Solving the Multi-story Space Assignment Problem as a Generalized Quadratic Assignment Problem

Guilherme H. I. Azevedo
Production Engineering Department, Universidade Federal Fluminense
Rua Passo da Pátria, 156, Bloco D, 306, São Domingos, Niterói – RJ, CEP 24210-240

Artur Alves Pessoa
Production Engineering Department, Universidade Federal Fluminense
Rua Passo da Pátria, 156, Bloco D, 306, São Domingos, Niterói – RJ, CEP 24210-240

Peter M. Hahn
Electrical and Systems Engineering Department, University of Pennsylvania
Philadelphia, PA 19104 USA

J. MacGregor Smith
Mechanical Engineering Department, University of Massachusetts
Amherst, MA

RESUMO
O Problema de alocação espacial em múltiplos andares (MSAP) consiste em atribuir departamentos aos andares de um edifício considerando custos associados aos fluxos entre os departamentos e aos fluxos dos departamentos para as escadas no caso de evacuação. Um algoritmo de branch-and-bound anterior para este problema se baseou no Problema de Alocacao Tridimensional Quadrática Generalizado (GQ3AP). Neste artigo, apresentamos uma nova redução do MSAP para o mais simples Problema de Alocacao Quadrática Generalizado (GQAP). Relatamos experimentos utilizando um método proposto anteriormente para o GQAP onde a otimalidade das soluções de várias instâncias do MSAP foram provadas mais rapidamente e com menos nós.

PALAVRAS CHAVE. Arranjo físico, Alocacao quadratica, Reducoes, Algoritmos exatos.
Área principal: Otimização.

ABSTRACT
The Multi-story Space Assignment Problem (MSAP) consists of assigning departments to floors in a building considering costs associated to flows among departments and flows from departments to stairways in case of evacuation. A previous branch-and-bound algorithm for this problem was based on the Generalized Quadratic 3-dimensional Assignment Problem (GQ3AP). In this paper, we present a reduction from the MSAP to the simpler Generalized Quadratic Assignment Problem (GQAP). We report experiments using a previously proposed method for the GQAP, where the optimality of solutions for several MSAP instances was proved faster and using fewer nodes.

KEYWORDS. Physical layout, Quadratic assignment, Reductions, Exact algorithms.
Main area: Optimization.
1. Introduction

1.1 Background and problem formulation

The Multi-story Space Assignment Problem arises from the following considerations:

Given \( M \) departments, where for each department \( i \) the number of people per department and the amount of space required for the department \( a_i \) is known, how shall one allocate the \( M \) departments to \( N \) floors, such that the evacuation time is minimized?

The objective here is to find an assignment that minimizes the sum of all one-way evacuation costs. Certain constraints must be met: 1) no department may be split between different floors; and 2) the available space on each floor may not be exceeded.

A secondary, but nevertheless important, issue: Consider the same \( M \) departments and \( N \) floors. For each pair of departments \((i,k)\) a certain flow of units (i.e., occupants or materials) \( f(i,k) \) is known and for each pair of floors \((j,n)\) a corresponding cost to move one unit of flow between floors \( d(j,n) \) is known. The secondary objective is to encourage assignments that minimize the sum of all transportation costs, given that the above constraints are met. Both primary and secondary issues are dealt with in the Multi-Story Assignment Problem (MSAP), defined here:

The following notation is suggested:

\( M := \) number of departments need to be placed

\( N := \) number of available floors

\( S := \) number of stairways

\( a_i := \) square feet of floor area needed by department \( i \)

\( A_j := \) maximum usable floor area on floor \( j \)

\( f_{ik} := \) flow units from department \( i \) to department \( k \)

\( d_{jn} := \) cost per flow-unit travel between floor \( j \) and floor \( n \).

\( e_{js} := \) distance between floor \( j \) and stairway exit \( s \) from building

\( \lambda_i := \) Poisson arrival rate of persons in the evacuation from department \( i \)

\( \mu_s := \) service rate capacity of stairway \( s \)

\( x_{ij} := 1 \) if department \( i \) is assigned to floor \( j \), and is 0 otherwise.

