

A new semidefinite programming relaxation for the quadratic assignment problem and its computational perspectives

E. de Klerk, R. Sotirov, U. Truetsch

June 21, 2013

Abstract

Recent progress in solving quadratic assignment problems (QAPs) from the QAPLIB test set has come from mixed integer linear or quadratic programming models that are solved in a branch-and-bound framework. Semidefinite programming bounds for QAP have also been studied in some detail, but their computational impact has been limited so far, mostly due to the restrictive size of the early relaxations. Some recent progress has been made by studying smaller SDP relaxations and by exploiting group symmetry in the QAP data. In this work we introduce a new SDP relaxation where the matrix variables are only of the order of the QAP dimension, and we show how one may exploit group symmetry in the problem data for this relaxation. We also provide a detailed numerical comparison with related bounds from the literature.

Keywords: quadratic assignment problem, semidefinite programming bounds.

AMS subject classification: 90C22, 90C27

Introduction

In this work we will study the quadratic assignment problem (QAP), introduced in 1957 by Koopmans and Beckmann [17], in its trace formulation:

$$\min_{X \in \Pi_n} \text{tr}(AXBX^T),$$

where $A, B \in \mathbb{S}^{n \times n}$ (symmetric $n \times n$ matrices), and Π_n denotes the set of $n \times n$ permutation matrices.

It is well-known that the QAP is an NP-hard problem, and difficult to solve in practice, even for values of $n \leq 30$. As the authors of the recent book [8] aptly put it:

‘All main algorithmic techniques for the exact solution of NP-hard problems have been used for attacking the QAP. The QAP, however, bravely resisted.’

Computational progress has been measured (and driven) in recent years by the success in solving QAP instances from the QAPLIB library [9]. Notable progress in the last decade or so includes work by Nyberg and Westerlund [20], and Fischetti, Monaci and Salvagnin [13], who used mixed integer linear programming (MILP) formulations, and by Anstreicher et al. [5, 7, 4, 3] who used convex quadratic programming (QP) relaxations in a branch-and-bound setting to solve previously unsolved QAPLIB instances. A detailed review of computational progress up to 2009 is given in [8, Chapter 8].

A natural question is whether other convex programming bounds (than LP or QP) also offer computational perspectives. In particular, semidefinite programming (SDP) bounds for QAPs have been studied extensively, starting with the seminal work of Zhao, Karisch, Rendl and Wolkowicz [26] in 1998. The problem with the relaxation studied by these authors is that it involves a matrix variable of order n^2 , and can therefore only be solved efficiently by interior point methods for $n \leq 15$, say. This limitation has prompted research into exploiting group symmetry of the QAP data matrices A and B to obtain smaller SDP problems; see De Klerk and Sotirov [10, 11]. It has also prompted recent research into SDP relaxations of QAP where the matrix variables are of order n ; see Mittelman and Peng [19] and Peng et al. [22, 21]. In both these lines of research the authors were able to compute the best-known lower bounds for some QAPLIB instances.

The work presented in this paper is in the spirit of these earlier works on SDP relaxations, and the contributions are threefold:

- We introduce a new SDP relaxation (called the *eigenspace relaxation*) where the matrix variables are of order n ;
- We compare the performance of the new relaxation to other convex relaxations for the QAPLIB test set, and do a theoretical comparison of some SDP bounds as well;
- We show how to exploit group symmetry in the QAP data matrices for the new relaxation, and illustrate the theory with the example of the QAP reformulation of the traveling salesman problem (TSP).

Our numerical results show that the new relaxation compares favourably to the best SDP bound from [22, 21] for $n \leq 25$. Moreover, the new relaxation is more suitable for exploiting group symmetry in the QAP data.

The ultimate aim of our work is to present an appropriate choice of an SDP relaxation that can be successfully used at a given node within a branch-and-bound framework, where both the size and the group symmetry properties of the underlying QAP problem are taken into account.

1 Semidefinite programming relaxations of the QAP

In this section we review two SDP bounds from the literature and introduce a new relaxation, called the *eigenspace relaxation*.

1.1 The QAP-R₃ relaxation by Zhao, Karish, Rendl and Wolkowicz

In 1998, Zhao, Karish, Rendl and Wolkowicz [26] derived a seminal semidefinite relaxation of the QAP, called QAP-R₃. The resulting lower bounds have proved to be quite strong in practice. Here we will use its reformulation by Povh and Rendl [23].

Before doing so, recall that the Kronecker product $C \otimes D$ of matrices $C = (C_{ij}) \in \mathbb{R}^{m \times n}$ and $D = (D_{ij}) \in \mathbb{R}^{r \times s}$ is the $mr \times ns$ block matrix with block (i, j) given by $C_{ij}D$ ($i = 1, \dots, m, j = 1, \dots, n$). We will also use the notation E_{ij} to denote a matrix with (i, j) th entry equal to 1, and all other entries zero, and J and I to denote the all-ones and identity matrices of order n , respectively.

$$\left. \begin{array}{ll}
\min_{\Upsilon \in \mathbb{R}^{n^2 \times n^2}} & \text{trace}([B \otimes A] \Upsilon) \\
s.t. & \text{trace}([I \otimes E_{ii}] \Upsilon) = 1 \quad i = 1, \dots, n, \\
& \text{trace}([E_{ii} \otimes I] \Upsilon) = 1 \quad i = 1, \dots, n, \\
& \text{trace}((J \otimes J) \Upsilon) = n^2, \\
& \text{trace}(\mathcal{G} \Upsilon) = 0, \\
& \Upsilon \succeq 0, \\
& \Upsilon \geq 0,
\end{array} \right\} \quad (\text{QAP-R}_3)$$

where $\mathcal{G} := [I \otimes (J - I) + (J - I) \otimes I]$.

If we define $\text{vec}(\cdot)$ as the operator that maps an $n \times n$ matrix to an n^2 -vector by stacking its columns, then the matrix variable Υ may be viewed as a relaxation of $\text{vec}(X) \text{vec}(X)^\top$ for $X \in \Pi_n$. Thus we may view Υ as having the following block structure:

$$\Upsilon := \begin{pmatrix} \Upsilon^{(11)} & \dots & \Upsilon^{(1n)} \\ \vdots & \ddots & \vdots \\ \Upsilon^{(n1)} & \dots & \Upsilon^{(nn)} \end{pmatrix}, \quad (1)$$

where $\Upsilon^{(ij)} \in \mathbb{R}^{n \times n}$ ($1 \leq i, j \leq n$). The off-diagonal elements of the blocks $\Upsilon^{(ii)}$ are zero ($i = 1, \dots, n$), while the $\text{diag}(\Upsilon^{(ij)}) = 0$ if $i \neq j$, where $\text{diag} : \mathbb{R}^{n \times n} \rightarrow \mathbb{R}^n$ is the operator that maps an $n \times n$ matrix to its diagonal.

We will also use the following properties for feasible solutions Υ .

Theorem 1.1 ([23]). *A matrix Υ that is feasible for (QAP-R₃) satisfies*

- (i) $\Upsilon - \text{diag}(\Upsilon) \text{diag}(\Upsilon)^\top \succeq 0$,
- (ii) $\text{trace}(\Upsilon^{(ii)}) = 1 \quad i, \quad \sum_{i=1}^n \Upsilon^{(ii)} = I$,
- (iii) $\Upsilon^{(ij)} e = \text{diag}(\Upsilon^{(ii)}) \quad i, j$,
- (iv) $\sum_{i=1}^n \Upsilon^{(ij)} = e \text{diag}(\Upsilon^{(jj)})^\top \quad j$,

where e denotes the all-ones vector.

