential.) Let π^* be an optimal solution to

$$\begin{array}{lrcl} \text{maximize} & b^T \pi & & \\ \text{subject to} & A_{\mathcal{W}}^T \pi & \leq & c_{\mathcal{W}}, \end{array} \tag{6}$$

the dual of (5), and let $x_{\mathcal{W}}^*$ be an optimal solution of (5). Then the vector $x^T = ((x_{\mathcal{W}}^*)^T, 0) \in \Re^n$ is optimal for (3) if

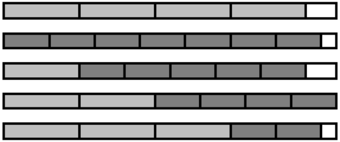
$$c - A^T \pi^* \geq 0. \tag{7}$$

Given the linear program (3) and a set $\mathcal{W}$ such that (5) is feasible:
Solve (5) obtaining x^* and π^*.
while $(c - A^T\pi^* \ngeq 0)$ **do** (major iteration)
    Choose $\mathcal{P} \subset \{1, \ldots, n\} \setminus \mathcal{W}$. (price)
    Set $\mathcal{W} \leftarrow \mathcal{W} \cup \mathcal{P}$. (augment problem)
    (Optionally) If $\mathcal{W}$ is too big,
        reduce the size of $\mathcal{W}$. (purge)
    Solve (5) obtaining x^* and π^*. (solve)
end while

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

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

Revised Simplex Method

- $\max\{cx \mid Ax \leq b, x \geq 0\}$
- $B = \{1 \dots m\}$ basic variables
- $N = \{m+1 \dots m+n\}$ non-basic variables (will be set to lower bound 0)
- $A_B = [A_1 \dots A_m]$
- $A_N = [A_{m+1} \dots A_{m+n}]$

Standard form

$$\left[\begin{array}{c:c:c:c} A_B & A_N & 0 & b \\ \hdashline c_B & c_N & 1 & 0 \end{array}\right]$$

$$\begin{aligned} Ax &= A_N x_N + A_B x_B = b \\ A_B x_B &= b - A_N x_N \\ x_B &= A_B^{-1} b - A_B^{-1} A_N x_N \end{aligned}$$

basic feasible solution:

- $X_N = 0$
- A_B lin. indep.
- $X_B \geq 0$

$$\begin{aligned} z = c^T x &= c_B^T (A_B^{-1} b - A_B^{-1} A_N x_N) + c_N^T x_N = \\ &= c_B^T A_B^{-1} b + (c_N^T - c_B^T A_B^{-1} A_N) x_N \end{aligned}$$

Canonical form

$$\left[\begin{array}{c|c|c|c} I & A_B^{-1} A_N & 0 & A_B^{-1} b \\ \hline 0 & c_N^T - c_B^T A_B^{-1} A_N & 1 & -c_B^T A_B^{-1} b \end{array}\right]$$

In scalar form: the objective function is obtained by multiplying and subtracting constraints by means of multipliers π: $\pi = c_B^T A_B^{-1}$ (the dual variables)

$$z = \sum_{j=1}^{m} \left[c_j + \sum_{i=1}^{m} \pi_i a_{ij} \right] x_j + \sum_{j=m+1}^{m+n} \left[c_j + \sum_{i=1}^{m} \pi_i a_{ij} \right] x_j + \sum_{i=1}^{m} \pi_i b_i$$

Each basic variable has cost null in the objective function

$$c_j + \sum_{i=1}^{m} \pi_i a_{ij} = 0 \qquad j = 1, \ldots, m$$

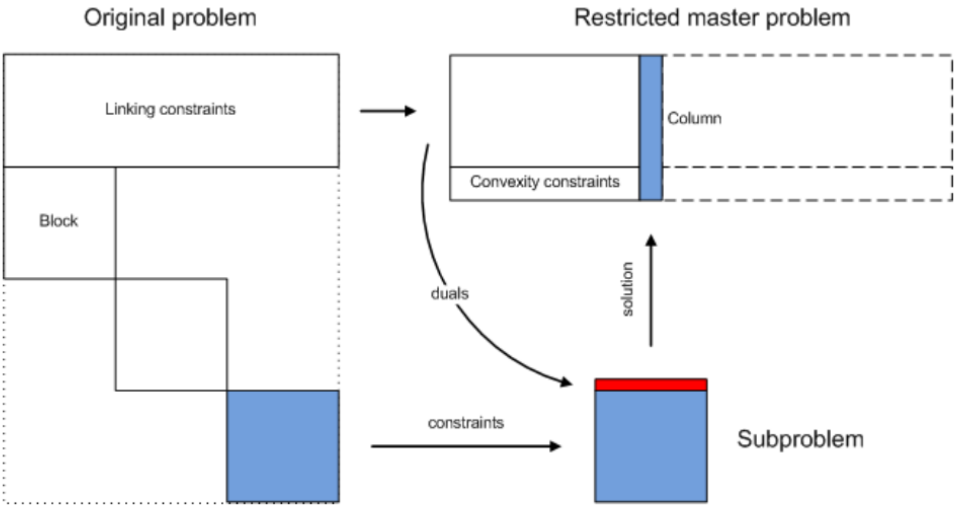
Reduced costs of non-basic variables:

