Transportation Problem

MIGUEL A. S. CASQUILHO
Technical University of Lisbon, Ave. Rovisco Pais, 1049-001 Lisboa, Portugal

The "Transportation Problem" is briefly presented, together with other problems that can be converted to it and thus solved by the same technique: the production scheduling; the transshipment problem; and the assignment problem

Key words: *transportation problem; supply; demand; optimization.*

1. Fundamentals and scope

In the supply chain environment, several problems related to transportation and others apparently unrelated can be formulated and solved by the technique used for the typical *transportation problem*, frequently simply denoted by the initials TP[1]. Besides the TP proper, we shall address: the (simple) *production scheduling*; the *transshipment problem*; and the *assignment problem* (AP). These problems can be solved by their own algorithms: the TP, the production scheduling and the transshipment, by the "stepping-stone" method; and the AP by the Hungarian method. As all these problems are particular cases of Linear Programming (LP), the problems will be presented and then formulated as LP problems. Indeed, with the current availability of high quality LP software, it looks unnecessary to go into the details of those other methods.

The general goal is to "transport" (whatever that may be) goods to the customers at minimum global cost of transportation, according to the unit costs of transportation (certainly according to distance, etc.) from the sources to the destinations.

The problems mentioned are dealt with in the following sections, mainly based on examples.

2. The Transportation Problem

The Transportation Problem (TP) arises from the need of programming the optimal distribution of a *single* product from given sources (supply) to given destinations (demand).

The product is available in m sources, with known quantities (also said *capacities*), a_i, $i = 1..m$ (the dots denoting a range[2]), and is needed in n destinations, with known quantities (or capacities), b_j, $j = 1..n$, and it will be sent directly from the sources to the destinations at unit costs, c_{ij}, all these values being the known data. The objective is to find the quantities to be transported, x_{ij}, at minimum global cost,

---

[1] *Initialisms* (such as TP, LP, OVNI) and *acronyms* (Interpol) are nowadays frequent (see, *e.g.*, http://en.wikipedia.org/wiki/Acronym_and_initialism).

[2] This synthetic notation means that the variable takes all the integer values from the first to the last, both included.

Miguel Casquilho is Assistant Professor (retired) in the Departament of Chemical Engineering, Instituto Superior Técnico, Universidade Técnica de Lisboa (Technical University of Lisbon), Lisbon, Portugal. E-mail address: *mcasquilho@ist.utl.pt*.

usually in a given time period, such as a week. The problem can thus be formulated according to the model of Eq. {1}.

$$[\min] z = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij}$$

subject to

$$\begin{aligned} \sum_{j=1}^{n} x_{ij} &= a_i \qquad i = 1..m \\ \sum_{i=1}^{m} x_{ij} &= b_j \qquad j = 1..n \\ x_{ij} &\geq 0 \qquad \forall i, j \end{aligned} \qquad \{1\}$$

The physical (whole) units of *x*, *a* and *b* can be, *e.g.*, kg (or m^3, bags, etc.), and *c* in \$ / kg (with \$ a generic money unit, such as dollar, euro). The scheme in Figure 1 can make the problem clear.

