Vision, Modeling, and Visualization (2017)
M. Hullin, R. Klein, T. Schultz, A. Yao (Eds.)

Efficient Lifted Relaxations of the Quadratic Assignment Problem

Oliver Burghard and Reinhard Klein

Bonn University, Germany

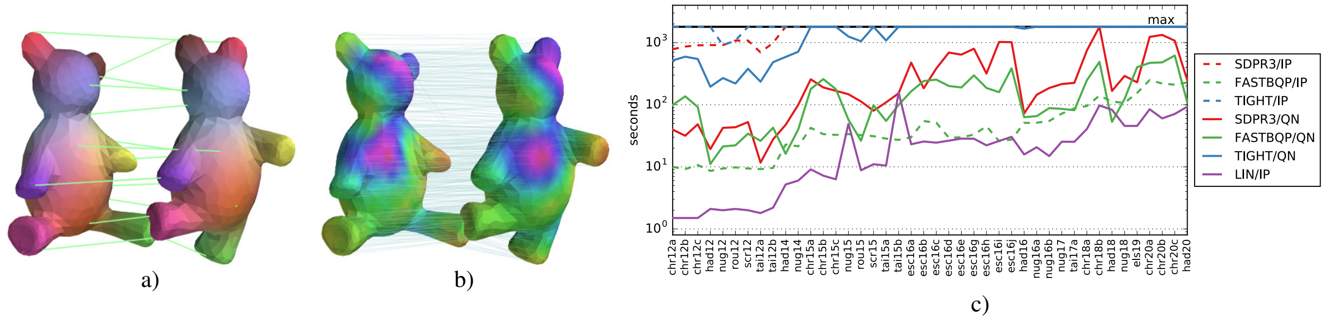

Figure 1: *QAPs can be used to model point assignment problems such as in (a), which are then further refined using linear assignment problems (b). We show that solving QAPs with linear relaxations is often sufficient and expectedly much faster, even when compared to fast approximative solvers for the SDPs (c) [WSvdHT16].*

Abstract

*Quadratic assignment problems (QAPs) and quadratic assignment matchings (QAMs) recently gained a lot of interest in computer graphics and vision, e.g. for shape and graph matching. Literature describes several convex relaxations to approximate solutions of the NP-hard QAPs in polynomial time. We compare the convex relaxations recently introduced in computer graphics and vision to established approaches in discrete optimization. Building upon a unified constraint formulation we theoretically analyze their solution spaces and their approximation quality. Experiments on a standard benchmark as well as on instances of the shape matching problems support our analysis. It turns out that often the bounds of a tight linear relaxation are competitive with the bounds of semidefinite programming (SDP) relaxations, while the linear relaxation is often much faster to calculate. Indeed, for many instances the bounds of the linear relaxation are only slightly worse than the SDP relaxation of Zhao [ZKRW98, PR09], which itself is at least as accurate as the relaxations currently used in computer graphics and vision. Solving the SDP relaxations can often be accelerated considerably from hours to minutes using the recently introduced approximation method for trace bound SDPs [WSvdHT16], but nonetheless calculating linear relaxations is faster in most cases. For the shape matching problem all relaxations generate the optimal solution, only that the linear relaxation does so faster. Our results generalize as well to QAMs for which we deliver new relaxations. Furthermore by interpreting the Product Manifold Filter [VLR*17] in the context of QAPs we show how to automatically calculate correspondences between shapes of several hundred points.*

CCS Concepts

•***Mathematics of computing*** → *Semidefinite programming; Convex optimization;* •***Computing methodologies*** → *Shape analysis;*

1. Introduction

Assigning two point sets, which were sampled on two different surfaces, onto each other is a discretization of the shape matching problem. We initially assume that the points have a one-to-one correspondence, which can then be represented by an n-element permutation contained in the symmetric group S_n. Good point-to-point assignments have small isometric distortion [BBM05, FS06, KKBL15] defined over the pairwise geodesic distances $d_{ij} \in \mathbb{R}$ and $d'_{ij} \in \mathbb{R}$, and some parameter σ:

$$\min_{\phi \in S_n} \sum_{ij} \exp\left(-\left(d_{ij} - d'_{\phi(i)\phi(j)}\right)^2 / \sigma^2\right) \tag{1}$$

DOI: 10.2312/vmv.20171267

The minimization of the simple functional already yields good point-to-point assignments as shown in Figure 1a. The minimization is an instance of a quadratic assignment problem (QAP) - a difficult discrete optimization problem whose solution is our main interest. For two cost matrices $\mathbf{A} \in \mathbb{R}^{n^2 \times n^2}, \mathbf{B} \in \mathbb{R}^{n \times n}$ and noting $\mathbf{A}_{pq,rs}$ for $\mathbf{A}_{pn+q,rn+s}$ the quadratic assignment problem (QAP) minimizes:

$$\text{(QAP-}\phi) \quad \min_{\phi \in S_n} \sum_{ij} \mathbf{A}_{i\phi(i),j\phi(j)} + 2\sum_i \mathbf{B}_{i\phi(i)} \qquad (2)$$

Setting $\mathbf{A}_{ik,jl} = \exp\left(-(d_{ij} - d'_{kl})^2/\sigma^2\right)$ transforms Eq. (1) in this new form. $\mathbf{B}$ can be removed from the formulation as the diagonal of $\mathbf{A}$ has the same effect.

Several practically relevant hard matching problems can be modeled as a QAP, such as matching shapes as above or matching feature points between images [BMP02, SRS07, CMC*09, EKG13]. Sadly approximating QAPs to any precision is already NP-hard [SG76] and solving instances with as little as 30 points is typically not considered any more practical.

Convex relaxations estimate a lower bound on the cost and can be used to approximate solutions with small costs. The idea is to drop non-convex constraints from the problem formulation such that the optimization becomes a convex one. Because a solution of the original problem is still a feasible solution of the relaxation, the efficiently computable minimal cost of the relaxation are a lower bound for the original costs. Furthermore as the solution of the relaxation fulfills all but the dropped constraints, projecting it onto the feasibility set of the original problem estimates a solution. The cost difference from the estimated to the minimal solution is smaller than the cost difference from the estimated solution to the lower bound of the relaxation, which is therefore a measure of the quality of the relaxation and is called the optimality gap.

Recent methods in computer graphics and vision relax QAPs into semidefinite programs [KKBL15, WSvdHT16], which we compare to already established convex relaxations of discrete optimization [AJ94, ZKRW98, PR09]. Interestingly on shape matching problems a carefully built linear programming relaxations is only slightly inferior to the best SDP relaxations, but solving it is often much faster. We furthermore investigate the approximation of the SDP relaxations by quasi-Newton minimization of the Lagrange dual [WSvdHT16] and which for certain relaxation is an order of magnitude faster than calculating their solution with interior-point methods.

We show theoretically that the SDP relaxation of Zhao [ZKRW98] yields lower bounds at least as large as the SDP relaxations currently used in computer graphics and vision, and this claim is supported by our practical evaluation of the QAPLIB [BKR97] benchmark and on typical shape matching problems. Interestingly the shape matching problems result in QAPs which are solved exactly with all investigated relaxations, only the linear methods are typically faster.

Furthermore we show how to use these relaxations to solve the related quadratic assignment matching (QAM) for which we present novel relaxations. Last but not least we show, that we can interpret the product manifold filter [VLR*17] as an iterative minimization of a QAP. This insight allows us to calculate correspondences of several hundred points *without* predefined correspondences.

2. Related Work

Koopmans [KB57] introduced a first restricted version of the QAP to locate economic activities. The more general formulation of Eq. (2) was presented by Lawler [Law63] soon after. Since then a variety of discrete optimization problems have been reformulated as QAPs, which therefore themselves became an important research topic. There is much related work on solving QAPs, reviewed in several good surveys [BCPP98, LdABN*07, Cel13], and in the following we only present the most relevant developments for our work.

Solving arbitrary QAPs is NP-hard as for example the traveling salesman problem can be modeled as a QAP. Despite extensive research, solving QAP instances with $n \geq 30$ is still not considered to be practical. Relaxations provide lower bounds for the minimal cost and estimate a solution. If the estimated solution is insufficient then the exact solution can be determined by Branch and Bound methods [Gil62, Law63, Ans03]. Hereby the solution space is traversed and subspaces, whose lower bound is larger than the cost of the currently best solution, are discarded. The Gilmore-Lawler bound [Gil62, Law63] is one of the earliest lower bounds and it is still used, due to its fast calculation. There are many other relaxations such as spectral relaxations [LH05, ADK13] (which were among the first used in computer graphics and vision), linear programs [HG98, KCCE99], mixed linear integer programs [KB78, FY83, AJ94], quadratic constraint quadratic programs [LMS*10] and semidefinite programs [GW95, LS91, Kar95, ZKRW98, PR09].

