Well-solvable cases of the QAP with block-structured matrices

Eranda Çela[*] Vladimir G. Deineko[†] Gerhard J. Woeginger[‡]

Abstract

We investigate special cases of the quadratic assignment problem (QAP) where one of the two underlying matrices carries a simple block structure. For the special case where the second underlying matrix is a monotone anti-Monge matrix, we derive a polynomial time result for a certain class of cut problems. For the special case where the second underlying matrix is a product matrix, we identify two sets of conditions on the block structure that make this QAP polynomially solvable respectively NP-hard.

Keywords: combinatorial optimization; computational complexity; cut problem; balanced cut; Monge condition; product matrix.

1 Introduction

The *Quadratic Assignment Problem* (QAP) is an important and well-studied problem in combinatorial optimization; we refer the reader to the book [4] by Çela and the recent book [2] by Burkard, Dell'Amico & Martello for comprehensive surveys on this problem. The QAP in Koopmans-Beckmann form [9] takes as input two $n \times n$ square matrices $A = (a_{ij})$ and $B = (b_{ij})$ with real entries. The goal is to find a permutation π that minimizes the objective function

$$Z_\pi(A,B) \;:=\; \sum_{i=1}^{n}\sum_{j=1}^{n} a_{\pi(i)\pi(j)}\, b_{ij}. \tag{1}$$

Here π ranges over the set S_n of all permutations of $\{1, 2, \ldots, n\}$. In general, the QAP is extremely difficult to solve and hard to approximate. One branch of research on the

[*] cela@opt.math.tu-graz.ac.at. Institut für Optimierung und Diskrete Mathematik, TU Graz, Steyrergasse 30, A-8010 Graz, Austria

[†] Vladimir.Deineko@wbs.ac.uk. Warwick Business School, The University of Warwick, Coventry CV4 7AL, United Kingdom

[‡] gwoegi@win.tue.nl. Department of Mathematics and Computer Science, TU Eindhoven, P.O. Box 513, 5600 MB Eindhoven, Netherlands

QAP concentrates on the algorithmic behavior of strongly structured special cases; see for instance Burkard & al [1], Deineko & Woeginger [7], Çela & al [6], or Çela, Deineko & Woeginger [5] for typical results in this direction. We will contribute several new results to this research branch.

Results of this paper. Our first result is motivated by a *balanced* multi-cut problem where a group of entities has to be divided into q clusters of equal size with the objective of minimizing the overall connection cost between different clusters; we refer the reader to Lengauer [10] for a variety of applications of this and other cut problems in VLSI design. Pferschy, Rudolf & Woeginger [12] show that if the connection costs carry a certain anti-Monge structure, the balanced multi-cut problem can be solved efficiently. In the language of the QAP, the connection costs can be summarized in an anti-Monge matrix A and the balanced multi-cut structure can be encoded by a block-structured 0-1 matrix B. The result of [12] then states that the corresponding special case of the QAP is always solved by the identity permutation.

In this paper, we generalize the result of [12] to the case where the entities are to be divided into q clusters of prescribed (but not necessarily equal) sizes. In our generalization the connection costs are given by a *monotone* anti-Monge matrix A, and the cluster structure is specified by a block-structured 0-1 multi-cut matrix B that lists the clusters in order of non-decreasing size. We show that the resulting special case of the QAP (again) is always solved by the identity permutation. Our proof method strongly hinges on the monotonicity of matrix A; in fact it can be seen that without monotonicity the result would break down (see Example 3.6 in Section 3).

Our second result concerns a wide class of specially structured QAPs that are loosely related to the multi-cut QAP in the preceding paragraph. Matrix A is now a product matrix, and hence a special anti-Monge matrix with a particularly nice structure. Matrix B is now a block matrix with some fixed block pattern P, and hence a generalization of the multi-cut matrices in the multi-cut QAP. In comparison to the multi-cut QAP, the structure of matrix A has become more restricted, while the structure of matrix B has become more general. The resulting special case of the QAP is called the *Product-Block QAP* with block pattern P.

On the positive side, we identify conditions on the block pattern P that render the Product-Block QAP polynomially solvable. One main ingredient of the polynomial algorithm is the concavity of certain underlying functions, and the other main ingredient is an extensive enumeration of cases. On the negative side, we identify conditions on the block pattern P that make the Product-Block QAP NP-hard. The positive conditions as well as the negative conditions on the pattern exploit the connections to an underlying continuous quadratic program.

Organization of the paper. Section 2 introduces all the relevant matrix classes, and also states some simple observations on the QAP. Section 3 contains our results on the multi-cut problem on anti-Monge matrices, and Section 4 presents our results on the Product-Block QAP. Section 5 concludes the paper by listing some open problems.

2 Definitions and preliminaries

All matrices in this paper are symmetric and have real entries. In order to avoid trouble with the standard models of computation, we will sometimes assume for our complexity results that the matrix entries are *rational* numbers; this assumption will always be stressed and stated explicitly in the corresponding theorem.

For a $q \times q$ matrix $P = (p_{ij})$, we say that an $n \times n$ matrix $B = (b_{ij})$ is a *block matrix with block pattern P* if the following holds: (i) there exists a partition of the row and column set $\{1, \ldots, n\}$ into q (possibly empty) intervals $I_1, \ldots, I_q$ such that for $1 \le k \le q-1$ all elements of interval I_k are smaller than all elements of interval I_{k+1}; (ii) for all indices i and j with $1 \le i, j \le n$ and $i \in I_k$ and $j \in I_\ell$, we have $b_{ij} = p_{k\ell}$. The sets $I_1, \ldots, I_q$ form the so-called row and column blocks of matrix B.

