A 0-1 Linear Programming Formulation for the Berth Assignment Problem.

Serigne Gueye (1), Sophie Michel (2), Adnan Yassine (2).

(1) LIA - Université d’Avignon  
(2) ISEL - LMAH - Université du Havre.

Logistiqua 2011 - May 31
1 CONTEXT and PROBLEM
2 MODEL derived from QAP
3 LINEARISATION and VALID INEQUALITIES
4 ALGORITHM and NUMERICAL RESULTS
5 PERSPECTIVES
Le Havre, 2011

BERTH ALLOCATION PROBLEM [*]

Problem

Find when and where a ship has to assigned in order to minimize the sum of waiting times and the transhipment costs.

Assumptions

• **Sections** of equal size, but the number of sections is unsufficient for all ships.

• **n container ships** with:
  - a number of sections occupied by the ship,
  - an arrival date, a processing time and a weight (importance),
  - the container flow matrix between ships.

Problem
Find where a ship has to assigned in order to minimize the transhipment costs.

Assumptions
• Sections of equal size s, but the number of sections is insufficient for all ships.
• \( n \) container ships with:
  • a number of sections occupied by the ship \( i \), \( b_i \),
  • the container flow matrix between ships \( i \) and \( j \), \( F = (f_{ij})_{1 \leq i, j \leq n} \).
EXAMPLE with $n = 3$, $s = 50m$

- $b = (3, 1, 2)$ and $F = \begin{pmatrix} 0 & 10 & 20 \\ 8 & 0 & 2 \\ 10 & 10 & 0 \end{pmatrix}$

- An assignment $\pi_1$ with distance matrix: $D(\pi_1) = \begin{pmatrix} 0 & 100 & 175 \\ 100 & 0 & 75 \\ 175 & 75 & 0 \end{pmatrix}$

and cost: $C(\pi_1) = \sum_{i=1}^{3} \sum_{j=1}^{3} f_{ij} d_{\pi_1(i)\pi_1(j)}(\pi_1) =$

$(10 + 8) \times 100 + (20 + 10) \times 175 + (10 + 2) \times 75 = 7950$ meters
EXAMPLE with \( n = 3, \ s = 50 m \)

- An assignment \( \pi_2 \) with distance matrix: 
  
  \[
  D(\pi_2) = \begin{pmatrix}
  0 & 125 & 200 \\
  125 & 0 & 75 \\
  200 & 75 & 0
  \end{pmatrix}
  \]

and cost:

\[
C(\pi_2) = (10 + 8) \times 200 + (20 + 10) \times 125 + (10 + 2) \times 75 = 8250 m.
\]

**OBJECTIVE:** FIND THE MINIMAL ASSIGNMENT COST.
The Berth Assignment Problem is an NP-Complete problem

Garey & Johnson, Computers and Intractibility.

The special case where:

$$b_i = 1 \forall i = 1, \cdots, n$$

and

$$f_{ij} = f_{ji} \in \{0, 1\} \forall i, j = 1, \cdots, n$$

is NP-Complete (optimal linear arrangement problem).
FORMULATION

Let

\[ x_{ik} = \begin{cases} 
1 & \text{if the ship } i = 1, \cdots, n \text{ is assigned to location } k = 1, \cdots, n \\
0 & \text{otherwise}
\end{cases} \]

and the distance between locations \( k = 1, \cdots, n \) and \( l = 1, \cdots, n \):

\[
d_{kl}(x) = \begin{cases} 
\frac{1}{2} \sum_{m=1}^{n} b_m x_{mk} + \sum_{u=\min{k,l}+1}^{\max{k,l}-1} \sum_{m=1}^{n} b_m x_{mu} + \frac{1}{2} \sum_{m=1}^{n} b_m x_{ml} & \text{if } |k - l| \geq 2 \\
\frac{1}{2} \sum_{m=1}^{n} b_m x_{mk} + \frac{1}{2} \sum_{m=1}^{n} b_m x_{ml} & \text{if } |k - l| = 1 \\
0 & \text{if } k = l
\end{cases}
\]
**FORMULATION**