There are relaxations of varying sizes, for example using $\mathcal{O}(n^2)$ [PZLT15, dKST15] or $\mathcal{O}(n^4)$ [FY83, AJ94, ZKRW98] variables. Our interest is in linear and SDP relaxations over the lifted permutation matrices using $\mathcal{O}(n^4)$ variables (as will be introduced below), which are known to deliver tight bounds. They depend on the second-order Birkhoff polytope [JK96a, JK96b, JK01, AM14], whose affine subspace is known but not all of its facets.

Recently there is growing interest in relaxations as copositive programs [PVZ15, Bur12, PR09, BMP12], which minimize a linear objective under linear constraints over the convex set of copositive matrices, i.e. matrices with $\mathbf{x}^T\mathbf{A}\mathbf{x} \geq 0$ for all $\mathbf{x} \geq 0$. Although solving copositive programs is NP-hard, as several NP-hard problems can be modeled as copositive programs, there are SDP approximations of copositive programs to any accuracy.

Solving large SDPs with current interior point solvers is costly, especially the semidefinite programming relaxations of QAP of size $\mathcal{O}(n^4)$, and several methods specialize in solving or approximating QAP relaxations and the related binary integer programs. For example by reformulating the convex program with a separable cost function [BV06], by approximating the relaxation with the bundle method [RS06] and recently with a fast approximative semidefinite program solver of trace bound SDPs [WSVDH13, WSvdHT16]. We quickly present and evaluate the latter as for some relaxations it is a magnitude faster than current interior point solvers.

Notation. $(\mathbf{a}, \mathbf{b}, \mathbf{c}, \dots)$ are vectors, $\mathbf{a} \leq \mathbf{b}$ is component-wise less or equal, and $\|\mathbf{a}\|_{1/2}$ is the one/two norm. $(\mathbf{A}, \mathbf{B}, \mathbf{C}, \dots)$ are matrices, $\text{rank}(\mathbf{A})/\text{tr}(\mathbf{A})/\|\mathbf{A}\|_F$ is their rank/trace/Frobenius norm and $\langle \mathbf{A}, \mathbf{B}\rangle = \text{tr}(\mathbf{A}^T\mathbf{B})$ their inner-product. We write $[\mathbf{A}] \in \mathbb{R}^{n^2}$ for the row-wise unrolling of a matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$. For a matrix $\mathbf{Y} \in$

$\mathbb{R}^{n^2 \times n^2}$ we write $\mathbf{Y}_{pq,rs}$ as a shorthand for $\mathbf{Y}_{p\cdot n+q, r\cdot n+s}$. Furthermore $\mathbf{1}$ is a vector of ones of suitable dimension and $\mathcal{S}^n$ are the real, symmetric $n \times n$ matrices. For $\mathbf{A}, \mathbf{B} \in \mathcal{S}^n$ we denote the Loewner order with $\succcurlyeq$, that is $\mathbf{A} \succcurlyeq \mathbf{B}$ iff $\mathbf{A} - \mathbf{B}$ positive semidefinite.

3. Solving QAP

A permutation ϕ can be represented by a matrix $\mathbf{X}^\phi$ where $\mathbf{X}^\phi_{ij} = 1$ if $\phi(i) = j$ and 0 otherwise. Then the following is an equivalent formulation of (QAP-ϕ):

$$\text{(QAP)} \quad \min_X \ [\mathbf{X}]^T \mathbf{A} [\mathbf{X}] + 2\langle \mathbf{B}, \mathbf{X} \rangle \tag{3}$$
$$\text{s.t.} \quad \mathbf{X} \in \mathbb{R}^{n \times n},\ \mathbf{X}^2_{pq} = \mathbf{X}_{pq}\ \forall p,q$$
$$0 \le \mathbf{X},\ \mathbf{X1} = \mathbf{1},\ \mathbf{X}^T\mathbf{1} = \mathbf{1}$$

The convex hull of the permutation matrices $\{\mathbf{X}^\phi \mid \phi \in S_n\}$ is called the *Birkhoff polytope* Π^n [AM14]:

$$\Pi^n := \text{conv}\left(\{\mathbf{X}^\phi \mid \phi \in S_n\}\right)$$
$$= \{\mathbf{X} \in \mathbb{R}^{n \times n} \mid 0 \le \mathbf{X},\ \mathbf{X1} = \mathbf{1},\ \mathbf{X}^T\mathbf{1} = \mathbf{1}\}$$

3.1. Lifted variables

We reformulate (QAP) by replacing the quadratic factors $\mathbf{X}_{pq}\mathbf{X}_{rs}$ with the lifted variables $\mathbf{Y}_{pq,rs}$. Each permutation ϕ induces a lifted feasible solution of the form $(\mathbf{X}^\phi, \mathbf{Y}^\phi) := (\mathbf{X}^\phi, [\mathbf{X}^\phi][\mathbf{X}^\phi]^T)$, which is called a second-order permutation. The convex hull of the second-order permutations is called the *second-order Birkhoff polytope* Π^n_2 [AM14]:

$$\Pi^n_2 := \text{conv}\left(\left\{\left(\mathbf{X}^\phi, \mathbf{Y}^\phi\right) \mid \phi \in S_n\right\}\right) \tag{4}$$

Several authors explored the second-order Birkhoff polytope and the isomorphic QAP-polytope [JK96a] (which describes the lifted permutations with a graph structure):

Theorem 1. *[JK96a, JK01] The affine hull of Π^n_2 is:*

$$\mathbf{X}\mathit{1} = \mathbf{X}^T\mathit{1} = \mathit{1} \tag{5a}$$
$$\mathbf{Y} = \mathbf{Y}^T,\ \text{diag}(\mathbf{Y}) = [\mathbf{X}] \tag{5b}$$
$$\mathbf{Y}_{kq,ks} = \mathbf{Y}_{qk,sk} = 0 \qquad \forall k \forall q \ne s \tag{5c}$$
$$\textstyle\sum_k \mathbf{Y}_{pq,ks} = \sum_k \mathbf{Y}_{pq,sk} = \mathbf{X}_{pq} \qquad \forall p,q,s \tag{5d}$$

Theorem 2. *[JK96a, AM14] Some facets of Π^n_2 are given by:*

$$0 \le \mathbf{Y} \tag{6}$$

For $3 \le m \le n-3$ and $(i_0, \dots, i_m), (j_0, \dots, j_m) \in [1:n]^{m+1}$ pairwise disjunct other facets are:

$$\textstyle\sum_r \mathbf{Y}_{i_r j_r, i_0 j_0} + \mathbf{Y}_{i_r j_r, i_r j_r} \le \mathbf{X}_{i_0 j_0} + \sum_r \sum_s \mathbf{Y}_{i_r j_r, i_s r_s} \tag{7}$$

And there are additional currently unknown facets.