These observations will be used later when comparing different relaxations.

The price of using the strong (QAP-R₃) relaxation is the high computational effort required for QAP instances of size $n > 15$, since the matrix variable Υ is of order n^2 . The aim in this work is to study alternative SDP relaxations that can be used for higher dimensional instances.

1.2 Minimal trace sum-matrix splitting (SDRMS-sum) by Mittelman, Peng et al.

Recently Mittelman, Peng et al. [19, 21] introduced a new class of semidefinite programming relaxations which is based on a matrix splitting approach. In the subsequent work of Peng et al. [22], a new matrix splitting variant is considered and a detailed numerical analysis is given. The main advantage of these relaxations is that they involve matrix variables of order n (as opposed to n^2 for (QAP-R₃)).

In our work we concentrate on the minimal trace sum-matrix splitting introduced by Peng et al. [22] which seems to offer the most promising results. To this end, recall that a matrix M is called a sum-matrix if

$$M = ue^\top + eu^\top$$

for some $u \in \mathbb{R}^n$.

Mittelman, Peng et al. [22, 21] propose a sum-matrix splitting:

$$B = B_1 - B_2, \quad (2)$$

where B_2 is a positive semidefinite matrix and

$$B_1 := ue^\top + eu^\top + \text{Diag}(d) \quad (3)$$

with $u, d \in \mathbb{R}^n$, and where the operator $\text{Diag} : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times n}$ maps an n -vector to an $n \times n$ diagonal matrix in the obvious way.

The idea is that one now has, for $X \in \Pi_n$,

$$XBX^T = Xue^T + eu^T X^T + \text{Diag}(Xd) - XB_2X^T,$$

and that the quadratic term may be linearized by using the fact that $B_2 \succeq 0$. Introducing a new variable Y_2 that corresponds to XB_2X^T , one may replace the condition $Y_2 - XB_2X^T = 0$ by the tractable condition $Y_2 - XB_2X^T \succeq 0$. The latter is indeed tractable, since it may be rewritten as the linear matrix inequality

$$\begin{bmatrix} I_m & (\hat{B}_2 X)^T \\ \hat{B}_2 X & Y_2 \end{bmatrix} \succeq 0,$$

where $B_2 = \hat{B}_2^\top \hat{B}_2$, by the Schur complement theorem.

The matrix B_2 and the vectors u, d are obtained in [22] by solving the auxiliary problem:

$$\min_{u, d \in \mathbb{R}^n} \left\{ \sum_{i=1}^n (2u_i + d_i) \mid B_2 = ue^\top + eu^\top + \text{Diag}(d) - B \succeq 0 \right\}.$$

Defining $B_s = B_1 + B_2$, one may introduce variables Y and Y_s that correspond to XBX^T and XB_sX^T , respectively, in the same way as for Y_2 .

Peng et al. [22] introduced additional constraints based on the observation that, if $Y = XBX^T$, then the i th row of Y is a permutation of the i th row of XB (for any $i \in \{1, \dots, n\}$).

In particular, the minimum off-diagonal element of row i of Y equals the minimum off-diagonal element of row i of XB . This may be represented by requiring,

$$(X \min B_{\text{off}}^+)_i \leq Y_{ij} \leq (X \max B_{\text{off}}^-)_i \quad i \neq j, \quad (4)$$

where the max (respectively min) operator of any matrix M is defined as a column vector where the i th entry equals the maximal (respectively minimal) element of the i th row of M , and

$$B_{\text{off}}^\pm := B \pm \|B\| I.$$

Analogously, we may include similar inequalities for Y_s and Y_2 .

Similarly, we may use the fact that the norms of the i th row of Y and the i th row of XB should be equal. Since this condition is not convex, it can be relaxed to the second order cone constraint that requires that the norm of the i th row of Y should be at most the norm of the i th row of XB .

We represent this constraint by defining $\mathcal{L}_2(M)$ as the n -vector of row norms of $M \in \mathbb{R}^{n \times n}$, so that the relevant constraint becomes:

$$\mathcal{L}_2(Y) \leq X \mathcal{L}_2(B). \quad (5)$$

As before, we may include the analogous constraints for Y_2 and Y_s .

Thus the final SDRMS-sum relaxation from [22] takes the form:

$$\begin{array}{l}
\min_{\substack{Y, Y_s, Y_2 \in \mathbb{S}^{n \times n} \\ X \in \mathbb{R}^{n \times n}}} \text{trace}(AY) \\
s.t. \quad \left. \begin{array}{l}
\begin{bmatrix} I_m & (\hat{B}_2 X)^\top \\ \hat{B}_2 X & Y_2 \end{bmatrix} \succeq 0, \\
Xue^\top + eu^\top X^\top + \text{diag}(Xd) - Y = Y_2, \\
Xue^\top + eu^\top X^\top + \text{diag}(Xd) + Y_2 = Y_s, \\
Xe = X^\top e = e; \quad X \geq 0, \\
\text{diag}(Y_2) = X \text{diag} B_2, \\
Y_2 e = X B_2 e, \\
\mathcal{L}_2(Y) \leq X \mathcal{L}_2(B); \quad \mathcal{L}_2(Y_2) \leq X \mathcal{L}_2(B_2); \quad \mathcal{L}_2(Y_s) \leq X \mathcal{L}_2(B_s), \\
\left(X \min B_{off}^+ \right)_i \leq Y_{ij} \leq \left(X \max B_{off}^- \right)_i \quad i \neq j, \\
\left(X \min B_{2off}^+ \right)_i \leq Y_{2ij} \leq \left(X \max B_{2off}^- \right)_i \quad i \neq j, \\
\left(X \min B_{s_{off}}^+ \right)_i \leq Y_{s_{ij}} \leq \left(X \max B_{s_{off}}^- \right)_i \quad i \neq j.
\end{array} \right\} \text{(SDRMS-sum)}
\end{array}$$

1.3 A new "eigenspace" SDP relaxation

We now propose a new SDP relaxation based on the spectral decomposition of the QAP data matrix B :

$$B = \sum_{i=1}^n \lambda_i q^{(i)} q^{(i)\top},$$

where $\lambda_i \in \mathbb{R}$ represents the i -th largest eigenvalue of B and $q^{(i)} \in \mathbb{R}^n$ the corresponding, normalized eigenvector. Note that

$$\begin{aligned}
XBX^\top &= \sum_{i=1}^n \lambda_i Xq^{(i)} q^{(i)\top} X^\top \\
&= \sum_{i=1}^m \lambda_i \left(Xq^{(i)} \right) \left(Xq^{(i)} \right)^\top.
\end{aligned}$$

This formulation suggests the idea of introducing new matrix variables: $Q^{(i)}$ that correspond to $(Xq^{(i)}) (Xq^{(i)})^\top$. To construct the new relaxation we add the (tractable) constraint

$$Q^{(i)} \succeq \left(Xq^{(i)} \right) \left(Xq^{(i)} \right)^\top$$

in its equivalent formulation

$$\begin{bmatrix} Q^{(i)} & Xq^{(i)} \\ (Xq^{(i)})^\top & 1 \end{bmatrix} \succeq 0. \quad (6)$$