A *multi-cut matrix B* is a block matrix whose pattern matrix has 0's along the main diagonal and 1's everywhere else. Intuitively speaking, every block I_k in a multi-cut matrix represents a cluster of data points; data points in the same cluster are very similar to each other (and hence at distance 0), whereas data points from different clusters are dissimilar and far away from each other. A multi-cut matrix is in *normal form*, if its block sizes are in non-decreasing order with $|I_1| \le |I_2| \le \cdots \le |I_q|$; note that the rows and columns of every multi-cut matrix can easily be permuted into this normal form.

For a real number $\lambda > 0$, a *1-λ-1 block matrix* is a block matrix with the following block pattern $P(\lambda)$:

$$P(\lambda) \;=\; \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & \lambda \end{pmatrix} \tag{2}$$

Note that for $0 < \mu < \lambda$, any 1-λ-1 block matrix can be written as a non-negative linear combination of two 1-μ-1 block matrices.

An $n \times n$ matrix $A = (a_{ij})$ is *monotone*, if $a_{ij} \le a_{i,j+1}$ and $a_{ij} \le a_{i+1,j}$ holds for all i, j, that is, if the entries in every row and every column are in non-decreasing order. Matrix A is a *sum matrix*, if there are (not necessarily positive) real numbers $\alpha_1, \ldots, \alpha_n$ such that $a_{ij} = \alpha_i + \alpha_j$ for $1 \le i, j \le n$. Matrix A is a *product matrix*, if there are *non-negative* real numbers $\alpha_1, \ldots, \alpha_n$ such that

$$a_{ij} \;=\; \alpha_i\,\alpha_j \qquad \text{for } 1 \le i, j \le n. \tag{3}$$

If $\alpha_1 \le \alpha_2 \le \cdots \le \alpha_n$ holds, then (3) gives a *monotone* product matrix. Matrix A is an *anti-Monge matrix*, if its entries are non-negative and satisfy the anti-Monge inequalities

$$a_{ij} + a_{rs} \;\ge\; a_{is} + a_{rj} \qquad \text{for } 1 \le i < r \le n \text{ and } 1 \le j < s \le n. \tag{4}$$

In other words, in every 2×2 submatrix the sum of the entries on the main diagonal dominates the sum of the entries on the other diagonal. This property essentially dates back to the work of Gaspard Monge [11] in the 18th century. Much research has been done on the effects of Monge structures in combinatorial optimization, and we refer the reader to the survey [3] by Burkard, Klinz & Rudolf for more information on Monge and anti-Monge structures. It can be shown (see Rudolf & Woeginger [14]; Burkard & al [1]) that a symmetric matrix is a *monotone* anti-Monge matrix, if and only if it can be written as a non-negative linear combination of 1-2-1 block matrices; in other words, the 1-2-1 block matrices in (2) form the extremal rays of the cone of monotone anti-Monge matrices. Furthermore every (arbitrary, not necessarily monotone) anti-Monge matrix can be written as a sum of a monotone anti-Monge matrix and an appropriately chosen sum matrix.

A matrix belongs to the λ-*generalized monotone anti-Monge cone*, λ-GMAM cone for short, if it can be written as a non-negative linear combination of 1-λ-1 block matrices. Note that the 2-GMAM cone coincides with the standard monotone anti-Monge cone, and that for $0 < \mu < \lambda$ the μ-GMAM cone properly contains the λ-GMAM cone.

We close this section with two simple but useful results from the QAP folklore.

Fact 2.1 *Consider a QAP where A is a sum matrix and where all row and column sums of B are the same. Then all permutations yield the same objective value.*

Proof. Assume that the entries of matrix A are given by $a_{ij} = \alpha_i + \alpha_j$ for $1 \le i, j \le n$, and let β denote the row and column sum of matrix B. Then for every permutation π, the objective value in (1) equals $Z_\pi(A, B) = 2\beta \sum_{i=1}^{n} \alpha_i$. □

Fact 2.2 *If a permutation π solves the QAP between matrices A' and B and the QAP between matrices A'' and B to optimality, then π also solves the QAP between $A' + A''$ and B to optimality.* □

3 Multi-cut problems on anti-Monge matrices

In this section, we consider the special case of the QAP where matrix A is a monotone anti-Monge matrix and where matrix B is a multi-cut matrix in normal form. We will first resolve a highly restricted special case in Section 3.1, and then deduce a polynomial time result for the general version in Section 3.2.

3.1 A highly restricted special case

In this section we consider the special case of the QAP where the $n \times n$ matrix A is a 1-2-1 block matrix and where the $n \times n$ matrix B is a multi-cut matrix with only two blocks (and hence a standard cut matrix). We denote the sizes of the three blocks of matrix A by r, s, t, and the sizes of the two blocks of matrix B by u and v; see Figure 1 for an illustration. As B is in normal form, we have $u \le v$. Note furthermore that $r + s + t = u + v = n$, and that $s + t \le 2v$.

