RECENT ADVANCES AND CHALLENGES

IN QUADRATIC ASSIGNMENT AND RELATED PROBLEMS

Yi-Rong Zhu

A DISSERTATION

in

Electrical and Systems Engineering

Presented to the Faculties of the University of Pennsylvania

in Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy

2007

____

Supervisor of Dissertation

____

Graduate Group Chairperson

DEDICATION

To My Parents

ACKNOWLEDGEMENTS

I am deeply indebted to many individuals who have helped me greatly during preparation of this dissertation. I particularly thank and recognize the guidance and patience of my advisor, Professor Peter M. Hahn. Without his instruction, this work could not have been possible. I appreciate the very much opportunity of being his student. And I am confident that my future professional career will deeply benefit from his knowledge and advice.

I feel very fortunate to have worked with Professor Monique Guignard-Spielberg, to whom I express my thanks and admiration. Her insight and advice for combinatorial optimization are valuable to me.

I am grateful to Professor Ali Jadbabaie and Professor Zhi-Long Chen, for serving as my committee and providing their helpful comments and suggestions.

I am pleased to thank the office staffs of the Electrical and Systems Engineering Department, in particular, Shelley Brown, for their numerous caring supports.

I also would like to recognize the help from my fellow students. I wish to single out Bum-Jin Kim, David J. Chang, Ya-Chi Tsao and Huafang Cui.

I would like to dedicate this dissertation to my parents Jin-Qian Zhu and Feng-Xia Yu for their support, patience and encouragement all the time.

ABSTRACT

RECENT ADVANCES AND CHALLENGES

IN QUADRATIC ASSIGNMENT AND RELATED PROBLEMS

Yi-Rong Zhu

Supervisor: Professor Peter M. Hahn

The quadratic assignment problem (QAP) is known as one of the most interesting and challenging problems in combinatorial optimization. This dissertation contributes to the theoretical, algorithmic and applicable understanding of quadratic assignment and its related problems.

The study includes four areas: (i) investigating the inherent relationship of the 3-dimensional assignment problem (3AP) to the quadratic assignment problem (QAP) and the quadratic 3-dimensional assignment problem (Q3AP); (ii) understanding the level-1 reformulation-linearization technique (RLT) formulation of the generalized quadratic assignment problem (GQAP) and comparing lower bounds from different RLT based models; (iii) modeling new applications of the multi-story space assignment problem (MSAP) and the crossdock door assignment problem (CDAP), and developing solution methods for an innovative assignment problem, the generalized quadratic 3-dimensional assignment problem (GQ3AP); and (iv) introducing the level-3 RLT formulation of the QAP for the first time and illustrating its great promise to provide superior lower bounds.

Solution methodologies studied in this work include the reformulation-linearization technique (RLT), Lagrangian dual procedure, and branch-and-bound enumeration. The methods devised herein effectively exploit the mathematical structure found within the RLT formulations. This consists of both theoretical and computational studies, including specially designed Lagrangian dual procedures that take advantage of the block-diagonal structures, and tradeoffs between linearization size and strength. Computational experiments were conducted by either implementing or extending the usefulness of existing algorithmic tools to solve application problems.

Table of Contents

List of Tables

List of Figures

1. Introduction

1.1 Introduction

The quadratic assignment problem (QAP) is known as one of the most interesting and challenging problems in combinatorial optimization. The QAP models find applications in facility location, computer manufacturing, scheduling, building layout design, and process communications. As described in the facility location context, QAP deals with assigning N facilities to N locations with the cost of each possible assignment being the flow between each pair of facilities multiplied with the distance between their assigned locations. The problem is to assign all facilities to different locations with the goal of minimizing the total cost.

In this dissertation, I present the recent advances and challenges in quadratic assignment and related problems and specifically identify my contributions to this research. My contributions comprise four distinct studies:

1. investigating the inherent relationship of the 3-dimensional assignment problem (3AP) to the quadratic assignment problem (QAP) and the quadratic 3-dimensional assignment problem (Q3AP);
2. understanding the level-1 reformulation-linearization technique (RLT) formulation of the generalized quadratic assignment problem (GQAP) and comparing lower bounds from different RLT based models;
3. modeling the multi-story space assignment problem (MSAP) and the crossdock door assignment problem (CDAP), and developing solution

methods for the generalized quadratic 3-dimensional assignment problem (GQ3AP);

4. introducing the level-3 RLT formulation of the QAP and illustrating its great promise to provide superior lower bounds.

1.2 Organization of the dissertation

Chapter 2 gives the background and literature search. It covers the previous research on quadratic assignment and related problems, together with the reformulation-linearization technique (RLT) that has been employed throughout this study.

Chapter 3 investigates the relationship of the 3-dimensional assignment problem (3AP) to the quadratic assignment problem (QAP) and the quadratic 3-dimensional assignment problem (Q3AP). My contribution to the research is conceiving, implementing and testing the method of solving a 3AP of size N as a QAP of size $2N$. Experimental results on solving the 3AP as a QAP or Q3AP are also presented.

Chapter 4 provides the level-1 RLT formulation of the generalized quadratic assignment problem (GQAP) and its Lagrangian relaxation. My contribution is the bounding strength comparison on different RLT based models.

Chapter 5 introduces the formulations of the multi-story space assignment problem (MSAP) and the crossdock door assignment problem (CDAP) as an innovative model of the generalized quadratic 3-dimensional assignment problem (GQ3AP). My main contribution is the analysis of the RLT dual-ascent method, upon which a lower bound algorithm for the GQ3AP may be designed. A detailed branch-and-bound procedure based on a level-1 RLT dual-ascent procedure has been developed. Numerical results for a set of MSAP problem instances are presented to demonstrate the effective implementation of the algorithm.

Chapter 6 reports the level-3 RLT formulation of the QAP, and its preliminary lower bound calculations. My main contribution is the analysis of the RLT-3 dual-ascent method, upon which a lower bound algorithm for the QAP may be designed. I also show how, by definitions of the cubic assignment problem (CAP) and the biquadratic assignment problem (BQAP), the hierarchy of zero-one assignment problems is directly connected to the RLT hierarchy of the QAP.

Chapter 7 summarizes the accomplishments of this dissertation and makes suggestions for the direction that future research on the quadratic assignment problem and its extensions should take.

1.3 Contributions of the dissertation

This dissertation presents a study of quadratic assignment and its related problems, specifically, the understanding in theoretical, algorithmic and applicable aspects. My contributions to this study include:

<u>Addressing the theoretical relationship of the 3-dimensional assignment problem (3AP) to the quadratic assignment problem (QAP) and the quadratic 3-dimensional assignment problem (Q3AP).</u> The 3AP has $N! \times N!$ feasible solutions, with each solution is completely determined by two permutations on $\{1,...,N\}$. This fact leads to manipulating the cost matrix of a corresponding size $2N$ QAP such that lifting the

original 3-dimensional N^3 assignment polytope to the higher dimensional QAP space of $(2N)^2 \times (2N)^2$. Computational experiments confirm the tighter lower bounds when solving the 3AP as a QAP. The level-1 reformulation-linearization technique (RLT) formulation of the Q3AP when relaxing the symmetric constraints shows a special block-diagonal structure which can be reduced to a series of 3APs. And the Q3AP branch-and-bound algorithm is able to solving the 3AP by forcing all the quadratic coefficients in the objective function to be zero. Besides the different order functions over the same 3-dimensional assignment matrix, this dissertation demonstrates that the problems of 3AP and Q3AP are also linked together through their solution methods.

Comparing lower bounds from different reformulation-linearization technique (RLT) based formulations on the generalized quadratic assignment problem (GQAP). In the application of RLT techniques to the GQAP, this dissertation presents three different level-1 RLT based formulations to illustrate their bounding strength. Numerical results show that when symmetric constraints on the complementary pairs are enforced, the RLT form multiplying each of the defining constraints with each variable x_{ij} provides both advantage of strength and convenience for further programming implementation, such as branch-and-bound algorithms and Lagrangian relaxations.

Investigating an innovative assignment problem, the generalized quadratic 3-dimensional assignment problem (GQ3AP), and establishing theoretical basis for its level-1 RLT dual-ascent procedure. The GQ3AP model is newly defined to handle applications arise from multi-story evacuation design and crossdock layout design. The GQ3AP formulation, solution structure and its RLT form are detailed to accommodate the level-1 dual-ascent procedure. Steps for an algorithm to calculate GQ3AP lower bounds and a branch-and-bound algorithm for solving the GQ3AP exactly are outlined. Computational experiments show that the branch-and-bound algorithm embedded with the dual-ascent procedure is capable to solve several instances of multi-story space assignment problem (MSAP) for the first time.

Developing the level-3 RLT formulation of the QAP for the first time, and providing theoretical outlines for its algorithmic implementation. This dissertation describes the construction of the level-3 QAP model, followed by the resulting mathematical formulation and model analysis. Steps of the level-3 dual-ascent procedure are listed, which will produce a nondecreasing sequence of lower bounds. Experimental comparisons on bounding quality provided by different algorithms are given to reveal the superior level-3 RLT lower bounds for QAP instances smaller than $N = 20$.

Establishing a hierarchy of zero-one assignment problems which is parallel to the RLT hierarchy of the QAP. The cubic assignment problem (CAP) is defined for the first time in the dissertation, which resembles the level-2 RLT formulation of the QAP. Similar relationship can also be discovered between the biquadratic assignment problem (BQAP) and the level-3 RLT QAP formulation. Therefore, a hierarchy of zero-one assignment problems is established to parallel the RLT hierarchy of the QAP, providing new theoretical connections among quadratic assignment and its related problems.

Applying RLT techniques to various assignment problems to forecast its potential in solving large classes of similarly difficult combinatorial optimization problems. The RLT techniques have been successfully applied to the class of quadratic assignment and related problems throughout the dissertation, namely, QAP, GQAP and GQ3AP, showing advances in lower bound calculations and branch-and-bound algorithms. Especially, the level-3 RLT formulation of the QAP has proven its outstanding performance in the quality of lower bound calculations.

2. Background and Literature Search

2.1 The quadratic assignment problem (QAP)

2.1.1 Background and problem formulations

Koopmans and Beckmann (1957) first introduced the quadratic assignment problem (QAP) as a mathematical model in analysis of the location of economic activities. The QAP stated as a facility location problem is to assign N facilities to N locations such that the total interaction cost of all possible flow-distance products between the locations to which the facilities are assigned plus the allocation costs of facilities to locations are minimized. Given the flow matrix $\boldsymbol{F} = \left[f_{ik} \right] \in \mathbb{R}^{N \times N}$ where f_{ik} is the flow from facility i to facility k, the distance matrix $\boldsymbol{D} = \left[d_{jn} \right] \in \mathbb{R}^{N \times N}$ where d_{jn} is the distance from location j to location n, and the cost matrix $\boldsymbol{B} = \left[b_{ij} \right] \in \mathbb{R}^{N \times N}$ where b_{ij} is the allocation cost of placing facility i at location j, the QAP in Koopmans-Beckmann form can be modeled as

$$\min \quad \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{N}\sum_{n=1}^{N} f_{ik} d_{jn} x_{ij} x_{kn} + \sum_{i=1}^{N}\sum_{j=1}^{N} b_{ij} x_{ij} \tag{2-1a}$$

$$\text{s.t.} \quad \sum_{i=1}^{N} x_{ij} = 1 \qquad j = 1,\dots,N, \tag{2-1b}$$

$$\sum_{j=1}^{N} x_{ij} = 1 \qquad i = 1,\dots,N, \tag{2-1c}$$

$$x_{ij} \in \{0,1\} \qquad i, j = 1,\dots,N. \tag{2-1d}$$

Each assignment of facilities to locations is represented by an $N \times N$ solution matrix $\boldsymbol{X} = \left[x_{ij} \right]$ where $x_{ij} = 1$ if facility i is being placed at location j or $x_{ij} = 0$ otherwise. Notice that $\boldsymbol{X} = \left[x_{ij} \right]$ is a permutation matrix.

A more general form of the QAP was proposed in Lawler (1963) by considering N^4 cost coefficients $c_{ijkn}\ \forall \left(i, j, k, n = 1, \ldots, N \right)$. The Lawler form of QAP is as follows.

$$\min \ \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{k=1}^{N} \sum_{n=1}^{N} c_{ijkn} x_{ij} x_{kn} \tag{2-2}$$

s.t. (2-1b), (2-1c) and (2-1d).

It is seen that the QAP in (2-1) is a special case of the above quadratic form by setting $c_{ijkn} = f_{ik} d_{jn}$ and $c_{ijij} = f_{ii} d_{jj} + b_{ij}$.

Lawler (1963) also introduced the concept of an $N^2 \times N^2$ solution matrix $\boldsymbol{Y}$ which is a Kronecker product of the $N \times N$ permutation matrix $\boldsymbol{X}$ with itself. The Kronecker product is a matrix operation on two arbitrarily dimensional matrices resulting in a larger matrix with special block structure. Given the $M \times N$ matrix $\boldsymbol{A} = \left[a_{ij} \right]$ and the $P \times Q$ matrix $\boldsymbol{B} = \left[b_{kn} \right]$, the Kronecker product of the two matrices is

$$\boldsymbol{A} \otimes \boldsymbol{B} = \left[a_{ij} b_{kn} \right] = \left[a_{ij} \boldsymbol{B} \right] \qquad \forall \left(i, j, k, n \right), \tag{2-3}$$

$\boldsymbol{A} \otimes \boldsymbol{B}$ is seen to be an $MP \times NQ$ block matrix, with the $\left(i, j \right)$th block in all MN blocks is the $P \times Q$ matrix $a_{ij} \boldsymbol{B}$. See Graham (1981). Therefore,

$$
\boldsymbol{Y} = \boldsymbol{X} \otimes \boldsymbol{X} = \begin{bmatrix} x_{11}\boldsymbol{X} & \cdots & x_{1N}\boldsymbol{X} \\ \vdots & \ddots & \vdots \\ x_{N1}\boldsymbol{X} & \cdots & x_{NN}\boldsymbol{X} \end{bmatrix}
$$

$$
= \left[y_{ijkn} \right]_{N^2 \times N^2} = \left[\boldsymbol{Y}_{[ij]} \right]_{N \times N} = \begin{bmatrix} \boldsymbol{Y}_{[11]} & \cdots & \boldsymbol{Y}_{[1N]} \\ \vdots & \ddots & \vdots \\ \boldsymbol{Y}_{[N1]} & \cdots & \boldsymbol{Y}_{[NN]} \end{bmatrix} \tag{2-4a}
$$

where $y_{ijkn} = x_{ij}x_{kn} = x_{kn}x_{ij} = y_{knij} \qquad \forall\left(i, j, k, n = 1, \dots, N\right), k \neq i, n \neq j$, (2-4b)

$$
y_{ijij} = x_{ij} \qquad \forall\left(i, j = 1, \dots, N\right), \tag{2-4c}
$$

$$
y_{ijin} = 0 \qquad \forall\left(i, j, n = 1, \dots, N\right), n \neq j, \tag{2-4d}
$$

$$
y_{ijkj} = 0 \qquad \forall\left(i, j, k = 1, \dots, N\right), k \neq i. \tag{2-4e}
$$

Here, (2-4b)-(2-4e) are indicated by the solution structure $\boldsymbol{X}$ of QAP and the definition of $\boldsymbol{Y}$. Furthermore, the matrix $\boldsymbol{Y}$ is a partitioned matrix whose elements are composed of the Null matrix (with all the elements being zero) and matrix $\boldsymbol{X}$, and $\boldsymbol{Y}$ also exhibits a gross pattern identical to that of the matrix $\boldsymbol{X}$. An example of a $\boldsymbol{Y}$ matrix and its corresponding $\boldsymbol{X}$ matrix for the QAP of $N = 3$ are shown in Figure 2-1.

Figure 2-1. Example of QAP solution matrix $\boldsymbol{Y}$ and its $\boldsymbol{X}$ matrix.

$$
\boldsymbol{Y} = \left(\begin{array}{ccc:ccc:ccc}
\{0\} & 0 & 0 & 0 & \{0\} & 0 & 0 & 0 & \{1\} \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
\hdashline
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
\{1\} & 0 & 0 & 0 & \{0\} & 0 & 0 & 0 & \{0\} \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\hdashline
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
\{0\} & 0 & 0 & 0 & \{1\} & 0 & 0 & 0 & \{0\}
\end{array}\right) \quad
\boldsymbol{X} = \left(\begin{array}{c:c:c}
0 & 0 & 1 \\
\hdashline
1 & 0 & 0 \\
\hdashline
0 & 1 & 0
\end{array}\right)
$$

Equation (2-4b) shows an important property of the QAP solution matrix $\boldsymbol{Y}$. If an element y_{ijkn} $\left(k \neq i \text{ and } n \neq j\right)$ is part of a solution (i.e., equals to one) then it has a "complementary element" y_{knij} that is also in that solution. These complementary pairs are always in two different submatrices that never occupy the same submatrix row or submatrix column. However, the pair creates a connection between submatrices that provides valuable information in lower bound calculation. Elements defined by equations (2-4d) and (2-4e) are referred to as "disallowed elements", which are always being zero. Elements defined by equation (2-4c) are termed "leaders" and are shown bracketed in Figure 2-1. A leader has no complementary element. Notice that if there is any 1's in a submatrix $x_{ij}\boldsymbol{X}$, its leader is always unity and vice versa. Notice, too, that one and only one unity element may exist in each row and each column of solution matrix $\boldsymbol{Y}$.

Let Γ_N denotes the set of all permutation of $\{1,...,N\}$ and $\varphi \in \Gamma_N$. The QAP in Koopmans-Beckmann form can be written as

$$\min_{\varphi \in \Gamma_N} \sum_{i=1}^{N}\sum_{k=1}^{N} f_{ik} d_{\varphi(i)\varphi(k)} + \sum_{i=1}^{N} b_{i\varphi(i)}, \tag{2-5}$$

where each $f_{ik} d_{\varphi(i)\varphi(k)}$ is the cost caused by assigning facility i to location $\varphi(i)$ and facility k to location $\varphi(k)$, and each permutation φ represents an allocation of facilities to locations. The corresponding Lawler form in permutation is

$$\min_{\varphi \in \Gamma_N} \sum_{i=1}^{N}\sum_{k=1}^{N} c_{ik\varphi(i)\varphi(k)}. \tag{2-6}$$

The QAP in Koopmans-Beckmann form can be presented in a more compact trace formulation for the application of spectral theory and semidefinite programming. Let the inner product of two real $N \times N$ matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ be defined by $\langle \boldsymbol{A}, \boldsymbol{B} \rangle \equiv \sum_{i=1}^{N} \sum_{j=1}^{N} a_{ij} b_{ij}$. Given a permutation $\varphi \in \Gamma_N$ and the associated permutation matrix $\boldsymbol{X}_\varphi \in \Pi$ where Π is the set of all $N \times N$ permutation matrices, it turns out that $\boldsymbol{X}_\varphi \boldsymbol{D} \boldsymbol{X}_\varphi^T = \left[d_{\varphi(i)\varphi(k)} \right]$, therefore the Koopmans-Beckmann QAP can be stated below.

$$\min_{\boldsymbol{X} \in \Pi} \ \left\langle \boldsymbol{F}, \boldsymbol{X} \boldsymbol{D} \boldsymbol{X}^T \right\rangle + \left\langle \boldsymbol{B}, \boldsymbol{X} \right\rangle \tag{2-7}$$

With the definition of trace function on an $N \times N$ matrix being defined as the sum of the elements on its main diagonal, it comes $\sum_{i=1}^{N} \sum_{k=1}^{N} f_{ik} d_{\varphi(i)\varphi(k)} = tr\left(\boldsymbol{F} \boldsymbol{X} \boldsymbol{D}^T \boldsymbol{X}^T \right)$, $\sum_{i=1}^{N} b_{i\varphi(i)} = tr\left(\boldsymbol{B} \boldsymbol{X}^T \right)$. Thus, the trace formulation, which first introduced by Edwards (1980), can be written as follows.

$$\min_{\boldsymbol{X} \in \Pi} \ \mathrm{tr}\left(\boldsymbol{F} \boldsymbol{X} \boldsymbol{D}^T + \boldsymbol{B} \right) \boldsymbol{X}^T \tag{2-8}$$

Since the first formulation appeared in facility layout problem, the QAP has practical applications in several areas: scheduling and routing, process control, data transmission and classification, and ergonometric design. The recent surveys on applications and solution methods are Pardalos et al. (1994), Burkard et al. (1998), Burkard (2002), Anstreicher (2003) and Loiola et al. (2006), as well as, the books by Pardalos and Wolkowicz (1994) and Çela (1998).

2.1.2 Lower bounds

Despite its wide applications in various areas, the QAP is known to be notoriously difficult to solve. The QAP belongs to the class of the hardest combinatorial optimization problems. In general, instances of size $N > 30$ cannot be solved in reasonable CPU time. Sahni and Gonzalez (1976) proved that the QAP is NP-hard and that unless P=NP, it is not even possible to find an ε-approximation algorithm, for a constant ε.

Due to the inherent computational complexity of the QAP, the study of lower bounds is critically important in exact algorithms, which employ implicit enumeration, like branch-and-bound techniques. Tight and efficient lower bounds are essential for the performance of these algorithms.

The Gilmore-Lawler bound (GLB) presented by Gilmore (1962) and Lawler (1963) is one of the best known lower bounds for QAP. Though GLB is easy to compute, its gap from the optimal solution deteriorates very fast as the size of the problems grow. Recently, researchers have tried to obtain tighter lower bounds based on semidefinite programming (SDP), reformulation-linearization technique (RLT) and other lift-and-project procedures. A number of benchmark problem instances have been solved to optimality for the first time, see QAPLIB homepage (http://www.seas.upenn.edu/qaplib/). These developments on the lower bounds include: the interior point bound by Resende et al. (1995), the level-1 RLT (or RLT1) dual-ascent bound by Hahn and Grant (1998), the dual-based bound by Karisch et al. (1999), the convex quadratic programming bound by Anstreicher and Brixius (2001), the level-2 RLT (or RLT2) interior point bound by Ramakrishnan et al. (2002), the bundle method bound by Rendl and Sotirov (2007), the lift-and-project SDP bound by Burer and Vandenbussche (2006), and the Hahn-Hightower level-2 RLT dual-ascent bound by Adams et al. (2007). Among all these new lower bounds, the tightest are the lift-and-project SDP bound and the two level-2 RLT bounds. However, taking speed or efficiency into consideration, the most competitive and promising bounds in branch-and-bound algorithms are the level-1 RLT dual-ascent bound by Hahn and Grant (1998), the convex quadratic programming bound by Anstreicher and Brixius (2001), and the Hahn-Hightower level-2 RLT dual-ascent bound by Adams et al. (2007). The reformulation-linearization technique (RLT) is explained in more detail in Section 2.5 of this dissertation.

For $i, j = 1, \ldots, N$, let l_{ij} denote the solution value of the linear assignment problem (LAP)

$$\min \quad \sum_{k=1}^{N} \sum_{n=1}^{N} c_{ijkn} x_{kn} \tag{2-9a}$$

$$\text{s.t.} \qquad \boldsymbol{X} \in \Pi, \; x_{ij} = 1 . \tag{2-9b}$$

It is clear that for all $i, j = 1, \ldots, N$, $l_{ij} x_{ij} \le \left(\sum_{k=1}^{N} \sum_{n=1}^{N} c_{ijkn} x_{kn} \right) x_{ij}$, then the QAP is bounded by $\text{GLB} \equiv \text{LAP}(\boldsymbol{L})$, where $\text{LAP}(\boldsymbol{L})$ denotes the solution value of the LAP with cost matrix of $\boldsymbol{L} = \left[l_{ij} \right]$. For a general Lawler QAP in form of (2-2), the computation of GLB requires the solution of $N^2 + 1$ LAPs.

Based on the RLT technique for general zero-one polynomial programs by Adams and Sherali (1986, 1990), Adams and Johnson (1994) gave an equivalent level-1 RLT formulation for the QAP,

$$\min \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{N}\sum_{n=1}^{N} c_{ijkn} y_{ijkn} \tag{2-10a}$$

$$\text{s.t.} \quad \sum_{\substack{i=1 \\ i \neq k}}^{N} y_{ijkn} = x_{kn} \qquad j,k,n = 1,\ldots,N;\ j \neq n, \tag{2-10b}$$

$$\sum_{\substack{j=1 \\ j \neq n}}^{N} y_{ijkn} = x_{kn} \qquad i,k,n = 1,\ldots,N; i \neq k, \tag{2-10c}$$

$$y_{ijkn} = y_{knij} \qquad i,j,k,n = 1,\ldots,N; i < k,\ j \neq n, \tag{2-10d}$$

$$y_{ijkn} \geq 0 \qquad i,j,k,n = 1,\ldots,N; i \neq k,\ j \neq n, \tag{2-10e}$$

$$\sum_{i=1}^{N} x_{ij} = 1 \qquad j = 1,\ldots,N, \tag{2-10f}$$

$$\sum_{j=1}^{N} x_{ij} = 1 \qquad i = 1,\ldots,N, \tag{2-10g}$$

$$x_{ij} \in \{0,1\} \qquad i,j = 1,\ldots,N, \tag{2-10h}$$

where each y_{ijkn} represents the product $x_{ij}x_{kn}$. They have also proved that without the symmetry constraints (2-10d) on the complementary pairs, the solution value of its linear programming (LP) relaxation is exactly GLB. In general, the above mixed-integer linear programming (MILP) model (2-10) has a significant larger number of variables and constraints, which makes it computationally challenging.

Resende et al. (1995) investigated the LP relaxation of the level-1 RLT formulation using an interior point algorithm for LP (Karmarkar and Ramakrishnan, 1991). This approach produced better bounds than those obtained by Adams and Johnson (1994), but required prohibitively computational cost for branch-and-bound implementation. Karisch et al. (1999) showed that many bounding schemes for the QAP can be acquired from the dual of the level-1 LP model. Computationally, the most successful of these methods is the bounding strategy proposed by Hahn and Grant (1998) with reduction steps in a dual framework. Strategy using the level-2 RLT form gives even tighter bounds, but at the price of extremely increased problem size. By solving the level-2 LP program, Ramakrishnan et al. (2002) was able to get the optimal solutions for QAPs of size up to 12 without branching. Adams et al. (2007) extended their work to the level-2 RLT formulation. Their dual-ascent approach is much more efficient than directly solving the level-2 LP problem, though storage requirements are still a non-trivial issue for larger instances.

Consider the trace formulation (2-8) of a Koopmans-Beckmann QAP with symmetric matrices $\boldsymbol{F}$ and $\boldsymbol{D}$. One can derive the eigenvalue bounds by relaxing the feasible set Π to the set of orthogonal matrices Θ, where $\Theta = \left\{ \boldsymbol{X} \,\middle|\, \boldsymbol{X}\boldsymbol{X}^T = \boldsymbol{X}^T\boldsymbol{X} = \boldsymbol{I} \right\}$ and $\boldsymbol{I}$ is the $N \times N$ identity matrix. By Finke et al. (1987), Rendl and Wolkowicz (1992), the lower bound on the quadratic term is

$$\min_{\boldsymbol{X} \in \Theta} \ \operatorname{tr}\left(\boldsymbol{F}\boldsymbol{X}\boldsymbol{D}\boldsymbol{X}^T\right) = \left\langle \lambda\left(\boldsymbol{F}\right), \lambda\left(\boldsymbol{D}\right) \right\rangle_- \ , \tag{2-11}$$

as $\lambda\left(\boldsymbol{A}\right) \in \mathbb{R}^N$ denotes the vector of eigenvalues of a symmetric matrix $\boldsymbol{A}$, and for vectors u and v, $\left\langle u, v \right\rangle_-$ denotes the minimum product $\left\langle u, v \right\rangle_- \equiv \min_{\varphi} \sum_{i=1}^{N} u_i v_{\varphi(i)}$ where φ is a permutation of $\left\{1, \ldots, N\right\}$. Therefore,

$$\left\langle \lambda\left(\boldsymbol{F}\right), \lambda\left(\boldsymbol{D}\right) \right\rangle_- + \mathrm{LAP}(\boldsymbol{B}) \tag{2-12}$$

is a valid lower bound for the QAP in (2-8). Both papers also consider strengthening methods, since the simple eigenvalue bound in (2-12) is too weak to be of use computationally. The approach by Hadley et al. (1992) for a tighter relaxation is based on the representation for $\Theta \cap \mathrm{E}$, where E denotes the set of matrices with row and column sums equal to one $\left(\boldsymbol{Xe} = \boldsymbol{X}^T\boldsymbol{e} = \boldsymbol{e}\right)$ and $\boldsymbol{e} = (1,...,1)^T$. The incorporation of $\Theta \cap \mathrm{E}$ in the objective function follows from the fact that for any $\boldsymbol{X} \in \Pi$, the vector of ones is both a left and right eigenvector with eigenvalue 1. This leads directly to the SDP formulation of QAP as

$$\min \quad \mathrm{tr}\left(\boldsymbol{FXDX}^T + \boldsymbol{BX}^T\right) \tag{2-13a}$$

$$\text{s.t.} \qquad \boldsymbol{XX}^T = \boldsymbol{X}^T\boldsymbol{X} = \boldsymbol{I} \tag{2-13b}$$

$$\boldsymbol{Xe} = \boldsymbol{X}^T\boldsymbol{e} = \boldsymbol{e} \tag{2-13c}$$

$$x_{ij}^2 - x_{ij} = 0 \qquad\qquad i, j = 1,...,N. \tag{2-13d}$$

From this formulation, the SDP relaxation of QAP can be obtained. Consider the $N^2 \times N^2$ matrix $\boldsymbol{Y}$ by definition (2-4a), define the linear operators from $\mathbb{R}^{N^2 \times N^2}$ to $\mathbb{R}^{N \times N}$,

$$\mathrm{bdiag}\left(\boldsymbol{Y}\right) \equiv \sum_{i=1}^{N} \boldsymbol{Y}_{[ii]} \tag{2-14a}$$

$$\left(\mathrm{odiag}\left(\boldsymbol{Y}\right)\right)_{ij} \equiv \mathrm{tr}\boldsymbol{Y}_{[ij]} \qquad\qquad i, j = 1,...,N, \tag{2-14b}$$

and let $\hat{\boldsymbol{E}}$ be the $2N \times N^2$ matrix,

$$\hat{\boldsymbol{E}} = \begin{pmatrix} \boldsymbol{e}^T \otimes \boldsymbol{I} \\ \boldsymbol{I} \otimes \boldsymbol{e}^T \end{pmatrix}. \tag{2-14c}$$

The basic SDP bound for a homogeneous $\left(\boldsymbol{B} = 0\right)$ Koopmans-Beckmann QAP by Zhao et al. (1998) is

$$\min \quad \left(\boldsymbol{D} \otimes \boldsymbol{F}\right) \bullet \boldsymbol{Y} \tag{2-15a}$$

$$\text{s.t.} \quad \operatorname{bdiag}(\boldsymbol{Y}) = \boldsymbol{I} \tag{2-15b}$$

$$\operatorname{odiag}(\boldsymbol{Y}) = \boldsymbol{I} \tag{2-15c}$$

$$\hat{\boldsymbol{E}} \bullet \boldsymbol{Y} = 2N \tag{2-15d}$$

$$\boldsymbol{Y} - \operatorname{diag}(\boldsymbol{Y}) \operatorname{diag}(\boldsymbol{Y})^T \succeq 0, \tag{2-15e}$$

here $\operatorname{diag}(\boldsymbol{A})$ is the vector of diagonal elements of matrix $\boldsymbol{A}$. The quality of the SDP bounds is competitive with the best existing lower bounds for QAP, yet with prohibitively high computational time.

The SDP bounds can be reinforced by imposing nonnegativity constraints on all elements of $\boldsymbol{Y}$, as well as constraints on certain elements of $\boldsymbol{Y}$ to be zero, as indicated by the definition (2-4). Anstreicher and Brixius (2001) proposed a SDP interpretation of a basic eigenvalue bound. Rendl and Sotirov (2007) applied bundle methods to handle the large number of constraints that appear in the SDP relaxations for QAP. Burer and Vandenbussche (2006) have used an augmented Lagrangian technique to obtain very tight SDP bounds on a lift-and-project QAP relaxations introduced by Lovász and Schrijver (1991).

2.2 The generalized quadratic assignment problem (GQAP)

2.2.1 Background and problem formulations

The generalized quadratic assignment problem (GQAP) studies a class of problems that optimally assign M entities to N destinations subject to the resource limitation at each destination. These problems arise naturally in yard management, where containers are to be located in the storage areas with limited capacity, and in distributed computing where processing tasks are to be assigned to processors with limited computing resources. The GQAP is a generalization of the QAP that multiple entities can be assigned to a single destination if only such assignment does not violate the resource capacity at destinations.

Lee and Ma (2004) proposed the first formulation of GQAP. Their study involves a facility location problem in manufacturing where M facilities to be located among N fixed locations given the space constraint at each possible location, with the objective to minimize the total installation and interaction transportation cost. The formulation of the GQAP is then

$$\min \quad \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{k=1}^{M}\sum_{n=1}^{N} f_{ik} d_{jn} x_{ij} x_{kn} + \sum_{i=1}^{M}\sum_{j=1}^{N} b_{ij} x_{ij} \tag{2-16a}$$

$$\text{s.t.} \quad \sum_{i=1}^{M} s_{ij} x_{ij} \le S_j \qquad j = 1,\ldots,N, \tag{2-16b}$$

$$\sum_{j=1}^{N} x_{ij} = 1 \qquad i = 1,\ldots,M, \tag{2-16c}$$

$$x_{ij} \in \{0,1\} \qquad i = 1,\ldots,M;\ j = 1,\ldots,N, \tag{2-16d}$$

where

M the number of facilities,

N the number of locations,

f_{ik} the commodity flow from facility i to facility k,

d_{jn} the distance from location j to location n,

b_{ij} the cost of installing facility i at location j,

s_{ij} the space requirement if facility i is installed at location j,

S_j the space available at location j,

x_{ij} binary variable, $x_{ij} = 1$ iff facility i is installed at location j.

The objective function (2-16a) sums the costs of installation and quadratic interactivity. The knapsack constraints (2-16b) impose space limitations at each location, and the multiple choice constraints (2-16c) ensure that each facility is to be installed at exactly one location.

In more general case of the GQAP, the quadratic cost between entities is known but is not necessarily decomposable to be the product of flows and distances. Similarly as the Lawler QAP model, the linear cost b_{ij} can also be combined into the $M^2 \cdot N^2$ cost coefficients $c_{ijkn}\ \forall\left(i,k=1,\ldots,M;\, j,n=1,\ldots,N\right)$. The general GQAP formulation is

$$\min \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{k=1}^{M}\sum_{n=1}^{N} c_{ijkn} x_{ij} x_{kn} \tag{2-17}$$

s.t. (2-16b), (2-16c) and (2-16d).

Here, the solution matrix $\boldsymbol{X} = \left[x_{ij}\right]$ is an $M \times N$ matrix which determines the "assignment" of each entities to the corresponding location.

Now consider an $M^2 \times N^2$ solution matrix $\boldsymbol{Y}$ which is a Kronecker product of the $M \times N$ assignment matrix $\boldsymbol{X}$ with itself. Then,

$$\boldsymbol{Y} = \boldsymbol{X} \otimes \boldsymbol{X} = \begin{bmatrix} x_{11}\boldsymbol{X} & \cdots & x_{1N}\boldsymbol{X} \\ \vdots & \ddots & \vdots \\ x_{M1}\boldsymbol{X} & \cdots & x_{MN}\boldsymbol{X} \end{bmatrix} = \left[y_{ijkn}\right]_{M^2 \times N^2} \tag{2-18a}$$

where $y_{ijkn} = x_{ij}x_{kn} = x_{kn}x_{ij} = y_{knij} \qquad \forall\left(i,k=1,\ldots,M;\, j,n=1,\ldots,N\right), k \neq i$, (2-18b)

$$y_{ijij} = x_{ij} \qquad \forall\left(i=1,\ldots,M;\, j=1,\ldots,N\right), \tag{2-18c}$$

$$y_{ijin} = 0 \qquad \forall\left(i=1,\ldots,M;\, j,n=1,\ldots,N\right), n \neq j\,. \tag{2-18d}$$

Here, (2-18b)-(2-18d) are indicated by the solution structure $\boldsymbol{X}$ of the GQAP and the definition of $\boldsymbol{Y}$. Also, the matrix $\boldsymbol{Y}$ is a partitioned matrix whose elements are composed of the Null matrix and matrix $\boldsymbol{X}$, and $\boldsymbol{Y}$ exhibits a gross pattern identical to that of the matrix $\boldsymbol{X}$. An example of a $\boldsymbol{Y}$ matrix and its corresponding $\boldsymbol{X}$ matrix for the GQAP of $M=4$ and $N=3$ are shown in Figure 2-2.