Every second-order permutation fulfills the following semidefinite constraints, whose equivalence follows from the Schur complement:

$$\mathbf{Y} \succcurlyeq [\mathbf{X}][\mathbf{X}]^T \qquad \left(\Leftrightarrow \begin{pmatrix} 1 & [\mathbf{X}]^T \\ [\mathbf{X}] & \mathbf{Y} \end{pmatrix} \succcurlyeq 0\right) \tag{8}$$

Because Eq. (8) is a convex constraint fulfilled on all second-order permutations it is fulfilled on their convex hull as well:

Proposition 1. *Eq. (8) holds for all second-order permutations as well as for any tuple in Π^n_2.*

3.2. QAP relaxations

In the following we define several convex relaxations of (QAP). Over a fixed representation of the solutions with variables, the convex relaxation with the largest lower bounds is the one with the smallest convex solution space that still contains all valid solutions, i.e. the convex hull of the valid solutions. Over the lifted permutations (LIN-OPT) is the convex relaxation with the largest lower bound as it restricts the solutions to the second-order Birkhoff polytope Π^n_2:

$$\text{(LIN-OPT)} \quad \min_{\mathbf{X},\mathbf{Y}} \langle \mathbf{A}, \mathbf{Y} \rangle + 2\langle \mathbf{B}, \mathbf{X} \rangle$$
$$\text{s.t.} \quad (\mathbf{X}, \mathbf{Y}) \in \Pi^n_2$$

Despite being a linear program, solving (LIN-OPT) is difficult. Not only are not all facets of Π^n_2 known, but it is also NP-hard:

Theorem 3. *(LIN-OPT) and (QAP) have the same minimal cost. Because the decision problem of QAP ("is there a solution of cost less than x") is NP-complete, so (LIN-OPT) is NP-hard as well.*

Proof. Every minimal solution of (LIN-OPT) can be expressed as a convex combination of second-order permutations $\phi_1, \dots, \phi_k \in S_n$: $(\mathbf{X}, \mathbf{Y}) = \sum_i \alpha_i (\mathbf{X}^{\phi_i}, \mathbf{Y}^{\phi_i})$ with $\alpha_i > 0, \sum_i \alpha_i = 1$. As the cost $f(\mathbf{X}, \mathbf{Y}) := \langle \mathbf{A}, \mathbf{Y} \rangle + 2\langle \mathbf{B}, \mathbf{X} \rangle$ is *linear* in $\mathbf{X}$ and $\mathbf{Y}$, we have $f(\mathbf{X}, \mathbf{Y}) = \sum_i \alpha_i f(\mathbf{X}^{\phi_i}, \mathbf{Y}^{\phi_i})$. Because $f(\mathbf{X}, \mathbf{Y})$ is minimal, the summands of $0 = \sum_i \alpha_i (f(\mathbf{X}^{\phi_i}, \mathbf{Y}^{\phi_i}) - f(\mathbf{X}, \mathbf{Y}))$ are not negative and therefore all 0. Thus $f(\mathbf{X}^{\phi_i}, \mathbf{Y}^{\phi_i}) = f(\mathbf{X}, \mathbf{Y})$ for all i. □

(LIN) [FY83, AJ94] is an efficiently solvable linear programming approximation of (LIN-OPT). It replaces Π^n_2 with the approximation of Theorems 1 and 2, dropping the exponential number of facets from Eq. (7). It has the same affine subspace as Π^n_2, as Eq. (5c) follows from Eqs. (10b) and (10c):

$$\text{(LIN)} \quad \min_{\mathbf{X},\mathbf{Y}} \langle \mathbf{A}, \mathbf{Y} \rangle + 2\langle \mathbf{B}, \mathbf{X} \rangle$$
$$\text{s.t.} \quad \mathbf{X1} = \mathbf{X}^T\mathbf{1} = \mathbf{1} \tag{10a}$$
$$0 \le \mathbf{Y},\ \text{diag}(\mathbf{Y}) = [\mathbf{X}] \tag{10b}$$
$$\textstyle\sum_k \mathbf{Y}_{pq,ks} = \sum_k \mathbf{Y}_{pq,sk} = \mathbf{X}_{pq} \quad \forall p,q,s \tag{10c}$$
$$\text{with} \quad \mathbf{X} \in \mathbb{R}^{n \times n}; \mathbf{Y} \in \mathcal{S}^{n^2}$$

A tighter relaxation (SDP-R3) follows by adding the semidefinite constraint from Eq. (8) to (LIN). It was introduced in [ZKRW98] and reformulated as presented here in [PR09]:

$$\text{(SDP-R3)} \quad \min_{\mathbf{X},\mathbf{Y}} \langle \mathbf{A}, \mathbf{Y} \rangle + 2\langle \mathbf{B}, \mathbf{X} \rangle$$
$$\text{s.t.} \quad \text{as in (LIN) and}$$
$$0 \preccurlyeq \begin{pmatrix} 1 & [\mathbf{X}]^T \\ [\mathbf{X}] & \mathbf{Y} \end{pmatrix} \tag{11a}$$

Adding one of the following two equivalent non-convex constraints

$$\text{rank}\begin{pmatrix} 1 & [\mathbf{X}]^T \\ [\mathbf{X}] & \mathbf{Y} \end{pmatrix} = 1 \quad \Leftrightarrow \quad Y = [\mathbf{X}][\mathbf{X}]^T$$

to (SDP-R3) makes it equivalent to (QAP), as $\mathbf{X}_{pq} \in \{0,1\}$ follows from $Y = [\mathbf{X}][\mathbf{X}]^T$ and $\mathbf{X}_{pq}^2 = \mathbf{Y}_{pq,pq} = \mathbf{X}_{pq}$.

We compare the above relaxation against two state-of-the-art relaxations of the recent computer graphics [KKBL15] and vision literature [WSvdHT16]. Both are relaxations of a variation of the QAP, which minimizes over the partial permutations, i.e. it assigns subsets of the nodes onto each other. Here we present their relaxations adapted to (QAP) and postpone a discussion of the differences to Section 3.5:

$$\text{(TIGHT)} \quad \min_{\mathbf{X},\mathbf{Y}} \langle \mathbf{A},\mathbf{Y}\rangle + 2\langle \mathbf{B},\mathbf{X}\rangle$$

$$\text{s.t.} \quad 0 \le \mathbf{X},\ \mathbf{X1} = \mathbf{X}^T\mathbf{1} = \mathbf{1} \tag{12a}$$

$$0 \le \mathbf{Y},\ \text{tr}(\mathbf{Y}) = n,\ \mathbf{1}^T\mathbf{Y1} = n^2 \tag{12b}$$

$$\mathbf{Y}_{pq,ks} \le \mathbf{X}_{pq} \qquad \forall p,q,k,s \tag{12c}$$

$$\mathbf{Y}_{kq,ks} = \mathbf{Y}_{qk,sk} = 0 \qquad \forall k \forall q \forall s \ne q \tag{12d}$$

$$0 \preccurlyeq \begin{pmatrix} 1 & [\mathbf{X}]^T \\ [\mathbf{X}] & \mathbf{Y} \end{pmatrix} \tag{12e}$$

$$\text{with} \quad \mathbf{X} \in \mathbb{R}^{n\times n}; \mathbf{Y} \in \mathcal{S}^{n^2}$$

$$\text{(FASTBQP)} \quad \min_{\mathbf{X},\mathbf{Y}} \langle \mathbf{A},\mathbf{Y}\rangle + 2\langle \mathbf{B},\mathbf{X}\rangle$$

$$\text{s.t.} \quad \mathbf{X1} = \mathbf{X}^T\mathbf{1} = \mathbf{1} \tag{13a}$$

$$\text{diag}(\mathbf{Y}) = [\mathbf{X}] \tag{13b}$$

$$\mathbf{Y}_{kq,ks} = \mathbf{Y}_{qk,sk} = 0 \quad \forall k \forall q \forall s \ne q \tag{13c}$$

$$0 \preccurlyeq \begin{pmatrix} 1 & [\mathbf{X}]^T \\ [\mathbf{X}] & \mathbf{Y} \end{pmatrix} \tag{13d}$$

$$\text{with} \quad \mathbf{X} \in \mathbb{R}^{n\times n}; \mathbf{Y} \in \mathcal{S}^{n^2}$$

We say that a relaxation (A) *dominates* another relaxation (B) and write (A) $\ge$ (B) if they have the same cost function and if the feasibility set of (A) is contained in the feasibility set of (B). In this case the minimum of (B) is a *lower bound* for the minimum of (A).

Theorem 4. *Using this notation the following relations hold:*

$$\text{(LIN-OPT)} \ge \text{(SDP-R3)}$$
$$\text{(SDP-R3)} \ge \text{(LIN)}$$
$$\text{(SDP-R3)} \ge \text{(TIGHT)}$$
$$\text{(SDP-R3)} \ge \text{(FASTBQP)}$$

All relaxations have $\mathcal{O}(n^4)$ variables and $\mathcal{O}(n^4)$ constraints, except for (FASTBQP) which has only $\mathcal{O}(n^3)$ constraints.

Proof. We show all relations by showing that the feasibility sets are subsets of each other. Due to Proposition 1 the feasible set of (LIN-OPT) is a subset of the feasibility set of (SDP-R3). The feasibility sets of (LIN) + Eq. (11a), (TIGHT) + Eq. (10b) + Eq. (10c) and (FASTBQP) + Eq. (10b) + Eq. (10c) are equivalent to the feasible set of (SDP-R3). □

We conclude that, the lower bound of (SDP-R3) is at least as large as the other lower bounds, except for the lower bound of (LIN-OPT) which cannot be efficiently computed. Typically the larger the lower bound, the less projection onto the original feasible set changes the solution, so that (SDP-R3) is as well likely to have one of the tightest duality gaps. Despite the bounds of (LIN) being weaker than the bounds of (SDP-R3), solving a linear program is often faster than solving a semidefinite program, so that (LIN) offers a trade-off between performance and the quality of the bounds.