Introducing a symmetric matrix variable Y that corresponds to XBX^\top as before, we have

$$Y = \sum_{i=1}^n \lambda_i Q^{(i)}. \quad (7)$$

Since we consider orthonormal eigenvectors, we further have

$$\sum_{i=1}^n Q^{(i)} = I_n. \quad (8)$$

Furthermore, it follows that

$$\begin{aligned} Q^{(i)} e &= X \left(q^{(i)} q^{(i)\top} \right) X^\top e \\ &= X q^{(i)} \left(q^{(i)\top} e \right) \\ &= \left(q^{(i)\top} e \right) X q^{(i)} \quad i = 1, \dots, n \end{aligned} \quad (9)$$

denotes a valid constraint. Finally, one has that

$$\text{diag} \left(Q^{(i)} \right) = X \text{diag} \left(q^{(i)} q^{(i)\top} \right) \quad i = 1, \dots, n \quad (10)$$

is satisfied. To complete the eigenspace relaxation, we also add the valid linear inequalities (4) and the second order cone constraint (5). Thus we obtain the final eigenspace relaxation:

$$\left. \begin{array}{l} \min_{\substack{Y, Q^{(1)}, \dots, Q^{(m)} \in \mathbb{S}^{n \times n} \\ X \in \mathbb{R}^{n \times n}}} \text{trace} (AY) \\ s.t. \quad \left[\begin{array}{cc} Q^{(i)} & Xq^{(i)} \\ (Xq^{(i)})^\top & 1 \end{array} \right] \succeq 0 \quad i = 1, \dots, n, \\ \sum_{i=1}^m Q^{(i)} = I, \\ \sum_{i=1}^m \lambda_i Q^{(i)} = Y, \\ \text{diag} \left(Q^{(i)} \right) = X \text{diag} \left(q^{(i)} q^{(i)\top} \right) \quad i = 1, \dots, n, \\ Q^{(i)} e = \left(q^{(i)\top} e \right) X q^{(i)} \quad i = 1, \dots, n, \\ \mathcal{L}_2 (Y) \leq X \mathcal{L}_2 (B), \\ \left(X \min B_{off}^+ \right)_i \leq Y_{ij} \leq \left(X \max B_{off}^- \right)_i \quad i \neq j, \\ X e = X^\top e = e; \quad X \geq 0. \end{array} \right\} \text{(Eigenspace relaxation)}$$

2 Relations between the SDP bounds

In this section we show that the QAP-R₃ bound is at least as tight the eigenspace relaxation bound and the SDRMS-sum bounds respectively. This is not surprising in view of the much larger size of the QAP-R₃ program, but the proof is still insightful.

We will see in the next section via numerical examples that there is no ordering between the SDRMS-sum and eigenspace relaxation bounds.

Theorem 2.1. *The (QAP-R₃) bound dominates the eigenspace relaxation bound.*

Proof. Assume that Υ is a given feasible solution of QAP-R₃ and that it therefore has the block structure (1).

We will construct a feasible solution of the eigenspace relaxation with the same objective, namely $\text{tr} ([B \otimes A] \Upsilon)$.

Note that, using $B = \sum_{k=1}^n \lambda_k q^{(k)} q^{(k)\top}$, we may rewrite the objective value at Υ as

$$\begin{aligned}
\text{trace}([B \otimes A] \Upsilon) &= \text{tr} \left(A \sum_{i,j=1}^n B_{ij} \Upsilon^{(i,j)} \right) \\
&= \text{trace} \left(A \sum_{i,j=1}^n \left(\sum_{k=1}^n \lambda_k q_i^{(k)} q_j^{(k)\top} \right) \Upsilon^{(i,j)} \right) \\
&= \text{trace} \left(A \sum_{k=1}^n \lambda_k \left(\sum_{i,j=1}^n q_i^{(k)} q_j^{(k)\top} \Upsilon^{(i,j)} \right) \right) \\
&=: \text{trace} \left(A \sum_{k=1}^n \lambda_k Q^{(k)} \right),
\end{aligned}$$

where we have defined:

$$Q^{(k)} := \sum_{i,j=1}^n q_i^{(k)} q_j^{(k)\top} \Upsilon^{(i,j)} \quad (k = 1, \dots, n). \quad (11)$$

If one also defines a matrix $X \in \mathbb{R}^{n \times n}$ with i th column given by the diagonal of $\Upsilon^{(ii)}$ ($i = 1, \dots, n$), then one may verify that this gives a feasible solution of the eigenspace relaxation. Indeed, note that X is nonnegative and satisfies $Xe = X^\top e = e$, by Theorem 1.1(ii).

Moreover, since we have $\Upsilon - \text{diag}(\Upsilon) \text{diag}(\Upsilon)^\top \succeq 0$ by Theorem 1.1(i), we may take the Hadamard (componentwise) product of the left hand side with the positive semidefinite matrix $q^{(k)} q^{(k)\top} \otimes J$ to obtain the positive semidefinite matrix:

$$\left(q^{(k)} q^{(k)\top} \otimes J \right) \circ \Upsilon - \left(q^{(k)} q^{(k)\top} \otimes J \right) \circ \left(\text{diag}(\Upsilon) \text{diag}(\Upsilon)^\top \right) \succeq 0. \quad (12)$$

The first left hand side term has the same block structure as Υ , and block (i, j) is given by $q_i^{(k)} q_j^{(k)} \Upsilon^{(i,j)}$. Thus the sum of all the blocks is precisely $Q^{(k)}$, by (11).

In other words, one has

$$(e \otimes I)^\top \left[\left(q^{(k)} q^{(k)\top} \otimes J \right) \circ \Upsilon \right] (e \otimes I) = Q^{(k)}. \quad (13)$$

Similarly, by construction:

$$(e \otimes I)^\top \left[\left(q^{(k)} q^{(k)\top} \otimes J \right) \circ \left(\text{diag}(\Upsilon) \text{diag}(\Upsilon)^\top \right) \right] (e \otimes I) = X q^{(k)} \left(X q^{(k)} \right)^\top. \quad (14)$$

Thus (12), (13), and (14) imply $Q^{(k)} - X q^{(k)} \left(X q^{(k)} \right)^\top \succeq 0$, which, by the Schur complement theorem, is equivalent to

$$\begin{bmatrix} Q^{(k)} & X q^{(k)} \\ \left(X q^{(k)} \right)^\top & 1 \end{bmatrix} \succeq 0,$$

as required.

Next we wish to show that $\sum_{k=1}^n Q^{(k)} = I$. Letting δ_{ij} denote the Kronecker delta, we have

$$\begin{aligned}
\sum_{k=1}^n Q^{(k)} &= \sum_{k=1}^n \sum_{i,j=1}^n q_i^{(k)} q_j^{(k)\top} \Upsilon^{(i,j)} \\
&= \sum_{i,j=1}^n \left(\sum_{k=1}^n q_i^{(k)} q_j^{(k)\top} \right) \Upsilon^{(i,j)} \\
&= \sum_{i,j=1}^n \delta_{ij} \Upsilon^{(i,j)} \quad \left(\text{since } \sum_{k=1}^n q^{(k)} q^{(k)\top} = I \right) \\
&= \sum_{i=1}^n \Upsilon^{(i,i)} = I \quad \left(\text{by Theorem 1.1(i)}. \right)
\end{aligned}$$

Also,

$$\begin{aligned}
\text{diag}(Q^{(k)}) &= \sum_{i,j=1}^n q_i^{(k)} q_j^{(k)\top} \text{diag}(\Upsilon^{(i,j)}) \\
&= \sum_{i=1}^n q_i^{(k)} q_i^{(k)\top} \text{diag}(\Upsilon^{(i,i)}) \quad \left(\text{since } \text{diag}(\Upsilon^{(i,j)}) = 0 \text{ if } i \neq j \right) \\
&= X \text{diag} \left(q^{(k)} q^{(k)\top} \right).
\end{aligned}$$

In a similar way one may show that $Q^{(k)} e = (q^{(k)\top} e) X q^{(k)}$.