Figure 2-2. Example of GQAP solution matrix $\boldsymbol{Y}$ and its $\boldsymbol{X}$ matrix.

$$\boldsymbol{Y} = \left(\begin{array}{ccc|ccc|ccc}
\{1\} & 0 & 0 & 0 & \{0\} & 0 & 0 & 0 & \{0\} \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
\hline
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\{1\} & 0 & 0 & 0 & \{0\} & 0 & 0 & 0 & \{0\} \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
\hline
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
\{0\} & 0 & 0 & 0 & \{1\} & 0 & 0 & 0 & \{0\} \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
\hline
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
\{0\} & 0 & 0 & 0 & \{0\} & 0 & 0 & 0 & \{1\}
\end{array}\right) \qquad \boldsymbol{X} = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Equation (2-18b) gives the complementary pairs in the GQAP solution matrix $\boldsymbol{Y}$. If an element $y_{ijkn}\ (k \neq i)$ is part of a solution (i.e., equals to one) then its complementary element y_{knij} is also in that solution. These complementary pairs are always in two different submatrices that never occupy the same submatrix row. However, the pair creates communication between submatrices that can be exploited in improving bounds. Elements defined by equation (2-18d) are referred to as "disallowed elements", which are always being zero. Elements defined by equation (2-18c) are termed "linear-cost elements" and are shown bracketed in Figure 2-2. A linear-cost element has no complementary element. Notice that if there is any 1's in a submatrix $x_{ij}\boldsymbol{X}$, its linear-cost element is always unity and vice versa. Notice, too, that one and only one unity element may exist in each row of solution matrix $\boldsymbol{Y}$.

2.2.2 Complexity and related problems

The GQAP is a generalization of the QAP, and a generalization of the generalized assignment problem (GAP) as well. Much of the important literature on the QAP is mentioned in the previous chapter of this paper.

The GAP deals with the minimum cost assignment of a set of tasks to a set of agents, wherein each task is assigned to one and only one agent and the resource constraint on each agent is satisfied. The GAP has many applications in flexible manufacturing, resource scheduling, facility locating, vehicle routing and other fields. Consider M tasks to be assigned to N agents, and let b_{ij} be the cost of task i to be assigned to agent j, s_{ij} be the resource requirement of task i to be assigned to agent j and S_j be the available resource of agent j. The GAP is formulated as follows.

$$\min \sum_{i=1}^{M}\sum_{j=1}^{N} b_{ij} x_{ij} \tag{2-19}$$

s.t. (2-16b), (2-16c) and (2-16d),

where in the $M \times N$ solution matrix $\boldsymbol{X} = \left[x_{ij} \right]$, $x_{ij} = 1$ if task i is assigned to agent j and $x_{ij} = 0$ otherwise. Ross and Soland (1975) gave the first formal definition of the GAP. Fisher et al. (1986) showed that the GAP is an NP-hard combinatorial optimization problem. Since the first computational work by Ross and Soland (1975) which was a branch-and-bound algorithm based on the Lagrangian relaxation, numerous exact algorithms and heuristics have been devised. Cattrysse and Van Wassenhove (1992) reviewed both exact and heuristic methods for solving the GAP. Most approaches were based on branch-and-bound techniques with bounds provided by heuristics and by relaxations of knapsack constraints (2-16b) or multiple choice constraints (2-16c). Fisher et al. (1986) investigated the Lagrangian relaxation of dualizing constraints (2-16c) with multipliers set by a heuristic adjustment method. Guignard and Rosenwein (1989) tightened the above Lagrangian relaxation by devising a dual-ascent procedure which also added a surrogate constraint. Barcia and Jörnsten (1990) combined Lagrangian decomposition and a bound improvement sequence algorithm to solve the GAP. Savelsbergh (1997) employed both column generation and branch-and-bound to generate optimal solutions to a set partitioning formulation of the GAP. Recent development on the exact solution methods includes Nauss (2003), Haddadi and Ouzia (2004). Nauss (2003) presented a special-purpose branch-and-bound algorithm that utilizes linear programming cuts, feasible solution generators, Lagrangian relaxation, and subgradient optimization to solve hard GAP problems with up to 3000 binary variables. Haddadi and

Ouzia (2004) provided a breadth-first branch-and-bound algorithm for the GAP with largest-upper-bound-next branching strategy. They applied a new heuristic at each iteration of the subgradient method to construct a feasible assignment by solving a smaller GAP. The feasible solution found was then subjected to a solution improvement heuristic. See Nauss (2006) for a recent survey.

Though the GQAP is recently formulated, its special cases have been studied extensively. Those related problems, other than the QAP and the GAP, as listed include the multiprocessor assignment problem of Magirou et al. (1989), the task assignment and multiway cut problems by Magirou (1992), the process allocation problem by Sofianopoulou (1992a, 1992b), the constrained task assignment problem by Billionnet et al. (1995), the quadratic semi-assignment problem by Billionnet et al. (2001), the constrained module allocation problem by Elloumi et al. (2003), the constrained-memory allocation problem by Roupin (2004), and the service allocation problem by Cordeau (2007).

2.2.3 Previous computational research

The study on the GQAP is relatively limited. Lee and Ma (2004) presented three linearization approaches and a branch-and-bound algorithm to optimally solve the problem. In their branch-and-bound algorithm, the best feasible solution found by means of two greedy heuristics was taken as an initial upper bound for the branch-and-bound enumeration. Then, at each node of the search tree a lower bound was computed by a series of GAPs with the cost associated with the set of already assigned and yet unassigned entities. They gave computational results for 26 test instances, the most difficult of which was of 16 entities and 7 locations.

The other literature on the exact solution procedure for the GQAP is Hahn et al. (2006a). They presented the level-1 RLT formulation of the general version GQAP and a useful Lagrangian relaxation. Then a dual-ascent procedure was proposed to solve the Lagrangian relaxation so as to provide a lower bound in the branch-and-bound enumeration. Their exact solution method reported speed improvement on problem instances designed by Lee and Ma (2004), and was capable of optimally solving instances up to 20 entities and 15 locations.

Cordeau et al. (2006) worked on heuristic approach to obtain good suboptimal assignments using reasonable computing time. However, their memetic heuristic was not able to find exact solutions. Kim (2006) developed a simulated annealing heuristic method with competitive performance.

2.3 The multidimensional assignment problem (MAP)

2.3.1 Background

The multidimensional (sometimes referred to as multi-index) assignment problem (MAP) is a natural extension of the linear (2-dimensional) assignment problem. This class of combinatorial problems was studied for the first time by Pierskalla (1968). The general idea is that additional dimensions, like time, space and other factors, may be taken into consideration when entities are to be assigned to locations.

2.3.2 The 3-dimensional assignment problem (3AP)

The most famous examples of MAP are the 3-dimensional assignment problem (3AP), or the so-called axial 3-dimensional assignment problem (A3AP) and the so-called planar 3-dimensional assignment problem (P3AP).

Given N^3 cost coefficients b_{ijp}, the A3AP is presented as

$$\min \quad \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{p=1}^{N} b_{ijp}x_{ijp} \tag{2-20a}$$

$$\text{s.t.} \quad \sum_{i=1}^{N}\sum_{j=1}^{N} x_{ijp} = 1 \qquad p = 1,\dots,N\,, \tag{2-20b}$$

$$\sum_{i=1}^{N}\sum_{p=1}^{N} x_{ijp} = 1 \qquad j = 1,\dots,N\,, \tag{2-20c}$$

$$\sum_{j=1}^{N}\sum_{p=1}^{N} x_{ijp} = 1 \qquad i = 1,\dots,N\,, \tag{2-20d}$$

$$x_{ijp} \in \{0,1\} \qquad i, j, p = 1,\dots,N\,. \tag{2-20e}$$

If b_{ijp} is the cost of worker i performs job j on machine p, it follows that $x_{ijp} = 1$ if worker i performs job j on machine p, and $x_{ijp} = 0$, otherwise. A feasible solution to the A3AP is a 3-dimensional permutation array, that is, a 0-1 array that satisfies the above assignment constraints.

Given two permutations $\varphi, \phi \in \Gamma_N$, the A3AP can be stated as

$$\min_{\varphi,\phi\in\Gamma_N} \quad \sum_{i=1}^{N} c_{i\varphi(i)\phi(i)}\,. \tag{2-21}$$

The A3AP has $N! \times N!$ feasible solutions. Karp (1972) showed that the A3AP is NP-hard. Applications of the A3AP as mentioned in Pierskalla (1967a, 1968) arise in capital investment, dynamic facility location and satellite launching.

In terms of graph theory, the A3AP is described as follows. Given 3 disjoint N-sets A_1, A_2, and A_3. For each triple in $A_1 \times A_2 \times A_3$, a cost b_{ijp} is known. The A3AP is to find N triples such that each element from $A_1 \cup A_2 \cup A_3$ occurs in exactly one triple and the total cost is minimized.

As for the exact approaches to solve the A3AP to optimality, several branch-and-bound algorithms have been proposed. Balas and Saltzman (1991) introduced a Lagrangian relaxation incorporating a class of facet inequalities and solved it by a modified subgradient procedure to produce the lower bounds. They also presented a novel branching strategy allowing several variables to be fixed at each node and the size of the branching tree to be reduced. Burkard and Rudolf (1993) reported computational experiments with different branch-and-bound schemes. Their tests showed that the classical branching rule with a reduction step in every node of the search tree and bounds computed by subgradient methods yielded the best results.

Given N^3 cost coefficients b_{ijp}, the formulation of the P3AP is

$$\min \quad \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{p=1}^{N} b_{ijp}x_{ijp} \tag{2-22a}$$

$$\text{s.t.} \quad \sum_{i=1}^{N} x_{ijp} = 1 \qquad j, p = 1,\ldots,N, \tag{2-22b}$$

$$\sum_{j=1}^{N} x_{ijp} = 1 \qquad i, p = 1,\ldots,N, \tag{2-22c}$$

$$\sum_{p=1}^{N} x_{ijp} = 1 \qquad i, j = 1,..., N , \tag{2-22d}$$

$$x_{ijp} \in \{0,1\} \qquad i, j, p = 1,..., N . \tag{2-22e}$$

The feasible solutions to the P3AP correspond to Latin squares, see Laywine and Mullen (1998). As for the geometric interpretation of the P3AP, given 3 disjoint N-sets A_1, A_2, and A_3. For each triple in $A_1 \times A_2 \times A_3$, a cost b_{ijp} is known. The P3AP is to find N^2 triples such that each pair of elements from $(A_1 \times A_2) \cup (A_1 \times A_3) \cup (A_2 \times A_3)$ occurs in exactly one triple and the total cost is minimized.

Frieze (1983) proved that the P3AP is NP-hard. The P3AP finds interesting applications in time tabling problems. Euler and Verge (1996) studied time tables and related polyhedra. There is not much research published on solving the P3AP. However, a branch-and-bound procedure can be found in Magos and Miliotis (1994).

2.3.3 The general case of the MAP

The A3AP and P3AP are two types of 3-dimensional assignment problems. As one could imagine, in case of a k-dimensional assignment problem, there are $k-1$ possible variants. Spieksma (2000) gave the compact formulation of the MAP, together with a survey on its complexity, applications and approximation methods. He used another parameter q to specify a k-dimensional assignment problem, referred to the resulting problem as a q-fold kAP. Appa et al. (2006) established a framework of multidimensional assignment polytopes for polyhedral analysis.

The general formulation of the axial MAP is

$$\min \quad \sum_{i_1=1}^{N} \cdots \sum_{i_k=1}^{N} c_{i_1 \ldots i_k} x_{i_1 \ldots i_k} \tag{2-23a}$$

$$\text{s.t.} \quad \sum_{i_2=1}^{N} \cdots \sum_{i_k=1}^{N} x_{i_1 \ldots i_k} = 1 \qquad i_1 = 1,\ldots,N\,, \tag{2-23b}$$

$$\sum_{i_1=1}^{N} \cdots \sum_{i_{d-1}=1}^{N} \sum_{i_{d+1}=1}^{N} \cdots \sum_{i_k=1}^{N} x_{i_1 \ldots i_k} = 1 \qquad d = 2,\ldots,k-1; \text{and } i_d = 1,\ldots,N\,, \tag{2-23c}$$

$$\sum_{i_1=1}^{N} \cdots \sum_{i_{k-1}=1}^{N} x_{i_1 \ldots i_k} = 1 \qquad i_k = 1,\ldots,N\,, \tag{2-23d}$$

$$x_{i_1 \ldots i_k} \in \{0,1\} \qquad i_1, i_2, \ldots, i_k = 1,\ldots,N\,, \tag{2-23e}$$

with N^k cost coefficients $c_{i_1 \ldots i_k}$. More simply, it turns out that solving the axial MAP involves finding $k-1$ permutations $\varphi_1, \varphi_2, \ldots, \varphi_{k-1} \in \Gamma_N$ that minimize the following objective function

$$\sum_{i=1}^{N} c_{i\varphi_1(i)\varphi_2(i)\ldots\varphi_{k-1}(i)}\,. \tag{2-24}$$

The axial MAP is NP-hard in general, but in the case that the array of the cost coefficients is a Monge array, it is polynomially solvable by the identical permutations $\varphi_i = ik$, for $i = 1,\ldots,k-1$ (see Burkard, Klinz and Rudolf (1996)). However, Burkard, Rudolf and Woeginer (1996) showed that the axial MAP remains to be NP-hard for $k \geq 3$, if the cost array is inverse Monge. The axial MAP has recently been modeled in study of a class of data association problems for multi-sensor data fusion and multi-target tracking. Burkard and Çela (1999) surveyed recent developments and applications in the axial MAP.

Throughout the remainder of this dissertation, the axial 3-dimensional assignment problem (A3AP) is referred to simply as the 3-dimensional assignment problem (3AP).

2.4 The quadratic 3-dimensional assignment problem (Q3AP)

2.4.1 Background and problem formulation

The quadratic 3-dimensional assignment problem (Q3AP) was first introduced in a technical memorandum by Pierskalla (1967b). Yet, no work on this topic was published in the open literature until Samra, Ding and Hahn (2003, 2004) re-discovered the Q3AP when working on a wireless communication design problem. The motivation is the implementation of a hybrid Automatic Repeat reQuest (ARQ) scheme in multiple packet transmissions of finding optimal symbol mappings that minimize bit error rate (BER).

The Q3AP is formulated as follows.

$$\min \left\{ \begin{array}{l} \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{p=1}^{N} b_{ijp}x_{ijp} + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{p=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{N}\sum_{\substack{n=1\\n\neq j}}^{N}\sum_{\substack{q=1\\q\neq p}}^{N} c_{ijpknq}x_{ijp}x_{knq} \\ : \boldsymbol{x} \in \boldsymbol{I} \cap \boldsymbol{J} \cap \boldsymbol{P}, \boldsymbol{x} \text{ binary} \end{array} \right\}, \tag{2-25a}$$

where $\boldsymbol{I}, \boldsymbol{J}, \boldsymbol{P}$ are disjoint sets with identical cardinality $|\boldsymbol{I}| = |\boldsymbol{J}| = |\boldsymbol{P}| = N$,

$$\boldsymbol{I} = \left\{ \boldsymbol{x} \geq 0 : \sum_{j=1}^{N}\sum_{p=1}^{N} x_{ijp} = 1, \forall \left(i = 1, ..., N \right) \right\}, \tag{2-25b}$$

$$\boldsymbol{J} = \left\{ \boldsymbol{x} \geq 0 : \sum_{i=1}^{N}\sum_{p=1}^{N} x_{ijp} = 1, \forall \left(j = 1, ..., N \right) \right\}, \tag{2-25c}$$

$$\boldsymbol{P} = \left\{ \boldsymbol{x} \geq 0 : \sum_{i=1}^{N} \sum_{j=1}^{N} x_{ijp} = 1, \forall \left(p = 1, \ldots, N \right) \right\}. \tag{2-25d}$$

The problem of Q3AP is so named as its objective optimizes a quadratic function over the 3-dimensional assignment polytope $\boldsymbol{x} \in \boldsymbol{I} \cap \boldsymbol{J} \cap \boldsymbol{P}$. The quadratic expression in the objective function does not include any $x_{ijp} x_{knq}$ terms for which $k = i$ or $n = j$ or $q = p$, since if $k = i$ and $n = j$ and $q = p$ then $x_{ijp} x_{knq} = x_{ijp}$; otherwise $x_{ijp} x_{knq} = 0$.

2.4.2 Complexity and preliminary solution methods

The Q3AP is an extension of the quadratic assignment problem and the axial 3-dimensional assignment problem, both of which are NP-hard. Therefore, the Q3AP itself is also NP-hard.

Hahn et al. (2006b) presented a basic branch-and-bound algorithm based on the level-1 RLT formulation of the Q3AP. They also investigated four heuristic methods to provide optimal or near optimal solutions within reasonable computational time. Both exact and approximate solution methods were applied to a set of benchmark instances created from the QAP instances. The branch-and-bound algorithm was able to optimally solve the instances up to size of 13.

2.5 The reformulation-linearization technique (RLT)

2.5.1 Introduction

The reformulation-linearization technique (RLT) is a strategy developed for generating tight linear programming relaxations for discrete and continuous nonconvex problems with application to not only construct exact solution algorithms but design powerful heuristic procedures as well. The first effort focuses on linear and polynomial zero-one programming problems and mixed-integer zero-one programming problems (Adams and Sherali 1986, 1990; Sherali and Adams 1990, 1994), with its several enhancement and extensions working on the more general family of discrete and continuous programming problems (Sherali and Adams 1998, 1999).

In its implication to the mixed-integer zero-one programs involving n binary variables, RLT establishes an n-level hierarchy of relaxations spanning from the ordinary linear programming relaxation to the convex hull of feasible solutions. For a given level $d \in \{1,...,n\}$, RLT constructs various polynomial factors of degree d comprised of the product of some d binary variables x_j or their complements $\left(1-x_j\right)$. The procedure essentially works via two steps. First it <u>reformulates</u> the problem in which redundant nonlinear restrictions are generated by multiplying each of the defining constraints with the product factors. Then it <u>linearizes</u> each distinct nonlinear term by replacing it with a single continuous variable throughout the objective and constraints. Hence, the original problem develops into a mixed-integer zero-one linear representation in a higher dimensional space. At each level of the RLT hierarchy, the resulting continuous relaxation is at least as tight as its previous level, with the highest n-th level representing the convex hull of the feasible region. The facets of the convex hull of feasible solutions in terms of the original problem variables are obtained through a standard projection operation on the final relaxation.

RLT has been applied to solve various specific discrete and continuous nonconvex programming problems presenting advances in computation. Even the weakest level-1 RLT formulation was shown to subsume and unify many published linearization schemes in the literature by helping design algorithms that provide significantly tighter lower bounds and often times yield an optimal solution.

RLT is among the class of lift-and-project schemes to represent the general zero-one polytope arising from combinatorial optimization problems as the projection of a higher dimensional representation. Other popular methods developed are by Lovász and Schrijver (1991), and recently by Lasserre (2001). Their common feature is the construction of an n-level hierarchy of relaxations which leads to the exact convex hull at the highest n-th level. The relaxations are linear (in the case of RLT) or semidefinite (in the case of Lovász-Schrijver and Lasserre). Laurent (2003) gave a full description and comparison of these three hierarchies. With successful experiments in the QAP and other problems, the RLT approach has shown desirable quality in tight lower bounds and fast runtimes which encourages further research.

2.5.2 Applications to the quadratic assignment problem

The RLT hierarchy has been used on the quadratic assignment problem (QAP) showing computational advances on developing strong lower bounds.

Consider the standard mathematical formulation of the QAP stated below.

$$\min \left\{ \sum_{i=1}^{N}\sum_{j=1}^{N} b_{ij}x_{ij} + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{N}\sum_{\substack{n=1\\n\neq j}}^{N} c_{ijkn}x_{ij}x_{kn} : \boldsymbol{x}\in\boldsymbol{X},\, \boldsymbol{x} \text{ binary} \right\}, \tag{2-26a}$$

$$\text{where } \boldsymbol{x}\in\boldsymbol{X} \equiv \left\{ \boldsymbol{x}\geq 0 : \sum_{i=1}^{N} x_{ij} = 1, \forall\left(j=1,\ldots,N\right); \sum_{j=1}^{N} x_{ij} = 1, \forall\left(i=1,\ldots,N\right) \right\}. \tag{2-26b}$$

There is no quadratic term $x_{ij}x_{kn}$ in the objective when $k=i$ or $n=j$ since the constraints force $x_{ij}x_{kn}=x_{ij}$ if $k=i$ and $n=j$, and $x_{ij}x_{kn}=0$ otherwise.

The level-1 RLT formulation of QAP is constructed as follows. In its reformulation step, multiply each of $2N$ equality constraints and each of N^2 nonnegativity restrictions (which are rewritten in the form of variables x_{kn}) by each of N^2 binary variables x_{ij}. Append all these new restrictions. Express the resulting products in the order $x_{ij}x_{kn}$. Substitute $x_{ij}=x_{ij}^2$ and set $x_{ij}x_{kn}=0$ if $\left(k=i \text{ and } n\neq j\right)$ or $\left(k\neq i \text{ and } n=j\right)$ throughout the formulation. Then, in the linearization step, substitute every occurrence of each product $x_{ij}x_{kn}$ $\left(k\neq i \text{ and } n\neq j\right)$ with a single nonnegative continuous variable y_{ijkn}. And enforce the trivial restrictions that $y_{ijkn}=y_{knij}$ $\forall\left(i,j,k,n=1,\ldots,N\right), k>i, n\neq j$. The level-1 RLT formulation then merges.

[RLT1]

$$\min \sum_{i=1}^{N}\sum_{j=1}^{N} b_{ij}x_{ij} + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{N}\sum_{\substack{n=1\\n\neq j}}^{N} c_{ijkn}y_{ijkn} \tag{2-27a}$$

$$\text{s.t.} \quad \sum_{\substack{k=1\\k\neq i}}^{N} y_{ijkn} = x_{ij} \qquad i,j,n=1,\ldots,N; n\neq j, \tag{2-27b}$$

$$\sum_{\substack{n=1 \\ n \neq j}}^{N} y_{ijkn} = x_{ij} \qquad i, j, k = 1, ..., N; k \neq i, \qquad \text{(2-27c)}$$

$$y_{ijkn} = y_{knij} \qquad i, j, k, n = 1, ..., N; k > i, n \neq j, \qquad \text{(2-27d)}$$

$$y_{ijkn} \geq 0 \qquad i, j, k, n = 1, ..., N; k \neq i, n \neq j, \qquad \text{(2-27e)}$$

$$\boldsymbol{x} \in \boldsymbol{X}, \boldsymbol{x} \text{ binary}. \qquad \text{(2-27f)}$$

As presented in Adams and Johnson (1994), the level-1 RLT relaxation was shown to subsume and unify alternate linear formulations of the QAP, and the resulting lower bounds to dominate the majority of other bounding techniques in terms of the strength of their continuous relaxations. Due to the special block-diagonal structure of the Lagrangian dual which relaxing symmetric constraints (2-27d) on the complementary pairs, a dual-ascent procedure was developed to solve the dual of the relaxation and obtain lower bounds. Specifically, a decomposition strategy partitioned a size N QAP into N^2+1 separate linear assignment problems, N^2 of size $N-1$ and one of size N. Resende et al. (1995) solved the continuous relaxation of the level-1 RLT formulation to optimality using an interior point algorithm. They used a smaller formulation via the substitution of variables suggested by constraints (2-27d). Hahn and Grant (1998) gave a different interpretation of the same decomposition for lower bound calculation by taking advantage of the restriction (2-27d). Algorithms implemented with the level-1 RLT gained remarkable success as several hard QAP instances were solved for the first time in the literature. See QAPLIB homepage (http://www.seas.upenn.edu/qaplib/) for reference.

Recently, efforts have been done on the level-2 RLT representation of the QAP. The formulation is constructed following the similar procedure as the level-1 case. In its reformulation step, multiply each of $2N$ equality constraints and each of N^2 nonnegativity restrictions (which are rewritten in variables x_{kn}) by each of N^2 binary variables x_{ij}. Again, multiply each of $2N$ equality constraints and each of N^2 nonnegativity restrictions (which are rewritten in variables x_{pq}) by each of $N^2\left(N-1\right)^2$ pairwise products of variables $x_{ij}x_{kn}\left(k \neq i \text{ and } n \neq j\right)$. Append all these new restrictions. Express the various resulting products in the order $x_{ij}x_{kn}$ and $x_{ij}x_{kn}x_{pq}$. Substitute $x_{ij} = x_{ij}^2$, and reduce $x_{ij}x_{ij}x_{kn}$ and $x_{kn}x_{ij}x_{kn}$ to $x_{ij}x_{kn}$. Set $x_{ij}x_{kn} = 0$ if $\left(k = i \text{ and } n \neq j\right)$ or $\left(k \neq i \text{ and } n = j\right)$ in all quadratic and cubic expressions. And set $x_{ij}x_{kn}x_{pq} = 0$ if $\left(p = i \text{ and } q \neq j\right)$, $\left(p = k \text{ and } q \neq n\right)$, $\left(p \neq i \text{ and } q = j\right)$ or $\left(p \neq k \text{ and } q = n\right)$ in all cubic expressions. Then, in the linearization step, substitute every occurrence of each product $x_{ij}x_{kn}\left(k \neq i \text{ and } n \neq j\right)$ with a single nonnegative continuous variable y_{ijkn}. And substitute every occurrence of each product $x_{ij}x_{kn}x_{pq}\left(p \neq k \neq i \text{ and } q \neq n \neq j\right)$ with a single nonnegative continuous variable z_{ijknpq}. Enforce the restrictions that $y_{ijkn} = y_{knij}\ \forall\left(i, j, k, n = 1, ..., N\right), k > i, n \neq j$, and also enforce the restrictions that $z_{ijknpq} = z_{knijpq} = z_{ijpqkn} = z_{knpqij} = z_{pqijkn} = z_{pqknij}$ $\forall\left(i, j, k, n, p, q = 1, ..., N\right), p > k > i, q \neq n \neq j$. Then, the level-2 RLT formulation is given below, where the coefficients d_{ijknpq} found in the objective function are all zero. Notice that the level-2 RLT model here allows nonzero d_{ijknpq} values, so that it generally handles cubic assignment problems which will be defined later.

$\left[\text{RLT2}\right]$

$$\min \sum_{i=1}^{N}\sum_{j=1}^{N} b_{ij}x_{ij} + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{N}\sum_{\substack{n=1\\n\neq j}}^{N} c_{ijkn}y_{ijkn} + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{N}\sum_{\substack{n=1\\n\neq j}}^{N}\sum_{\substack{p=1\\p\neq i,k}}^{N}\sum_{\substack{q=1\\q\neq j,n}}^{N} d_{ijknpq}z_{ijknpq} \qquad \text{(2-28a)}$$

$$\text{s.t.} \quad \sum_{\substack{p=1\\p\neq i,k}}^{N} z_{ijknpq} = y_{ijkn} \qquad i,j,k,n,q=1,\dots,N;\ q\neq n\neq j, k\neq i, \qquad \text{(2-28b)}$$

$$\sum_{\substack{q=1\\q\neq j,n}}^{N} z_{ijknpq} = y_{ijkn} \qquad i,j,k,n,p=1,\dots,N;\ p\neq k\neq i, n\neq j, \qquad \text{(2-28c)}$$

$$z_{ijknpq} = z_{knijpq} = z_{ijpqkn} = z_{knpqij} = z_{pqijkn} = z_{pqknij}$$

$$i,j,k,n,p,q=1,\dots,N;\ p>k>i, q\neq n\neq j, \qquad \text{(2-28d)}$$

$$z_{ijknpq} \geq 0 \qquad i,j,k,n,p,q=1,\dots,N;\ p\neq k\neq i, q\neq n\neq j, \qquad \text{(2-28e)}$$

$$\sum_{\substack{k=1\\k\neq i}}^{N} y_{ijkn} = x_{ij} \qquad i,j,n=1,\dots,N;\ n\neq j, \qquad \text{(2-28f)}$$

$$\sum_{\substack{n=1\\n\neq j}}^{N} y_{ijkn} = x_{ij} \qquad i,j,k=1,\dots,N;\ k\neq i, \qquad \text{(2-28g)}$$

$$y_{ijkn} = y_{knij} \qquad i,j,k,n=1,\dots,N;\ k>i, n\neq j, \qquad \text{(2-28h)}$$

$$y_{ijkn} \geq 0 \qquad i,j,k,n=1,\dots,N;\ k\neq i, n\neq j, \qquad \text{(2-28i)}$$

$$\boldsymbol{x}\in\boldsymbol{X}, \boldsymbol{x} \text{ binary}. \qquad \text{(2-28j)}$$

Ramakrishnan et al. (2002) produced superior lower bounds (equal to the optimal solution value) by the continuous QAP level-2 RLT relaxation. They enforced the constraints (2-28d) to combine complementary variables and reduce the number of constraints and variables. However, because of the overwhelming problem size, their interior point method was only able to compute for instances up to size 12 though.

Adams et al. (2007) handled the level-2 RLT form via a Lagrangian approach, which extended from their earlier work on level-1 RLT, by exploiting the plentiful block-diagonal structure. As a result, the branch-and-bound algorithm embedded with the level-2 RLT bounding strength to fathom nodes within the enumeration efficiently solved difficult instances of size 30.

2.6 Conclusion

This chapter has reviewed the quadratic assignment problem (QAP) with its related problems and the reformulation-linearization technique (RLT), which are employed for further discussion in Chapters 3, 4, 5 and 6. Chapter 3 investigates the inherent relationship of the 3-dimensional assignment problem (3AP) to the quadratic assignment problem (QAP) and the quadratic 3-dimensional assignment problem (Q3AP). Chapter 4 improves the understanding of the generalized quadratic assignment problem (GQAP) and its level-1 RLT formulation. Chapter 5 extends the study of GQAP and Q3AP to new applications. Chapter 6 covers the level-3 RLT model of the QAP and presents its superior lower bound calculation.

3. Relationship of the 3-dimensional Assignment Problem (3AP) to Other Assignment Problems

3.1 Introduction

Assignment problems, in their most general cases, deal with assigning a set of agents to some set of tasks. Any agent can be assigned to a task, incurring a corresponding cost which depends on the agent-task assignment. It is required to perform all tasks by assigning exactly one and only one agent to each task in a way that minimizes the total cost of the assignment.

Assignment problems vary in their description. Specifically, with different dimensionality of the objective function, one has two-dimensional, three-dimensional and multi-dimensional assignment problems. With different order of the objective function, one has linear and quadratic assignment problems. With the knapsack constraint on each agent instead of the multiple choice constraint, one has the generalized assignment problem with possible many-task-to-one-agent assignment pair. Furthermore, in the solution approaches of linearization and relaxation options, one has different level of the reformulation-linearization technique (RLT) formulation for each assignment problem model.

In this chapter, I address the inherent relationship on the 3-dimensional assignment problem (3AP), the quadratic assignment problem (QAP) and the quadratic 3-dimensional assignment problem (Q3AP). In a later chapter, I will show that the hierarchy of the quadratic, cubic and bi-quadratic assignment is directly related to the

RLT hierarchy. The purpose there is to present a broader and more connected understanding of the QAP and its related assignment problems.

3.2 Two equivalent 3AP formulations

The 3-dimensional assignment problem (3AP) can be stated as a 0-1 programming problem with its objective to minimize the cost function over a 3-dimensional assignment polytope, see Balas and Saltzman (1991).

$$\min \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{p=1}^{N} b_{ijp}x_{ijp} \tag{3-1a}$$

$$\text{s.t.} \quad \sum_{i=1}^{N}\sum_{j=1}^{N} x_{ijp} = 1 \qquad p = 1,\dots,N, \tag{3-1b}$$

$$\sum_{i=1}^{N}\sum_{p=1}^{N} x_{ijp} = 1 \qquad j = 1,\dots,N, \tag{3-1c}$$

$$\sum_{j=1}^{N}\sum_{p=1}^{N} x_{ijp} = 1 \qquad i = 1,\dots,N, \tag{3-1d}$$

$$x_{ijp} \in \{0,1\} \qquad i, j, p = 1,\dots,N. \tag{3-1e}$$

An equivalent formulation follows, by letting $x_{ijp} = u_{ij}w_{ip}$, where u_{ij} and w_{ip} are both binary variables. It holds since the 3AP has $N! \times N!$ feasible solutions, with each solution is completely determined by two permutations on $\{1,\dots,N\}$.

$$\min \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{p=1}^{N} b_{ijp}u_{ij}w_{ip} \tag{3-2a}$$

$$\text{s.t.} \quad \sum_{i=1}^{N} u_{ij} = 1 \qquad j = 1,\dots,N, \tag{3-2b}$$

$$\sum_{j=1}^{N} u_{ij} = 1 \qquad i = 1,\ldots,N\,, \tag{3-2c}$$

$$\sum_{i=1}^{N} w_{ip} = 1 \qquad p = 1,\ldots,N\,, \tag{3-2d}$$

$$\sum_{p=1}^{N} w_{ip} = 1 \qquad i = 1,\ldots,N\,, \tag{3-2e}$$

$$u_{ij}, w_{ip} \in \{0,1\} \qquad i, j, p = 1,\ldots,N\,. \tag{3-2f}$$

Or, more concisely, the 3AP can be presented as

$$\min \left\{ \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{p=1}^{N} b_{ijp} u_{ij} w_{ip} : \boldsymbol{u}, \boldsymbol{w} \in \boldsymbol{X} ; \boldsymbol{u}, \boldsymbol{w} \text{ binary} \right\}, \tag{3-3a}$$

where $\boldsymbol{x} \in \boldsymbol{X} \equiv \left\{ \boldsymbol{x} \geq \boldsymbol{0} : \sum_{i=1}^{N} x_{ij} = 1, \forall \left(j = 1,\ldots,N \right); \sum_{j=1}^{N} x_{ij} = 1, \forall \left(i = 1,\ldots,N \right) \right\}$. (3-3b)

3.3 Solving the 3AP as a QAP

The representation of the 3AP in (3-3) inspires the disclosure of relationship between the 3AP and QAP. This relationship was suggested by my advisor, Professor Peter M. Hahn of the University of Pennsylvania, along with an outline of the proof (personal communication, March 25, 2006).

