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

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

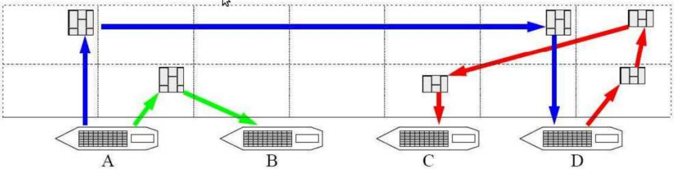
(1) LIA - Université d'Avignon
(2) ISEL - LMAH - Université du Havre.

Logistiqua 2011 - May 31

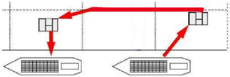
CONTEXT

Le Havre, 2011

Slides Vacca et al, Optimization of Container Terminal Operations, EPFL, 2007



BERTH ALLOCATION PROBLEM [*]



Problem

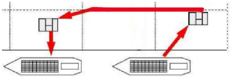
Find when and where a ship has to assigned in order to **minimize** the sum of waiting times *and* the transhipment costs.

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.

[*] Brown et al - 1994; Lim - 1998; Chen and Hsieh - 1999; Imai et al - 2003; Cordeau et al - 2005

BERTH ASSIGNMENT PROBLEM



Problem

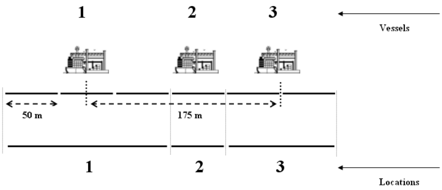
Find where a ship has to assigned in order to **minimize** the transhipment costs.

Assumptions

- **Sections** of equal size s, ~~but the number of sections is unsufficient 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 π_1 with distance matrix: $D(\pi_1) = \begin{pmatrix} 0 & 100 & 175 \\ 100 & 0 & 75 \\ 175 & 75 & 0 \end{pmatrix}$

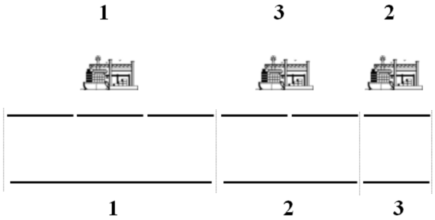


and cost: $C(\pi_1) = \sum_{i=1}^{3} \sum_{j=1}^{3} f_{ij} d_{\pi_1(i)\pi_1(j)}(\pi_1) =$

$$(10+8)*100 \;+\; (20+10)*175 \;+\; (10+2)*75 = 7950 \;\; meters$$

EXAMPLE with $n = 3$, $s = 50m$

- An assignment π_2 with distance matrix: $D(\pi_2) = \begin{pmatrix} 0 & 125 & 200 \\ 125 & 0 & 75 \\ 200 & 75 & 0 \end{pmatrix}$



and cost:
$C(\pi_2) = (10 + 8) * 200 + (20 + 10) * 125 + (10 + 2) * 75 = 8250m$.

OBJECTIVE : FIND THE MINIMIMAL ASSIGNMENT COST.

COMPLEXITY

Theorem

The Berth Assignment Problem is an NP-Complete problem

Garey & Johnson, Computers and Intractibility.

The special case where:

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

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 :

$$
\begin{array}{ll}
Min & \sum\limits_{i,j,m=1}^{n} b_m f_{ij} \sum\limits_{\substack{k,l=1\\|l-k|\geq 2}}^{n} \Big(\sum\limits_{u=min\{k,l\}+1}^{max\{k,l\}-1} x_{mu} \Big) x_{ik} x_{jl} \\
s\text{-}t : (1) & \sum\limits_{i=1}^{n} x_{ik} = 1, \ \forall\ k = 1, \cdots, n \\
(2) & \sum\limits_{k=1}^{n} x_{ik} = 1, \ \forall\ i = 1, \cdots, n \\
(3) & x \in \{0,1\}^{n^2}
\end{array}
$$