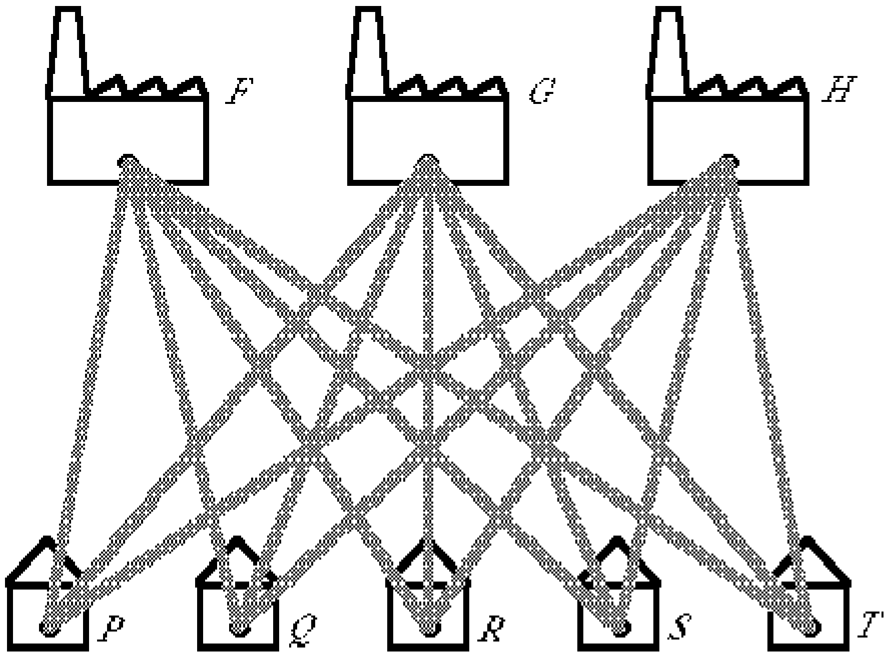

Figure 1 — Transportation Problem: from 3 factories to 5 warehouses.

Eq. {1} is, of course, in all its components (including the last one, of non-negativity of the variables), an instance of Linear Programming (LP). The notation "[max]" ([min], [opt]) means that the maximum of *both* sides is required, and not that the maximum of *z*, the objective function, is equal to the right-hand side (otherwise not yet known).

It is remarkable that the TP can be envisaged as an integer programming problem. The *x*'s will always be, namely in the optimum, multiples of the greatest common divisor of the set of *a*'s and *b*'s. So, if these are integers, the *x*'s will be integers; if, *e.g.*, these are multiples of 7, so will they be, etc.. If the problem is stated with "decimals", as 4,7, the *x*'s will be multiples of 0,1, so, with appropriate multiplication by a suitable constant, the results will be integer.

Any problem having the above structure can be considered a TP, whatever may be the subject under analysis.

EXAMPLE

A company, as in Figure 1, has 3 production centres, factories F, G and H, in given locations (different or even coincident) with production capacities of 100, 120 and 120 ton (per day), respectively, of a certain product with which it must supply 5 warehouses, P, Q, R, S and T, needing 40, 50, 70, 90 and 90 ton (per day), respectively. The unit costs of transportation, the matrix **C**, are those in Table 1. Determine the most economical transportation plan, matrix **X**.

Table 1 — Costs of transportation ($ / ton) from the factories to the warehouses

	P	Q	R	S	T	
F	4	1	2	6	9	100
G	6	4	3	5	7	120
H	5	2	6	4	8	120
	40	50	70	90	90	

RESOLUTION

Introduce the transportation matrix, **X**, in Eq. {2}, the values of whose elements must be found. (Notice that *z*, in Eq. {1}, does not result from a product of matrices !)

$$\mathbf{X} = \begin{bmatrix} x_{11} & x_{12} & x_{13} & x_{14} & x_{15} \\ x_{21} & x_{22} & x_{23} & x_{24} & x_{25} \\ x_{31} & x_{32} & x_{33} & x_{34} & x_{35} \end{bmatrix} \qquad \{2\}$$

The problem could be solved by the adequate “stepping-stone” method (which is very efficient), but its formulation leads directly to the LP in Eq, {3}.

$$\begin{aligned} [\min] z = {} & c_{11}x_{11} + c_{12}x_{12} + c_{13}x_{13} + c_{14}x_{14} + c_{15}x_{15} + \\ & + c_{21}x_{21} + \ldots + c_{25}x_{25} + \\ & + c_{31}x_{31} + \ldots + c_{35}x_{35} \end{aligned}$$

subject to

$$\begin{aligned} x_{11} + x_{12} + x_{13} + x_{14} + x_{15} &= a_1 \\ x_{21} + x_{22} + x_{23} + x_{24} + x_{25} &= a_2 \\ x_{31} + x_{32} + x_{33} + x_{34} + x_{35} &= a_3 \end{aligned} \qquad \{3\}$$

$$\begin{aligned} x_{11} + x_{21} + x_{31} &= b_1 \\ x_{12} + x_{22} + x_{32} &= b_2 \\ &\ldots \\ x_{15} + x_{25} + x_{35} &= b_5 \end{aligned}$$

A TP has really $m + n - 1$ *independent* constraints (not $m + n$), as one of the constraints shown above is superfluous (dependent). This is due to the nature of the TP, in which total supply must equal total demand.

