Semi-definite Programming Relaxation of Quadratic Assignment Problems based on Nonredundant Matrix Splitting

Jiming Peng, Tao Zhu[*], Hezhi Luo[†], Kim-Chuan Toh[‡]

May 3, 2013

Abstract

Quadratic Assignment Problems (QAPs) are known to be among the most challenging discrete optimization problems. Recently, a new class of semi-definite relaxation (SDR) models for QAPs based on matrix splitting has been proposed [25, 28]. In this paper, we consider the issue of how to choose an appropriate matrix splitting scheme so that the resulting relaxation model is easy to solve and able to provide a strong bound. For this, we first introduce a new notion of the so-called redundant and non-redundant matrix splitting and show that the relaxation based on a non-redundant matrix splitting can provide a stronger bound than a redundant one. Then we propose to follow the minimal trace principle to find a non-redundant matrix splitting via solving an auxiliary semi-definite programming problem (SDP). We show that applying the minimal trace principle directly leads to the so-called orthogonal matrix splitting introduced in [28]. To find other non-redundant matrix splitting schemes whose resulting relaxation models are relatively easy to solve, we elaborate on two splitting schemes based on the so-called one-matrix and the sum-matrix. We analyze the solutions from the auxiliary problems for these two cases and characterize when they can provide a non-redundant matrix splitting. The lower bounds from these two splitting schemes are compared theoretically. Promising numerical results on some large QAP instances are reported, which further validate our theoretical conclusions.

Key words. Quadratic Assignment Problem (QAP), Semidefinite Programming (SDP), Semidefinite Relaxation (SDR), Matrix Splitting, Lower Bound.

[*]Corresponding author. Department of Industrial and Enterprise System Engineering, University of Illinois at Urbana-Champaign. Urbana, IL 61801, USA. Email: pengj, taozhu1@illinois.edu. The work of the first two authors are supported by AFOSR grant FA9550-09-1-0098 and NSF grant DMS 09-15240 ARRA.

[†]Department of Applied Mathematics, Zhejiang University of Technology, Hangzhou, Zhejiang 310032, People's Republic of China. Email:hzluo@zjut.edu.cn. The work of this author is supported by the National Natural Science Foundation of China under grant 11071219 and the Zhejiang Provincial Natural Science Foundation of China under grants Y13A010050 and Y13A010077.

[‡]Department of Mathematics, National University of Singapore, 10 Lower Kent Ridge Road, Singapore 119076. Email: mattohkc@nus.edu.sg.

1 Introduction

Given matrices A, B, we consider the quadratic assignment problem (denoted by QAP) of the following form

$$\min_{X \in \Pi} \mathrm{Tr}\big(AXBX^T\big) \tag{1}$$

where $\mathrm{Tr}(\cdot)$ denotes the trace of a matrix and Π is the set of permutation matrices. We assume that A and B are $n \times n$ symmetric matrices throughout this paper. QAP was first introduced by Koopmans and Beckmann [22] for facility location and has applications in many areas such as chip design [9, 18], image analysis and processing [26, 32], and communications [4]. For more applications of QAP, we refer the reader to the survey paper [24].

It is well-known that QAP is NP-hard. Searching for the global solution of QAP usually involves the branch and bound (B&B) method. A crucial issue in the B&B method is how to compute strong lower bounds efficiently. Various relaxations and bounds for QAPs have been proposed in the literature. Roughly speaking, these bounds can be categorized into two groups. The first group includes several bounds that are not very strong but can be computed efficiently such as the well-known Gilmore-Lawler bound (GLB)[13, 23], the bound based on projection [31] (denoted by PB) and the bound based on convex quadratic programming (denoted by QPB) [3]. The second group contains strong bounds that require expensive computation such as the bounds derived from lifted integer linear programming [1, 2, 15] and bounds based on SDRs [29, 35].

Among all the relaxation models, we are particularly interested in the semidefinite relaxation models which can provide relatively stronger lower bounds compared with other relaxations based on linear and quadratic programming. A popular way to derive the SDRs of QAP is to relax the rank-1 matrix $vec(X)vec(X)^T$ to a $n^2 \times n^2$ positive semidefinite matrix with nonnegative elements, where $vec(X)$ denotes the n^2-dimensional vector obtained from X by stacking its columns sequentially into a long vector. Though much progress has been obtained in solving the SDR based on the gram matrix $vec(X)vec(X)^T$ [29, 5, 10], the large number of $O(n^4)$ variables and constraints in these relaxations still make them formidable for medium size QAP instances with the current computation facilities. Recently, Ding and Wolkowicz [11] introduced a new SDR of QAP based on matrix lifting. The resulting SDR model has only $O(n^2)$ variables and constraints and thus can be solved using open source SDP solvers for QAPs of size $n \le 30$, though it still remains a computational challenge for $n \ge 30$.

In our recent work [28], a new framework to derive cheap and strong SDR for QAP based on various matrix splitting schemes was introduced. It is shown that some relaxation models in [25, 28] can provide competitive bounds comparing with other relaxation models in the literature. However, since there is a large variety in matrix splitting schemes, it is unclear which splitting scheme can lead to the strongest relaxation.

In this paper, we attempt to address the issue of selecting a matrix splitting scheme whose resulting relaxation model can provide a strong bound. Our first major contribution of the paper is to introduce a new notion of the so-called redundant and non-redundant positive semidefinite (PSD) matrix splitting and use the new notion to select a good matrix splitting scheme whose associated SDR can be solved efficiently and yet is still able to provide a strong bound. For this, we first show that for any given redundant matrix splitting, there exists a corresponding non-redundant matrix splitting whose SDR can provide a stronger bound. To find such a non-redundant matrix splitting, we propose to solve some auxiliary SDP problems following the minimal trace principle[1]. In particular, we show that a straightforward application of the minimal trace principle leads to the so-called orthogonal matrix splitting introduced in [25, 28]. We also illustrate that for a given matrix, there may exist multiple non-redundant matrix splitting schemes.

Secondly, to further help select a non-redundant splitting scheme whose corresponding SDR can be solved relatively efficiently, we consider two specific matrix splitting schemes based on the so-called one-matrix and the sum-matrix. We investigate the theoretical properties of the optimal solutions to the auxiliary SDP problems under these two circumstances and characterize when the derived matrix splitting schemes are non-redundant. We also compare the two lower bounds from the one-matrix and sum-matrix splitting schemes and show that, under certain conditions, the lower bound derived from the sum-matrix splitting is stronger.

Thirdly, based on the rank information at the optimal solution to the auxiliary problem, we present a new implementation of the relaxation model which leads to substantial improvement over the implementation in [28]. Numerical experiments show that the bound based on the new non-redundant matrix splitting schemes and implementation is very competitive with existing bounds including the bounds based on other matrix splitting schemes, and they can be computed more efficiently.

The paper is organized as follows. In Section 2, we first introduce the notion of redundant and non-redundant matrix splitting and show that the SDR based on a non-redundant PSD splitting can provide a stronger lower bound than a redundant one. The minimal trace principle is proposed to find a non-redundant matrix splitting scheme. In particular, we show that a direct application of the minimal trace principle leads to the so-called orthogonal PSD matrix splitting introduced in [28]. In Section 3, we apply the minimal trace principle to the one-matrix and sum-matrix splitting whose corresponding SDRs are relatively easy to solve. We give conditions under which these two splitting schemes are non-redundant and compare the lower bounds derived from their corresponding SDRs. In Section 4, we present the SDR models of QAPs based on the three matrix splitting schemes. Numerical results on some large QAP instances from QAPLIB [7] are presented in Section 5.

Throughout this paper, we use upper case letters to denote matrices and lower case letters for vectors. $E \in \Re^{n\times n}$ denotes the matrix of ones and $e \in \Re^n$ denotes the vector of ones. $\mathcal{S}^n$ denotes the set of $n \times n$ real symmetric matrices and $\mathcal{S}^n_+$ denotes the set of $n \times n$ symmetric positive semi-definite (PSD) matrices. We use $X \succeq 0$ to denote $X \in \mathcal{S}^n_+$ and $X \geq 0$ to denote $X_{ij} \geq 0$ for all i, j. $[X]_{ij}$ denotes the (i, j) element of X. For a given $B \in \mathcal{S}^n$, $\mathrm{diag}\,(B)$ denotes the vector consisting of the diagonal elements of B, and B_{off} denotes the matrix which has the same off-diagonal elements as B but with zero diagonal elements. For a vector d, $\mathrm{diag}\,(d)$ denotes the diagonal matrix whose diagonal is d. Let $\max(B)$ (or $\min(B)$) denote the column vector whose i-th component is the maximal element (or minimal element) in the i-th row (denoted by $B_{i,:}$) of B. $\mathcal{L}_2(B)$ denotes the column vector whose i-th component is the 2-norm of the i-th row of B. $\lambda_{\max}(B)$ and $\lambda_{\min}(B)$ denote the largest eigenvalue and the smallest eigenvalue, respectively.

[1]The minimal trace principle is chosen because, as shown in our analysis in Section 2, the final solution matrix following the minimal principle has the minimal rank and the rank information on the splitting matrix can be further used to reduce the memory requirement and simplify the relaxation model as discussed in Section 4.

2 Redundant and Non-redundant Matrix Splitting

As shown in [28], there exist various matrix splitting schemes for a given matrix B and it is unclear which splitting can lead to the strongest relaxation. In this section, we first introduce a new notion of the so-called redundant and non-redundant matrix splitting and show that for any given redundant matrix splitting, there exists another non-redundant matrix splitting that can provide a stronger relaxation. To find such a non-redundant positive semidefinite (PSD) splitting, we refer to the minimal trace principle. The relationship between the non-redundant matrix splitting based on minimal trace principle and the orthogonal PSD splitting schemes introduced in [28] will be discussed as well.

We start with the following definition from [28].

Definition 2.1. *Given matrix B, we call matrix pair (B_1, B_2) a PSD matrix splitting of B if it satisfies*

$$B = B_1 - B_2,\ \ B_1, B_2 \succeq 0.$$

In particular, if the additional constraint $B_1B_2 = 0$ is satisfied, then we call (B_1, B_2) an orthogonal PSD splitting of B.

