A Survey of the Quadratic Assignment Problem, with Applications

Clayton W. Commander
The University of Florida

April 4, 2003

Abstract

The Quadratic Assignment Problem (QAP) is one of the most interesting and most challenging combinatorial optimization problems in existence. This thesis will be a survey of the QAP. An introduction discussing the origins of the problem will be provided first. Next, formal problem descriptions and mathematical formulations will be given. Issues pertaining to the computational complexity of the QAP, lower bounds and exact algorithms will also be addressed. Some commonly used heuristic procedures will then be introduced. Finally, some applications of the QAP will be analyzed.

1 Introduction

What is the optimal way to wire a computer backboard? How are the locations of clinics within a hospital decided? What possible linkages could there be between these two problems? Most would agree that at first glance, they are seemingly unrelated beyond the fact that both are decision problems. One might even propose that such decisions are made arbitrarily. However, it is the solution to these and countless other problems that contains the key to their correlation. They are all modeled by one of the most challenging problems in *combinatorial optimization*. This problem has been a focus of researchers for over four decades; it is known as the *quadratic assignment problem*.

The quadratic assignment problem (QAP) was originally introduced in 1957 by Tjalling C. Koopmans and Martin Beckman [26] who were trying to model a facilities location problem [10]. Since then, it has been among the most studied problems in all of combinatorial optimization. Many scientists including mathematicians, computer scientists, operations research analysts, and economists have used the QAP to model a variety of optimization problems.

This thesis will present a general overview of the QAP, emphasizing many of its applications. An introduction providing a brief historical overview will be given first. Next, formal problem descriptions and mathematical formulations will be provided. Section 3 will be a discussion of the computational complexity issues associated with the problem. Lower bounds and exact algorithms will be the focus of section 4. Next, some commonly utilized heuristic procedures will be introduced. The focus of section 6 will be the analysis of some selected applications of the QAP. In two cases, the means by which the QAP was applied and any resultant findings will be examined in great detail.

As stated above, Koopmans and Beckman (K-B) first derived the quadratic assignment problem while they were attempting to model a facilities location problem. In [26], K-B argue that in industry, efficient allocation of indivisible resources "are in many cases at the root of increasing returns to the scale of production, whether arising within the plant or firm, or in relation to a cluster of firms through so-called 'external economies'."

K-B realized however, that the mathematical complications that would arise in an attempt to define a general theory about location problems would be immense. Therefore, they restricted their study to a few individual problems, hoping that future research would expand the knowledge of the problem develop a general theory of location.

For over four decades, scientists have been studying the QAP, and have made significant discoveries in the study of assignment problems. Over the years, the QAP has been used to model such things as hospital design [13], computer backboard design [4, 38], scheduling problems [8, 18], and of course location problems [26].

2 Problem Formulations

Before rigorously defining the problem statement, a general discussion of both the linear assignment problem (LAP) and the quadratic assignemnt problem (QAP) will be given. The latter being a more complicated generalization of the former, and the topic of this survey. An explanation of the kiy differences between the two aforementioned problems will be given. The author feels that providing such an introduction will help readers to better understand the idea behind the more complicated definition of the QAP.

2.1 Problem Descriptions

2.1.1 The Linear Assignment Problem

A commonly used intuitive introduction to the assignment problem as used by Hanan and Kurtzberg [25], involves the assignment or n people to n jobs. For each job assignment, there is a related cost, c_{ij}, of assigning person i to job j. The objective is to assign each person to one and only one job in such a manner that minimizes the sum of each assignment cost, i.e., the total cost.

Mathematically, the above problem can be formulated as follows:

$$\min \sum_{i=1}^{n} c_{i\pi(i)},$$

over all permutations $\pi \in S_n$, where S_n is the set of permutations of $\{1, 2, \ldots, n\}$, and $j = \pi(i)$ is the job assignment of person i. Notice that each set of assignments is a permutation of a set on n integers; hence, there are $n!$ distinct permutations from which to choose the optimal assignment, i.e., there are $n!$ distinct ways in which n jobs can be assigned to n people. As noted in [5] notice that for large values of n, a brute force approach of *enumeration*, or examining all possible permutations, is simply not feasible. For example, if one were to attempt to assing $n = 10$ people to 10 jobs, in the manner described above, they would quickly be deterred by the fact that they would have to examine 10!, or approximately 3.63 million different permutations. Clearly, more efficient algorithms must be employed when attempting to solve nontrivial forms of the LAP.