Setting $Y = \sum_{i,j=1}^n B_{ij} \Upsilon^{(i,j)}$ we still need to verify that $\mathcal{L}_2(Y) \leq X \mathcal{L}_2(B)$.

Letting $e_k \in \mathbb{R}^n$ denote the k th standard unit vector, we have

$$\begin{aligned}
\mathcal{L}_2^2(Y)_k &\equiv \|e_k^\top Y\|^2 = \left(\sum_{i,j=1}^n B_{ij} e_k^\top \Upsilon^{(i,j)} \right) \left(\sum_{r,s=1}^n B_{rs} \Upsilon^{(s,r)} e_k \right) \\
&= \sum_{i,j,r,s=1}^n B_{ij} B_{rs} e_k^\top \Upsilon^{(i,j)} \Upsilon^{(s,r)} e_k \\
&\leq \sum_{j,s=1}^n \mathcal{L}_2(B)_j \mathcal{L}_2(B)_s e_k^\top \left(\sum_{i=1}^n \Upsilon^{(i,j)} \sum_{r=1}^n \Upsilon^{(s,r)} \right) e_k \\
&= \sum_{j,s=1}^n \mathcal{L}_2(B)_j \mathcal{L}_2(B)_s \Upsilon_{kk}^{(j,j)} \Upsilon_{kk}^{(s,s)} \quad \left(\text{by Theorem 1.1(iv)} \right) \\
&= (X \mathcal{L}_2(B))_k^2,
\end{aligned}$$

as required. Finally, one may verify in a similar way that the constraints

$$\left(X \min B_{off}^+ \right)_i \leq Y_{ij} \leq \left(X \max B_{off}^- \right)_i \quad i \neq j$$

are satisfied. \square

We will see from numerical examples in the next section that the (QAP-R₃) bound can be (and usually is) strictly better than the eigenspace relaxation bound.

Similarly, one may show that the (QAP-R₃) bound is at least as tight as the (SDRMS-sum) bound. Since the proof is similar to that of the previous theorem, we only sketch the proof.

Theorem 2.2. *The (QAP-R₃) bound dominates the (SDRMS-sum) bound, and the two bounds do not coincide in general.*

Proof. As before, let Υ be a feasible solution of QAP-R₃ that has the block structure (1). If one sets:

$$Y = \sum_{i,j=1}^n B_{ij} \Upsilon^{(i,j)}, Y_2 = \sum_{i,j=1}^n (B_2)_{ij} \Upsilon^{(i,j)}, Y_s = \sum_{i,j=1}^n (B_s)_{ij} \Upsilon^{(i,j)},$$

and, also as before, define the matrix $X \in \mathbb{R}^{n \times n}$ with i th column given by $\text{diag}(\Upsilon^{(ii)})$ ($i = 1, \dots, n$), then one may verify that this gives a feasible solution of (SDRMS-sum). The rest of the proof is completely analogous to that of Theorem 2.1, and is therefore omitted. \square

3 Numerical results

In this section we present numerical results for the relaxations introduced in the first section as well as for some other lower bounds from the literature. We tested all relaxations on the same machine (six core 3.33 GHz processor). Furthermore, we ran the relaxations under Matlab 2011b and used CPLEX¹ for the linear programming problems and SeDuMi [24] or SDPT3 [25] for solving SDP problems.

For the SDRMS-sum and eigenspace relaxations we ran both orderings of the input parameters A and B , and took as lower bound the best bound; we added up both run-times to obtain the total computational time. To express the quality of lower bounds, we defined the *gap* as the following ratio between optimal solution opt (or best known feasible solution) and the computed lower bound lb via:

$$gap = 1 - \frac{\text{opt}}{\text{lb}}.$$

Furthermore, in Table 1 we only ran instances for which computing the bounds took less than 10,000 seconds. If the computation took longer than that, an empty space appears in the table.

In addition to the SDP bounds, we also include data in Table 1 for the level 1 and 2 *reformulation-linearization technique* (RLT) bounds for QAP. These are LP bounds studied in [2] (RLT1 bounds), and [1] (RLT2 bounds).

Table 1: Comparison of various different QAP-relaxations

	RLT1		RLT2		QAP-R3		(SDRMS)-SUM		eigenspace rel.	
	gap	time	gap	time	gap	time	gap	time	gap	time
Nug3	0%	0.67	0%	1.19	0%	0.41	0%	5.18	0%	2.56
Nug5	0%	3.51	0%	52.50	0%	0.49	0%	7.36	0%	4.18
Nug6	0%	7.64	0%	686.34	0%	1.61	1.86%	8.75	2.46%	5.23
Nug7	0%	13.76	0%	5506.65	0%	13.98	4.82%	10.72	5.10%	3.55
Nug8	4.91%	27.46	0%	23517.13	0.23%	73.64	10.35%	11.79	9.07%	3.93
Nug12	9.53%	558.09			1.74%	11808.54	9.51%	12.98	8.73%	13.45
Nug14	8.99%	1141.69			0.39%	79819.95	6.83%	16.36	6.45%	30.45
Nug15							7.33%	19.06	7.20%	37.72

From Table 1 we see that the respective computational times for the RLT1 and RLT2 LP relaxations and the QAP-R₃ SDP relaxation are already prohibitive for instances of dimension less than 15, due to the rapidly growing sizes of the underlying optimization problems.

¹<http://www-03.ibm.com/software/products/us/en/ibmilogcpleoptistud/>

The sum-matrix splitting and eigenspace SDP relaxations require less computational time but their respective bounds are weaker than the QAP-R₃ bounds. This observation is illustrated in Figures 1 and 2, where we only plot the results for the smaller *Nugent* TSPLIB instances.