As pointed out in [28], there exist many PSD matrix splitting schemes. If a PSD splitting (B_1, B_2) of matrix B is available, then we can obtain the following basic SDR for QAPs:[2]

$$\mu(B_1, B_2) = \min_{Y \in \mathcal{Y}(B_1,B_2)} \mathrm{Tr}(AY), \tag{2}$$

where the feasible set $\mathcal{Y}(B_1, B_2)$ is defined by

$$\mathcal{Y}(B_1, B_2) = \left\{ Y \in \mathcal{S}^n \;\middle|\; \begin{array}{c} \exists\, Y_1, Y_2 \in \mathcal{S}^n_+, X \in \Re^{n\times n} \text{ satisfying } Y = Y_1 - Y_2, \\ Y_1 - XB_1X^T \succeq 0, \quad Y_2 - XB_2X^T \succeq 0, \\ \mathrm{diag}\,(Y_1) = X\mathrm{diag}\,(B_1), \quad Y_1 e = XB_1 e, \\ \mathrm{diag}\,(Y_2) = X\mathrm{diag}\,(B_2), \quad Y_2 e = XB_2 e, \\ Xe = X^T e = e, \quad X \geq 0 \end{array} \right\}. \tag{3}$$

We now introduce the following definition.

Definition 2.2. *A PSD matrix splitting (B_1, B_2) is said to be redundant (or non-redundant) if there exists (or does not exist) a nonzero matrix $R \succeq 0$ satisfying*

$$B_1 - R \succeq 0, \quad B_2 - R \succeq 0.$$

From the above definition, we immediately have

[2] For simplicity of discussion, in all the theoretical analysis of this work, we consider only the basic model (2) which is slightly different from the full SDR model to be described in Section 4. However, since in the full model we only add some convex constraints on the elements of Y, one can easily extend the results for the basic model to the full model.

Proposition 2.3. *Given a matrix B. A PSD matrix splitting (B_1, B_2) is non-redundant if and only if 0 is the optimal solution of the following SDP*

$$\begin{aligned} &\max \quad \mathrm{Tr}(R) && (4)\\ &s.t. \quad B_1 - R \succeq 0, B_2 - R \succeq 0, R \succeq 0. && (5) \end{aligned}$$

We next recall a well-known result regarding the doubly stochastic matrices. A real $n \times n$ matrix $M = (M_{ij})$ is doubly stochastic if the entries of M are non-negative, and each row and column of M sums to 1 [27]. The following result is from [27, Theorem 2, Birkhoff's theorem].

Lemma 2.4. *The set of $n \times n$ doubly stochastic matrices is a convex set whose extreme points are the permutation matrices.*

Based on Lemma 2.4, we first establish a result regarding a redundant PSD splitting.

Theorem 2.5. *If a PSD matrix splitting (B_1, B_2) of the matrix B is redundant with matrix R, then we have*

$$\mathcal{Y}(B_1 - R, B_2 - R) \subseteq \mathcal{Y}(B_1, B_2), \tag{6}$$

where $\mathcal{Y}(\cdot)$ is the set as defined in (3).

Proof. Since the PSD matrix splitting (B_1, B_2) is redundant, there exists nontrivial $R \succeq 0 \in \mathcal{S}^n$ such that

$$B_1 - R \succeq 0, \quad B_2 - R \succeq 0.$$

Clearly, $(B_1 - R, B_2 - R)$ is also a PSD splitting of B.

Now let $Y \in \mathcal{Y}(B_1 - R, B_2 - R)$, i.e., there exist $(Y_1, Y_2, X) \in \mathcal{S}_+^n \times \mathcal{S}_+^n \times \Re^{n\times n}$ such that

$$Y = Y_1 - Y_2, \tag{7}$$

$$Y_1 - X(B_1 - R)X^T \succeq 0, \quad Y_2 - X(B_2 - R)X^T \succeq 0, \tag{8}$$

$$\mathrm{diag}\,(Y_1) = X\mathrm{diag}\,(B_1 - R), \quad Y_1 e = X(B_1 - R)e, \tag{9}$$

$$\mathrm{diag}\,(Y_2) = X\mathrm{diag}\,(B_2 - R), \quad Y_2 e = X(B_2 - R)e, \tag{10}$$

$$Xe = X^T e = e, \quad X \geq 0. \tag{11}$$

Since X is a $n \times n$ doubly stochastic matrix, by Lemma 2.4, it can be expressed as a convex combination of permutation matrices. Let $|\Pi|$ be the cardinality of the set of permutation matrices. Therefore, there exists $\lambda_i \geq 0, \forall i = 1, \cdots, |\Pi|$ such that

$$X = \sum_{i=1}^{|\Pi|} \lambda_i \hat{X}_i, \quad \sum_{i=1}^{|\Pi|} \lambda_i = 1, \quad \hat{X}_i \in \Pi, \ \forall i = 1, \cdots, |\Pi|. \tag{12}$$

Define

$$Y_R = \sum_{i=1}^{|\Pi|} \lambda_i \hat{X}_i R \hat{X}_i^T. \tag{13}$$

Since $\hat{X}_i \in \Pi$ for all i and $R \succeq 0$, from (12) we have $Y_R \succeq 0$. Further, it is easy to verify that

$$\text{diag}(Y_R) = \sum_{i=1}^{|\Pi|} \lambda_i \hat{X}_i \text{diag}(R) = X\text{diag}(R), \quad Y_R e = \sum_{i=1}^{|\Pi|} \lambda_i \hat{X}_i Re = XRe. \tag{14}$$

It remains to show

$$Y_R - XRX^T \succeq 0. \tag{15}$$

Since $R \succeq 0$, we have $R = VV^T$ for some $V \in \mathcal{S}^n$. Therefore, for any $d \in \Re^n$, from (13) we obtain

$$\begin{aligned} d^T XRX^T d &= \left(V^T X^T d\right)^T \left(V^T X^T d\right) = \left\|V^T X^T d\right\|_2^2 = \left\| \sum_{i=1}^{|\Pi|} \lambda_i V^T \hat{X}_i^T d \right\|_2^2 \\ &\leq \sum_{i=1}^{|\Pi|} \lambda_i \left\| V^T \hat{X}_i^T d \right\|_2^2 = \sum_{i=1}^{|\Pi|} \lambda_i d^T \left(\hat{X}_i R \hat{X}_i^T\right) d = d^T Y_R d, \end{aligned}$$

where the inequality follows from the fact that the function $\|\cdot\|_2^2$ is convex and because $\sum_{i=1}^{|\Pi|} \lambda_i = 1$ and $\lambda_i \geq 0$, $i = 1, \ldots, |\Pi|$. The above relation means that (15) holds true.

Now let us define

$$\bar{Y}_1 = Y_1 + Y_R, \qquad \bar{Y}_2 = Y_2 + Y_R.$$

Since $Y = \bar{Y}_1 - \bar{Y}_2$, and

$$\begin{aligned} \bar{Y}_1 - XB_1X^T &= Y_1 - X(B_1 - R)X^T + Y_R - XRX^T \succeq 0, \\ \bar{Y}_2 - XB_2X^T &= Y_1 - X(B_2 - R)X^T + Y_R - XRX^T \succeq 0, \end{aligned}$$

from (7)-(11) and (14)-(15) one can easily verify $Y \in \mathcal{Y}(B_1, B_2)$. This proves (6). □

From Theorem 2.5, we see that given a QAP with matrices (A, B) and a PSD matrix splitting (B_1, B_2) of B, if the matrix splitting (B_1, B_2) is redundant, then we have

$$\min_{Y \in \mathcal{Y}(B_1 - R, B_2 - R)} \text{Tr}(AY) \geq \min_{Y \in \mathcal{Y}(B_1, B_2)} \text{Tr}(AY). \tag{16}$$

Therefore, in order to derive a strong lower bound, a non-redundant PSD matrix splitting should be used.

We next discuss how to find a non-redundant PSD matrix splitting for a given matrix B. Inspired by Proposition 2.3, we consider the following auxiliary problem induced by the minimal trace principle:

$$\begin{aligned} \text{(MTMS-PSD)} \quad \min \quad & \text{Tr}(B_1) \\ s.t. \quad & B_1 - B_2 = B, \\ & B_1 \succeq 0, \quad B_2 \succeq 0. \end{aligned}$$

It is easy to see that the above problem is strictly feasible.

For a given matrix B, let Q be an orthogonal matrix whose columns are the eigenvectors of the matrix B associated with the eigenvalues $\{\lambda_1, \cdots, \lambda_n\}$, i.e., $B = \sum_{i=1}^n \lambda_i q_i q_i^T$ where q_i is the i-th column of Q. Let us define

$$B^+ = \sum_{i:\lambda_i \geq 0} \lambda_i q_i q_i^T, \quad B^- = -\sum_{i:\lambda_i < 0} \lambda_i q_i q_i^T. \tag{17}$$

The splitting (B^+, B^-) is precisely the orthogonal PSD splitting introduced in [28]. Our next result establishes the equivalence between the optimal solution to the MTMS-PSD problem and the orthogonal PSD splitting.

Theorem 2.6. *The optimal solution (B_1^*, B_2^*) to the problem (MTMS-PSD) is given by (B^+, B^-). Furthermore, the splitting (B^+, B^-) is non-redundant.*

Proof. Denote the optimal solution to the MTMS-PSD problem by (B_1^*, B_2^*). We first show $(B_1^*, B_2^*) = (B^+, B^-)$. Let P be the projection matrix defined by

$$P = \sum_{i:\lambda_i \geq 0} q_i q_i^T.$$

It follows immediately that

$$\mathrm{Tr}(B_1^*) \geq \mathrm{Tr}(B_1^* P) = \mathrm{Tr}\big(B_1^* P^2\big) = \mathrm{Tr}(P B_1^* P) \geq \mathrm{Tr}(P(B_1^* - B_2^*)P) = \mathrm{Tr}\big(B^+\big),$$

where the first inequality follows from the relation

$$\mathrm{Tr}(B_1^*(I - P)) \geq 0.$$

Here I denotes the identity matrix in $\Re^{n \times n}$. Similarly, one has

$$\mathrm{Tr}(B_2^*) \geq \mathrm{Tr}(B_2^*(I - P)) = \mathrm{Tr}((I - P)B_2^*(I - P)) \geq \mathrm{Tr}\big(B^-\big).$$

Therefore, we have

$$\mathrm{Tr}(B_1^*) + \mathrm{Tr}(B_2^*) \geq \mathrm{Tr}\big(B^+\big) + \mathrm{Tr}\big(B^-\big),$$

and the equality holds if and only if

$$\mathrm{Tr}(B_1^*(I - P)) = 0, \quad \mathrm{Tr}(B_2^* P) = 0. \tag{18}$$

Since all the matrices B_1^*, B_2^*, P and $I - P$ are positive semi-definite. Relation (18) holds if and only if

$$B_1^* = B_1^* P = P B_1^*, \quad B_2^* P = P B_2^* = 0. \tag{19}$$

Since $B_1^* - B_2^* = B = B^+ - B^-$, we thus have

$$B_1^* = PBP = B^+, \quad B_2^* = -(I - P)B(I - P) = B^-.$$

It remains to show that the matrix splitting (B^+, B^-) is non-redundant. Suppose to the contrary that (B^+, B^-) is a redundant splitting of B, i.e., there exists $R \neq 0 \succeq 0$ such that

$$B_1 = B^+ - R \succeq 0, \quad B_2 = B^- - R \succeq 0, \quad B_1 - B_2 = B.$$

Then we have

$$\mathrm{Tr}\big(B^+B^-\big) = \mathrm{Tr}((B_1+R)(B_2+R)) \geq \mathrm{Tr}(B_1B_2) + \mathrm{Tr}\big(R^2\big) > 0,$$

which contradicts to the relation $\mathrm{Tr}\big(B^+B^-\big) = 0$. This finishes the proof of the theorem. □

We remark that for the orthogonal PSD splitting (B^+, B^-), it is easy to see that

$$\mathrm{Rank}(B) = \mathrm{Rank}(B^+) + \mathrm{Rank}(B^-).$$

Since for any matrix splitting $B = B_1 - B_2$, one has

$$\mathrm{Rank}(B) \leq \mathrm{Rank}(B_1) + \mathrm{Rank}(B_2).$$

We thus have the following corollary.

