Layout Optimization in a Fitness Industry: An Application of QAP Formulation and Heuristic Approaches

1Ibrahim Miraç Eligüzel, 1Eren Özceylan and 2Yakup Atasagun

1Department of Industrial Engineering, Gaziantep University, Gaziantep, 27100, Turkey
2Department of Industrial Engineering, Konya Technical University, Konya, 42250, Turkey

Abstract: The Quadratic Assignment Problem (QAP) is the problem of determining a one-to-one and onto assignment between two sets, each consisting of n objects (e.g., n facilities and n locations) so as to minimize the sum of the distance × flow associated with pairs of assignments. In this study, QAP formulation is applied to a fitness center to assign the sports equipment to the related locations on the layout. Facilities are considered as sports equipment and locations are considered the required area for equipment on the layout. A case study with 19 sports equipment is considered. The objective of the study is to allocate the equipment to the 19 available locations to minimize the total walking distance while considering the interactions between sports equipment. An Integer Programming (IP) formulation is applied using GUROBI package program. 2-opt and tabu search algorithms are also applied to find the feasible/optimal solutions in a reasonable time. The results show that the heuristic approaches are more effective than IP in terms of computational time. The optimal result which is obtained by IP, 2-opt and tabu search provides a better layout with 55.47% less distance × flow compared to the current layout.

Keywords: 2-opt, Fitness Center, Integer Programming, Layout Optimization, Quadratic Assignment Problem, Tabu Search

Introduction

There are several studies in literature that consider the effective and efficient solutions for the real life problems. One of them deals with the green inventory routing and aims to minimize of total cost by application of augmented Tabu Search algorithm (Alinaghian et al., 2021). Another study in that concern focuses on the waste collection via multi-trip capacitated arc routing problem and solution is proposed by utilizing the improved Max-Min Ant System (Tirkolaee et al., 2019a). In addition, (Tirkolaee et al., 2019b) conducted a study to solve aforementioned problem with vehicle routing and application of simulated annealing. Another aspect of real life application comprehends the disaster relief. In that aspect, (Davoodi and Goli, 2019) proposed a study which concerns the disaster relief model based on covering tour in Iran and solution is gathered via application of hybrid benders decomposition and variable neighborhood search approach. From the same point of view, an uncertain integrated approach is adopted for simultaneously locating temporary health centers in affected areas, assigning affected regions to these centers and delivering supplies to these centers by utilizing harmony search algorithm (Alinaghian and Goli, 2017). Another problem which has wide range of application is the p-hub median problem. Ghaffarinasab et al. (2020) proposed a study focuses on p-hub median problem under hose and hybrid demand uncertainties and solution is represented with Tabu Search algorithm. In addition to aforementioned studies, a product portfolio problem (Goli et al., 2019), cell formation and production scheduling (Goli et al., 2021), demand prediction (Goli et al., 2020; 2018) and disaster relief locating and routing problems (Goli and Malmir, 2020) are focused in literature for real life applications.

In the proposed study, Quadratic Assignment Problem (QAP) is considered which deals with allocating a set of facilities to a set of locations, with the cost being a function of the distance and flow between the facilities (Erdogan and Tansel, 2011). The QAP was introduced by (Koopmans and Beckmann, 1957) as a mathematical model for the location of a set of indivisible economical activities (Lawler, 1963). Since then, the
QAP, one of the most difficult problems in the NP-hard class, models many real-life problems in several areas such as facility location (Elshafei, 1977), parallel and distributed computing (Czapinski, 2013), combinatorial data analysis (James et al., 2009) and image processing (Schellewald et al., 2007). For a review of methodological and case-study based papers in QAP, the reader is referred to (Loiola et al., 2007) and Bhati and Rasool (2014).

Additionally, several heuristic and meta-heuristic approaches such as particle swarm optimization algorithm (Pradeepmon et al., 2018), artificial bee colony optimization algorithm (Dokeroglu et al., 2019) and integrated genetic algorithm with tabu search (Zhang et al., 2020) have been proposed for QAP due to its NP-hard nature.