2.1.2 The Quadratic Assignment Problem

As previously mentioned, the focus of this paper is more complicated generalization of the linear assignment problem, known as the quadratic assignment problem. In addition to a *cost matrix*, as in the LAP above, there is also a so-called *distance matrix* involved. In order to preserve consistency, we will once again refer to Hanan and Kurtzberg [25] and their interpretation of the QAP which uses the assignment of offices to people. In the QAP, we are given a cost matrix $C = [c_{ij}]$, where c_{ij} is the measure of the affinity between person i and person j. We are also given a distance matrix $D = [d_{kl}]$, where d_{kl} represents the distance between office k and office l. Assume that person i is assigned to office $p(i)$, and that person j is assigned to office $p(j)$. Then the cost associated with this assignment is taken as $c_{ij}d_{p(i)p(j)}$. Thus, the total cost of all office assignments will be the sum of each $c_{ij}d_{p(i)p(j)}$ over all i, j. The optimal assignment will be that one in which the total cost is minimal. "If the affinity represents the amount of 'face-to-face communication', then the assignment which we desire is the one which minimizes the total amount of walking distance for the people" [25].

As with the LAP, there are $n!$ permutations from which to choose the optimal assignment. However, the reader should be made aware that there is a key difference between these two problems which makes the QAP considerably more difficult to solve. Unlike the LAP in which the assignment of job j to person i was made independently of the assisnments of the other employees, with the QAP the assignments are **not** independent. That is, when considering an assignment of person i to office k, one must consider the assignments of all other people who have some nonzero affinity for person i.

2.2 Formal Problem Statements

2.2.1 Koopmans-Beckman QAP

Let C and D be two $n \times n$ matrices such that $C = [c_{ij}]$ and $D = [d_{ij}]$. As above, consider the set of positive integers $\{1, 2, \ldots, n\}$, and let S_n be the set of permutations of $\{1, 2, \ldots, n\}$. Then the quadratic[1] assignment problem can be defined as follows:

$$\min \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} d_{\pi(i)\pi(j)},$$

over all permutations $\pi \in S_n$. The above formulation is known as the *Koopmans-Beckman QAP* [26].

As a matter of convinence, the convention of [C]ela [10] will be adopted, and this problem will be refered to as QAP(C,D). Stated in words, the objective of the quadratic assignment problem with cost matrix C and distance matrix D is to find the permutation $\pi_0 \in S_n$ that minimizes the double summation over all i, j. As in [10], it should be understood that the notation $d_{\pi(i)\pi(j)}$ as used above, refers to permuting the rows and columns of the matrix D by some permutation π. That is, $D^{\pi} = [d^{\pi}_{ij}] = d_{\pi(i)\pi(j)}$, for $1 \leq i, j \leq n$. In the same manner, given an n-dimensional vector $V = [v_i]$, a permutation of the elements of V by a permutation π will be denoted as $V^{\pi} = [v^{\pi}_i] = v_{\pi(i)}$ [10].

[1] The use of the descriptive "quadratic" is to denote that the cost function contains a term of degree two.

2.2.2 A Quadratic 0-1 Formulation

What follows is an equivalent formulation of the QAP as a quadratic 0-1 integer program. This formulation was originally used by Koopmans-Beckman [26]. The formulation is based on the one-to-one relationship between the permutation $\pi \in S_n$ and a set of so-called *permutation matrices* defined as follows. Let $X = [x_{ij}]$ be an $n \times n$ matrix. Then X is called a permutation matrix if it satisfies the following three conditions:

$$\begin{aligned} \sum_{i=1}^{n} x_{ij} = 1, &\qquad j = 1, \ldots, n; \\ \sum_{j=1}^{n} x_{ij} = 1, &\qquad i = 1, \ldots, n; \\ x_{ij} \in \{0,1\}, &\qquad i,j = 1, \ldots, n. \end{aligned}$$

If the above conditions are met, then QAP(C,D) can be formulated as follows.

$$\min \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{n}\sum_{l=1}^{n} c_{ij} d_{kl} x_{ik} x_{jl}$$

$$\begin{aligned} \text{s.t. } \sum_{i=1}^{n} x_{ij} = 1, &\qquad j = 1, \ldots, n; \\ \sum_{j=1}^{n} x_{ij} = 1, &\qquad i = 1, \ldots, n; \\ x_{ij} \in \{0,1\}, &\qquad i,j = 1, \ldots, n. \end{aligned}$$