Consider an arbitrary permutation π for the formulation (1) of the QAP, and let x (respectively, y and z) be the number of rows from A's first block (respectively, its

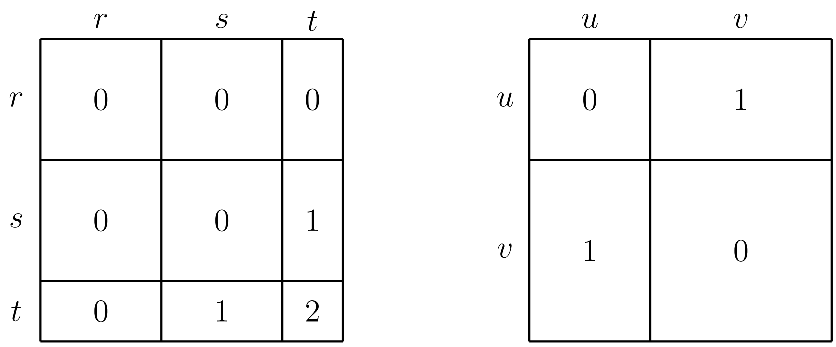

Figure 1: The notation r, s, t, u, v for the 1-2-1 block matrix A and the multi-cut matrix B used in Section 3.1.

second and third block) that π assigns to the first block of matrix B; the remaining $r - x$ (respectively, $s - y$ and $t - z$) rows from these blocks are assigned to the second block of matrix B. The corresponding objective value can then be written as

$$\begin{aligned} f(x, y, z) &= 2y(t - z) + 2z(s - y) + 4z(t - z) \\ &= -4z^2 - 4yz + 2ty + (2s + 4t)z. \end{aligned} \tag{5}$$

The variables x, y, z are integers with $0 \le x \le r$, $0 \le y \le s$, and $0 \le z \le t$, that satisfy $x + y + z = u$. Of course, the optimal objective value of the QAP coincides with the optimal objective value of this non-linear integer program IP.

Since the value of $f(x, y, z)$ in (5) does not depend on x, we will drop variable x from our further considerations. Since $x = u - y - z$ and $u + v = r + s + t$, the constraint $0 \le x \le r$ can be rewritten as $s + t - v \le y + z \le u$. Next we define a continuous programming relaxation CPR of the IP, in which y and z are real variables. Furthermore, we relax the upper bound constraint $y + z \le u$ to the less restrictive constraint $y + z \le v$. Therefore the CPR has the objective of minimizing (5) subject to the constraints

$$0 \le y \le s; \qquad 0 \le z \le t; \qquad s + t - v \le y + z \le v. \tag{6}$$

The following two lemmas derive lower bounds on the optimal objective value of the CPR, and hence also on the optimal objective value of the IP and the QAP.

Lemma 3.1 *If $t \le v \le s + t$, then the objective value of the CPR (and hence also of the QAP) is at least $\gamma_1 := 2t\,(s + t - v)$.*

Proof. We start with two auxiliary inequalities. The condition $t \le v$ yields

$$2ts \;\ge\; 2t\,(s + t - v) \;=\; \gamma_1, \tag{7}$$

and $s + t \le 2v$ yields

$$2(2v - s)\,(s + t - v) \;\ge\; 2t\,(s + t - v) \;=\; \gamma_1. \tag{8}$$

Since the Hessian matrix of function f is indefinite, its minimizers lie on the boundary of the feasible region defined by (6). We distinguish six cases on the six bounding lines.

(Case 1). The minimizer satisfies $y = 0$. The problem turns into the minimization of $g(z) = -4z^2 + (2s+4t)z = 2z(s+2t-2z)$ subject to $0 \le z \le t$ and $s+t-v \le z \le v$, and hence subject to $s+t-v \le z \le t$. As $g(z)$ is concave, it is minimized at the boundary. Inequalities (7) and (8) show that $g(t) = 2st$ and $g(s+t-v) = 2(2v-s)\,(s+t-v)$ are both at least γ_1.

(Case 2). The minimizer satisfies $y = s$. The problem is to minimize $g(z) = 2(t-z)(2z+s)$ subject to $0 \le z \le t$ and $t-v \le z \le v-s$, and hence subject to $0 \le z \le v-s$. Again $g(z)$ is concave, and (7) and (8) show that the values $g(0) = 2st$ and $g(v-s) = 2(s+t-v)(2v-1)$ at the boundary are at least γ_1.

(Case 3). The minimizer satisfies $z = 0$. The problem is to minimize $g(y) = 2ty$ subject to $0 \le y \le s$ and $s+t-v \le y \le v$. Then $y \ge s+t-v$ implies $g(y) \ge 2t\,(s+t-v) = \gamma_1$.

(Case 4). The minimizer satisfies $z = t$. The problem is to minimize $g(y) = 2t(s-y)$ subject to $0 \le y \le s$ and $s-v \le y \le v-t$. Then $y \le v-t$ implies $g(y) \ge 2t\,(s+t-v) = \gamma_1$.

(Case 5). The minimizer satisfies $y+z = s+t-v$. The problem is to minimize the increasing linear function $g(z) = 2(2v-s-t)z + 2(s+t-v)t$ subject to $0 \le z \le t$ and $t-v \le z \le s+t-v$. Then $z \ge 0$ yields $g(z) \ge 2(s+t-v)t = \gamma_1$.

(Case 6). The minimizer satisfies $y+z = v$. The problem is to minimize the decreasing linear function $g(z) = 2vt - 2(2v-s-t)z$ subject to $0 \le z \le t$ and $v-s \le z \le v$. Then $z \le t$ yields $g(z) \ge 2t(s+t-v) = \gamma_1$. □

Lemma 3.2 *If $v < t$, then the objective value of the CPR (and hence also of the QAP) is at least $\gamma_2 := 2v\,(s+2t-2v)$.*

Proof. This proof is analogous to the proof of the preceding lemma, except that we use a different set of bounds and inequalities. From $s+t \le 2v$ we conclude

$$2(s+t-v)(2v-s) \;=\; s\,(2v-s-t) + \gamma_2 \;\ge\; \gamma_2. \tag{9}$$

This time the feasible region is bounded by only four straight lines. The case $z = 0$ is impossible, since then $s+t-v \le y+z = y \le s$ implies the contradiction $t \le v$. Also the case $z = t$ is impossible, since then $t = z \le y+z \le v$ implies $t \le v$. Hence we distinguish only four cases on the four bounding lines.