\( y_{is} := 1 \) if department \( i \) is assigned to stairway \( s \), and is 0 otherwise.

The problem is,

\[
\min \sum_{i=1}^{M} \sum_{j=1}^{N} a_i x_{ij} \sum_{s=1}^{S} e_{js} x_{ij} + \sum_{i=1}^{M} \sum_{j=1}^{N} \sum_{s=1}^{S} d_{jn} x_{ij} y_{is}
\]

subject to the following constraints on \( X \) and \( Y \):

\[
\sum_{j=1}^{N} a_i x_{ij} \leq A_{\varphi} \quad (\varphi = 1, 2, \ldots, N)
\]
\[
\sum_{i=1}^{N} x_{ij} = 1 \quad (i=1,2,\Lambda , M) \tag{3}
\]

\[
M \sum_{i=1}^{\lambda} \lambda_{iy} \leq m_{s} \quad (s=1,2,\Lambda , S) \tag{4}
\]

\[
\sum_{i=1}^{E} y_{is} = 1 \quad (i=1,2,\Lambda , M) \tag{5}
\]

\[
x_{ij} = 0, 1 \quad (i = 1, 2, \ldots, M; \ j = 1, 2, \ldots, N), \tag{6}
\]

\[
y_{is} = 0, 1 \quad (i = 1, 2, \ldots, M; \ s = 1, 2, \ldots, S), \tag{7}
\]

X and Y are said to be a ‘solution’.

Eq. 2 makes sure that the capacity of each space is not exceeded, Eq. 3 makes sure that each department gets assigned only one space, Eq. 4 makes sure that the capacity of each escape route is not exceeded, and Eq. 5 makes sure that each department is assigned only one primary escape route.

With precisely the same model, one can subdivide departments, so that their subdivisions can be assigned independently to stairways. If desired, intra-department flow-unit values can be chosen so that sub-departments will always be assigned to a single floor. Though, the model is general enough to allow sub-departments to be located one over another on different floors. We have therefore added experiments to show the simplicity and effectiveness of such an approach. In a real situation, the choice of how to subdivide departments and how to avoid spreading them too much would be a subject for future research.

If floors are large, so that distances traveled are significant, floors can be divided into smaller entities, such as “compartments”. In this case the distance matrix would represent distances between “compartments” rather than distances between floors.

1.1 Related work

Hahn, Smith and Zhu (2008) formulated the MSAP as part of their inter-disciplinary work on multi-story building design and evacuation planning. In order to solve the MSAP, they first gave consideration to using the algorithms developed for the Quadratic 3-dimensional Assignment Problem (Q3AP) by Hahn, et al. (2008). But, it became clear that the Q3AP was limited to assigning only one department to each space available in the building. Thus, it became necessary to extend the Q3AP model to one that would accommodate multiple department placements. Thus, a new model was pursued, i.e., the Generalized Quadratic 3-dimensional Assignment Problem (GQ3AP). A description of the GQ3AP is given in the dissertation of Y-R Zhu (2007).

The GQ3AP exact solution algorithm is capable of solving MSAP problem instances of up to 13 departments, 8 floors and 3 stairways. This limits its usefulness. Thus, it became necessary to seek other exact solution methods. This is the motivation for the work reported in this paper, where we reduce the MSAP to the Generalized Quadratic Assignment Problem (GQAP). As far as we know, the first work on the GQAP is due to Lee and Ma (2004) where they presented a branch-and-bound algorithm to optimally solve the problem. The best known heuristic for the GQAP is based on the CHR approach of Ahlaticioglu and Guignard (2008). Recently, Pessoa, et al. (2008) proposed a new branch-and-bound algorithm for the GQAP based on a lagrangean relaxation lower bound. In this work, instances with up to 30 facilities and 20 locations have been solved.

