Decison-Aid Methodologies in Transportation

Optimization Exercise 4

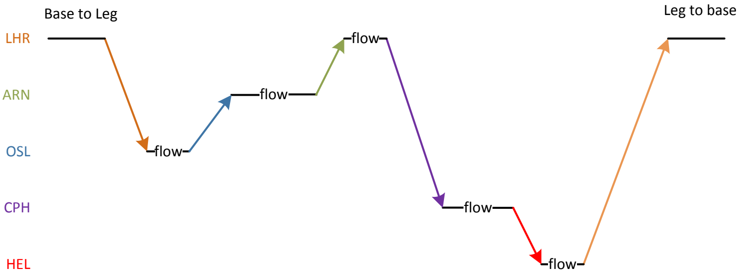
Tomáš Robenek

Transport and Mobility Laboratory
EPFL

April 29, 2014

Aircraft Rotation/Fleet Assignment Problem – Input

Aircraft Rotation Problem – Output



Model I

F	–	set of flight legs to be covered
K	–	set of fleet types
M^k	–	number of available aircraft of type k
c_i^k	–	operating cost minus the revenue of aircraft type k to flight leg i
N^k	–	set of nodes in the time-space network of aircraft type k
G^k	–	set of ground nodes in the time-space network of aircraft type k
$O(k,n)$	–	set of flight legs originating at node n in fleet k's time-space network
$I(k,n)$	–	set of flight legs terminating at node n in fleet k's time-space network
n^+	–	ground arc originating in node n
n^-	–	ground arc terminating in node n
$CL(k)$	–	set of flight legs of fleet k
$CG(k)$	–	set of ground arcs of fleet k

$$f_i^k = \begin{cases} 1 & \text{if and only if flight leg } i \text{ is to be operated with an aircraft type } k, \\ 0 & \text{otherwise.} \end{cases}$$

y_a^k – number of aircraft type k on the ground arc a

TRANSP-OR

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Model II

$$\max \quad \sum_{i\in F}\sum_{k\in K} c_i^k \cdot f_i^k \tag{1}$$

$$\text{s.t.} \quad \sum_{k\in K} f_i^k = 1, \qquad \forall i \in I, \tag{2}$$

$$y_{n^+}^k + \sum_{i\in O(k,n)} f_i^k - y_{n^-}^k - \sum_{i\in I(k,n)} f_i^k = 0, \qquad \forall n \in N^k, \forall k \in K, \tag{3}$$

$$\sum_{a\in CG(k)} y_a^k + \sum_{i\in CL(k)} f_i^k \leq M^k, \qquad \forall k \in K, \tag{4}$$

$$f_i^k \in \{0,1\}, \qquad \forall i \in F, \forall k \in K \tag{5}$$

$$y_a^k \geq 0, \qquad \forall a \in G^k, \forall k \in K. \tag{6}$$

TRANSP-OR

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

- this is the basic model, your model will have more constraints
- we don't have limit on the fleet size (we don't need variable y), but since there is a pullout cost and operating cost, the minimization function will minimize the fleet size
- be careful on what are the decision variables!
- to calculate the fare profit of the leg, use

  ```
  minl(value1, value2)
  ```

  function
- constraints to cover:
  - start/end in the base
  - come back to the same base, that the plane left
  - 2 legs can be connected only when the airport is the same
  - turnaround constraint
  - all legs covered
  - flow conservation

- how to use arc representations in OPL:
- ```
  tuple ArcLeg{
          int start;
          int end;
  }
  ```
- ```
  forall(i in Legs)
         sum(a in Arcs, b in Bases: a.end==i)...
  ```

TRANSP-OR

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

References

L. Clarke, E. Johnson, G. Nemhauser, and Z. Zhu, *The Aircraft Rotation Problem*, Annals of Operations Research **69** (1997), 33–46.

Lloyd Clarke, Ellis Johnson, George Nemhauser, and Zhongxi Zhu, *The aircraft rotation problem*, Annals of Operations Research **69** (1997), no. 0, 33–46 (English).