Consider the QAP formulation of size $2N$,

$$\min \left\{ \sum_{m=1}^{2N} \sum_{n=1}^{2N} \sum_{s=1}^{2N} \sum_{t=1}^{2N} Q_{mnst} x_{mn} x_{st} : \boldsymbol{x} \in \boldsymbol{X'}, \boldsymbol{x} \text{ binary} \right\}, \tag{3-4a}$$

where $\boldsymbol{x} \in \boldsymbol{X'} \equiv \left\{ \boldsymbol{x} \geq \boldsymbol{0} : \sum_{m=1}^{2N} x_{mn} = 1, \forall \left(n = 1,\ldots,2N \right); \sum_{n=1}^{2N} x_{mn} = 1, \forall \left(m = 1,\ldots,2N \right) \right\}$. (3-4b)

If one expects that the possible solutions to the above QAP will be within two exclusive sections of the $2N \times 2N$ permutation matrix only where

$$x_{mn} \left(m \le N \text{ and } n \le N \right) \text{ or } \left(m > N \text{ and } n > N \right), \tag{3-5}$$

then (3-5) holds by restricting possible solutions to be

$$x_{mn} = 0 \qquad \text{if } \left(m \le N \text{ and } n > N \right) \text{ or } \left(m > N \text{ and } n \le N \right), \tag{3-6}$$

or by letting

$$Q_{mnst} = \infty \qquad \text{if } \begin{array}{l} \left(m \le N \text{ and } n > N \right) \text{ or } \left(m > N \text{ and } n \le N \right) \\ \text{or } \left(s \le N \text{ and } t > N \right) \text{ or } \left(s > N \text{ and } t \le N \right) \end{array}. \tag{3-7}$$

Thus, the QAP in (3-4) may be rewritten as

$$\min \left\{ \begin{array}{l} \sum_{m=1}^{N}\sum_{n=1}^{N}\sum_{s=1}^{N}\sum_{t=1}^{N} Q_{mnst} x_{mn} x_{st} + \sum_{m=1}^{N}\sum_{n=1}^{N}\sum_{s=N+1}^{2N}\sum_{t=N+1}^{2N} Q_{mnst} x_{mn} x_{st} \\ + \sum_{m=N+1}^{2N}\sum_{n=N+1}^{2N}\sum_{s=1}^{N}\sum_{t=1}^{N} Q_{mnst} x_{mn} x_{st} + \sum_{m=N+1}^{2N}\sum_{n=N+1}^{2N}\sum_{s=N+1}^{2N}\sum_{t=N+1}^{2N} Q_{mnst} x_{mn} x_{st} \\ : \boldsymbol{x} \in \boldsymbol{X'}, \boldsymbol{x} \text{ binary;} \\ x_{mn} = 0, \text{ if } \left(m \le N \text{ and } n > N \right) \text{ or } \left(m > N \text{ and } n \le N \right) \end{array} \right\}. \tag{3-8}$$

Furthermore, since $\boldsymbol{x} \in \boldsymbol{X'}, \boldsymbol{x}$ binary and $x_{mn} = 0$ if $\left(m \le N \text{ and } n > N \right)$ or $\left(m > N \text{ and } n \le N \right)$, one has

$$\sum_{m=1}^{N} x_{mn} = 1, \forall \left(n = 1, ..., N \right); \sum_{n=1}^{N} x_{mn} = 1, \forall \left(m = 1, ..., N \right) \tag{3-9a}$$

and $$\sum_{m=N+1}^{2N} x_{mn} = 1, \forall \left(n = N+1, ..., 2N \right); \sum_{n=N+1}^{2N} x_{mn} = 1, \forall \left(m = N+1, ..., 2N \right). \tag{3-9b}$$

For $\left(m \le N, n \le N, s > N, t > N \right)$

let $$Q_{mnst} = \begin{cases} b_{ijp} & m = i, n = j, s = i + N, t = p + N \\ 0 & \text{otherwise} \end{cases}; \tag{3-10a}$$

$$\text{For} \begin{pmatrix} m \le N, n \le N, s \le N, t \le N \\ \text{or } m > N, n > N, s \le N, t \le N \\ \text{or } m > N, n > N, s > N, t > N \end{pmatrix}$$

let $\quad Q_{mnst} = 0$. (3-10b)

Therefore, the objective function that is minimized in (3-8) can be simplified as following

$$\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{p=1}^{N} Q_{i,j,i+N,p+N} x_{ij} x_{i+N,p+N} , \tag{3-11}$$

where x_{ij} and $x_{i+N,p+N}$ $\forall\left(i, j, p = 1,..., N\right)$ are binary variables which restricted to constraints (3-9). Notice that x_{ij} and $x_{i+N,p+N}$ $\forall\left(i, j, p = 1,..., N\right)$ do not overlap.

Thus, letting $u_{ij} = x_{ij}, w_{ip} = x_{i+N,p+N}$ $\forall\left(i, j, p = 1,..., N\right)$, one finally has the following formulation

$$\min \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{p=1}^{N} b_{ijp} u_{ij} w_{ip} \tag{3-12a}$$

$$\text{s.t.} \quad \sum_{i=1}^{N} u_{ij} = 1 \qquad j = 1,..., N , \tag{3-12b}$$

$$\sum_{j=1}^{N} u_{ij} = 1 \qquad i = 1,..., N , \tag{3-12c}$$

$$\sum_{i=1}^{N} w_{ip} = 1 \qquad p = 1,..., N , \tag{3-12d}$$

$$\sum_{p=1}^{N} w_{ip} = 1 \qquad i = 1,..., N , \tag{3-12e}$$

$$u_{ij}, w_{ip} \in \{0,1\} \qquad i, j, p = 1,..., N . \tag{3-12f}$$

Therefore, for a 3AP problem in (3-3) of size N, one can construct a corresponding QAP problem in (3-4) of size $2N$ where

for $\left(m \le N, n \le N, s > N, t > N\right)$,

$$Q_{mnst} = \begin{cases} b_{ijp} & m = i, n = j, s = i + N, t = p + N \\ 0 & \text{otherwise} \end{cases}, \tag{3-13a}$$

for $\begin{pmatrix} m \le N, n \le N, s \le N, t \le N \\ \text{or } m > N, n > N, s \le N, t \le N \\ \text{or } m > N, n > N, s > N, t > N \end{pmatrix}$,

$$Q_{mnst} = 0, \tag{3-13b}$$

and for $\begin{matrix} \left(m \le N \text{ and } n > N\right) \text{ or } \left(m > N \text{ and } n \le N\right) \\ \text{or } \left(s \le N \text{ and } t > N\right) \text{ or } \left(s > N \text{ and } t \le N\right) \end{matrix}$,

$$Q_{mnst} = \infty. \tag{3-13c}$$

If one obtains the solution $\boldsymbol{x} = \left(x_{mn}\right)_{2N \times 2N}$ to the QAP problem in (3-4), then one also obtains the solution to the 3AP in (3-3) by letting

$$u_{ij} = x_{mn}, \text{ for } \left(m \le N, n \le N, i \le N, j \le N, i = m, j = n\right), \tag{3-14a}$$

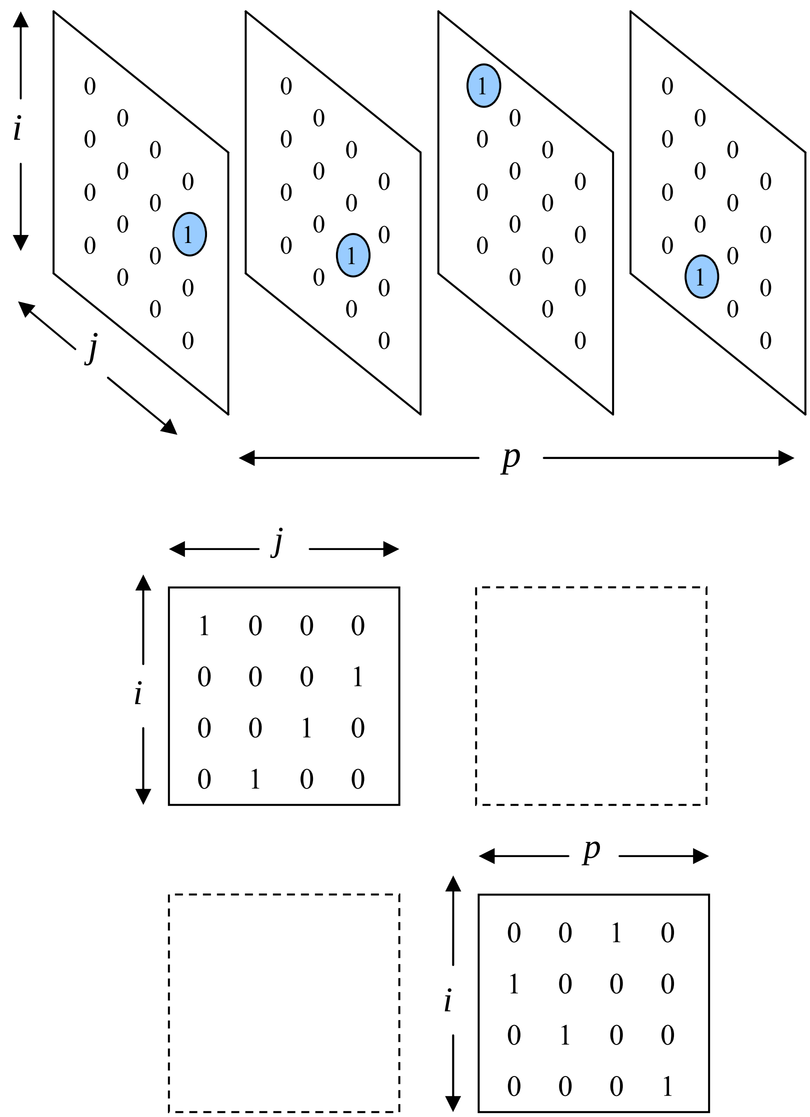
and $$w_{ip} = x_{mn}, \text{ for } \left(m > N, n > N, i \le N, p \le N, i = m - N, p = n - N\right), \tag{3-14b}$$

with the same objective value.

By solving the 3AP as a corresponding QAP, one actually lifts the original 3-dimensional N^3 assignment polytope to a higher dimensional space $\left(2N\right)^2 \times \left(2N\right)^2$ of the QAP. Section 3.5 shows the improvement of bounding quality provided by the QAP model.

Figure 3-1 gives a 3AP solution matrix of size 4 and its corresponding QAP solution matrix.

Figure 3-1. Example of a 3AP solution matrix of size 4 and its corresponding QAP solution matrix of size 8.



3.4 The level-1 RLT formulation of the Q3AP

The Q3AP is an extension of the QAP that can be represented as optimizing a quadratic function over a 3-dimensional assignment solution matrix, and it is given by the following formulation. The Q3AP is introduced earlier in Section 2.4 of this dissertation.

$$\min \left\{ \begin{array}{l} \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{p=1}^{N} b_{ijp}x_{ijp} + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{p=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{N}\sum_{\substack{n=1\\n\neq j}}^{N}\sum_{\substack{q=1\\q\neq p}}^{N} c_{ijpknq}x_{ijp}x_{knq} \\ : \boldsymbol{x} \in \boldsymbol{I} \cap \boldsymbol{J} \cap \boldsymbol{P}, \boldsymbol{x} \text{ binary} \end{array} \right\}, \tag{3-15a}$$

where $\boldsymbol{I}, \boldsymbol{J}, \boldsymbol{P}$ are disjoint sets with identical cardinality $|\boldsymbol{I}| = |\boldsymbol{J}| = |\boldsymbol{P}| = N$,

$$\boldsymbol{I} = \left\{ \boldsymbol{x} \geq 0 : \sum_{j=1}^{N}\sum_{p=1}^{N} x_{ijp} = 1, \forall \left(i = 1, ..., N \right) \right\}, \tag{3-15b}$$

$$\boldsymbol{J} = \left\{ \boldsymbol{x} \geq 0 : \sum_{i=1}^{N}\sum_{p=1}^{N} x_{ijp} = 1, \forall \left(j = 1, ..., N \right) \right\}, \tag{3-15c}$$

$$\boldsymbol{P} = \left\{ \boldsymbol{x} \geq 0 : \sum_{i=1}^{N}\sum_{j=1}^{N} x_{ijp} = 1, \forall \left(p = 1, ..., N \right) \right\}. \tag{3-15d}$$

The level-1 RLT formulation of the Q3AP, which was presented in Hahn et al. (2006b) and in Kim (2006), is constructed as follows. In its reformulation step, multiply each of $3N$ equality constraints and each of N^3 nonnegativity restrictions in (3-15b)-(3-15d) (which are rewritten in the form of variables x_{knq}) by each of N^3 binary variables x_{ijp}. Append all these new restrictions. Express the resulting product in the order $x_{ijp}x_{knq}$. Throughout the formulation, substitute $x_{ijp} = x_{ijp}^2$. If $k = i$ or $n = j$ or $q = p$, set $x_{ijp}x_{knq} = 0$ unless all three equalities hold. Then, in the linearization step, for every occurrence of each product $x_{ijp}x_{knq}$ with $k \neq i, n \neq j$ and $q \neq p$, substitute a single nonnegative continuous variable y_{ijpknq}. And, enforce the trivial restrictions that $y_{ijpknq} = y_{knqijp} \quad \forall (i, j, p, k, n, q = 1, \ldots, N), k > i, n \neq j, q \neq p$. The level-1 RLT formulation 3RLT is then given by.

[3RLT]

$$\min \quad \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{p=1}^{N} b_{ijp} x_{ijp} + \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{p=1}^{N} \sum_{\substack{k=1 \\ k \neq i}}^{N} \sum_{\substack{n=1 \\ n \neq j}}^{N} \sum_{\substack{q=1 \\ q \neq p}}^{N} c_{ijpknq} y_{ijpknq} \tag{3-16a}$$

$$\text{s.t.} \quad \sum_{\substack{k=1 \\ k \neq i}}^{N} \sum_{\substack{n=1 \\ n \neq j}}^{N} y_{ijpknq} = x_{ijp} \qquad i, j, p, q = 1, \ldots, N; q \neq p, \tag{3-16b}$$

$$\sum_{\substack{k=1 \\ k \neq i}}^{N} \sum_{\substack{q=1 \\ q \neq p}}^{N} y_{ijpknq} = x_{ijp} \qquad i, j, p, n = 1, \ldots, N; n \neq j, \tag{3-16c}$$

$$\sum_{\substack{n=1 \\ n \neq j}}^{N} \sum_{\substack{q=1 \\ q \neq p}}^{N} y_{ijpknq} = x_{ijp} \qquad i, j, p, k = 1, \ldots, N; k \neq i, \tag{3-16d}$$

$$y_{ijpknq} = y_{knqijp} \qquad i, j, p, k, n, q = 1, \ldots, N; k > i, n \neq j, q \neq p, \tag{3-16e}$$

$$y_{ijpknq} \geq 0 \qquad i, j, p, k, n, q = 1, \ldots, N; k \neq i, n \neq j, q \neq p, \tag{3-16f}$$

$$\boldsymbol{x} \in \boldsymbol{I} \cap \boldsymbol{J} \cap \boldsymbol{P}, \boldsymbol{x} \text{ binary}. \tag{3-16g}$$

Problem Q3AP and 3RLT are equivalent in the following sense. Given any feasible solution $\boldsymbol{x}$ to the Q3AP, there exists a $\boldsymbol{y}$ such that $(\boldsymbol{x}, \boldsymbol{y})$ is feasible to the 3RLT with the same objective value. Conversely, for any feasible solution $(\boldsymbol{x}, \boldsymbol{y})$ to the 3RLT, the corresponding $\boldsymbol{x}$ is feasible to the Q3AP with the same objective value.

PROOF.

For a given $\boldsymbol{x}$ satisfying the constraints of the Q3AP given in (3-15a) through (3-15d), compute $y_{ijpknq} = x_{ijp}x_{knq} \; \forall(i,j,p,k,n,q)$. It is trivially true that $(\boldsymbol{x},\boldsymbol{y})$ is a feasible solution to the 3RLT and the value of objective functions of Q3AP and 3RLT matches as long as (3-15a)-(3-15d) are part of the 3RLT.

In order to prove the other direction of equivalency, it aims to show a feasible solution $(\boldsymbol{x},\boldsymbol{y})$ to the 3RLT satisfies $y_{ijpknq} = x_{ijp}x_{knq} \; \forall(i,j,p,k,n,q),\; k \neq i, n \neq j, p \neq q$. In fact, it is sufficient to show that if $(\boldsymbol{x},\boldsymbol{y})$ is feasible to the 3RLT then y_{ijpknq} is 0 unless x_{ijp} and x_{knq} are both equal to 1 in which case y_{ijpknq} is also 1.

If $x_{ijp} = 0$ for given i, j and p, then (3-16b), (3-16c), (3-16d) and (3-16f) together imply that $y_{ijphst} = 0 \; \forall(h,s,t), h \neq i, s \neq j, t \neq p$.

If $x_{knq} = 0$ for given k, n and q, then (3-16b), (3-16c), (3-16d), (3-16e) and (3-16f) together imply that $y_{luwknq} = 0 \; \forall(l,u,w), l \neq k, u \neq n, w \neq q$.

Therefore, if $x_{ijp} = x_{knq} = 0$ for given $i, j, p, k \neq i, n \neq j$ and $q \neq p$, then $y_{ijpknq} = 0$.

Besides, if $x_{ijp} = x_{knq} = 1$ for given $i, j, p, k \neq i, n \neq j$ and $q \neq p$, then it must be shown that $y_{ijpknq} = 1$.

From (3-15b) $\sum_{u=1}^{N}\sum_{w=1}^{N} x_{iuw} = 1 \; \forall(i)$, which implies that if $x_{ijp} = 1$, then

$$x_{iuw} = 0 \; \forall(u,w), u \neq j, w \neq p; \tag{3-17a}$$

$$x_{ijw} = 0 \; \forall(w), w \neq p; \tag{3-17b}$$

and $\quad x_{iup} = 0 \ \forall (u), u \neq j$. (3-17c)

Thus, similarly as above, (3-16b), (3-16c), (3-16d) and (3-16f) together imply that

$$y_{iuwhst} = 0 \ \forall (u,w,h,s,t), u \neq j, w \neq p, h \neq i, s \neq j \text{ or } u, t \neq p \text{ or } w; \tag{3-18a}$$

$$y_{ijwhst} = 0 \ \forall (w,h,s,t), w \neq p, h \neq i, s \neq j, t \neq p \text{ or } w; \tag{3-18b}$$

and $\quad y_{iuphst} = 0 \ \forall (u,h,s,t), u \neq j, h \neq i, s \neq j \text{ or } u, t \neq p$. (3-18c)

So, $\quad y_{iuwknq} = 0 \ \forall (u,w), u \neq j \text{ or } n, w \neq p \text{ or } q, k \neq i$; (3-19a)

$$y_{ijwknq} = 0 \ \forall (w), w \neq p \text{ or } q, k \neq i; \tag{3-19b}$$

and $\quad y_{iupknq} = 0 \ \forall (u), u \neq j \text{ or } n, k \neq i$. (3-19c)

By (3-16d) and (3-16e), it turns out that

$$\sum_{\substack{u=1 \\ u \neq n}}^{N} \sum_{\substack{w=1 \\ w \neq q}}^{N} y_{iuwknq} = x_{knq} \ \forall (i), i \neq k. \tag{3-20}$$

Therefore,

$$\sum_{\substack{u=1 \\ u \neq n}}^{N} \sum_{\substack{w=1 \\ w \neq q}}^{N} y_{iuwknq} = \sum_{\substack{u=1 \\ u \neq j,n}}^{N} \sum_{\substack{w=1 \\ w \neq p,q}}^{N} y_{iuwknq} + \sum_{\substack{w=1 \\ w \neq p,q}}^{N} y_{iuwknq} + \sum_{\substack{u=1 \\ u \neq j,n}}^{N} y_{iuwknq} + y_{ijpknq} = y_{ijpknq} = x_{knq}, \tag{3-21}$$

which implies that if $x_{knq} = 1$ then $y_{ijpknq} = 1$. This completes the proof. □

The Lagrangian dual of the 3RLT, with constraints (3-16e) being relaxed into the objective function, possesses a block-diagonal structure that can be exploited. In particular, the 3RLT then reduces to $N^3 + 1$ 3APs, one over each (i, j, p) triplet in (3-16b), (3-16c) and (3-16d) and one over $\boldsymbol{x} \in \boldsymbol{I} \cap \boldsymbol{J} \cap \boldsymbol{P}$. Hahn et al. (2006b) presented a branch-and-bound algorithm involving a dual procedure for calculating level-1 RLT lower bounds. They took advantage of the symmetric constraints (3-16e) to obtain a smaller formulation by reducing the number of variables and constraints. They strictly enforced constraints (3-16e) so that their method provided tighter lower bounds than the model with constraints (3-16e) being relaxed.

In the following paragraphs, I describe my experiments using the Q3AP branch-and-bound algorithm developed by the University of Pennsylvania research team in order to solve the 3AP. This is done by forcing all the quadratic coefficients c_{ijpknq} of the objective function to be zero. By doing so, the 3-dimensional variables in the 3AP are expanded into the higher dimensional Q3AP representation. Thus, the problems of Q3AP and 3AP have become closely connected together, in spite of the different order objective functions, over the same 3-dimensional assignment matrix.

3.5 Computational experiments

In this section, I present the computational results on the experiments of solving the 3AP as a QAP or a Q3AP. All 3AP instance data are chosen from Balas and Saltzman (1991) test problems. For each size, there are five instances, whose cost matrices were generated from uniform distribution between integer [0, 99]. I use the strict QAP code from Hahn and Grant (1998), and the Q3AP code from Hahn et al. (2006b). All experiments were conducted on a single 1.2GHz CPU of a Sun Fire V880 server.

Table 3-1 shows the comparison of the lower bounds with the optimal solution values on the sixty 3AP test instances from Balas and Saltzman (1991). For each size, the average value over five instances is given. The OPTIMAL column shows the average optimal solution values for each size. The SO^0 and SO^c columns list the root bound (at the root node of the search tree) obtained by the subgradient optimization procedure without and with cuts, respectively, in Balas and Saltzman (1991). The HM+PT+1_T column gives the tightest value from four lower bound techniques by Kim (2006). The 3AP-QAP column presents the root bounds by solving the 3AP as a QAP, which is described in Section 3.3. One can observe that the 3AP-QAP lower bound calculation outperforms other methods, except once with a tiny gap.

Table 3-1. Comparison of lower bound techniques on the 3AP instances.

Size	OPTIMAL	SO^0	SO^c	HM+PT+1_T	3AP-QAP
4	42.2	39.96	40.77	40.2	42.19
6	40.2	35.25	35.75	35.0	39.30
8	23.8	18.55	19.09	18.8	22.97
10	19.0	15.58	16.43	14.8	18.46
12	15.6	13.63	13.78	12.6	15.04
14	10.0	6.67	7.20	6.4	8.18
16	10.0	6.50	6.67	6.2	7.95
18	6.4	3.59	3.78	2.2	4.21
20	4.8	1.49	1.88	1.4	2.36
22	4.0	1.45	1.70	0.8	1.85
24	1.8	0.15	0.23	0.2	0.22
26	1.0	0.11*	0.19*	0.2	0.20

-For each size, the average value over five instances solved to optimality is given.
(* Three instances for $N = 26$).

Table 3-2 presents the branch-and-bound runtime and node count of solving the 3AP as a QAP and solving the 3AP as a Q3AP, compared with the results from Balas and Saltzman (1991). The method of solving 3AP as a QAP requires much more time. Recall that the QAP and Q3AP codes utilized in these experiments were the original codes for solving QAP and Q3AP problems. In order to implement their use for 3AP solutions, an artifice of raising costs of disallowed objective function coefficients served to reduce the solution set to only valid 3AP solutions. As a consequence, the codes in these experiments underwent large volumes of unnecessary operations, and thus took much longer than would codes specifically designed for this purpose. Future investigations should address this important limitation of our experiments.

Table 3-2. Performance of branch-and-bound algorithms on the 3AP instances.

Size	Balas&Saltzman		3AP-QAP		3AP-Q3AP	
	CPU sec.	Nodes	CPU sec.	Nodes	CPU sec.	Nodes
4	0.01	2.4	0.02	33.8	0.01	31.6
6	0.02	9.2	0.13	79.8	0.02	124.0
8	0.04	21.8	1.54	159.8	0.03	383.0
10	0.06	29.6	6.65	206.2	0.04	674.8
12	0.08	26.2	28.43	305.2	0.05	1,099.4
14	0.41	128.2	212.19	975.8	0.27	6,282.0
16	1.24	378.8	780.47	1,043.0	0.70	12,627.2
18	1.74	427.8	3,194.97	2,100.6	1.44	21,303.4
20	5.25	869.6	>5,000	N/A	10.22	140,961.6
22	11.02	1,605.0			21.38	247,693.2
24	18.85	1,815.8			281.87	3,290,514.8
26	41.74	3,716.8			1,823.37	19,322,907.4

-Each line in the table represents the averages over five instances in each size.
-CPU runtime in seconds is normalized to the speed of V880 server.
-N/A = not available.

3.6 Conclusion

This chapter reports the relationship on the quadratic assignment problem (QAP) and its related assignment problems. I begin with the method of solving the 3-dimensional assignment problem (3AP) as a corresponding QAP. The level-1 reformulation-linearization technique (RLT) formulation of the quadratic 3-dimensional assignment problem (Q3AP) when relaxing the symmetric constraints shows a special block-diagonal structure which can be reduced to a series of 3APs. I also present the experimental results of lifting the 3AP polytope to the higher dimensional spaces of QAP and Q3AP.

4. The Level-1 Reformulation-Linearization Technique (RLT) Formulation of the Generalized Quadratic Assignment Problem (GQAP)

4.1 Introduction

The generalized quadratic assignment problem (GQAP) is presented in the following mathematical programming model

$$\min \quad \sum_{i=1}^{M}\sum_{j=1}^{N} b_{ij}x_{ij} + \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{M}\sum_{n=1}^{N} c_{ijkn}x_{ij}x_{kn} \tag{4-1a}$$

$$\text{s.t.} \quad \sum_{i=1}^{M} s_{ij}x_{ij} \le S_j \qquad j=1,\dots,N\,, \tag{4-1b}$$

$$\sum_{j=1}^{N} x_{ij} = 1 \qquad i=1,\dots,M\,, \tag{4-1c}$$

$$x_{ij} \in \{0,1\} \qquad i=1,\dots,M;\ j=1,\dots,N\,. \tag{4-1d}$$

What follows is the application of the reformulation-linearization technique (RLT) to get the equivalent linearized reformulation of the GQAP. This reformulation was first outlined by Professor Hahn. Here are the details, provided by me for the paper Hahn et al. (2006a).

First, multiply each constraint in (4-1b) and (4-1c), written as $\sum_{k=1}^{M} s_{kn}x_{kn} \le S_n$ and $\sum_{n=1}^{N} x_{kn} = 1$, by each of $M \times N$ variables x_{ij}. Next, explicitly include the requirements $x_{ij}x_{kn} = x_{kn}x_{ij}\ (\forall i, j, k, n), k > i$. Then, define a nonnegative, continuous variable

$y_{ijkn} \equiv x_{ij} x_{kn}$ $(\forall i, j, k, n)$, and substitute each product of variables $x_{ij} x_{kn}$ with a single variable y_{ijkn} throughout the formulation. Finally, consider the solution structure of the GQAP indicating that $y_{ijin} = 0$ $(\forall i, j, n; n \neq j)$ and $y_{ijij} = x_{ij}$ $(\forall i, j)$. Thus, an equivalent linearized mixed-integer programming formulation LIPR emerges.

[LIPR]

$$\min \sum_{i=1}^{M} \sum_{j=1}^{N} b_{ij} x_{ij} + \sum_{i=1}^{M} \sum_{j=1}^{N} \sum_{\substack{k=1 \\ k \neq i}}^{M} \sum_{n=1}^{N} c_{ijkn} y_{ijkn} \tag{4-2a}$$

$$\text{s.t.} \quad \sum_{k=1}^{M} s_{kn} y_{ijkn} \leq S_n x_{ij} \qquad i = 1, \ldots, M;\ j, n = 1, \ldots, N\ , \tag{4-2b}$$

$$\sum_{n=1}^{N} y_{ijkn} = x_{ij} \qquad i, k = 1, \ldots, M;\ j = 1, \ldots, N; k \neq i\ , \tag{4-2c}$$

$$y_{ijkn} = y_{knij} \qquad i, k = 1, \ldots, M;\ j, n = 1, \ldots, N; k > i\ , \tag{4-2d}$$

$$\sum_{i=1}^{M} s_{ij} x_{ij} \leq S_j \qquad j = 1, \ldots, N\ , \tag{4-2e}$$

$$\sum_{j=1}^{N} x_{ij} = 1 \qquad i = 1, \ldots, M\ , \tag{4-2f}$$

$$y_{ijkn} \geq 0 \qquad i, k = 1, \ldots, M;\ j, n = 1, \ldots, N; k \neq i\ , \tag{4-2g}$$

$$x_{ij} \in \{0, 1\} \qquad i = 1, \ldots, M;\ j = 1, \ldots, N\ . \tag{4-2h}$$

Problem GQAP and LIPR are equivalent in the following sense. Given any feasible solution $\boldsymbol{x}$ to the GQAP, there exists a $\boldsymbol{y}$ such that $(\boldsymbol{x}, \boldsymbol{y})$ is feasible to the LIPR with the same objective value. Conversely, for any feasible solution $(\boldsymbol{x}, \boldsymbol{y})$ to the LIPR, the corresponding $\boldsymbol{x}$ is feasible to the GQAP with the same objective value.

PROOF.

For a given $\boldsymbol{x}$ satisfying the constraints of the GQAP given in (4-1), compute $y_{ijkn} = x_{ij}x_{kn} \ \forall(i, j, k, n)$. It is trivially true that $(\boldsymbol{x}, \boldsymbol{y})$ is a feasible solution to the LIPR and the value of objective functions of GQAP and LIPR matches as long as (4-1b), (4-1c) and (4-1d) are part of the LIPR.

In order to prove the other direction of equivalency, it aims to show that a feasible solution $(\boldsymbol{x}, \boldsymbol{y})$ to the LIPR satisfies $y_{ijkn} = x_{ij}x_{kn} \ \forall(i, j, k, n), k \neq i$. In fact, it is sufficient to show that if $(\boldsymbol{x}, \boldsymbol{y})$ is feasible to the LIP then y_{ijkn} is 0 unless x_{ij} and x_{kn} are both equal to 1 in which case y_{ijkn} is also 1.

If $x_{ij} = 0$ for given i and j, then (4-2b), (4-2c) and (4-2g) together imply that $y_{ijpq} = 0 \ \forall(p, q), p \neq i$.

If $x_{kn} = 0$ for given k and n, then (4-2b), (4-2c), (4-2d) and (4-2g) together imply that $y_{rtkn} = 0 \ \forall(r, t), r \neq k$.

Therefore, if $x_{ij} = x_{kn} = 0$ for given $i, j, k \neq i$ and n, then $y_{ijkn} = 0$.

Besides, if $x_{ij} = x_{kn} = 1$ for given $i, j, k \neq i$ and n, then it must be shown that $y_{ijkn} = 1$.

From (4-2f) $\sum_{t=1}^{N} x_{it} = 1 \ \forall(i)$, which implies that if $x_{ij} = 1$ then $x_{it} = 0 \ \forall(t), t \neq j$. Thus, similarly as above, (4-2b), (4-2c) and (4-2g) together imply that $y_{itpq} = 0 \ \forall(t, p, q), t \neq j, p \neq i$. So, $y_{itkn} = 0 \ \forall(t), t \neq j, k \neq i$.

By (4-2c) and (4-2d), it turns out that $\sum_{t=1}^{N} y_{itkn} = x_{kn}$. Therefore, $\sum_{t=1}^{N} y_{itkn} = \sum_{\substack{t=1 \\ t\neq j}}^{N} y_{itkn} + y_{ijkn} = y_{ijkn} = x_{kn}$, which implies that if $x_{kn} = 1$ then $y_{ijkn} = 1$. This completes the proof. □

Notice that a by-product of the proof is that whenever all x_{ij} are binary in the LIPR model, so are all y_{ijkn}. Thus, the LIPR is really a pure integer programming problem.

4.2 Bounding strength comparison of three level-1 RLT based formulations

If one follows the RLT operations on zero-one quadratic programming problems suggested by Adams and Sherali (1986), another set of constraints will be added to the above formulation. Those constraints are given by multiply each constraint in (4-1b), written as $\sum_{k=1}^{M} s_{kn} x_{kn} \leq S_n$, by each $(1 - x_{ij})\ \forall (i, j)$. Then it leads to the LIPT formulation.

[LIPT]

$$\min \sum_{i=1}^{M}\sum_{j=1}^{N} b_{ij}x_{ij} + \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{\substack{k=1 \\ k\neq i}}^{M}\sum_{n=1}^{N} c_{ijkn}y_{ijkn} \tag{4-3a}$$

s.t. $$\sum_{k=1}^{M} s_{kn}x_{kn} - \sum_{k=1}^{M} s_{kn}y_{ijkn} \leq S_n - S_n x_{ij}$$

$$i = 1,\ldots,M;\ j, n = 1,\ldots,N, \tag{4-3b}$$

$$\sum_{k=1}^{M} s_{kn}y_{ijkn} \leq S_n x_{ij} \qquad i = 1,\ldots,M;\ j, n = 1,\ldots,N, \tag{4-3c}$$

$$\sum_{n=1}^{N} y_{ijkn} = x_{ij} \qquad i,k = 1,...,M;\ j = 1,...,N;\ k \neq i, \qquad (4\text{-}3d)$$

$$y_{ijkn} = y_{knij} \qquad i,k = 1,...,M;\ j,n = 1,...,N;\ k > i, \qquad (4\text{-}3e)$$

$$\sum_{i=1}^{M} s_{ij} x_{ij} \leq S_j \qquad j = 1,...,N, \qquad (4\text{-}3f)$$

$$\sum_{j=1}^{N} x_{ij} = 1 \qquad i = 1,...,M, \qquad (4\text{-}3g)$$

$$y_{ijkn} \geq 0 \qquad i,k = 1,...,M;\ j,n = 1,...,N;\ k \neq i, \qquad (4\text{-}3h)$$

$$x_{ij} \in \{0,1\} \qquad i = 1,...,M;\ j = 1,...,N. \qquad (4\text{-}3i)$$

Though, constraints (4-3b) are redundant in the equivalence proof of GQAP and LIPR, they provide additional information $\sum_{k=1}^{M} s_{kn} x_{kn} - S_n \leq \sum_{k=1}^{M} s_{kn} y_{ijkn} - S_n x_{ij}$, which is not included in the continuous relaxation $\overline{\text{LIPR}}$ of LIPR, where binary constraints (4-2h) become nonnegative constraints on the variables. However, the $\overline{\text{LIPT}}$ model does not improve the linear relaxation bound for the GQAP, which will be covered later in computational experiments of this section. Moreover, the $\overline{\text{LIPR}}$ model provides the convenience of implementation for adding the resource constraints in the dual-ascent procedure for the GQAP, see Hahn et al. (2006a). Thus when applying RLT technique to the generalized quadratic 3-dimensional assignment problem in section 5.3.2, I multiple the inequality constraints with each variable $x_{ijp}\ \forall(i, j, p)$ only, not $(1 - x_{ijp})\ \forall(i, j, p)$.

Now, reconsider the PROOF in Section 4.1. If one has $x_{ij} = 0$ for given i and j, then (4-2c) and (4-2g) together are sufficient for $\sum_{q=1}^{N} y_{ijpq} = x_{ij}$, $p \neq i$, which implies that

$y_{ijpq} = 0 \; \forall (p,q), p \neq i$. For $x_{kn} = 0$ with given k and n, (4-2c), (4-2d) and (4-2g) are sufficient to obtain $y_{rtkn} = 0 \; \forall (r,t), r \neq k$, too. Therefore, when the $\boldsymbol{x}$ binary restrictions are enforced, constraints (4-2b) are redundant in the LIPR form. Hahn et al. (2006a) used the linearized reformulation LIP of the GQAP, which is repeated below.

$[\text{LIP}]$

$$\min \; \sum_{i=1}^{M}\sum_{j=1}^{N} b_{ij}x_{ij} + \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{\substack{k=1 \\ k\neq i}}^{M}\sum_{n=1}^{N} c_{ijkn}y_{ijkn} \tag{4-4a}$$

$$\text{s.t.} \quad \sum_{n=1}^{N} y_{ijkn} = x_{ij} \qquad i,k = 1,\ldots,M;\; j = 1,\ldots,N;\; k \neq i, \tag{4-4b}$$

$$y_{ijkn} = y_{knij} \qquad i,k = 1,\ldots,M;\; j,n = 1,\ldots,N;\; k > i, \tag{4-4c}$$

$$\sum_{i=1}^{M} s_{ij}x_{ij} \leq S_j \qquad j = 1,\ldots,N, \tag{4-4d}$$

$$\sum_{j=1}^{N} x_{ij} = 1 \qquad i = 1,\ldots,M, \tag{4-4e}$$

$$y_{ijkn} \geq 0 \qquad i,k = 1,\ldots,M;\; j,n = 1,\ldots,N;\; k \neq i, \tag{4-4f}$$

$$x_{ij} \in \{0,1\} \qquad i = 1,\ldots,M;\; j = 1,\ldots,N. \tag{4-4g}$$

Constraints (4-2b) are added later in their dual-ascent procedure for tightening the continuous relaxation bound provided by the $\overline{\text{LIP}}$, since (4-2b) is not implied by constraints (4-2c)-(4-2h) of the linear problem $\overline{\text{LIPR}}$.