The fact that the current solvers, such as Excel or Excel/Cplex, natively accept the TP in table form makes the constraints in Eq. {3} readily available for solution. (This would not be the case with other common solvers such as Lindo[3], in which all the $m.n$ variables and $m + n - 1$ equations have to be explicitly inserted.)

The solution is given in Table 2, with a minimum global cost of $z^* = 1\,400$ $ (per day, the period considered). In this particular problem, there are two (multiple) solutions, the other differing in $x_{11} = 10$, $x_{12} = 50$, $x_{31} = 30$ and $x_{32} = 0$.

Table 2 — Quantities to be transported (ton) from the factories to the warehouses

	P	Q	R	S	T	
F	40	20	40			
G			30		90	120
H		30		90		120
	40	50	70	90	90	

■

The TP is, naturally, "balanced", i.e., the total supply is equal to the total demand. In cases where there is excess *supply*, the problem can be readily converted to a TP by creating (at cost 0) a *fictitious destination*; or if there is excess *demand*, a *fictitious source*. So, product could be left "at home" or, possibly, bought from some competitor to guarantee the supply to the customer, respectively.

3. The production scheduling

The (simple) production scheduling problem will be presented through the example in Hillier & Lieberman [2006, pp 330–331][4].

EXAMPLE

Table 3 — Production scheduling data for Northern Airplane Co.

Month	Scheduled installations	Max. production	Unit cost of production	Unit cost of storage
1	10	25	1,08	0,015
2	15	35	1,11	0,015
3	25	30	1,10	0,015
4	20	10	1,13	0,015

The "transportation" in the production scheduling is not in space, but in time, between months in this esample. Surely, production cannot, however, be made in a certain month to be supplied in a previous month, so that the corresponding unit costs must be prohibited, by making them "very large", say, M (the classical "big M"), infinity, or, for computing purposes, sufficiently large (in this problem, e.g., 100). The data of Table 3 become those in Table 4, adding the storage costs and introducing a fictitious fifth month for balancing.

[3] Free (student) limited version available from *http://www.lindo.com* .
[4] See http://web.ist.utl.pt/mcasquilho/acad/or/TP/HL.ProdSched.pdf .

In order to “guess” a sufficiently large *M*, try some “reasonably” large value, i.e., at least large compared to the other cost values in the problem. If this value is effective in the solution (prohibiting the related *x*’s), it is a good choice, but, if it is not effective (too small), try a greater new value. If the value is “never” sufficiently large, then, the problem has no physical solution (is *impossible*), although it always has a mathematical one.

Table 4 — TP-like data for the Northern Airplane Co. problem

Month	1	2	3	4	(5)	Supply
1	1,080	1,095	1,110	1,125	0	25
2	*M*	1,110	1,125	1,140	0	35
3	*M*	*M*	1,100	1,115	0	30
4	*M*	*M*	*M*	1.130	0	10
Demand	10	15	25	20	30	**(100)**

The solution has a (minimum) global cost of 77,3 $ with the production schedule given in Table 5.

Table 5 — Production schedule for the Northern Airplane Co. problem

Month	1	2	3	4	(5)	Supply
1	**10**	**10**	**5**	**0**	0	25
2	—	**5**	**0**	**0**	30	35
3	—	—	**20**	**10**	0	30
4	—	—	—	**10**	0	10
Demand	10	15	25	20	30	**(100)**

4. The transshipment problem and the assignment problem

The transshipment problem and the assignment problem (AP) can be considered problems reducible to TP’s. The transshipment is typically treated like a TP, whereas the AP has the Hungarian algorithm, which is very efficient. Notwithstanding, this algorithm will not be presented, as the AP is a particular LP and is appropriately solved by the usual LP software.