Corollary 2.7. *For any given matrix B, the optimal solution to the MTMS-PSD problem is also optimal to the following rank minimization problem*

$$\min \quad \mathrm{Rank}(B_1) + \mathrm{Rank}(B_2) \tag{20}$$

$$s.t. \quad B_1 - B_2 = B, \quad B_1, B_2 \succeq 0. \tag{21}$$

Since the rank of the splitting matrices will be further used to reduce the memory requirement and speed up the solving process for the relaxation model, the minimal rank solution is very appealing from both a theoretical and computational viewpoint.

Based on Theorem 2.6, for a given matrix B, if its non-redundant splitting is unique, then the SDR based on the orthogonal PSD matrix splitting will be the strongest among all the PSD matrix splittings. However, as one can see from the following example, the non-redundant PSD matrix splitting of a matrix might not be unique.

Example 2.8. Consider the matrix

$$B = \begin{pmatrix} 0 & 1 & 2 \\ 1 & 0 & 1 \\ 2 & 1 & 0 \end{pmatrix}.$$

By solving the MTMS-PSD problem with the above B, we obtain the optimal solution as follows

$$B_1^* = \begin{pmatrix} 1.0774 & 0.7887 & 1.0774 \\ 0.7887 & 0.5774 & 0.7887 \\ 1.0774 & 0.7887 & 1.0774 \end{pmatrix}, \quad B_2^* = \begin{pmatrix} 1.0774 & -0.2113 & -0.9226 \\ -0.2113 & 0.5774 & -0.2113 \\ -0.9226 & -0.2113 & 1.0774 \end{pmatrix}.$$

By Theorem 2.6, (B_1^*, B_2^*) is an orthogonal and non-redundant PSD matrix splitting of B. One can easily check that $\mathrm{Tr}(B_1^*B_2^*) = 0$.

Now, let us choose

$$B_1 = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}, \quad B_2 = \begin{pmatrix} 1 & 0 & -1 \\ 0 & 1 & 0 \\ -1 & 0 & 1 \end{pmatrix}.$$

It is easy to see that (B_1, B_2) is a PSD matrix splitting of B. Since $\text{Tr}(B_1 B_2) \neq 0$, by Theorem 2.6, (B_1, B_2) is not the optimal solution to the MTMS-PSD problem. We next show that (B_1, B_2) is also non-redundant. Suppose to the contrary that (B_1, B_2) is redundant, i.e., there exists $R \neq 0 \succeq 0$ satisfying

$$B_1 - R \succeq 0, \quad B_2 - R \succeq 0.$$

Since $B_1 = E$ and $B_1 - R \succeq 0$, it must hold that $R = \alpha E$ for some $0 < \alpha \leq 1$. On the other hand, for any $0 < \alpha \leq 1$, one can easily check that the matrix $B_2 - \alpha E$ is not positive semidefinite.

The non-uniqueness of the non-redundant PSD matrix splitting for a given matrix B shown in the above example illustrates that it is nontrivial to find the strongest SDR based on matrix splitting. In the next section, we will discuss how to find other non-redundant matrix splitting schemes whose corresponding SDR is relatively easy to solve.

3 Two Matrix Splitting Schemes based on the Minimal Trace Principle

In this section, we first use the minimal trace principle to derive two matrix splitting schemes and characterize conditions under which the constructed matrix splitting is non-redundant. Then, we compare the lower bounds provided by the SDRs of QAPs based on these two matrix splitting schemes. We start by stating an assumptions regarding the QAPs throughout this section.

Assumption 3.1. *At least one matrix (A or B) in the underlying QAP has zeros on its diagonal.*

It should be pointed out that the above assumption is quite reasonable and most QAP instances from the QAP library indeed satisfy such a condition. On the other hand, suppose both matrices A and B have nonzero diagonals. Let d_a and d_b denote the vectors consisting of the diagonal elements from A and B, respectively. We thus can write A and B as $A = A_0 + \text{diag}(d_a), B = B_0 + \text{diag}(d_b)$. It follows immediately that

$$\text{Tr}\big(AXBX^T\big) = \text{Tr}\big(A_0 X B_0 X^T\big) + d_a^T X d_b$$

The second term in the above relation reduces to a linear assignment problem which can be solved via its linear programming relaxation. For the ease of discussion, in the remaining part of this section, we assume that the matrix A has zeros on its diagonal.

Under Assumption 3.1, we can easily show that for any diagonal matrix D, one has

$$\text{Tr}\big(AXBX^T\big) = \text{Tr}\big(AX(B - D)X^T\big) \quad \forall\, X \in \Pi. \tag{22}$$

That is, we can arbitrarily adjust the diagonal of B without affecting the objective value of the QAP problem.

3.1 Minimal trace one-matrix splitting

Let us first consider a special case of the MTMS-PSD problem when $B_1 = tE$ for $t \geq 0$ where $E = ee^T$ is the all-1 matrix. In such a scenario, the MTMS-PSD problem reduces to the auxiliary

SDP problem considered in [28]:

$$\begin{array}{ll} \min & \alpha \\ \text{s. t.} & \alpha E - B \succeq 0, \quad \alpha \geq 0. \end{array} \tag{23}$$

If the problem (23) is feasible, then the optimal solution α of problem (23) is used to split the matrix B into the following form used in [25]:

$$B = \alpha E - (\alpha E - B).$$

However, as pointed out in [28], the problem (23) is in general infeasible. As a remedy for such an infeasibility issue, we propose to split the matrix $B - \beta I$ into the following form:

$$B - \beta I = \alpha E - B_2, \quad \text{with } B_2 = \alpha E + \beta I - B \succeq 0, \tag{24}$$

where $\alpha \geq 0$ and β are parameters to be identified. We call $(\alpha, \beta I)$ the one-matrix splitting of B. The above splitting is particularly attractive because the matrix $B_1 = \alpha E + \beta I$ is invariant under permutation. Therefore, it can substantially reduce the computational cost for solving the relaxed problem.

As in the previous section, in order to find a non-redundant matrix splitting for B, we propose to solve the following auxiliary problem:

$$\min \left\{ n(\alpha + \beta) : \alpha E + \beta I - B \succeq 0, \quad (\alpha, \beta) \in \Re^2 \right\}, \tag{25}$$

We next present an interesting result regarding problem (25) and the detailed proof of the theorem is given in Appendix 1.

Theorem 3.2. *Let (α, β) be the optimal solution of problem (25). If the matrix B is nonnegative, then $\alpha > 0$.*

Theorem 3.2 states for any nonnegative matrix, we can always extract a positive scalar of the all one-matrix that is invariant for any permutation. Therefore, it is desirable to use the one-matrix PSD splitting. Our next theorem explores conditions under which the one-matrix splitting derived from problem (25) is non-redundant.

Theorem 3.3. *Let (α, β) be the optimal solution of problem (25). Then the following statements hold:*

(i) If B is nonnegative, then $(\alpha E, \alpha E + \beta I - B)$ is a non-redundant PSD matrix splitting of $B - \beta I$;

(ii) If $\beta > 0$, then $(\alpha E + \beta I, \alpha E + \beta I - B)$ is a redundant PSD matrix splitting of B.

Proof. We first consider statement (i). By Theorem 3.2, $\alpha > 0$. We have $B_1 = \alpha E, B_2 = \alpha E + \beta I - B$. It is easy to see that (B_1, B_2) is a PSD matrix splitting of $B - \beta I$. We now prove that (B_1, B_2) is non-redundant. Suppose to the contrary that (B_1, B_2) is redundant, i.e., there exists nonzero matrix $R \succeq 0$ satisfying

$$B - \beta I = (B_1 - R) - (B_2 - R), \quad B_1 - R \succeq 0, \quad B_2 - R \succeq 0.$$

Since $B_1 = \alpha E$, $\alpha > 0$ and $B_1 - R \succeq 0$, it must hold that $R = \tau E$ for some $0 < \tau \leq \alpha$. Therefore,

$$B_2 - R = (\alpha - \tau)E + \beta I - B \succeq 0.$$

This implies that $(\alpha - \tau, \beta)$ is a feasible solution of the problem (25). Note that $n(\alpha - \tau + \beta) < n(\alpha + \beta)$, which contradicts the optimality of (α, β) with respect to problem (25). This proves statement (i).