In our newly proposed reduction, the number of generated GQAP facilities is twice the number of MSAP departments and the number of GQAP locations is given by the sum of the number of MSAP floors and the number of MSAP stairways. Hence, an MSAP instance with 13...
departments, 8 floors and 3 stairways would generate a GQAP instance with 26 facilities and 11 locations. In order to solve the problem exactly, we use the same lower bounding algorithm as in (Pessoa, et al., 2008), but with an unsophisticated branch-and-bound method. In our experiments, this new algorithm was able to exactly solve several MSAP instances less time and using fewer nodes than were required by the earlier more complex GQ3AP algorithm.

This paper is organized as follows. In Section 2, we describe our reduction from the MSAP to the GQAP. In Section 3, we describe our experiments, including small changes in the method of (Pessoa et al., 2008) that are necessary to get a good performance for the MSAP converted instances. Finally, in Section 4, we summarize our conclusions and future work.

2 The MSAP reduction to GQAP:

2.1 The Generalized Quadratic Assignment problem

The Generalized Quadratic Assignment Problem (GQAP) objective is to minimize the total cost of assigning $M$ facility to $N$ locations, given the facility’s space requirement, each location’s available space, the flow-units between all facility pairs, the costs of a flow-unit between locations (e.g., floors) and the facility (e.g., department) installation cost. Each facility must be assigned to only one location and the total space available at each location must be respected. The total cost to be minimized is the sum of assignment costs and flow-distance costs.

The objective of the MSAP is to minimize the total cost of assigning $M$ facilities to $N$ locations and to $S$ stairways, given each facility’s space requirement, the number of people in each department, each location’s available space, each stairway’s capacity, the flow-units between all facility pairs, cost of a flow-unit between all pairs of locations and the installation cost for each facility. Each facility must be assigned to only one location and the total space available at each location must be respected. The total cost to be minimized is the sum of assignment costs and flow-distance costs.

The two descriptions show just how similar the GQAP and MSAP problems are. This suggests that a MSAP instance can be reduced to a GQAP instance. The next sub-section shows the reduction method. In order to differentiate the MSAP facilities from the GQAP facilities, we always refer to the former as departments. For the same reason, we always refer to the MSAP locations as floors.

2.2 Notation and Definitions

We use the following notation and definitions.

**GQAP definitions:**

- $M$ – number of facilities
- $N$ – number of locations
- $f_{ik}$ – flow-units from facility $i$ to facility $k$
- $d_{jn}$ – distance from location $j$ to location $n$
- $c_{ij}$ – cost of installing facility $i$ at location $j$
- $a_i$ – space needed to install facility $i$
- $A_j$ – space available at location $j$
MSAP definitions:

- $M'$ – number of departments
- $N'$ – number of floors
- $S'$ – number of stairways
- $f_{ik}'$ – flow-units from department $i$ to department $k$
- $d_{jn}'$ – distance from floor $j$ to floor $n$
- $\epsilon_{jn}'$ – distance between floor $j$ and stairway $s$ exit from building
- $a_i'$ – space needed to install department $i$
- $A_j'$ – space available at floor $j$
- $\lambda_i'$ – Poisson arrival rate of persons at stairway exit in the evacuation from department $i$
- $\mu_s'$ – Service rate capacity of stairway $s$

2.3 Reduction

The main idea is creating different GQAP locations associated to MSAP floors and to MSAP stairways. Then, we create two GQAP facilities for each MSAP department: one is assigned to a floor-location and the other to a stairway-location. The first facility is referred to as a floor-department facility or simply a floor-department, for short. The second facility is referred to as a stairway-department facility or simply a stairway-department, for short. The total number of facilities will be $2M'$, and the total number of locations $N' + S'$.

$M = 2M'$
$N = N' + S'$

The facilities numbered from 1 to $M'$ are the floor-department facilities, and the ones numbered from $M' + 1$ to $2M'$ are stairway-department facilities. The locations numbered from 1 to $N'$ are the floor-locations, and the locations numbered from $N' + 1$ to $N' + S'$ are the stairway-location.