(Case 1). The minimizer satisfies $y = 0$. The problem is to minimize $g(z) = -4z^2 + (2s+4t)z = 2z(s+2t-2z)$ subject to $0 \le z \le t$ and $s+t-v \le z \le v$, and hence subject to $s+t-v \le z \le v$. Function $g(z)$ is concave, and we have $g(v) = 2v(s+2t-2v) = \gamma_2$, and (9) implies that $g(s+t-v) = 2(s+t-v)(2v-s) \ge \gamma_2$.

(Case 2). The minimizer satisfies $y = s$. The problem is to minimize $g(z) = 2(t-z)(2z+s)$ subject to $0 \le z \le t$ and $t-v \le z \le v-s$, and hence subject to $t-v \le z \le v-s$. Once again $g(z)$ is concave, and $g(t-v) = 2v(s+2t-2z) = \gamma_2$, and (9) implies that $g(v-s) = 2(s+t-v)(2v-s) \ge \gamma_2$.

(Case 3). The minimizer satisfies $y + z = s + t - v$. The problem is to minimize the increasing linear function $g(z) = 2(2v - s - t)z + 2(s + t - v)t$ subject to $0 \le z \le t$ and $t - v \le z \le s + t - v$. Then $z \ge t - v$ yields $g(z) \ge g(t - v) = \gamma_2$.

(Case 4). The minimizer satisfies $y+z = v$. The problem is to minimize the decreasing linear function $g(z) = 2vt - 2(2v - s - t)z$ subject to $0 \le z \le t$ and $v - s \le z \le v$. Then $z \le v$ yields $g(z) \ge g(v) = \gamma_2$. □

Theorem 3.3 *If A is a 1-2-1 block matrix and B is a multi-cut matrix in normal form with two blocks, then the identity permutation solves the QAP to optimality.*

Proof. If $v \ge s + t$, then the identity permutation yields an objective value of 0 which clearly is optimal. If $t \le v \le s + t$, then the identity permutation sets $y = s + t - v$ and $z = 0$ and yields objective value γ_1; by the lower bound in Lemma 3.1 this is optimal. If $v < t$, then the identity permutation sets $y = 0$ and $z = t - v$ and yields objective value γ_2; by the lower bound in Lemma 3.2 this is optimal. □

3.2 The general case

Now we are ready to establish our main result for the general case, where the multi-cut matrix B has an arbitrary number of blocks.

Theorem 3.4 *If A is a monotone anti-Monge matrix and B is a multi-cut matrix in normal form, then the identity permutation solves the QAP to optimality.*

Proof. The proof is done in two steps. In the first step, we assume that matrix A is a 1-2-1 block matrix. Let $I_1, \ldots, I_q$ with $|I_1| \le |I_2| \le \cdots \le |I_q|$ denote the blocks of matrix B. Consider an optimal permutation π^* for the QAP, and assume that π^* assigns the row set J_k of A to block I_k of B, where $1 \le k \le q$. The submatrix A' of A induced by the rows and columns in $J_k \cup J_{k+1}$ and the submatrix B' induced by the rows and columns in $I_k \cup I_{k+1}$ satisfy the conditions of Theorem 3.3. According to the theorem, we may repartition J_k and J_{k+1} such that all elements in J_k precede all the elements in J_{k+1}, as imposed by the identity permutation. Repeated application of such repartitioning eventually transforms π^* into the identity permutation without worsening the objective value.

In the second step, we consider the most general case with an arbitrary monotone anti-Monge matrix A. As A can be written as a non-negative linear combination of 1-2-1 block matrices, and as the identity permutation optimally solves the QAP between any 1-2-1 block matrix and matrix B, the identity permutation also optimally solves the QAP between A and B according to Fact 2.2. □

Next, we want to demonstrate that Theorem 3.4 generalizes the following proposition from the Monge literature.

Proposition 3.5 *(Pferschy, Rudolf & Woeginger [12])*
If A is a symmetric (not necessarily monotone) anti-Monge matrix and B is a multi-cut matrix with all blocks of identical size, the identity permutation solves the QAP to optimality.

Proof. First note that all row and column sums in matrix B are identical. The anti-Monge matrix A can be written as the sum of a monotone anti-Monge matrix A' and a sum matrix A''. Theorem 3.4 yields that the identity permutation optimally solves the QAP between A' and B, and Fact 2.1 yields that every permutation (and in particular the identity permutation) optimally solves the QAP between A'' and B. $\square$

The following two examples illustrate that in a certain sense the statement in Theorem 3.4 is best possible. If we allow A to be a general and not necessarily monotone anti-Monge matrix, the statement fails. If we take A from a slightly larger λ-generalized monotone anti-Monge cone λ-GMAM with $\lambda < 2$ (and do not restrict it to the standard cone with $\lambda = 2$), the statement fails.

Example 3.6 *Consider the QAP with the following non-monotone anti-Monge matrix A and the following multi-cut matrix B in normal form:*

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

Then the identity permutation has objective value 4*, whereas the permutation that switches the first row (and column) of A with its third row (and column) has a better objective value of* 2*.*

Example 3.7 *For $\lambda < 2$, consider the QAP with the following matrix A in the λ-GMAM cone and the following multi-cut matrix B in normal form:*

$$A = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & \lambda & \lambda \\ 1 & 1 & \lambda & \lambda \end{pmatrix} \qquad\qquad B = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix}$$

Then the identity permutation has objective value 8*, whereas the permutation that switches the second row (and column) of A with its fourth row (and column) has a better objective value of* $4 + 2\lambda$*.*

Finally, we show that another natural extension of the special case in Theorem 3.4 yields an NP-hard problem.