Next we turn to statement (ii). Since $\beta > 0$ and $\alpha \geq 0$, we have $\alpha E + \beta I \succ 0$. Because $B = (\alpha E + \beta I) - (\alpha E + \beta I - B)$, and $\alpha E + \beta I - B \succeq 0$, $(\alpha E + \beta I, \alpha E + \beta I - B)$ is a PSD splitting of B. Let $R = \eta(\alpha E + \beta I - B) \succeq 0$, and $\eta \in (0, 1)$ is chosen such that

$$\lambda_{\max}(R) < \beta.$$

It is easy to show that $\alpha E + \beta I - R \succeq 0$ and $\alpha E + \beta I - B - R \succeq 0$. Therefore, the matrix splitting $(\alpha E + \beta I, \alpha E + \beta I - B)$ is redundant. □

Theorem 3.3 implies that if problem (25) has an optimal solution (α, β) with $\beta > 0$, then the resulting SDR can be further improved by using a non-redundant PSD splitting of B. When $\beta \leq 0$, the one-matrix splitting might be a very good choice due to the simplicity of the resulting SDR model. We also point out that when B is the Hamming distance matrix of the hypercube in $\mathfrak{R}^m$, as proved in [25], the one-matrix splitting based on the minimal trace principle is also the orthogonal PSD splitting of B. In such a case, the optimal solution to problem (25) is $(\alpha^*, \beta^*) = (\frac{m}{2}, 0)$.

In our experiments, we also observe that for some QAP instances such as Tai20b, Tai25b, Tai35b, Tai40b and Tai50b, problem (25) has an optimal solution with $\beta < 0$ as listed in Table 1. We can further check from Table 3 and Table 4 of Section 5 that for these QAP instances, the SDR based on the one-matrix splitting scheme can provide a stronger lower bound than that based on the orthogonal PSD matrix splitting scheme.

Table 1: Optimal solution (α, β) of problem (25)

Prob.	α	β
Tai20b	420.4951	-2.5025
Tai25b	558.5071	-2.5238
Tai35b	584.1755	-2.0483
Tai40b	797.8480	-3.5505
Tai50b	969.1780	-5.2837

3.2 Minimal trace sum-matrix splitting

In this subsection, we combine the minimal trace principle and the so-called sum-matrix to construct a non-redundant matrix splitting for a given matrix B. First we recall the following definition [6].

Definition 3.4. *A matrix M is called a sum-matrix if*

$$M = ue^T + eu^T \quad (26)$$

for some $u \in \mathrm{R}^n$. The sum-matrix has the following property:

$$X(ue^T + eu^T)X^T = Xue^T + eu^TX^T, \quad \forall X \in \Omega, \tag{27}$$

where $\Omega = \{X \in \Re^{n\times n} \mid Xe = X^Te = e\}$.

Given a matrix B, we can decompose it into two parts as $B = \bar{B} + ue^T + eu^T$. Correspondingly, we have

$$\mathrm{Tr}\big(AXBX^T\big) = \mathrm{Tr}\big(AX\bar{B}X^T\big) + 2e^TAXu.$$

Using the above decomposition, we can reduce the contribution of the quadratic term in the objective function by moving the cost to a linear term.[3] In what follows we use the sum-matrix to construct a non-redundant matrix splitting framework. Based on relation (22), we propose a splitting of the following form:

$$B - D = ue^T + eu^T - B_2, \quad B_2 \succeq 0, \quad D = \mathrm{Diag}(d), \tag{28}$$

where $d \in \mathrm{R}^n$ and $u \in \mathrm{R}^n$ are vectors to be found. We call a pair (u, D) satisfying (28) a sum-matrix splitting of the matrix B. Like in the last subsection, we use the minimal trace principle to find a non-redundant sum-matrix splitting of the matrix B as follows:

$$\begin{aligned} \min_{u\in\Re^n, d\in\Re^n} \quad & e^T(2u + d) \\ \text{s. t.} \quad & ue^T + eu^T + D - B \succeq 0, \quad D = \mathrm{Diag}(d). \end{aligned} \tag{29}$$

It is easy to see that the above problem is strictly feasible. It should be pointed out that for simplicity of the model, we can also impose the constraint that $D = \beta I$ for some parameter $\beta \in \Re$. In such a circumstance, the sum-matrix splitting $(u, \beta I)$ of B includes the one-matrix splitting $(\alpha, \beta I)$ as a special case where $u = \frac{\alpha}{2}e$ for some $\alpha \in \Re$.

3.3 Relations between the lower bounds

In this subsection, we compare the two lower bounds provided by the SDRs based on the one-matrix splitting and the sum-matrix splitting described in the previous subsections.

Let $(\alpha, \beta I)$ and $(u, \beta I)$ be respectively the minimal trace one-matrix splitting and sum-matrix splitting of B, and $\mu_1(\alpha, \beta I)$ and $\mu_2(u, \beta I)$ be the corresponding lower bound derived from the following two SDRs of QAPs, i.e.,

$$\mu_1(\alpha, \beta I) = \min_{(X,Y_2)\in\mathcal{Y}_{(\alpha,\beta I)}} \alpha\mathrm{Tr}(AE) - \mathrm{Tr}(AY_2), \tag{30}$$

$$\mu_2(u, \beta I) = \min_{(X,Y_2)\in\mathcal{Y}_{(u,\beta I)}} 2e^TAXu - \mathrm{Tr}(AY_2), \tag{31}$$

[3] We note that a similar approach (called the reduction method) has been used to improve the GLB and eigenvalue bound for QAPs with nonsymmetric matrices in the literature [6, 8, 12, 30]. One simple choice is $u = \min(B_{\text{off}})$. For more details on the reduction method, we refer to Section 7.5.2 of [6].

where

$$\mathcal{Y}_{(\alpha,\beta I)} = \left\{ (X, Y_2) \in \Re^{n\times n} \times \mathcal{S}^n \;\middle|\; \begin{array}{c} Y_2 - \alpha E - X(\beta I - B)X^T \succeq 0, \\ \mathrm{diag}\,(Y_2) = \alpha e + X\mathrm{diag}\,(\beta I - B), \\ Y_2 e = n\alpha e + X(\beta I - B)e, \\ Xe = X^T e = e, \quad X \geq 0 \end{array} \right\},$$

$$\mathcal{Y}_{(u,\beta I)} = \left\{ (X, Y_2) \in \Re^{n\times n} \times \mathcal{S}^n \;\middle|\; \begin{array}{c} Y_2 - Xue^T - eu^T X^T + X(B - \beta I)X^T \succeq 0, \\ \mathrm{diag}\,(Y_2) = 2Xu - X\mathrm{diag}\,(B - \beta I), \\ Y_2 e = nXu + eu^T e - X(B - \beta I)e, \\ Xe = X^T e = e, \quad X \geq 0 \end{array} \right\}.$$

Note that here we only deal with a special case of problem (29) by setting $d = \beta e$ to get:

$$\begin{array}{lll} (32) & \min\limits_{u\in\Re^n,\beta\in\Re} & 2e^T u + n\beta \\ & \text{s. t.} & ue^T + eu^T + \beta I - B \succeq 0. \end{array}$$

Lemma 3.5. *Suppose that $(\bar{\alpha}, \bar{\beta})$ and $(\hat{u}, \hat{\beta})$ are the optimal solution of problems (25) and (32), respectively. Then $\bar{\beta} \geq \hat{\beta}$.*

Proof. Denote (u, β) any feasible solution to problem (32). Define

$$\alpha = \frac{2u^T e}{n}, \quad v = u - \frac{\alpha}{2}e.$$

It follows immediately

$$u = \frac{\alpha}{2}e + v, \quad v^T e = 0.$$

Using the above notation, we can rewrite problem (32) as

$$\begin{array}{lll} (33) & \min\limits_{v\in\Re^n,\beta\in\Re} & n\alpha + n\beta \\ & \text{s. t.} & \alpha E + ve^T + ev^T + \beta I - B \succeq 0; \\ & & e^T v = 0. \end{array}$$

We next show that the optimal solution to the above problem can be obtained explicitly. Denote the optimal solution of problem (33) by $(\hat{\alpha}, \hat{\beta})$ and $P = I - \frac{E}{n}$. From the constraint $e^T v = 0$ we obtain

$$P(\hat{\alpha}E + ve^T + ev^T + \hat{\beta}I - B)P = \hat{\beta}P - PBP \succeq 0,$$

which implies

$$\hat{\beta} \geq \lambda_{\max}(PBP).$$

Here $\lambda_{\max}(PBP)$ denotes the largest eigenvalue of the matrix PBP. Similarly, we have

$$(I - P)(\hat{\alpha}E + ve^T + ev^T + \hat{\beta}I - B)(I - P) = \hat{\alpha}E + \frac{\hat{\beta}}{n}E - \frac{e^T Be}{n^2}E \succeq 0,$$

which implies

$$\hat{\alpha} + \frac{\hat{\beta}}{n} \geq \frac{e^T Be}{n^2}.$$

It follows that

$$\hat{\alpha} + \hat{\beta} \geq \frac{e^T Be}{n^2} + \frac{(n-1)\hat{\beta}}{n} \geq \frac{e^T Be}{n^2} + \frac{(n-1)}{n}\lambda_{\max}(PBP).$$

Now let us choose

$$v = \frac{e^T Be}{n} e - Be, \quad \beta = \lambda_{\max}(PBP), \quad \alpha = \frac{e^T Be}{n^2} - \frac{\beta}{n}.$$

One can easily verify that (α, β, v) satisfy all the constraints in problem (33). Therefore, we can conclude that at the optimal solution of problem (33), it must hold

$$\hat{\alpha} = \frac{e^T Be}{n^2} - \frac{1}{n}\lambda_{\max}(PBP), \quad \hat{\beta} = \lambda_{\max}(PBP).$$

On the other hand, if $(\bar{\alpha}, \bar{\beta})$ is the optimal solution to problem (25), then by following a similar process, one can show that

$$\bar{\beta} \geq \lambda_{\max}(PBP) = \hat{\beta}.$$

This completes the proof of the lemma. □

Based on Lemma 3.5, we can establish the following result regarding the two lower bounds $\mu_1(\alpha, \bar{\beta}I)$ and $\mu_2(u, \hat{\beta}I)$.