As can be seen above, the QAP has been drawing researchers’ attention worldwide, not only because of its complexity and theoretical importance, but also because of its practical ability. In practical studies, facility layout or location problems are at the first place where the QAP is used (Hosseini-Nasab et al., 2018). However, the papers which apply QAP formulation to a service sector problem are very limited (Saremi et al., 2008). One of the studies that is applied on service sector considers the patient rotation in hospital by utilizing QAP (Feng and Su, 2015). Patient rotation is modeled as integer programing and results demonstrates the 11.55% decrease in average walking time for Tongji Hospital. Another study that utilizes QAP problem, focuses on the virtual machine placement for cloud computing services (Stefanello et al., 2019). They proposed two hybrid algorithms, which are greedy randomized adaptive search procedure and a biased random-key genetic algorithm by applying path-relinking strategy and variable neighborhood decent. Both of proposed algorithms demonstrate efficient and effective performance on small and large scale test data. The study by (Zhou et al., 2019) that also considers the virtual machine concentrates on virtual machine migration for the costumers who aims to migrate their services from local clusters to cloud data centers. They generate the multi-objective integer non-linear programming which reduced to multi-object QAP and solved by optimization problem solvers. In addition, (Li et al., 2019) used QAP formulation for multi-access edge computing in which energy efficient joint content placement and scheduling taken into consideration with constraints that includes wireless and backhaul transmissions. Moreover, QAP is considered to prevent the human error by locating medications in automated dispensing cabinets in the study conducted by (Pazour and Meller, 2012).

There are several QAP studies that focus on real life application and apply heuristics as a solution method. For instance, QAP is applied on website structure improvement and ant colony is utilized for solution by (Qahri Saremi et al., 2008). In that aspect, genetic algorithm based approach is applied by (Rahhari et al., 2014) on wind farm layout optimization. In another study proposed by (Abdel-Basset et al., 2018), whale algorithm integrated with tabu search for QAP is proposed and utilized in order to be used for locating hospital departments. In addition, (Tsutsui and Fujimoto, 2009) used parallel genetic algorithm for solving GPU computation.

In this study, the layout of a fitness center is tried to be re-designed. Our motivation to utilize layout of a fitness center derives from to demonstrate the application of existing problem from different perspective and applicability of QAP. The problem is considered as a QAP. In addition, two different heuristic approaches are applied and tested on the problem. While the sports equipment is considered as machines, their required spaces are considered as locations. Members of the fitness center are assumed as products which are visiting the sport equipment (machines) in order. Because of the 19 equipment, the problem size is 19x19. All the required data (flow and distance) are gathered from a fitness center in Gaziantep, Turkey. The QAP is solved as an IP formulation using GUROBI package program. Although IP model provides the optimal solution, 2-opt and tabu search algorithms are applied to reach the optimal solution in a reasonable time. Although 2-opt algorithm is a heuristic approach for local search, there are several studies which apply 2-opt algorithm in QAP papers. Okano et al. (2018) proposed study that investigates the local search techniques on QAP problem and divided 2-opt into two strategies which are first improvement and best improvement. In addition, (Chaparala et al., 2014) conducted study that applies 2-opt on QAP to compute capabilities of current GPUs in which, 2-opt demonstrates computed cost is within the 0.1% of best known value and results are gathered in reasonable time even for large datasets. In terms of tabu search applications in QAP, (Zhu et al., 2010) proposes a single instruction multiple data tabu search by utilizing graphic hardware acceleration on QAP and they achieved the reach solution 20-45 time faster than standard CPU code. Dejam et al. (2012) generates the algorithm by combining cuckoo and tabu search and they received more optimized solution through applications than single algorithms such genetic and honey bee algorithms. Another study conducted by (Fescioglu-Unver and Kokar, 2011) applies two strategies to improve tabu search on QAP instances.

A brief review above indicates that there is a still gap in the QAP application in service systems. To the best knowledge of the authors, the observed contributions are summarized below:

- Considering a fitness center layout problem as a QAP firstly in the literature
- Bringing a real life case problem to the QAP literature
- Proposing and testing two different heuristic/meta-heuristic approaches
In this study, the applied mathematical formulation and the algorithms of 2-opt and tabu search are described through section 2. After that the real case of a fitness center is introduced in section 3. The results of three approaches and comparison with the current layout are discussed in section 4. Section 5 briefly concludes the paper and gives the future research directions regarding QAP.