Lemma 3.8 *The QAP remains NP-hard, even if A is a monotone matrix and B is a multi-cut matrix.*

Proof. The proof is done by means of a reduction from the Graph Bisection problem (see for instance Garey & Johnson [8]) which is known to be NP-hard. The input for Graph Bisection consists of an undirected graph $G = (V, E)$ on $n = 2r$ vertices, together with an integer bound t. The goal is to decide whether there exists a partition of V into two subsets V_1 and V_2 with $|V_1| = |V_2| = r$, such that at most t edges in E connect V_1 to V_2.

We construct the following QAP instance. The $n \times n$ matrix A is the sum of the adjacency matrix A' of graph G and of the sum matrix A'' that is defined by $a''_{ij} = 2i+2j$ for $1 \le i, j \le n$; note that A indeed is monotone. The $n \times n$ matrix B is the multi-cut matrix with two blocks of size r. It is straightforward to verify with the help of Facts 2.1 and 2.2 that the Graph Bisection instance has answer YES, if and only if the constructed QAP instance has a feasible solution with objective value at most $n^2(n+1) + t$. $\square$

4 The Product-Block QAP

In this section we study the so-called *Product-Block QAP*, the special case where matrix A is a product matrix and where matrix B is a block matrix with some fixed pattern P. Throughout this section we assume that all patterns (and hence all considered block matrices) have rational entries. We stress that pattern P, and in particular the number q of blocks in P, are not part of the input. (Note that if P is part of the input, then one may choose $P = B$ so that matrix B essentially remains unrestricted.) We remind the reader that matrix P and all other matrices in this paper are symmetric.

The following definitions play a central role in our investigations. A *bad ensemble* for a $q \times q$ pattern matrix $P = (p_{ij})$ consists of the following:

- two indices r and s with $1 \le r < s \le q$,
- a real number γ with $0 \le \gamma \le 1$,
- real numbers ℓ_i with $0 \le \ell_i \le 1$ for $i \in \{1, \ldots, q\} \setminus \{r, s\}$.

With every bad ensemble, we associate the quadratic program QP-1 for non-negative real variables $x_1, \ldots, x_q$ in Figure 2. We will only consider ensembles for which the feasible region specified by (12a)–(12c) is non-empty. The crucial property of a bad ensemble is that QP-1 has a unique minimizer $(x_1^*, \ldots, x_q^*)$, and that this minimizer satisfies

$$0 < x_r^* < \gamma \qquad \text{and} \qquad 0 < x_s^* < \gamma. \tag{10}$$

In a similar spirit, we introduce *very bad ensembles* for a $q \times q$ pattern matrix $P = (p_{ij})$ that consist of the following:

- two indices r and s with $1 \le r < s \le q$,
- real numbers ℓ_i and u_i with $0 \le \ell_i < u_i \le 1$ for $i = 1, \ldots, q$.

With a very bad ensemble, we associate the quadratic program QP-2 in Figure 3. We stress that the upper bound constraints $x_i < u_i$ in QP-2 are strict. The crucial properties of a very bad ensemble are that QP-2 has a *unique* minimizer $(x_1^*, \ldots, x_q^*)$, that all the x_i^* are rational, and that

$$\ell_r < x_r^* < u_r \qquad \text{and} \qquad \ell_s < x_s^* < u_s. \tag{11}$$

$$\begin{aligned}
\min \quad & \sum_{i=1}^{q}\sum_{j=1}^{q} p_{ij}\, x_i x_j \\
s.t. \quad & \sum_{i=1}^{q} x_i = 1 && \text{(12a)} \\
& x_r + x_s = \gamma && \text{(12b)} \\
& x_i = \ell_i \qquad \text{for } i \in \{1,\ldots,q\} \setminus \{r,s\} && \text{(12c)}
\end{aligned}$$

Figure 2: The continuous quadratic program QP-1.

$$\begin{aligned}
\min \quad & \sum_{i=1}^{q}\sum_{j=1}^{q} p_{ij}\, x_i x_j \\
s.t. \quad & \sum_{i=1}^{q} x_i = 1 && \text{(13a)} \\
& \ell_i \le x_i < u_i \qquad \text{for } i = 1,\ldots,q. && \text{(13b)}
\end{aligned}$$

Figure 3: The continuous quadratic program QP-2.

In Sections 4.1 and 4.2 we will prove the following theorem, which settles the computational complexity of the Product-Block QAP for two large families of patterns.

Theorem 4.1 *Consider the Product-Block QAP with a fixed rational block pattern P.*

(i) If P does not allow any bad ensemble, the QAP is polynomially solvable.

(ii) If P has a very bad ensemble, the QAP is NP-hard.

In general, it is not straightforward to see whether a given pattern matrix allows a bad or very bad ensemble. The following two corollaries extract two clean and tidy pattern classes that are covered by Theorem 4.1.