Theorem 3.6. *Assume that $(\alpha, \bar{\beta}I)$ and $(u, \hat{\beta}I)$ are the minimal trace one-matrix splitting and sum-matrix splitting of B, respectively. Then we have*

$$\mu_2(u, \hat{\beta}I) \geq \mu_1(\alpha, \bar{\beta}I). \tag{34}$$

Proof. Let Y be an optimal solution of the problem (31). Then there exists X such that

$$\begin{aligned} & Y - Xue^T - eu^T X^T - X(\hat{\beta}I - B)X^T \succeq 0, \\ & \operatorname{diag}(Y) = 2Xu + X\operatorname{diag}(\hat{\beta}I - B), \\ & Ye = nXu + eu^T e + X(\hat{\beta}I - B)e, \\ & Xe = X^T e = e, \quad X \geq 0, \end{aligned} \tag{35}$$

and

$$\mu_2(u, \hat{\beta}I) = 2e^T AXu - \operatorname{Tr}(AY). \tag{36}$$

Since X is a $n \times n$ doubly stochastic matrix, similar to the proof of Theorem 2.5, we can infer that there exist $\lambda_i \geq 0, \forall i = 1, \ldots, |\Pi|$ such that

$$X = \sum_{i=1}^{|\Pi|} \lambda_i \hat{X}_i, \quad \sum_{i=1}^{|\Pi|} \lambda_i = 1, \quad \hat{X}_i \in \Pi \;\forall\, i,$$

which further implies

$$\mathrm{diag}(Xue^T) = \sum_{i=1}^{|\Pi|} \lambda_i \mathrm{diag}(\hat{X}_i ue^T) = \sum_{i=1}^{|\Pi|} \lambda_i \hat{X}_i \mathrm{diag}(ue^T) = \sum_{i=1}^{|\Pi|} \lambda_i \hat{X}_i u = Xu.$$

The above relation, together with $eu^T X^T = (Xue^T)^T$, yields

$$\mathrm{diag}(eu^T X^T) = \mathrm{diag}(Xue^T) = Xu. \tag{37}$$

Since X is a doubly stochastic matrix, we have

$$I - XX^T \succeq 0. \tag{38}$$

Let us define

$$\hat{Y} = Y - Xue^T - eu^T X^T + \alpha E + (\bar{\beta} - \hat{\beta})I. \tag{39}$$

It follows

$$\begin{aligned}
&\hat{Y} - \alpha E - X(\bar{\beta} I - B)X^T \\
&= Y - Xue^T - eu^T X^T - X(\hat{\beta} I - B)X^T + (\bar{\beta} - \hat{\beta})(I - XX^T).
\end{aligned}$$

By Lemma 3.5, $\bar{\beta} - \hat{\beta} \geq 0$, it follows immediately from (35), (37) and (38) that

$$\begin{aligned}
&\hat{Y} - \alpha E - X(\bar{\beta} I - B)X^T \succeq 0, \\
&\hat{Y} e = n\alpha e + X(\bar{\beta} I - B)e, \\
&\mathrm{diag}\,(\hat{Y}) = \alpha e + X \mathrm{diag}\,(\bar{\beta} I - B).
\end{aligned}$$

These, together with (35), imply that $\hat{Y} \in \mathcal{Y}_{(\alpha, \bar{\beta} I)}$. Thus

$$\alpha \mathrm{Tr}(AE) - \mathrm{Tr}\left(A\hat{Y}\right) \geq \mu_1(\alpha, \bar{\beta} I). \tag{40}$$

On the other hand, under Assumption 3.1, we can deduce from (39) that

$$\alpha \mathrm{Tr}(AE) - \mathrm{Tr}\left(A\hat{Y}\right) = 2e^T AXu - \mathrm{Tr}(AY),$$

which, together with (36) and (40), implies that

$$\mu_2(u, \hat{\beta} I) \geq \mu_1(\alpha, \bar{\beta} I). \qquad \square$$

Theorem 3.6 shows that under certain conditions, the SDR (31) based on the sum-matrix splitting is at least as good as the SDR (30) based on the one-matrix splitting. However, in our numerical experiments, we have observed that in most cases, we have $\mu_2(u, D) > \mu_1(\alpha, D)$. Theorem 3.6 provides a partial explanation for such a phenomenon. It should also be pointed out that, as illustrated by the numerical results in Section 5, there is no dominance relation between the bounds derived from the three different matrix splitting schemes described in this work.

4 SDRs of QAPs based on minimal trace matrix splitting

In this section, we present the SDR models of QAPs based on the three matrix splitting schemes discussed in Section 2 and 3 using the framework introduced in [28], which combine a technique to reduce the dimension of PSD constraints.

We first present the SDR model of QAP derived from the so-called orthogonal PSD matrix splitting (denoted by SDRMS-SVD) which can be derived by using the singular value decomposition (SVD) of B. Let (B_1, B_2) be the orthogonal PSD splitting of B. By Corollary 2.7, we have $\mathrm{Rank}(B_1) + \mathrm{Rank}(B_2) \le n$. Because $B_1 \succeq 0$ and $B_2 \succeq 0$, we have $B_i = \widehat{B}_i^T \widehat{B}_i$ for some $\widehat{B}_i \in \Re^{m_i \times n}$, $i = 1, 2$. Based on the well-known Schur complement lemma, the quadratic PSD constraints

$$Y_i - XB_iX^T \succeq 0, \quad i = 1, 2,$$

can equivalently be replaced by the PSD constraints of smaller scale

$$\begin{pmatrix} I_{m_i \times m_i} & \widehat{B}_i X^T \\ X\widehat{B}_i^T & Y_i \end{pmatrix}_{(m_i+n)\times(m_i+n)} \succeq 0, \quad i = 1, 2. \tag{41}$$

Note that we can write $B_i = \widehat{B}_i^T \widehat{B}_i$ by setting $\widehat{B}_i = \widehat{\Lambda}_i \widehat{V}_i^T$, where $\widehat{\Lambda}_i$ is an $m_i \times m_i$ diagonal matrix whose diagonal elements are the square roots of the non-zero eigenvalues of $\widehat{B}_i$, and $\widehat{V}_i$ is an $n \times m_i$ matrix whose columns are the corresponding eigenvectors.

Note that the new SDRMS-SVD model is different from the SDRMS-SVD model in [28] in two aspects: First the SDP constraints on Y_1 and Y_2 are simplified (see (41)) by using the rank information of B_1 and B_2, respectively. Secondly, we add extra constraints on the matrix $Y_1 + Y_2$. For self-completeness, we describe the full model below:

$$\begin{array}{lll}
\text{(SDRMS-SVD)} & \min & \mathrm{Tr}(A(Y_1 - Y_2)) \\
 & s.t. & \mathrm{diag}\,(Y_1) = X\mathrm{diag}\,(B_1), \quad Y_1 e = XB_1 e; \\
 & & \mathrm{diag}\,(Y_2) = X\mathrm{diag}\,(B_2), \quad Y_2 e = XB_2 e; \\
 & & (X\min([B_1]_{\mathrm{off}}))_i \le [Y_1]_{i,j} \le (X\max([B_1]_{\mathrm{off}}))_i, \quad \forall i \ne j \\
 & & (X\min([B_2]_{\mathrm{off}}))_i \le [Y_2]_{i,j} \le (X\max([B_2]_{\mathrm{off}}))_i, \quad \forall i \ne j \\
 & & (X\min([B]_{\mathrm{off}}))_i \le [Y_1 - Y_2]_{ij} \le (X\max([B]_{\mathrm{off}}))_i, \quad \forall i \ne j \\
 & & (X\min([B_1+B_2]_{\mathrm{off}}))_i \le [Y_1 + Y_2]_{ij} \le (X\max([B_1+B_2]_{\mathrm{off}}))_i, \ \forall i \ne j \\
 & & \mathcal{L}_2(Y_1) \le X\mathcal{L}_2(B_1), \quad \mathcal{L}_2(Y_2) \le X\mathcal{L}_2(B_2) \\
 & & \mathcal{L}_2(Y_1 - Y_2) \le X\mathcal{L}_2(B), \quad \mathcal{L}_2(Y_1 + Y_2) \le X\mathcal{L}_2(B_1 + B_2) \\
 & & \begin{pmatrix} I_{m_1 \times m_1} & \widehat{B}_1 X^T \\ X\widehat{B}_1 & Y_1 \end{pmatrix} \succeq 0, \quad \begin{pmatrix} I_{m_2 \times m_2} & \widehat{B}_2 X^T \\ X\widehat{B}_2 & Y_2 \end{pmatrix} \succeq 0 \\
 & & X \ge 0, \quad Xe = X^T e = e.
\end{array}$$

Next, we present the SDR model of QAPs based on minimal trace sum-matrix splitting (denoted by SDRMS-SUM). Let (u, D) be the optimal solution of problem (29). The minimal trace sum-matrix splitting of B is given by

$$B_1 = ue^T + eu^T + D, \quad B_2 = ue^T + eu^T + D - B \succeq 0.$$

We observe that for all the tested examples, the rank of B_2 tends to be much smaller than n. This is not surprising since in (29), we are minimizing the trace of B_2, which is a proxy for minimizing the rank of B_2. Let $\widehat{B}_2$ be a $m \times n$ matrix satisfying $B_2 = \widehat{B}_2^T \widehat{B}_2$. The SDRMS-SUM model is defined as follows:

$$
\begin{array}{lll}
\text{(SDRMS-SUM)} & \min & 2e^T AXu - \text{Tr}(AY_2) \\
& s.t. & \text{diag}\,(Y_2) = X\text{diag}\,(B_2), \quad Y_2 e = XB_2 e, \\
& & Y_1 = Xue^T + eu^T X^T + \text{diag}\,(Xd) \\
& & (X\min([B_2]_{\text{off}}))_i \leq [Y_2]_{i,j} \leq (X\max([B_2]_{\text{off}}))_i \quad \forall i \neq j \\
& & (X\min([B]_{\text{off}}))_i \leq [Y_1 - Y_2]_{i,j} \leq (X\max([B]_{\text{off}}))_i \quad \forall i \neq j \\
& & (X\min([B_1 + B_2]_{\text{off}}))_i \leq [Y_1 + Y_2]_{i,j} \leq (X\max([B_1 + B_2]_{\text{off}}))_i, \quad \forall i \neq j \\
& & \begin{pmatrix} I_{m\times m} & \widehat{B}_2 X^T \\ X\widehat{B}_2^T & Y_2 \end{pmatrix} \succeq 0, \quad \mathcal{L}_2(Y_2) \leq X\mathcal{L}_2(B_2) \\
& & \mathcal{L}_2(Y_1 - Y_2) \leq X\mathcal{L}_2(B), \quad \mathcal{L}_2(Y_1 + Y_2) \leq X\mathcal{L}_2(B_1 + B_2) \\
& & X \geq 0, \quad Xe = X^T e = e.
\end{array}
$$

We note that it is possible to obtain a tighter lower bound based on the intersection of the feasible regions of (SDRMS-SVD) and (SDRMS-SUM) by considering the two different splittings $(Y_1^{\text{SVD}}, Y_2^{\text{SVD}})$ and $(Y_1^{\text{sum}}, Y_2^{\text{sum}})$ in (SDRMS-SVD) and (SDRMS-SUM), respectively. However, our numerical experience show that the tighter lower bound is usually only slightly better than the best lower bound obtained from (SDPRMS-SVD) and (SDRMS-SUM) individually but the computational cost is substantially larger than the total cost of solving (SDRMS-SVD) and (SDRMS-SUM) individually. Thus in this paper, we shall not consider such a tighter lower bound.