3.3. Fast approximation of semidefinite programs

While (LIN) can be solved efficiently with an interior-point solver, solving the SDP relaxations (SDP-R3), (TIGHT) and (FASTBQP) with interior-point solvers is often slow. Recently the authors of [WSvdHT16] proposed a fast approximation method for SDPs, whose solution has a fixed trace, which all previous SDP relaxation have ($\text{tr}(Y) = n$). For completeness we quickly rephrase the necessary steps.

After subsituting $\mathbf{X}$ and $\mathbf{Y}$ by the new variable

$$\mathbf{Z} = \mathbf{Z}[\mathbf{X},\mathbf{Y}] := \begin{pmatrix} 1 & [\mathbf{X}]^T \\ [\mathbf{X}] & \mathbf{Y} \end{pmatrix} .$$

we express (SDP-R3), (TIGHT) and (FASTBQP) in the form:

$$\text{(SDP-F)} \quad \min_{\mathbf{Z}\succcurlyeq 0} \langle \mathbf{A}_0,\mathbf{Z}\rangle$$
$$\text{s.t.} \quad \langle \mathbf{B}_i,\mathbf{Z}\rangle = b_i \quad i \in 1,\dots,j$$
$$\langle \mathbf{B}_i,\mathbf{Z}\rangle \le b_i \quad i \in j+1,\dots,J$$

We approximate (SDP-F) with another convex program (SDP-A):

$$\text{(SDP-A)} \quad \min_{\mathbf{Z}\succcurlyeq 0} \langle \mathbf{A}_0,\mathbf{Z}\rangle + \frac{1}{2\gamma}\|\mathbf{Z}\|_F^2$$
$$\text{s.t.} \quad \text{as in (SDP-F)}$$

The difference of the minimal costs of (SDP-F) and (SDP-A) depends on $\|\mathbf{Z}\|_F$ and γ. Let $\lambda_i \ge 0$ be the eigenvalues of $\mathbf{Z}$, then

$$\|\mathbf{Z}\|_F^2 = \sum \lambda_i^2 \le \left(\sum \lambda_i\right)^2 = \text{tr}(\mathbf{Z})^2 = (n+1)^2 \tag{14}$$

and the minimal costs of (SDP-A) and (SDP-F) are related by

$$c_{\text{SDP-A}} - \frac{(n+1)^2}{2\gamma} < c_{\text{SDP-F}} \le c_{\text{SDP-A}} . \tag{15}$$

For γ large enough the solutions of (SDP-F) and (SDP-A) are arbitrarily close and solving variants of (SDP-A) yields arbitrary precise upper and lower bounds on (SDP-F).

Let $\Pi(\mathbf{C}) = \sum_i \max(0,\lambda_i)\ \phi_i\phi_i^T$ be the projection of a matrix $\mathbf{C}$ with the eigendecomposition $\mathbf{C} = \sum_i \lambda_i\phi_i\phi_i^T$ onto the positive semidefinite cone $\{\mathbf{X} \in \mathcal{S}_n \mid \mathbf{X} \succcurlyeq 0\}$. Then (SDP-A) can be minimized with a quasi-Newton method on the dual problem:

Theorem 5 ([WSVDH13, WSvdHT16])**.** *If (SDP-F) is feasible, so is (SDP-A) for which then strong duality holds. Instead of minimizing (SDP-A) we can maximize its dual:*

$$\max_{\mathbf{u}\in\hat{\mathbf{U}}} d_\gamma(\mathbf{u}) = -\mathbf{b}^T\mathbf{u} - \frac{\gamma}{2}\|\Pi(\mathbf{C}(\mathbf{u}))\|_F^2 \tag{16}$$

where

$$\hat{\boldsymbol{U}} = \left\{\boldsymbol{u} \in \mathbb{R}^J \mid \boldsymbol{u}_i \ge 0 \ \forall i \in j+1,\dots,J\right\} \tag{17}$$

$$\mathbf{C}(\mathbf{u}) = -\mathbf{A}_0 - \sum_i \mathbf{u}_i\mathbf{B}_i . \tag{18}$$

The dual is once but not twice differentiable and its gradient is

$$(\nabla d_\gamma)_i = +\langle \mathbf{B}_i, \Pi(\mathbf{C}(\mathbf{u}))\rangle - b_i \ . \tag{19}$$

3.4. Extracting a solution

The above convex relaxations solve for $\mathbf{X}$ and $\mathbf{Y}$, yet what is required is a permutation $\phi \in S_n$. If $\mathbf{Z}[\mathbf{X},\mathbf{Y}] \succcurlyeq 0$ and $\mathrm{rank}(\mathbf{Z}[\mathbf{X},\mathbf{Y}]) = 1$ then $\mathbf{X}$ is indeed a solution of (QAP), as noted in the comments of (SDP-R3). But often the rank of $\mathbf{Z}[\mathbf{X},\mathbf{Y}]$ is larger than 1. In this case we can project $\mathbf{X}$ onto the possible permutation matrices Π^n by solving the Linear Assignment Problem:

$$\min_{\phi \in S_n} \left\| \mathbf{X}^\phi - \frac{n}{[\mathbf{X}]^T \mathbf{1}} \mathbf{X} \right\|_1 \tag{20}$$

Often better results are obtained using the randomized approach described in [LMS*10]. Let $\xi \in \mathbb{R}^{n^2}$ be a vector sampled from a centered multidimensional normal distribution $\mathcal{N}(0,\mathbf{Z})$ with covariance $\mathbf{Z}$. For any matrix $\mathbf{A}_0$ the expected value of $\xi^T \mathbf{A}_0 \xi$ is

$$\mathrm{E}_{\xi \sim \mathcal{N}(0,\mathbf{Z})}[\xi^T \mathbf{A}_0 \xi] = \langle \mathbf{Z}, \mathbf{A}_0 \rangle \ .$$

Thus (SDP-F) solves for a matrix $\mathbf{Z}$ such that sampling from $\mathcal{N}(0,\mathbf{Z})$ fulfills and minimizes (SDP-F) *in expectation*. It is therefore reasonable to sample several solutions $\xi_i \sim \mathcal{N}(0,\mathbf{Y})$, project each onto the permutations and choose the one with minimal cost:

$$\phi_i = \min_{\phi \in S_n} \left\| [\mathbf{X}^\phi] - \frac{n}{1^T \xi_i} \xi_i \right\|_1$$
$$\phi = \min_i \ [\mathbf{X}^{\phi}]^T \mathbf{A}_0 [\mathbf{X}^\phi] + 2\,\langle B, \mathbf{X}^\phi \rangle$$

3.5. Quadratic Assignment Matching

A partial permutation is a bijection from k of n elements onto k of m elements. Quadratic Assignment Matching (QAM) [KKBL15] is a generalization of QAP to partial permutations, used for example for partial graph matching. QAM minimizes the following cost defined with the matrices $\mathbf{A} \in \mathbb{R}^{nm \times nm}$ and $\mathbf{B} \in \mathbb{R}^{n \times m}$ over the partial permutations:

$$\text{(QAM)} \quad \min \ \sum_{ij} \mathbf{A}_{\phi(i)\psi(i),\phi(j)\psi(j)} + \sum_i \mathbf{B}_{\phi(i)\psi(i)} \tag{21}$$
$$\text{s.t.} \quad \phi\colon \{1,\dots,k\} \to \{1,\dots,n\},\ \phi \text{ injective}$$
$$\psi\colon \{1,\dots,k\} \to \{1,\dots,m\},\ \psi \text{ injective}$$

We can model (QAM) as a QAP of size $\hat{n} = n+m-k$ by adding extra nodes on both shapes (loosely following an idea from [MD58]). With $\mathbf{A}' \in \mathbb{R}^{\hat{n}^2 \times \hat{n}^2}$, $\mathbf{B}' \in \mathbb{R}^{\hat{n}^2}$ as follows

$$\mathbf{A}'_{pq,rs} = \begin{cases} \mathbf{A}_{pq,rs} & p,r \le n \text{ and } q,s \le m \\ 0 & \text{otherwise} \end{cases}$$

$$\mathbf{B}'_{pq} = \begin{cases} \mathbf{B}_{pq} & p \le n \text{ and } q \le m \\ 0 & \text{otherwise} \end{cases}$$

we define the following problem:

$$\text{(QAM-QAP)} \quad \text{Minimize the QAP defined by } \mathbf{A}', \mathbf{B}'$$
$$\text{s.t.} \quad \mathbf{X}_{pq} = 0 \quad \forall p > n\, \forall q > m$$

