Friday, December 10

Program of the day:

- TSP konkurrence
- Delayed column generation — more formally
- Branch-and-price algorithms
- Example: VRP problems solved through delayed column generation
- Evaluation
Dantzig-Wolfe Decomposition

The problem is split into a master problem and a subproblem

- Tighter bounds
- Better controle 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.
Decomposition

If model has “block” structure

\[
\begin{align*}
\text{max} \quad & c^1 x^1 + c^2 x^2 + \ldots + c^K x^K \\
\text{s.t.} \quad & A^1 x^1 + A^2 x^2 + \ldots + A^K x^K = b \\
& D^1 x^1 + D^2 x^2 \\
& \quad \quad \quad \quad \quad \quad \vdots \\
& \quad \quad \quad \quad \quad \quad D^K x^K \leq d_K \\
& x^1 \in \mathbb{Z}_{+}^{n_1} \quad x^2 \in \mathbb{Z}_{+}^{n_2} \quad \ldots \quad x^K \in \mathbb{Z}_{+}^{n_K}
\end{align*}
\]

Lagrangian relaxation

Objective becomes

\[
\begin{align*}
& c^1 x^1 + c^2 x^2 + \ldots + c^K x^K \\
& -\lambda \left( A^1 x^1 + A^2 x^2 + \ldots + A^K x^K - b \right)
\end{align*}
\]

Decomposed into

\[
\begin{align*}
\text{max} \quad & c^1 x^1 - \lambda A^1 x^1 + c^2 x^2 - \lambda A^2 x^2 + \ldots + c^K x^K - \lambda A^K x^K + b \\
\text{s.t.} \quad & D^1 x^1 + D^2 x^2 \\
& \quad \quad \quad \quad \quad \quad \vdots \\
& \quad \quad \quad \quad \quad \quad D^K x^K \leq d_K \\
& x^1 \in \mathbb{Z}_{+}^{n_1} \quad x^2 \in \mathbb{Z}_{+}^{n_2} \quad \ldots \quad x^K \in \mathbb{Z}_{+}^{n_K}
\end{align*}
\]

Model is separable
Dantzig-Wolfe decomposition

If model has “block” structure

\[
\begin{align*}
\text{max} & \quad c^1 x^1 + c^2 x^2 + \ldots + c^K x^K \\
\text{s.t.} & \quad A^1 x^1 + A^2 x^2 + \ldots + A^K x^K = b \\
& \quad D^1 x^1 + \leq d_1 \\
& \quad + D^2 x^2 \leq d_2 \\
& \quad \ldots \leq \vdots \\
& \quad D^K x^K \leq d_K \\
x^1 \in \mathbb{Z}^{n_1}_+ \quad x^2 \in \mathbb{Z}^{n_2}_+ \quad \ldots \quad x^K \in \mathbb{Z}^{n_K}_+
\end{align*}
\]

Substituting each set \( X^k, k = 1, \ldots, K \) in original model getting Master Problem

\[
\begin{align*}
\text{max} & \quad c^1(\sum_{t \in T_1} \lambda_{1,t} x^{1,t}) + c^2(\sum_{t \in T_2} \lambda_{2,t} x^{2,t}) + \ldots + c^K(\sum_{t \in T_K} \lambda_{K,t} x^{K,t}) \\
\text{s.t.} & \quad A^1(\sum_{t \in T_1} \lambda_{1,t} x^{1,t}) + A^2(\sum_{t \in T_2} \lambda_{2,t} x^{2,t}) + \ldots + A^K(\sum_{t \in T_K} \lambda_{K,t} x^{K,t}) = b \\
& \quad \sum_{t \in T_k} \lambda_{k,t} = 1 \quad k = 1, \ldots, K \\
& \quad \lambda_{k,t} \in \{0, 1\}, \quad t \in T_k \quad k = 1, \ldots, K
\end{align*}
\]
Strength of linear master model

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

\[
\begin{align*}
\text{max} & \quad c^1x^1 + c^2x^2 + \ldots + c^kx^k \\
\text{s.t.} & \quad A^1x^1 + A^2x^2 + \ldots + A^kx^k = b \\
& \quad x^1 \in \text{conv}(X^1), \quad x^2 \in \text{conv}(X^2), \quad \ldots \quad x^k \in \text{conv}(X^k)
\end{align*}
\]

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)

\[ z^{LPM} = z^{LD} \]
Delayed column generation, linear master

(minimization problem)

