METHODS OF DISTRIBUTION, LAYOUT AND HUNGARIAN METHOD

Dávid Šimko
TU of Košice, Faculty BERG, Institute of Logistics, Park Komenského 14, 043 84 Košice, Slovakia,
dav.simko@gmail.com

Keywords: distribution, lay-out, allocation, transportation problems
Abstract: The article analyzes and describes theoretical basics of various methods used for lay-out, allocation or solving various types of transportation problems. Lay-out and allocation represent two of the very important decisions which is made by macrologistics management. Nowadays we know several types of solving these problems so it is very important to be able to chose the right way and solve these problems as optimally as it is possible. This article is going to describe some of them.

1 Layout

Layout, as the layout of production, non-production and storage capacities in the company is undoubtedly one of the most important tasks because it has a direct impact not only on the company economy but also affects safety at work and social environment of the company.

Layout design is usually one-off task. The usual case is also the gradual creation of layout, where the company objects are filled with manufacturing equipment and storage areas in the longer term with advancing other processes, respectively. increasing the capacity of existing processes. The last case is to design the ideal layout with a finite number of machines, equipment and storage areas in a given area [1].

We know several ways how to solve the layout problems, for example the quadratic assignment problem, linear assignment problem, loop layout design, value stream mapping, group technology, etc. and now we describe some of them.

1.1 Loop layout design

A common layout for flexible manufacturing systems is a loop network with machines arranged in a cycle and materials transported in only one direction around the cycle. Traffic congestion is usually used as the measure for evaluating a loop layout, which is defined as the number of times a part traverses the loop before its processing is completed.

The essence of the problem is how to determine the order of machines around the loop subject to a set of part-route constraints so as to optimize some measures. A hybrid approach of genetic algorithms and neighborhood search is developed for solving the problem. The proposed method is tested on hypothetical problems. Computational results demonstrate that genetic algorithms can be a promising approach for loop layout design in flexible manufacturing systems.

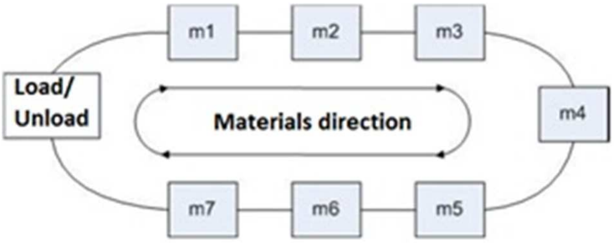

Figure 1 Scheme of the loop layout design [1]

Following Afentukis, we consider a system in which machines are arranged in a loop and materials arc transported in one direction around the loop (Figure 1). There is one loading/unloading station where parts enter and leave the system. Suppose that there are n machines in the system, $M = \{0, I, 2,. . - ,n\}$, where 0 denotes the loading/unloading station. A loop layout design can be represented as s permutation of machines *(m1, m2,.. . , m,)* with s prefix of loading/unloading station 0. Each part is characterised by its part-route, the sequence of machines it must visit to complete its processing. For s given part, suppose that processing on machine *j* immediately follows processing on machine *i*. If the position of machine *j* is lower than that of machine *i*, then the part must cross the loading/unloading station, which is called s reload. The number of reloads necessary to complete the processing for a part is defined as the measure of traffic congestion. The problem is to find out the best layout (or machines permutation) by optimising some measure of performances subject to a set of part-route constraints.

There are two measures commonly used in evaluating such loop layout: *minsum* and *minmax*. The *minsum* measure attempts to find out the permutation of machines which minimises the total number of reloads for all parts and seeks n more balanced congestion among parts; while the *minmax* measure attempts to find out the permutation of machines which minimises the maximum reload among *n* family of parts and tries to reduce the aggregate congestion of the system.

1.2 Quadratic assignment problem

The quadratic assignment problem (QAP) was introduced by Koopmans and Beckmann in 1957 as a mathematical model for the location of a set of indivisible economical activities. Consider the problem of allocating a set of facilities to a set of locations, with the cost being a function of the distance and flow between the facilities, plus costs associated with a facility being placed at a certain location (Figure 2). The objective is to assign each facility to a location such that the total cost is minimized. Specifically, we are given three $n \times n$ input matrices with real elements $F = (f_{ij})$, $D = (d_{kl})$ and $B = (b_{ik})$, where f_{ij} is the flow between the facility i and facility j, d_{kl} is the distance between the location k and location l, and b_{ik} is the cost of placing facility i at location k.