Numerical experiments in this section show the improvement of bounding quality afforded by $\overline{\text{LIPR}}$ and $\overline{\text{LIPT}}$, compared with that afforded by the $\overline{\text{LIP}}$. Table 4-1 reports the lower bound calculations by $\overline{\text{LIP}}$, $\overline{\text{LIPR}}$ and $\overline{\text{LIPT}}$. All test instances are taken from Cordeau et al. (2006). The Cordeau instances are labeled with three numbers: M, N, and a parameter $f \in \{1,...,100\}$, which indicates the tightness of capacity constraints. All linear programming experiments were performed using GAMS. The Minimum column gives the best-known objective function minimum, since only a few instances have been solved to optimality. The $\overline{\text{LIP}}$, $\overline{\text{LIPR}}$ and $\overline{\text{LIPT}}$ columns list their linear programming bounds. % gap is the percentage of gap closed by $\overline{\text{LIPR}}$, i.e., $\left(\overline{\text{LIPR}} - \overline{\text{LIP}}\right) / \left(\text{Minimum} - \overline{\text{LIP}}\right) \times 100$.

Table 4-1. Performance of level-1 RLT based continuous relaxation bounds.

Instance Code	Minimum	$\overline{\text{LIP}}$	$\overline{\text{LIPR}}$	$\overline{\text{LIPT}}$	% Gap
20-15-35	1,471,896	1,022,008	1,341,185	1,341,185	70.95%
20-15-55	1,723,638	1,219,733	1,499,638	1,499,638	55.55%
20-15-75	1,953,188	1,378,065	1,738,220	1,738,220	62.62%
30-06-95	5,160,920	3,894,239	5,038,440	5,038,440	90.33%
30-07-75	4,383,923	3,477,379	4,203,378	4,203,378	80.08%
30-08-55	3,501,695	3,124,536	3,471,331	3,471,331	91.95%
30-10-65	3,620,959	2,816,330	3,480,870	3,480,870	82.59%
30-20-35	3,379,359	2,428,503	2,987,227	2,987,227	58.76%
30-20-55	3,593,105	2,291,918	3,138,152	3,138,152	65.04%
30-20-75	4,050,938	2,481,252	3,741,046	3,741,046	80.26%
30-20-95	5,710,645	3,477,926	4,889,298	4,889,298	63.21%
35-15-35	4,456,670	3,466,747	3,985,468	3,985,468	52.40%
35-15-55	4,639,128	3,319,052	4,289,733	4,289,733	73.53%
35-15-75	6,301,723	3,893,377	5,568,696	5,568,696	69.56%
35-15-95	6,689,421	3,946,264	5,923,302	5,923,302	72.07%
40-07-75	7,405,793	6,130,945	7,111,425	7,111,425	76.91%
40-09-95	7,667,719	5,197,627	7,312,719	7,312,719	85.63%
40-10-65	7,265,559	5,556,816	6,907,665	6,907,665	79.06%
50-10-65	10,513,029	9,397,738	10,421,100	10,421,100	91.76%
50-10-75	11,217,503	8,080,991	10,820,925	10,820,925	87.36%
50-10-95	12,845,598	9,002,886	12,471,647	12,471,647	90.27%

4.3 Lagrangian relaxation strategy and its properties[1]

In this section, consider all variables y_{ijkn} in the LIPR are binary. The resulting integer programming model LIPR is highly degenerate. Of all $MN^2 + M(M-1)N + \frac{M^2N^2}{2} + M + N$ constraints (4-2b)-(4-2f), only the last N knapsack constraints of (4-2e) and M equality constraints of (4-2f) have nonzero right-hand-side (RHS) values. The continuous relaxation $\overline{\text{LIPR}}$ of LIPR, will yield a lower bound $V\left(\overline{\text{LIPR}}\right)$ on the optimal value of LIPR, i.e., $V(\text{LIPR})$, and thus on $V(\text{GQAP})$. Here, $V(\bullet)$ is the optimal value of the problem. A stronger bound may be obtained by Lagrangian relaxation if the Lagrangian subproblems do not have the Integrity Property (Geoffrion, 1974).

Now, consider dualizing the symmetric constraints (4-2d). The Lagrangian subproblem for given Lagrangian multipliers $\boldsymbol{\lambda} = \left[\lambda_{ijkn}\right]_{i,j,k,n;k>i}$, is then

$\left[\text{LR}(\boldsymbol{\lambda})\right]$

$$\min \sum_{i=1}^{M}\sum_{j=1}^{N} b_{ij}x_{ij} + \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{M}\sum_{n=1}^{N} c_{ijkn}y_{ijkn} + \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{\substack{k=1\\k>i}}^{M}\sum_{n=1}^{N} \lambda_{ijkn}\left(y_{knij} - y_{ijkn}\right) \tag{4-5a}$$

$$\text{s.t.} \quad \sum_{k=1}^{M} s_{kn}y_{ijkn} \leq S_n x_{ij} \qquad i = 1,\dots,M;\ j,n = 1,\dots,N\ , \tag{4-5b}$$

$$\sum_{n=1}^{N} y_{ijkn} = x_{ij} \qquad i,k = 1,\dots,M;\ j = 1,\dots,N; k \neq i\ , \tag{4-5c}$$

[1] The material for this section was provided by Professor Monique Guignard of the University of Pennsylvania, see Guignard (2006).

$$\sum_{i=1}^{M} s_{ij} x_{ij} \le S_j \qquad j = 1,\ldots,N\,, \tag{4-5d}$$

$$\sum_{j=1}^{N} x_{ij} = 1 \qquad i = 1,\ldots,M\,, \tag{4-5e}$$

$$y_{ijkn} \in \{0,1\} \qquad i,k = 1,\ldots,M;\ j,n = 1,\ldots,N; k \ne i\,, \tag{4-5f}$$

$$x_{ij} \in \{0,1\} \qquad i = 1,\ldots,M;\ j = 1,\ldots,N\,. \tag{4-5g}$$

The Lagrangian dual is therefore an optimization problem over the multipliers λ

$[\mathrm{LD}]$

$$\max_{\lambda} \ V\left[\mathrm{LR}(\lambda)\right],$$

its optimal value is the Lagrangian bound on $V(\mathrm{LIPR}) = V(\mathrm{GQAP})$. Problem $\mathrm{LR}(\lambda)$ does not have the Integrity Property, so that the Lagrangian bound may be better than the linear programming (LP) bound $V\left(\overline{\mathrm{LIPR}}\right)$, in the sense of being closer to $V(\mathrm{LIPR})$.

One can now take advantage of the Integer Linearization Property (Geoffrion 1974, Geoffrion and McBride 1978, and Guignard 2003) to solve $\mathrm{LR}(\lambda)$ easily. Ignoring temporarily the constraints that are solely over x_{ij}, i.e., (4-5d), (4-5e) and (4-5g), one can see that the only true constraints remaining are (4-5b) and (4-5c), and the problem decomposes into one problem for each (i, j), in fact for each x_{ij}, which plays the role of a right-hand-side parameter, whose value can only be 0 or 1. If x_{ij} is 0, all associated y_{ijkn} are also 0. If x_{ij} is 1, let β_{ij} be the optimal value of the $(i, j)^{\text{th}}$ nonoverlapping subproblem $\mathrm{LR}(\lambda)^{ij}$,

$\left[\mathrm{LR}(\lambda)^{ij}\right]$

$$\min \sum_{\substack{k=1\\k<i}}^{M}\sum_{n=1}^{N}\left(c_{ijkn}+\lambda_{knij}\right)y_{ijkn}+\sum_{\substack{k=1\\k>i}}^{M}\sum_{n=1}^{N}\left(c_{ijkn}-\lambda_{ijkn}\right)y_{ijkn} \tag{4-6a}$$

$$\text{s.t.} \quad \sum_{k=1}^{M} s_{kn}y_{ijkn} \le S_n \qquad n=1,\dots,N\,, \tag{4-6b}$$

$$\sum_{n=1}^{N} y_{ijkn} = 1 \qquad k=1,\dots,M;\, k\ne i\,, \tag{4-6c}$$

$$y_{ijkn}\in\{0,1\} \qquad k=1,\dots,M;\, n=1,\dots,N;\, k\ne i\,, \tag{4-6d}$$

and then $\mathrm{LR}(\lambda)$ is equivalent to

$\left[\mathrm{GAP}(\lambda)\right]$

$$\min \sum_{i=1}^{M}\sum_{j=1}^{N}\left(b_{ij}+\beta_{ij}\right)x_{ij} \tag{4-7a}$$

$$\text{s.t.} \quad \sum_{i=1}^{M} s_{ij}x_{ij} \le S_j \qquad j=1,\dots,N\,, \tag{4-7b}$$

$$\sum_{j=1}^{N} x_{ij} = 1 \qquad i=1,\dots,M\,, \tag{4-7c}$$

$$x_{ij}\in\{0,1\} \qquad i=1,\dots,M;\, j=1,\dots,N\,. \tag{4-7d}$$

The above problems are actually the generalized assignment problem (GAP) which has a linear objective function. Thus, the solution of $\mathrm{LR}(\lambda)$ requires solving exactly $M\times N+1$ GAPs, and the final bound is likely to be tighter than the linear programming bound $V\left(\overline{\mathrm{LIPR}}\right)$. Given the current practical limits on GQAP problem sizes, it seems that any good mixed-integer programming (MIP) solver, for instance

CPLEX or EXCEL, can solve such generalized assignment problems to optimality with a fraction of a second. Even though this may exhibit erratic convergence, initial experiments could use subgradient optimization for updating the multipliers λ.

4.4 Conclusion

This chapter discusses the level-1 reformulation-linearization technique (RLT) formulation of the generalized quadratic assignment problem (GQAP), and one of its potential Lagrangian relaxations. A future step will be to compare numerically the linear programming (LP) and the Lagrangian relaxation (LR) bounds for some reasonably sized problems. If successful, this approach may lead to more understanding of the highly degenerate RLT models and the corresponding Lagrangian relaxations.

5. The Generalized Quadratic 3-dimensional Assignment Problem (GQ3AP)

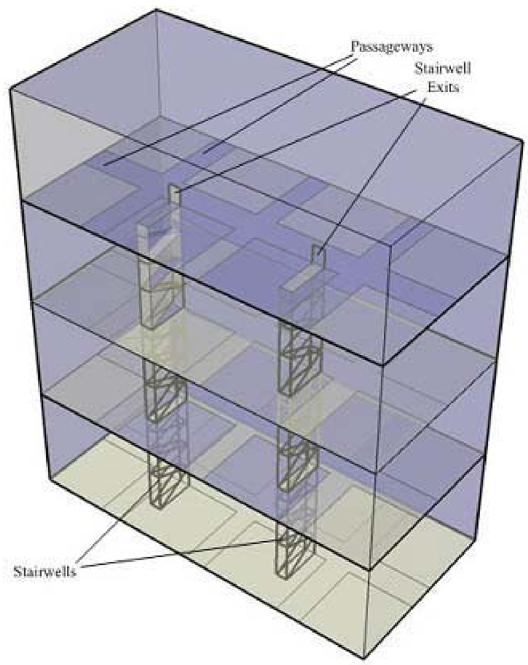
5.1 Motivation and problem background

5.1.1 Optimization of multi-story facility and evacuation design

Many facilities have certain functions or site conditions that require them to accommodate their occupants on multiple stories. Examples of these facilities are hospitals, office buildings, hotels, factories, warehouses, department stores, oil platforms, naval vessels, and so on. Multi-story facilities should be designed to provide an efficient and safe environment to meet the needs of the people and equipments that occupy it. The facility circulation is crucial to its infrastructure and layout, as it connects the interactive components of the facility and serves as the pathway for building services as well as for emergency egress. In building evacuation problems, the design of the stairwell widths, configuration of the landings, number of stairwells, their location, exits and egress paths are all critically linked. It is a travesty that people are often trampled in a push to get out of the narrow exists in sports stadia and in burning buildings.

The multi-story space assignment problem (MSAP) can be described as locating departments within a multi-story facility, so as to minimize the inter-department interaction cost and the cost to evacuate the building. As will be argued later, this problem is a generalization of both the generalized quadratic assignment problem (GQAP) and the quadratic 3-dimensional assignment problem (Q3AP). This chapter begins with the introduction of the component problems GQAP (more details given in Section 2.2) and Q3AP (more details given in Section 2.4).

Figure 5-1. Multi-story facility.



The GQAP covers a broad class of problems that involve the minimization of a total pair-wise interaction cost among M entities and where placement of these entities into N possible destinations is dependent upon existing resource capacities at each destination. These problems include finding the assignment of facilities to fixed locations given limited capacities at each possible location. The Lee and Ma (2004) version of the GQAP can be stated as an practical equipment location problem where it is desired to locate M equipments among N fixed locations, where for each pair of equipments (i,k) there is a certain flow of commodities $f(i,k)$ and for each pair of locations (j,n) there is a corresponding distance $d(j,n)$. The two-way interaction transportation cost between equipments i and k, given that i is assigned to location j and k is assigned to location n, is $f(i,k)\cdot d(j,n)+f(k,i)\cdot d(n,j)$. The objective is to find an assignment that minimizes the sum of all such transportation costs given the resource constraints.

Since the GQAP is relatively new, little work has been done to apply it to building design or facility location. Notable is the work by Meller and Bozer (1997) on the multi-floor facility layout problem (MFFLP). They defined the facility layout problem as finding a feasible, non-overlapping arrangement of departments (with given area requirements such as size and shape) within a facility (building) to minimize the interaction cost between departments. Interaction cost between two departments is stated as the flow times the distance between those departments. In general, departments have unequal area requirements, and some of them may be constrained a priori to certain locations in the building. They proceeded to solve the MFFLP in two stages. Stage one assigned departments to floors in an attempt to minimize the inter-floor handling cost, which was formulated as a GQAP. Stage two determined the layout of each floor based on the assignments of stage one. They argued that this approach not only comes up with near optimum solutions, but that it also has management appeal, since there are many management considerations that are not easy to quantify as costs.

Regarding the Q3AP, the interest in it stems from the fact that it is applied to problems where the objective is to minimize linear and quadratic costs associated with a pair of independent simultaneous one-to-one assignments. Such a problem arises in the design of wireless communication systems, wherein a digital message is repeated twice. During each of the repeats, the assignment of data words to transmitted symbols is modified. The Q3AP models the problem of optimizing the two assignments in such a way that the transmission errors are minimized. In the design of multi-story buildings, it is necessary to simultaneously assign facilities to locations and the same facilities to escape exits in the face of quadratic costs. Whereas, the assignments in the Q3AP are one-to-one, the assignments in multi-story building design tend to be many-to-one.

5.1.2 The multi-story space assignment problem (MSAP)[2]

This subsection begins with the necessary assumptions, definitions, and notation underlying the mathematical formulation of the MSAP.

Assumptions: (1) Distribution of the occupants on each floor is uniform and their travel time to and through the corridor and stairwell segments is a state dependent function. (2) The footprint of the facility is a polygon or can be closely approximated by a simple polygon. The polygon need not be convex by it is closed and bounded. (3) The travel distance metric is assumed to be rectilinear and the area per floor is given. The structural system either is pre-defined or else is to be designed after this analysis.

Definition: (1) Occupants are the people who occupy the facility and must be evacuated. There are a total of O occupants $O = \sum_{i=1}^{M} O_i$ where M is the total number of

[2] The material for this subsection was provided by Professor Peter M. Hahn of the University of Pennsylvania and Professor J. MacGregor Smith of the University of Massachusetts Amherst.

departments. (2) The circulation topology is the collection of corridors and stairwells on each floor. This is either known or has to be generated.

Given M departments, where for each department i the number of people O_i per department and the amount of space required for the department a_i is known, the primal issue here is to allocate the M departments to N floors, such that the sum of all one-way evacuation costs are minimized. Certain constraints must be met as no department may be split between different floors and the available space on each floor may not be exceeded. A secondary, but nevertheless important, issue is to encourage assignments that minimize the sum of all transportation costs, as for each pair of departments (i,k) a certain flow of units $f(i,k)$ 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. Both primary and secondary issues are dealt within the multi-story space assignment problem (MSAP), defined below.

Given the following parameters,

M number of departments needed to be placed,

N number of available floors,

P number of stairwells,

a_{ij} square feet of floor area needed by department i, if assigned to floor j,

A_j maximum usable floor area for 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_{jp} distance between floor j and stairwell exit p from building,

λ_{ip} arrival rate of persons in the evacuation from department i, if assigned to stairwell p,

μ_p service rate capacity of stairwell p,

x_{ij} binary variable, $x_{ij} = 1$ iff department i is assigned to floor j,

y_{ip} binary variable, $y_{ip} = 1$ iff department i is assigned to stairwell p.

The problem is to minimize the cost function

$$\min \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P}\sum_{k=1}^{M}\sum_{n=1}^{N}\left(f_{ik}d_{jn} + \lambda_{ip}e_{jp} \right) x_{ij}y_{ip}x_{kn} \tag{5-1a}$$

$$\text{s.t.} \quad \sum_{i=1}^{M} a_{ij}x_{ij} \le A_j \qquad j = 1,\ldots,N, \tag{5-1b}$$

$$\sum_{j=1}^{N} x_{ij} = 1 \qquad i = 1,\ldots,M, \tag{5-1c}$$

$$\sum_{i=1}^{M} \lambda_{ip}y_{ip} \le \mu_p \qquad p = 1,\ldots,P, \tag{5-1d}$$

$$\sum_{p=1}^{P} y_{ip} = 1 \qquad i = 1,\ldots,M, \tag{5-1e}$$

$$x_{ij} \in \{0,1\} \qquad i = 1,\ldots,M;\ j = 1,\ldots,N, \tag{5-1f}$$

$$y_{ip} \in \{0,1\} \qquad i = 1,\ldots,M;\ p = 1,\ldots,P. \tag{5-1g}$$

Equation (5-1b) makes sure that the capacity of each floor is not exceeded, (5-1c) makes sure that each department gets assigned to only one floor, (5-1d) makes sure that the capacity of each stairwell is not exceeded, and (5-1e) makes sure that each department is assigned to only one primary stairwell.

The MSAP, as given in equations (5-1), combines a Q3AP with a GQAP, thus making it an NP-hard problem. The methods developed for solving the Q3AP and the GQAP are an excellent basis for developing heuristics as well as exact methods for generating sub-optimal and optimal solutions.

5.1.3 Optimization of crossdock layout design

Many companies utilize complex distribution networks to transport various types of products to meet the customers demand. To be competitive, they try to shape the product moving in and out of the warehouse in the most cost-effective, efficient, and timely way. Crossdocking is a valuable logistics technique used in the retail and trucking industries to rapidly consolidate shipments from disparate sources and realize economics of scale in outbound transportation.

As stated in Bartholdi and Gue (2004), crossdocking essentially eliminates the costly handling and storage functions of a warehouse, while still allowing it to serve its receiving and shipping functions. The idea is to transfer shipments directly from incoming to outgoing trailers, without storage in between. Shipments typically spend less than 24 hours in a crossdock, sometimes less than an hour. With the process of moving shipments from the receiving dock (inbound) to the shipping dock (outbound), bypassing storage, crossdocking reduces inventory carrying cost, transportation cost, and other costs associated with material handling. Crossdocking is an important logistics strategy for companies in the retail, grocery, and other distribution industries, such as less-thantruckload (LTL) trucking firms, which seek to consolidate shipments to achieve transportation economics.

Research topics on the design and operational problems in crossdocking include the size of the crossdock, the number of doors, the width of the dock, shapes for different dock sizes, and assigning inbound and outbound trailer doors which is addressed here. Peck (1983), Tsui and Chang (1990, 1992), and Bartholdi and Gue (2000) have studied the optimization models to lay out crossdocks. A good layout places high-flow trailers near one another, but not so closes as to cause congestion.

5.1.4 The crossdock door assignment problem (CDAP)

The crossdock door assignment problem (CDAP) was proposed in Tsui and Chang (1990, 1992) with the objective of minimizing the weighted distance between incoming and outgoing trailers based on door-to-door distance.

Given the following parameters,

M number of origins,

N number of destinations,

f_{ik} the number of trips required by the material handling equipment to move items originating from i to the crossdock door where freight destined for k is being consolidated,

d_{jq} the distance between receiving door j and shipping door q,

x_{ij} binary variable, $x_{ij} = 1$ iff origin i is assigned to receiving door j,

y_{kq} binary variable, $y_{kq} = 1$ iff destination k is assigned to shipping door q.

The formulation of CDAP is then

$$\min \sum_{i=1}^{M}\sum_{j=1}^{M}\sum_{k=1}^{N}\sum_{q=1}^{N} f_{ik} d_{jq} x_{ij} y_{kq} \qquad (5\text{-}2a)$$

$$\text{s.t.} \quad \sum_{i=1}^{M} x_{ij} = 1 \qquad j = 1,\ldots,M\ , \qquad (5\text{-}2b)$$

$$\sum_{j=1}^{M} x_{ij} = 1 \qquad i = 1,\ldots,M\ , \qquad (5\text{-}2c)$$

$$\sum_{k=1}^{N} y_{kq} = 1 \qquad q = 1,\ldots,N\ , \qquad (5\text{-}2d)$$

$$\sum_{q=1}^{N} y_{kq} = 1 \qquad k = 1,\ldots,N\ , \qquad (5\text{-}2e)$$

$$x_{ij} \in \{0,1\} \qquad i, j = 1,\ldots,M\ , \qquad (5\text{-}2f)$$

$$y_{kq} \in \{0,1\} \qquad k, q = 1,\ldots,N\ . \qquad (5\text{-}2g)$$

Equation (5-2b) makes sure that each receiving door is assigned to only one origin, (5-2c) makes sure that each origin gets assigned to only one receiving door, (5-2d) makes sure that each shipping door is assigned to only one destination, and (5-2e) makes sure that each destination is assigned to only one shipping door.

Peck (1983) gave similar models but also considered different types of freight and material handling systems. Bartholdi and Gue (2000) described models that guide a local search routine in assigning destination trailers to terminal doors so as to minimize the total labor cost. Their layout models balanced the cost of moving freight from incoming trailer to outgoing trailers with the cost of delays due to different types of congestion.

One can view the CDAP as a Q3AP (P.M. Hahn, personal communication, February 13, 2006) if the number of receiving doors is equal to the number of shipping doors, since the Q3AP minimizes the cost over two permutation matrices of the same size. In that case, the Q3AP six-dimensional cost matrix can be rewritten as $c_{ijpknq} = f_{ik} d_{jq}$, where $\boldsymbol{F} = [f_{ik}]$ and $\boldsymbol{D} = [d_{jq}]$ are $N \times N$ matrices. To test this hypothesis, I adapted the exact Q3AP solution algorithm to solve a size $M = N = 10$ CDAP test instance. This test instance was solved exactly in just 2,197.7 seconds on a Sun Ultra 10 workstation with a single 360 MHz CPU and required the evaluation of just 4,162 nodes. It is interesting to notice that a Q3AP test problem of the same size (where all six-dimensional cost coefficients are arbitrary) took about four times longer to solve on the same workstation and required the evaluation of about 2.5 times of the nodes. To adjust the Q3AP to solve a CDAP with M not equal to N, the product $f_{ik} d_{jq}$ is manipulated in a way that forces the solution of the Q3AP to be a solution of a CDAP whose $\boldsymbol{X}$ and $\boldsymbol{Y}$ matrices are of different sizes. To accomplish this, it is necessary to set

$$f_{ik} = \infty \qquad \text{if } (i > M, M < N) \text{ or } (k > N, N < M), \tag{5-3a}$$

$$d_{jq} = \infty \qquad \text{if } (j > M, M < N) \text{ or } (q > N, N < M), \tag{5-3b}$$

$$f_{ik} d_{jq} = 0 \qquad \text{if } \begin{array}{l} (i > M, j > M, M < N) \\ \text{or } (k > N, q > N, N < M) \end{array}. \tag{5-3c}$$

It is easy to show that the constraints for the Q3AP are the same as those for the CDAP. Thus, it has been shown that the CDAP is a special case of the Q3AP. One must be careful that, when using the Q3AP to solve a CDAP, one must take the transpose of the feasible Q3AP solutions in order to get the corresponding CDAP solutions.

Now consider a more general version of the CDAP. First, here are a few observations about the limitations of formulations (5-2) and (5-3): matrices $\boldsymbol{X}$ and $\boldsymbol{Y}$ are necessarily square. But the flow matrix $\boldsymbol{F}$ and the distance matrix $\boldsymbol{D}$ are not. The fact that $\boldsymbol{X}$ and $\boldsymbol{Y}$ are square is limiting. There would not be very many practical situations where each origin would have its own receiving door and each destination have its own shipping door. Thus, a more general form with additional parameters has been defined (P.M. Hahn, personal communication, February 23, 2006),

J number of receiving doors,

Q number of shipping doors.

The formulation becomes

$$\min \sum_{i=1}^{M}\sum_{j=1}^{J}\sum_{k=1}^{N}\sum_{q=1}^{Q} f_{ik} d_{jq} x_{ij} y_{kq} \tag{5-4a}$$

$$\text{s.t.} \quad \sum_{i=1}^{M} s_i x_{ij} \le S_j \qquad j = 1,\dots,J\,, \tag{5-4b}$$

$$\sum_{j=1}^{J} x_{ij} = 1 \qquad i = 1,\dots,M\,, \tag{5-4c}$$

$$\sum_{k=1}^{N} t_k y_{kq} \le T_q \qquad q = 1,\dots,Q\,, \tag{5-4d}$$

$$\sum_{q=1}^{Q} y_{kq} = 1 \qquad k = 1,\dots,N\,, \tag{5-4e}$$

$$x_{ij} \in \{0,1\} \qquad i = 1,\dots,M;\ j = 1,\dots,J\,, \tag{5-4f}$$

$$y_{kq} \in \{0,1\} \qquad k = 1,\dots,N;\ q = 1,\dots,Q\,. \tag{5-4g}$$

Equation (5-4b) makes sure that the capacity of each receiving door is not exceeded, and (5-4c) makes sure that each origin gets assigned to only one receiving door. While (5-4d) makes sure that the capacity of each shipping door is not exceeded, and (5-4e) makes sure that each destination is assigned to only one shipping door.

5.2 Problem definition for the GQ3AP

Professor Hahn also defined a new class of assignment problem, the generalized quadratic 3-dimensional assignment problem (GQ3AP), which combines the Q3AP with the GQAP and which allows to deal with the MSAP as well as the CDAP (Guignard et al. 2006). The GQ3AP is given by minimizing the objective function

$$\min \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P}\sum_{k=1}^{M}\sum_{n=1}^{N}\sum_{q=1}^{P} c_{ijpknq} u_{ij} u_{kn} w_{ip} w_{kq} \tag{5-5a}$$

subject to the following constraints on $\boldsymbol{u}$

$$\sum_{i=1}^{M} s_{ij} u_{ij} \le S_j \qquad j = 1, 2, ..., N , \tag{5-5b}$$

$$\sum_{j=1}^{N} u_{ij} = 1 \qquad i = 1, 2, ..., M , \tag{5-5c}$$

$$u_{ij} \in \{0,1\} \qquad i = 1, 2, ..., M ; j = 1, 2, ..., N , \tag{5-5d}$$

where S_j is the available resource at location j and s_{ij} is the resource requirement of equipment i, and subject to the following constraints on $\boldsymbol{w}$

$$\sum_{i=1}^{M} t_{ip} w_{ip} \le T_p \qquad p = 1, 2, ..., P , \tag{5-5e}$$

$$\sum_{p=1}^{P} w_{ip} = 1 \qquad i = 1, 2, ..., M , \tag{5-5f}$$

$$w_{ip} \in \{0,1\} \qquad i = 1, 2, ..., M ; p = 1, 2, ..., P , \tag{5-5g}$$

where T_p is the available resource at location p and t_{ip} is the resource requirement of equipment i.

Now, if the six-dimensional cost matrix $\boldsymbol{C}$ can be written as

$$c_{ijpknq} = f_{ik} d_{jn} + \lambda_{ip} e_{jp}\,, \tag{5-6}$$

where all the notations are defined just as before in the MSAP.

Then the GQ3AP objective function becomes

$$\min \ \sum_{i=1}^{M} \sum_{j=1}^{N} \sum_{p=1}^{P} \sum_{k=1}^{M} \sum_{n=1}^{N} \sum_{q=1}^{P} \left(f_{ik} d_{jn} + \lambda_{ip} e_{jp} \right) u_{ij} u_{kn} w_{ip} w_{kq}\ . \tag{5-7a}$$

But, the function being minimized can be simplified as

$$\begin{aligned} & \sum_{i=1}^{M} \sum_{j=1}^{N} \sum_{p=1}^{P} \sum_{k=1}^{M} \sum_{n=1}^{N} \sum_{q=1}^{P} \left(f_{ik} d_{jn} + \lambda_{ip} e_{jp} \right) u_{ij} u_{kn} w_{ip} w_{kq} \\ & = \sum_{i=1}^{M} \sum_{j=1}^{N} \sum_{p=1}^{P} \sum_{k=1}^{M} \sum_{n=1}^{N} \left(f_{ik} d_{jn} + \lambda_{ip} e_{jp} \right) u_{ij} u_{kn} w_{ip} \sum_{q=1}^{P} w_{kq} \\ & = \sum_{i=1}^{M} \sum_{j=1}^{N} \sum_{p=1}^{P} \sum_{k=1}^{M} \sum_{n=1}^{N} \left(f_{ik} d_{jn} + \lambda_{ip} e_{jp} \right) u_{ij} w_{ip} u_{kn} \end{aligned} \tag{5-7b}$$

which is the same objective function as for the MSAP. And the constraints for the GQ3AP are essentially the same as for the MSAP.

Now consider the case of the CDAP. If one rewrites the six-dimensional cost matrix $\boldsymbol{C}$ as

$$c_{ijpknq} = f_{ik} d_{jq}\,, \tag{5-8}$$

where $\boldsymbol{F} = \left[f_{ik} \right],\ i,k = 1,\ldots,M$ and $\boldsymbol{D} = \left[d_{jq} \right],\ j = 1,\ldots,J; q = 1,\ldots,Q$, then it is possible to recast the CDAP as a GQ3AP, where $\boldsymbol{D}$ is the receiving/shipping distance in crossdocking facility and $\boldsymbol{F}$ is the number of moving trips of goods, just as before. The difference in (5-8) is that the number of origins is equal to the number of destinations (a perhaps serious limitation).

Then the objective is to minimize the function below.

$$
\begin{aligned}
&\sum_{i=1}^{M}\sum_{j=1}^{J}\sum_{p=1}^{Q}\sum_{k=1}^{M}\sum_{n=1}^{J}\sum_{q=1}^{Q} f_{ik}d_{jq}u_{ij}u_{kn}w_{ip}w_{kq} \\
&=\sum_{i=1}^{M}\sum_{j=1}^{J}\sum_{k=1}^{M}\sum_{n=1}^{J}\sum_{q=1}^{Q} f_{ik}d_{jq}u_{ij}u_{kn}w_{kq}\sum_{p=1}^{Q} w_{ip} \\
&=\sum_{i=1}^{M}\sum_{j=1}^{J}\sum_{k=1}^{M}\sum_{q=1}^{Q} f_{ik}d_{jq}u_{ij}w_{kq}\sum_{n=1}^{J} u_{kn} \\
&=\sum_{i=1}^{M}\sum_{j=1}^{J}\sum_{k=1}^{M}\sum_{q=1}^{Q} f_{ik}d_{jq}u_{ij}w_{kq}
\end{aligned}
\tag{5-9}
$$

Similarity between (5-9) and (5-4a) is striking, as the only difference being that in (5-9) the upper ranges of i and k are equal.

To permit the GQ3AP to solve a CDAP where the number of origins is not equal to the number of destinations, it is necessary to place restrictions on the product $f_{ik}d_{jq}$ in a way that forces a solution of the GQ3AP to be a solution of a CDAP. Namely, one must set

$$f_{ik}=0 \qquad \text{if } \left(i>M, M<N\right) \text{ or } \left(k>N, N<M\right). \tag{5-10}$$

Finally, it is necessary to assure that the constraints for this modified GQ3AP are indeed the same as for the CDAP. Thus, it has been shown that the CDAP is a special case of the GQ3AP.

5.3 Reformulation-linearization technique (RLT) on the GQ3AP

5.3.1 Two equivalent GQ3AP formulations

If one takes out the linear terms with coefficients b_{ijp} that are included in the (5-5a), then the formulation of GQ3AP is

$$\min \left\{ \begin{array}{l} \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P} b_{ijp} u_{ij} w_{ip} + \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P}\sum_{\substack{k=1\\k\neq i}}^{M}\sum_{n=1}^{N}\sum_{q=1}^{P} c_{ijpknq} u_{ij} u_{kn} w_{ip} w_{kq} \\ : \boldsymbol{u} \in \boldsymbol{U}, \boldsymbol{v} \in \boldsymbol{W}; \boldsymbol{u}, \boldsymbol{v} \text{ binary} \end{array} \right\}, \tag{5-11a}$$

$$\text{where } \boldsymbol{u} \in \boldsymbol{U} \equiv \left\{ \boldsymbol{u} \geq \boldsymbol{0} : \sum_{i=1}^{M} s_{ij} u_{ij} \leq S_j, \forall \left(j = 1,...,N \right); \sum_{j=1}^{N} u_{ij} = 1, \forall \left(i = 1,...,M \right) \right\}, \tag{5-11b}$$

$$\text{and } \boldsymbol{w} \in \boldsymbol{W} \equiv \left\{ \boldsymbol{w} \geq \boldsymbol{0} : \sum_{i=1}^{M} t_{ip} w_{ip} \leq T_p, \forall \left(p = 1,...,P \right); \sum_{p=1}^{P} w_{ip} = 1, \forall \left(i = 1,...,M \right) \right\}. \tag{5-11c}$$

An equivalent formulation of the GQ3AP can be stated as

$$\min \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P} b_{ijp} x_{ijp} + \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P}\sum_{\substack{k=1\\k\neq i}}^{M}\sum_{n=1}^{N}\sum_{q=1}^{P} c_{ijpknq} x_{ijp} x_{knq} \tag{5-12a}$$

$$\text{s.t.} \quad \sum_{j=1}^{N}\sum_{p=1}^{P} x_{ijp} = 1 \qquad i = 1,...,M, \tag{5-12b}$$

$$\sum_{i=1}^{M}\sum_{p=1}^{P} s_{ij} x_{ijp} \leq S_j \qquad j = 1,...,N, \tag{5-12c}$$

$$\sum_{i=1}^{M}\sum_{j=1}^{N} t_{ip} x_{ijp} \leq T_p \qquad p = 1,...,P, \tag{5-12d}$$

$$x_{ijp} \in \{0,1\} \qquad i = 1,...,M; j = 1,...,N; p = 1,...,P. \tag{5-12e}$$

The GQ3AP in (5-12) can be constructed from (5-11) by letting $u_{ij} w_{ip} = x_{ijp}$ and $u_{kn} w_{kq} = x_{knq}$. The equivalence of the two objective functions is trivially true, so are the constraints on binary variables. Constraints (5-12b) come from multiplying the second equality constraints in (5-11b) with the second equality constraints in (5-11c) as $\sum_{j=1}^{N} u_{ij} \sum_{p=1}^{P} w_{ip} = \sum_{j=1}^{N}\sum_{p=1}^{P} u_{ij} w_{ip} = \sum_{j=1}^{N}\sum_{p=1}^{P} x_{ijp} = 1$. Constraints (5-12c) follow from multiplying the first inequality constraints in (5-11b) with the second equality constraints in (5-11c) as $\sum_{i=1}^{M} s_{ij}u_{ij}\sum_{p=1}^{P} w_{ip} = \sum_{i=1}^{M}\sum_{p=1}^{P} s_{ij}u_{ij}w_{ip} = \sum_{i=1}^{M}\sum_{p=1}^{P} s_{ij}x_{ijp} \le S_j$. Constraints (5-12d) result from multiplying the first inequality constraints in (5-11c) with the second equality constraints in (5-11b) as $\sum_{i=1}^{M} t_{ip}w_{ip}\sum_{j=1}^{N} u_{ij} = \sum_{i=1}^{M}\sum_{j=1}^{N} t_{ip}u_{ij}w_{ip} = \sum_{i=1}^{M}\sum_{j=1}^{N} t_{ip}x_{ijp} \le T_p$.