2.2.3 Trace Formulation

What follows is a brief discussion of another formulation of a given QAP(C,D) which is based on the traces of the matrices C and D. This formulation will be referred to when discussing the formulation of lower bounds in a later chapter. Recall that the *trace* of a square matrix is defined as the sum of its diagonal elements. That is, given an $n \times n$ matrix A, then $trace(A) = \sum_{i=1}^{n} a_{ii}$. Given cost, distance, and permutation matrices as previously defined, then QAP(C,D) is equivalently defined as

$$\min \, trace(CXD^tX^t),$$

$$\text{s.t. } X \in \Pi_X,$$

where Π_X represents the set of permutation matrices, and $\cdot^t$ is the transpose of the given matrix.

3 Computational Complexity

In this section, some issues pertaining to the computational complexity of the QAP will be discussed. As seen in the previous section, enumeration of all $n!$ feasible solutions leads to an overwhelming number of permutations one would have to search to find the optimal solution, suggesting that the QAP is indeed a formidable problem. In fact, the QAP belongs to the class of computationally hard problems known as ***NP*-complete**.

The proof that the QAP is indeed *NP*-complete was first shown by Sahni and Gonzalez [39] in 1976. Belonging to this class of problems suggests that an algorithm which solves the problem to optimality in polynomial time is unlikely to exist [16]. What's more is that Sahni and Gonzalez [39] also proved that any routine that finds even an ϵ-approximate solution is also *NP*-complete, thus making the QAP among the "hardest of the hard" of all combinatorial optimization problems. In [32], Pardalos et al. explain how other famous problems from the class *NP*-hard such as the traveling salesman problem and the band-width reduction problem are special cases of the QAP.

4 Lower Bounds and Exact Algorithms

4.1 Lower Bounds

The most studied topic on the QAP is the calculation of lower bounds [10]. The importance of lower bounds is two-fold. Not only are they an essential component of branch and bound procedures, which will be introduced in §4.2, they are also used to evaluate the goodness of solutions produced by heuristics. When using branch and bound procedures, both the tightness of the bounds and the associated computing time are considered[2] [32]. When testing heuristics, generally the thightness of the bound is most emphasized [10]. What follows is a brief discussion of the three main classes of lower bounds: Gilmore-Lawler bound [19, 27], eigenvalue related bounds [17, 37], and bounds based on reformulations [2, 9]. A brief conclusion will then be given mentioning some other bounding procedures with references to them.

4.1.1 The Gilmore-Lawler Bound

The Gilmore-Lawler bound (GLB) was one of the first lower bounds ever proposed for the QAP [18, 27]. The GLB places a lower bound on the

[2] Note that bounds which are both tight and computationally cheap to calculate have not yet been discovered [9].

optimal solution of QAP(C,D) based on the solution of a linear assignment problem whose cost matrix elements are generated by some special inner products defined between the elements of C and D [10, 32]. The GLB is simple and quick to compute, requiring only O(n3) computation time for a Koopmans-Beckman QAP. The downside to the GLB is that it is not tight. In general, tightness of the GLB is inversely proportional to n, the number of instances [10].

4.1.2 Eigenvalue Related Bounds

Creating lower bounds for a given QAP(C,D) based on the eigenvalues of C and D has been researched extensively [17, 22, 23, 37]. All such bounds are based on the trace formulation of QAP(C,D) (see §2.2.3). These eigenvalue based bounds are generally the best bounds as far as tightness is concerned; however, they are computed using an iterative process with each iteration requiring $O(n^3)$ computing time [32]. Such high computation time often removes the option of using the tighter eigenvalue based bounds when applying a branch and bound procedure.

4.1.3 Reformulation Based Bounds

So-called reformulation bounds are computed by an iterative process as in the case of the eigenvalue based bounds mentioned above. Not surprisingly, the reformulation bounds, like the eigenvalue based bounds, are also expensive to compute. For each iteration, $n^2 + 1$ linear assignment problems of size n must be solved. Since the running time for the k^{th} iteration is $O(kn^5)$, this is another class of bounds which are not efficiently calculated [32].