Finally, the SDR of QAP based on minimal trace one-matrix splitting (denoted by SDRMS-ONE) discussed in the previous section is a variant of the models introduced in [25, 28]. Let (α, β) be the solution of problem (25). Then $B_2 = \alpha E + \beta I - B \succeq 0$. Again, we observed that for all the tested examples, $m = \text{Rank}(B_2) < n$. Let $\hat{B}_2$ be a $m \times n$ matrix such that $B_2 = \widehat{B}_2^T \widehat{B}_2$. The model is described below:

$$
\begin{array}{lll}
\text{(SDRMS-ONE)} & \min & \alpha\text{Tr}(AE) - \text{Tr}(AY) \\
& s.t. & \text{diag}\,(Y) = X\text{diag}\,(B_2), \quad Ye = XB_2 e \\
& & (X\min([B_2]_{\text{off}}))_i \leq [Y]_{i,j} \leq (X\max([B_2]_{\text{off}}))_i, \ \forall i \neq j \\
& & \begin{pmatrix} I_{m\times m} & \widehat{B}_2 X^T \\ X\widehat{B}_2^T & Y \end{pmatrix} \succeq 0, \quad \mathcal{L}_2(Y - \alpha E - \beta I) \leq X\mathcal{L}_2(B), \\
& & X \geq 0, \quad Xe = X^T e = e.
\end{array}
$$

5 Numerical Experiments

In this section, we report some numerical results of the three relaxation models based on non-redundant matrix splitting schemes given in the previous section on QAP instances from the

QAP library [7]. For a comparison between the SDRMS-SVD model and other existing relaxation models we refer to our earlier work [28]. We remind the readers that while the SDRMS-SVD is almost identical to the F-SVD model used in [28], the SDRMS-ONE model in this work is different from the model used in [25] in terms of the parameters, the cuts and constraints. We also mention that adding extra constraints to the SDRMS-ONE model as in [25] will further improve the lower bound. However, in our experiments, we found that those improvements are usually associated with the particular instances and thus might not be substantial for generic QAPs. On the other hand, it may increase the computational cost of the resulting relaxation model. The SDRMS-SUM bound is derived by applying the relaxation framework introduced in [28] to the new sum-matrix splitting scheme. As QAP(A,B) is equivalent to QAP(B,A), we compute the lower bounds for both orderings of the A, B matrices and report the stronger one only for all the three models.

In our experiments, all the problems were solved in MATLAB R2009b on a 3.33GHz Intel Core 2 Duo PC with 8GB memory. For QAPs of small and median sizes ($n \leq 70$), the SDR problems were automatically generated by CVX 1.2 [14] and solved by the SDP solver SDPT3 [34] (see Table 3 and Table 4). In our numerical experiments, we observed that CVX would consume more than 50% of the total computation time just to generate the SDP data for the relaxation problem when $n > 40$, and the amount of computer memory required by CVX becomes prohibitively large when $n > 70$, in addition to taking excessively long computer time to solve the resulting SDP problem. Thus for large scale QAPs ($n > 70$), the SDR problems were solved by using the new SDP solver – SDPNAL [36], which is designed to solve large scale SDP problems to moderate accuracy, and the input SDP data was coded on our own in order to control its structure (see Table 5). We should emphasize that our own routine substantially cut down the time taken and the memory needed to generate the SDP data as compared to that consumed by CVX.

In our numerical experiments, SDPNAL usually stops with an approximate SDP solution where the maximum of the relative primal infeasibility, dual infeasibility and duality gap is in the order to 10^{-5} to 10^{-6}. In such a case, we use the procedure described in [20] to find a rigorous lower bound for our relaxation model. In all the tables, the relative gap is computed by

$$R_{gap} = 1 - \frac{\text{Lower bound}}{\text{Optimal or best known feasible objective value}}$$

and the CPU time (in seconds) to compute the bound is listed under the column "CPU". We use the boldface font to highlight the strongest of the three bounds. We note that for several medium and large instances, the SDRMS-SUM bounds have exceeded the best-known bounds reported in QAPLIB [7]. We list those bounds in a separate table (Table 6) for ease of reference.

For the SDRMS-ONE model, we also report the matrix splitting parameters α, β in the tables. For the QAP instances that are associated with a Hamming or Manhattan distance matrix, the bounds computed by splitting the distance matrices are always better than the one based on the non-distance matrix. This confirms the results in [25] from a different perspective (see Tables 3, 4 and 5). The SDRMS-ONE bounds may be stronger than the SDRMS-SVD if the matrix splitting parameter $\beta \leq 0$ because the matrix splitting $(\alpha E + \beta I, \alpha E + \beta I - B)$ is a non-redundant matrix splitting of matrix B (see Tai20b, Tai25b in Table 3, Tai35b, Tai40b, Tai50b, Ste36c in Table 4 and Tai80b in Table 5). In some cases, the SDRMS-ONE bounds may be stronger than the SDRMS-SVD even if $\beta > 0$ (see Tai12b and Tai60b in Tables 3 and 4).

	OPT/Feas	SDRMS-SVD	SDRMS-ONE	Improved bounds
Tai12b	39464925	28523049	28863106	28957953
Tai60b	608215054	494775609	515731315	515943103

Table 2: Improvd bounds using non-redundant matrix splitting

This is because a new non-redundant matrix splitting $(\alpha E+\beta I-R, \alpha E+\beta I-B-R)$ can be obtained by solving the following SDP

$$\begin{array}{ll} \max & \mathrm{Tr}(R) \\ \text{s. t.} & R \succeq 0,\ \alpha E+\beta I-R \succeq 0, \\ & \alpha E+\beta I-B-R \succeq 0. \end{array}$$

For the instances Tai12b and Tai60b, their SDRMS-ONE bounds can thus be improved by using the new non-redundant matrix splitting according to Theorem 2.5. But the improvements of the bounds are marginal as one can see from Table 2 because β is very small (which shows the R-redundant matrix splitting is actually very "close" to its corresponding non-redundant matrix splitting).

For all the QAP instances tested, SDRMS-SUM bounds are always stronger than SDRMS-ONE (see Tables 3, 4 and 5). Theorem 3.6 provides an interesting explanation for such a phenomena. For most QAP instances, SDRMS-SUM bounds are stronger than the SDRMS-SVD bounds (see Tables 3 and 4). This is because the sum-matrix splitting is not only non-redundant, but also can reduce the contribution of the quadratic term in the objective function. One exception here is the TaiXXc instances (see Tai64c in Table 4 where the SDRMS-SVD bound is stronger than the SDRMS-SUM bound. This is possibly due to the fact that the matrix in the TaiXXc instances has a very specific sparse block structure and the orthogonal PSD matrix splitting can preserve such a desirable structure, while the sum matrix splitting fails to retain such a structure. The same reasoning can also be used to explain the SDRMS-ONE bounds for the TaiXXc instances (see the α, β values for Tai64c in Table 4.

In terms of computation time, we note that for small and medium scale instances, the SDRMS-SVD model is the most expensive, while the SDRMS-ONE model is the cheapest. This is not surprising due to their model complexities. Overall, SDRMS-SUM is usually preferred to SDRMS-SVD and SDRMS-ONE considering the quality of the bounds and the complexity of the model. In Table 4, we also compare the CPU time of two implementations for the SDRMS-SVD model, with and without using the rank information. As one can see from the table, the CPU time used by the implemented algorithm using the rank information is at most two-third of the CPU time of the implementation without using the rank information.

Table 5 has shown that SDPNAL is an very effective SDP solver for large scale SDP problems. Although, it usually does not provide an accurate solution to the SDP problem as SDPT3 does, the accuracy level is good enough for our purpose of estimating the lower bounds.

	SDRMS-SUM		SDRMS-SVD		SDRMS-ONE			
Prob.	R_{gap}	CPU	R_{gap}	CPU	R_{gap}	CPU	α	β
Chr12a	**11.02%**	8	11.41%	8	20.66%	4	42.5	86.0
Chr12b	24.65%	6	**17.55%**	6	30.03%	4	42.5	86.0
Chr12c	**11.87%**	6	12.46%	7	28.84%	4	42.5	86.0
Chr15a	24.80%	24	**21.76%**	11	32.50%	6	44.2	134.7
Chr15b	35.33%	10	**34.39%**	12	51.85%	6	44.2	134.7
Chr15c	**7.31%**	10	7.89%	12	18.88%	5	44.2	134.7
Had12	**1.63%**	6	2.90%	8	3.33%	4	4	0
Had14	**1.69%**	8	2.72%	11	3.05%	5	5	0
Had16	**2.45%**	11	3.09%	13	3.55%	6	5	0
Had18	**2.43%**	14	3.08%	18	3.53%	7	6	0
Nug15	**7.30%**	10	7.57%	12	7.83%	6	3	0
Nug16a	**7.02%**	11	8.14%	14	8.39%	6	3.5	0
Nug16b	**8.95%**	11	9.52%	13	10.24%	6	3	0
Nug17	**9.01%**	13	9.24%	11	9.70%	7	3.5	0
Nug18	**9.17%**	15	9.38%	17	9.74%	8	3.5	0
Nug20	**9.03%**	19	9.30%	24	9.69%	10	3.5	0
Nug21	**9.15%**	22	9.60%	28	10.09%	11	4	0
Nug22	**8.68%**	26	9.15%	33	9.54%	12	5.5	0
Nug24	**8.89%**	32	9.29%	40	9.69%	15	4	0
Nug25	**9.05%**	36	9.19%	46	9.75%	17	4	0
Nug27	**7.91%**	48	8.41%	60	8.79%	20	5	0
Nug28	**8.38%**	52	8.73%	65	8.96%	22	4.5	0
Had20	**2.07%**	18	2.86%	25	3.41%	10	6.5	0
Kra30a	**16.78%**	62	16.84%	82	17.72%	25	207.5	0
Kra30b	**17.68%**	63	17.76%	83	18.69%	22	207.5	0
Nug12	**9.34%**	6	10.03%	7	10.72%	4	2.5	0
Nug14	**6.80%**	8	7.10%	10	7.20%	5	3	0
Nug30	**8.43%**	67	8.54%	79	8.93%	28	4.5	0
Rou12	13.40%	6	**11.16%**	7	14.50%	4	50.5	122.8
Rou15	14.77%	10	**13.20%**	11	15.77%	5	47.5	134.1
Rou20	16.87%	20	**16.02%**	24	17.70%	10	53.8	152.2
Scr12	**7.25%**	6	7.85%	7	8.79%	4	2.5	0
Scr15	**10.02%**	9	10.53%	12	11.70%	6	3	0
Scr20	**16.00%**	20	16.39%	24	17.28%	10	3.5	0
Tai12b	**8.61%**	6	23.69%	7	23.71%	4	210.3	0.7
Tai20b	**6.39%**	20	46.16%	28	30.53%	11	420.5	-2.5
Tai25b	**15.15%**	40	23.70%	54	20.58%	18	558.5	-2.5
Tai30b	**12.72%**	68	14.68%	94	14.82%	31	666.1	0.74
Tho30	**12.24%**	67	13.09%	79	13.62%	27	5.5	0

Table 3: Selected bounds for QAPs of small sizes ($n \leq 30$), computed using CVX with the SDPT3 solver.