To show the equality of (QAM) and (QAM-QAP) let ϕ be a solution of (QAM-QAP). Then $\mathbf{X} = \mathbf{X}^\phi$ it is of the form:

$$\mathbf{X} = \begin{pmatrix} \mathbf{E} & \mathbf{F} \\ \mathbf{G} & 0 \end{pmatrix} \text{ with } \mathbf{E} \in \mathbb{R}^{n \times m}, \mathbf{F} \in \mathbb{R}^{n \times (n-k)}, \mathbf{G} \in \mathbb{R}^{(m-k) \times m}$$

$\mathbf{X}$ has exactly one 1 in each row and column and in total $n+m-k$ ones. Thus $\mathbf{F}$ has $n-k$, $\mathbf{G}$ has $m-k$ and $\mathbf{E}$ has k ones. Let $(\phi(1),\psi(1)),\dots,(\phi(k),\psi(k))$ be the indices of the ones of $\mathbf{E}$ then ϕ, ψ defines a solution of QAM. On the other hand if ϕ and ψ is a solution of QAM then we can define a solution of (QAM-QAP) of equal cost by setting $\mathbf{E}_{\phi(i)\psi(i)} = 1$ and 0 otherwise, and filling $\mathbf{F}$ and $\mathbf{G}$ such that the constraints are met. Due to the definition of $\mathbf{A}'$ the costs of both solutions are equivalent and one is optimal if and only if the other is, thus:

Theorem 6. *The problems (QAM) and (QAM-QAP) are equivalent and a solution of one leads to the solution of the other.*

We can therefore utilize the previous relaxations (SDP-R3) and (LIN) to solve (QAM), which we name (QAM-SDP-R3) and (QAM-LIN).

But we can also build smaller, less tight relaxations by dropping F and/or G altogether, turning equality into inequality constraints where necessary:

$$\text{(QAM-LD)} \quad \min_{\mathbf{X},\mathbf{Y}} \ \langle \mathbf{A},\mathbf{Y}\rangle + 2\langle \mathbf{B},\mathbf{X}\rangle$$
$$\text{s.t.} \quad 0 \le \mathbf{X},\ \mathbf{X1} \le \mathbf{1},\ \mathbf{X}^T\mathbf{1} \le \mathbf{1},\ \mathbf{1}^T\mathbf{X1} = k \tag{22a}$$
$$0 \le \mathbf{Y},\ \mathrm{diag}(\mathbf{Y}) = [\mathbf{X}],\ \mathbf{1}^T\mathbf{Y1} = k^2 \tag{22b}$$
$$\max(\textstyle\sum_k \mathbf{Y}_{pq,ks}, \sum_k \mathbf{Y}_{pq,sk}) \le \mathbf{X}_{pq} \ \forall p,q,s \tag{22c}$$
$$\mathbf{Y}_{kq,ks} = \mathbf{Y}_{qk,sk} = 0 \qquad \forall k \forall q \forall s \ne q \tag{22d}$$
$$0 \preccurlyeq \begin{pmatrix} 1 & [\mathbf{X}]^T \\ [\mathbf{X}] & \mathbf{Y} \end{pmatrix} \tag{22e}$$
$$\text{with} \quad \mathbf{X} \in \mathbb{R}^{n \times m}; \mathbf{Y} \in \mathcal{S}^{nm}$$

This is similar to the QAM relaxation of [KKBL15], which we add for completeness:

$$\text{(QAM-TIGHT)} \quad \min_{\mathbf{X},\mathbf{Y}} \ \langle \mathbf{A},\mathbf{Y}\rangle + 2\langle \mathbf{B},\mathbf{X}\rangle$$
$$\text{s.t.} \quad 0 \le \mathbf{Y},\ \mathrm{tr}(\mathbf{Y}) = k,\ \mathbf{1}^T\mathbf{Y1} = k^2 \tag{23a}$$
$$\mathbf{Y}_{pq,ks} \le \mathbf{X}_{pq} \qquad \forall p,q,k,s \tag{23b}$$
$$\text{and Eqs. 22a, 22d, 22e and } \mathbf{X}, \mathbf{Y} \text{ as above}$$

Theorem 7. *For the relaxations the following relations hold:*

(QAM-SDP-R3) $\ge$ (QAM-LIN)
(QAM-SDP-R3) $\ge$ (QAM-LD) $\ge$ (QAM-TIGHT)

(QAM-SDP-R3) and (QAM-LIN) use $(n+m-k+1)^4$ variables and (QAM-LD) and (QAM-TIGHT) use $(nm+1)^2$ variables.

Proof. The first relation follows from Theorem 4. "(QAM-LD) $\ge$ (QAM-TIGHT)": The feasibility set of (QAM-TIGHT) + Eq. (22b) + Eq. (22c) is equivalent to the feasibility set of (QAM-LD). "(QAM-SDP-R3) $\ge$ (QAM-LD)": Let $\mathbf{X}$, $\mathbf{Y}$ be a solution of (QAM-SDP-R3). Then $\mathbf{X}' = \mathbf{E}$ with $\mathbf{E}$ as above and $\mathbf{Y}' \in \mathcal{S}^{nm}$ with $\mathbf{Y}'_{pq,rs} = \mathbf{Y}_{pq,rs} \forall p,q,r,s$ are a valid solution of (QAM-LD) and by construction of $\mathbf{A}'$ of equal cost. □

In conclusion modelling (QAM) as (QAP-SDP-R3) gives lower bounds at least as large as the other relaxations. For k large solving (QAM) via (QAM-LIN) is likely fastest and still gives good results. If k is small (QAM-QAP) is significantly larger than the relaxation (QAM-LD), which are therefore fastest. The lower bound of (QAM-LD) is always larger than the lower bound of (QAM-TIGHT).

4. Evaluation and Applications

We compare solving QAPs by the following methods:

- solving the linear programming relaxation (LIN) with the Mosek [Mos10] state-of-the-art interior-point solver ("LIN/IP"),
- solving one of the semidefinite programming relaxations with the Mosek interior-point solver (".../SDP")
- approximating the semidefinite programming relaxations via maximization of the dual of (SDP-A) with the L-BFGS [Noc80] quasi-Newton method (".../QN").

For a relaxation of a QAP instance let c^- be the lower bound, c^+ be the minimal cost of 100 projected solutions as discussed in Section 3.4 and c^* be the minimal cost of the instance. The relative error $\frac{c^+-c^*}{|c^*|}$ measures the quality of the solution. It is bounded by the relative optimality gap $\frac{c^+-c^-}{|c^*|}$, whose calculation is independent of the minimal cost c^*.

4.1. QAPLIB

QAPLIB [BKR97] is a collection of QAP instances of various authors. Despite being released in 1997 it is commonly used to benchmark QAP solvers, as the difficulty of QAPs increases quickly when their dimensions grow. Our first evaluation is on the QAPLIB instances of dimension 20 and less. Figure 2 shows the relative optimality gap and the relative error, and Figure 1c depicts the times the different algorithms used. The table in Figure 3 depicts the fraction of the instances, in which one method had smaller optimality gap/error than the other.

Solving problem instances depends on the convex relaxation as well as on the solver and this dependency on the solver makes a reliable practical evaluation of the relaxations difficult. For convex programs with strong duality, remedy comes from primal-dual solvers. They not only minimize the objective, but delimit it with lower and upper bounds. Once the bounds are sufficiently close the global optimum has reliably been found. The Mosek [Mos10] interior-point solver is such a primal-dual solver, which we therefore strive to use in our evaluation when possible.

On the downside interior-point solvers can be slow, so that not all instances can be solved with an interior-point solver in a reasonable time. For the relaxations (SDP-R3) and (TIGHT) solving with the Mosek interior-point solver even the smaller instances took hours, which we therefore did only on the instances "chr12a" to "chr15c".