4.1.4 Other Bounding Procedures

There are other classes of bounds which are different from the three major classes listed above. In the early 1990's, a new class of lower bounds for the QAP was introduced by Li, Pardalos, Ramakrishnan and Resende [29], which are based on optimal partitioning schemes. Their bounding procedure is relatively inexpensive, requiring only $O(n^3)$ time and produces effective bounds for branch and bound procedures. In fact, the GLB (see §4.1.1) is a special case of those proposed by Li, Pardalos, et al. [29]. Other lower bounding procedures which are based on dual formulation [24], and linear programming relaxations [35] are also effective.

4.2 Exact Algorithms

There are three main methods used to find the global optimal solution for a given QAP: *dynamic programming, cutting plane techniques, and branch and bound procedures*. Research has shown that the latter is the most successful for solving instances of the QAP. Even still, due to the overwhelming complexity of the QAP, problems of size greater than $n = 15$ remain nearly intractable [32]. Since branch and bound procedures are generally the most helpful for solving QAPs, this section will be restricted to providing a description of such algorithms. In [33], Pitsoulis gives an excellent description of the branch and bound technique, which will now be mirrored.

Branch and bound algorithms receive their name from an intuitive description of how they are executed. First, a heuristic procedure is used to generate a suboptimal, but suitable, initial feasible solution. This initial solution is used as an upper bound. Next, the problem is separated into a finite number of subproblems, with a lower bound being established for each. A so-called *search tree* is formed by the repetition of the decomposition/lower bounding being process applied to each subproblem. However, many of the newly formulated subproblems are not considered due to a pre-established lower bound [33]. What is happening is that an optimal permutation is being constructed iteratively, one element at a time. Branch and bound techniques have evolved greatly over the past 40 years[3], starting with Gilmore [19] who in 1962 solved a QAP of size $n = 8$, to the solution of the nug30, a QAP of size $n = 30$ in 2000 by Anstreicher, et al. [1].

5 Heuristics

5.1 Suboptimal Algorithms

In the first section of this chapter, an introduction will be given of some *heuristics*, or suboptimal algorithms that are often used to estimate solutions for instances of the QAP. These procedures, while not providing the global optimal solution, can produce good answers within reasonable time constraints. The discovery of new heuristics which provide good answers quickly are highly sought after. There are five basic categories of heuristics for the QAP:

- Construction methods.
- Limited enumeration methods.

[3]For a detailed discussion of the evolution of the QAP, see [9]

- Improvement methods.
- Simulated annealing techniques.
- Genetic algorithms.

5.1.1 Construction Methods

Construction methods create suboptimal permutations by starting with a partial permutation which is initially empty. The permutation is expanded by repetitive assignments based on set selection criterion until the permutation is complete. One of the oldest heuristics in use is a construction method algorithm. The CRAFT (Computerized Relative Allocation of Facilities Technique), used for the layout of facilities was first introduced by Armour and Buffa [2] in 1963.

5.1.2 Limited Enumeration Techniques

Limited enumeration methods are motivated when one expects that an acceptable suboptimal solution can be found early during a brute force enumeration examination [32]. Such an enumeration could be terminated by imposing either a time limit or an iteration limit. Also, lowering the upper bound when no improvement is found after a number of steps will result in larger jumps in the search tree (see §4.2), thus speeding up the process.

5.1.3 Improvement Methods

Improvement methods are the most researched class of heuristic [32]. The two methods which are the most popular are the local search and the tabu search. Both methods work by starting with an initial basic feasible solution and then attempting to improve it. The local search iteratively seeks a better solution in the neighborhood of the current solution, terminating when no better solution exists within that neighborhood [33]. The tabu search [20, 21] works similarly to the local search; however, it is sometimes more favorable since it was designed to overcome the problem of a heuristic getting trapped at local optima.

5.1.4 Simulated Annealing Methods

This group of heuristics, which is also used for overcoming local optima, receives its name from the physical process which it imitates. This process, called annealing moves high energy particles to lower energy states with the lowering of the temperature, thus cooling a material to a steady state.

Initially, in the initial state of the heuristic, the algorithm is lenient and capable of moving to a worse solution. However, with each iteration the algorithm becomes stricter requiring a better solution at each step [33]. For more on these methods, see [10, 5, 11].

5.1.5 Genetic Algorithms

Genetic algorithms also receive their name from an intuitive explanation of the manner in which they behave. This explanation is based on Darwin's theory of natural selection [32]. Genetic algorithms store a set of solutions and then work to replace these solutions with better ones based on some fitness criterion, usually the objective function value [33]. Genetic algorithms are parallel and are helpful when applied in such an environment [32, 33].