The Koopmans-Beckmann version of the QAP (1) can be formulated as follows: Let n be the number of facilities and locations and denote by N the set $N = \{1,2,...,n\}$.

$$min_{\varphi \in S_n} \sum_{i=1}^{n} \sum_{j=1}^{n} f_{ij}\, d_{\varphi(i)\varphi(j)} + \sum_{i=1}^{n} b_{i\varphi(i)} \quad (1)$$

where S_n is the set of all permutations $\varphi : N \rightarrow N$. Each individual product $f_{ij}d\varphi(i)\varphi(j)$ is the cost of assigning facility to location $\varphi(i)$ and facility j to location $\varphi(j)$. In the context of facility location the matrices F and D are symmetric with zeros in the diagonal, and all the matrices are nonnegative. An instance of a QAP with input matrices F,D and B will be denoted by QAP (F,D,B), while we will denote an instance by QAP (F,D), if there is no linear term (i.e., $B= 0$). A more general version of the QAP was introduced by Lawler. In this version (2) we are given a four-dimensional array $C = (c_{ijkl})$ of coefficients instead of the two matrices F and D and the problem can be stated as

$$min_{\varphi \in S_n} \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij\varphi(i)\varphi(j)} + \sum_{i=1}^{n} b_{i\varphi(i)} \quad (2)$$

Clearly, a Koopmans-Beckmann problem QAP (F,D,B) can be formulated as a Lawler QAP by setting c_{ijkl}: = $f_{ij}d_{kl}$ for all i,j,k,l with $i \neq j$ or $k \neq l$ and c_{iikk}:=$f_{ii}d_{kk}$+b_{ik}, otherwise. Although extensive research has been done for more than three decades, the QAP, in contrast with its linear counterpart the linear assignment problem (LAP), remains one of the hardest optimization problems and no exact algorithm can solve problems of size n >20 in reasonable computational time. In fact, Sahni and Gonzalez have shown that the QAP is NP-hard and that even finding an approximate solution within some constant factor from the optimal solution cannot be done in polynomial time unless P=NP. These results hold even for the Koopmans-Beckmann QAP with coefficient matrices fulfilling the triangle inequality. So far only for a very special case of the Koopmans-Beckmann QAP, the dense linear arrangement problem a polynomial time approximation scheme has been found , due to Arora, Frieze, and Kaplan. In addition to facility layout problems, the QAP appears in applications such as backboard wiring, computer manufacturing, scheduling, process communications, turbine balancing, and many others.

One of the earlier applications goes back to Steinberg and concerns backboard wiring. Different devices such as controls and displays have to be placed on a panel, where they have to be connected to each other by wires. The problem is to find a positioning of the devices so as to minimize the total wire length. Let n be the number of devices to be placed and let d_{kl} denote the wire length from position k to position l. The flow matrix F= (f_{ij}) is given by

$$f_{ij} = \begin{cases} 1 \text{ if device } i \text{ is connected to device } j \\ 0 \text{ otherwise.} \end{cases}$$

Then the solution to the corresponding QAP will minimize the total wire length.

Another application in the context of location theory is a campus planning problem due to Dickey and Hopkins. The problem consists of planning the sites of n buildings in a campus, where d_{kl} is the distance from site k to site l, and f_{ij} is the traffic intensity between building i and building j. The objective is to minimize the total walking distance between the buildings.