When solving the SDP approximation (SDP-F) we limit the approximation introduced error to 1% of the (known) minimal cost by choosing γ accordingly. The reliability of the results then *only* depends on the minimization of a convex function on a convex domain with a quasi-Newton method. From our experiments we draw the following conclusions:

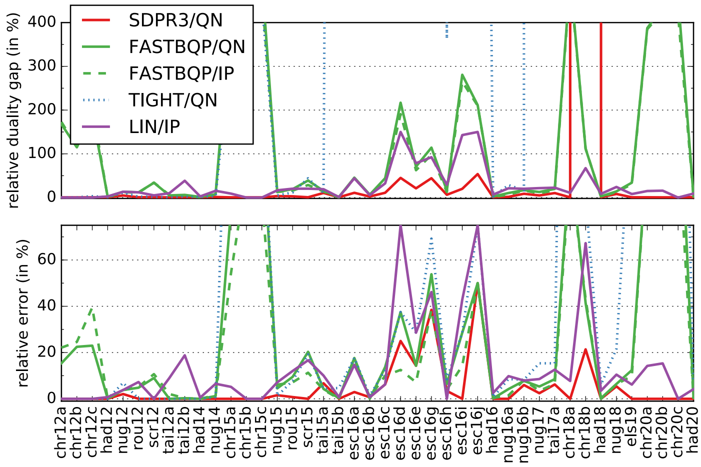

Figure 2: *Optimality gap and relative error on the QAPLIB instances.*

The quasi-Newton method and the interior-point solver yield similar results for the (FASTBQP) relaxation as well as for the (SDP-R3) relaxation on the instances "chr12a" to "chr15c" and for the (TIGHT) relaxation on the instances "chr12a" to "nug14". Thus in most cases the quasi-Newton approximation gave reliable estimations of the lower bound. Only when solving (TIGHT) on the instances "chr15a" to "chr15c" did the quasi-Newton method fail to converge and delivered a much smaller lower bound then the interior-point solver, whose lower bounds were much more comparable to the results of (SDP-R3). This is especially important as research hints that the relaxations (TIGHT) and (SDP-R3) are indeed *equal* ([DML17], proof of lemma (2) in the appendix).

After this note of caution we proceed with interpreting the results. (SDP-R3) (and possibly (TIGHT)) provides in nearly all cases the smallest relative error and the tightest bounds as expected from Theorem 4. Where this is not the case it might be due to missing convergence of the quasi-Newton solver and due to the randomization of the upper bound. (LIN) and (FASTBQP) have similar relative errors and optimality gaps, although in a few examples (FASTBQP) has a very large relative error and optimality gap.

(SDP-R3) is not only known for its good results but also for the long time interior-point solvers require solving it. For most instances (LIN) and (FASTBQP) are the fastest methods, which is clearly demonstrated in Figure 1c showing the solver time relative to (LIN/IP). Approximating (SDP-R3) and (TIGHT) by maximizing the dual of (SDP-A) with a quasi-Newton method is much faster than solving (SDP-F) with an interior-point solver. Indeed in nearly all cases the interior-point solver requires much more than 30 minutes. This is a striking difference, from seconds (LIN) to minutes (SDP-R3/QN) to hours (SDP-R3/IP) (note the logarithmic scale in Figure 1c).

The various SDP relaxations differ not only in their solutions but also in their timings. Typically adding constraints to a relaxation restricts the solution space and decreases the number of iterations,

	Optimality gap			Error		
	SDPR3	LIN	FAST	SDPR3	LIN	FAST
SDPR3	-	0.81	0.98	-	0.67	0.86
LIN	0.19	-	0.67	0.10	-	0.43
FAST	0.02	0.33	-	0.05	0.48	-

Figure 3: *Fraction of experiments, where methods have smaller optimality gap/error (FAST is FASTBQP, TIGHT was left out due to uncertainties of the quasi-Newton solver).*

but increases the time for each iteration. (TIGHT) has the most constraints and a feasible set at least as large as (SDP-R3) and accordingly takes the most time to solve. (FASTBQP) on the other hand has got only $\mathcal{O}(n^3)$ constraints and takes the least time to solve.

4.2. Shape matching

Graph and shape matching is an application of the QAP in computer graphics and vision [SRS07, KKBL15, VLR*17]. In the following we evaluate the performance of our relaxations and solvers to match shapes from the Tosca [BBK08] and the Shrec [GBP07] datasets.

On both surfaces, that are to be matched, we choose all points of extremal average geodesic distance in a geodesic neighborhood of $1/5$ the geodesic diameter, which are usually located at semantically meaningful locations. Then we iteratively add the geodesically farthest point until we have n points. Let d_{ij} and d'_{ij} be the pairwise geodesic distances on both shapes, let σ be the mean of the distances from each point to its closest neighbor. Then we define the geodesic distortion as:

$$\mathbf{A}^{\text{ISO}}_{pq,rs} = \exp\left(-\frac{(d_{pr} - d'_{qs})^2}{\sigma^2}\right) \tag{24}$$

Good assignments have low distortions and the minimizer of the QAP defined by $\mathbf{A}^{\text{ISO}}$ is often a good assignment of the points [LH05, KKBL15].

Figure 4 shows several example shapes with sampled and assigned points on isometric and near-isometric shapes. Figure 5 and Figure 6 depict the relative optimality gaps and the times required to solve a

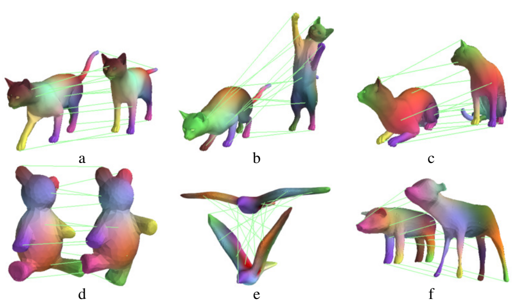

Figure 4: *Solving A^{ISO} on pairs of shapes sampled at 15 points. Corresponding points have random colors, which are then diffused over the shape. (b,e) match the intrinsic symmetry and is not a failure case. Points are sampled as described in the text and do not necessarily agree exactly, e.g. head in (b,d).*

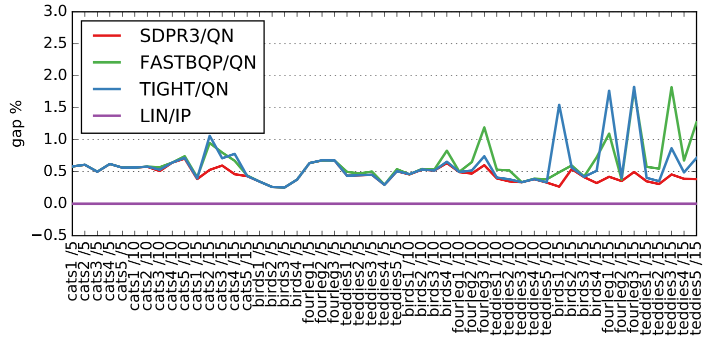

Figure 5: *Optimality gap after solving the QAP of Eq. (24) with several methods.*

series of test cases such as the examples in Figure 4 sampled with 5, 10 or 15 points.

In the previous evaluation of QAPLIB the instances led to varying optimality gaps and the optimality gaps can be seen as a measure of hardness of the QAP instance. Interestingly, the QAP instances from Eq. (24) lead to a very tight relaxation, i.e. a small optimality gap. (LIN) has an optimality gap of 0 in all cases and the optimality gap of the SDP relaxations stems from the approximation with (SDP-A) and can be further reduced by increasing γ. Therefore with the correct γ *all* relaxations result in the optimal solution.

Yet the methods differ greatly in the time required to solve. Solving (LIN) and possibly (FASTBQP) with an interior-point solver is the fastest method although fast approximations of (SDP-R3) and (FASTQAP) with (SDP-A) are only by a factor 2-4 slower. Approximating (TIGHT) with (SDP-A) or solving the SDP relaxation of (SDP-R3) with an interior-point solver is still one order of magnitude slower.