$$\bar{c}_j = c_j + \sum_{i=1}^{m} \pi_i a_{ij} \qquad j = m+1, \ldots, m+n$$

If basis is optimal then $\bar{c}_j \leq 0$ for all $j = m+1, \ldots, m+n$.

Note: (multipliers) $\pi = -y_i$ (dual variables)

Dantzig Wolfe Decomposition with Column Generation



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

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

We could also solve optimization problem

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

which is equivalent to knapsack problem

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

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

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

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, assuming the simplex is not cycling?

  No, since the objective function is improving. We know the best solution among existing columns. If we generate an already existing column, then we will not improve the objective. However, we may be in the case of degenerate tableaux.

Consider the following LP problem

$$\begin{aligned} \text{maximize} \quad & x_1 + 2x_2 + 4x_3 + 8x_4 + 16x_5 \\ \text{st} \quad & x_1 + 2x_2 + 3x_3 + 4x_4 + 5x_5 \leq 5 \\ & 7x_1 + 5x_2 - 3x_3 - 2x_4 \leq 0 \\ & x_1, x_2, x_3, x_4, x_5 \geq 0 \end{aligned}$$

Let x_6, x_7 be the two slack variables.

- $[x_3, x_5]$ and $[x_5, x_7]$ are both optimal bases and they both give the same optimal solution
- the tableau is degenerate, that is, a $\bar{b}$ term is null, that is, one of the basis variables has value 0.
- we may have different tableaux for the same vertex, and hence that a tableau implies a unique vertex (but a vertex does not imply a unique tableau).
- the optimal dual solutions are two different (we need different multipliers): $[3.21.87], [3.2, 0]$, respectively.

Tailing off effect

Column generation may converge slowly in the end

- We do not need exact solution, just lower bound
- Solving master problem for subset of columns does not give valid lower bound (why?)
- Instead we may use Lagrangian relaxation of joint constraint
- "guess" Lagrangian multipliers equal to dual variables from master problem

Valid dual bounds in delayed CG

Linear relaxation of the reduced master problem:

$$z_{LRMP} = \max\left\{c\lambda \mid \bar{A}\lambda \leq b, \lambda \geq 0\right\}$$

Note: $z_{LRMP} \ngeq z_{LMP}$ (LMP Lin. relax. master problem)

However, during colum generation we have access to a dual bound so that we can terminate the process when a desired solution quality is reached.

When we know that

$$\sum_{j \in J} \lambda_j \leq \kappa \qquad J \text{ is the unrestricted set of columns}$$

for an optimal solution of the master, we cannot improve z_{RMP} by more than κ times the largest reduced cost obtained by the Pricing Problem (PP):

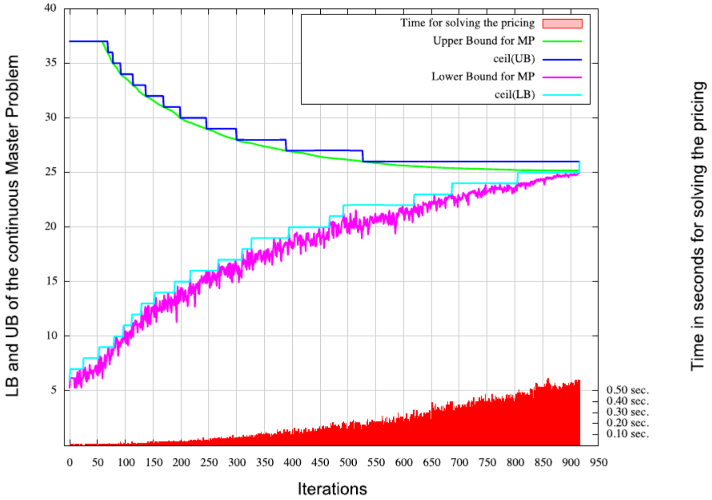
$$z_{LRMP} + \kappa z_{PP} \geq z_{LMP}$$

(It can be shown that this bound coincides with the Lagrangian dual bound.)