Methodology

In this section, the applied approaches namely integer programming, 2-opt and tabu search algorithms are described, respectively.

Mathematical Formulation of QAP

In this study, the initial formulation by (Koopmans and Beckmann, 1957) is used. The notations of the model are given in Table 1.

The Koopmans-Beckmann formulation of the QAP is as follows:

\[
\begin{align*}
\text{Model:} \\
\min \sum_{i,j} a_{ij} b_{ij} x_{ik} x_{jl} \\
\text{Subject to:} \\
\sum_{i=1}^{n} x_{ij} = 1 \quad \forall j = 1, \ldots, n \\
\sum_{j=1}^{n} x_{ij} = 1 \quad \forall i = 1, \ldots, n \\
x_{ij} \in \{0,1\} \quad \forall i, j = 1, \ldots, n
\end{align*}
\]

Table 1: Notations

<table>
<thead>
<tr>
<th>Sets</th>
<th>i, j</th>
<th>Facility (sports equipment)</th>
</tr>
</thead>
<tbody>
<tr>
<td>k, l</td>
<td></td>
<td>Locations (an area on the fitness center layout)</td>
</tr>
<tr>
<td>Parameters</td>
<td>n</td>
<td>Number of facilities (number of locations)</td>
</tr>
<tr>
<td>a_{ik}</td>
<td>The distance between locations k and l</td>
<td></td>
</tr>
<tr>
<td>b_{ij}</td>
<td>The flow (number of members) per unit time between facilities i and j</td>
<td></td>
</tr>
<tr>
<td>Decision variable</td>
<td>x_{ik}</td>
<td>1, facility (sports equipment) i is assigned to location (an area on the fitness center layout) k; 0, otherwise</td>
</tr>
</tbody>
</table>

The pseudo code of 2-opt algorithm

```
iteration ← 1
while iteration < max_iter do
    Create a random solution T (T represents a single dimensional sequence of sports equipment assuming that each element of the sequence is assigned to areas A to S respectively, such as the first element of T is assigned to area A, the second element of T is assigned to area B and so on.)
    Calculate Obj(T) = objective function value of T
    T_{best} ← T, Obj_{best} ← Obj(T)
    repeat
        Max_impr ← 0
        for i = 1 to n-1 do
            for j = i+1 to n do
                Calculate Impr(ij) = Improvement on the Obj(T) when i_th and j_th elements of the T are swapped
                if Impr(ij) > Max_impr
                    Max_impr ← Impr(ij)
                    i∗ ← i
                    j∗ ← j
                end if
            end for
        end if
        if Max_impr > 0
            T ← swap i∗th and j∗th elements of T
            Calculate Obj(T)
        end if
    end repeat
    if Obj(T) < Obj_{best}
        Obj_{best} ← Obj(T)
        T_{best} ← T
    end if
    iteration ← iteration + 1
end while
Report Obj_{best} and T_{best}
```
The objective function (Eq. 1) is to assign each sports equipment to a location on the fitness center layout such that the total flow × distance is minimized. Constraints (2) and (3) imply that sports equipment can be assigned to only a location and vice versa. Constraint (4) is the sign constraint shows the binary decision variable. Due to nonlinearity of the objective function, many authors have proposed methods for linearizing the quadratic form of the objective function (Eq. 1) by introducing additional variables. The linearization method of (Frieze and Yadegar, 1983) is used in this study.

2-opt Algorithm

The 2-opt algorithm which is one of the most famous simple local search algorithms was first proposed by (Croes, 1958) for Travelling Salesman Problem (TSP).

The algorithm starts with an initial TSP tour and considers the swapping of the first and second nodes of the initial solution. If the resulting solution’s objective function value is better than that of the initial solution, then these node pair is stored as a candidate for future consideration; otherwise, it is discarded. Then the algorithm considers the swapping of the first and the third nodes. If this exchange generates a better solution, then it is stored as a candidate for future consideration; otherwise, it is discarded and so on. Thus, whenever a better candidate node pair is found, the algorithm discards the previous best candidate. This procedure continues until all the pair-wise exchanges are considered (Bashiri and Karimi, 2012).