The method to reduce a transshipment problem to a TP is simply to consider that the transshipment points are supply points or demand points or both, by inserting them on the supply side or the demand side (or both). Upon inserting these transhipments as referred, each individual capacity must be “corrected” by adding to it the original global capacity.

The AP is a particular case of the TP, having a square cost matrix and being soluble by considering all the values of supply as 1 and all the values of demand also as 1.

EXAMPLE, TRANSSHIPMENT

Refer to Problem 9.3[5] from Bronson & Naadimuthu [1997BRO], represented in Figure 2.

[5] See http://web.ist.utl.pt/mcasquilho/acad/or/TP/BronsonNaad97_transsh.pdf .

RESOLUTION

In order to convert the transshipment to a TP, identify: *(a)* every pure supply point (producing only), usually labelled with a positive quantity, such as Point 1 with +95 units; *(b)* every pure demand point (receiving only), usually labelled with a negative quantity, such as Point 5 with –30 units; and *(c)* every mixed point (producing or receiving), labelled with a positive (if net producer) or negative (if net receiver) quantity, such as Point 3 with +15 units. Make the original TP balanced, which results here in a dummy destination (say, Point 7), and register the original capacity of the TP, Q (here Q = 180).

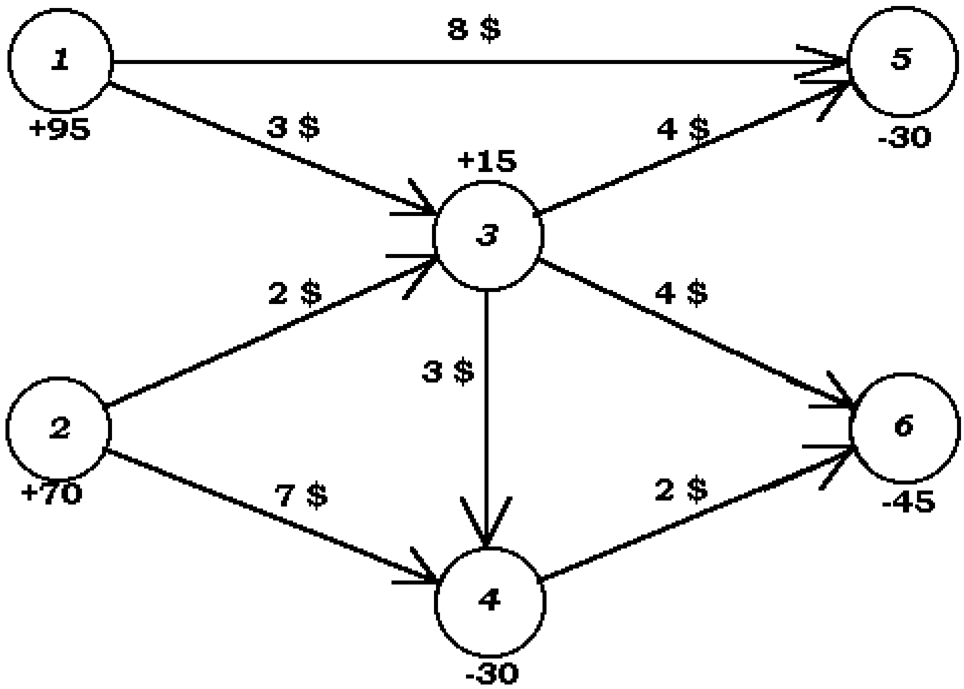

Figure 2 — Transshipment problem: sources (1 and 2), destinations (5 and 6), and transshipment points (3 and 4).

The cost matrix becomes the one in Table 6. (The number of times Q is inserted on the supply side and on the demand side is, of course, the same, thus maintaining the equilibrium necessary to a TP.) The solution is in Table 7.