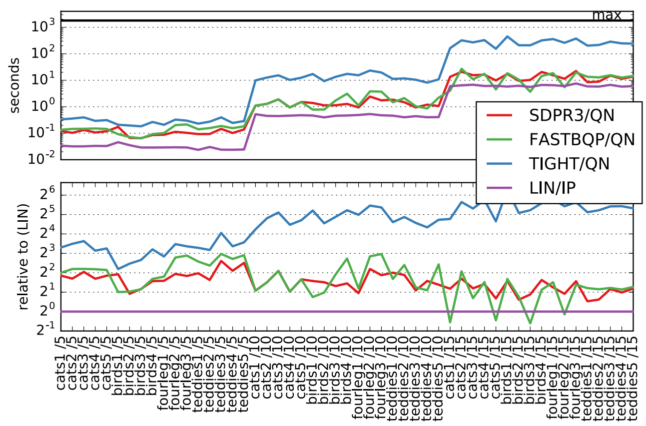

Figure 6: *Solver times for the different methods to solve the QAP of Eq. (24).*

4.2.1. Product Manifold Filter

The product manifold filter (PMF) [VLR*17] uses a few predefined correspondences to infer improved ones. Typically the method is applied iteratively until convergence. Let the current correspondences be encoded in a matrix $\mathbf{X}_i \in \mathbb{R}^{n\times n}$ where $(\mathbf{X}_i)_{pq} = 1$ if points p and q are assigned and 0 otherwise. Let σ be as above and let r and r' be measures of locality on both shapes. Then we calculate new correspondences ϕ_{i+1} by solving the Linear Assignment Problem:

$$\text{(PMF-LP)} \quad \phi_{i+1} = \arg\min_{\phi \in S_n} \; [\mathbf{X}^{\phi}]^T \; \mathbf{A}^{\text{PMF}} \; [\mathbf{X}_i]$$

$$\text{with} \quad \mathbf{A}^{\text{PMF}}_{pq,rs} = \begin{cases} 0 & d_{pr} < r \wedge d'_{qs} > 2r' \\ 0 & d'_{qs} < r' \wedge d_{pr} > 2r \\ \exp\left(-\frac{(d_{pr}+d'_{qs})^2}{\sigma^2}\right) & \text{otherwise} \end{cases}$$

Requiring only a few initial correspondences, this method was shown to compute correspondences of several hundred points. We show next that the above formulation can be interpreted as an iterative minimization of the following QAP:

$$\text{(PMF-QAP)} \qquad \phi^* = \arg\min_{\phi \in S_n} \; \frac{1}{2}[\mathbf{X}^{\phi}]^T \; \mathbf{A}^{\text{PMF}} \; [\mathbf{X}^{\phi}] \qquad (25)$$

First we relax both formulations onto the Birkhoff polytope Π^n:

$$\text{(PMF-LP')} \qquad \mathbf{X}_{i+1} = \arg\min_{\mathbf{X} \in \Pi^n} \; [\mathbf{X}]^T \; \mathbf{A}^{\text{PMF}} \; [\mathbf{X}_i] \qquad (26)$$

$$\text{(PMF-QAP')} \qquad \mathbf{X}^* = \arg\min_{\mathbf{X} \in \Pi^n} \; \frac{1}{2}[\mathbf{X}]^T \; \mathbf{A}^{\text{PMF}} \; [\mathbf{X}] \qquad (27)$$

Theorem 8. *The stationary points of (PMF-LP') are the local minima of (PMF-QAP'). If* $\mathbf{A}^{PMF}$ *is not positive definite, (PM-QAP') might have several local minima.*

Proof. $\mathbf{X}^*$ is a local minimum of $f(\mathbf{X}) = \frac{1}{2}[\mathbf{X}]^T \mathbf{A}^{\text{PMF}}[\mathbf{X}]$ if and only if $f(\mathbf{X}^*)$ grows in any direction d which does not leave the convex set Π^n. Such directions can be parameterized by $\mathbf{X} \in \Pi^n$ with $d = \mathbf{X} - \mathbf{X}^*$:

$$\begin{aligned} & \mathbf{X}^* \text{ is a local minimum of } f(\mathbf{X}) \\ \Leftrightarrow\; & \left((\nabla_{\mathbf{X}} f)|_{\mathbf{X}=\mathbf{X}^*}\right)^T [\mathbf{X} - \mathbf{X}^*] \geq 0 \quad \forall \mathbf{X} \in \Pi^n \\ \Leftrightarrow\; & [\mathbf{X}^*]^T \mathbf{A}^{\text{PMF}}[\mathbf{X}] \geq [\mathbf{X}^*]^T \mathbf{A}^{\text{PMF}}[\mathbf{X}^*] \quad \forall \mathbf{X} \in \Pi^n \\ \Leftrightarrow\; & \mathbf{X}^* = \arg\min_{\mathbf{X}} [\mathbf{X}^*]^T \mathbf{A}^{\text{PMF}}[\mathbf{X}] \end{aligned} \qquad \square$$

By interpreting PMF as a local minimization of a QAP energy we can remove the requirement of predefined correspondences. We first solve (PMF-QAP) over a small set of points ($n \approx 15$) and then refine the solution using (PMF-LP), which has only $\mathcal{O}(n^2)$ instead of $\mathcal{O}(n^4)$ variables and scales better. Some example applications can be seen in Figure 7, which shows the correspondences after (PMF-QAP) and after (PMF-LP).

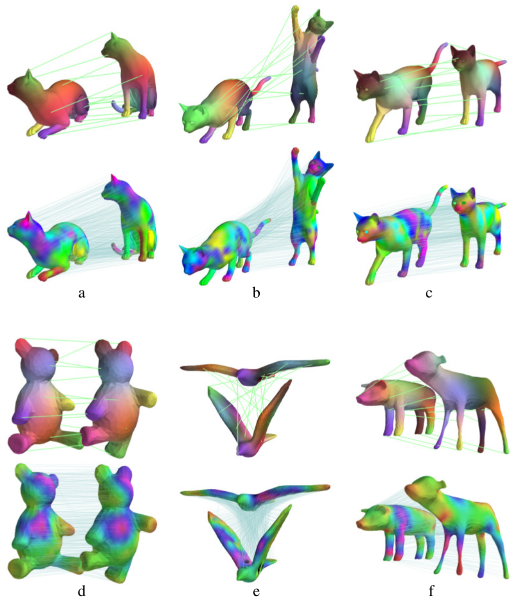

Figure 7: *Matching 15 points with (PMF-QAP) and refining the result with (PMF-LP). Maps are shown by mapping a random color signal. (a,b,d,e) matched correctly, although (b,e) match the intrinsic symmetry which happens in little less than half of the times; Failures: QAP twisted the feet in (c), finding the k best solutions with Branch&Bound could help; (PMF-LP) had problems converging on the shoulder of (f).*

5. Conclusion

We compared several methods to solve the quadratic assignment problem with a focus on their application to shape matching. Our results show that the formulation as a linear program (LIN) is often preferable to the more complicated SDP relaxations. At least for $n \leq 15$ it is nearly always faster. Its bounds are often tighter than the bounds of (FASTBQP). If required, (SDP-R3) yields lower bounds at least as large as the others. Furthermore, approximating (SDP-R3) or (TIGHT) using (SDP-A) [WSvdHT16] is always much faster than using state-of-the-art interior-points solvers. We showed how to use these insights to solve quadratic matching problems and how to utilize the results for shape matching. Furthermore our interpretation of (PMF) as a (QAP) allows to remove the requirement of predefined correspondences. Investigating the possibilities to iterate several low-cost solutions with a Branch and Bound approach might be interesting possibilities for future work.

References

[ADK13] AFLALO Y., DUBROVINA A., KIMMEL R.: Spectral generalized multi-dimensional scaling. *CoRR abs/1311.2187* (2013). 2

[AJ94] ADAMS W. P., JOHNSON T. A.: Improved linear programming-based lower bounds for the quadratic assignment problem. *DIMACS series in discrete mathematics and theoretical computer science 16* (1994), 43–75. 2, 3

[AM14] AURORA P. K., MEHTA S. K.: New facets of the qap-polytope. *arXiv preprint arXiv:1409.0667* (2014). 2, 3

[Ans03] ANSTREICHER K. M.: Recent advances in the solution of quadratic assignment problems. *Mathematical Programming 97*, 1 (2003), 27–42. 2

[BBK08] BRONSTEIN A. M., BRONSTEIN M. M., KIMMEL R.: *Numerical geometry of non-rigid shapes*. Springer Science & Business Media, 2008. 7

[BBM05] BERG A. C., BERG T. L., MALIK J.: Shape matching and object recognition using low distortion correspondences. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on* (2005), vol. 1, IEEE, pp. 26–33. 1

[BCPP98] BURKARD R. E., CELA E., PARDALOS P. M., PITSOULIS L. S.: The quadratic assignment problem. In *Handbook of combinatorial optimization*. Springer, 1998, pp. 1713–1809. 2