5.1.6 Greedy Randomized Adaptive Search Procedure (GRASP)

GRASP is a relatively new heuristic used to solve combinatorial optimization problems. At each iteration, a solution is computed. The final solution is taken as the one which is the best after all GRASP iterations are performed[4]. The GRASP was first applied to the QAP in 1994 by Li, Pardalos, and Resende [28]. They applied the GRASP to 88 instances of the QAP, finding the best known solution in almost every case, and improved solutions for a few instances [32].

5.2 Generating Test Problems

Generating QAPs with known optimal permutations is a valuable tool to possess when one wants to test the quality of a new heuristic. One of the first of these generators was introduced in 1988 by Palubetskis in [31]. In 1992, Li and Pardalos proposed the so-called Li&Pardalos' generator [30]. Li&Pardalos' generator produces instances of QAPs with known optimal solutions, of which those instances produced by the Palubetskis' generator are special cases [10].

6 Applications

What follows is a detailed analysis of two seemingly unrelated decision problems and it will be shown that they can both be modeled as quadratic assignment problems. A brief history of each problem, an explanation of the

[4]For detailed explanations of the GRASP, see [15] [28], or [33].

research conducted, and a discussion of any results will follow. Finally, a list of other problems that are modeled as QAPs will be given.

6.1 Steinberg Wiring Problem

The first question asked in §1 was about the optimal wiring of computer backboards; that is, the placing of the components of a computer backboard in such a manner that the total length of interconnecting wiring is minimized [4]. Minimizing the total length of the wiring will improve computing time, and is cost effective for the manufacturer of the backboard. These reasons among others have made this problem a major research topic of computer scientists, electrical engineers, and operations research analysts for over 40 years. The problem was first introduced in a 1961 paper by Leon Steinberg [40], a research scientist at the St. Paul, Minnesota, think tank Remington Rand Univac [14]. The general problem of backboard wiring was later dubbed the Steinberg Wiring Problem after his original contribution. In his paper, Steinberg attempted to optimally place 34 components having a total of 2625 interconnections onto a backboard with 36 positions [40]. A geometric interpretation of the backboard is given below in Figure 6.1. The objective is to minimize the total length of wire used to interconnect the components. The particular units of length are not important. Define a unit as the length of an interconnecting wire that connects two components that are directly adjacent to one another (vertically or horizontally).

The problem statement introduced by Brixius and Anstreicher in [4] will be used here as opposed to the original formulation made by Steinberg in [40] due to the brevity of the former. As is the case with even the simplest assignment problems, it is convenient to add dummy components, in this case two, so that the total number of positions on the backboard equals the total number of components. Let be the number of wires connecting component i to component k, and be the distance from component j to component l on the backboard, then the general Steinberg Wiring Problem (SWP) can be formulated as follows:

$$
\begin{aligned}
\min \sum_{i,j,k,l} & c_{ik} d_{jl} x_{ij} x_{kl} \\
\text{s.t.} \sum_j x_{ij} &= 1, \qquad i = 1, \dots, n, \\
\sum_i x_{ij} &= 1, \qquad j = 1, \dots, n, \\
x_{ij} &\in \{0, 1\}, \qquad i, j = 1, \dots, n,
\end{aligned}
$$

where $x_{ij} = 1$ iff component i is placed at position j on the backboard

[4]. Notice that this problem statement matches the Quadratic 0-1 Integer Formulation as defined in §2.2.2. We see that in fact, the Steinberg Wiring Problem is an example of a QAP.

Recall that in Chapter 5 many heuristic procedures were introduced that are applied to QAPs with the hope of producing reasonable solutions. Until 2001, research had shown that the tabu search method (see §5.1.3) best produced the smallest known objective value of 9526 for the SWP as described above using the interconnection information provided by Steinberg in [40]. This solution was first discovered in 1990 and has been independently rediscovered many times since. One possible permutation which yields this objective value is:

(12,19,30,11,2,3,22,20,10,21,5,4,13,15,31,32,28,29,24,14,17,
18,16,9,8,7,6,23,33,34,25,35,27,26,1,36),

which corresponds to the placement of the components as shown in Figure 6.2 [7].[5] Notice that the two dummy components (35 and 36) are assigned to positions that are diagonally opposite from one another on the backboard as one might expect [40].