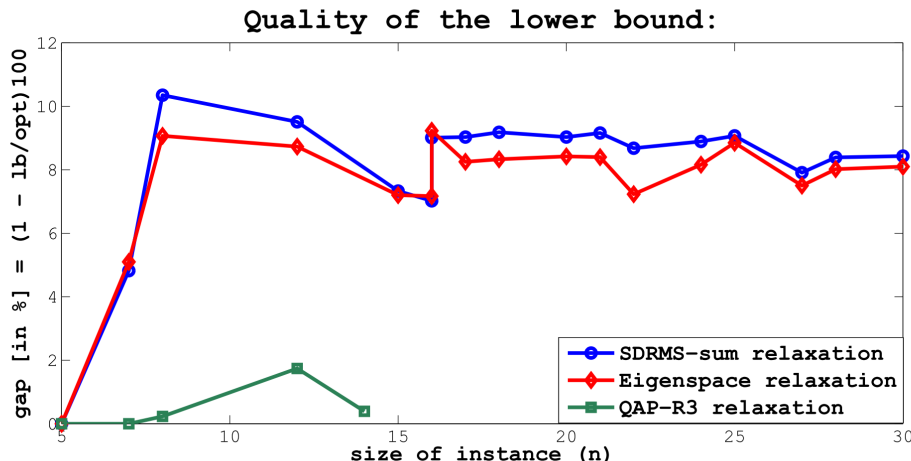


Figure 1: Comparison of three SDP bounds on Nugent instances: the relaxation gap

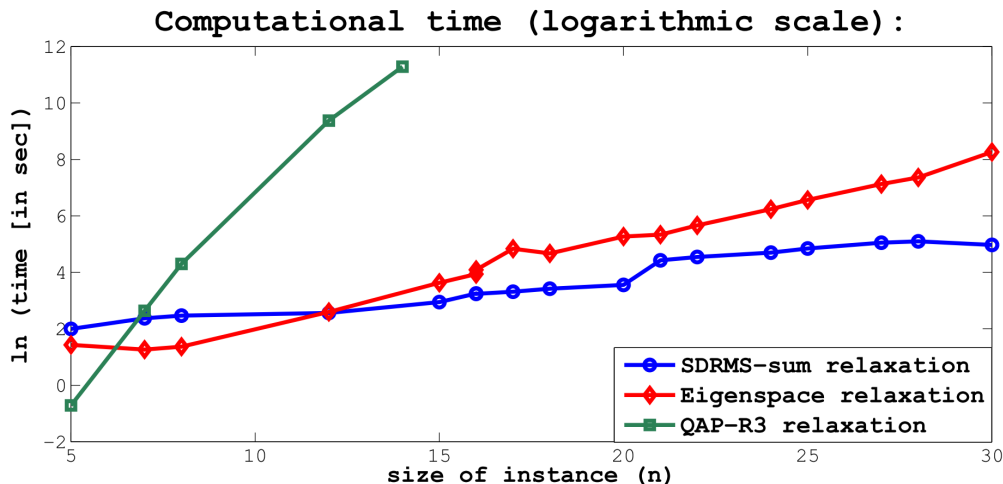


Figure 2: Comparison of three SDP bounds on Nugent instances: the CPU times

In what follows we concentrate on bounds that may be readily computed for a range of values $n > 15$. In addition to the SDRMS-sum and eigenspace bounds, we consider the well-known Gilmore-Lawler bound (GLB) [14, 18] as well as the QPB0 quadratic programming bound introduced by Anstreicher and Brixius [5]. Both the GLB and QPB0 bounds may be computed very cheaply; we include these bounds only as reference point for the quality of bounds that may be obtained at very low computational costs, and do not include the actual computational times for these bounds, since these are negligible compared to those for the SDP bounds. The results are shown in Tables 3, 4, and 5 for QAPLIB instances.

Table 3: A comparison of lower bounds (part 1)

	(SDRMS-SUM)		eigenspace rel		GLB	QPB0
	gap in %	time	gap in %	time	gap in %	gap in %
Nug5	0%	7.36	0%	4.18	0%	8,86%
Nug7	4.82%	10.72	5.10%	3.55	7.43%	16,71%
Nug8	10.35%	11.79	9.07%	3.93	13.08%	24,11%
Nug12	9.51%	12.98	8.73%	13.45	14.71%	18,65%
Nug15	7.33%	19.06	7.20%	37.72	16.26%	14,73%
Nug16a	7.02%	25.82	7.17%	51.35	18.39%	11,34%
Nug16b	9.01%	25.49	9.23%	59.77	17.58%	15,66%
Nug17	9.03%	27.53	8.25%	125.98	19.86%	12,87%
Nug18	9.18%	30.63	8.33%	106.95	19.48%	12,77%
Nug20	9.03%	34.97	8.42%	194.20	19.96%	13,81%
Nug21	9.16%	83.48	8.40%	207.60	24.82%	16,89%
Nug22	8.68%	94.13	7.23%	288.45	30.95%	15,17%
Nug24	8.89%	110.04	8.16%	511.26	23.28%	14,24%
Nug25	9.07%	127.04	8.85%	711.15	23.37%	13,39%
Nug27	7.91%	156.39	7.50%	1247.24	29.29%	12,63%
Nug28	8.39%	164.00	8.02%	1577.58	26.71%	12,81%
Nug30	8.43%	144.34	8.10%	3876.26	25.88%	13,32%
Chr12a	11.02%	42.89	10.17%	15.53	24.15%	305,91%
Chr12b	24.65%	43.23	18.09%	14.96	26.65%	245,85%
Chr12c	11.87%	42.87	11.27%	14.87	28.50%	306,53%
Chr15a	24.80%	54.82	20.58%	37.08	43.16%	596,43%
Chr15b	35.34%	57.05	32.79%	42.08	41.76%	737,60%
Chr15c	7.32%	58.07	5.49%	37.87	35.13%	604,75%
Chr18a	18.21%	69.99	15.53%	99.72	38.92%	691,18%
Chr18b	0.00%	68.65	0.00%	77.99	0.00%	148,61%
Chr20a	1.64%	87.62	1.64%	187.31	1.92%	464,85%
Chr20b	2.68%	88.31	2.49%	208.02	4.44%	440,89%
Chr20c	37.59%	38.38	22.25%	229.97	39.18%	674,26%
Chr22a	3.12%	103.10	2.70%	433.54	3.77%	428,03%
Chr22b	2.89%	105.87	2.36%	357.65	4.17%	442,04%
Chr25a	14.54%	71.52	14.30%	1153.75	27.16%	667,48%
Els19	15.85%	83.49	11.47%	263.23	30.45%	356,44%
Esc16a	46.61%	81.69	17.13%	52.92	44.12%	28,32%
Esc16b	6.85%	62.77	2.74%	56.07	24.66%	17,74%
Esc16c	45.26%	65.74	16.10%	59.64	48.13%	43,78%
Esc16d	96.14%	63.36	77.91%	65.29	81.25%	255,63%
Esc16e	83.57%	64.75	37.11%	47.35	57.14%	90,29%
Esc16f	0 %	56.76	0 %	40.78	0.00%	0.00%

It is clear from the tables that the SDP bounds compare favourable to the GLB and QPB0 bounds in terms of the gap. On the other hand, the SDP bounds are significantly more expensive to compute.

Moreover, the eigenspace relaxation provides stronger lower bounds than SDRMS-sum for most of the QAPLIB instances, but also requires more computational time. This is illustrated by the scatter-plot in Figure 3, where the gaps and computational times are plotted of ten QAPLIB instances with $n = 16$.

The overall picture is therefore what one would expect: the quality of the lower bound improves as the

Table 4: A comparison of lower bounds (part 2)

	(SDRMS)-SUM		eigenspace rel		GLB	QPB0
	gap in %	time	gap in %	time	gap in %	gap in %
Esc16g	81.09%	62.05	30.56%	49.49	53.85%	78,38%
Esc16h	35.87%	23.14	6.98%	47.12	37.25%	29,57%
Esc16i	100.00%	63.70	100.00%	42.32	100.00%	326,71%
Esc16j	100.00%	62.72	89.21%	49.36	87.50%	218,31%
Esc32a	96.92%	164.50	56.81%	6204.21	73.08%	225,16%
Esc32b	39.95%	149.99	38.96%	4921.40	42.86%	65,48%
Esc32c	53.25%	156.85	10.03%	7272.47	45.48%	29,44%
Esc32d	60.70%	169.39	24.02%	4836.53	47.00%	54,62%
Esc32e	100.00%	134.48	100.00%	3495.79	100.00%	8493,30%
Esc32f	100.00%	136.09	100.00%	3504.75	100.00%	8493,50%
Esc32g	100.00%	136.55	100.00%	3601.56	100.00%	1406,19%
Esc32h	48.11%	158.01	13.04%	7294.27	41.32%	46,38%
Had12	1.64%	47.77	2.23%	15.46	7.02%	4,23%
Had14	1.71%	55.81	1.94%	28.03	8.52%	3,94%
Had16	2.47%	69.46	2.45%	54.52	9.73%	3,75%
Had18	2.44%	78.20	2.60%	98.49	10.86%	4,35%
Had20	2.08%	91.80	2.42%	171.16	10.92%	3,88%
Kra30a	16.78%	156.10	12.09%	4833.83	23.10%	23,63%
Kra30b	17.68%	162.34	13.29%	4772.80	24.45%	25,55%
Kra32	18.49%	318.78	13.09%	9399.28	24.02%	29,57%
Lipa20a	0.29%	86.93	0.29%	224.30	0.43%	14,34%
Lipa20b	0.04%	88.75	0.00%	202.42	0.00%	10,86%
Lipa30a	0.18%	175.05	0.18%	3664.58	0.24%	10,61%
Lipa30b	0.06%	226.54	0%	4784.35	0%	4,65%
Rou12	13.40%	49.12	9.42%	33.63	14.12%	15,23%
Rou15	14.77%	64.60	11.31%	61.22	15.71%	16,37%
Rou20	16.87%	41.33	14.17%	198.78	17.31%	17,85%
Scr12	7.25%	48.34	7.72%	33.23	11.31%	74,46%
Scr15	10.02%	62.27	9.31%	64.41	12.52%	76,71%
Scr20	16.00%	41.83	15.11%	222.24	21.14%	80,21%
Tail10a	13.17%	40.25	7.86%	23.81	17.92%	16,96%
Tail10b	11.43%	40.31	9.32%	23.79	51.23%	196,26%
Tail12a	11.08%	46.34	7.62%	30.81	12.70%	14,57%
Tail12b	8.61%	45.11	18.09%	33.93	75.20%	183,05%
Tail15a	15.19%	58.65	12.51%	59.15	15.64%	16,35%
Tail15b	0.46%	60.21	0.69%	109.53	78.28%	574,82%
Tail17a	16.36%	72.48	12.94%	88.64	16.08%	16,93%