Conversely, given the GQ3AP model in (5-12), by letting $x_{ijp} = u_{ij}w_{ip}$, $\sum_{j=1}^{N}\sum_{p=1}^{P} x_{ijp}$ becomes $\sum_{j=1}^{N}\sum_{p=1}^{P} u_{ij}w_{ip} = \sum_{j=1}^{N} u_{ij}\sum_{p=1}^{P} w_{ip} = \sum_{j=1}^{N} u_{ij} = \sum_{p=1}^{P} w_{ip} = 1$, $\sum_{i=1}^{M}\sum_{p=1}^{P} s_{ij}x_{ijp}$ becomes $\sum_{i=1}^{M}\sum_{p=1}^{P} s_{ij}u_{ij}w_{ip} = \sum_{i=1}^{M} s_{ij}u_{ij}\sum_{p=1}^{P} w_{ip} = \sum_{i=1}^{M} s_{ij}u_{ij} \le S_j$, and $\sum_{i=1}^{M}\sum_{j=1}^{N} t_{ip}x_{ijp}$ becomes $\sum_{i=1}^{M}\sum_{j=1}^{N} t_{ip}u_{ij}w_{ip} = \sum_{i=1}^{M} t_{ip}w_{ip}\sum_{j=1}^{N} u_{ij} = \sum_{i=1}^{M} t_{ip}w_{ip} \le T_p$. Therefore, the model of (5-11) can also be derived from the formulation (5-12).

5.3.2 The level-1 RLT formulation of the GQ3AP

The next step is to transform the original model of the GQ3AP into an equivalent linearized mixed-integer programming model G3RLT, similar to those successfully used on Q3AP and GQAP. First, multiply each of equality constraints (5-12b) and inequality constraints (5-12c)-(5-12d), written as $\sum_{n=1}^{N}\sum_{q=1}^{P} x_{knq} = 1$, $\sum_{k=1}^{M}\sum_{q=1}^{P} s_{kn}x_{knq} \le S_n$ and $\sum_{k=1}^{M}\sum_{n=1}^{N} t_{kn}x_{knq} \le T_q$, by each of N^3 binary variables x_{ijp}. Append all these new restrictions. Express the resulting product in the order $x_{ijp}x_{knq}$. Next, explicitly include the trivial restrictions that $x_{ijp}x_{knq} = x_{knq}x_{ijp} \quad \forall(i, j, p, k, n, q), k > i$. Then, linearize every occurrence of each product $x_{ijp}x_{knq}$ with a single nonnegative continuous variable $y_{ijpknq} = x_{ijp}x_{knq}$. Finally, consider the solution structure of the GQ3AP, which indicates that $y_{ijpinq} = 0 \ \forall(i, j, p, n, q; n \neq j \text{ or } q \neq p)$ and $y_{ijpijp} = x_{ijp} \ \forall(i, j, p)$. The level-1 RLT formulation G3RLT then results in.

[G3RLT]

$$\min \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P} b_{ijp}x_{ijp} + \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P}\sum_{\substack{k=1\\k\neq i}}^{M}\sum_{n=1}^{N}\sum_{q=1}^{P} c_{ijpknq}y_{ijpknq} \tag{5-13a}$$

$$\text{s.t.} \quad \sum_{n=1}^{N}\sum_{q=1}^{P} y_{ijpknq} = x_{ijp} \qquad i,k = 1,\dots,M;\ j = 1,\dots,N;\ p = 1,\dots,P; k \neq i, \tag{5-13b}$$

$$\sum_{k=1}^{M}\sum_{q=1}^{P} s_{kn}y_{ijpknq} \leq S_n x_{ijp} \quad i = 1,\dots,M;\ j,n = 1,\dots,N;\ p = 1,\dots,P, \tag{5-13c}$$

$$\sum_{k=1}^{M}\sum_{n=1}^{N} t_{kq}y_{ijpknq} \leq T_q x_{ijp} \quad i = 1,\dots,M;\ j = 1,\dots,N;\ p,q = 1,\dots,P, \tag{5-13d}$$

$$y_{ijpknq} = y_{knqijp} \qquad i,k = 1,\dots,M;\ j,n = 1,\dots,N;\ p,q = 1,\dots,P; k > i, \tag{5-13e}$$

$$\sum_{j=1}^{N}\sum_{p=1}^{P} x_{ijp} = 1 \qquad i = 1,\dots,M, \tag{5-13f}$$

$$\sum_{i=1}^{M}\sum_{p=1}^{P} s_{ij}x_{ijp} \leq S_j \qquad j = 1,\dots,N, \tag{5-13g}$$

$$\sum_{i=1}^{M}\sum_{j=1}^{N} t_{ip}x_{ijp} \leq T_p \qquad p = 1,\dots,P, \tag{5-13h}$$

$$y_{ijpknq} \geq 0 \qquad i,k = 1,\dots,M;\ j,n = 1,\dots,N;\ p,q = 1,\dots,P; k \neq i, \tag{5-13i}$$

$$x_{ijp} \in \{0,1\} \qquad i = 1,\dots,M;\ j = 1,\dots,N;\ p = 1,\dots,P. \tag{5-13j}$$

Problem GQ3AP and G3RLT are equivalent in the following sense. Given any feasible solution $\boldsymbol{x}$ to the GQ3AP, there exists a $\boldsymbol{y}$ such that $(\boldsymbol{x},\boldsymbol{y})$ is feasible to the G3RLT with the same objective value. Conversely, for any feasible solution $(\boldsymbol{x},\boldsymbol{y})$ to the G3RLT, the corresponding $\boldsymbol{x}$ is feasible to the GQ3AP with the same objective value.

PROOF.

For a given $\boldsymbol{x}$ satisfying the constraints of the GQ3AP given in (5-12), compute $y_{ijpknq} = x_{ijp}x_{knq}\ \forall(i,j,p,k,n,q)$. It is trivially true that $(\boldsymbol{x},\boldsymbol{y})$ is a feasible solution to the G3RLT and the value of objective functions of GQ3AP and G3RLT matches as long as (5-12b)-(5-12e) are part of the G3RLT.

In order to prove the other direction of equivalency, it is necessary to show that a feasible solution $(\boldsymbol{x},\boldsymbol{y})$ to the G3RLT satisfies $y_{ijpknq} = x_{ijp}x_{knq}\ \forall(i,j,p,k,n,q),\ k \neq i$, which guarantees that the objective values of GQ3AP and G3RLT are equal. In fact, it is sufficient to show that if $(\boldsymbol{x},\boldsymbol{y})$ is feasible to the G3RLT then y_{ijpknq} is 0 unless x_{ijp} and x_{knq} are both equal to 1 in which case y_{ijpknq} is also 1.

If $x_{ijp} = 0$ for given i,j and p, then (5-13b)-(5-13d) and (5-13i) together imply that $y_{ijphst} = 0\ \forall(h,s,t), h \neq i$.

If $x_{knq} = 0$ for given k,n and q, then (5-13b)-(5-13e) and (5-13i) together imply that $y_{luwknq} = 0\ \forall(l,u,w), l \neq k$.

Therefore, if $x_{ijp} = x_{knq} = 0$ for given $i,j,p,k \neq i,n$ and q, then $y_{ijpknq} = 0$.

Besides, if $x_{ijp} = x_{knq} = 1$ for given $i, j, p, k \neq i, n$ and q, then it must be shown that $y_{ijpknq} = 1$.

From (5-13f) $\sum_{u=1}^{N}\sum_{w=1}^{P} x_{iuw} = 1 \ \forall(i)$, which implies that if $x_{ijp} = 1$, then

$$x_{iuw} = 0 \ \forall(u,w), u \neq j, w \neq p; \tag{5-14a}$$

$$x_{ijw} = 0 \ \forall(w), w \neq p; \tag{5-14b}$$

and $$x_{iup} = 0 \ \forall(u), u \neq j. \tag{5-14c}$$

Thus, similarly as above, (5-13b)-(5-13d) and (5-13i) together imply that

$$y_{iuwhst} = 0 \ \forall(u,w,h,s,t), u \neq j, w \neq p, h \neq i; \tag{5-15a}$$

$$y_{ijwhst} = 0 \ \forall(w,h,s,t), w \neq p, h \neq i; \tag{5-15b}$$

and $$y_{iuphst} = 0 \ \forall(u,h,s,t), u \neq j, h \neq i. \tag{5-15c}$$

So, $$y_{iuwknq} = 0 \ \forall(u,w), u \neq j, w \neq p, k \neq i; \tag{5-16a}$$

$$y_{ijwknq} = 0 \ \forall(w), w \neq p, k \neq i; \tag{5-16b}$$

and $$y_{iupknq} = 0 \ \forall(u), u \neq j, k \neq i. \tag{5-16c}$$

By (5-13b) and (5-13e), it turns out that

$$\sum_{u=1}^{N}\sum_{w=1}^{P} y_{iuwknq} = x_{knq} \ \ \forall(i), i \neq k. \tag{5-17}$$

Therefore,

$$\sum_{u=1}^{N}\sum_{w=1}^{P} y_{iuwknq} = \sum_{\substack{u=1 \\ u \neq j}}^{N}\sum_{\substack{w=1 \\ w \neq p}}^{P} y_{iuwknq} + \sum_{\substack{w=1 \\ w \neq p}}^{P} y_{ijwknq} + \sum_{\substack{u=1 \\ u \neq j}}^{N} y_{iupknq} + y_{ijpknq} = y_{ijpknq} = x_{knq}, \tag{5-18}$$

which implies that if $x_{knq} = 1$ then $y_{ijpknq} = 1$. This completes the proof. □

The mathematical structure of G3RLT (5-13) can be readily exploited via Lagrangian duality. An important step of establishing the Lagrangian dual of a combinatorial optimization problem is to partition the constraints into two sets: the constraints to be dualized into objective function using suitable Lagrangian multipliers and the remaining constraints which restrict the relatively easy subproblems. The development of the Lagrangian dual for the GQ3AP is covered in Section 5.4.

5.3.3 Identifying the GQ3AP solution matrix and cost matrix

Now consider the $M^2 \times N^2 \times P^2$ matrix $\boldsymbol{Y}$ as a Kronecker product of the 3-dimensional $M \times N \times P$ assignment solution matrix $\boldsymbol{X}$ with itself, since mathematically the Kronecker product is an operation on two matrices of arbitrary size resulting in a block matrix. That is,

$$
\begin{aligned}
\boldsymbol{Y} &= \boldsymbol{X} \otimes \boldsymbol{X} \\
&= \left\{ \begin{array}{c}
\begin{bmatrix} x_{111}\boldsymbol{X} & \cdots & x_{11P}\boldsymbol{X} \\ \vdots & \ddots & \vdots \\ x_{1N1}\boldsymbol{X} & \cdots & x_{1NP}\boldsymbol{X} \end{bmatrix}_{i=1} \\
\vdots \\
\begin{bmatrix} x_{M11}\boldsymbol{X} & \cdots & x_{M1P}\boldsymbol{X} \\ \vdots & \ddots & \vdots \\ x_{MN1}\boldsymbol{X} & \cdots & x_{MNP}\boldsymbol{X} \end{bmatrix}_{i=M}
\end{array} \right\} \\
&= \left[y_{ijpknq} \right]_{M^2 \times N^2 \times P^2} = \left[\boldsymbol{Y}_{[ijp]} \right]_{M \times N \times P} \\
&= \left\{ \begin{array}{c}
\begin{bmatrix} \boldsymbol{Y}_{[111]} & \cdots & \boldsymbol{Y}_{[11P]} \\ \vdots & \ddots & \vdots \\ \boldsymbol{Y}_{[1N1]} & \cdots & \boldsymbol{Y}_{[1NP]} \end{bmatrix}_{i=1} \\
\vdots \\
\begin{bmatrix} \boldsymbol{Y}_{[M11]} & \cdots & \boldsymbol{Y}_{[M1P]} \\ \vdots & \ddots & \vdots \\ \boldsymbol{Y}_{[MN1]} & \cdots & \boldsymbol{Y}_{[MNP]} \end{bmatrix}_{i=M}
\end{array} \right\}
\end{aligned} \tag{5-19a}
$$

where $y_{ijpknq} = x_{ijp}x_{knq} = x_{knq}x_{ijp} = y_{knqijp}$

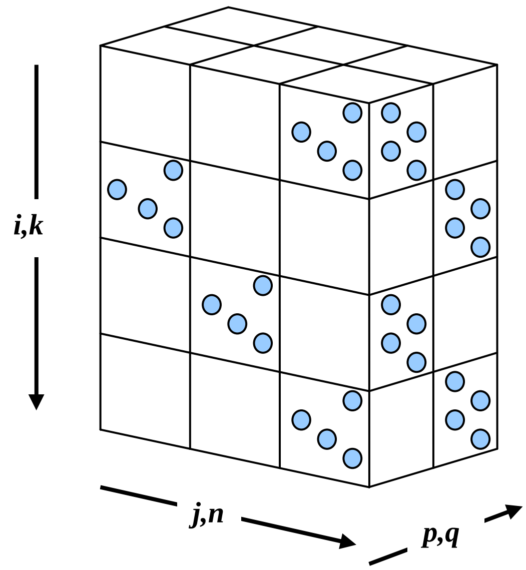
$$\forall\left(i,k = 1,\ldots,M;\, j,n = 1,\ldots,N;\, p,q = 1,\ldots,P\right), k \neq i\,, \tag{5-19b}$$

$$y_{ijpijp} = x_{ijp} \qquad \forall\left(i = 1,\ldots,M;\, j = 1,\ldots,N;\, p = 1,\ldots,P\right), \tag{5-19c}$$

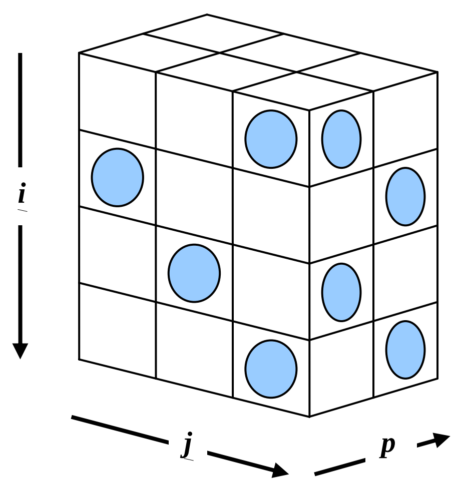
$$y_{ijpinq} = 0 \qquad \forall\left(i = 1,\ldots,M;\, j,n = 1,\ldots,N;\, p,q = 1,\ldots,P\right), n \neq j \text{ or } q \neq p\,. \tag{5-19d}$$

As indicated in (5-19a), every submatrix $\boldsymbol{Y}_{[ijp]}$ of $\boldsymbol{Y}$ is an $M \times N \times P$ 3-dimensional matrix. Here, (5-19b)-(5-19d) are indicated by the solution structure $\boldsymbol{X}$ of the GQ3AP and the definition of $\boldsymbol{Y}$. Figure 5-2 presents an example of $\boldsymbol{Y}$ matrix and its corresponding $\boldsymbol{X}$ matrix for the GQ3AP of $M = 4$, $N = 3$ and $P = 2$. The matrix $\boldsymbol{Y}$ is a partitioned matrix whose elements are composed of the Null matrix (all elements being zero) and matrix $\boldsymbol{X}$, and furthermore $\boldsymbol{Y}$ exhibits a gross pattern identical to that of the matrix $\boldsymbol{X}$.

Figure 5-2. Example of GQ3AP solution matrix $\boldsymbol{Y}$ and its $\boldsymbol{X}$ matrix.



Matrix $\boldsymbol{Y}$



Matrix $\boldsymbol{X}$

Equation (5-19b) gives the complementary pairs in the GQ3AP solution matrix $\boldsymbol{Y}$, which is very important to the effectiveness of the dual-ascent procedure for the GQ3AP lower bound calculation. It says that if an element y_{ijpknq} is involved in an assignment (i.e., equal to 1) then it has a complementary element y_{knqijp} that is also equal to 1. These complementary pairs have an interesting property; they are always in two different 3-dimensional submatrices that never occupy the same $k-\text{plane}$, here the $k-\text{plane}$ is the plane along (n,q) in the submatrix $\boldsymbol{Y}_{[ijp]}$. The equality between the pair creates a valuable communication between submatrices that can be exploited in calculating the GQ3AP lower bounds.

Certain elements of $\boldsymbol{Y}$ as indicated by (5-19d) are always zero. These elements are referred to as “disallowed elements”. Elements described in (5-19c) are termed “linear-cost elements”. Observe that if there are any 1’s in a 3-dimensional submatrix $\boldsymbol{Y}_{[ijp]}$ its linear-cost element is always unity and vice versa. Observe, too, that one and only one unity element may exist in each $i-\text{plane}$, which is the plane along (j,p) in matrix $\boldsymbol{Y}$. These constraints follow directly from the definition of $\boldsymbol{Y}$ as the product of two 3-dimensional assignment matrices. A linear-cost is special in another way; it has no complementary element because $y_{ijpijp} = x_{ijp}x_{ijp}$.

Now, suppose one arranges the $M^2 \times N^2 \times P^2$ nonnegative cost coefficient $c_{ijpknq}\ (i,k=1,...,M;\ j,n=1,...,N;\ p,q=1,...,P)$ in an $M^2 \times N^2 \times P^2$ matrix $\boldsymbol{C}$, similar to matrix $\boldsymbol{Y}$ and indexed in precisely the same way. That is the following.

$$\boldsymbol{C} = \left[c_{ijpknq}\right]_{M^2 \times N^2 \times P^2} = \left[\boldsymbol{C}_{[ijp]}\right]_{M \times N \times P}$$

$$= \left\{ \begin{array}{c} \begin{bmatrix} \boldsymbol{C}_{[111]} & \cdots & \boldsymbol{C}_{[11P]} \\ \vdots & \ddots & \vdots \\ \boldsymbol{C}_{[1N1]} & \cdots & \boldsymbol{C}_{[1NP]} \end{bmatrix}_{i=1} \\ \vdots \\ \begin{bmatrix} \boldsymbol{C}_{[M11]} & \cdots & \boldsymbol{C}_{[M1P]} \\ \vdots & \ddots & \vdots \\ \boldsymbol{C}_{[MN1]} & \cdots & \boldsymbol{C}_{[MNP]} \end{bmatrix}_{i=M} \end{array} \right\} \qquad (5\text{-}20)$$

Since $\boldsymbol{Y}$ comprises $M \times N \times P$ copies of 3-dimensioal assignment solution matrix $\boldsymbol{X}$, it is easy to see that the 3-dimensional submatrices $\boldsymbol{C}_{[ijp]}$ of $\boldsymbol{C}$ would have to each meet the constraints on $\boldsymbol{X}$. Thus, a 3-dimensional submatrix of $\boldsymbol{C}$ may be thought of as a generalized 3-dimensional assignment subproblem of the GQ3AP, similar to the concept of a submatrix of the QAP cost matrix can be treated as a LAP (Hahn and Grant, 1998). The generalized 3-dimensional assignment problem (G3AP) is an assignment problem over the same constraints (5-12b)-(5-12e) on $\boldsymbol{X}$ with only linear cost considered in the objective function.

The existence of complementary pairs opens the door to considerably flexibility in the $\boldsymbol{C}$ matrix. If one element of a complementary pair is involved in an assignment, so is the other. Therefore, for each complementary pair, cost can be shifted at will between the two elements. For instance, the cost of both elements can be added and placed at the first element while the cost of the second becomes zero or vice versa.

It can be shown that certain operations may be performed on matrix $\boldsymbol{C} = \left[c_{ijpknq}\right]$ that will change the cost $Z(\boldsymbol{X})$ of assignment $\boldsymbol{X}$ in such a way that all assignment costs are shifted by an identical amount, thus preserving their order with respect to cost. These operations are divided into two classes (see Grant (1989) for proofs):

Class 1: Subtraction of a constant from all non-disallowed elements from a $k-$plane of a 3-dimensional submatrix $\boldsymbol{C}_{[ijp]}$ and the corresponding addition of this constant to either another $k-$plane of the submatrix or to the linear cost element b_{ijp} of the submatrix. The term "disallowed" was defined in (5-19d).

Class 2: Addition or subtraction of a constant to all non-disallowed elements of any $i-$plane in matrix $\boldsymbol{C}$.

Class 1 operations maintain the cost of all assignments, but permit redistribution of element costs within a given 3-dimensional submatrix. This follows because any submatrix involved in a 3-dimensional assignment must itself have the form of a 3-dimensional assignment matrix. Consequently, the transfer of cost from one submatrix plane to another will maintain the cost of all assignments that pass through the 3-dimensional submatrix. Furthermore, since the linear cost of an involved 3-dimensional submatrix must also be involved in the 3-dimensional assignment, the transfer of cost from a submatrix plane to the linear cost, and vice versa, will also leave the cost of all assignments that intersect the 3-dimensional submatrix unchanged.

In contrast, Class 2 operations work on the $M^2 \times N^2 \times P^2$ matrix level, and change the cost of ALL 3-dimensional assignments by the amount added to or subtracted from the $i-$plane of the matrix. This is true because one and only one cost element in an $i-$plane of $\boldsymbol{C}$ can be involved in the overall cost of a 3-dimensional assignment.

5.4 A dual-ascent procedure lower bound calculation

5.4.1 Linear programming considerations

In order to apply the Lagrangian dual procedure to solve the G3RLT, it is useful first to obtain a smaller model via the substitution of variables based on constraints (5-13e), thereby reducing the number of variables and constraints, in addition to halving the number of nonnegativity restrictions in (5-13i). Then, one uses the fact that $x_{ijp}^2 = x_{ijp}$ and replace x_{ijp} by y_{ijpijp}, so the y_{ijpijp}'s become 0-1 variables in the new model. Finally, it is necessary to replace the binary restrictions (5-13j) on $x_{ijp} = y_{ijpijp}$ by $0 \le x_{ijp} = y_{ijpijp} \le 1$, so that contrary to G3RLT being a mixed-integer programming problem, the new G3RLT1 is a continuous optimization problem. One obtains the following model.

$[\text{G3RLT1}]$

$$\min \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P} b_{ijp} y_{ijpijp} + \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P}\sum_{\substack{k=1\\k>i}}^{M}\sum_{n=1}^{N}\sum_{q=1}^{P} \tilde{c}_{ijpknq} y_{ijpknq} \qquad \text{(5-21a)}$$

where $\tilde{c}_{ijpknq} = c_{ijpknq} + c_{knqijp}$

s.t.

$$\sum_{n=1}^{N}\sum_{q=1}^{P} y_{ijpknq} = y_{ijpijp} \qquad i,k = 1,\dots,M;\ j = 1,\dots,N;\ p = 1,\dots,P; k > i, \qquad \text{(5-21b)}$$

$$\sum_{n=1}^{N}\sum_{q=1}^{P} y_{knqijp} = y_{ijpijp} \qquad i,k = 1,\dots,M;\ j = 1,\dots,N;\ p = 1,\dots,P; k < i, \qquad \text{(5-21c)}$$

$$\sum_{\substack{k=1\\k>i}}^{M}\sum_{q=1}^{P} s_{kn} y_{ijpknq} + \sum_{\substack{k=1\\k<i}}^{M}\sum_{q=1}^{P} s_{kn} y_{knqijp} \le S_n$$

$$i = 1,\dots,M;\ j,n = 1,\dots,N;\ p = 1,\dots,P, \qquad \text{(5-21d)}$$

$$\sum_{\substack{k=1\\k>i}}^{M}\sum_{n=1}^{N} t_{kq} y_{ijpknq} + \sum_{\substack{k=1\\k<i}}^{M}\sum_{n=1}^{N} t_{kq} y_{knqijp} \le T_q$$

$$i = 1,\ldots,M;\; j = 1,\ldots,N;\; p,q = 1,\ldots,P, \qquad \text{(5-21e)}$$

$$\sum_{j=1}^{N}\sum_{p=1}^{P} y_{ijpijp} = 1 \qquad i = 1,\ldots,M, \qquad \text{(5-21f)}$$

$$\sum_{i=1}^{M}\sum_{p=1}^{P} s_{ij} y_{ijpijp} \le S_j \qquad j = 1,\ldots,N, \qquad \text{(5-21g)}$$

$$\sum_{i=1}^{M}\sum_{j=1}^{N} t_{ip} y_{ijpijp} \le T_p \qquad p = 1,\ldots,P, \qquad \text{(5-21h)}$$

$$0 \le y_{ijpknq} \le 1 \qquad i,k = 1,\ldots,M;\; j,n = 1,\ldots,N;\; p,q = 1,\ldots,P; k > i, \text{(5-21i)}$$

$$0 \le y_{ijpijp} \le 1 \qquad i = 1,\ldots,M;\; j = 1,\ldots,N;\; p = 1,\ldots,P. \qquad \text{(5-21j)}$$

Notice that a weakened form is used in both equations (5-21d) and (5-21e), since $x_{ijp} = y_{ijpijp} \le 1$, as can be seen from the right-hand-side (RHS) of equations (5-13c) and (5-13d).

The reduction in the number of variables is not done just for the purpose of reducing computational effort. Tied with the benefit gained from the complementary pairs, one can shift cost at will between the two elements. It is essential to achieving much tighter lower bounds than would be reached without this significantly important step. The prior successful experiences with developing tight lower bounds for the QAP (Hahn and Grant, 1998), GQAP (Hahn et al., 2006a), and Q3AP (Hahn et al., 2006b) make sure of the effectiveness and efficacy of this procedure.

Now consider the following Lagrangian relaxation problem $\mathrm{LR}(\rho,\rho')$ on the G3RLT1, whereby constraints (5-21b) and (5-21c) are placed into the objective function using multipliers ρ and ρ' respectively.

$\left[\mathrm{LR}(\rho,\rho')\right]$

minimize:

$$\sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P} b_{ijp}y_{ijpijp} + \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P}\sum_{\substack{k=1\\k>i}}^{M}\sum_{n=1}^{N}\sum_{q=1}^{P}\tilde{c}_{ijpknq}y_{ijpknq}$$
$$+\sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P}\sum_{\substack{k=1\\k>i}}^{M}\rho_{ijpk}\left(y_{ijpijp}-\sum_{n=1}^{N}\sum_{q=1}^{P}y_{ijpknq}\right)+\sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P}\sum_{\substack{k=1\\k<i}}^{M}\rho'_{ijpk}\left(y_{ijpijp}-\sum_{n=1}^{N}\sum_{q=1}^{P}y_{knqijp}\right)$$

(5-22)

s.t. (5-21d)-(5-21j).

Here, the objective function of Lagrangian relaxation $\mathrm{LR}(\rho,\rho')$ can be reorganized as

$$\sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P} \tilde{b}_{ijp}y_{ijpijp} + \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P}\sum_{\substack{k=1\\k>i}}^{M}\sum_{n=1}^{N}\sum_{q=1}^{P}\overline{c}_{ijpknq}y_{ijpknq}\,, \tag{5-23a}$$

where

$$\tilde{b}_{ijp} = b_{ijp} + \sum_{\substack{k=1\\k>i}}^{M}\rho_{ijpk} + \sum_{\substack{k=1\\k<i}}^{M}\rho'_{ijpk} \qquad i=1,\dots,M;\ j=1,\dots,N;\ p=1,\dots,P, \tag{5-23b}$$

and

$$\overline{c}_{ijpknq} = \tilde{c}_{ijpknq} - \rho_{ijpk} - \rho'_{knqi} \qquad i,k=1,\dots,M;\ j,n=1,\dots,N;\ p,q=1,\dots,P; k>i\,. \tag{5-23c}$$

Therefore, the Lagrangian dual problem LD1 is

[LD1]

maximize $\theta(\rho,\rho')$ where

$$\theta(\boldsymbol{\rho},\boldsymbol{\rho}') = \min_{y}\left\{\sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P}\tilde{b}_{ijp}y_{ijpijp} + \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P}\sum_{\substack{k=1\\k>i}}^{M}\sum_{n=1}^{N}\sum_{q=1}^{P}\overline{c}_{ijpknq}y_{ijpknq}\right\} \tag{5-24}$$

s.t. (5-21d)-(5-21j).

Then, consider a further Lagrangian relaxation problem $\mathrm{LR}(\boldsymbol{\rho},\boldsymbol{\rho}',\boldsymbol{\gamma})$, whereby constraints (5-21f) are also placed into the objective function using multipliers $\boldsymbol{\gamma}$.

$\left[\mathrm{LR}(\boldsymbol{\rho},\boldsymbol{\rho}',\boldsymbol{\gamma})\right]$

minimize:

$$\sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P}\tilde{b}_{ijp}y_{ijpijp} + \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P}\sum_{\substack{k=1\\k>i}}^{M}\sum_{n=1}^{N}\sum_{q=1}^{P}\overline{c}_{ijpknq}y_{ijpknq} + \sum_{i=1}^{M}\gamma_i\left(1-\sum_{j=1}^{N}\sum_{p=1}^{P}x_{ijp}\right) \tag{5-25}$$

s.t. (5-21d), (5-21e), (5-21g)-(5-21j).

Here, the objective function of Lagrangian relaxation $\mathrm{LR}(\boldsymbol{\rho},\boldsymbol{\rho}',\boldsymbol{\gamma})$ can be simplified as

$$\sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P}\overline{b}_{ijp}y_{ijpijp} + \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P}\sum_{\substack{k=1\\k>i}}^{M}\sum_{n=1}^{N}\sum_{q=1}^{P}\overline{c}_{ijpknq}y_{ijpknq} + \sum_{i=1}^{M}\gamma_i\ , \tag{5-26a}$$

where

$$\overline{b}_{ijp} = \tilde{b}_{ijp} - \gamma_i = b_{ijp} + \sum_{\substack{k=1\\k>i}}^{M}\rho_{ijpk} + \sum_{\substack{k=1\\k<i}}^{M}\rho'_{ijpk} - \gamma_i$$

$$i = 1,\ldots,M;\ j = 1,\ldots,N;\ p = 1,\ldots,P. \tag{5-26b}$$

Finally, the Lagrangian dual problem LD2 is as shown

[LD2]

maximize $\theta(\boldsymbol{\rho},\boldsymbol{\rho}',\boldsymbol{\gamma})$ where

$$\theta\left(\boldsymbol{\rho},\boldsymbol{\rho}',\boldsymbol{\gamma}\right)=\min_{y}\left\{\sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P}\overline{b}_{ijp}y_{ijpijp}+\sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P}\sum_{\substack{k=1\\k>i}}^{M}\sum_{n=1}^{N}\sum_{q=1}^{P}\overline{c}_{ijpknq}y_{ijpknq}\right\}+\sum_{i=1}^{M}\gamma_i \quad (5\text{-}27)$$

s.t. (5-21d), (5-21e), (5-21g)-(5-21j).

LD2 amounts to solving the following Lagrangian subproblems for given multipliers $\boldsymbol{\rho},\boldsymbol{\rho}'$ and $\boldsymbol{\gamma}$.

$$\sum_{i=1}^{M}\gamma_i+\min\sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P}\left\{\begin{array}{l}\left(b_{ijp}+\sum\limits_{\substack{k=1\\k>i}}^{M}\rho_{ijpk}+\sum\limits_{\substack{k=1\\k<i}}^{M}\rho'_{ijpk}-\gamma_i\right)y_{ijpijp}\\ +\sum\limits_{\substack{k=1\\k>i}}^{M}\sum\limits_{n=1}^{N}\sum\limits_{q=1}^{P}\left\{\begin{array}{l}\left(\tilde{c}_{ijpknq}-\rho_{ijpk}-\rho'_{knqi}\right)y_{ijpknq}\\ \text{s.t.}\\ \sum\limits_{\substack{k=1\\k>i}}^{M}\sum\limits_{q=1}^{P}s_{kn}y_{ijpknq}+\sum\limits_{\substack{k=1\\k<i}}^{M}\sum\limits_{q=1}^{P}s_{kn}y_{knqijp}\le S_n,\left(\forall n\right)\\ \sum\limits_{\substack{k=1\\k>i}}^{M}\sum\limits_{n=1}^{N}t_{kq}y_{ijpknq}+\sum\limits_{\substack{k=1\\k<i}}^{M}\sum\limits_{n=1}^{N}t_{kq}y_{knqijp}\le T_q,\left(\forall q\right)\\ 0\le y_{ijpknq}\le 1,\left(\forall k,n,q\right)\\ k=1,\ldots,M;n=1,\ldots,N;q=1,\ldots,P;k>i\end{array}\right\}\\ \\ \text{s.t.}\\ \sum\limits_{i=1}^{M}\sum\limits_{p=1}^{P}s_{ij}y_{ijpijp}\le S_j,\left(\forall j\right)\\ \sum\limits_{i=1}^{M}\sum\limits_{j=1}^{N}t_{ip}y_{ijpijp}\le T_p,\left(\forall p\right)\\ 0\le y_{ijpijp}\le 1,\left(\forall i,j,p\right)\\ i=1,\ldots,M;j=1,\ldots,N;p=1,\ldots,P\end{array}\right\}$$

(5-28)

Observe first that in formulation (5-28), if the multipliers $\boldsymbol{\rho}$ and $\boldsymbol{\rho}'$ are positive quantities, then these amounts are subtracted from $k-\text{planes}$ within the 3-dimensional modified cost submatrices whose elements become $\bar{c}_{ijpknq} = \left(\tilde{c}_{ijpknq} - \rho_{ijpk} - \rho'_{knqi}\right)$ and added to the corresponding linear cost element b_{ijp}, which are easily recognized to be Class 1 operations defined earlier. Observe next that, if the multipliers $\boldsymbol{\gamma}$ are also positive quantities, then these amounts are subtracted from i-planes within the 3-dimensional matrix of modified linear cost $\tilde{b}_{ijp} = \left(b_{ijp} + \sum_{\substack{k=1 \\ k>i}}^{M} \rho_{ijpk} + \sum_{\substack{k=1 \\ k<i}}^{M} \rho'_{ijpk} \right)$ and added to the value of $\theta\left(\boldsymbol{\rho}, \boldsymbol{\rho}', \boldsymbol{\gamma}\right)$, which are recognized to be Class 2 operations. Therefore, the operations implied by LD2 constituting both Class 1 and Class 2 operations to achieve the level-1 RLT dual-ascent procedure bound.

If operations on $\boldsymbol{C}$ decrease the cost by an amount Z' and are performed in a way that keeps the elements of $\boldsymbol{C}$ nonnegative, then no assignment cost can become negative and the following relationship holds.

$$Z' \leq \min_{\boldsymbol{X}} \; Z\left(\boldsymbol{X}\right) \tag{5-29}$$

Thus the dual-ascent procedure bound calculation for the GQ3AP can be addressed as maximizing the sum of downward cost shifts Z' permitted by Class 1 and Class 2 operations under the constraints that no cost element in $\boldsymbol{C}$ is driven negative.

5.4.2 The level-1 RLT dual-ascent procedure

In this subsection, I describe the level-1 RLT dual-ascent procedure for the GQ3AP programmed by Professor Hahn.

The level-1 RLT dual-ascent procedure algorithm is based on the concept that constant amounts are subtracted from the $k-\text{planes}$ of each and every 3-dimensional modified cost coefficient submatrix $\left[\tilde{c}_{ijpknq}\right]$ and added to the corresponding linear cost element b_{ijp} of the submatrix (Class 1 operations). Then, constant amounts are subtracted from the $i\text{-planes}$ of the modified linear cost matrix $\left[\tilde{b}_{ijp}\right]$ and added to $\theta\left(\rho,\rho',\gamma\right)$, which comprises a lower bound value (Class 2 operations). Here are the steps,

1. For each of the $M \times N \times P$ 3-dimentsional modified cost submatrices, first collect into the submatrix those costs corresponding to complementary variables y_{ijpknq} and y_{knqijp}. Then, within the submatrix, subtract the minimum modified cost from each $k\text{-plane}$ and add it to the corresponding linear cost element. Adding complementary costs first assures the largest possible transfer of quadratic to linear costs.
2. After Step 1 has been done for all the submatrices, subtract the minimum cost from each $i\text{-plane}$ of the linear cost matrix and add it to a reduction constant which constitutes a lower bound to the problem. If the resulting pattern of zeros satisfies the problem constraints, the procedure terminates with an optimum solution to the GQ3AP. If not, continue to Step 3. Stop if the bound increase is poor.