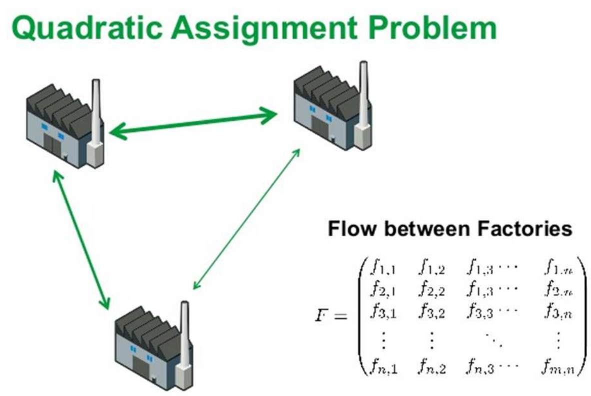

Figure 2 Scheme of the quadratic assignment problem [2]

In the field of ergonomics Burkard and Offermann showed that QAPs can be applied to typewriter keyboard design. The problem is to arrange the keys in a keyboard such as to minimize the time needed to write some text. Let the set of integers $N = \{1, 2, ... n\}$ denote the set of symbols to be arranged. Then f_{ij} denotes the frequency of the appearance of the pair of symbols i and j. The entries of the distance matrix $D = d_{kl}$ are the times needed to press the key in position l after pressing the key in position k for all the keys to be assigned. Then a permutation $\varphi \in S_n$ describes an assignment of symbols to keys. An optimal solution $\varphi *$ for the QAP minimizes the average time for writing a text.

A similar application related to ergonomic design, is the development of control boards in order to minimize eye fatigue by McCormick. There are also numerous other applications of the QAP in different fields e.g. hospital lay-out (Elshafei), ranking of archeological data (Krarup and Pruzan), ranking of a team in a relay race (Heffley), scheduling parallel production lines (Geoffrion and Graves), and analyzing chemical reactions for organic compounds (Ugi, Bauer, Friedrich, Gasteiger, Jochum, and Schubert) [1].

2 Transportation problems

The transportation problem is concerned with finding the minimum cost of transporting a single commodity from a given number of sources (e.g. factories) to a given number of destinations (e.g. warehouses). These types of problems can be solved by general network methods, but here we use a specific transportation algorithm.

The data of the model include:

1. The level of supply at each source and the amount of demand at each destination.

2. The unit transportation cost of the commodity from each source to each destination.

Since there is only one commodity, a destination can receive its demand from more than one source. The objective is to determine how much should be shipped from each source to each destination so as to minimize the transportation cost.

In the following picture (Figure 3) we can see a model with *m* sources and *n* destinations. The amount of supply available at source *i* is a_i and the demand required at destination *j* is b_j. X_{ij} represents the quantity of supply transported from source *i* to destination *j* and the cost associated with this movement is represented by c_{ij}.

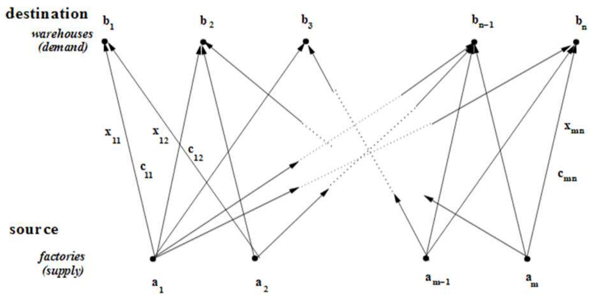

Figure 3 Scheme of the transportation problem [3]

According to this the total cost (3) of transporting the commodity from all the sources to all the destinations is

$$Total\ cost = \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} \qquad (3)$$

And the main goal is to minimise (4) this total costs, so the following problem must be solved

$$\min z = \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} \qquad (4)$$

While

$$\sum_{j=1}^{n} x_{ij} \le a_i \ for\ i = 1, \dots, m$$
$$\sum_{j=1}^{m} x_{ij} \ge b_j \ for\ i = 1, \dots, n$$
$$x_{ij} \ge 0 \ for\ all\ i\ and\ j$$

When the total supply is equal to the total demand then the transportation model is said to be balanced. A transportation model in which the total supply and total demand are unequal is called unbalanced. It is always possible to balance an unbalanced transportation problem.

If the transportation model is unbalanced we introduce a dummy source (fictitious factory/warehouse) which helps us to balance the capacities or the demands in the transportation model. Since the source doesn´t exist, no shipping from the source will occur, so the unit transportation cost can be set to zero.

2.1 *The North-West Corner Method*

Consider the problem represented by the following transportation table (Table 1). The number in the bottom right of cell (*i; j*) is c_{ij}, the cost of transporting 1 unit from source *i* to destination *j*. Values of *xij*, the quantity actually transported from source *i* to destination *j*, will be entered in the top left of each cell. Note that there are 3 factories and 4 warehouses and so *m= 3, n= 4* [4].