LINEARISATION

$$\text{Objectif : } Min \sum_{i,j,m=1}^{n} b_m f_{ij} \sum_{\substack{k,l=1\\|l-k|\geq 2}}^{n} \Big(\sum_{u=min\{k,l\}+1}^{max\{k,l\}-1} x_{mu} \Big) x_{ik} x_{jl}$$

Let $y_{kml} = \sum_{u=min\{k,l\}+1}^{max\{k,l\}-1} x_{mu}$ and $t_{imj} = \sum_{\substack{k,l=1\\|l-k|\geq 2}}^{n} y_{kml} x_{ik} x_{jl}$

Theorem

- $t_{imj} = \underset{1\leq k,l\leq n}{Max}\, y_{kml} x_{ik} x_{jl} = \begin{cases} 1 & \text{if } m \text{ is between } i \text{ and } j \\ 0 & \text{otherwise} \end{cases}$
- $\begin{array}{rcl} y_{kml} x_{ik} x_{jl} & \geq & y_{kml} + x_{ik} + x_{jl} - 2 \\ y_{kml} x_{ik} x_{jl} & \geq & 0 \end{array}$

LINEARISATION

Lemma

$$
\begin{array}{lll}
(LBAP) & Min & \sum_{i,j,m=1}^{n} f_{ij} b_m t_{imj} \\
s\text{-}t: & (1\text{-}3) & and \\
 & (4) & \forall\ i,j,m = 1,\cdots,n,\ i \neq j,\ m \neq i,j \\
 & & t_{imj} \geq \sum_{u=min\{k,l\}+1}^{max\{k,l\}-1} x_{mu} + x_{ik} + x_{jl} - 2,\ \forall k,l, |k-l| \geq 2 \\
 & & t_{imj} \geq 0
\end{array}
$$

The linear relaxation of (LBAP) has an optimal value = 0 !

Proof: $x_{ik} = \frac{1}{n}$ and $t_{imj} = 0$ $\square$.

VALID INEQUALITIES

Remark: $t_{imj} = \begin{cases} 1 & \textit{if } m \textit{ is between } i \textit{ and } j \\ 0 & \textit{otherwise} \end{cases}$

Theorem

(5) $t_{imj} + t_{mij} + t_{ijm} = 1$ *is valid forall* $1 \leq i < m < j \leq n$.

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 $\quad$ is inconsistant (i.e, impossible).
$t[1][0][3] = 1$ $\quad$ *i.e* 0 *between* 1 *and* 3

$\Rightarrow$ 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 $A_S^a = \{t_{(i_k,m_k,j_k)} = a_k, k = 1, 2, ..., m\}$ be an assignment ($a_k \in \{0, 1\}$).

A_S^a is consistant if a position for all ships can be found (else inconsistant)

Valid Inequalities: if A_S^a is inconsistant 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)

ALGORITHM

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 $A_S^{t_S^*}$.
4. Check consistency of $A_S^{t_S^*}$ (by Constraint Programming).
5. if $A_S^{t_S^*}$ is inconsistant then add the corresponding cut to $(LBAP)$ and return to step 1
6. else return to step 2 and choose another set S.

$\Longrightarrow$ **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

Form.	lb	lb time	nb Opt	ub	gap
chr15 - (5)	10568.6	16.2	6	11672.8	10.5
chr15 - (5)+cut	11444.8	188.2	14	11672.8	2
eil - (5)	288848.3	1.7	2	398835.9	38.1
eil - (5)+cut	346830.7	46.8	4	398835.9	15

Inst eil19.dat with $n = 8$:

Form.	lb	lb time	Opt	gap	nodes	b&b time
(5)	55566.0	0.0	67128.0	17.2	21985	132.9
(5)+cut	63447.3	0.1	67128.0	5.5	21	12.3

PERSPECTIVES

- Theory : Are valid inequalities are facet ?
- Theory : Is consistency problem an NP-Complete problem ?
- Theory : Study others formulations (TSP)
- Practical : Integrate time dimension, limited ressources,...