3. Scan the matrix of modified linear costs for positive elements. Record locations of the non-positive (i.e., zero) linear costs as this information will be needed in Step 4. Divide each positive linear cost element into $M-1$ approximately equal parts and add each part to a different (non linear-cost) k-plane of its submatrix. Positive linear costs are divided into $M-1$ approximately equal parts as follows. First of all, it must be emphasized that it is essential that round-off errors must never be allowed in manipulating the $\boldsymbol{C}$ matrix. Thus, the $\boldsymbol{C}$ matrix in the algorithm is always integer. So, to divide the linear cost into $M-1$ almost equal parts, the procedure is to divide by $M-1$ and round that down to the nearest integer. That provides $M-2$ equal parts. Then those parts are added up and subtracted from the original linear cost to calculate the remaining part. The motivation here is that by replacing costs into submatrices, there is a new opportunity to make those linear costs which had low or zero values after Step 2 larger, permitting even more cost to be eventually moved to the lower bound. Step 3 leaves the cost matrix $\boldsymbol{C}$ dramatically rearranged. Because of the rearrangement, the process can be repeated, i.e., additional costs can be subtracted from within submatrices and moved to the linear cost element. The result is an iterative procedure that produces growth in the lower bound with each round. This growth, while significant, diminishes with each round, so at some point it does not pay to continue the process.

4. Repeat Step1, except this time be sure to do the collection of complementary costs and subtractions first for those submatrices whose linear costs were zero prior to Step 3. Go to Step 2.

The collecting of complementary costs prior to solving each submatrix is explained by the substitution of variables that eliminated constraints (5-13e) in G3RLT. Step 1 and Step 4 are fully explained by formulation LD1. Since Step 3 consists only of Class 1 operations, it is also explained. Step 2 is clarified by formulation LD2 which constitutes both Class 1 and Class 2 operations, as are all four steps of the level-1 dual-ascent procedure.

5.4.3 Adding the resource constraints

Equality constraints (5-21b), (5-21c) and (5-21f) are dualized and dealt with appropriately in the construction of the level-1 RLT dual-ascent procedure. The resource constraints (5-21d)-(5-21e), and (5-21g)-(5-21h) must be enforced in other ways. For the most part, it is done by calculating dual-ascent procedure lower bounds on only those assignments that satisfy the four sets of resource constraints. However, an ingenious method is implemented that improves the dual-ascent procedure bound value, by taking into consideration the resource constraints. Constraints (5-21d)-(5-21e) are enforced during the level-1 RLT lower bound calculation by raising to a very high cost value those submatrix elements whose selection would violate these two sets of constraints. Thus, it is possible to improve the lower bound beyond that available if only the equality constraints (5-21b) and (5-21c) are enforced.

Consider the Class 1 subtraction operations defined before. After collecting complementary costs in a given submatrix $\boldsymbol{C}_{[ijp]}$, minimum costs are subtracted from submatrix k-planes and added to the corresponding linear cost element b_{ijp}. Prior to these subtractions, an important step has been added. Each element in the n-plane containing the linear cost element is examined, here the n-plane is the plane along (k,q) in the submatrix $\boldsymbol{Y}_{[ijp]}$. If including this element in a solution would render infeasible solution pattern regarding constraints (5-21d) for this submatrix, then the cost for this element is raised to an arbitrarily large number called GREAT and dictated only by the integer size limitations of the computer. The same process is done in every q-plane for examining the infeasibility posed by constraints (5-21e), while the q-plane is the plane along (k,n) in the submatrix $\boldsymbol{Y}_{[ijp]}$.

Similar operations are also conducted in the linear cost matrix so as to enforce the resource constraints (5-21g) and (5-21h). This process effectively bars infeasible elements from inclusion in a level-1 RLT dual-ascent procedure solution, and focuses the reduction efforts on those elements still eligible for consideration. The ensuing subtractions permit additional cost to be extracted from each and every submatrix, leading to a much-improved lower bound.

5.4.4 Advantages of the level-1 RLT dual-ascent procedure

The level-1 RLT dual-ascent procedure has a number of valuable properties that makes it ideal for use in a branch-and-bound algorithm for solving the GQ3AP.

1. The level-1 RLT dual-ascent procedure is still among competing lower bounding techniques for the QAP, as demonstrated from experimental results, see Table 2

of Loiola et al. (2006) and Drezner et al. (2005). Moreover, it is the only method that is able to solve the GQAP of size 20×15, see Hahn et al. (2006a).

2. Similar to a number of other lower bounding techniques, the level-1 RLT dual-ascent procedure generates a series of non-decreasing lower bounds for the GQ3AP. More importantly, with each such lower bound, it modifies the GQ3AP objective function so that a new GQ3AP is generated whose feasible solution set is identical to that of the original and whose objective function values are merely lessened by the amount of the lower bound.
3. The dual-ascent procedure for a given partial assignment can be stopped as soon as the lower bound on the assumed partial assignment exceeds an upper bound on the original problem. It can also be stopped, in favor of making an additional partial assignment, when it becomes obvious that from its slow progress it is unlikely to ever reach the upper bound.
4. In a branch-and-bound algorithm, fathoming decisions can easily be recorded by arbitrarily increasing the costs of those elements in the cost matrix that correspond to partial assignments which have been eliminated as possibly being optimum. This modifies the GQ3AP, but is assured to fully enumerate all feasible solutions of the original problem.

5.4.5 The level-1 RLT dual-ascent procedure in branch-and-bound

The level-1 RLT dual-ascent procedure is utilized within a branch-and-bound algorithm as the auxiliary procedure for computing lower bounds. This method is similar to the branch-and-bound algorithm for the GQAP by Hahn et al. (2006a) since the GQ3AP trees search must be limited to only those assignments that meet the resource constraints.

Branching follows the conventional technique of selecting a single 3-dimensional $i-j-p$ assignment at the first (highest) level, as well as at subsequent levels of partial assignment. In order to implement this selection, a linear cost is chosen to be involved in the 3-dimensional assignment. Based on the selection of linear cost b_{ijp}, the 3-dimensional submatrix $\boldsymbol{C}_{[ijp]}$ is involved in the assignment. The remaining submatrices of cost matrix $\boldsymbol{C}$ whose i-plane containing submatrix $\boldsymbol{C}_{[ijp]}$ disappear (as they cannot be involved in the assignment) and the problem is thus reduced to a GQ3AP of size $(M-1)\times N\times P$. It turns out that one k-plane likewise disappears from each submatrix of the remainders in the cost matrix $\boldsymbol{C}$, which make all the submatrices to be $(M-1)\times N\times P$ in size as well.

It is the application of the level-1 RLT dual-ascent procedure lower bounding calculation on the $(M-1)\times N\times P$ size problem that attempts to fathom a partial assignment postulated by the selection of linear cost b_{ijp}. By fathoming, one calculates a lower bound and tests it against the best-known upper bound. If the best-known upper bound is exceeded, the partial assignment is eliminated from the problem.

Recall in the previous sections, the dual-ascent procedure moves costs out of the $\boldsymbol{C}$ matrix into a lower bound value, leaving a modified matrix $\boldsymbol{C}'$. For subsequent branch-and-bound operations along a given partial assignment path, the strategy is to take advantage of this fact and to use this reduced cost matrix $\boldsymbol{C'}$ for setting up subsequent subproblems deeper into the tree. Thus, lower bounds are calculated not from the original problem, but from the subproblems that the level-1 RLT dual-ascent procedure already processed at earlier (higher) levels of partial assignment. Using the modified matrix $\boldsymbol{C'}$ of each of these subproblems has the additional benefit that the subproblems is brought closer to dual solution, making it more likely that better feasible solutions will be found or that lower bounds will exceed upper bounds, thus cutting off a branch. The choice of the number of dual iterations at a given partial assignment is a dynamic decision, based on the progress of the lower bound achieved after a fixed number of iterations. If after a small number of iterations at the current partial assignment, sufficient progress is not reached, the lower bounding attempt stops and the algorithm proceeds to make an additional assignment.

Tree search is depth-first and based on single $i-j-p$ assignment. The order in which i assignments are made is very important and has profound influence on branch-and-bound runtime. The branching strategy follows the same strategies as those developed by Hahn et al. (2001) for the QAP. Levels in the search tree are defined by how many partial assignments are made. The root is where no partial assignments have been made and is considered level 0. A single partial assignment is considered level 1, etc. Since the tree search involves one-to-one-to-one 3-dimensional assignments, at each level of the tree it is necessary only to examine a given i and its assignments to all possible (i.e., feasible) j's and p's.

In the GQ3AP, more than one i can be reside at a given j or p. Therefore, one must exhaust all possible j's and p's at a given level, before one can determine if a better solution exists or if a branch of the tree at that level can be cut off. The exception to this rule is that no further consideration need be given to evaluating assignments if the resource constraints notify that the locations can accept no further assignments.

The search strategy for selecting the next node is a simple depth-first strategy. If fathoming a given node is unsuccessful, an additional partial assignment is made, thus increasing the depth into the tree. If a node has been fathomed successfully, the next available node is selected. Partial assignment at a given level are selected according to a set of look-ahead bounding calculations that essentially determine the difficulty of eliminating the branch containing that assignment. Assignments selected in the order of decreasing difficulty are made when it is desired to find attractive feasible solutions to replace the upper bound, early in the search.

When the algorithm accumulates sufficient fathoming information, permanent decisions can be made on the assignment matrix that certain elements of the assignment matrix $\boldsymbol{Y} = \left[y_{ijpknq} \right]$ are zero. These are recorded in the modified cost matrix $\boldsymbol{C}'$ by setting the corresponding element costs to be a very large value GREAT. This effectively bars the "decided-zero" elements from inclusion in a level-1 RLT dual-ascent procedure solution, and focuses the reduction efforts on those elements still eligible for consideration. The communication of permanent decisions generally permits additional cost to be extracted from the matrix, resulting in an improved lower bound.

5.5 Computational experiments

The branch-and-bound code for the GQ3AP was programmed in FORTRAN 77 by Professor Hahn, using the techniques developed previously for the branch-and-bound algorithms of the Q3AP and the GQAP. Here I present the results I achieved using this code for a set of MSAP problem instances with varying number of departments (M), floors (N), stairwells (P), and evacuation population flow characteristics. I am grateful to Prof. J. MacGregor Smith from University of Massachusetts Amherst for providing the instance data for computational testing.

Figure 5-3 illustrates the general arrangement of the stairwells. I have varied the number of stories in these experiments from seven to eight and the number of departments from ten to thirteen and the number of stairwells from two to three. Figure 5-4 shows a typical facility section configuration for the stairwells and stories and evacuation target at the ground level. For the first example, it is assumed that the occupants of the seven-story building comprise ten different departments and that there are two stairwells strategically located that can be accessed on every floor.

Figure 5-3. Multi-story plan.

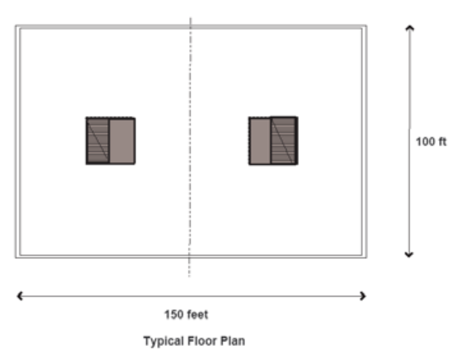
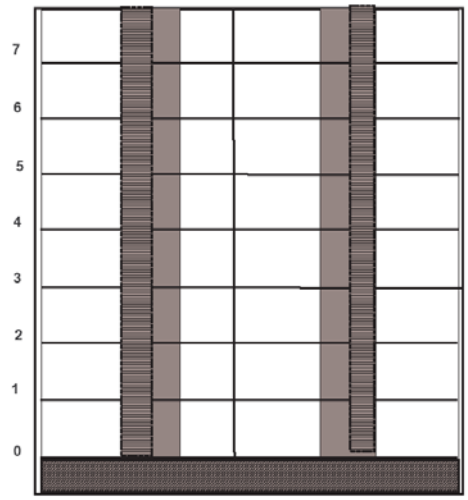


Figure 5-4. Multi-story facility.



A typical set of flows for the ten departments are given in the flow matrix $\boldsymbol{F}$.

$$\boldsymbol{F} = \begin{pmatrix} 0 & 10 & 0 & 3 & 0 & 5 & 4 & 5 & 5 & 2 \\ 0 & 0 & 5 & 4 & 5 & 0 & 0 & 0 & 4 & 4 \\ 0 & 0 & 0 & 1 & 0 & 3 & 4 & 7 & 4 & 7 \\ 0 & 0 & 0 & 0 & 4 & 6 & 10 & 3 & 0 & 2 \\ 0 & 0 & 0 & 0 & 0 & 5 & 3 & 8 & 7 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 7 & 8 & 3 & 9 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 9 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & 3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 9 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

In this first example, it is assumed that the following area requirements $\left[a_{ij}\right]$ for the ten departments.

$$\left[a_{ij}\right]=\begin{pmatrix} 15 & 15 & 15 & 15 & 15 & 15 & 15 & 15 & 15 & 15 \\ 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 \\ 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 \\ 15 & 15 & 15 & 15 & 15 & 15 & 15 & 15 & 15 & 15 \\ 15 & 15 & 15 & 15 & 15 & 15 & 15 & 15 & 15 & 15 \\ 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 \\ 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 \\ 15 & 15 & 15 & 15 & 15 & 15 & 15 & 15 & 15 & 15 \\ 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 \\ 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 & 7.5 \end{pmatrix}$$

And the following evacuation flows $\left[\lambda_{ip}\right]$ for the ten departments.

$$\left[\lambda_{ip}\right]=\begin{pmatrix} 100 & 100 \\ 50 & 50 \\ 40 & 40 \\ 120 & 120 \\ 100 & 100 \\ 65 & 65 \\ 70 & 70 \\ 150 & 150 \\ 50 & 50 \\ 70 & 70 \end{pmatrix}$$

To complete the constraints for the ten-department, seven-story, two-stairwell example, the floor space on each floor is chosen to be 15,000 sq.ft. units and each stairwell capacity to be 1500 people/minute.

In the following numerical experiments, the value of $\boldsymbol{\lambda}$ plays an important role in the final outcome. $\boldsymbol{\lambda}=\left[\lambda_{ip}\right]$ is defined as the arrival rate of persons in the evacuation from department i to stairwell p. Besides being the restricted parameter for the stairwell capacity constraints (5-1d), $\boldsymbol{\lambda}$ in the objective function (5-1a) is the measurement of how important escape costs are compared to everyday inter-department travel costs. With larger λ_{ip} values, the MSAP design puts more weight on the evacuation cost, which makes the runtime longer. I have chosen three different orders of magnitude of $\boldsymbol{\lambda} = \left[\lambda_{ip}\right]$ vector. The first choice is that λ_{ip} are all between 40 and 150, resulting in a hard problem. The second choice is that λ_{ip} are all between 4 and 15, resulting in a medium-hard problem and the third choice is that λ_{ip} are all between 0.4 and 1.5, resulting in an easy problem.

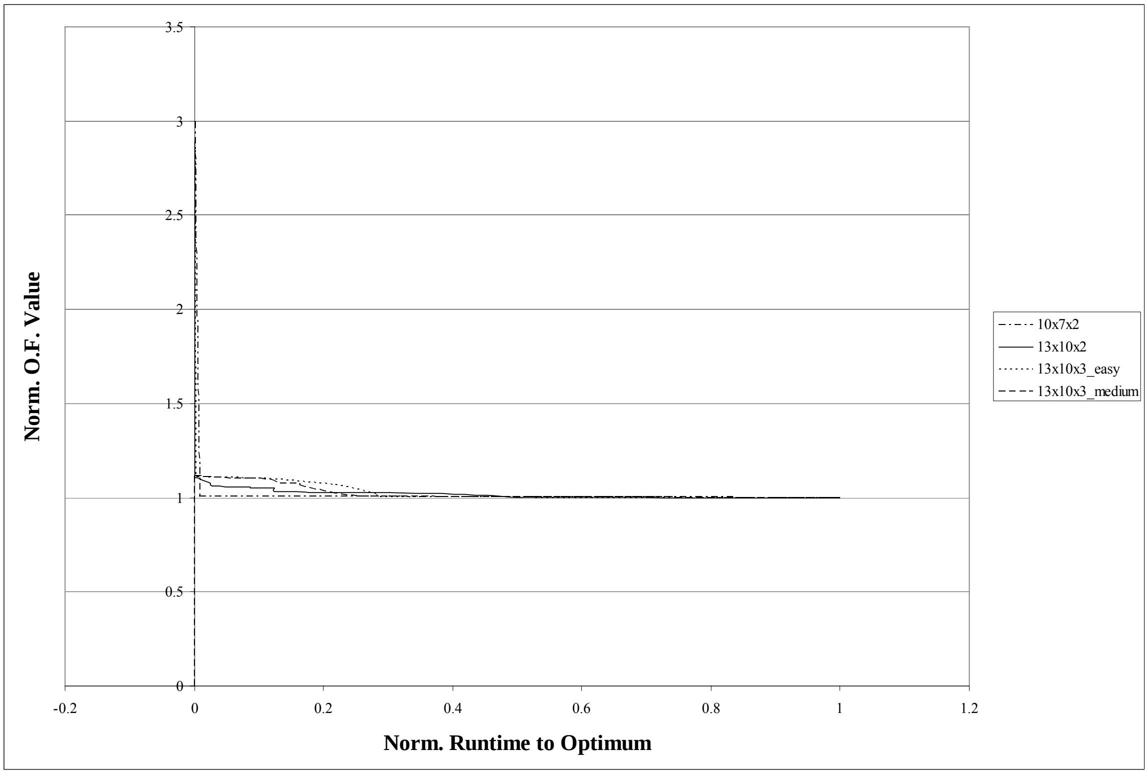
Table 5-1 presents the algorithm results for a small series of experiments. The first experiment in this table refers to the seven-story building, ten-department and two-stairwell example described above. The experiments in Table 5-1 were performed on a single 900MHz CPU of a Sun Blade 1000 workstation or on a single 1.2GHz CPU of a Sun Fire V880 server. Column one provides the number of departments, number of floors and number of stairwells. Column two presents the ratio of the average magnitude of the $\lambda_{ip} e_{jp}$ term in the objective function to the average magnitude of the $f_{ik} d_{jn}$ term. Column three shows the optimal objective function value of each example. Column four reports the number of optima found in the branch-and-bound search. Column five gives the number of nodes (partial assignments) evaluated in the search. Column six lists the total runtime in seconds, normalized to the speed of the faster V880 server and column seven addresses the runtime in seconds to the optimum.

Table 5-1. Branch-and-bound algorithm results.

M, N, P	OF Ratio	Optimum	# Optima	# Nodes	Runtime	Time to Opt.
10, 7, 2	22.4	2,582	408	964,134	730	19
13, 8, 2 easy	3.03	692	2	41,606,506	53,169	29,380
13, 8, 2 hard	30.32	2,788	1	125,090,900	168,100	33,431
13, 8, 3 easy	0.33	351.049	19,683	4,521,819	26,610	13,461
13, 8, 3 medium	3.29	630.5	46,902	32,607,159	170,592	82,317
13, 8, 3 hard	32.92	3,086	59,049	818,696,328	2,330,368	76,791

Figure 5-5 illustrates the performance of the algorithm as a function of time for selected problem instances. It demonstrates that the algorithm encounters very good quality feasible solutions early in the branch-and-bound search process. Furthermore, the optimal solution is eventually reached in a very short amount of time, thus indicating the potential heuristic value of the branch-and-bound algorithm for generating heuristic solutions quickly. While a performance guarantee of such a heuristic is not available, it still represents a viable approach to generating good solutions quickly, since the cutoff-time period is small in comparison to the total running time for the algorithm.

Figure 5-5. Algorithm solution graph.



5.6 Conclusion

This chapter analyses a new defined assignment problem, so-called the generalized quadratic 3-dimensional assignment problem (GQ3AP), which is the generalization of both the generalized quadratic assignment problem (GQAP) and the quadratic 3-dimensional assignment problem (Q3AP). This GQ3AP arises in many applications such as multi-story evacuation design and crossdock layout design. I describe a Lagrangian dual for the GQ3AP based on a level-1 reformulation-linearization technique (RLT) dual-ascent procedure similar to those successfully used for the GQAP and Q3AP. My experimental results show the branch-and-bound algorithm embedded with the dual-ascent procedure has solved several instances of multi-story space assignment problem (MSAP) for the first time.

6. The Level-3 Reformulation-linearization Technique (RLT) Formulation of the Quadratic Assignment Problem (QAP)

6.1 Introduction

The QAP is among the most difficult combinatorial optimization problems. This is unfortunate, since a vast array of applications arises in facility layout and design. Solving general problems of size greater than $N = 30$, i.e. with more than 900 binary variables, is still computationally impractical.

Although the QAP is NP-hard, this complexity is not sufficient to explain its difficulty, as other classes of NP-hard problems can be solved far more efficiently than the QAP. The majority of QAP test problems have a homogeneous objective function, and this contributes to their difficulty. Such homogeneity tends to produce bounds that are less effective in pruning partial solutions within binary search trees. Among exact algorithms, branch-and-bound methods are the most successful, but lack of tight lower bounds has been one of the major stumbling blocks.

The prior computational experience using at first level-1 and then level-2 RLT QAP formulations has indicated promising research directions. The resulting linear representations, Problems $\overline{\text{RLT1}}$ and $\overline{\text{RLT2}}$, are increasingly large in size and highly degenerate. In order to solve these problems, Hahn and Grant (1998) and Adams et al. (2006) have presented a dual-ascent strategy that exploits the block-diagonal structure of constraints in the level-1 and level-2 forms, respectively. This strategy is a powerful extension of that found in Adams and Johnson (1994).

Problem $\overline{\text{RLT2}}$, in particular, provides sharp lower bounds, as shown in Table 1 of Loiola et al. (2006), and consequently leads to very competitive exact solution approaches. A striking outcome, documented in Table 2 of Loiola et al. (2006), is the relatively few number of nodes considered in the binary search tree to verify optimality. This leads to marked success in solving difficult QAP instances of size $N \geq 24$ in record computational time.

In this chapter, I introduce the level-3 RLT formulation of the QAP, and show that the hierarchy of quadratic, cubic and biquadratic assignment problems is directly related to the RLT hierarchy. Also presented are the superior lower bounds provided by the level-3 RLT QAP model.

6.2 The biquadratic assignment problem (BQAP)

The QAP minimizes a quadratic function over an assignment matrix. Burkard et al. (1994) gave the definition of the biquadratic assignment problem (BQAP) which arose in the field of VLSI synthesis. The BQAP is to minimize a weighted sum of products of four binary variables subject to multiple choice constraints on those variables.

Given two 4-dimensional matrices of N^4 elements $\bar{\boldsymbol{A}} = \left[\alpha_{ikpg}\right]$ and $\bar{\boldsymbol{B}} = \left[\beta_{jnqh}\right]$, the BQAP can be written as

$$\min \left\{ \begin{array}{l} \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{N}\sum_{n=1}^{N}\sum_{p=1}^{N}\sum_{q=1}^{N}\sum_{g=1}^{N}\sum_{h=1}^{N} \alpha_{ikpg}\beta_{jnqh}x_{ij}x_{kn}x_{pq}x_{gh} \\ : \boldsymbol{x} \in \boldsymbol{X}, \boldsymbol{x} \text{ binary} \end{array} \right\}, \tag{6-1a}$$

$$\text{where } \boldsymbol{x} \in \boldsymbol{X} \equiv \left\{ \boldsymbol{x} \geq 0 : \sum_{i=1}^{N} x_{ij} = 1, \forall\left(j = 1,\ldots,N\right); \sum_{j=1}^{N} x_{ij} = 1, \forall\left(i = 1,\ldots,N\right) \right\}. \tag{6-1b}$$

Notice that the constraints of the BQAP are the usual multiple choice constraints over the assignment matrix. Thus, the solution matrix $\boldsymbol{X} = \left[x_{ij} \right]$ to the BQAP is also a permutation matrix. By representing the variables x_{ij} by a permutation of the set $\{1,...,N\}$, one gets the following formulation in permutation as

$$\min_{\varphi\in\Gamma_N} \sum_{i=1}^{N}\sum_{k=1}^{N}\sum_{p=1}^{N}\sum_{g=1}^{N} \alpha_{ikpg}\beta_{\varphi(i)\varphi(k)\varphi(p)\varphi(g)}, \tag{6-2}$$

where Γ_N denotes the set of all permutations of $\{1,...,N\}$ and $\varphi \in \Gamma_N$.

If the coefficients $\boldsymbol{E} = \left[e_{ijknpqgh} \right] \in \mathbb{R}^{N^8}$ are the costs associated with the products of four binary variables, the BQAP becomes

$$\min \left\{ \begin{array}{l} \displaystyle\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{N}\sum_{n=1}^{N}\sum_{p=1}^{N}\sum_{q=1}^{N}\sum_{g=1}^{N}\sum_{h=1}^{N} e_{ijknpqgh}x_{ij}x_{kn}x_{pq}x_{gh} \\ : \boldsymbol{x} \in \boldsymbol{X}, \boldsymbol{x} \text{ binary} \end{array} \right\}. \tag{6-3}$$

An alternative way is to show linear costs $\boldsymbol{B} = \left[b_{ij} \right] \in \mathbb{R}^{N^2}$, quadratic costs $\boldsymbol{C} = \left[c_{ijkn} \right] \in \mathbb{R}^{N^4}$, cubic costs $\boldsymbol{D} = \left[d_{ijknpq} \right] \in \mathbb{R}^{N^6}$, and biquadratic costs $\boldsymbol{E} = \left[e_{ijknpqgh} \right] \in \mathbb{R}^{N^8}$ individually. Now the BQAP is

$$\min \left\{ \begin{array}{l} \displaystyle\sum_{i=1}^{N}\sum_{j=1}^{N} b_{ij}x_{ij} + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{N}\sum_{\substack{n=1\\n\neq j}}^{N} c_{ijkn}x_{ij}x_{kn} + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{N}\sum_{\substack{n=1\\n\neq j}}^{N}\sum_{\substack{p=1\\p\neq i,k}}^{N}\sum_{\substack{q=1\\q\neq j,n}}^{N} d_{ijknpq}x_{ij}x_{kn}x_{pq} \\ \displaystyle+\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{N}\sum_{\substack{n=1\\n\neq j}}^{N}\sum_{\substack{p=1\\p\neq i,k}}^{N}\sum_{\substack{q=1\\q\neq j,n}}^{N}\sum_{\substack{g=1\\g\neq i,k,p}}^{N}\sum_{\substack{h=1\\h\neq j,n,q}}^{N} e_{ijknpqgh}x_{ij}x_{kn}x_{pq}x_{gh} \\ : \boldsymbol{x} \in \boldsymbol{X}, \boldsymbol{x} \text{ binary} \end{array} \right\}.$$

(6-4)

6.3 The cubic assignment problem (CAP) and the level-2 RLT formulation of the QAP

Similarly, I define for the first time the cubic assignment problem (CAP), which is to minimize the weighted sum of products of three binary variables over the same assignment matrix. If one uses the notations above, the CAP can be formulated as

$$\min \quad \left\{ \begin{array}{l} \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{N}\sum_{n=1}^{N}\sum_{p=1}^{N}\sum_{q=1}^{N} d_{ijknpq} x_{ij} x_{kn} x_{pq} \\ : \boldsymbol{x} \in \boldsymbol{X}, \boldsymbol{x} \text{ binary} \end{array} \right\}, \tag{6-5}$$

Or,

$$\min \quad \left\{ \begin{array}{l} \sum_{i=1}^{N}\sum_{j=1}^{N} b_{ij} x_{ij} + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{\substack{k=1 \\ k\neq i}}^{N}\sum_{\substack{n=1 \\ n\neq j}}^{N} c_{ijkn} x_{ij} x_{kn} + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{\substack{k=1 \\ k\neq i}}^{N}\sum_{\substack{n=1 \\ n\neq j}}^{N}\sum_{\substack{p=1 \\ p\neq i,k}}^{N}\sum_{\substack{q=1 \\ q\neq j,n}}^{N} d_{ijknpq} x_{ij} x_{kn} x_{pq} \\ : \boldsymbol{x} \in \boldsymbol{X}, \boldsymbol{x} \text{ binary} \end{array} \right\}. \tag{6-6}$$

If one introduces the variables y_{ijkn} and z_{ijknpq} to substitute the products of $y_{ijkn} = x_{ij} x_{kn}$ and $z_{ijknpq} = x_{ij} x_{kn} x_{pq}$ respectively, the formulation of the CAP is the same as the level-2 RLT model of the QAP, which is repeated below. (Construction details of Problem RLT2 given in Section 2.5.2).

[RLT2]

$$\min \quad \sum_{i=1}^{N}\sum_{j=1}^{N} b_{ij} x_{ij} + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{\substack{k=1 \\ k\neq i}}^{N}\sum_{\substack{n=1 \\ n\neq j}}^{N} c_{ijkn} y_{ijkn} + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{\substack{k=1 \\ k\neq i}}^{N}\sum_{\substack{n=1 \\ n\neq j}}^{N}\sum_{\substack{p=1 \\ p\neq i,k}}^{N}\sum_{\substack{q=1 \\ q\neq j,n}}^{N} d_{ijknpq} z_{ijknpq} \tag{6-7a}$$

$$\text{s.t.} \quad \sum_{\substack{p=1 \\ p\neq i,k}}^{N} z_{ijknpq} = y_{ijkn} \qquad i, j, k, n, q = 1, ..., N; q \neq n \neq j, k \neq i, \tag{6-7b}$$

$$\sum_{\substack{q=1\\q\neq j,n}}^{N} z_{ijknpq} = y_{ijkn} \qquad i, j, k, n, p = 1,..., N;\, p \neq k \neq i, n \neq j\,, \tag{6-7c}$$

$$z_{ijknpq} = z_{knijpq} = z_{ijpqkn} = z_{knpqij} = z_{pqijkn} = z_{pqknij}$$

$$i, j, k, n, p, q = 1,..., N;\, p > k > i, q \neq n \neq j\,, \tag{6-7d}$$

$$z_{ijknpq} \geq 0 \qquad i, j, k, n, p, q = 1,..., N;\, p \neq k \neq i, q \neq n \neq j\,, \tag{6-7e}$$

$$\sum_{\substack{k=1\\k\neq i}}^{N} y_{ijkn} = x_{ij} \qquad i, j, n = 1,..., N;\, n \neq j\,, \tag{6-7f}$$

$$\sum_{\substack{n=1\\n\neq j}}^{N} y_{ijkn} = x_{ij} \qquad i, j, k = 1,..., N;\, k \neq i\,, \tag{6-7g}$$

$$y_{ijkn} = y_{knij} \qquad i, j, k, n = 1,..., N;\, k > i, n \neq j\,, \tag{6-7h}$$

$$y_{ijkn} \geq 0 \qquad i, j, k, n = 1,..., N;\, k \neq i, n \neq j\,, \tag{6-7i}$$

$$\boldsymbol{x} \in \boldsymbol{X}, \boldsymbol{x} \text{ binary}. \tag{6-7j}$$

Since the additional constraints (6-7b)-(6-7i) in Problem RLT2 are derived completely from the substitutions of $y_{ijkn} = x_{ij}x_{kn}$ and $z_{ijknpq} = x_{ij}x_{kn}x_{pq}$, one can use the level-2 RLT QAP model to solve the CAP.

6.4 The level-3 RLT formulation of the QAP

One could expect the level-3 RLT QAP to be used for solving the BQAP. The level-3 RLT formulation is constructed as follows. In its reformulation step, multiply each of $2N$ equality constraints and each of N^2 nonnegativity restrictions (which are rewritten in variables x_{kn}) by each of N^2 binary variables x_{ij}. Then, multiply each of

$2N$ equality constraints and each of N^2 nonnegativity restrictions (which are rewritten in variables x_{pq}) by each of $N^2(N-1)^2$ pairwise products of variables $x_{ij}x_{kn}$ $(k \neq i$ and $n \neq j)$. Then, multiply each of $2N$ equality constraints and each of N^2 nonnegativity restrictions (which are rewritten in variables x_{gh}) by each of $N^2(N-1)^2(N-2)^2$ pairwise products of variables $x_{ij}x_{kn}x_{pq}$ $(p \neq k \neq i$ and $q \neq n \neq j)$. Append all these new restrictions. Express the various resulting products in the order $x_{ij}x_{kn}$, $x_{ij}x_{kn}x_{pq}$ and $x_{ij}x_{kn}x_{pq}x_{gh}$. Substitute $x_{ij} = x_{ij}^2$, reduce $x_{ij}x_{ij}x_{kn}$ and $x_{kn}x_{ij}x_{kn}$ to $x_{ij}x_{kn}$, and reduce $x_{ij}x_{ij}x_{kn}x_{pq}$, $x_{kn}x_{ij}x_{kn}x_{pq}$ and $x_{pq}x_{ij}x_{kn}x_{pq}$ to $x_{ij}x_{kn}x_{pq}$. Set $x_{ij}x_{kn} = 0$ if $(k = i$ and $n \neq j)$ or $(k \neq i$ and $n = j)$ in all quadratic expressions. And set $x_{ij}x_{kn}x_{pq} = 0$ if $(p = i$ and $q \neq j)$, $(p = k$ and $q \neq n)$, $(p \neq i$ and $q = j)$ or $(p \neq k$ and $q = n)$ in all cubic expressions. And set $x_{ij}x_{kn}x_{pq}x_{gh} = 0$ if $(g = i$ and $h \neq j)$, $(g = k$ and $h \neq n)$, $(g = p$ and $h \neq q)$, $(g \neq i$ and $h = j)$, $(g \neq k$ and $h = n)$ or $(g \neq p$ and $h = q)$ in all biquadratic expressions. Then, in the linearization step, substitute every occurrence of each product $x_{ij}x_{kn}$ $(k \neq i$ and $n \neq j)$ with a single nonnegative continuous variable y_{ijkn}. And, substitute every occurrence of each product $x_{ij}x_{kn}x_{pq}$ $(p \neq k \neq i$ and $q \neq n \neq j)$ with a single nonnegative continuous variable z_{ijknpq}. Also, substitute every occurrence of each product $x_{ij}x_{kn}x_{pq}x_{gh}$ $(g \neq p \neq k \neq i$ and $h \neq q \neq n \neq j)$ with a single nonnegative continuous variable $v_{ijknpqgh}$. Enforce the symmetric restrictions that $y_{ijkn} = y_{knij}\ \forall(i, j, k, n = 1,..., N), k > i, n \neq j$. Also, enforce the symmetric restrictions that

$z_{ijknpq} = z_{knijpq} = z_{ijpqkn} = z_{knpqij} = z_{pqijkn} = z_{pqknij}$

$\forall(i, j, k, n, p, q = 1,..., N), p > k > i, q \neq n \neq j$, and enforce the symmetric restrictions that

$$v_{ijknpqgh} = v_{knijpqgh} = v_{ijpqkngh} = v_{knpqijgh} = v_{pqijkngh} = v_{pqknijgh} = v_{ijknghpq} = v_{knijghpq} = v_{ijpqghkn} = v_{knpqghij}$$
$$= v_{pqijghkn} = v_{pqknghij} = v_{ijghknpq} = v_{knghijpq} = v_{ijghpqkn} = v_{knghpqij} = v_{pqghijkn} = v_{pqghknij} = v_{ghijknpq} = v_{ghknijpq}$$
$$= v_{ghijpqkn} = v_{ghknpqij} = v_{ghpqijkn} = v_{ghpqknij}$$

$\forall(i, j, k, n, p, q, g, h = 1,..., N), g > p > k > i, h \neq q \neq n \neq j$, too. Then, the level-3 RLT formulation of QAP is given below, where the coefficients d_{ijknpq} and $e_{ijknpqgh}$ found in the objective function are all zero. Notice that Problem RLT3 allows nonzero d_{ijknpq} and $e_{ijknpqgh}$ values, so that it generally handles biquadratic assignment problems.