Table 6 — Cost matrix for the transshipment problem

	3	4	5	6	(7)	Supply
1	3	M	8	M	0	95
2	2	7	M	M	0	70
3	0	3	4	4	0	15+Q
4	M	0	M	2	0	Q
Demand	Q	30+Q	30	45	75	(180+2Q)

So: from Point 1, 20 units go to Point 3, and 75 stay home; from Point 2, 70 units go to Point 3; from Point 3, 30 go to Point 4, etc.; and Point 4 just receives 30 (from Point 3), with its quantity (equal to Q) meaning it was not used as a transshipment point.

Table 7 — Solution to the transshipment problem

	3	4	5	6	(7)	Supply
1	20	—	0	—	75	95
2	70	0	—	—	0	70
3	90	30	30	45	0	15+*Q*
4	—	180	—	0	0	*Q*
Demand	*Q*	30+*Q*	30	45	75	(180+2*Q*)

EXAMPLE, ASSIGNMENT

Suppose n tasks are to be accomplished by n workers and the workers have the abilities for every task as given in Table 8.

Table 8 — Ability of each worker for each task

	1	*2*	*3*	*4*
1	15	16	14	14
2	14	14	13	15
3	13	15	13	14
4	15	16	14	14

These "positive" abilities are converted to "costs", replacing each element by its difference to their maximum (here, 16). After that, solve the AP as a TP with each supply equal to 1 and each demand also equal to 1. The solution to this example is in Table 9.

Table 9 — Assignments (solution)

	1	*2*	*3*	*4*
1	1	0	0	0
2	0	0	0	1
3	0	0	1	0
4	0	1	0	0

So, Worker 1 does Task 1, Worker 2 does Task 4, etc. (i.e., 1-1, 2-4, 3-3, 4-2), at a minimum global cost of 5 cost units. This problem has multiple solutions, another being 1-2, 2-4, 3-3, 4-1.

As this problem has, obviously, always n elements of value 1 (the assignments) and the remaining $n^2 - n$ of value zero, its optimum solution is very *degenerate* if, as was done, it is considered a TP (degenerate in relation to the $m + n - 1$ possible positive cells in a TP). This is an argument in favour of the Hungarian method, but the strength of that method is not significant when common software is used.

4. Conclusions

In the supply chain, several problems related to transportation can be formulated and solved by the technique used for the typical transportation problem (TP), with its own very efficient algorithm (stepping-stone): the (simple) production scheduling; the transshipment problem; and the assignment problem (AP). The AP can be solved by its own algorithm, but the availability of software for Linear Programming makes it practical to solve them as TP's, after convenient simple conversions, other algorithms becoming unnecessary

Acknowledgements

This work was done at "Centro de Processos Químicos do IST" (Chemical Process Research Centre of IST), Department of Chemical Engineering, Technical University of Lisbon. Computations were done on the central system of CIIST (Computing Centre of IST).

Bibliography

– [1997BRO] BRONSON, Richard and Govindasami NAADIMUTHU, 2010, "Schaum's Outline of theory and problems of Operations Research", 2.nd ed., McGraw-Hill, New York, NY (USA).