In this study, the 2-opt algorithm is applied to above mentioned fitness center layout problem as follows: The solution is encoded as a single dimensional sequence of the sports equipment such as a single dimensional node visiting sequence of a TSP problem. While decoding, it is assumed that each element of the sequence is assigned to areas (locations) A to S respectively. For example the first element of the encoded solution is assigned to area A, the second element is assigned to area B and so on. Finally the objective function is calculated by summing flow × distance.

2-opt algorithm produces a single improved solution for an initial solution. Thus, for the above mentioned fitness center layout problem, the 2-opt algorithm is repeated for a number of iterations in which the algorithm starts with different initial solutions. Pseudo code of the 2-opt algorithm for the fitness center layout problem is given in Table 2.

Tabu Search

Tabu search algorithm is introduced by (Glover, 1989), which focuses on the local search. The main part of the Tabu search algorithm is updating the best solution list during the searching and whole solutions gather a priority value. In addition, Tabu search includes a tabu list to store the history of the search process, a mechanism for acceptance or rejection of a new allocation in the neighborhood and a mechanism that leads the change procedure between neighborhood diversification and intensification strategies (Loiola et al., 2007). Pseudo code of applied Tabu search in proposed study is given in Table 3.

Table 3: The pseudo code of Tabu search algorithm

| neighbors, number of iteration, N = number of generated neighbors |
| current_solution = create initial solution randomly with respect to dimension of given input data (flow and distance matrix) |
| number of iteration = decide on maximum number of iteration |
| Create a list for short term memory |
| Create a list for long term memory |
| Create threshold length value for short term memory |
| while number of iteration > 0: |
| Best_solution = current_solution |
| move on the neighbor solutions from current_solution |
| cost = store the sequence of neighbor solutions for index to length(neighbors) |
| calculate the cost for each neighbors |
| sort the neighbors by cost |
| rank = sorted list of the neighbors by cost |
| for j to N:
| check the neighbor is already in created solution or not |
| if neighbor is not in created solution: |
| if current_solution not in long term memory: |
| Penalize the current_solution with 1 |
| if objective value of current_solution < objective value of best_solution: |
| best_solution = current_solution |
| else: |
| cur_cost = objective value of current_solution penalized by 1 |
| increment the long term memory for the current visit |
| if cur_cost < best_solution: |
| best_solution = current_solution |
| break |
| elif objective value of neighbors[j, :-2] < objective value of best_solution: |
| current_solution = neighbors[j, :-1] |
| append the neighbors[j, -2:] to short term memory |
| if length of short term memory > threshold length value for short term memory: |
| update short term memory |
| if current_solution not in long term memory without 1st element |
| if current_solution not in long term memory: |
| penalize current_solution by 1 (set key penalty to one) |
| best_solution = current_solution |
| else: |
| cur_cost = objective value of current_solution + penalization in long term memory for current_solution |
| increment the frequency for the current visit |
| if cur_cost < objective value of best_solution: |
| best_solution = current_solution |
| number of iteration += 1 |

443
Fig. 1: Flowchart of tabu search algorithm

- **Parameter setting**
  - Generate initial solution
- **Generate a candidate list of current solution as short-term and long-term memory**
  - Number of iteration < maximum number of iteration
    - Increase the number of iteration by 1
    - If current solution < objective value of best solution, best_solution = current_solution
    - If current solution = objective value of current solution + penalization in long-term memory for current solution, increment the frequency for the current visit
    - Current solution = worst neighbor solution
    - If objective values of worst two neighbor solution < best solution, take the list of short-term memory from the 1st element
  - If length of short-term memory > threshold length value for short-term memory, update short-term memory without 1st element
  - Calculate fitness values
  - Short the neighbors by fitness values
    - Check if neighbor is already in created solution
      - If yes, take the list of short-term memory from the 1st element
      - If no, penalize the current solution with 1 and increment the long-term memory for the current visit
  - If penalized current solution < objective value of best solution, best_solution = current_solution
  - Objective value of current solution < objective value of best solution, best_solution = current_solution
  - If current solution not in long-term memory, penalize current_solution by 1 (set key penalty to one) best_solution = current_solution
  - If current solution < objective value of best_solution, best_solution = current_solution
  - If length of short-term memory > threshold length value, update short-term memory without 1st element
  - Take the list of short-term memory from the 1st element
  - Objective value of current solution < objective value of best solution, best_solution = current_solution
  - End
Detailed demonstration of tabu search is given in Fig. 1. It can be said that there is a decision box that checks whether the solution is already generated or not. According to the answer, procedure of tabu search implemented, which are aspiration criteria and tabu list operations) till the maximum number of iteration reached.