[RLT3]

$$\begin{aligned} \min \quad & \sum_{i=1}^{N}\sum_{j=1}^{N} b_{ij}x_{ij} + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{N}\sum_{\substack{n=1\\n\neq j}}^{N} c_{ijkn}y_{ijkn} + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{N}\sum_{\substack{n=1\\n\neq j}}^{N}\sum_{\substack{p=1\\p\neq i,k}}^{N}\sum_{\substack{q=1\\q\neq j,n}}^{N} d_{ijknpq}z_{ijknpq} \\ & + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{N}\sum_{\substack{n=1\\n\neq j}}^{N}\sum_{\substack{p=1\\p\neq i,k}}^{N}\sum_{\substack{q=1\\q\neq j,n}}^{N}\sum_{\substack{g=1\\g\neq i,k,p}}^{N}\sum_{\substack{h=1\\h\neq j,n,q}}^{n} e_{ijknpqgh}v_{ijknpqgh} \end{aligned} \tag{6-8a}$$

$$\text{s.t.} \quad \sum_{\substack{g=1\\g\neq i,k,p}}^{N} v_{ijknpqgh} = z_{ijknpq} \qquad i, j, k, n, p, q, h = 1,..., N; h \neq q \neq n \neq j, p \neq k \neq i, \tag{6-8b}$$

$$\sum_{\substack{h=1\\h\neq j,n,q}}^{N} v_{ijknpqgh} = z_{ijknpq} \qquad i, j, k, n, p, q, g = 1,..., N; g \neq p \neq k \neq i, q \neq n \neq j, \tag{6-8c}$$

$$v_{ijknpqgh} = v_{knijpqgh} = v_{ijpqkngh} = v_{knpqijgh} = v_{pqijkngh} = v_{pqknijgh}$$
$$= v_{ijknghpq} = v_{knijghpq} = v_{ijpqghkn} = v_{knpqghij} = v_{pqijghkn} = v_{pqknghij}$$
$$= v_{ijghknpq} = v_{knghijpq} = v_{ijghpqkn} = v_{knghpqij} = v_{pqghijkn} = v_{pqghknij}$$
$$= v_{ghijknpq} = v_{ghknijpq} = v_{ghijpqkn} = v_{ghknpqij} = v_{ghpqijkn} = v_{ghpqknij}$$

$$i, j, k, n, p, q, g, h = 1,..., N; g > p > k > i, h \neq q \neq n \neq j, \qquad (6\text{-}8d)$$

$$v_{ijknpqgh} \geq 0$$

$$i, j, k, n, p, q, g, h = 1,..., N; g \neq p \neq k \neq i, h \neq q \neq n \neq j, \qquad (6\text{-}8e)$$

$$\sum_{\substack{p=1 \\ p \neq i,k}}^{N} z_{ijknpq} = y_{ijkn} \qquad i, j, k, n, q = 1,..., N; q \neq n \neq j, k \neq i, \qquad (6\text{-}8f)$$

$$\sum_{\substack{q=1 \\ q \neq j,n}}^{N} z_{ijknpq} = y_{ijkn} \qquad i, j, k, n, p = 1,..., N; p \neq k \neq i, n \neq j, \qquad (6\text{-}8g)$$

$$z_{ijknpq} = z_{knijpq} = z_{ijpqkn} = z_{knpqij} = z_{pqijkn} = z_{pqknij}$$

$$i, j, k, n, p, q = 1,..., N; p > k > i, q \neq n \neq j, \qquad (6\text{-}8h)$$

$$z_{ijknpq} \geq 0 \qquad i, j, k, n, p, q = 1,..., N; p \neq k \neq i, q \neq n \neq j, \qquad (6\text{-}8i)$$

$$\sum_{\substack{k=1 \\ k \neq i}}^{N} y_{ijkn} = x_{ij} \qquad i, j, n = 1,..., N; n \neq j, \qquad (6\text{-}8j)$$

$$\sum_{\substack{n=1 \\ n \neq j}}^{N} y_{ijkn} = x_{ij} \qquad i, j, k = 1,..., N; k \neq i, \qquad (6\text{-}8k)$$

$$y_{ijkn} = y_{knij} \qquad i, j, k, n = 1,..., N; k > i, n \neq j, \qquad (6\text{-}8l)$$

$$y_{ijkn} \geq 0 \qquad i, j, k, n = 1,..., N; k \neq i, n \neq j, \qquad (6\text{-}8m)$$

$$\boldsymbol{x} \in \boldsymbol{X}, \boldsymbol{x} \text{ binary}. \qquad (6\text{-}8n)$$

Just as one is able to apply the level-2 RLT QAP model to solve the CAP, one can solve the BQAP using the level-3 RLT QAP model, since the additional constraints (6-8b)-(6-8m) in the above formulation are derived completely from the substitutions of $y_{ijkn} = x_{ij}x_{kn}$, $z_{ijknpq} = x_{ij}x_{kn}x_{pq}$ and $v_{ijknpqgh} = x_{ij}x_{kn}x_{pq}x_{gh}$. Therefore, the expansion involving the multiplication of binary variables in the objective functions provides an n-hierarchy of assignment problems that has a direct relation with the n-hierarchy of the RLT models of the QAP.

6.5 The level-3 RLT dual-ascent procedure of the QAP

Just like Problems RLT1 and RLT2, Problem RLT3 is equivalent to the QAP when the binary constraints on $\boldsymbol{x}$ are enforced. While RLT1 implies $y_{ijkn} = x_{ij}x_{kn}$ and RLT2 additionally implies $z_{ijknpq} = x_{ij}x_{kn}x_{pq}$, RLT3 also implies $v_{ijknpqgh} = x_{ij}x_{kn}x_{pq}x_{gh}$. When relaxing the $\boldsymbol{x}$ binary constraints, $\overline{\text{RLT3}}$ gives the tightest lower bounds of all three RLT models, since $\overline{\text{RLT2}}$ ($\overline{\text{RLT1}}$) can be derived from $\overline{\text{RLT3}}$ with no constraints imposed on $\boldsymbol{v}$ (and $\boldsymbol{z}$). Although Problem $\overline{\text{RLT3}}$ provides very sharp bounds, the formulation is considerably larger in size than QAP, $\overline{\text{RLT1}}$ and $\overline{\text{RLT2}}$. It is also highly degenerate, because from all the numerous equality constraints found in Problem $\overline{\text{RLT3}}$, only $2N$ constraints in $\boldsymbol{x} \in \boldsymbol{X}$ have nonzero right-hand-side (RHS) values. The challenge is to extract tight bounds from this formulation without paying a heavy computational price. Fortunately, one need not solve Problem $\overline{\text{RLT3}}$ to optimality, considering the fact that every dual feasible solution provides a lower bound. The strategy is to quickly compute near-optimal dual solutions.

One can obtain a smaller formulation of $\overline{\text{RLT3}}$ via the substitution suggested by constraints (6-8d), (6-8h) and (6-8l) without affecting the bounding strength. Those remaining variables turn out to be $v_{ijknpqgh}\left(g > p > k > i, h \neq q \neq n \neq j\right)$, $z_{ijknpq}\left(p > k > i, q \neq n \neq j\right)$ and $y_{ijkn}\left(k > i, n \neq j\right)$. This makes constraints (6-8d), (6-8h) and (6-8l) unnecessary. Here I do not perform such operations, but instead exploit a block-diagonal structure present within Lagrangian subproblems. Suppose constraints (6-8d), (6-8h) and (6-8l) are dualized into the objective function using some Lagrangian multipliers. Let $\overline{b}_{ij}$, $\overline{c}_{ijkn}$, $\overline{d}_{ijknpq}$ and $\overline{e}_{ijknpqgh}$ denote the objective coefficients associated with x_{ij}, y_{ijkn}, z_{ijknpq} and $v_{ijknpqgh}$ respectively, modified from b_{ij}, c_{ijkn}, d_{ijknpq} and $e_{ijknpqgh}$ in the original objective function of $\overline{\text{RLT3}}$ by dualizing (6-8d), (6-8h) and (6-8l). The modified model $\overline{\text{RLT3M}}$ is presented below.

$\left[\overline{\text{RLT3M}}\right]$

$$\min \quad \left\{ \begin{array}{l} \displaystyle\sum_{i=1}^{N}\sum_{j=1}^{N}\overline{b}_{ij}x_{ij} + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{N}\sum_{\substack{n=1\\n\neq j}}^{N}\overline{c}_{ijkn}y_{ijkn} + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{N}\sum_{\substack{n=1\\n\neq j}}^{N}\sum_{\substack{p=1\\p\neq i,k}}^{N}\sum_{\substack{q=1\\q\neq j,n}}^{N}\overline{d}_{ijknpq}z_{ijknpq} \\ \displaystyle+\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{N}\sum_{\substack{n=1\\n\neq j}}^{N}\sum_{\substack{p=1\\p\neq i,k}}^{N}\sum_{\substack{q=1\\q\neq j,n}}^{N}\sum_{\substack{g=1\\g\neq i,k,p}}^{N}\sum_{\substack{h=1\\h\neq j,n,q}}^{n}\overline{e}_{ijknpqgh}v_{ijknpqgh} \\ \text{s.t.} \\ \text{(6-8b),(6-8c),(6-8e),(6-8f),(6-8g),(6-8i),(6-8j),(6-8k),(6-8m)}, \boldsymbol{x} \in \boldsymbol{X} \end{array} \right\} \quad (6\text{-}9)$$

The binary constraints on $\boldsymbol{x}$ in (6-9) are relaxed temporarily, which will be explained later.

Now Problem $\overline{\text{RLT3M}}$ is ready for decomposition. Adams et al. (2006) proved the following LEMMA.

LEMMA 6-1. Consider any feasible and bounded linear program of the form

$$[\text{LP}] \quad \hat{Z} = \min\left\{\boldsymbol{c}^T\boldsymbol{x} + \boldsymbol{g}^T\boldsymbol{w} : \boldsymbol{Bw} \geq \boldsymbol{d}x_i \text{ for some chosen } \boldsymbol{i}, \boldsymbol{Ax} \geq \boldsymbol{b}\right\}, \tag{6-10}$$

where $\boldsymbol{Bw} \geq \boldsymbol{d}$ and $\boldsymbol{Ax} \geq \boldsymbol{b}$ denote feasible and bounded polyhedral sets, with $\boldsymbol{Ax} \geq \boldsymbol{b}$ enforcing $x_i \geq 0$. Then an optimal solution $(\hat{\boldsymbol{x}}, \hat{\boldsymbol{w}})$ to LP can be obtained by solving

$$\tilde{Z} = \min\left\{(\boldsymbol{c} + \Delta\boldsymbol{e}_i)^T \boldsymbol{x} : \boldsymbol{Ax} \geq \boldsymbol{b}\right\}, \tag{6-11}$$

where

$$\Delta = \min\left\{\boldsymbol{g}^T\boldsymbol{w} : \boldsymbol{Bw} \geq \boldsymbol{d}\right\}, \tag{6-12}$$

and where $\boldsymbol{e}_i$ is the unit column vector having a 1 in position i and zeroes elsewhere. Here, $\hat{\boldsymbol{x}} = \tilde{\boldsymbol{x}}$ and $\hat{\boldsymbol{w}} = \tilde{\boldsymbol{w}}\, \tilde{x}_i$ with $\tilde{\boldsymbol{x}}$ solving (6-11) and $\tilde{\boldsymbol{w}}$ solving (6-12), so that $\hat{Z} = \tilde{Z}$.

Based upon LEMMA 6-1 and the decomposition of $\overline{\text{RLT2}}$, one can decompose $\overline{\text{RLT3M}}$ into a series of assignment problems. Namely, the following THEOREM applies.

THEOREM 6-2. Problem $\overline{\text{RLT3M}}$ (6-9) can be solved by the assignment problem

$$\min \quad \left\{\sum_{i=1}^{N}\sum_{j=1}^{N}\left(\overline{b}_{ij} + \gamma_{ij}\right)x_{ij} : \boldsymbol{x} \in \boldsymbol{X}\right\}, \tag{6-13}$$

where for each (i, j), γ_{ij} is computed as

$$\gamma_{ij} = \min \left\{ \begin{array}{l} \sum_{\substack{k=1 \\ k \neq i}}^{N} \sum_{\substack{n=1 \\ n \neq j}}^{N} \left(\overline{c}_{ijkn} + \eta_{ijkn} \right) y_{ijkn} \\ \text{s.t.} \\ \sum_{\substack{k=1 \\ k \neq i}}^{N} y_{ijkn} = 1, \forall \left(n \neq j \right) \\ \sum_{\substack{n=1 \\ n \neq j}}^{N} y_{ijkn} = 1, \forall \left(k \neq i \right) \\ y_{ijkn} \geq 0, \forall \left(k \neq i, n \neq j \right) \\ k, n = 1, \ldots, N \end{array} \right\}, \tag{6-14}$$

and where for each $\left(i, j, k, n\right)$ with $k \neq i$ and $n \neq j$, η_{ijkn} is computed as

$$\eta_{ijkn} = \min \left\{ \begin{array}{l} \sum_{\substack{p=1 \\ p \neq i,k}}^{N} \sum_{\substack{q=1 \\ q \neq j,n}}^{N} \left(\overline{d}_{ijknpq} + \varphi_{ijknpq} \right) z_{ijknpq} \\ \text{s.t.} \\ \sum_{\substack{p=1 \\ p \neq i,k}}^{N} z_{ijknpq} = 1, \forall \left(q \neq j, n \right) \\ \sum_{\substack{q=1 \\ q \neq j,n}}^{N} z_{ijknpq} = 1, \forall \left(p \neq i, k \right) \\ z_{ijknpq} \geq 0, \forall \left(p \neq i, k; q \neq j, n \right) \\ p, q = 1, \ldots, N \end{array} \right\}, \tag{6-15}$$

and where for each $\left(i, j, k, n, p, q\right)$ with $p \neq k \neq i$ and $q \neq n \neq j$, φ_{ijknpq} is computed as

$$\varphi_{ijknpq} = \min \left\{ \begin{array}{l} \sum_{\substack{g=1 \\ g \neq i,k,p}}^{N} \sum_{\substack{h=1 \\ h \neq j,n,q}}^{N} \bar{e}_{ijknpqgh} v_{ijknpqgh} \\ \text{s.t.} \\ \sum_{\substack{g=1 \\ g \neq i,k,p}}^{N} v_{ijknpqgh} = 1, \forall \left(h \neq j,n,q \right) \\ \sum_{\substack{h=1 \\ h \neq j,n,q}}^{N} v_{ijknpqgh} = 1, \forall \left(g \neq i,k,p \right) \\ v_{ijknpq} \geq 0, \forall \left(g \neq i,k,p; h \neq j,n,q \right) \\ g,h = 1,\ldots,N \end{array} \right\}. \tag{6-16}$$

PROOF.

For any (i,j,k,n,p,q) with $p \neq k \neq i$ and $q \neq n \neq j$, treat the equality constraints (6-8b)-(6-8c) and the nonnegativity constraints (6-8e) of (6-9) as $\boldsymbol{Bw} \geq \boldsymbol{d}x_i$ of (6-10), with x_i of (6-10) represented by variable z_{ijknpq} and $\boldsymbol{w}$ of (6-10) represented by variables $v_{ijknpqgh}$ with $g \neq p \neq k \neq i$ and $h \neq q \neq n \neq j$, and treat the remaining variables and constraints of (6-9) as $\boldsymbol{x}$ and $\boldsymbol{Ax} \geq \boldsymbol{b}$ respectively. Then apply LEMMA 6-1, so that the resulting problem of the form (6-11) contains no $v_{ijknpqgh}$ term for the chosen (i,j,k,n,p,q). Denote Δ of (6-12) as φ_{ijknpq}, so that the objective coefficient of z_{ijknpq} changes from $\bar{d}_{ijknpq}$ to $\bar{d}_{ijknpq} + \varphi_{ijknpq}$. Now, (6-9) becomes

$$\min \left\{ \begin{array}{l} \sum_{i=1}^{N} \sum_{j=1}^{N} \bar{b}_{ij} x_{ij} + \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{\substack{k=1 \\ k \neq i}}^{N} \sum_{\substack{n=1 \\ n \neq j}}^{N} \bar{c}_{ijkn} y_{ijkn} \\ + \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{\substack{k=1 \\ k \neq i}}^{N} \sum_{\substack{n=1 \\ n \neq j}}^{N} \sum_{\substack{p=1 \\ p \neq i,k}}^{N} \sum_{\substack{q=1 \\ q \neq j,n}}^{N} \left(\bar{d}_{ijknpq} + \varphi_{ijknpq} \right) z_{ijknpq} \\ \text{s.t.} \\ \text{(6-8f),(6-8g),(6-8i),(6-8j),(6-8k),(6-8m)}, \boldsymbol{x} \in \boldsymbol{X} \end{array} \right\}. \tag{6-17}$$

The rest of proof is to apply LEMMA 6-1 twice, which was included in the THEOREM proof from the $\overline{\text{RLT2}}$ decomposition by Adams et al. (2006). First, effectively remove all z_{ijknpq} with $p \neq k \neq i$ and $q \neq n \neq j$, where (6-17) becomes

$$\min \left\{ \begin{array}{l} \sum_{i=1}^{N}\sum_{j=1}^{N}\bar{b}_{ij}x_{ij} + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{\substack{k=1 \\ k\neq i}}^{N}\sum_{\substack{n=1 \\ n\neq j}}^{N}\left(\bar{c}_{ijkn}+\eta_{ijkn}\right)y_{ijkn} \\ \\ \text{s.t.} \\ \text{(6-8j),(6-8k),(6-8m)}, \boldsymbol{x}\in\boldsymbol{X} \end{array} \right\}. \tag{6-18}$$

And then remove all y_{ijkn} with $k \neq i$ and $n \neq j$, till the problem is finally reduced to (6-13) only in variables x_{ij}. This completes the proof. □

The optimal (binary) solution to (6-9) can be obtained as follows. Let $\tilde{\boldsymbol{x}}$, $\tilde{\boldsymbol{y}}$, $\tilde{\boldsymbol{z}}$ and $\tilde{\boldsymbol{v}}$ denote the computed optimal (extreme point) solutions to (6-13), (6-14), (6-15) and (6-16), respectively. By LEMMA 6-1, $\left(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}}\right)$ solves (6-18) where

$$\bar{\boldsymbol{x}} = \tilde{\boldsymbol{x}}, \tag{6-19a}$$

and $\bar{y}_{ijkn} = \tilde{y}_{ijkn}\tilde{x}_{ij}$, $\forall\left(i,j,k,n\right)$ with $k \neq i$ and $n \neq j$; (6-19b)

Also by LEMMA 6-1, given that $\left(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}}\right)$ is optimal to (6-18), $\left(\bar{\bar{\boldsymbol{x}}},\bar{\bar{\boldsymbol{y}}},\bar{\bar{\boldsymbol{z}}}\right)$ solves (6-17) where

$$\bar{\bar{\boldsymbol{x}}} = \bar{\boldsymbol{x}}, \tag{6-20a}$$

$$\bar{\bar{\boldsymbol{y}}} = \bar{\boldsymbol{y}}, \tag{6-20b}$$

and $\bar{\bar{z}}_{ijknpq} = \tilde{z}_{ijknpq}\bar{y}_{ijkn}$, $\forall\left(i,j,k,n,p,q\right)$ with $p \neq k \neq i$ and $q \neq n \neq j$; (6-20c)

Again by LEMMA 6-1, given that $\left(\bar{\bar{\boldsymbol{x}}},\bar{\bar{\boldsymbol{y}}},\bar{\bar{\boldsymbol{z}}}\right)$ is optimal to (6-17), $\left(\hat{\boldsymbol{x}},\hat{\boldsymbol{y}},\hat{\boldsymbol{z}},\hat{\boldsymbol{v}}\right)$ solves (6-9), where

$$\hat{\boldsymbol{x}} = \overline{\overline{\boldsymbol{x}}}, \tag{6-21a}$$

$$\hat{\boldsymbol{y}} = \overline{\overline{\boldsymbol{y}}}, \tag{6-21b}$$

$$\hat{\boldsymbol{z}} = \overline{\overline{\boldsymbol{z}}}, \tag{6-21c}$$

and $\hat{v}_{ijknpqgh} = \tilde{v}_{ijknpqgh}\overline{\overline{z}}_{ijknpq}$,

$$\forall (i, j, k, n, p, q, g, h) \text{ with } g \neq p \neq k \neq i \text{ and } h \neq q \neq n \neq j. \tag{6-21d}$$

Altogether, one obtains $(\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}, \hat{\boldsymbol{z}}, \hat{\boldsymbol{v}})$, which is optimal to (6-9), in terms of $(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}, \tilde{\boldsymbol{z}}, \tilde{\boldsymbol{v}})$ as

$$\hat{\boldsymbol{x}}: \ \hat{\boldsymbol{x}} = \tilde{\boldsymbol{x}}, \tag{6-22a}$$

$$\hat{\boldsymbol{y}}: \ \hat{y}_{ijkn} = \tilde{y}_{ijkn}\tilde{x}_{ij}, \ \forall (i, j, k, n) \text{ with } k \neq i \text{ and } n \neq j, \tag{6-22b}$$

$$\hat{\boldsymbol{z}}: \ \hat{z}_{ijknpq} = \tilde{z}_{ijknpq}\tilde{y}_{ijkn}\tilde{x}_{ij}, \ \forall (i, j, k, n, p, q) \text{ with } p \neq k \neq i \text{ and } q \neq n \neq j, \tag{6-22c}$$

and $\hat{\boldsymbol{v}}: \ \hat{v}_{ijknpqgh} = \tilde{v}_{ijknpqgh}\tilde{z}_{ijknpq}\tilde{y}_{ijkn}\tilde{x}_{ij}$,

$$\forall (i, j, k, n, p, q, g, h) \text{ with } g \neq p \neq k \neq i \text{ and } h \neq q \neq n \neq j. \tag{6-22d}$$

Now, since (6-13)-(6-16) are assignment problems, the extreme points are binary so that $\tilde{\boldsymbol{x}}$, $\tilde{\boldsymbol{y}}$, $\tilde{\boldsymbol{z}}$ and $\tilde{\boldsymbol{v}}$ are binary, which makes $(\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}, \hat{\boldsymbol{z}}, \hat{\boldsymbol{v}})$ an optimal binary solution to (6-9).

THEOREM 6-2 and its proof show how to decompose $\overline{\text{RLT3M}}$ into one assignment problem (6-13) of size N, N^2 assignment problems (6-14) of size $N-1$, $N^2(N-1)^2$ assignment problems (6-15) of size $N-2$, and $N^2(N-1)^2(N-2)^2$ assignment problems (6-16) of size $N-3$. This motivates a Lagrangian approach for determining the optimal set of dual multiplier values for constraints (6-8d), (6-8h) and (6-8l), and hence for obtaining the optimal objective value of the continuous relaxation Problem $\overline{\text{RLT3}}$. I present below a dual-ascent procedure, much similar to that employed in Adams et al. (2006) for Problem $\overline{\text{RLT2}}$, which provide a monotonic nondecreasing sequence of lower bounds for the QAP via Problem $\overline{\text{RLT3}}$. Notice that with the symmetric constraints (6-8d) connecting the twenty-four $\boldsymbol{v}$ variables, it potentially leads to extracting as much as possible from the associated cost matrix $\boldsymbol{E}$, thereby increasing the lower bound Z. This observation also applies to the cost matrix $\boldsymbol{D}$ by constraints (6-8h) and to the cost matrix $\boldsymbol{C}$ by constraints (6-8l). Here are the steps.

1. Initialize (6-9) by assigning $\overline{e}_{ijknpqgh} = e_{ijknpqgh} = 0$ for $\forall(i, j, k, n, p, q, g, h)$ with $g \neq p \neq k \neq i$ and $h \neq q \neq n \neq j$, $\overline{d}_{ijknpq} = d_{ijknpq} = 0$ for $\forall(i, j, k, n, p, q)$ with $p \neq k \neq i$ and $q \neq n \neq j$, $\overline{c}_{ijkn} = c_{ijkn}$ for $\forall(i, j, k, n)$ with $k \neq i$ and $n \neq j$, and $\overline{b}_{ij} = b_{ij}$ for $\forall(i, j)$, where $e_{ijknpqgh}$, d_{ijknpq}, c_{ijkn} and b_{ij} are objective coefficients taken from $\overline{\text{RLT3}}$. Set the initial lower bound $Z = 0$. Set the iteration counter to be 0.

2a. For each (i, j), distribute the coefficient $\overline{b}_{ij}$ among the $(N-1)^2$ coefficients $\overline{c}_{ijkn}$ for all $k \neq i$ and $n \neq j$ by increasing each such $\overline{c}_{ijkn}$ by $\overline{b}_{ij}/(N-1)$ and decreasing $\overline{b}_{ij}$ to 0. This is equivalent, for each (i, j), to adding $\overline{b}_{ij}/(N-1)$ times each of the $N-1$ equations $\sum_{n \neq j} y_{ijkn} - x_{ij} = 0$ for all $k \neq i$ found in (6-8k) to the objective of (6-9).

2b. For each (i, j, k, n) with $i \neq j$ and $k \neq n$, distribute the updated coefficient $\overline{c}_{ijkn}$ among the $(N-2)^2$ coefficients $\overline{d}_{ijknpq}$ for all $p \neq i, k$ and $q \neq j, n$ by increasing each such $\overline{d}_{ijknpq}$ by $\overline{c}_{ijkn}/(N-2)$ and decreasing $\overline{c}_{ijkn}$ to 0. This is equivalent, for each (i,j,k,n) with $i \neq j$ and $k \neq n$, to adding $\bar{c}_{ijkn}/(N-2)$ times each of the $N-2$ equations $\sum_{q \neq j,n} z_{ijknpq} - y_{ijkn} = 0$ for all $p \neq i,k$ found in (6-8g) to the objective of (6-9).

2c. For each (i,j,k,n,p,q) with $i \neq j$, $k \neq n$ and $p \neq q$, distribute the updated coefficient $\bar{d}_{ijknpq}$ among the $(N-3)^2$ coefficients $\bar{e}_{ijknpqgh}$ for all $g \neq i,k,p$ and $h \neq j,n,q$ by increasing each such $\bar{e}_{ijknpqgh}$ by $\bar{d}_{ijknpq}/(N-3)$ and decreasing $\bar{d}_{ijknpq}$ to 0. This is equivalent, for each (i,j,k,n,p,q) with $i \neq j$, $k \neq n$ and $p \neq q$, to adding $\bar{d}_{ijknpq}/(N-3)$ times each of the $N-3$ equations $\sum_{h \neq j,n,q} v_{ijknpqgh} - z_{ijknpq} = 0$ for all $g \neq i,k,p$ found in (6-8c) to the objective of (6-9).

3. Use THEOREM 6-2 to sequentially solve (6-9) as $N^2(N-1)^2(N-2)^2 + N^2(N-1)^2 + N^2 + 1$ assignment problems.

3a. Solve $N^2(N-1)^2(N-2)^2$ assignment problem (6-16) of size $N-3$ to obtain $\tilde{\boldsymbol{v}}$ and the value φ_{ijknpq} as follows. Sequentially consider all (i,j,k,n,p,q) with $p \neq k \neq i$ and $q \neq n \neq j$, beginning with those (i,j,k,n,p,q) for which $\bar{d}_{ijknpq}$ prior to step 2c was 0. For a selected (i,j,k,n,p,q), change the coefficient $\bar{e}_{ijknpqgh}$ for each $g \neq i,k,p$ and $h \neq j,n,q$ to a percentage of the sum of $\bar{e}_{ijknpqgh}$, $\bar{e}_{knijpqgh}$, $\bar{e}_{ijpqkngh}$, $\bar{e}_{knpqijgh}$, $\bar{e}_{pqijkngh}$, $\bar{e}_{pqknijgh}$, $\bar{e}_{ijknghpq}$, $\bar{e}_{knijghpq}$, $\bar{e}_{ijpqghkn}$, $\bar{e}_{knpqghij}$, $\bar{e}_{pqijghkn}$, $\bar{e}_{pqknghij}$, $\bar{e}_{ijghknpq}$, $\bar{e}_{knghijpq}$, $\bar{e}_{ijghpqkn}$, $\bar{e}_{knghpqij}$, $\bar{e}_{pqghijkn}$, $\bar{e}_{pqghknij}$, $\bar{e}_{ghijknpq}$, $\bar{e}_{ghknijpq}$, $\bar{e}_{ghijpqkn}$, $\bar{e}_{ghknpqij}$, $\bar{e}_{ghpqijkn}$, and $\bar{e}_{ghpqknij}$, and equally adjust the latter twenty-three values so that the sum stays constant. Upon solving this assignment problem, place the corresponding equality constraints (6-8b) and (6-8c) into the objective function with the optimal dual multipliers, effectively readjusting the $\overline{e}_{ijknpqgh}$ values for $g \neq i,k,p$ and $h \neq j,n,q$ and increasing $\overline{d}_{ijknpq}$ by φ_{ijknpq}. Proceed through all such (i,j,k,n,p,q) indices where $p \neq k \neq i$ and $q \neq n \neq j$.

3b. Solve $N^2(N-1)^2$ assignment problem (6-15) of size $N-2$ to obtain $\tilde{\boldsymbol{z}}$ and the value η_{ijkn} as follows. Sequentially consider all (i,j,k,n) with $k \neq i$ and $n \neq j$, beginning with those (i,j,k,n) for which $\overline{c}_{ijkn}$ prior to step 2b was 0. For a selected (i,j,k,n), change the coefficient $\overline{d}_{ijknpq}$ for each $p \neq i,k$ and $q \neq j,n$ to a percentage of the sum of $\overline{d}_{ijknpq}$, $\overline{d}_{knijpq}$, $\overline{d}_{ijpqkn}$, $\overline{d}_{knpqij}$, $\overline{d}_{pqijkn}$, and $\overline{d}_{pqknij}$, and equally adjust the latter five values so that the sum stays constant. Upon solving this assignment problem, place the corresponding equality constraints (6-8f) and (6-8g) into the objective function with the optimal dual multipliers, effectively readjusting the $\overline{d}_{ijknpq}$ values for $p \neq i,k$ and $q \neq j,n$ and increasing $\overline{c}_{ijkn}$ by η_{ijkn}. Proceed through all such (i,j,k,n) indices where $k \neq i$ and $n \neq j$.

3c. Solve N^2 assignment problem (6-14) of size $N-1$ to obtain $\tilde{\boldsymbol{y}}$ and the value γ_{ij} as follows. Sequentially consider all (i,j), beginning with those (i,j) for which $\overline{b}_{ij}$ prior to step 2a was 0. For a selected (i,j), change the coefficient $\overline{c}_{ijkn}$ for each $k \neq i$ and $n \neq j$ to a percentage of the sum of $\overline{c}_{ijkn}$ and $\overline{c}_{knij}$, and then adjust $\overline{c}_{knij}$ so that the sum stays constant. Upon solving this assignment problem, place the corresponding equality constraints (6-8j) and (6-8k) into the objective function with the optimal dual multipliers, effectively readjusting the $\overline{c}_{ijkn}$ values for $k \neq i$ and $n \neq j$ and increasing $\overline{b}_{ij}$ by γ_{ij}. Proceed through all such (i, j) indices.

3d. Solve the assignment problem (6-13) of size N to obtain $\tilde{\boldsymbol{x}}$. Upon doing so, place the equality constraints of $\boldsymbol{X}$ into the objective function with the optimal dual multipliers, adjusting the value of $\overline{b}_{ij}$ and the lower bound Z. Here, Z is increased by the nonnegative objective value to the minimization problem of (6-13). Proceed to step 4.

4. If the binary optimal solution $(\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}, \hat{\boldsymbol{z}}, \hat{\boldsymbol{v}})$, computed as (6-22), to (6-9) is feasible to RLT3, i.e., if it satisfies (6-8d), (6-8h) and (6-8l), stop with $(\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}, \hat{\boldsymbol{z}}, \hat{\boldsymbol{v}})$ optimal to problem QAP. If it is not feasible to RLT3, stop if some predetermined number of iterations has been performed. Otherwise, increase the iteration counter by 1 and return to step 2a.

The dual-ascent procedure will produce a nondecreasing sequence of lower bounds since Step 1 is input with all variables having nonnegative reduced costs.

6.6 Computational results of the level-3 RLT lower bound calculation

Several years ago Professor Hahn coded in FORTRAN a preliminary dual-ascent algorithm that calculates QAP level-3 RLT root lower bounds. To demonstrate the potential of level-3 RLT, I tested this code and performed a series of experiments using benchmark QAP instances. Table 6-1 compares the performance of the QAP level-3

RLT algorithm with the best lower bound values achieved for these instances by any other methods. Data sets for these problems may be found on the QAPLIB website (http://www.seas.upenn.edu/qaplib/). For each test problem, Table 6-1 presents the level-3 RLT lower bound achieved after 700 iterations of the algorithm. The Optimum column gives the optimal value. The Lower Bound column lists the level-3 RLT lower bound for each QAP instance. The Runtime column shows the CPU seconds normalized to a Dell 7150 machine on a single 733MHz CPU. The Best Previous and Method columns provide the previous best lower bounds and its achieving algorithm from Table 1 of Loiola et al. (2006).

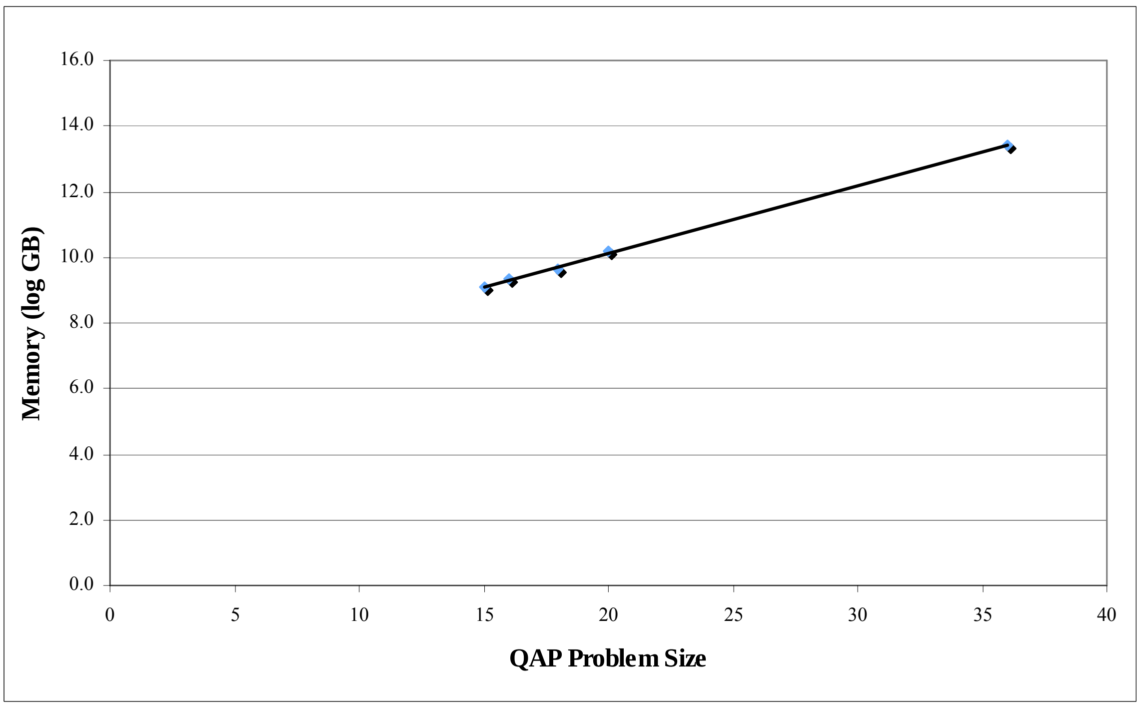
Table 6-1. The QAP level-3 RLT lower bounds.

Instance	Optimum	Lower Bound	Runtime	Best Previous	Method
Nug12	578	577.15*	1,468	578	RLT2
Nug15	1,150	1,149.74*	16,671	1,150	RLT2
Nug18	1,930	1,930**	86,951	1,905	RLT2
Nug20	2,570	2,569.19*	304,274	2,508+	RLT2
Had16	3,720	3,718.11*	~15,000	3,672	RLT2
Had18	5,358	5,357.67*	44,680	5,299	RLT2
Had20	6,922	6,919.1	48,020	6,811	RLT2
Rou15	354,210	354,210**	951	350,207	RLT2
Rou20	725,520	725,314.4	252,282	695,123	RLT2
Tai20a	703,482	703,482**	254,432	671,685	RLT2

* Optimum verified by the level-3 RLT lower bound code.
** Problem solved exactly by the level-3 RLT lower bound code.
\+ Recently corrected result by the level-2 RLT lower bound code.