	SDRMS-SUM		SDRMS-SVD		SVD-FULL		SDRMS-ONE			
Prob.	R_{gap}	CPU	R_{gap}	CPU	R_{gap}	CPU	R_{gap}	CPU	α	β
Kra32	**18.49%**	52	18.59%	102	18.59%	170	19.27%	28	207.5	0
Sko42	**7.58%**	187	7.94%	356	7.94%	685	8.25%	95	5.5	0
Sko49	**6.92%**	406	7.35%	733	7.35%	1533	7.68%	241	6	0
Sko56	**6.87%**	736	7.11%	1546	7.11%	3171	7.48%	481	6.5	0
Sko64	**6.29%**	1441	6.61%	3242	6.61%	5671	6.78%	915	7	0
Ste36a	**18.83%**	83	19.34%	167	19.34%	318	19.60%	55	5.5	0
Ste36b	**18.41%**	66	23.71%	155	23.72%	303	58.62%	55	49.9	36.1
Ste36c	**14.33%**	81	17.74%	210	17.74%	333	15.30%	57	5390.6	-380.6
Tai35b	**13.78%**	75	21.47%	170	21.47%	224	16.83%	55	584.2	-2.1
Tai40b	**11.13%**	131	14.30%	335	14.30%	444	13.98%	82	797.8	-3.6
Tai50b	**13.61%**	403	16.70%	1131	16.70%	1830	15.58%	200	969.2	-5.3
Tai60b	**10.18%**	928	18.32%	3394	18.32%	4714	14.96%	481	1090.5	0.5
Tai64c	60.92%	1226	**2.40%**	1243	NA	NA	73.73%	400	0.5	11.0
Tho40	**12.60%**	191	13.20%	274	13.20%	539	13.53%	71	5.5	0
Wil50	**3.92%**	477	4.07%	821	4.07%	1473	4.35%	181	6.5	0

Table 4: Selected bounds for QAPs of median sizes ($30 < n \leq 70$), computed using CVX with the SDPT3 solver. 'NA' denote the solver failed due to out of memory.

problem	best solution	best lower bound based on SDRMS-SUM SDRMS-SVD	SDRMS-SUM $R_{gap}(\%)$ | CPU	SDRMS-SVD $R_{gap}(\%)$ | CPU
sko72	† 6.62560000 4	6.24190000 4	**5.79** | 11:41	5.99 | 16:22
sko81	† 9.09980000 4	8.57920000 4	**5.72** | 16:15	5.95 | 22:23
sko90	† 1.15534000 5	1.09251000 5	**5.44** | 22:51	5.61 | 31:42
sko100a	† 1.52002000 5	1.44088000 5	**5.21** | 31:09	5.39 | 42:34
sko100b	† 1.53890000 5	1.45645000 5	**5.48** | 32:00	5.54 | 42:25
sko100c	† 1.47862000 5	1.40110000 5	**5.24** | 13:52	5.53 | 42:34
sko100d	† 1.49576000 5	1.41513000 5	**5.39** | 9:52	5.62 | 42:30
sko100e	† 1.49150000 5	1.41248000 5	**5.30** | 12:39	5.57 | 40:55
sko100f	† 1.49036000 5	1.40848000 5	**5.49** | 13:02	5.80 | 42:42
tai80b	† 8.18415043 8	7.23968001 8	**11.54** | 17:55	16.26 | 25:36
tai100b	† 1.18599614 9	1.06455129 9	**10.24** | 36:32	18.71 | 52:08
tai150b	† 4.98896643 8	4.42788590 8	**11.36** | 2:44:55	11.71 | 2:32:29
tho150	† 8.13339800 6	7.61238100 6	**6.46** | 32:59	6.69 | 2:26:48
wil100	† 2.73038000 5	2.64720000 5	**3.05** | 30:44	3.14 | 42:48
tai256c	† 4.47592940 7	4.38491950 7	73.55 | 6:53:05	**2.03** | 1:04:09

Table 5: Selected bounds for QAPs of large sizes ($n > 70$), computed using SDPNAL. The symbol (†) means that the best solution is only a feasible solution. The computation time is reported in the format of hours:minutes:seconds.

	best feasible solution	best lower bound computed in this paper	R_{gap}(%) | CPU
sko90	† 1.15534000 5	1.09251000 5 (SUM)	5.44 | 22:51
sko100a	† 1.52002000 5	1.44088000 5 (SUM)	5.21 | 31:09
sko100b	† 1.53890000 5	1.45645000 5 (SUM)	5.48 | 32:00
sko100c	† 1.47862000 5	1.40110000 5 (SUM)	5.24 | 13:52
sko100d	† 1.49576000 5	1.41513000 5 (SUM)	5.39 | 9:52
sko100e	† 1.49150000 5	1.41248000 5 (SUM)	5.30 | 12:39
sko100f	† 1.49036000 5	1.40848000 5 (SUM)	5.49 | 13:02
tai35b	† 2.83315445 8	2.46180672 8 (SUM)	13.80 | 1:08
tai40b	† 6.37250948 8	5.66070838 8 (SUM)	11.17 | 1:48
tai50b	† 4.58821517 8	3.98749304 8 (SUM)	13.67 | 3:30
tai60b	† 6.08215054 8	5.45971917 8 (SUM)	10.23 | 7:01
tai80b	† 8.18415043 8	7.23968001 8 (SUM)	11.54 | 17:55
tai100b	† 1.18599614 9	1.06455129 9 (SUM)	10.24 | 36:32
tai150b	† 4.98896643 8	4.42788590 8 (SUM)	11.36 | 2:44:55
tho150	† 8.13339800 6	7.61238100 6 (SUM)	6.46 | 32:59
wil100	† 2.73038000 5	2.64720000 5 (SUM)	3.05 | 30:44

Table 6: New bounds for selected QAPLIB instances. The previously best known bounds for these instances were reported in the QAP library [16] and the paper [28]. In particular, the QAP library has updated the bounds based on the numerical results in an earlier version of this paper where some constraints in the SDRMS-SUM model were missing.

6 Conclusions

In this paper, we considered the issue of how to choose an appropriate matrix splitting scheme so that the resulting SDR for QAPs can provide a strong lower bound. To obtain such a desirable relaxation, we introduced the notion of redundant and non-redundant matrix splitting and showed that for every redundant splitting, there is a corresponding non-redundant splitting whose resulting SDR can provide a stronger bound. To find a non-redundant matrix splitting, we proposed to solve some auxiliary SDP problems. The properties of the optimal solutions to these SDP problems were investigated. These explored properties not only help to select the matrix splitting scheme, but also lead to a more concise and effective implementation of the relaxation model.

A new SDR for QAPs based on the sum matrix and the minimal trace principle was derived. It was shown that in special cases, the new SDR can provide a stronger bound than the one from the one-matrix splitting. Numerical results also indicates that for most tested instances, the new SDR can provide stronger bounds than the those based on two other matrix splitting schemes.

On the other hand, we should point out that although in this paper we have presented several ways to select a non-redundant PSD matrix splitting scheme to construct a strong SDR, it remains an open question on how to find the strongest SDR based on matrix splitting. Such a difficulty is possibly due to the multiplicity of the non-redundant splitting schemes. Even for the three selected splitting schemes, we could not find any dominance relationship among them. Further study is needed to address such an issue.

Acknowledgement

We would like to thank the two anonymous referees and the Associate Editor for their helpful suggestions that led to substantial improvement on the presentation of the paper. We also thank Etienne de Klerk for pointing out some missing constraints in our previous implementation of the SDRMS-SUM model in CVX.

References

[1] W. Adams, M. Guignard, P. Hahn, and W. Hightower. A level-2 reformulation-linearization technique bound for the quadratic assignment problem. *European Journal Operations Research*, 180:983–996, 2007.

[2] W. Adams and T. Johnson. Improved linear programming-based lower bounds for the quadratic assignment problem. pages 43–75, 1994.

[3] K. Anstreicher and N. Brixius. A new lower bound via projection for the quadratic assignment problem. *Mathematical Programming*, 80:341–357, 2001.

[4] G. Ben-David and D. Malah. Bounds on the performance of vector-quantizers under channel errors. *Information Theory, IEEE Transactions on*, 51(6):2227–2235, 2005.

[5] S. Burer and D. Vandenbussche. Solving lift-and-project relaxations of binary integer programs. *SIAM Journal on Optimization*, 16(3):726–750, 2006.

[6] R. Burkard, M. Dell'Amico, and S. Martello. *Assignment Problems.* Society for Industrial and Applied Mathematics, Philadelphia, 2009.

[7] R. Burkard, S. Karisch, and F. Rendl. Qaplib–a quadratic assignment problem library. *Journal of Global Optimization*, 10(4):391–403, 1997. Recent updates on QAPLIB are avaliable at *http://www.seas.upenn.edu/qaplib/.*

[8] K. Conrad. *Das quadratische Zuweisungsproblem und zwei seiner Spezialfalle.* Mohr Siebeck, Tubingen, 1971.

[9] S. de Carvalho Jr and S. Rahmann. Microarray layout as a quadratic assignment problem. In *Proceedings of the German Conference on Bioinformatics*, volume 83, pages 11–20, 2006.

[10] E. De Klerk and R. Sotirov. Exploiting group symmetry in semidefinite programming relaxations of the quadratic assignment problem. *Mathematical Programming*, 122(2):225–246, 2010.

[11] Y. Ding and H. Wolkowicz. A low-dimensional semidefinite relaxation for the quadratic assignment problem. *Mathematics of Operations Research*, 34(4):1008–1022, 2009.