Case Study

The layout optimization of fitness center in Gaziantep University campus is considered in this study. There are totally 19 sports equipment in the fitness center. Sports equipment is shown in Fig. 2 with their codes.

Due to the number of sports equipment, the layout of the fitness center is divided into 19 parts from A to S as shown in Fig. 3. The current layout of the fitness center is also shown in Fig. 4. For instance, the sport equipment BEP is located on the A area.

Although there is a current layout, the manager asks a new layout which minimizes the total distance/time of people. Due to narrowness of the place, crowded and looking for next sport equipment is not desired. The problem is assigning 19 sports equipment (Fig. 2) to 19 places (Fig. 3) while minimizing the total walking distance. There are mainly two parameters for the QAP as mentioned in the previous section. \( A = (a_{ij})_{n \times n} \) which corresponds the distances between areas are calculated using AutoCAD. The distance matrix is given in Table 4.

It must be noted that the distance matrix is symmetric.

![Fig. 2: 19 sports equipment in the fitness center](image1)

![Fig. 3: Layout of the fitness center](image2)

![Fig. 4: Current layout of the sports equipment](image3)
The second matrix for the QAP is related with the relation between sports equipment (stations). To do so, sports programs which are followed by the members are investigated. There are totally 10 different programs which visit different sports equipment in different orders. 10 sports programs are given in Table 5 (from/to table) with their corresponding sports equipment. According to Table 5, while a member who follows the first program has to visit BOL-ELB-LEX-CRC-VRLP-VB and PRT sport stations, a member who follows the second program has to visit SHP-LPR-PRT and SRO stations. All programs are followed by different numbers of members. Therefore, the activity relations between equipment are calculated by multiplying the numbers in Table 5 and the number of members of each sport program. The final $B = (\tilde{D}_{ij})_{n \times n}$ matrix which correspond the flows between equipment is given in Table 6. For instance, 18 members are visiting CRC equipment after BOL equipment in one day.
Apart given the parameters above, the assumptions of the study are mentioned below:

- The required area of each sports station is equal and same
- Every member follows the sports program in given order on each day
- Waiting for the equipment is ignored

Results and Discussion

The case given in previous section is solved using three different solution techniques in this section. All runs are taken on a server with 3.00 GHz Intel Core processor and 2 GB of RAM. The IP formulation is applied using GUROBI 9.1 package program. The optimal solution is obtained as 427,800.00. The computation time required to solve the model is 4,400.00 sec. The optimal assignment is shown in Table 7 and Fig. 5.

When the optimal layout is compared with the current layout, it is clear to see that all locations of sports equipment are changed. Total traveled distance × flow equals to 960,700.00 in the current layout. Optimal solution reduces the total distance × flow value to 427,800.00 by 55.47%. Although the IP finds the optimal solution, the computational time which is greater than one hour does not satisfy the decision maker. Therefore, 2-opt and tabu search algorithms are tested on the same case. 2-opt is coded using C#. The iteration is limited as 500. The algorithm is run ten times and optimal solution is obtained in all runs. The optimal solution is found in 2.5 sec less than 500 iterations. Tabu search algorithm is also applied to make a comparison between 2-opt algorithm. Tabu search algorithm is coded by utilizing Python. The iteration is limited as 8000. The algorithm is run ten times as 2-opt algorithm. However, optimum solution is gathered only seven of the ten runs which is shown in Table 8. According to Table 8, the computation time varies between 8.50 and 67.00 sec. Figure 6 demonstrates the number of iterations and corresponding objective values. From the Fig. 6, it can be said that tabu search algorithm fluctuates till 7000 iterations. However, it becomes roughly stable after 7000 iterations.