In 2001, Brixius and Anstreicher [4] implemented a Gilmore-Lawler bound based branch and bound algorithm (see §4.2) to solve the SWP. Their algorithm required approximately $7.75 * 10^8$ nodes in the search tree, and took approximately 186 hours of CPU time to complete on an 800 MHz Pentium III PC [3]. Their algorithm concluded that 9526 is in fact the optimal solution. The techniques used by Brixius and Anstreicher [4] produced the optimal solutions to the SWP within reasonably good time conditions; however, they remain confident that "there is still room for improvement in the overall time required to solve the problem" [4].

6.2 Hospital Layout

Designing a hospital is a formidable task to undertake. In such an environment where many lives are at stake, it is important that the design team take the necessary precautions to ensure that the facility layout is the most beneficial to both the patients and the care providers.

In 1975, Alwalid N. Elshafei [13] of the Institute of National Planning in Cairo, Egypt, investigated the optimal assignment of specific departments, or clinics (emergency room, X-ray, etc.) within a hospital. An optimal assignment as defined by Elshafei is one which minimizes the total distance traveled by patients between clinics, measured in patient-meters per year (mpy) [13]. For instance, it comes as no surprise that the Emergency Room

[5] For more information, see QAPLIB [7] problem Ste36a.

department (ER) at nearly every hospital in the world is located at the front of the facility, thus minimizing the total distance a patient in need of urgent care must travel before being treated. The idea of placing the ER anywhere else is intuitively self-defeating and illogical.

In [13], Elshafei focused specifically on the growing problem of overcrowding of the Out-patient department at a major hospital in Cairo. The department was comprised of 17 clinics treating an average of 700 patients per day. The poor placement of the clinics combined with the increasingly overwhelming volume of traffic between them was causing delays and heavy congestion [13]. To overcome this obstacle, Elshafei formulated the above as a decision problem and used operations research techniques to form a better layout of the department.

The task at hand was to assign n clinics to n locations within the department. Let c_{ik} be the known yearly flow between clinic i and clinic k, and d_{jl} be the known distance between location j and location l. Then the problem of assigning the clinics to the locations whereby the total travel distance is minimized may be formulated as the following 0-1 integer program:

$$\begin{aligned} \min \sum_{i,j,k,l} & c_{ik} d_{jl} x_{ij} x_{kl} \\ \text{s.t.} \quad \sum_j x_{ij} = 1, & \qquad i = 1, \dots, n, \\ \sum_i x_{ij} = 1, & \qquad j = 1, \dots, n, \\ x_{ij} \in \{0, 1\}, & \qquad i, j = 1, \dots, n, \end{aligned}$$

where $x_{ij} = 1$ if clinic i is to be placed at location j [13]. As in the previous example, the above formulation exactly matches the Quadratic 0-1 Integer Formulation from §2.2.2. Therefore, the problem of optimally assigning locations of clinics within a hospital department is another example of a quadratic assignment problem.

In [13], a two-part heuristic designed by Elshafei and Bazaraa (E&B) is presented and used to solve the Hospital Layout problem. The first part determines an initial solution, while the second part deals with improving the initial solution[6]. As previously stated, the department in question consists of 17 clinics together with a receiving and recording room. Since the recording room does not involve patients, it is not considered. Therefore, the goal is to find a good assignment of the other 18 independent facilities. All but one of these facilities requires the same amount of floor-space, the exception requiring twice the area of the others. This larger clinic will therefore be assigned

[6]For detailed explanation of the heuristic, see pages 6-7 of [13].

two adjacent rooms, yielding a total of $n = 19$ facilities to be assigned to 19 rooms [13].

E&B implemented their heuristic with the known distance and flow matrices, given in [13]. After creating an initial pattern, the procedure began searching for improvements by swapping pairs of assignments. Once no improvement was possible by this pairwise swapping, a new pattern was selected and the pairwise swapping routine repeated. This new pattern was selected from the patterns created by the pairwise swapping of the previous step; however, having not been better answers, these patterns were stored in ascending order based on their respective cost, mpy. The algorithm stopped when no better solution was obtained after testing 50 patterns [13].