- with convexity constraints $\sum_{j \in J} \lambda_j \leq 1$ then $\kappa = 1$
- when $\mathbf{c} = 1$ we can set $\kappa = z_{LMP}$ and derive the better dual bound $\frac{z_{LRMP}}{1 - z_{PP}} \geq z_{LMP}$

Convergence in CG

In general the dual bound is not monotone during the iterations, for a problem of minimum:



[plot by Stefano Gualandi, Milan University]

Mixed Integer Linear Programs

- The primary use of column generation is in this context
  (in LP simplex is better)
- column generation re-formulations often give much stronger bounds than the original LP relaxation
- Often column generation referred to as branch-and-price

Branch-and-Price
Terminology

- Master Problem
- Restricted Master Problem
- Subproblem or Pricing Problem
- Branch and cut:
  Branch-and-bound algorithm using cuts to strengthen bounds.
- Branch and price:
  Branch-and-bound algorithm using column generation to derive bounds.

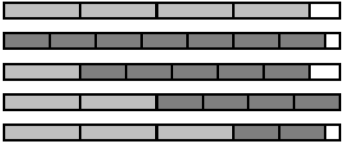
Branch-and-price

- LP-solution of master problem may have fractional solutions
- Branch-and-bound for getting IP-solution
- In each node solve LP-relaxation of master
- Subproblem may change when we add constraints to master problem
- Branching strategy should make subproblem easy to solve

Branch-and-price, example

The matrix A contains all different cutting patterns

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



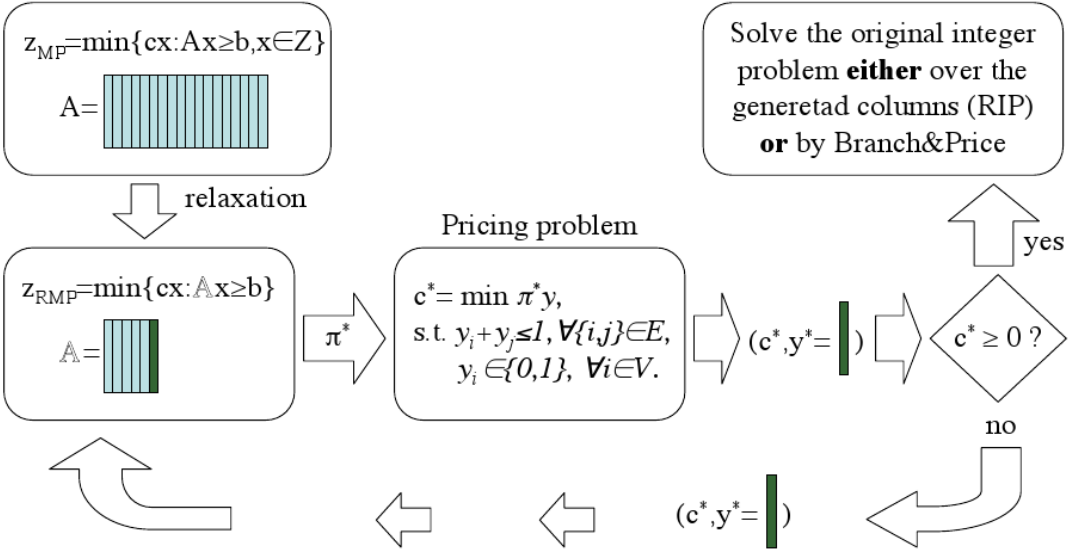
Problem

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

LP-solution $\lambda_1 = 1.375, \lambda_4 = 0.75$

Branch on $\lambda_1 = 0$, $\lambda_1 = 1$, $\lambda_1 = 2$

- Column generation may not generate pattern (4,0)
- Pricing problem is knapsack problem with pattern forbidden


z_MP=min{cx:Ax≥b,x∈Z}
A=
relaxation
z_RMP=min{cx:𝔸x≥b}
𝔸=
π*
Pricing problem
c*= min π*y,
s.t. y_i+y_j≤1, ∀{i,j}∈E,
y_i ∈{0,1}, ∀i∈V.
(c*,y*=)
c* ≥ 0 ?
yes
no
Solve the original integer problem either over the generetad columns (RIP) or by Branch&Price
(c*,y*=)

Heuristic solution (eg, in sec. 12.6)

- Restricted master problem will only contain a subset of the columns
- We may solve restricted master problem to IP-optimality
- Restricted master is a "set-covering-like" problem which is not too difficult to solve