Lemma

*(BAP) is equivalent to :*

\[
\begin{align*}
\text{Min} & \quad \sum_{i,j,m=1}^{n} b_m f_{ij} \sum_{k,l=1}^{n} \left( \sum_{u=\min\{k,l\}+1}^{\max\{k,l\}-1} x_{mu} \right) x_{ik} x_{jl} \\
\text{s-t :} & \quad (1) \quad \sum_{i=1}^{n} x_{ik} = 1, \ \forall \ k = 1, \ldots, n \\
& \quad (2) \quad \sum_{k=1}^{n} x_{ik} = 1, \ \forall \ i = 1, \ldots, n \\
& \quad (3) \quad x \in \{0, 1\}^{n^2}
\end{align*}
\]
Objectif: \( \text{Min} \sum_{i,j,m=1}^{n} b_m f_{ij} \sum_{k,l=1}^{n} (\sum_{u=\min\{k,l\}+1}^{\max\{k,l\}-1} x_{mu}) x_{ik} x_{jl} \)

Let \( y_{kml} = \sum_{u=\min\{k,l\}+1}^{\max\{k,l\}-1} x_{mu} \) and \( t_{imj} = \sum_{k,l=1}^{n} y_{kml} x_{ik} x_{jl} \)

Theorem

- \( t_{imj} = \max_{1 \leq k,l \leq n} y_{kml} x_{ik} x_{jl} = \begin{cases} 1 & \text{if } m \text{ is between } i \text{ and } j \\ 0 & \text{otherwise} \end{cases} \)
- \( y_{kml} x_{ik} x_{jl} \geq y_{kml} + x_{ik} + x_{jl} - 2 \)
- \( y_{kml} x_{ik} x_{jl} \geq 0 \)
Lemma

\((\text{LBAP})\) \quad \text{Min} \quad \sum_{i,j,m=1}^{n} f_{ij} b_{m} t_{imj}

\text{s-t :} \quad (1-3) \quad \text{and} \quad (4)

\forall i,j,m = 1, \ldots, n, i \neq j, m \neq i,j

\begin{align*}
\max\{k,l\} - 1 & \sum_{u=\min\{k,l\}+1}^{\max\{k,l\}-1} x_{mu} + x_{ik} + x_{jl} - 2, \forall k,l, |k - l| \geq 2 \\
t_{imj} & \geq 0
\end{align*}

The linear relaxation of \((\text{LBAP})\) has an optimal value = 0 !

Proof: \(x_{ik} = \frac{1}{n}\) and \(t_{imj} = 0 \Box\).
Remark:  \( t_{ijm} = \begin{cases} 1 & \text{if } m \text{ is between } i \text{ and } j \\ 0 & \text{otherwise} \end{cases} \)

Theorem

(5) \( t_{ijm} + t_{mij} + t_{ijm} = 1 \) is valid for all \( 1 \leq i < m < j \leq n \).

Proof

- \( m \) is between \( i \) and \( j \) or
- \( i \) is between \( m \) and \( j \) or
- \( j \) is between \( m \) and \( i \). \(\square\)
EXAMPLE with $n=4$

- $b = (2, 3, 3, 3)$ and $F = \begin{pmatrix} 0 & 79 & 32 & 57 \\ 79 & 0 & 96 & 62 \\ 32 & 96 & 0 & 89 \\ 57 & 62 & 89 & 0 \end{pmatrix}$

- Optimal solution of the linear relaxation of (LBAP+(5)):
  - $x = \begin{bmatrix} 0.333333 & 0 & 0.666667 & 0 \\ 0.666667 & 0.333333 & 0 & 0 \\ 0 & 0.666667 & 0 & 0.333333 \\ 0 & 0 & 0.333333 & 0.666667 \end{bmatrix}$
  - $t_{imj} \neq 0$ are
    \[
    \begin{cases}
    t[0][1][2] = 1 \\
    t[0][3][2] = 1 \\
    t[1][0][3] = 1 \\
    t[1][2][3] = 1 
    \end{cases}
    \]