computational demand increases.

4 Symmetry reduction

Surprisingly often, QAP instances have data matrices with large automorphism groups; for example, in [10] the symmetry of the QAPLIB instances is described in detail. As shown in [10], one may exploit such symmetry to reduce the size of the QAP- R_3 relaxation. In this section, we will show a similar result

Table 5: A comparison of lower bounds (part 3)

	(SDRMS)-SUM		eigenspace rel		GLB	QPB0
	gap in %	time	gap in %	time	gap in %	gap in %
Tail20a	17.80%	91.71	14.63%	202.76	17.46%	18,22%
Tail20b	6.39%	93.78	38.71%	302.57	88.40%	241,05%
Tail25a	17.92%	137.95	15.39%	888.91	17.55%	18,07%
Tail25b	15.15%	109.89	19.00%	2444.89	86.15%	143,98%
Tail30a	17.20%	175.18	15.09%	4329.12	17.24%	17,53%
Tail30b	12.72%	191.91	12.33%	9134.32	93.54%	812,46%
Tail35a	19.60%	370.74	17.76%	16056.85	19.44%	19,85%
Tail35b	13.79%	448.31	18.88%	25349.79	89.11%	99,96%
Tho30	12.24%	165.94	11.59%	5797.95	39.59%	21,06%

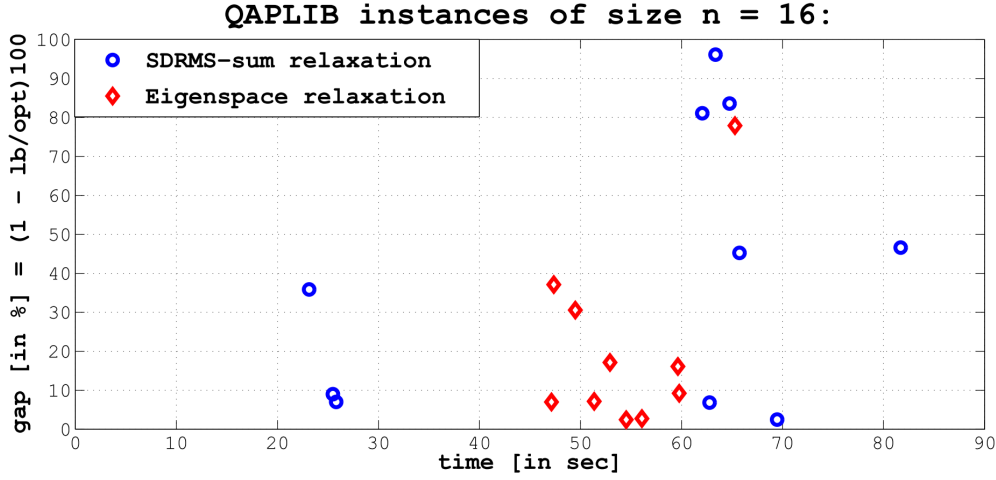


Figure 3: Comparison of the SDRMS-sum and eigenspace SDP bounds on QAPLIB instances of size $n = 16$

for the new eigenspace relaxation. We will first give some background on symmetry reduction, and then explain how this applies to the eigenspace relaxation. Finally, by way of example, we will show how the eigenspace relaxation of the traveling salesman problem, when formulated as a QAP, reduces to a linear program with one additional second-order cone constraint.

4.1 Background on symmetry

Let \mathcal{S}_n denote the symmetric group on $\{1, \dots, n\}$. With each permutation $\pi \in \mathcal{S}_n$, we associate an $n \times n$ permutation matrix $P_\pi \in \Pi_n$, defined by

$$(P_\pi)_{ij} = \begin{cases} 1 & \text{if } \pi(j) = i \\ 0 & \text{else} \end{cases} \quad (i, j = 1, \dots, n)$$

Thus, for any $X \in \mathbb{R}^{n \times n}$ one has

$$(P_\pi^\top X P_\pi)_{ij} = X_{\pi(i), \pi(j)} \quad (i, j = 1, \dots, n).$$

The automorphism group $\text{aut}(X)$ of a matrix X is the group of permutations π that satisfies

$$X_{ij} = X_{\pi(i), \pi(j)} \quad (i, j = 1, \dots, n) \iff X = P_\pi^\top X P_\pi.$$

The centralizer ring (or commutant) of a permutation group $\mathcal{G} \subset \mathcal{S}_n$ is the set of all matrices such with automorphism group \mathcal{G} :

$$\mathcal{A}_{\mathcal{G}} := \{X \in \mathbb{R}^{n \times n} \mid P_\pi^\top X P_\pi = X \ \forall \pi \in \mathcal{G}\}.$$

The centralizer ring $\mathcal{A}_{\mathcal{G}}$ is a matrix $*$ -algebra, i.e. a linear subspace of $\mathbb{R}^{n \times n}$ that is also closed under matrix multiplication and under taking the transpose.

The orthogonal projection of a matrix $X \in \mathbb{R}^{n \times n}$ onto $\mathcal{A}_{\mathcal{G}}$ is given by

$$\mathcal{R}_{\mathcal{A}_{\mathcal{G}}}(X) = \frac{1}{|\mathcal{G}|} \sum_{\pi \in \mathcal{G}} P_\pi^\top X P_\pi.$$

The projection operator is known as the Reynolds operator of \mathcal{G} .

4.2 Symmetry reduction for the eigenspace relaxation

We are interested in the situation where QAP data matrix A has a non-trivial automorphism group, and will show how the size of the eigenspace relaxation may typically be reduced in this situation.

Lemma 4.1. *Assume $X, Q^{(1)}, \dots, Q^{(n)}$ is optimal for the eigenspace relaxation, and let $\pi \in \text{aut}(A)$.*

Then the following is also optimal:

$$\bar{Q}^{(i)} = P_\pi^\top Q^{(i)} P_\pi \quad (i = 1, \dots, n), \quad \bar{X} = P_\pi X.$$

The proof is by direct verification, and is therefore omitted.

Since the optimal set of the eigenspace relaxation is convex, we immediately have the following corollary.