Run Simplex algorithm as if complete master problem was known

- Start with a basis solution
- Solve
  \[ x_B = A_B^{-1} b \]
  and find dual variables
  \[ y = c_B A_B^{-1} \]
- When choosing entering variable solve pricing problem which minimizes reduced costs
  \[ c_j^r = c_j - yA_j \]
- If \( c_j^r < 0 \) add corresponding column \( A_j \) to model and repeat
- If \( c_j^r \geq 0 \) stop
Cutting Stock Problem

*(minimization problem)*

\(a_{ij}\) is number of pieces of type \(i\) cut from pattern \(j\)

Master problem,

\[
\begin{align*}
\text{min} & \quad \sum_{j=1}^{n} u_j \\
\text{s.t.} & \quad \sum_{j=1}^{n} a_{ij} u_j \geq b_i \quad i = 1, \ldots, m, \quad j = 1, \ldots, n \\
& \quad u_j \in \mathbb{Z}_+
\end{align*}
\]

Solving linear master through delayed column generation

- Start with patterns which only contain one type \(i\)
- Solve restricted master
- Dual variables \(y_i\) say how “attractive” a type \(i\) is
- Pricing problem

\[
z^S = \min \quad 1 - \sum_{i=1}^{m} y_i x_i
\]

\[
\text{s.t.} \quad \sum_{i=1}^{m} w_i x_i \leq L
\]

\[x \geq 0, \text{integer}\]

- stop if \(z^S \geq 0\)
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.

- One says that discarded columns are “priced out”.
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

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

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
Branch-and-price, example

Better branching strategy

- Branch 1: item $i$ and item $j$ are cut from *same* pattern
- Branch 2: item $i$ and item $j$ are cut from *different* patterns

Pricing problem

- Branch 1: “glue together” the two items
- Branch 2: multiple-choice knapsack problem

Will the branching terminate?

- Sooner or later we will have defined exactly which items go into same pattern, i.e. all patterns
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

Heuristic solution

- 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
Vehicle Routing Problem with Time Windows

A delivery company shall visit 5 customers in some pre-described time-intervals \([a_i, b_i]\) as follows:

<table>
<thead>
<tr>
<th>customer (i)</th>
<th>(a_i)</th>
<th>(b_i)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>

The depot is considered as node 0 and 6. The travel time \(t_{ij}\) between customer \(i\) and \(j\), and corresponding cost \(c_{ij}\) is

\[
c_{ij} = t_{ij} = |j - i| \quad i = 1, \ldots, 6, \ j = 0, \ldots, 6
\]

There are at most five vehicles available, and the cost of a vehicle is 6 units, i.e. \(c_{0j} = 6\). The travel time from the depot to the first customer is \(t_{0j} = 1\).

1. Find a feasible initial solution to the problem.
2. Use column generation to solve the problem to LP-optimality.
3. Write up all feasible routes, and solve the problem using CPLEX to integer optimality.
4. Compare the LP solution with the integer solution.
Graph

\[ c_{ij} = t_{ij} = |j - i| \quad c_{0j} = 6, \; t_{0j} = 1 \]

Possible tours (in lexicographic order, dominated removed)

<table>
<thead>
<tr>
<th>tour</th>
<th>cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>0,1,2,4,6</td>
<td>11</td>
</tr>
<tr>
<td>0,(wait),2,4,6</td>
<td>10</td>
</tr>
<tr>
<td>0,3,2,4,6</td>
<td>11</td>
</tr>
<tr>
<td>0,3,(wait),4,6</td>
<td>10</td>
</tr>
<tr>
<td>0,3,5,6</td>
<td>9</td>
</tr>
<tr>
<td>0,3,5,4,6</td>
<td>11</td>
</tr>
<tr>
<td>0,(wait),(wait),4,6</td>
<td>8</td>
</tr>
<tr>
<td>0,(wait),5,4,6</td>
<td>10</td>
</tr>
<tr>
<td>0,(wait),5,6</td>
<td>7</td>
</tr>
</tbody>
</table>
Column Generation

- **Initial Solution.** We can choose the trivial routes:

<table>
<thead>
<tr>
<th>tour</th>
<th>cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>0,1,6</td>
<td>11</td>
</tr>
<tr>
<td>0,2,6</td>
<td>10</td>
</tr>
<tr>
<td>0,3,6</td>
<td>9</td>
</tr>
<tr>
<td>0,4,6</td>
<td>8</td>
</tr>
<tr>
<td>0,5,6</td>
<td>7</td>
</tr>
</tbody>
</table>

which gives a solution of 45.