Corollary 4.2 *The Product-Block QAP is polynomially solvable whenever the rational pattern matrix $P = (p_{ij})$ satisfies*

$$p_{ii} + p_{jj} \;\le\; 2p_{ij} \qquad \text{for } 1 \le i, j \le n. \tag{14}$$

Proof. Assume for the sake of contradiction that there is a bad ensemble and consider the corresponding minimizer $(x_1^*, \ldots, x_q^*)$. As every variable x_i with $i \notin \{r, s\}$ is frozen at $x_i = \ell_i$ and as variable x_s can be replaced by $\gamma - x_r$ according to (12b), QP-1 boils down to minimizing the quadratic function

$$g(x_r) \;=\; (p_{rr} + p_{ss} - 2p_{rs})\, x_r^2 + c_1\, x_r + c_0$$

subject to the constraint $0 \leq x_r \leq \gamma$; here the coefficients c_0 and c_1 are certain real numbers that depend on the ensemble. Condition (14) implies that the coefficient of the quadratic term x_r^2 is non-positive, so that $g(x_r)$ is concave and takes its minimum at $x_r = 0$ or $x_r = \gamma$. This contradicts (10), and hence Theorem 4.1.(i) applies. $\square$

Corollary 4.3 *The Product-Block QAP is NP-hard whenever there exist two indices r and s, for which the rational pattern matrix $P = (p_{ij})$ satisfies*

$$p_{rr} > p_{rs} \quad \textit{and} \quad p_{ss} > p_{rs}. \tag{15}$$

Proof. We construct a very bad ensemble for the 2×2 submatrix of P spanned by rows (and columns) r and s. If we set $\ell_r = \ell_s = 0$ and $u_r = u_s = 1$, routine calculations show that QP-2 is minimized at the rational point

$$x_r^* \;=\; \frac{p_{ss} - p_{rs}}{p_{rr} + p_{ss} - 2p_{rs}} \quad \text{and} \quad x_s^* \;=\; \frac{p_{rr} - p_{rs}}{p_{rr} + p_{ss} - 2p_{rs}}. \tag{16}$$

By (15) the numerators and denominators in (16) are positive, so that $0 < x_r^*, x_s^* < 1$. Hence this ensemble indeed is very bad and Theorem 4.1.(ii) applies. $\square$

As the quadratic programs QP-1 and QP-2 are easy to analyze for $q = 2$, some routine calculations yield the following corollary.

Corollary 4.4 *The Product-Block QAP with a 2×2 rational block pattern $P = (p_{ij})$ is NP-hard if $p_{11} > p_{12}$ and $p_{22} > p_{12}$, and it is polynomially solvable otherwise.* $\square$

4.1 Proof of the polynomial time result

In this section we prove the positive statement in Theorem 4.1.(i). Throughout we consider an $n \times n$ product matrix $A = (\alpha_i\, \alpha_j)$; we assume without loss of generality that $\alpha_1 \leq \alpha_2 \leq \cdots \leq \alpha_n$ so that A is monotone, and we furthermore assume that the values α_i are normalized so that $\sum_{i=1}^n \alpha_i = 1$. For a subset J of $\{1, \ldots, n\}$ we denote $\alpha(J) = \sum_{i \in J} \alpha_i$. Matrix B is a block matrix with a $q \times q$ pattern P that does not allow any bad ensemble; the blocks of B are $I_1, \ldots, I_q$.

Consider a permutation π that for $k = 1, \ldots, q$ assigns the row set J_k^π of A to block I_k of B. We say that the sets J_i^π and J_j^π are *separable*, if either all elements of J_i^π are less than or equal to all elements of J_j^π, or if all elements of J_i^π are greater than or equal to all elements of J_j^π.

Lemma 4.5 *There exists an optimal permutation π, such that for all $r \neq s$ the sets J_r^π and J_s^π are separable.*

Proof. Let π be an optimal permutation. We denote $y_k = \alpha(J_k^\pi)$ for $1 \le k \le q$, and we observe that the objective value of the QAP can be written as $\sum_{i=1}^q \sum_{j=1}^q p_{ij} y_i y_j$. We define an ensemble with $\gamma = y_r + y_s$, and with $\ell_i = u_i = y_i$ for $i \notin \{r, s\}$. In QP-1 we freeze every variable x_i with $i \notin \{r, s\}$ at its current value $x_i = y_i$, and we furthermore substitute $x_s = \gamma - x_r$. The resulting quadratic program asks to minimize a uni-variate quadratic function $g(x_r)$ subject to the single constraint $0 \le x_r \le \gamma$. As the pattern P does not allow bad ensembles, the minimum is taken at the boundary. Hence function $g(x_r)$ is either concave over $[0, \gamma]$, or it is convex and its minimizer lies outside of $[0, \gamma]$.

For the QAP this means that the objective value is minimized if $\alpha(J_r^\pi)$ either becomes as small or as large as possible. As matrix A is monotone, this in turn means that set J_r^π should consist either of the $|J_r^\pi|$ smallest or the $|J_r^\pi|$ largest elements in $J_r^\pi \cup J_s^\pi$, and that set J_s^π should consist of the remaining elements. In other words, we are able to separate the sets J_r^π and J_s^π without worsening the objective value. Repeated application of this separation step will eventually transform permutation π into the desired form. $\square$

Theorem 4.6 *If the $q \times q$ pattern P does not allow any bad ensemble, the Product-Block QAP with block pattern P is solvable in time $O(q^2 q! + n \log n)$.*

Proof. By Lemma 4.5, there are only $q!$ many cases to check. After sorting the numbers α_i and after performing some appropriate preprocessing, the objective value for every such case can be determined in $O(q^2)$ time. $\square$

As q is not part of the input, Theorem 4.6 completes the proof of Theorem 4.1.(i). Note that Theorem 3.4 shows that a *single* permutation is optimal for all instances in the considered multi-cut problem, whereas Theorem 4.6 leaves us with a huge number of $q!$ candidate permutations for the Product-Block QAP. The following example illustrates that even for a 2×2 pattern P, there is no way of sharpening our statements to a single optimal permutation.