Corollary 4.2. *Assume $X, Q^{(1)}, \dots, Q^{(n)}$ is optimal for the eigenspace relaxation. Then the following is also optimal:*

$$\hat{Q}^{(i)} = \frac{1}{|\text{aut}(A)|} \sum_{\pi \in \text{aut}(A)} P_\pi^\top Q^{(i)} P_\pi \quad (i = 1, \dots, n), \quad \hat{X} = \frac{1}{|\text{aut}(A)|} \sum_{\pi \in \text{aut}(A)} P_\pi X.$$

As a consequence, we may assume that $Q^{(i)} \in \mathcal{A}_{\text{aut}(A)}$ ($i = 1, \dots, n$). Moreover, we may assume that, for each column of X , the entries corresponding to a given orbit of $\text{aut}(A)$ are equal. If $\mathcal{A}_{\text{aut}(A)}$ is low-dimensional as a vector space, then this gives a significant reduction in the number of variables. Moreover, since $\mathcal{A}_{\text{aut}(A)}$ is a matrix $*$ -algebra, it may be block-diagonalized by a suitable unitary transformation. Thus the linear matrix inequalities may be brought into block-diagonal form, and this may in turn be exploited by interior point algorithms. For more details on symmetry reduction in SDP, see e.g. [6]. We will limit our discussion here to an example of how one may exploit symmetry in the eigenspace relaxation of the QAP formulation of the traveling salesman problem.

Before doing so, it is important to note that the SDRMS-sum relaxation is much less amenable to symmetry reduction: the introduction of the vectors u and d there typically destroys the symmetry.

4.3 Example of symmetry reduction: the traveling salesman problem

The traveling salesman problem (TSP) may be stated as finding a Hamiltonian cycle of minimum length in a complete graph on n vertices with edge lengths given by the entries of a matrix $B \in \mathbb{S}^{n \times n}$.

It is well-known that the TSP may be written as the QAP:

$$\min_{X \in \Pi_n} \frac{1}{2} \text{trace}(AXBX^T),$$

by choosing the matrix A as the adjacency matrix of the standard Hamiltonian cycle:

$$A := \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 & 1 \\ 1 & 0 & 1 & 0 & \cdots & 0 \\ 0 & 1 & 0 & 1 & \ddots & \vdots \\ \vdots & & & \ddots & \ddots & \\ 0 & & & & 0 & 1 \\ 1 & 0 & \cdots & 0 & 1 & 0 \end{bmatrix}.$$

As a consequence, $\text{aut}(A)$ is the dihedral group D_n , and its commutant is the algebra of symmetric $n \times n$ circulant matrices.

As a result we may perform symmetry reduction on the eigenspace relaxation of the QAP formulation of TSP to obtain the following, equivalent, but reduced reformulation.

Theorem 4.3. *The bound of the eigenspace relaxation for TSP is given by (for odd n and with $B = \sum_{i=1}^n \lambda_i q^{(i)} q^{(i)\top}$):*

$$\min 2n \sum_{i=1}^n \lambda_i \gamma_1^{(i)}$$

subject to

$$\begin{aligned} 2 \sum_{j=1}^d \left(\sum_{i=1}^n \lambda_i \gamma_j^{(i)} \right)^2 &\leq \left(\frac{1}{n} e^\top \mathcal{L}_2(B) \right)^2 \\ \sum_{j=1}^d \cos\left(\frac{2\pi m j}{n}\right) \gamma_j^{(i)} &\geq \frac{-1}{2n} && i = 1, \dots, n, \quad m = 1, \dots, d \\ \sum_{i=1}^n \gamma_j^{(i)} &= 0 && j = 1, \dots, d \\ \sum_{j=1}^d \gamma_j^{(i)} &= \frac{1}{2n} ((e^\top q^{(i)})^2 - 1) && i = 1, \dots, n, \end{aligned}$$

where $d = \lfloor n/2 \rfloor$.

Proof. The proof is similar to the symmetry reduction for the (QAP-R₃) relaxation that is described in detail in [12]; we therefore only sketch the proof here. Since the dihedral group only has one orbit, we may assume that all columns of X have the same entries, by Corollary 4.2. This means that we may assume $X = \frac{1}{n} J$.

Moreover, also by Corollary 4.2, we have that the $Q^{(i)}$ s may be assumed to be symmetric circulant matrices. A symmetric circulant matrix is completely determined by the first $d+1$ entries of its first row, and its eigenvalues are linear functions of these entries. In particular, for a given $i \in \{1, \dots, n\}$, we may assume that $Q^{(i)}$ is determined by the first d entries in its first row, say $\gamma_j^{(i)}$ ($j = 0, \dots, d$), so that the $d+1$ distinct eigenvalues of $Q^{(i)}$ are:

$$\gamma_0^{(i)} + \sum_{j=1}^d 2\gamma_j^{(i)} \cos\left(\frac{2\pi m j}{n}\right), \quad m = 0, \dots, d.$$

(These and other properties of circulant matrices are reviewed in [15].)

If we consider the eigenspace relaxation constraint $Q^{(i)} - Xq^{(i)}(Xq^{(i)})^\top \succeq 0$, and substitute $X = \frac{1}{n}J$, we obtain

$$Q^{(i)} - \frac{1}{n^2} \left(e^\top q^{(i)} \right)^2 J \succeq 0. \quad (15)$$

The eigenspace relaxation constraint $\text{diag}(Q^{(i)}) = X \text{diag}(q^{(i)}q^{(i)\top})$ implies $\gamma_0^{(i)} = \frac{1}{n} \|q^{(i)}\|^2 = \frac{1}{n}$. Also, the constraint $Q^{(i)}e = (q^{(i)\top}e)Xq^{(i)}$ implies $\gamma_0^{(i)} + 2\sum_{j=1}^d 2\gamma_j^{(i)} = \frac{1}{n} (e^\top q^{(i)})^2$. This is precisely the eigenvalue of $Q^{(i)}$ that corresponds to the all-ones eigenvector e . Thus the linear matrix inequality in (15) reduces to the set of eigenvalue inequalities:

$$\frac{1}{n} + \sum_{j=1}^d 2\gamma_j^{(i)} \cos\left(\frac{2\pi mj}{n}\right) \geq 0, \quad m = 1, \dots, d,$$

as required. (The case $m = 0$ could be omitted since it corresponds to the eigenvector e .) Proceeding in this way the complete problem formulation in the statement of the theorem may be obtained. \square

Note that the eigenspace relaxation reduces to a linear program with one additional second-order cone constraint. The number of variables (the $\gamma_j^{(i)}$ s) is of the order n^2 and the number of linear constraints is of the same order.

In Table 6 we compare the eigenspace relaxation bound to the QAP-R₃ bound and the Held-Karp bound [16] for some small instances from the library TSPLIB.

Problem	Eigenspace	QAP-R3	Held-Karp
gr17	39.7%	3.7%	0%
gr21	26.7%	0%	0%
gr24	27.4%	0%	0%
fri26	27.6%	0.5%	0%
bays29	28.1%	1.0%	0.3%
Dantzig42	35.1%	n/a	0.3%
gr48	27.5%	n/a	1.7%
HK48	23.6%	n/a	0.3%
Eil51	54.2%	n/a	28.2%
Berlin52	29.8%	n/a	0%
Eil76	20.1%	n/a	0.2%

Table 6: Gaps on some small TSPLIB instances from various convex relaxations.

As one may expect, the eigenspace relaxation bound is significantly weaker than the QAP-R₃ and Held-Karp bound respectively, but it is also much cheaper to compute. In particular, it may be solved in a fraction of a second for the instances in Table 6.