As mentioned before, the number of variables grows dramatically with RLT level. $\overline{\text{RLT2}}$ code already run into memory limitations for of-the-current generation of computers for problem instances larger than $N = 36$. Those limitations have made it difficult, if not impossible to calculate level-3 RLT lower bounds for problem instances larger than $N = 20$, though $\overline{\text{RLT3}}$ code has shown even more promise for reducing the number of nodes that must be considered for providing optimality. Figure 6-1 below demonstrates the growth in random access memory (RAM) with problem instance size, required for level-3 root lower bound calculations. The linear extrapolation is based on data from the lower bound experiments on four Nugent instances reported in Table 6-1.

Figure 6-1. Memory required for level-3 RLT lower bound calculations.



6.7 Conclusion

This chapter reports the level-3 reformulation-linearization technique (RLT) formulation of the quadratic assignment problem (QAP) and its preliminary lower bound calculations. By extended definitions of the cubic and biquadratic assignment problems, I am able to establish a hierarchy of zero-one assignment problems parallel to the RLT hierarchy of the QAP. Thus, a new theoretical way of connecting the QAP and its related problems is established through their solution methods.

RLT techniques, while showing great promise, have to date received little investigation in terms of practical application. My purpose in this chapter is to show that practical means can be devised to make these techniques useful, not only for solving the QAP, but for solving large classes of similarly difficult combinatorial optimization problems.

7. Conclusion

7.1 Summary of results

It is hoped that this dissertation contributes to the understanding of the quadratic assignment problem (QAP) and its extensions. Solution methodologies studied in this work include the reformulation-linearization technique (RLT), Lagrangian dual procedure, and branch-and-bound enumeration. The methods devised herein effectively exploit the mathematical structure found within the RLT formulations. This consists of both theoretical and computational studies, including specially designed Lagrangian dual procedures that take advantage of the block-diagonal structures, and tradeoffs between linearization size and strength.

Computational experiments were conducted by either implementing or extending the usefulness of existing algorithmic tools to solve application problems. The chief contribution of this dissertation is the theoretical, algorithmic and applicable understanding of quadratic assignment and related problems, so as to engender interests in these computationally complex problems.

I address herein the inherent relationship on the 3-dimensional assignment problem (3AP) to the quadratic assignment problem (QAP) and the quadratic 3-dimensional assignment problem (Q3AP), so as to enhance the theoretical connection of the QAP and its related assignment problems. The key is the method of solving a 3AP of size N as a QAP of size $2N$.

I also compare the lower bounds given by three different level-1 RLT based models of the generalized quadratic assignment problem (GQAP); hence improve the understanding of its RLT form. Lagrangian relaxations of the level-1 RLT GQAP model may lead to potential improvement in lower bounds.

In order to solve new application problems arises in facility evacuation and crossdock design. I develop for the first time the theoretical basis for an innovative assignment problem, the generalized quadratic 3-dimensional assignment problem (GQ3AP). I also outline the steps for an algorithm to calculate the GQ3AP lower bounds and a branch-and-bound algorithm for solving the GQ3AP exactly. I then test Professor Hahn's implementation of the two algorithms. While the runtimes of the branch-and-bound algorithm are exponential in the problem size, the effectiveness of the algorithm for achieving good solutions early in branch-and-bound enumeration has been shown.

Towards the end of dissertation, I demonstrate that the hierarchy of the quadratic, cubic and biquadratic assignment is directly related to the RLT hierarchy of the QAP. I also develop for the first time the level-3 RLT model of the QAP. Using code provided by Professor Hahn, I present that the level-3 RLT QAP model is capable of superior lower bounds. Though limited by the requirement for RAM, the level-3 model demonstrates the tightest lower bounds ever for QAP instances with size $N \leq 20$.

7.2 Future research

There are a number of open questions and future directions generated by this study.

The first question concerns the improvement to the algorithms. The growth in runtime with the problem size is a severe limitation to the usefulness of exact solution methods. The QAP and its related problems GQAP, Q3AP, GQ3AP are very difficult combinatorial optimization problems. The quality of lower bounds is a critical factor in determining the runtime of the branch-and-bound methods. The research team at the University of Pennsylvania plans to investigate a number of methods for improving the tightness of the subproblem lower bounds and for improving the quality of the branching strategy.

Another question concerns the development of heuristic approaches. Heuristics have been designed to address hard combinatorial problems. The quality of the solutions is generally very good and often optimal. These methods are also the best way to provide feasible solutions in large industrial problems with tight constraints. Moreover, such solutions become good starting upper bounds for exact methods, thus reducing branch-and-bound runtimes. Specifically, the team plans to develop a heuristic method for solving the generalized quadratic 3-dimensional assignment problem, which has applications in facility design and the optimization of crossdock door assignment.

Final thoughts concern the design of specialized computers that combine large amounts of random access memory (RAM) with databus controls that assure fast transfer of data between disk and RAM. Also, analytical efforts that can possibly restructure the dual-ascent process may mitigate the requirement for the entire variable space being available in RAM in the process of calculating lower bounds.

Appendix A. Abbreviations and Notations

3AP	3-dimensional assignment problem
A3AP	Axial 3-dimensional assignment problem
BQAP	Biquadratic assignment problem
CAP	Cubic assignment problem
CDAP	Crossdock door assignment problem
G3AP	Generalized 3-dimensional assignment problem
GAP	Generalized assignment problem
GLB	Gilmore-Lawler bound
GQ3AP	Generalized quadratic 3-dimensional assignment problem
GQAP	Generalized quadratic assignment problem
LAP	Linear assignment problem
LP	Linear programming
LR	Lagrangian relaxation
MAP	Multidimensional assignment problem
MFFLP	Multi-floor facility layout problem
MILP	Mixed-integer linear programming
MSAP	Multi-story space assignment problem
P3AP	Planar 3-dimensional assignment problem
Q3AP	Quadratic 3-dimensional assignment problem
QAP	Quadratic assignment problem

RAM	Random access memory
RHS	Right-hand-side
RLT	Reformulation-linearization technique
SDP	Semidefinite programming

Appendix B. Formulations of Quadratic Assignment and Related Problems

Define sets

$$\boldsymbol{X} \equiv \left\{ \boldsymbol{x} \geq \boldsymbol{0} : \sum_{i=1}^{N} x_{ij} = 1, \forall \left(j = 1, \ldots, N \right); \sum_{j=1}^{N} x_{ij} = 1, \forall \left(i = 1, \ldots, N \right) \right\},$$

$$\boldsymbol{U} \equiv \left\{ \boldsymbol{u} \geq \boldsymbol{0} : \sum_{i=1}^{M} s_{ij} u_{ij} \leq S_j, \forall \left(j = 1, \ldots, N \right); \sum_{j=1}^{N} u_{ij} = 1, \forall \left(i = 1, \ldots, M \right) \right\},$$

$$\boldsymbol{W} \equiv \left\{ \boldsymbol{w} \geq \boldsymbol{0} : \sum_{i=1}^{M} t_{ip} w_{ip} \leq T_p, \forall \left(p = 1, \ldots, P \right); \sum_{p=1}^{P} w_{ip} = 1, \forall \left(i = 1, \ldots, M \right) \right\}.$$

The formulations of assignment problems covered in this paper can be written as below.

1. The linear assignment problem (LAP)

$$\min \left\{ \begin{array}{l} \displaystyle\sum_{i=1}^{N} \sum_{j=1}^{N} b_{ij} x_{ij} \\ : \boldsymbol{x} \in \boldsymbol{X}; \boldsymbol{x} \text{ binary} \end{array} \right\} \tag{B1}$$

2. The quadratic assignment problem (QAP)

$$\min \left\{ \begin{array}{l} \displaystyle\sum_{i=1}^{N} \sum_{j=1}^{N} b_{ij} x_{ij} + \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{\substack{k=1 \\ k \neq i}}^{N} \sum_{\substack{n=1 \\ n \neq j}}^{N} c_{ijkn} x_{ij} x_{kn} \\ : \boldsymbol{x} \in \boldsymbol{X}; \boldsymbol{x} \text{ binary} \end{array} \right\} \tag{B2}$$

3. The cubic assignment problem (CAP)

$$\min \left\{ \begin{array}{l} \sum_{i=1}^{N}\sum_{j=1}^{N} b_{ij}x_{ij} + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{N}\sum_{\substack{n=1\\n\neq j}}^{N} c_{ijkn}x_{ij}x_{kn} + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{N}\sum_{\substack{n=1\\n\neq j}}^{N}\sum_{\substack{p=1\\p\neq i,k}}^{N}\sum_{\substack{q=1\\q\neq j,n}}^{N} d_{ijknpq}x_{ij}x_{kn}x_{pq} \\ : \boldsymbol{x} \in \boldsymbol{X}; \boldsymbol{x} \text{ binary} \end{array} \right\} \quad \text{(B3)}$$

4. The biquadratic assignment problem (BQAP)

$$\min \left\{ \begin{array}{l} \sum_{i=1}^{N}\sum_{j=1}^{N} b_{ij}x_{ij} + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{N}\sum_{\substack{n=1\\n\neq j}}^{N} c_{ijkn}x_{ij}x_{kn} + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{N}\sum_{\substack{n=1\\n\neq j}}^{N}\sum_{\substack{p=1\\p\neq i,k}}^{N}\sum_{\substack{q=1\\q\neq j,n}}^{N} d_{ijknpq}x_{ij}x_{kn}x_{pq} \\ +\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{N}\sum_{\substack{n=1\\n\neq j}}^{N}\sum_{\substack{p=1\\p\neq i,k}}^{N}\sum_{\substack{q=1\\q\neq j,n}}^{N}\sum_{\substack{g=1\\g\neq i,k,p}}^{N}\sum_{\substack{h=1\\h\neq j,n,q}}^{N} e_{ijknpqgh}x_{ij}x_{kn}x_{pq}x_{gh} \\ : \boldsymbol{x} \in \boldsymbol{X}; \boldsymbol{x} \text{ binary} \end{array} \right\} \quad \text{(B4)}$$

5. The generalized (linear) assignment problem (GAP)

$$\min \left\{ \begin{array}{l} \sum_{i=1}^{M}\sum_{j=1}^{N} b_{ij}x_{ij} \\ : \boldsymbol{x} \in \boldsymbol{U}; \boldsymbol{x} \text{ binary} \end{array} \right\} \quad \text{(B5)}$$

6. The generalized quadratic assignment problem (GQAP)

$$\min \left\{ \begin{array}{l} \sum_{i=1}^{M}\sum_{j=1}^{N} b_{ij}x_{ij} + \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{M}\sum_{n=1}^{N} c_{ijkn}x_{ij}x_{kn} \\ : \boldsymbol{x} \in \boldsymbol{U}; \boldsymbol{x} \text{ binary} \end{array} \right\} \quad \text{(B6)}$$

7. The generalized cubic assignment problem (GCAP)

$$\min \left\{ \begin{array}{l} \sum_{i=1}^{M}\sum_{j=1}^{N} b_{ij}x_{ij} + \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{M}\sum_{n=1}^{N} c_{ijkn}x_{ij}x_{kn} + \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{M}\sum_{n=1}^{N}\sum_{\substack{p=1\\p\neq i,k}}^{M}\sum_{q=1}^{N} d_{ijknpq}x_{ij}x_{kn}x_{pq} \\ : \boldsymbol{x}\in\boldsymbol{U};\ \boldsymbol{x} \text{ binary} \end{array} \right\} \quad \text{(B7)}$$

8. The generalized biquadratic assignment problem (GBQAP)

$$\min \left\{ \begin{array}{l} \sum_{i=1}^{M}\sum_{j=1}^{N} b_{ij}x_{ij} + \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{M}\sum_{n=1}^{N} c_{ijkn}x_{ij}x_{kn} + \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{M}\sum_{n=1}^{N}\sum_{\substack{p=1\\p\neq i,k}}^{M}\sum_{q=1}^{N} d_{ijknpq}x_{ij}x_{kn}x_{pq} \\ +\sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{M}\sum_{n=1}^{N}\sum_{\substack{p=1\\p\neq i,k}}^{M}\sum_{q=1}^{N}\sum_{\substack{g=1\\g\neq i,k,m}}^{M}\sum_{h=1}^{N} e_{ijknpqgh}x_{ij}x_{kn}x_{pq}x_{gh} \\ : \boldsymbol{x}\in\boldsymbol{U};\ \boldsymbol{x} \text{ binary} \end{array} \right\} \quad \text{(B8)}$$

9. The 3-dimensional assignment problem (3AP)

$$\min \left\{ \begin{array}{l} \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{p=1}^{N} b_{ijp}u_{ij}w_{ip} \\ : \boldsymbol{u},\boldsymbol{w}\in\boldsymbol{X};\ \boldsymbol{u},\boldsymbol{w} \text{ binary} \end{array} \right\} \quad \text{(B9)}$$

10. The quadratic 3-dimensional assignment problem (Q3AP)

$$\min \left\{ \begin{array}{l} \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{p=1}^{N} b_{ijp}u_{ij}w_{ip} + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{p=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{N}\sum_{\substack{n=1\\n\neq j}}^{N}\sum_{\substack{q=1\\q\neq p}}^{N} c_{ijpknq}u_{ij}u_{kn}w_{ip}w_{kq} \\ : \boldsymbol{u},\boldsymbol{w}\in\boldsymbol{X};\ \boldsymbol{u},\boldsymbol{w} \text{ binary} \end{array} \right\} \quad \text{(B10)}$$

11. The cubic 3-dimensional assignment problem (C3AP)

$$\min \left\{ \begin{array}{l} \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{p=1}^{N} b_{ijp}u_{ij}w_{ip} + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{p=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{N}\sum_{\substack{n=1\\n\neq j}}^{N}\sum_{\substack{q=1\\q\neq p}}^{N} c_{ijpknq}u_{ij}u_{kn}w_{ip}w_{kq} \\ +\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{p=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{N}\sum_{\substack{n=1\\n\neq j}}^{N}\sum_{\substack{q=1\\q\neq p}}^{N}\sum_{\substack{f=1\\f\neq i,k}}^{N}\sum_{\substack{g=1\\g\neq j,n}}^{N}\sum_{\substack{h=1\\h\neq p,q}}^{N} d_{ijpknqfgh}u_{ij}u_{kn}u_{fg}w_{ip}w_{kq}w_{fh} \\ \\ :\boldsymbol{u},\boldsymbol{w}\in\boldsymbol{X};\boldsymbol{u},\boldsymbol{w}\text{ binary} \end{array} \right\} \tag{B11}$$

12. The biquadratic 3-dimensional assignment problem (BQ3AP)

$$\min \left\{ \begin{array}{l} \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{p=1}^{N} b_{ijp}u_{ij}w_{ip} + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{p=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{N}\sum_{\substack{n=1\\n\neq j}}^{N}\sum_{\substack{q=1\\q\neq p}}^{N} c_{ijpklq}u_{ij}u_{kn}w_{ip}w_{kq} \\ +\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{p=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{N}\sum_{\substack{n=1\\n\neq j}}^{N}\sum_{\substack{q=1\\q\neq p}}^{N}\sum_{\substack{f=1\\f\neq i,k}}^{N}\sum_{\substack{g=1\\g\neq j,n}}^{N}\sum_{\substack{h=1\\h\neq p,q}}^{N} d_{ijpklqfgh}u_{ij}u_{kn}u_{fg}w_{ip}w_{kq}w_{fh} \\ +\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{p=1}^{N}\sum_{\substack{k=1\\k\neq i}}^{N}\sum_{\substack{n=1\\n\neq j}}^{N}\sum_{\substack{q=1\\q\neq p}}^{N}\sum_{\substack{f=1\\f\neq i,k}}^{N}\sum_{\substack{g=1\\g\neq j,n}}^{N}\sum_{\substack{h=1\\h\neq p,q}}^{N}\sum_{\substack{l=1\\l\neq i,k,f}}^{N}\sum_{\substack{r=1\\r\neq j,n,g}}^{N}\sum_{\substack{v=1\\v\neq p,q,h}}^{N} e_{ijpknqfghlrv}u_{ij}u_{kn}u_{fg}u_{lr}w_{ip}w_{kq}w_{fh}w_{lv} \\ :\boldsymbol{u},\boldsymbol{w}\in\boldsymbol{X};\boldsymbol{u},\boldsymbol{w}\text{ binary} \end{array} \right\}$$

(B12)

13. The generalized 3-dimensional assignment problem (G3AP)

$$\min \left\{ \begin{array}{l} \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P} b_{ijp}u_{ij}w_{ip} \\ :\boldsymbol{u}\in\boldsymbol{U},\boldsymbol{w}\in\boldsymbol{W};\boldsymbol{u},\boldsymbol{w}\text{ binary} \end{array} \right\} \tag{B13}$$

14. The generalized quadratic 3-dimensional assignment problem (GQ3AP)

$$\min \left\{ \begin{array}{l} \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P} b_{ijp} u_{ij} w_{ip} + \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P}\sum_{\substack{k=1\\k\neq i}}^{M}\sum_{n=1}^{N}\sum_{q=1}^{P} c_{ijpknq} u_{ij} u_{kn} w_{ip} w_{kq} \\ : \boldsymbol{u} \in \boldsymbol{U}, \boldsymbol{v} \in \boldsymbol{W}; \boldsymbol{u}, \boldsymbol{v} \text{ binary} \end{array} \right\} \tag{B14}$$

15. The generalized cubic 3-dimensional assignment problem (GC3AP)

$$\min \left\{ \begin{array}{l} \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P} b_{ijp} u_{ij} w_{ip} + \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P}\sum_{\substack{k=1\\k\neq i}}^{M}\sum_{n=1}^{N}\sum_{q=1}^{P} c_{ijpknq} u_{ij} u_{kn} w_{ip} w_{kq} \\ + \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P}\sum_{\substack{k=1\\k\neq i}}^{M}\sum_{n=1}^{N}\sum_{q=1}^{P}\sum_{\substack{f=1\\f\neq i,k}}^{M}\sum_{g=1}^{N}\sum_{h=1}^{P} d_{ijpknqfgh} u_{ij} u_{kn} u_{fg} w_{ip} w_{kq} w_{fh} \\ : \boldsymbol{u} \in \boldsymbol{U}, \boldsymbol{w} \in \boldsymbol{W}; \boldsymbol{u}, \boldsymbol{w} \text{ binary} \end{array} \right\} \tag{B15}$$

16. The generalized biquadratic 3-dimensional assignment problem (GBQ3AP)

$$\min \left\{ \begin{array}{l} \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P} b_{ijp} u_{ij} w_{ip} + \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P}\sum_{\substack{k=1\\k\neq i}}^{M}\sum_{n=1}^{N}\sum_{q=1}^{P} c_{ijpknq} u_{ij} u_{kn} w_{ip} w_{kq} \\ + \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P}\sum_{\substack{k=1\\k\neq i}}^{M}\sum_{n=1}^{N}\sum_{q=1}^{P}\sum_{\substack{f=1\\f\neq i,k}}^{M}\sum_{g=1}^{N}\sum_{h=1}^{P} d_{ijpknqfgh} u_{ij} u_{kn} u_{fg} w_{ip} w_{kq} w_{fh} \\ + \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{p=1}^{P}\sum_{\substack{k=1\\k\neq i}}^{M}\sum_{n=1}^{N}\sum_{q=1}^{P}\sum_{\substack{f=1\\f\neq i,k}}^{M}\sum_{g=1}^{N}\sum_{h=1}^{P}\sum_{\substack{l=1\\l\neq i,k,f}}^{M}\sum_{r=1}^{N}\sum_{v=1}^{P} e_{ijpknqfghlrv} u_{ij} u_{kn} u_{fg} u_{lr} w_{ip} w_{kq} w_{fh} w_{lv} \\ : \boldsymbol{u} \in \boldsymbol{U}, \boldsymbol{w} \in \boldsymbol{W}; \boldsymbol{u}, \boldsymbol{w} \text{ binary} \end{array} \right\}$$

(B16)

Bibliography

Adams, W.P., and T.A. Johnson, 1994, Improved linear programming-based lower bounds for the quadratic assignment problem, in: Pardalos, P.M., and H. Wolkowicz (Eds.), Quadratic Assignment and Related Problems, DIMACS Series in Discrete Mathematics and Theoretical Computer Science, Vol. **16,** AMS, Providence, RI, 43-75.

Adams, W.P., and H.D. Sherali, 1986, A tight linearization and an algorithm for zero-one quadratic programming problems, Management Science **32**, 1274-1290.

Adams, W.P., and H.D. Sherali, 1990, Linearization strategies for a class of zero-one mixed integer programming problems, Operations Research **38**, 217-226.

Adams, W.P., M. Guignard, P.M. Hahn, and W.L. Hightower, 2007, A level-2 reformulation-linearization technique bound for the quadratic assignment problem, European Journal of Operational Research **180**, 983-996.

Anstreicher, K.M., and N.W. Brixius, 2001, A new bound for the quadratic assignment problem based on convex quadratic programming, Mathematical Programming **89**, 341-357.

Anstreicher, K.M., 2003, Recent advances in the solution of quadratic assignment problems, Mathematical Programming **97**, 27-42.

Appa, G., D. Magos, and I. Mourtos, 2006, On multi-index assignment polytopes, Linear Algebra and its Applications **416**, 224-241.

Balas, E., and M.J. Saltzman, 1991, An algorithm for the three-index assignment problem, Operations Research **39**, 150-161.

Barcia, P., and K. Jörnsten, 1990, Improved Lagrangean decomposition: an application to the generalized assignment problem, European Journal of Operational Research **46**, 84-92.

Bartholdi, J.J., and K.R. Gue, 2000, Reducing labor costs in an LTL crossdocking terminal, Operations Research **48**, 823-832.

Bartholdi, J.J., and K.R. Gue, 2004, The best shape for a crossdock, Transportation Science **38**, 235-244.

Billionnet, A., and S. Elloumi, 1995, An algorithm for finding the k-best allocations of a tree structured program, Journal of Parallel and Distributed Computing **26**, 225-232.

Billionnet, A., and S. Elloumi, 2001, Best reduction of the quadratic semi-assignment problem, Discrete Applied Mathematics **109**, 197-213.

Burer, S., and D. Vandenbussche, 2006, Solving lift-and-project relaxations of binary integer programs, SIAM Journal on Optimization **16**, 726-750.

Burkard, R.E., and R. Rudolf, 1993, Computational investigations on 3-dimensional axial assignment problems, Belgian Journal of Operations Research **32**, 85-98.

Burkard, R.E., E. Çela, and B. Klinz, 1994, On the biquadratic assignment problem, in: Pardalos, P.M., and H. Wolkowicz (Eds.), Quadratic Assignment and Related Problems, DIMACS Series in Discrete Mathematics and Theoretical Computer Science, Vol. **16,** AMS, Providence, RI, 117-146.

Burkard, R.E., B. Klinz, and R. Rudolf, 1996, Perspectives of Monge properties in optimization, Discrete Applied Mathematics **70**, 95-161.

Burkard, R.E., R. Rudolf, and G.J. Woeginger, 1996, Three dimensional axial assignment problems with decomposable cost coefficients, Discrete Applied Mathematics **65**, 123-169.

Burkard, R.E., E. Çela, P.M. Pardalos, and L.S. Pitsoulis, 1998, The quadratic assignment problem, in: Du, D.-Z., and P.M. Pardalos (Eds.), Handbook of Combinatorial Optimization, Vol. **3**, Kluwer Academic Publishers, Dordrecht, 241-337.

Burkard, R.E., and E. Çela, 1999, Linear assignment problems and extensions, in: Du, D.-Z., and P.M. Pardalos (Eds.), Handbook of Combinatorial Optimization, Supplement Vol. **A**, Kluwer Academic Publishers, Dordrecht, 75-149.

Burkard, R.E., 2002, Selected topics on assignment problems, Discrete Applied Mathematics **123**, 257-302.

Cattrysse, D.G., and L.N. Van Wassenhove, 1992, A survey of algorithms for the generalized assignment problem, European Journal of Operational Research **60**, 260-272.

Çela, E., 1998, The Quadratic Assignment Problem: Theory and Algorithms, Kluwer Academic Publishers, Dordrecht, The Netherlands.

Cordeau, J.-F., M. Gaudioso, G. Laporte, and L. Moccia, 2006, A memetic heuristic for the generalized assignment problem, INFORMS Journal on Computing **18**, 433-443.

Cordeau, J.-F., M. Gaudioso, G. Laporte, and L. Moccia, 2007, The service allocation problem at the Gioia Tauro maritime terminal, European Journal of Operational Research **176**, 1167-1184.

Drezner, Z., P.M. Hahn, and É.D. Taillard, 2005, Recent advances for the quadratic assignment problem with special emphasis on instances that are difficult for meta-heuristic methods, Annals of Operations Research **139**, 65-94.

Edwards, C.S., 1980, A branch and bound algorithm for the Koopmans-Beckmann quadratic assignment problem, Mathematical Programming Study **13**, 35-52.

Elloumi, S., F. Roupin, and E. Soutif, 2003, Comparison of different lower bounds for the constrained module allocation problem, Technical Report, TR CEDRIC, No.473.

Euler, R., and H. Le Verge, 1996, Time-tables, polyhedra and the greedy algorithm, Discrete Applied Mathematics **65**, 207-221.

Finke, G., R.E. Burkard, and F. Rendl, 1987, Quadratic assignment problems, Annals of Discrete Mathematics **31**, 61-82.

Fisher, M.L., R. Jaikumar, and L.N. Van Wassenhove, 1986, A multiplier adjustment method for the generalized assignment problem, Management Science **32**, 1095-1103.

Frieze, A.M., 1983, Complexity of a 3-dimensional assignment problem, European Journal of Operational Research **13**, 161-164.

Geoffrion, A.M., 1974, Lagrangean relaxation for integer programming, Mathematical Programming Study **2**, 82-114.

Geoffrion, A.M., and R. McBride, 1978, Lagrangean relaxation applied to capacitated facility location problems, AIIE Transactions **10**, 40-47.

Gilmore, P.C., 1962, Optimal and suboptimal algorithms for the quadratic assignment problem, SIAM Journal on Applied Mathematics **10**, 305-313.

Graham, A., 1981, Kronecker Products and Matrix Calculus with Applications, Ellis Horwood Limited, Chichester, 21-23.

Grant, T.L., 1989, An evaluation and analysis of the resolvent sequence method for solving the quadratic assignment problem, Master Thesis, Department of Systems Engineering, University of Pennsylvania, Philadelphia, PA 19104.

Guignard, M., and M.B. Rosenwein, 1989, An improved dual based algorithm for the generalized assignment problem, Operations Research **37**, 658-663.

Guignard, M., 2003, Lagrangean relaxation, Top, Vol. **11**, 151-228.

Guignard, M., 2006, Note on the solution of the Lagrangean subproblems for an RLT-1 form of the GQAP, Research Report 06-06-01, Department of Operations and Information Management, The Wharton School, University of Pennsylvania, Philadelphia, PA 19104.

Guignard, M., P.M. Hahn, Z. Ding, B.-J. Kim, and Y.-R. Zhu, 2006, Extensions and variations on quadratic assignment problems including the quadratic 3-dimensional assignment problem (Q3AP), Proceedings of 2006 NSF Design, Service, and Manufacturing Grantees and Research Conference, St. Louis, MO.

Haddadi, S., and H. Ouzia, 2004, Effective algorithm and heuristic for the generalized assignment problem, European Journal of Operational Research **153**, 184-190.

Hadley, S.W., F. Rendl, and H. Wolkowicz, 1992, A new lower bound via projection for the quadratic assignment problem, Mathematics of Operations Research **17**, 727-739.

Hahn, P.M., and T.L. Grant, 1998, Lower bounds for the quadratic assignment problem based upon a dual formulation, Operations Research **46**, 912-922.

Hahn, P.M., W.L. Hightower, T.A. Johnson, M. Guignard-Spielberg, and C. Roucairol, 2001, Tree elaboration strategies in branch-and-bound algorithms for solving the quadratic assignment problem, Yugoslav Journal of Operations Research **11**, 41-60.

Hahn, P.M., B.-J. Kim, M. Guignard, J.M. Smith, and Y.-R. Zhu, 2006a, An algorithm for the generalized quadratic assignment problem, to appear in Computational Optimization and Applications.

Hahn, P.M., B.-J. Kim, T. Stützle, S. Kanthak, W.L. Hightower, H. Samra, Z. Ding, and M. Guignard, 2006b, The quadratic three-dimensional assignment problem: exact and approximate solution methods, to appear in European Journal of Operational Research.

Held, M.P., P. Wolfe, and H.P. Crowder, 1974, Validation of subgradient optimization, Mathematical Programming **6**, 62-88.

Karisch, S.E., E. Çela, J. Clausen, and T. Espersen, 1999, A dual framework for lower bounds of the quadratic assignment problem based on linearization, Computing **63**, 351-403.

Karmarkar, N.K., and K.G. Ramakrishnan, 1991, Computational results of an interior point algorithm for large scale linear programming, Mathematical Programming **52**, 555-586.

Karp, R.M., 1972, Reducibility among combinatorial problems, in: Miller, R.E., and J.W. Thatcher (Eds.), Complexity of Computer Computations, Plenum Press, New York, 85-103.

Kim, B.-J., 2006, Investigation of methods for solving new classes of quadratic assignment problems (QAPs), Ph.D. Dissertation, University of Pennsylvania, Philadelphia, PA 19104.

Koopmans, T.C., and M.J. Beckmann, 1957, Assignment problems and the location of economic activities, Econometrica **25**, 53-76.

Lasserre, J.B., 2001, An explicit exact SDP relaxation for nonlinear 0-1 programs, in: Aardal, K., and B. Gerards (Eds.), Lecture Notes in Computer Science **2081**, Springer, Berlin Heidelberg, 293-303.

Laurent, M., 2003, A comparison of the Sherali-Adams, Lovász-Schrijver and Lasserre relaxations for 0-1 programming, Mathematics of Operations Research **28**, 470-496.

Lawler, E.L., 1963, The quadratic assignment problem, Management Science **9**, 586-599.

Laywine, C.F., and G.L. Mullen, 1998, Discrete Mathematics Using Latin Squares, John Wiley & Sons, New York.

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.

Loiola, E.M., N.M.M. de Abreu, P.O. Boaventura-Netto, P.M. Hahn, and T. Querido, 2006, A survey for the quadratic assignment problem, to appear in European Journal of Operational Research.

Lovász, L., and A. Schrijver, 1991, Cones of matrices and set-functions and 0-1 optimization, SIAM Journal on Optimization **1**, 166-190.

Magirou, V.F., and J.Z. Milis, 1989, An algorithm for the multiprocessor assignment problem, Operations Research Letters **8**, 351-356.

Magirou V.F., 1992, An improved partial solution to the task assignment and multiway cut problems, Operations Research Letters **12**, 3-10.

Magos, D., and P. Miliotis, 1994, An algorithm for the planar three-index assignment problem, European Journal of Operational Research **77**, 141-153.

Meller, R.D., and Y.A. Bozer, 1997, Alternative approaches to solve the multi-floor facility layout problem, Journal of Manufacturing Systems **16**, 192-203.

Nauss, R.M., 2003, Solving the generalized assignment problem: an optimizing and heuristic approach, INFORMS Journal on Computing **15**, 249-266.

Nauss, R.M., 2006, The generalized assignment problem, in: Karlof, J.K, (Ed.), Integer Programming: Theory and Practice, CRC Press, 39-55.

Pardalos, P.M., F. Rendl, and H. Wolkowicz, 1994, The quadratic assignment problem: a survey and recent developments, in: Pardalos, P.M., and H. Wolkowicz (Eds.), Quadratic Assignment and Related Problems, DIMACS Series in Discrete Mathematics and Theoretical Computer Science, Vol. **16,** AMS, Providence, RI, 1-42.

Pardalos, P.M., and H. Wolkowicz (Editors), 1994, Quadratic Assignment and Related Problems, DIMACS Series in Discrete Mathematics and Theoretical Computer Science, Vol. **16,** AMS, Providence, RI.

Peck, K.E., 1983, Operational analysis of freight terminals handling less than container load shipments, Ph.D. Dissertation, University of Illinois at Urbana-Champaign, Urbana, IL 61801.

Pierskalla, W.P., 1967a, The tri-substitution method for the three-dimensional assignment problem, Journal of the Canadian Operational Research Society **5**, 71-81.

Pierskalla, W.P., 1967b, The multi-dimensional assignment problem, Technical Memorandum No. 93, Operations Research Department, CASE Institute of Technology. <http://www.anderson.ucla.edu/faculty/william.pierskalla/Chronological_Bank/Research_and_Publication_Chro.html#Mathematical>

Pierskalla, W.P., 1968, The multidimensional assignment problem, Operations Research **16**, 422-431.

QAPLIB Home Page <http://www.seas.upenn.edu/qaplib/>

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

Rendl, F., and H. Wolkowicz, 1992, Applications of parametric programming and eigenvalue maximization to the quadratic assignment problem, Mathematical Programming **53**, 63-78.

Rendl, F., and R. Sotirov, 2007, Bounds for the quadratic assignment problem using the bundle method, Mathematical Programming **109**, 505-524.

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.

Ross, G.T., and R.M. Soland, 1975, A branch-and-bound algorithm for the generalized assignment problem, Mathematical Programming **8**, 91–103.

Roupin, F., 2004, From linear to semidefinite programming: an algorithm to obtain semidefinite relaxations for bivalent quadratic problems, Journal of Combinatorial Optimization **8**, 469-493.

Sahni, S., and T. Gonzalez, 1976, P-complete approximation problems, Journal of the Association for Computing Machinery **23**, 555-565.

Samra, H., Z. Ding, and P.M. Hahn, 2003, Optimal symbol mapping diversity for multiple packet transmissions, Proceedings of the 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing **4**, Hong Kong, IV-181-184.

Samra, H., Z. Ding, and P.M. Hahn, 2004, Symbol mapping diversity design for multiple packet transmissions, IEEE Transactions on Communications **53**, 810-817.

Savelsbergh, M., 1997, A branch-and-price algorithm for the generalized assignment problem, Operations Research **45**, 831-841.

Sherali, H.D., and W.P. Adams, 1990, A hierarchy of relaxations between the continuous and convex hull representations for zero-one programming problems, SIAM Journal on Discrete Mathematics **3**, 411-430.

Sherali, H.D., and W.P. Adams, 1994, A hierarchy of relaxations and convex hull characterizations for mixed-integer zero-one programming problems, Discrete Applied Mathematics **52**, 83-106.

Sherali, H.D., and W.P. Adams, 1998, Reformulation-linearization techniques for discrete optimization problems, in: Du, D.-Z., and P.M. Pardalos (Eds.), Handbook of Combinatorial Optimization, Vol. **1**, Kluwer Academic Publishers, Dordrecht, 479-532.

Sherali, H.D., and W.P. Adams, 1999, A Reformulation-Linearization Technique for Solving Discrete and Continuous Nonconvex Problems, Kluwer Academic Publishers, Dordrecht, The Netherlands.

Sofianopoulous, S., 1992a, Simulated annealing applied to the process allocation problem, European Journal of Operational Research **60**, 327-334.

Sofianopoulous, S., 1992b, The process allocation problem: a survey of the application of graph-theoretic and integer programming approaches, Journal of the Operational Research Society **43**, 407-413.

Spieksma, F.C.R., 2000, Multi index assignment problems: complexity, approximation, applications, in: Pardalos, P.M., and L.S. Pitsoulis (Eds.), Nonlinear Assignment Problems: Algorithms and Applications, Kluwer Academic Publishers, Dordrecht, 1-11.

Tsui, L.Y., and C.-H. Chang, 1990, A microcomputer based decision support tool for assigning dock doors in freight yards, Computers and Industrial Engineering **19**, 309-312.

Tsui, L.Y., and C.-H. Chang, 1992, An optimal solution to a dock door assignment problem, Computers and Industrial Engineering **23**, 283-286.

Zhao, Q., S.E. Karisch, F. Rendl, and H. Wolkowicz,, 1998, Semidefinite programming relaxations for the quadratic assignment problem, Journal of Combinatorial Optimization **2**, 71-109.