Table 1 Transportation table [3]

	W_1	W_2	W_3	W_4	Supply
F_1	10	0	20	11	20
F_2	12	7	9	20	25
F_3	0	14	16	18	15
Demand	10	15	15	20	

The north-west corner method generates an initial allocation according to the following procedure:

1. Allocate the maximum amount allowable by the supply and demand constraints to the variable x_{11} (i.e. the cell in the top left corner of the transportation tableau).
2. If a column (or row) is satisfied, cross it out. The remaining decision variables in that column (or row) are non-basic and are set equal to zero. If a row and column are satisfied simultaneously, cross only one out (it does not matter which).
3. Adjust supply and demand for the non-crossed out rows and columns.

4. Allocate the maximum feasible amount to the first available non-crossed out element in the next column (or row).
5. When exactly one row or column is left, all the remaining variables are basic and are assigned the only feasible allocation.

2.2 The Least-Cost Method

This method usually provides a better initial basic feasible solution than the North-West Corner method since it takes into account the cost variables in the problem:

1. Assign as much as possible to the cell with the smallest unit cost in the entire tableau. If there is a tie then choose arbitrarily.
2. Cross out the row or column which has satisfied supply or demand. If a row and column are both satisfied then cross out only one of them.
3. Adjust the supply and demand for those rows and columns which are not crossed out.
4. When exactly one row or column is left, all the remaining variables are basic and are assigned the only feasible allocation.

3 The Hungarian method

The assignment problem deals with assigning machines to tasks, workers to jobs, soccer players to positions, and so on. The goal is to determine the optimum assignment that, for example, minimizes the total cost or maximizes the team effectiveness. The assignment problem is a fundamental problem in the area of combinatorial optimization.

With one of the most effective methods for solving the assignment problems came Kuhn. Preparing of his algorithm was based on the work of the Hungarian mathematician Egerváry. Kuhn generalized Egerváry´s method, calling it " The Hungarian method" (in some literature is also referred to as "KUHN algorithm").

The Hungarian method does not respond to the degeneration of the solution that is the weak point of other algorithms and requires no initial solution obtained by another (approximate) method. If we continue to mention the initial solutions, it will be a specific case of baseline, which is part of the creation of the Hungarian algorithm itself.

The basic process of the solution (in case of the balanced transportation problem) can be described as follows:

First we create an initial solution, which does not suits to restriction of our transportation problem. From some sources are not transported all the material and not all consumer demands are satisfied. For this solution can be written these restrictive conditions (5), (6):

$$\sum_{j=1}^{n} x(i,j) \leq a \quad (i) \quad \text{for } i \in \{1,2, ..., m\} \quad (5)$$

$$\sum_{i=1}^{m} x(i,j) \leq b \quad (j) \quad \text{for } j \in \{1,2, ..., n\} \quad (6)$$

We improve the initial solution until in the previous relations don´t apply only conditions of equality. With that we fullfill the restrictive conditions. All material resources are assigned to all consumers and all consumer demands are satisfied. In this case, the algorithm ends and the result is optimal. The whole algorithm of The Hungarian method consists of four stages. A schematic illustration of a follow-up stages is shown in the following algorithm (Figure 4) [5].