Example 4.7 *Matrices A_1 and A_2 are monotone product matrices, and Corollary 4.2 shows that the pattern of block matrix B does not allow any bad ensemble:*

$$A_1 = \begin{pmatrix} 1 & 1 & 2 \\ 1 & 1 & 2 \\ 2 & 2 & 4 \end{pmatrix} \qquad A_2 = \begin{pmatrix} 1 & 2 & 2 \\ 2 & 4 & 4 \\ 2 & 4 & 4 \end{pmatrix} \qquad B = \begin{pmatrix} 0 & 2 & 2 \\ 2 & 1 & 1 \\ 2 & 1 & 1 \end{pmatrix}$$

By Lemma 4.5, there are two candidates for the optimal permutation: the identity permutation π_1 and the permutation π_2 that assigns the first/second/third row (and column) of matrix A_i respectively to the third/second/first row (and column) of matrix B. For the QAP between A_1 and B, permutation π_1 with objective value 21 loses to permutation π_2 with objective value 20. But for the QAP between A_2 and B, permutation π_1 with objective value 32 beats permutation π_2 with objective value 33.

4.2 Proof of the hardness result

In this section we prove the negative statement in Theorem 4.1.(ii). Throughout we consider a fixed $q \times q$ pattern P with a very bad ensemble, and its corresponding unique, rational minimizer $(x_1^*, \ldots, x_q^*)$. The optimal objective value of the quadratic program QP-2 is denoted $z^* = \sum_{i=1}^q \sum_{j=1}^q p_{ij}\, x_i^* x_j^*$.

If $x_i^* = 0$ holds for some i, then we may ignore this variable and its corresponding row and column in the pattern in our further considerations; hence we will assume without loss of generality that $x_i^* > 0$ holds for all i. We fix a large integer K such that $x_i^* K$ is integer for all i, and such that

$$K \;>\; \max\left\{\frac{2}{u_i - x_i^*},\, \frac{1}{x_i^*}\right\} \qquad \text{for } i = 1, \ldots, q. \tag{17}$$

and

$$K \;>\; \frac{1}{x_j^* - \ell_j} \qquad \text{for } j \in \{r, s\}. \tag{18}$$

The NP-hardness proof is done by means of a reduction from the following variant of the number partition problem (see Garey & Johnson [8]). An instance of Partition consists of a sequence $v_1, \ldots, v_m$ of positive rational numbers with $\sum_{k=1}^m v_k = 2$. The goal is to decide whether there exists a subset $M \subset \{1, \ldots, m\}$ such that $\sum_{k \in M} v_k = 1$ and $\sum_{k \notin M} v_k = 1$.

We begin the reduction by introducing the following numbers. We define an integer $L = mK$. Our choice of K in (17) yields $K > 1/x_i^*$, and consequently

$$x_i^* L \;>\; m \qquad \text{for all } i. \tag{19}$$

The bound (17) together with $L \geq K$ implies $x_i^* + 2/L < u_i$ for all i, and the bound (18) with $L \geq K$ implies $x_j^* - 1/L > \ell_j$ for $j \in \{r, s\}$.

Now let us construct an instance of the Product-Block QAP. The dimension is chosen as $n = L - 2$. The product matrix A is specified by the following α-values:

- For $k = 1, \ldots, m$, we introduce a partition-value $\alpha_k = (1 + v_k)/L$.
- For $k = m + 1, \ldots, n$, we introduce a dummy-value $\alpha_k = 1/L$.

Note that every α-value is at least $1/L$, and that the overall sum of all the α-values equals $(n + 2)/L = 1$. The block matrix B has blocks $I_1, \ldots, I_q$ and obeys the block pattern P. Block I_r has size $x_r^* L - 1$, block I_s has size $x_s^* L - 1$, and every remaining block I_i with $i \notin \{r, s\}$ has size $x_i^* L$. This completes the description of the QAP instance.

Lemma 4.8 *If the* Partition *instance has answer YES, then for the constructed instance of the QAP has a permutation with objective value at most z^*.*

Proof. Let $M \subset \{1, \ldots, m\}$ be a solution for the Partition instance. We define a permutation π that assigns the α-values of matrix A to the blocks of matrix B:

- To block I_r, we assign the $|M|$ partition-values α_k with $k \in M$ together with $x_r^* L - |M| - 1$ dummy-values. Note that the number $x_r^* L - |M| - 1$ is non-negative by (19), and note that the sum of all assigned α-values is x_r^*.
- To block I_s, we assign the remaining $m - |M|$ partition-values α_k with $k \notin M$ together with $x_s^* L - m + |M| - 1$ dummy-values. Note that $x_s^* L - m + |M| - 1$ is non-negative by (19), and note that the sum of all assigned α-values is x_s^*.
- To every remaining block I_i with $i \notin \{r, s\}$, we assign $x_i^* L$ dummy-values. The sum of all assigned α-values is x_i^*.

The resulting objective value for the QAP is $\sum_{i=1}^{q} \sum_{j=1}^{q} p_{ij}\, x_i^* x_j^*$, and hence coincides with z^*. □

Lemma 4.9 *If the constructed instance of the QAP has a permutation with objective value at most z^*, then the* Partition *instance has answer YES.*

Proof. Consider a permutation π for the QAP with objective value at most z^*. Let y_i denote the sum of all α-values that π assigns to block I_i, and let $\delta_i = y_i - |I_i|/L$. It is easily seen that $\delta_i \geq 0$ for all i and that $\sum_{i=1}^{q} \delta_i = 2/L$, which implies $0 \leq \delta_i \leq 2/L$. For $i \notin \{r, s\}$ we have $|I_i| = x_i^* L$, and hence $x_i^* \leq y_i \leq x_i^* + (2/L)$. For $j \in \{r, s\}$ we have $|I_j| = x_j^* L - 1$, and hence $x_j^* - (1/L) \leq y_j \leq x_j^* + (1/L)$. By our choice of K and L, these bounds imply $\ell_i \leq y_i < u_i$ for all i.