- LB = 1004 (gap: 24%)
CONSISTENCY, INCONSISTENCY

\[
t[0][1][2] = 1 \quad i.e \, 1 \, between \, 0 \, and \, 2 \\
t[0][3][2] = 1 \quad i.e \, 3 \, between \, 0 \, and \, 2 \quad \text{is inconsistent (i.e, impossible).} \\
t[1][0][3] = 1 \quad i.e \, 0 \, between \, 1 \, and \, 3
\]

⇒ Cut : \[ t[0][1][2] + t[0][3][2] + t[1][0][3] \leq 2. \]

Definition

Let \( S = \{(i_k, m_k, j_k), \, k = 1, ..., m\} \) be a set of 3-uplet of ships and \( A^a_S = \{t(i_k, m_k, j_k) = a_k, \, k = 1, 2, ..., m\} \) be an assignment \((a_k \in \{0, 1\})\).

\( A^a_S \) is consistent if a position for all ships can be found (else inconsistent)

Valid Inequalities: if \( A^a_S \) is inconsistent then the below cut is added:

\[
\sum_{k=1}^{m} [a_k t(i_k, m_k, j_k) + (1 - a_k)(1 - t(i_k, m_k, j_k))] \leq (m - 1)
\]

Separation: based on Constraint Programming Problem (Ilog Solver)
1. Solve the linear relaxation of (LBAP)\((5)\). Let \((x^*, t^*)\) the corresponding optimal value.

2. Choose a set \(S\) for which \(t^*_S\) is integer.

3. Derive the set \(A^{t^*_S}_S\).

4. Check consistency of \(A^{t^*_S}_S\) (by Constraint Programming).

5. If \(A^{t^*_S}_S\) is inconsistent then add the corresponding cut to \((LBAP)\) and return to step 1.

6. Else return to step 2 and choose another set \(S\).

\(\implies\) Solve the resulting formulation with Cplex
NUMERICAL RESULTS

Instances from the QAPLIB with $n = 7, \cdots, 15$:
- 18 inst. chr15 with $d = 19\%$ on average,
- 9 inst. eil19 with $d = 100\%$.

Time Limit : 30 mn

<table>
<thead>
<tr>
<th>Form.</th>
<th>lb</th>
<th>lb time</th>
<th>nb Opt</th>
<th>ub</th>
<th>gap</th>
</tr>
</thead>
<tbody>
<tr>
<td>chr15 - (5)</td>
<td>10568.6</td>
<td>16.2</td>
<td>6</td>
<td>11672.8</td>
<td>10.5</td>
</tr>
<tr>
<td>chr15 - (5)+cut</td>
<td>11444.8</td>
<td>188.2</td>
<td>14</td>
<td>11672.8</td>
<td>2</td>
</tr>
<tr>
<td>eil - (5)</td>
<td>288848.3</td>
<td>1.7</td>
<td>2</td>
<td>398835.9</td>
<td>38.1</td>
</tr>
<tr>
<td>eil - (5)+cut</td>
<td>346830.7</td>
<td>46.8</td>
<td>4</td>
<td>398835.9</td>
<td>15</td>
</tr>
</tbody>
</table>

Inst eil19.dat with $n = 8$:

<table>
<thead>
<tr>
<th>Form.</th>
<th>lb</th>
<th>lb time</th>
<th>Opt</th>
<th>gap</th>
<th>nodes</th>
<th>b&amp;b time</th>
</tr>
</thead>
<tbody>
<tr>
<td>(5)</td>
<td>55566.0</td>
<td>0.0</td>
<td>67128.0</td>
<td>17.2</td>
<td>21985</td>
<td>132.9</td>
</tr>
<tr>
<td>(5)+cut</td>
<td>63447.3</td>
<td>0.1</td>
<td>67128.0</td>
<td>5.5</td>
<td>21</td>
<td>12.3</td>
</tr>
</tbody>
</table>
• **Theory**: Are valid inequalities are facet?
• **Theory**: Is consistency problem an NP-Complete problem?
• **Theory**: Study others formulations (TSP)
• **Practical**: Integrate time dimension, limited resources,...