Moreover one may potentially add cutting planes to strengthen the relaxation. To this end, it is helpful to note the relation between the variables in the reduced eigenspace and QAP-R₃ relaxations. It is shown in [12] that the QAP-R₃ relaxation for TSP reduces to:

$$\min \frac{1}{2} \text{trace} \left(BX^{(1)} \right)$$

subject to

$$\begin{aligned} X^{(k)} &\geq 0, & k = 1, \dots, d \\ I + \sum_{k=1}^d X^{(k)} &= J, \\ I + \sum_{k=1}^d \cos\left(\frac{2\pi ik}{n}\right) X^{(k)} &\succeq 0, & i = 1, \dots, d \\ X^{(k)} &\in \mathbb{S}^{n \times n}, & k = 1, \dots, d, \end{aligned}$$

where $d = \lfloor n/2 \rfloor$.

The variables $X^{(k)}$ ($k = 1, \dots, d$) correspond to the distance matrices of the optimal tour: $X^{(1)}$ corresponds to the adjacency matrix of the optimal tour, $X^{(2)}$ to the adjacency matrix of vertices at distance 2 in the optimal tour, etc.

It is therefore insightful to note that one has the following correspondence between the variables in the eigenspace relaxation, and those of the QAP-R₃ relaxation.

Lemma 4.4. *Let $Q = [q^{(1)}, \dots, q^{(n)}]$. If $X^{(i)}$ ($i = 1, \dots, b$) is feasible for QAP-R₃, then*

$$\gamma_j^{(i)} = \frac{1}{2n} \left(Q^\top X^{(j)} Q \right)_{ii} \quad i = 1, \dots, n, j = 1, \dots, d$$

is feasible for the eigenspace relaxation, and the two objective values are equal.

As a consequence, one may add valid inequalities for $X^{(1)}$, like the subtour elimination inequalities, to the eigenspace relaxation. It remains a topic for future research to see if one may obtain computationally interesting relaxations in this way.

5 Concluding remarks

Our numerical results suggest a size-dependent choice of relaxations for solving the quadratic assignment problem within a branch-and-bound framework:

$$\begin{cases} \text{QAP-R}_3 & \text{if } n < 12 \\ \text{Eigenspace relaxation} & \text{if } 12 \leq n \leq 25 . \\ \text{SDRMS-sum} & \text{if } n > 25 \end{cases}$$

This choice also depends on the group symmetry of the QAP data matrices: if these have large automorphism groups, then one may use the eigenspace relaxation for larger values of n .

The future aim of this work is therefore to complete a branch-and-bound implementation along these lines, and to compare it with other complete solution approaches.

Acknowledgements

The authors would like to thank Kurt Anstreicher and Nathan Brixius for making their code for computing the QPB0 bound available to them, and Kim Chuan Toh for valuable discussions.

References

- [1] W.P. Adams, M. Guignard, P.M. Hahn, and W.L. Hightower. A level-2 reformulation-linearization technique bound for the quadratic assignment problem. *European Journal Operational Research*, **180**, 983–996, 2007.
- [2] W.P. Adams and T.A. Johnson. Improved linear programming-based lower bounds for the quadratic assignment problem. *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, **16**, 43–75, 1994.
- [3] K.M. Anstreicher. Recent advances in the solution of quadratic assignment problems. *Mathematical Programming B*, 97:27 – 42, 2003.
- [4] K.M. Anstreicher, N. Brixius, J.P. Goux, and J. Linderoth. Solving large quadratic assignment problems on computational grids. *Mathematical Programming*, 91:563–588, 2002.

- [5] K.M. Anstreicher and N.W. Brixius. A new bound for the quadratic assignment problem based on convex quadratic programming. *Math. Program. Ser. A*, 89:341 – 357, 2001.
- [6] C. Bachoc, D.C. Gijswijt, A. Schrijver, F. Vallentin. Invariant semidefinite programs. In *Handbook on Semidefinite, Cone and Polynomial Optimization*, M.F. Anjos and J.B. Lasserre (eds.), Springer, New York, 2012, pp. 219–269.
- [7] N.W. Brixius and K.M. Anstreicher. Solving quadratic assignment problems using convex quadratic programming relaxations. *Optimization Methods and Software*, 16:49–68, 2001.
- [8] R. Burkard, M. Dell'Amico, and S. Martello. *Assignment Problems*. SIAM, Philadelphia, 2009.
- [9] R.E. Burkard, S.E. Karisch, and F. Rendl. QAPLIB – a quadratic assignment problem library. *J. of Global Optimization*, 10(4):391–403, June 1997.
- [10] E. de Klerk and R. Sotirov. Exploiting group symmetry in semidefinite programming relaxations of the quadratic assignment problem, *Mathematical Programming Ser. A*, 122(2):225–246, 2010.
- [11] E. de Klerk and R. Sotirov. Improved semidefinite programming bounds for quadratic assignment problems with suitable symmetry. *Mathematical Programming*, 133(1), 75–91, 2012.
- [12] E. de Klerk, D.V. Pasechnik and R. Sotirov. On semidefinite programming relaxations of the traveling salesman problem. *SIAM Journal on Optimization*, 19(4), 1559–1573, 2008.
- [13] M. Fischetti, M. Monaci, and D. Salvagnin. Three ideas for the quadratic assignment problem. *Operations Research*, 60(4):954–964, July 2012.
- [14] P.C. Gilmore. Optimal and suboptimal algorithms for the quadratic assignment problem. *Journal of the Society of Industrial and Applied Mathematics*, 10(2):305–313, 1962.
- [15] R.M. Gray. Toeplitz and Circulant Matrices: A review. *Foundations and Trends in Communications and Information Theory*, 2(3):155–239, 2006.
- [16] M. Held and R.M. Karp. The traveling-salesman problem and minimum spanning trees. *Operations Research*, 18(6):1138–1162, 1970.
- [17] T.C. Koopmans and M.J. Beckmann. Assignment problems and the location of economic activities. *Econometrica*, 25:53–76, 1957.
- [18] E.L. Lawler. The quadratic assignment problem. *Management Science*, 9:586–599, 1963.
- [19] H. Mittelman and J. Peng. Estimating bounds for quadratic assignment problems associated with hamming and manhattan distance matrices based on semidefinite programming. *SIAM Journal on Optimization*, 20:3408–3426, 2010.
- [20] A. Nyberg and T. Westerlund. A new exact discrete linear reformulation of the quadratic assignment problem. *European Journal of Operational Research*, 220(2), 314–319, 2012.
- [21] J. Peng, H. Mittelman, and X. Li. A new relaxation framework for quadratic assignment problems based on matrix splitting. *Mathematical Programming Computation*, 2(1):59–77, 2010.
- [22] J. Peng, T. Zhu, H. Luo, and K.C. Toh. Semi-definite relaxation of quadratic assignment problems nonredundant matrix splitting. *Technical Report*, 2011.
- [23] J. Povh and F. Rendl. Copositive and semidefinite relaxations of the quadratic assignment problem. *Discrete Optim.*, 6(3):231–241, 2009.
- [24] J.F. Sturm, Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones, *Optimization Methods and Software*, 11/12:625–653, 1999.
- [25] K.C. Toh, M.J. Todd, and R.H. Tütüncü. SDPT3 — a Matlab software package for semidefinite programming. *Optimization Methods and Software*, 11:545–581, 1999.
- [26] Q. Zhao, S.E. Karisch, F. Rendl, and H. Wolkowicz. Semidefinite programming relaxations for the quadratic assignment problem. *J. Comb. Optim*, 2:71–109, 1998.