Flow-units between departments ($f_{ik}$)

The flow-units between floor-departments are the same as the flow-units between departments in the original MSAP problem.

$$f_{ik} = f_{ik}' \quad \forall \quad i, k \leq M'$$

The flow-units from a floor-department to a stairway-department are zero, unless both represent the same department of the original MSAP problem. In such case, the flow units represent the flow of people from the department to a stairway in the case of evacuation.

$$f_{ik} = \begin{cases} 
0, & \text{if } k \neq i + M' \\
\lambda_i, & \text{if } k = i + M'
\end{cases}$$

The flow-units from a stairway-department to a floor-department are always zero.

$$f_{ik} = 0 \quad \forall \quad i > M' \text{ and } k \leq M'$$

The flow-units between stairway-departments are always zero.

$$f_{ik} = 0 \quad \forall \quad i, k > M'$$
The GQAP flow matrix representing the MSAP flows is shown in Figure 1

![GQAP flow matrix](image1)

**Distance between locations ($d_{jn}$)**

The distances between floor-locations pairs are the same as the distances between floor pairs given in the MSAP.

$$d_{jn} = d'_{jn} \quad \forall \quad 1 \leq j, n \leq N'$$

The distance from any floor-location to any stairway-location is the distance between a floor and a stairway in the original MSAP.

$$d_{jn} = e_{jn} \quad i = n - N' \quad \forall \quad 1 \leq j \leq N' \quad \text{and} \quad N' < n \leq N' + S$$

The distance from any stairway-location to any floor-location is always zero (it could be any number that wouldn’t make any difference, because there won’t be flow between them).

$$d_{jn} = 0 \quad \forall \quad N' < j \leq N' + S \quad \text{and} \quad 1 \leq n \leq N'$$

The distance between any stairway-location pair is always zero (it could be any number that wouldn’t make any difference, because there won’t be flow between them).

$$d_{jn} = 0 \quad \forall \quad N' < j, n \leq N' + S$$

The GQAP distance matrix representing the MSAP distances is shown in Figure 2.

![GQAP distance matrix](image2)

**Installation Cost ($c_{ij}$)**

Since the MSAP doesn’t consider installation costs, all $c_{ij}$ could be zero, but to make sure floor-departments won’t be placed at stairway-locations and stairway-departments won’t be placed at floor-locations, the costs of those allocations will be set at infinity ($\infty$).
\[
\epsilon_{ij} = \begin{cases} 
\infty, & \text{if } i \leq M \text{ and } j > N \\
\infty, & \text{if } i > M \text{ and } j \leq N \\
0, & \text{if } i \leq M \text{ and } j \leq N \\
0, & \text{if } i > M \text{ and } j > N 
\end{cases}
\]

The GQAP installation cost matrix is shown in Figure 3

![GQAP installation cost matrix adapted from the MSAP](image1)

**Space needed to install departments \((a_i)\)**

For the floor-departments the space needed is the one needed by the department in the MSAP.
\[ a_i = a_i', \quad \forall \quad i \leq M \]

For the stairway-departments the space needed will be the arrival rate from the department at the stairway.
\[ a_i = \lambda_{ij}, \quad \forall \quad i > M, \quad j = i - M \]

The GQAP space needed to install departments is shown in Figure 4

![GQAP space needed to install the MSAP departments](image2)

**Space available at floors \((A_j)\)**

For the floor-locations the space available is the one available at the MSAP floors.
\[ A_j = A_j', \quad \forall \quad j \leq N \]

For the stairway-locations the space available is the capacity available for the MSAP stairways.
\[ A_j = \mu_{ij}', \quad \forall \quad j > N, \quad i = j - N \]
The GQAP space available at each MSAP floor or stairway (for MSAP department representations) is shown in Figure 5.