This means that $(y_1, \ldots, y_q)$ constitutes a feasible solution for the quadratic program QP-2 with objective value $\sum_{i=1}^{q} \sum_{j=1}^{q} p_{ij} y_i y_j \leq z^*$, and hence is a minimizer for QP-2. As we are working with a very bad ensemble, the minimizer is unique so that $y_i = x_i^*$ for all i. This leads to $\delta_i = 0$ for $i \notin \{r, s\}$, and $\delta_r = \delta_s = 1/L$. Consequently permutation π assigns all partition-values α_k to the two blocks I_r and I_s. If we define set M to contain all indices k for which the partition-value α_k is assigned to block I_r, it is easily seen that $\sum_{k \in M} v_k = 1$. Hence the Partition instance has answer YES. □

Lemma 4.8 and Lemma 4.9 together establish the correctness of our reduction. This completes the proof of Theorem 4.1.(ii).

5 Conclusions

We have studied a family of special cases of the quadratic assignment problem, where one matrix carries an anti-Monge structure and where the other matrix has a simple block structure. We identified a number of well-behaved cases that are solvable in polynomial time, and we also got some partial understanding of the borderline between certain easy and hard cases. Many questions remain open, and we will now list some of them.

Examples 3.6 and 3.7 describe scenarios that can not be settled with the methodology of Section 3. The precise complexity of these two scenarios remains open:

Problem 5.1 *Consider the QAP where A is a (general, not necessarily monotone) anti-Monge matrix and where B is a multi-cut matrix. Is this special case polynomially solvable?*

Problem 5.2 *For some fixed* $0 < \lambda < 2$*, consider the QAP where A lies in the generalized monotone anti-Monge cone* λ*-GMAM and where B is a multi-cut matrix. Is this special case NP-hard?*

Our results on the Product-Block QAP in Section 4 leave a considerable gap between the polynomially solvable area and the NP-hard area. This gap can be narrowed somewhat by allowing non-rational minimizers in the proof of Theorem 4.1.(ii), and by working with sufficiently precise rational approximations of all the involved real numbers; the technical details, however, would be gory. Corollaries 4.2 and 4.3 might indicate some vague connection to totally positive and totally negative matrices (see for instance Pinkus [13]) and to Eigenvalue spectra.

It is not clear that the gap actually can be closed, as for some patterns the problem might neither be polynomially solvable nor NP-hard. Schaefer's famous dichotomy theorem [15] states that every member of a large family of constraint satisfaction problems is either polynomially solvable or NP-complete. Is there a similar result for the Product-Block QAP?

Problem 5.3 *Is there a dichotomy theorem for the Product-Block QAP showing that every pattern P gives rise to either a polynomially solvable or an NP-complete problem?*

Acknowledgements. Part of this research was conducted while Vladimir Deineko and Gerhard Woeginger were visiting TU Graz, and were supported by the Austrian Science Fund (FWF): W1230, Doctoral Program "Discrete Mathematics". Vladimir Deineko acknowledges support by Warwick University's Centre for Discrete Mathematics and Its Applications (DIMAP). Gerhard Woeginger acknowledges support by DIAMANT (a mathematics cluster of the Netherlands Organization for Scientific Research NWO), and by the Alexander von Humboldt Foundation, Bonn, Germany.

References

[1] R.E. Burkard, E. Çela, G. Rote, and G.J. Woeginger (1998). The quadratic assignment problem with a monotone anti-Monge and a symmetric Toeplitz matrix: Easy and hard cases. *Mathematical Programming B82*, 125–158.

[2] R.E. Burkard, M. Dell'Amico, and S. Martello (2009). *Assignment Problems*. SIAM, Philadelphia.

[3] R.E. Burkard, B. Klinz, and R. Rudolf (1996). Perspectives of Monge properties in optimization. *Discrete Applied Mathematics 70*, 95–161.

[4] E. Çela (1998). *The Quadratic Assignment Problem: Theory and Algorithms.* Kluwer Academic Publishers, Dordrecht, The Netherlands.

[5] E. Çela, V.G. Deineko, and G.J. Woeginger (2012). Another well-solvable case of the QAP: Maximizing the job completion time variance. *Operations Research Letters 40*, 356–359.

[6] E. Çela, N. Schmuck, S. Wimer, and G.J. Woeginger (2011). The Wiener maximum quadratic assignment problem. *Discrete Optimization 8*, 411–416.

[7] V.G. Deineko and G.J. Woeginger (1998). A solvable case of the quadratic assignment problem. *Operations Research Letters 22*, 13–17.

[8] M.R. Garey and D.S. Johnson (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness.* Freeman, San Francisco.

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

[10] T. Lengauer (1990). *Combinatorial Algorithms for Integrated Circuit Layout.* John Wiley, Chichester.

[11] G. Monge (1781). Mémoires sur la théorie des déblais et des remblais. In *Histoire de l'Academie Royale des Sciences, Année M. DCCLXXXI, avec les Mémoires de Mathématique et de Physique, pour la même Année, Tirés des Registres de cette Académie*, Paris, 666–704.

[12] U. Pferschy, R. Rudolf, and G.J. Woeginger (1994). Monge matrices make maximization manageable. *Operations Research Letters 16*, 245–254.

[13] A. Pinkus (2010). *Totally Positive Matrices.* Cambridge University Press.

[14] R. Rudolf and G.J. Woeginger (1995). The cone of Monge matrices: extremal rays and applications. *Mathematical Methods of Operations Research 42*, 161–168.

[15] T.J. Schaefer (1978). The complexity of satisfiability problems. *Proceedings of the 10th Annual ACM Symposium on Theory of Computing (STOC'1978)*, 216–226.