Initially, the original layout had an associated cost of 13,973,298 mpy. The heuristic was applied and a best solution of 11,281,887 mpy was found, a decreased cost of over 19%[7]. The total computation time was 136 CPU seconds using an IBM 360/40 [13]. When compared with other problems with known solutions, the heuristic proved to produce good solutions.

The tools of optimization together with the work of great scientists such as Elshafei and Bazaraa eliminated nearly 20% of unnecessary traffic by patients, resulting in an overall more effective treatment center, by implementing a procedure which took less than two minutes to execute. As a result, their findings were implemented in a new layout of the department [13].

6.3 Other Examples

The two examples explained above provide reasonably good insight as to the applicability and importance of the QAP. However, they only represent a small fraction of the total number of decision problems modeled as QAPs. Other applications include dartboard design [12], typewriter keyboard design [6], scheduling [8], and production lines [18]; the list continues. As more applications arise, it is certain that the job of the combinatorial optimizer will never be complete.

References

[1] Anstreicher, K.M., N.W. Brixius, J. Linderoth, and J.-P. Goux (2002). Solving Large Quadratic Assignment Problems on Computational Grids. *Mathematical Programing, Series B* 91, 563-588.

[7] An optimal solution was found in 1992 by T. Mautor [7]. For more information, see QAPLIB problem Els19.

[2] Armour, G.C. and E.S. Buffa (1963). Heuristic Algorithm and Simulation Approach to Relative Location of Facilities. *Management Science* 9, 294-309.

[3] Assad, A.A., and W. Xu (1985). On Lower Bounds for a Class of Quadratic {0,1} Programs. *Operations Research Letters* 4, 175-180.

[4] Brixius, N.W. and K.M. Anstreicher (2001). The Steingberg Wiring Problem. To appear in, *The Sharpest Cut*, M. Grtschel, ed., SIAM.

[5] Burkard, R.E. and F.A. Rendl (1983). Thermodynamically Motivated Simulation Procedure for Combinatorial Optimization Problems. *European Journal of Operational Research* 17, 169-174.

[6] Burkard, R.E. and J. Offerman (1977). Entwurf von Schreibmaschinentastaturen mittels quadratischer zuordnungsprobleme. *Z. Operations Research* 21, B121-B123.

[7] Burkard, R.E., S.E. Karisch, and F. Rendl (1997). QAPLIB - A Quadratic Assignment Problem Library. *Journal of Global Optimization* 10, 391-403. An online version is available via the World Wide Web at the following URL: http://www.seas.upenn.edu/qaplib/

[8] Carlson, R.C. and G.L. Nemhauser (1966). Scheduling to minimize cost. *Operations Research* 14, 52-58.

[9] Carraresi, P. and F. Malucelli (1992). A New Lower Bound for the Quadratic Assignment Problem. Tech. Report TR-34/92, Universita di Pisa.

[10] Çela, E. (1998). *The Quadratic Assignment Problem: Theory and Algorithms*. Kluwer.

[11] Cerný, V. (1985). Thermodynamical Approach to the Traveling Salesman Problem: An Efficient Simulation Algorithm. *Journal of Optimization Theory and Applications* 45, 41-51.

[12] Eiselt, H.A. and G. Laporte (1991). A Combinatorial Optimization Problem Arising in Dartboard Design. *The Journal of the Operational Research Society* 42, 113-118.

[13] Elshafei, A.N. (1977). Hospital Layout as a Quadratic Assignment Problem. *Operations Research Quarterly* 28, 167-179.

[14] Engineering Research Associates and Remington Rand UNIVAC Records (CBI 124), Charles Babbage Institute, University of Minnesota, Minneapolis.

[15] Feo, T.A. and M.G.C. Resende (1994). Greedy Randomized Adaptive Search Procedures. Tech. report, AT&T Bell Laboratories, Murray Hill, NJ 07974-2070.

[16] Garey M.R. and D.S. Johnson (1990). *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W.H. Freeman Company.

[17] Finke, G., R.E. Burkard, and F. Rendl (1987) Quadratic Assignment Problems. *Annals of Discrete Mathematics* 31, 61-82.

[18] Geoffrion, A.M., and G.W. Graves (1976). Scheduling Parallel Production Lines with Changeover Costs: Practical Applications of a Quadratic Assignment/LP Approach. *Operations Research* 24, 595-610.

[19] Gilmore, P.C. (1962). Optimal and Suboptimal Algorithms for the Quadratic Assignment Problem. *SIAM Journal on Applied Mathematics* 10, 305-313.