- **Column generation.** We start with formulation

\[
\begin{align*}
\min & \quad 11x_1 + 10x_2 + 9x_3 + 8x_4 + 7x_5 \\
\text{s.t.} & \quad x_1 \geq 1 \\
& \quad x_2 \geq 1 \\
& \quad x_3 \geq 1 \\
& \quad x_4 \geq 1 \\
& \quad x_5 \geq 1
\end{align*}
\]

the dual variables corresponding to the constraints are

\[
y_1 = 11, \quad y_2 = 10, \quad y_3 = 9, \quad y_4 = 8, \quad y_5 = 7
\]

The reduced cost of a tour is now the “original” cost of the edges minus the dual variables corresponding to the customers visited. The most beneficial tour is \((0,1,2,4,6)\) having the cost \(11 - (11 + 10 + 8) = -18\). Since the reduced cost of the new column is
negative, we add it to the problem, getting:

\[
\begin{align*}
\text{min} & \quad 11x_1 + 10x_2 + 9x_3 + 8x_4 + 7x_5 + 11x_6 \\
\text{s.t.} & \quad x_1 + x_6 \geq 1 \\
& \quad x_2 + x_6 \geq 1 \\
& \quad x_3 + x_6 \geq 1 \\
& \quad x_4 + x_7 \geq 1 \\
& \quad x_5 + x_7 \geq 1
\end{align*}
\]

the dual variables are

\[y_1 = 0, \quad y_2 = 3, \quad y_3 = 9, \quad y_4 = 8, \quad y_5 = 7\]

The most beneficial tour is \((0, 3, 5, 4, 6)\) having the reduced cost \(11 - (9 + 8 + 7) = -13\), which is added to the problem:

\[
\begin{align*}
\text{min} & \quad 11x_1 + 10x_2 + 9x_3 + 8x_4 + 7x_5 + 11x_6 + 11x_7 \\
\text{s.t.} & \quad x_1 + x_6 \geq 1 \\
& \quad x_2 + x_6 \geq 1 \\
& \quad x_3 + x_7 \geq 1 \\
& \quad x_4 + x_7 \geq 1 \\
& \quad x_5 + x_7 \geq 1
\end{align*}
\]

the dual variables are

\[y_1 = 1, \quad y_2 = 10, \quad y_3 = 4, \quad y_4 = 0, \quad y_5 = 7\]

Adding the tour \((0, 3, 2, 4, 6)\) at reduced cost \(11 - (10 + 4 + 0) = -3\) gives

\[
\begin{align*}
\text{min} & \quad 11x_1 + 10x_2 + 9x_3 + 8x_4 + 7x_5 + 11x_6 + 11x_7 + 11x_8 \\
\text{s.t.} & \quad x_1 + x_6 \geq 1 \\
& \quad x_2 + x_6 \geq 1 \\
& \quad x_3 + x_7 \geq 1 \\
& \quad x_4 + x_7 \geq 1 \\
& \quad x_5 + x_7 \geq 1
\end{align*}
\]
with dual variables

\[ y_1 = 4, \quad y_2 = 7, \quad y_3 = 4, \quad y_4 = 0, \quad y_5 = 7 \]

The most beneficial tour is \((0, 3, 5, 6)\), having reduced cost \(9 - (4 + 7) = -2\). Thus

\[
\begin{align*}
\min & \quad 11x_1 + 10x_2 + 9x_3 + 8x_4 + 7x_5 + 11x_6 + 11x_7 + 11x_8 + 9x_9 \\
\text{s.t.} & \quad x_1 + x_6 + x_7 + x_8 + x_9 \geq 1 \\
& \quad x_2 + x_6 + x_7 + x_8 + x_9 \geq 1 \\
& \quad x_3 + x_6 + x_7 + x_8 + x_9 \geq 1 \\
& \quad x_4 + x_6 + x_7 + x_8 + x_9 \geq 1 \\
& \quad x_5 + x_7 + x_9 \geq 1
\end{align*}
\]

with dual variables

\[ y_1 = 2, \quad y_2 = 9, \quad y_3 = 2, \quad y_4 = 0, \quad y_5 = 7 \]

Now, all tours have reduced costs \(\geq 0\), so we are done. The bound found by column generation is thus \(z_{LP} = 20\).

- **Write up all feasible solutions.** The problem with all 9 tours added to the formulation, solved to IP-optimality, gives the solution \(z_{IP} = 20\).

- **Compare bounds.** We have \(z_{IP} = z_{LP}\) so the bounds are tight.