❖

9.3 For the data of Fig. 9-2, determine a shipping schedule that meets all demands at a minimum total cost.

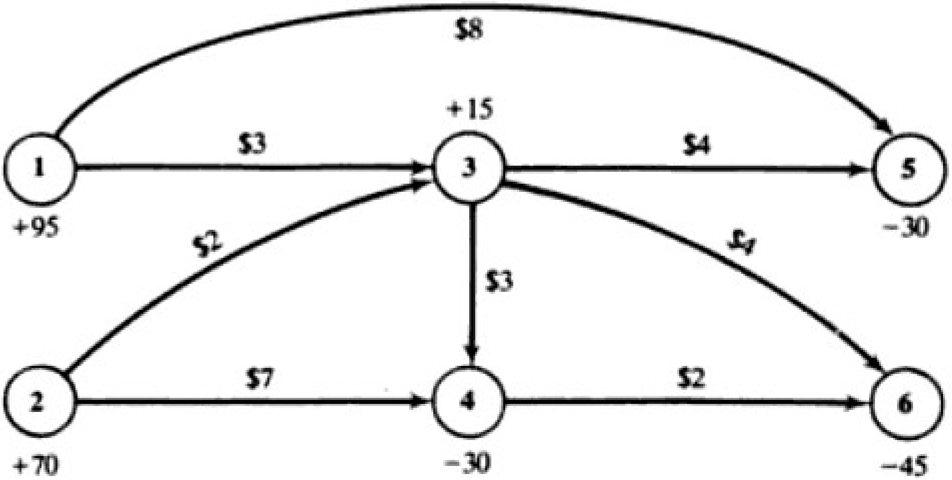

Fig. 9-2

Locations 1 and 2 are sources, while locations 5 and 6 are destinations. Location 3 is both a source and a junction, whereas location 4 serves both as a destination and a junction. Because total supply is 180 units but total demand is only 105 units, location 7 is created as a dummy destination with a demand of $180 - 105 = 75$ units. Since every junction is made both a source and a destination, by adding 180 units to both its supply and its demand, the transportation tableau will involve sources 1, 2, 3, 4, and destinations 3, 4, 5, 6, 7. Besides the costs given in Fig. 9-2, we have zero as the cost from a junction (as a source) to itself (as a destination), zero as the cost from any source to the dummy, and an excessive amount ($10 000) as the cost over any nonexistent link (e.g., $1 \rightarrow 6$).

Tableau 3 is the optimal transportation tableau. Location 3 receives 20 units from location 1 and 70 units from location 2, whereupon it redistributes these units along with its own initial supply of 15 units to locations 4, 5, and 6. After all demands have been satisfied, location 1 will remain with 75 units, indicated

Destinations

	3	4	5	6	(dummy) 7	Supply	u_i
1	3 **20**	10000 (9994)	8 (1)	10000 (9993)	0 **75**	95	3
2	2 **70**	7 (2)	10000 (9994)	10000 (9994)	0 (1)	70	2
3	0 **90**	3 **30**	4 **30**	4 **45**	0 (3)	195	0
4	10000 (10003)	0 **180**	10000 (9999)	2 (1)	0 (6)	180	−3
Demand	180	210	30	45	75		
v_j	0	3	4	4	−3		

Tableau 3

in Tableau 3 by the allocation from location 1 to the dummy. The allocations $x_{33}^* = 90$ and $x_{44}^* = 180$ are book entries signifying the numbers of units that do not pass through junctions 3 and 4, respectively.

9.4 Solve Problem 1.13 by the Hungarian method.

Table 1-1 of Problem 1.13 is expanded to make the number of events equal to the number of swimmers; the result is Tableau 4A. As usual, costs (times) associated with the dummies, events 5 and 6, are taken to be zero. The rationale here is that events 5 and 6 do not exist, so they can be completed in zero time; swimmers assigned to these events will be the ones not entered in the four-swimmer relay.

The Hungarian method is initiated by subtracting 0 from every row of Tableau 4A and then subtracting 65, 69, 63, 55, 0, and 0 from columns 1 through 6, respectively; this generates Tableau 4B. Since this matrix does not contain a zero-cost feasible solution, we cover the existing zeros by as few horizontal and vertical lines as possible. One such covering is that shown in Tableau 4B; another, equally good, is obtained by replacing the line through row 3 by a line through column 4. The smallest uncovered element is 1, appearing in the (2, 2) position. Subtracting 1 from every uncovered element in Tableau 4B and adding 1 to every element covered by two lines—the (1, 5), (1, 6), (3, 5), (3, 6), (5, 5), and (5, 6) elements—we arrive at Tableau 4C.

Tableau 4C also does not contain a feasible zero-cost assignment. Repeating Step 3 of the Hungarian method, we determine that 1 is again the smallest uncovered element. Subtracting it from each uncovered

(Blank page)