![Fig. 5: Optimal layout of the sports equipment](image)

![Fig. 6: Convergence graph of tabu search algorithm](image)
The optimal solution should be considered as unequal, (ii) a decision support system should be also developed to provide flexibility for decision makers, (iii) an exact solution approach should be developed for large sized problems and (iv) the model should be applied to large GYM centers that include more equipment.

Acknowledgment

The authors would like to sincerely thank and acknowledge the guidance of the anonymous reviewers of the journal.

Table 7: Optimal assignments

<table>
<thead>
<tr>
<th>Location</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>J</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equipment</td>
<td>ECT</td>
<td>SHE</td>
<td>VB</td>
<td>TRO</td>
<td>PRE</td>
<td>VRLP</td>
<td>BEP</td>
<td>SRO</td>
<td>LPR</td>
<td>TRB</td>
</tr>
<tr>
<td>Location</td>
<td>K</td>
<td>L</td>
<td>M</td>
<td>N</td>
<td>O</td>
<td>P</td>
<td>Q</td>
<td>R</td>
<td>S</td>
<td></td>
</tr>
<tr>
<td>Equipment</td>
<td>MAT</td>
<td>ELB</td>
<td>BOL</td>
<td>LEX</td>
<td>CCR</td>
<td>PRT</td>
<td>DSL</td>
<td>SHP</td>
<td>LCR</td>
<td></td>
</tr>
</tbody>
</table>

Table 8: Summary of tabu search runs

<table>
<thead>
<tr>
<th>Runs</th>
<th>Found Solution</th>
<th>Elapse Time (sec)</th>
<th>Found Solution</th>
<th>Elapse Time (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>469,000.00</td>
<td>23.60</td>
<td>427,800.00</td>
<td>13.40</td>
</tr>
<tr>
<td></td>
<td>427,800.00</td>
<td>8.50</td>
<td>469,000.00</td>
<td>65.00</td>
</tr>
<tr>
<td></td>
<td>427,800.00</td>
<td>8.00</td>
<td>427,800.00</td>
<td>67.00</td>
</tr>
<tr>
<td></td>
<td>427,800.00</td>
<td>9.00</td>
<td>427,800.00</td>
<td>51.60</td>
</tr>
<tr>
<td></td>
<td>427,800.00</td>
<td>10.00</td>
<td>427,800.00</td>
<td>24.40</td>
</tr>
</tbody>
</table>

Conclusion

In this study, the allocation of sports equipment to the locations in a fitness center is studied. The considered problem contains 19 sport stations (equipment) for 19 alternative locations. Two required information namely the distance matrix between locations and the flow matrix (the number of members who visit the equipment in order) between sports equipment are obtained. The problem is modeled as a QAP and an integer programming formulation is applied. Although the optimal solution of the problem is obtained by the integer programming model using Gurobi package, two heuristic approaches namely 2-opt and tabu search algorithms are also applied to find the feasible/optimal solutions in a reasonable time. Optimal solution is also achieved using 2-opt and tabu search algorithms in very short computational times (less than 10 seconds) compared to the integer programming model (4.400 sec). The optimal solution shows that the new layout is 55.47% better than the current layout in terms of the total traveled distance × flow. From the manageral perspective, it provided that maximum profit of fitness center is accomplished by accurate layout out design and customer satisfaction is managed by reducing waiting time for each station. Therefore, more accurate managerial decisions can be made by gathering efficient and effective layout.

For the future studies, (i) the required spaces of each sports equipment should be considered as unequal, (ii) a decision support system should be also developed to provide flexibility for decision makers, (iii) an exact solution approach should be developed for large sized problems and (iv) the model should be applied to large GYM centers that include more equipment.

Author’s Contributions

İbrahim Miraç Eligüzel and Yakup Atasagun: Participated in drafting the research problem, outlining the solution procedure, and checking the methodology and results.

Eren Özceylan: Helped in writing the article.

Ethics

This article is original and contains unpublished material. The corresponding author confirms that all of the other authors have read and approved the manuscript and no ethical issues involved.

References