<table>
<thead>
<tr>
<th>Floor Location</th>
<th>Stairway Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A_1 12</td>
</tr>
<tr>
<td>2</td>
<td>A_2 12</td>
</tr>
<tr>
<td>3</td>
<td>2 A_3 2</td>
</tr>
</tbody>
</table>

Figure 5 GQAP space available at each MSAP floor or stairway

2.4 Rescaling

Rescaling instances can reduce knapsack-processing time. This can be done by dividing spaces available and needed by their greatest common divisor (GCD), observing the following rules:

- \( \lambda_i \) and \( \mu_i \) (Original MSAP)

Rescale the stairways’ capacity (\( \mu_i \)) and the evacuation flow to stairways from departments (\( \lambda_i \)) dividing them by \( \text{Div}_{i} = \text{GCD}(\lambda_i, \mu_i) \).

It’s very important to note that, when the adaptation to GQAP is done, the original \( \lambda_i \) (not rescaled) must be used to create \( f_{ik} \) (flow-units between departments) in order to not disturb the objective function.

- \( a_i \) and \( A_j \) (Original MSAP)

Rescale the space needed by the departments (\( a_i \)) and the space available to them at the floors (\( A_j \)) dividing them by \( \text{Div}_{j} = \text{GCD}(a_i, A_j) \).

3. Experiments

The experimental results that we report here are preliminary. We found several drawbacks when applying the method proposed in (Pessoa et al., 2008) directly to the converted MSAP instances.

First, the lower bound given by the lagrangean relaxation procedure was zero for several instances. It happens that the Volume algorithm is unable to find an improving direction when the initial cost matrix has many zeros. To obtain better lower bounds, we applied an initial random perturbation on the lagrangean multipliers and increased the initial step size. Then, the lower bound starts negative but ends up with a tight value.

Second, we realized that branching on variables associated to the stairway-departments is very ineffective, which was not detected by the branch-and-bound procedure used in (Pessoa et al., 2008). To overcome this, we used a very simple branching rule: always choose the facility with the smallest index. Since the floor-department facilities receive smaller indices, the stairway-department facilities are avoided. An important side effect of this rule is that the method may take too long to find good upper bounds. To avoid this, we have used the optimal values as initial upper bounds. Although this is not a fare comparison against the previous method, which used no initial upper bound, we point out that we could obtain similar results using a node selection rule that gives priority to nodes with smaller lower bounds. Since the number of nodes is always small, this turns out to be a feasible approach for the tested instances but would not be possible for larger instances due to the large amount of memory required.
We tested our algorithm in two sets of instances. The first set is the one used in (Hahn et al., 2008). The second set consists of larger instances proposed by Smith J.M. in a personal communication. All instance names start with the problem dimensions ($M\times N\times S$), followed by qualifiers regarding the stairway and floor utilizations and symmetry considerations (see Hahn et al. (2008) for details).

Table 1 shows a comparison between the new method and the previous one. The execution times of the new method correspond to a 3.0 GHz Intel Core 2 Duo E8400 machine, while the execution times of the previous method are the values reported in (Hahn et al., 2008). The machine used for the new method is about six times faster than the ones used for the previous method.