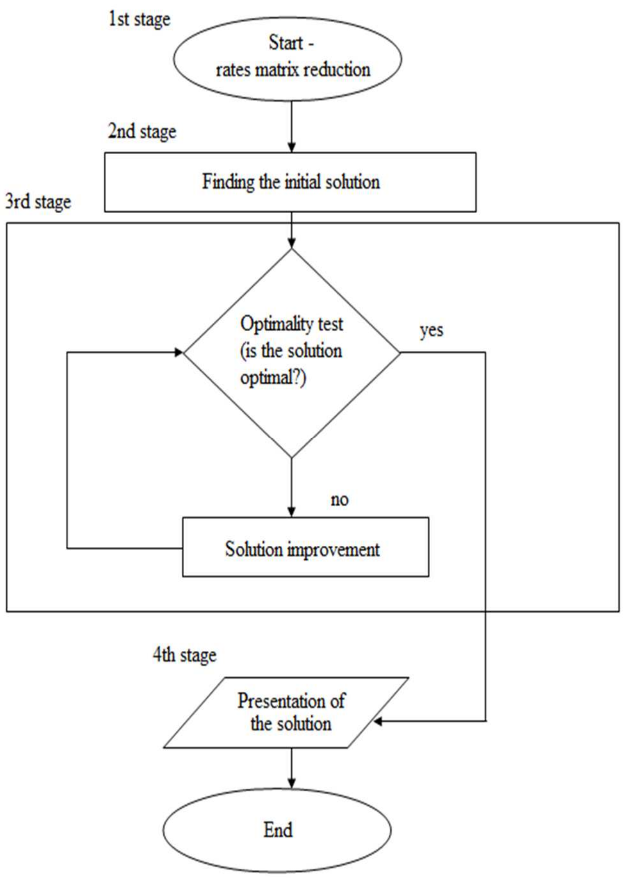

Figure 4 The Hungarian method algorithm

In the actual solution it means that the first two steps are executed once, while Steps 3 and 4 are repeated until an optimal assignment is found. The input of the algorithm is an *n* by *n* square matrix with only nonnegative elements.

Step 1: Subtract row minima

For each row, find the lowest element and subtract it from each element in that row.

Step 2: Subtract column minima

Similarly, for each column, find the lowest element and subtract it from each element in that column.

Step 3: Cover all zeros with a minimum number of lines

Cover all zeros in the resulting matrix using a minimum number of horizontal and vertical lines. If *n* lines are required, an optimal assignment exists among the zeros. The algorithm stops.

If less than *n* lines are required, continue with Step 4.

Step 4: Create additional zeros

Find the smallest element (call it *k*) that is not covered by a line in Step 3. Subtract *k* from all uncovered elements, and add *k* to all elements that are covered twice [6].

Conclusion

Decisions about the layout and allocation are very sensitive and delicate because they are made usually just once, when the corporation is building a new objects. Wrong allocation or layout of the productional and non-productional subjects, storage areas or distribution centres of the corporation can later cause huge problems.

As we can see there are plenty of ways how to solve these different problems of transportation, layout or allocation and it´s just our call which one we are going to use for solving these types of problems. It´s also important to know, that every single case has usually it´s own variables which are different from other cases. That means there can be different optimal method for every case.We have to choose the correct method according these variables and achieve as optimal result as we can.

References

[1] DVOŘÁČEK, R.: *Optimalizace logistického toku v podniku*. Dizertačná práca. Vysoké učení technické v Brně, 2009. (Original in Czech)

[2] FEIT, A.: SICSA Summer School on Computational interaction - 17, [Online], Available: http://www.slideshare.net/oulasvir/modelbased-user-interface-optimization-part-ii-letter-assignment-at-sicsa-summer-school-on-computational-interaction-2015?qid=f5503822-3a64-4bd2-85b1-5f1e96b43fc0&v=&b=&from_search=2 [12 Jun 2016], 2016.

[3] RAYMAN, J.: *Transportation Problems*, Chapter 7, University of Surrey, [Online], Available: http://personal.maths.surrey.ac.uk/st/J.F/chapter7.pdf [11 Jun 2016], 2016.

[4] *Aplikácia matematických metód v krízovom plánovaní*. Fakulta bezpečnostného inžinierstva. Žilinská univerzita v Žiline, [Online], Available: https://www.yumpu.com/xx/document/view/34103128/7-aplikacia-matematickach-metad-v-krazovom-planovana/9 [12 Jun 2016], 2016. (Original in Slovak)

[5] KAŠPAR, V.: *Základné metódy operačnej analýzy vo vojenskej doprave (dopravná a priraďovacia úloha)*. Vysoká škola dopravy a spojov Žilina. Žilina: VF VŠDS, 1995. (Original in Slovak)

[6] Hungarian Algorithm, [Online], Available: http://www.hungarianalgorithm.com/hungarianalgorithm.php [10 Jun 2016], 2016.

Review process

Single-blind peer reviewed process by two reviewers.