[20] Glover F. (1989). Tabu Search - Part 2. *ORSA Journal on Computing* 2, no. 1, 4-32.

[21] Glover, F. (1989). Tabu Search - Part 1. *ORSA Journal on Computing* 1, no. 3, 190-206.

[22] Hadley, S.W., F. Rendl, and H. Wolkowicz (1990). Bounds for the Quadratic Assignment Problem Using Continuous Optimization Techniques. *Proceedings of* 1^{st} *International Integer Programming and Combinatorial Optimization Conference (IPCO)*, 237-248.

[23] Hadley, S.W., F. Rendl, and H. Wolkowicz (1993). A New Lower Bound via Projection for the Quadratic Assignment Problem. *Mathematics of Operations Research* 17, 727-739.

[24] Hahn, P. and T. Grant (1998). Lower Bounds for the Quadratic Assignment Problem Based upon a Dual Formulation. *Operations Research* 46, 912-922.

[25] Hanan, M. and J.M. Kurtzberg (1972). A Review of the Placement and Quadratic Assignment Problems, *SIAM Review* 14, 324-342.

[26] Koopmans, T.C., and M. Beckman (1957). Assignment problems and the location of economic activities. *Econometric* 25, 53-76.

[27] Lawler, E.L. (1963). The Quadratic Assignment Problem. *Management Science* 9, 586-599.

[28] Li, Y., P.M. Pardalos, and M. Resende (1994). A Greedy Randomized Adaptive Search Procedure for the Quadratic Assignment Problem, in *Quadratic Assignment and Related Problems*, P.M. Pardalos and H. Wolkowicz, eds., *DIMACS Series in Discrete Mathematics and Theoretical Computer Science* 16, 237-261, AMS.

[29] Li, Y., P.M. Pardalos, K.G. Ramakrishnan, and M. Resende (1994). Lower Bounds for the Quadratic Assignment Problem. *Annals of Operations Research* 50, 387-410.

[30] Li.Y, and P.M. Pardalos (1992). Generating Quadratic Assignment Test Problems with Known Optimal Permutations. *Computational Optimization and Applications* 1, 163-184.

[31] Palubetskis, G.S. (1988). Generation of Quadratic Assignment Test Problems with Known Optimal Solutions (in Russian). *U.S.S.R. Comput. Maths. Math. Phys.* 28, 97-98.

[32] Pardalos, P.M., F. Rendl, and H. Wolkowicz (1994). The Quadratic Assignment Problem: A Survey and Recent Developments, in *Quadratic Assignment and Related Problems*, P.M. Pardalos and H. Wolkowicz, eds., *DIMACS Series in Discrete Mathematics and Theoretical Computer Science* 16, 1-42, AMS.

[33] Pitsoulis, L. (1994). A Sparse GRASP for Solving the Quadratic Assignment Problem. M.Eng. thesis, The University of Florida.

[34] Pollatschek, M.A., N. Gershoni, and Y.T. Radday (1976). Optimization of the Typewriter Keyboard by Computer Simulation. *Angewandte Informatik* 10, 438-439.

[35] Ramakrishnan, K.G., M.G.C. Resende, B. Ramachandran, and J.F. Pekny (2002). Tight QAP bounds via linear programming, *Combinatorial and Global Optimization*, P.M. Pardalos, A. Migdalas, and R.E. Burkard, eds., World Scientific Publishing Co., Singapore, pp. 297-303.

[36] Rendl, F. and H. Wolkowicz (1992). Applications of Parametric Programming and Eigenvalue Maximization to the Quadratic Assignment Problem. *Mathematical Programming* 53, 63-78.

[37] Rendl, F. and H. Wolkowicz (1992). Applications of Parametric Programming and Eigenvalue Maximization to the Quadratic Assignment Problem. *Mathematical Programming* 53, 63-78.

[38] Resende, M.G.C., K.G. Ramakrishnan, and Z. Drezner (1995). Computing Lower Bounds for the Quadratic Assignment Problem with an Interior Point Algorithm for Linear Programming. *Operations Research* 43, 781-791.

[39] Sahni, S. and T. Gonzalez (1976). P-complete Approximation Problems. *Journal of the ACM* 23, 1976, 555-565.

[40] Steinberg, L. (1961). The Backboard Wiring Problem: A Placement Algorithm. *SIAM Review* 3, 37-50.