<table>
<thead>
<tr>
<th>Instance</th>
<th>Optimal</th>
<th>New Method</th>
<th>Previous Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>10x7x2-loose</td>
<td>2582.00</td>
<td>1</td>
<td>2582.00</td>
</tr>
<tr>
<td>10x7x2-tight</td>
<td>2582.00</td>
<td>1</td>
<td>2582.00</td>
</tr>
<tr>
<td>14x7x2-large-tight</td>
<td>2739.50</td>
<td>1</td>
<td>2739.50</td>
</tr>
<tr>
<td>13x8x2-med-loose</td>
<td>689.50</td>
<td>110</td>
<td>652.82</td>
</tr>
<tr>
<td>13x8x2-med-tight</td>
<td>689.50</td>
<td>119</td>
<td>652.47</td>
</tr>
<tr>
<td>14x7x2-large-tight-NE</td>
<td>2739.50</td>
<td>1</td>
<td>2739.50</td>
</tr>
<tr>
<td>15x8x2-med-loose</td>
<td>689.50</td>
<td>110</td>
<td>652.82</td>
</tr>
<tr>
<td>13x8x2-med-tight</td>
<td>689.50</td>
<td>119</td>
<td>652.47</td>
</tr>
<tr>
<td>13x8x2-small-loose</td>
<td>383.35</td>
<td>243</td>
<td>338.94</td>
</tr>
<tr>
<td>13x8x3-large-tight</td>
<td>3086.00</td>
<td>3136</td>
<td>3082.25</td>
</tr>
<tr>
<td>13x8x3-med-loose</td>
<td>630.50</td>
<td>12</td>
<td>618.54</td>
</tr>
<tr>
<td>13x8x3-med-tight</td>
<td>630.50</td>
<td>12</td>
<td>616.14</td>
</tr>
<tr>
<td>13x8x3-small-loose</td>
<td>351.05</td>
<td>25</td>
<td>317.19</td>
</tr>
<tr>
<td>13x8x3-small-tight</td>
<td>351.05</td>
<td>25</td>
<td>315.63</td>
</tr>
<tr>
<td>17x17x4-small</td>
<td>N/A</td>
<td>N/A</td>
<td>1669.80</td>
</tr>
<tr>
<td>17x17x4-med</td>
<td>N/A</td>
<td>N/A</td>
<td>3540.25</td>
</tr>
<tr>
<td>17x17x4-high</td>
<td>4437.70</td>
<td>1</td>
<td>4437.70</td>
</tr>
</tbody>
</table>

Table 1 Comparison between the new method and the previous one

The most important difference between the two methods is the quality of the obtained lower bound. The new method proved the optimality of five instances without branching, and the average gap at the root node was 3.9% against 81.9% of the previous method. Although the new method uses the optimal value as an initial upper bound, the difference between the numbers of nodes is still remarkable. For example, for the 13x8x3-large-tight instance, where the previous method found the optimal solution relatively fast, the new method used little more than 3 thousands nodes, compared with the more than one hundred million nodes required by the earlier GQ3AP algorithm. For the set of larger instances, since no upper bound is known, the new method was able to solve only the ones where the root lower bound is tight.

4. Conclusions and Future Work

We have described a reduction from the MSAP to the GQAP that allows one to use previously proposed methods for the latter problem to solve the former one. Although the application of the GQAP algorithms proposed in (Pessoa et al., 2008) is not obvious, our experiments show that this approach is promising. The obtained lower bounds are dramatically better than the ones reported for the earlier GQ3AP method. Considering that our new solution algorithm involved a very simple branch-and-bound procedure illustrates the effectiveness of the improved bound quality.

As a future work, we intend to adapt the branch-and-bound procedure used in (Pessoa et al., 2008) to perform better for the MSAP converted instances. We also plan to generate larger MSAP instances and convert them to the GQAP format in order to enhance our benchmark. It would also be interesting to test previous heuristics for the GQAP, as the CHR heuristic of (Athlacioglou and Guignard, 2008), with the new instances. As a side effect of this work, we expect to obtain more
robust methods for the GQAP that will hopefully also perform better for the classical GQAP instances.

The GQ3AP model for the MASP is not without merit. The algorithm reported by Hahn, Smith and Zhu for calculating GQ3AP lower bounds is rudimentary at best. Thus, it behooves us to try to improve GQ3AP bound performance with the techniques already well understood in Pessoa et al. 2008.

Acknowledgements

This material is based upon work supported by the U.S. National Science Foundation under grant No DMI-0400155. AAP and GHA were partially financed by CNPq grants 301175/2006-3 and 121940/2008-0, respectively.

References


Lee, C.-G., and Z. Ma (2004), The generalized quadratic assignment problem, Research Report, Department of Mechanical and Industrial Engineering, University of Toronto, Toronto, Ontario M5S 3G8, Canada.