[BKR97] BURKARD R. E., KARISCH S. E., RENDL F.: Qaplib–a quadratic assignment problem library. *Journal of Global optimization 10*, 4 (1997), 391–403. 2, 6

[BMP02] BELONGIE S., MALIK J., PUZICHA J.: Shape matching and object recognition using shape contexts. *IEEE transactions on pattern analysis and machine intelligence 24*, 4 (2002), 509–522. 2

[BMP12] BAI L., MITCHELL J., PANG J.: *On QPCCs, QCQPs and copositive programs*. Tech. rep., Technical report, Rensselaer Polytechnic Institute, 2012. 2

[Bur12] BURER S.: Copositive programming. In *Handbook on semidefinite, conic and polynomial optimization*. Springer, 2012, pp. 201–218. 2

[BV06] BURER S., VANDENBUSSCHE D.: Solving lift-and-project relaxations of binary integer programs. *SIAM Journal on Optimization 16*, 3 (Jan 2006), 726–750. doi:10.1137/040609574. 2

[Cel13] CELA E.: *The quadratic assignment problem: theory and algorithms*, vol. 1. Springer Science & Business Media, 2013. 2

[CMC*09] CAETANO T. S., MCAULEY J. J., CHENG L., LE Q. V., SMOLA A. J.: Learning graph matching. *IEEE transactions on pattern analysis and machine intelligence 31*, 6 (2009), 1048–1058. 2

[dKST15] DE KLERK E., SOTIROV R., TRUETSCH U.: A new semidefinite programming relaxation for the quadratic assignment problem and its computational perspectives. *INFORMS Journal on Computing 27*, 2 (2015), 378–391. 2

[DML17] DYM N., MARON H., LIPMAN Y.: DS++: A flexible, scalable and provably tight relaxation for matching problems. *CoRR abs/1705.06148* (2017). URL: http://arxiv.org/abs/1705.06148. 6

[EKG13] EGOZI A., KELLER Y., GUTERMAN H.: A probabilistic approach to spectral graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence 35*, 1 (2013), 18–27. 2

[FS06] FUNKHOUSER T., SHILANE P.: Partial matching of 3d shapes with priority-driven search. In *ACM International Conference Proceeding Series* (2006), vol. 256, pp. 131–142. 1

[FY83] FRIEZE A., YADEGAR J.: On the quadratic assignment problem. *Discrete applied mathematics 5*, 1 (1983), 89–98. 2, 3

[GBP07] GIORGI D., BIASOTTI S., PARABOSCHI L.: Shrec: shape retrieval contest: Watertight models track. *Online]: http://watertight. ge. imati. cnr. it* (2007). 7

[Gil62] GILMORE P. C.: Optimal and suboptimal algorithms for the quadratic assignment problem. *Journal of the society for industrial and applied mathematics 10*, 2 (1962), 305–313. 2

[GW95] GOEMANS M. X., WILLIAMSON D. P.: Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM (JACM) 42*, 6 (1995), 1115–1145. 2

[HG98] HAHN P., GRANT T.: Lower bounds for the quadratic assignment problem based upon a dual formulation. *Operations Research 46*, 6 (1998), 912–922. 2

[JK96a] JÜNGER M., KAIBEL V.: A basic study of the qap-polytope. In *Institut Für Informatik, Universität zu Köln* (1996). 2, 3

[JK96b] JÜNGER M., KAIBEL V.: On the sqap-polytope. In *SIAM Journal of Optimization* (1996). 2

[JK01] JÜNGER M., KAIBEL V.: The qap-polytope and the star transformation. *Discrete Applied Mathematics 111*, 3 (2001), 283–306. 2, 3

[Kar95] KARISCH S. E.: *Nonlinear approaches for quadratic assignment and graph partition problems*. Inst. für Mathematik, Techn. Univ. Graz, 1995. 2

[KB57] KOOPMANS T. C., BECKMANN M.: Assignment problems and the location of economic activities. *Econometrica: journal of the Econometric Society* (1957), 53–76. 2

[KB78] KAUFMAN L., BROECKX F.: An algorithm for the quadratic assignment problem using bender's decomposition. *European Journal of Operational Research 2*, 3 (1978), 207–211. 2

[KCCE99] KARISCH S. E., CELA E., CLAUSEN J., ESPERSEN T.: A dual framework for lower bounds of the quadratic assignment problem based on linearization. *Computing 63*, 4 (1999), 351–403. 2

[KKBL15] KEZURER I., KOVALSKY S. Z., BASRI R., LIPMAN Y.: Tight relaxation of quadratic matching. In *Computer Graphics Forum* (2015), vol. 34, Wiley Online Library, pp. 115–128. 1, 2, 4, 5, 7

[Law63] LAWLER E. L.: The quadratic assignment problem. *Management science 9*, 4 (1963), 586–599. 2

[LdABN*07] LOIOLA E. M., DE ABREU N. M. M., BOAVENTURA-NETTO P. O., HAHN P., QUERIDO T.: A survey for the quadratic assignment problem. *European journal of operational research 176*, 2 (2007), 657–690. 2

[LH05] LEORDEANU M., HEBERT M.: A spectral technique for correspondence problems using pairwise constraints. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on* (2005), vol. 2, IEEE, pp. 1482–1489. 2, 7

[LMS*10] LUO Z.-Q., MA W.-K., SO A. M.-C., YE Y., ZHANG S.: Semidefinite relaxation of quadratic optimization problems. *IEEE Signal Processing Magazine 27*, 3 (2010), 20. 2, 5

[LS91] LOVÁSZ L., SCHRIJVER A.: Cones of matrices and set-functions and 0–1 optimization. *SIAM Journal on Optimization 1*, 2 (1991), 166–190. 2

[MD58] MENDELSOHN N., DULMAGE A.: The convex hull of sub-permutation matrices. *Rn 1* (1958), R2. 5

[Mos10] MOSEK A.: The mosek optimization software. *Online at http://www. mosek. com 54* (2010), 2–1. 6

[Noc80] NOCEDAL J.: Updating quasi-newton matrices with limited storage. *Mathematics of computation 35*, 151 (1980), 773–782. 6

[PR09] POVH J., RENDL F.: Copositive and semidefinite relaxations of the quadratic assignment problem. *Discrete Optimization 6*, 3 (2009), 231–241. 1, 2, 3

[PVZ15] PENA J., VERA J. C., ZULUAGA L. F.: Completely positive reformulations for polynomial optimization. *Mathematical Programming 151*, 2 (2015), 405–431. 2

[PZLT15] PENG J., ZHU T., LUO H., TOH K.-C.: Semi-definite programming relaxation of quadratic assignment problems based on nonredundant matrix splitting. *Computational Optimization and Applications 60*, 1 (2015), 171–198. 2

[RS06] RENDL F., SOTIROV R.: Bounds for the quadratic assignment problem using the bundle method. *Mathematical Programming 109*, 2-3 (Sep 2006), 505–524. doi:10.1007/s10107-006-0038-8. 2

[SG76] SAHNI S., GONZALEZ T.: P-complete approximation problems. *Journal of the ACM (JACM) 23*, 3 (1976), 555–565. 2

[SRS07] SCHELLEWALD C., ROTH S., SCHNÖRR C.: Evaluation of a convex relaxation to a quadratic assignment matching approach for relational object views. *Image and Vision Computing 25*, 8 (2007), 1301–1314. 2, 7

[VLR*17] VESTNER M., LITMAN R., RODOLA E., BRONSTEIN A., CREMERS D.: Product manifold filter: Non-rigid shape correspondence via kernel density estimation in the product space. *arXiv preprint arXiv:1701.00669* (2017). 1, 2, 7, 8

[WSVDH13] WANG P., SHEN C., VAN DEN HENGEL A.: A fast semidefinite approach to solving binary quadratic problems. In *Proceedings of* *the IEEE conference on computer vision and pattern recognition* (2013), pp. 1312–1319. 2, 4

[WSvdHT16] WANG P., SHEN C., VAN DEN HENGEL A., TORR P. H.: Large-scale binary quadratic optimization using semidefinite relaxation and applications. *IEEE transactions on pattern analysis and machine intelligence* (2016). 1, 2, 4, 8

[ZKRW98] ZHAO Q., KARISCH S. E., RENDL F., WOLKOWICZ H.: Semidefinite programming relaxations for the quadratic assignment problem. *Journal of Combinatorial Optimization 2*, 1 (1998), 71–109. 1, 2, 3