[12] C. Edwards. A branch and bound algorithm for the Koopmans-Beckmann quadratic assignment problem. *Combinatorial Optimization II*, pages 35–52, 1980.

[13] P. Gilmore. Optimal and suboptimal algorithms for the quadratic assignment problem. *SIAM Journal on Applied Mathematics*, 10:305–313, 1962.

[14] M. Grant, S. Boyd, and Y. Ye. CVX: Matlab software for disciplined convex programming. *Avialable at http://www. stanford. edu/boyd/cvx.*

[15] P. Hahn and T. Grant. Lower bounds for the quadratic assignment problem based upon a dual formulation. *Operations Research*, 46:912–922, 1998.

[16] Peter Hahn, Miguel Anjos, R.E. Burkard, S.E. Karisch and F. Rendl, *QAPLIB – A Quadratic Assignment Problem Library. Avialable at http://www.seas.upenn.edu/qaplib/.*

[17] P. M. Hahn, Y.-R. Zhu, M. Guignard, W.L. Hightower, and M.J. Saltzman. A Level-3 Reformulation-Linearization Technique-Based Bound for the Quadratic Assignment Problem. *INFORMS Journal on Computing*, to appear.

[18] M. Hanan and J. Kurtzberg. Placement techniques. *Design automation of digital systems*, 1:213–282, 1972.

[19] R. A. Horn and C. R. Johnson. *Matrix Analysis.* Cambridge University Press, Cambridge, UK, 1985.

[20] C. Jansson, D. Chaykin, and C. Keil. Rigorous error bounds for the optimal value in semidefinite programming. *SIAM Journal on Numerical Analysis*, 46:180–200, 2007.

[21] S.E. Karisch and F. Rendl. Lower bounds for the quadratic assignment problem via triangle decompositions. *Mathematical Programming*, 71:137-152, 1995.

[22] T. Koopmans and M. Beckmann. Assignment problems and the location of economic activities. *Econometrica*, 25(1):53–76, 1957.

[23] E. Lawler. The quadratic assignment problem. *Management Science*, 9:589–599, 1963.

[24] E. Loiola, N. de Abreu, P. Boaventura-Netto, P. Hahn, and T. Querido. A survey for the quadratic assignment problem. *European Journal of Operational Research*, 176(2):657–690, 2007.

[25] H. Mittelmann and J. Peng. Estimating Bounds for Quadratic Assignment Problems Associated with Hamming and Manhattan Distance Matrices Based on Semidefinite Programming. *SIAM Journal on Optimization*, 20:3408, 2010.

[26] L. Mukherjee, V. Singh, J. Peng, J. Xu, M. Zeitz, and R. Berezney. Generalized median graphs and applications. *Journal of Combinatorial Optimization*, 17(1):21–44, 2009.

[27] M. A. Nielsen and G. Vidal. Majorization and the interconversion of bipartite states. *Quantum Information and Computation*, 1(1):76–93, 2001.

[28] J. Peng, H. Mittelmann, and X. Li. A new relaxation framework for quadratic assignment problems based on matrix splitting. *Mathematical Programming Computation*, 2(1):59–77, 2010.

[29] F. Rendl and R. Sotirov. Bounds for the quadratic assignment problem using the bundle method. *Mathematical programming*, 109(2):505–524, 2007.

[30] C. Roucairol. A reduction method for quadratic assignment problems. *Methods of Operations Research*, 32:185–187, 1979.

[31] F. R. S.W. HADLEY and H. WOLKOWICZ. A new lower bound via projection for the quadratic assignment problem. *Mathematics of Operations Research*, 17:727–739, 1992.

[32] E. Taillard. Comparison of iterative searches for the quadratic assignment problem. *Location science*, 3(2):87–105, 1995.

[33] C. M. Theobald. An inequality for the trace of the product of two symmetric matrices. *Math. Proc. Cambridge Philos. Soc.*, 77:77–265, 1975.

[34] K. Toh, M. Todd, and R. Tutuncu. SDPT3: Matlab software package for semidefinite programming. *Optimization Methods and Software*, 11(12):545–581, 1999.

[35] Q. Zhao, S. Karisch, F. Rendl, and H. Wolkowicz. Semidefinite programming relaxations for the quadratic assignment problem. *J. Combinatorial Optimization*, 2:71–109, 1998.

[36] X. Zhao, D. Sun, and K. Toh. A Newton-CG augmented Lagrangian method for semidefinite programming. *SIAM J. Optim*, 20(4):1737–1765, 2010.

7 Appendix 1: Proof of Theorem 3.2

We first cite a well-known result for matrix product from [33] that will be used in the proof of Theorem 3.2.

Lemma 7.1. *Let $A, B \in \mathrm{S}^n$ with the eigenvalues $\lambda_i(A)$ and $\lambda_i(B)$, $i = 1, \ldots, n$ listed in nonincreasing order. Then*

$$\mathrm{Tr}(AB) \leq \sum_{i=1}^{n} \lambda_i(A)\lambda_i(B),$$

where the equality holds if and only if there is an orthogonal matrix P whose columns form a common set of eigenvectors for A and B and are ordered with respect to $\{\lambda_i(A)\}_{i=1}^n$ and $\{\lambda_i(B)\}_{i=1}^n$, such that $P^{-1}AP$ and $P^{-1}BP$ are diagonal.

Now we are ready to prove Theorem 3.2. Since (α, β) is an optimal solution of problem (25), there exists $U \in \mathrm{S}^n$ such that

$$n - \mathrm{Tr}(UE) = 0, \tag{42}$$

$$n - \mathrm{Tr}(U) = 0, \tag{43}$$

$$\mathrm{Tr}(U(\alpha E + \beta I - B)) = 0, \tag{44}$$

$$U \succeq 0, \quad \alpha E + \beta I - B \succeq 0. \tag{45}$$

From (42)-(44), we obtain directly

$$n(\alpha + \beta) = \mathrm{Tr}(UB). \tag{46}$$

From (44) and (45), we follow that

$$U(\alpha E + \beta I - B) = (\alpha E + \beta I - B)U = 0. \tag{47}$$

This implies that U and $\alpha E + \beta I - B$ can commute. By Theorem 1.3.12 in [19], U and $\alpha E + \beta I - B$ are simultaneously diagonalizable. Since $U \in \mathrm{S}^n$, $\alpha E + \beta I - B \in \mathrm{S}^n$, there is an orthogonal matrix P such that $P^{-1}UP$ and $P^{-1}(\alpha E + \beta I - B)P$ are diagonal. So, we have

$$\mathrm{Tr}(U(\alpha E + \beta I - B)) = \sum_{i=1}^{n} \lambda_i(U)\lambda_i(\alpha E + \beta I - B) = 0,$$

which in turn by (45) implies that

$$\lambda_i(U)\lambda_i(\alpha E + \beta I - B) = 0, \; i = 1, \ldots, n. \tag{48}$$

Due to the minimal trace principle, we have $m = \mathrm{Rank}(\alpha E + \beta I - B) < n$. Since $\alpha E + \beta I - B \succeq 0$, we assume that $\lambda_i(\alpha E + \beta I - B) > 0$, $i = 1, \ldots, m$. The above equality (48) then yields $\lambda_i(U) = 0$, $i = 1, \ldots, m$.

We now prove $UE \neq EU$. Suppose to the contrary that $UE = EU$. By Theorem 1.3.12 in [19], U and E are simultaneously diagonalizeable. Let

$$\lambda_1(U) = \ldots = \lambda_s(U) = 0 < \lambda_{m+1}(U) \leq \ldots \leq \lambda_n(U). \tag{49}$$

Note that the eigenvalues of E are $0, \ldots, 0, n$. Therefore, we have

$$\mathrm{Tr}(UE) = n\lambda_n(U),$$

which by (42) implies $\lambda_n(U) = 1$. Hence, we infer from (49) that

$$\mathrm{Tr}(U) = \sum_{i=1}^{n} \lambda_i(U) \leq n - m < n,$$

this contradicts (43).

Because $UE \neq EU$, from (47) we obtain $UB \neq BU$. Since $U \in \mathrm{S}^n$ and $B \in \mathrm{S}^n$, by Theorem 1.3.12 in [19], U and B are not simultaneously diagonalizable. Now using Lemma 7.1, we have

$$\mathrm{Tr}(UB) < \sum_{i=1}^{n} \lambda_i(U)\lambda_i(B),$$

which, together with (46), yields

$$n(\alpha + \beta) < \sum_{i=1}^{n} \lambda_i(U)\lambda_i(B). \tag{50}$$

Let $\lambda_{\max}(B)$ be the largest eigenvalue of B. Note that $\sum_{i=1}^{n} \lambda_i(U) = \mathrm{Tr}(U) = n$. Also, $\lambda_i(U) \geq 0$ for all i since $U \succeq 0$. It then follows from (50) that

$$\alpha + \beta < \lambda_{\max}(B). \tag{51}$$

On the other hand, from (45), we have

$$B - \alpha(E - I) - (\alpha + \beta)I \preceq 0.$$

This means that

$$\alpha + \beta \geq \lambda_{\max}(B - \alpha(E - I)). \tag{52}$$

If $\alpha = 0$, then the combination of (51) and (52) leads to a contradiction.

If $\alpha < 0$. Let $\rho(B)$ be the spectral radius of B. Since $B \in \mathrm{S}^n$ is non-negative, by Theorem 8.3.1 in [19], then $\rho(B)$ is an eigenvalue of B and there exists nontrivial $\hat{x} \geq 0 \in \Re^n$ such that $B\hat{x} = \rho(B)\hat{x}$. Without loss of generality, we can further assume that $\|\hat{x}\|_2 = 1$. Thus we have $\hat{x}^T B\hat{x} = \rho(B)$. Since $\hat{x} \geq 0$, it holds $\hat{x}^T(E - I)\hat{x} \geq 0$. It follows from (52) that

$$\begin{aligned}
\alpha + \beta &\geq \lambda_{\max}(B - \alpha(E - I)) \\
&= \max\left\{x^T(B - \alpha(E - I))x : x^T x = 1\right\} \\
&\geq \hat{x}^T(B - \alpha(E - I))\hat{x} \\
&\geq \hat{x}^T B\hat{x} = \rho(B) \\
&\geq \lambda_{\max}(B),
\end{aligned}$$

which contradicts to (51). Therefore, we can conclude $\alpha > 0$. This finishes the proof of the theorem. $\square$