Quantum Permutation Synchronization

Tolga Birdal[1,⋆] Vladislav Golyanik[2,⋆] Christian Theobalt[2] Leonidas Guibas[1]

[1]Stanford University [2]Max Planck Institute for Informatics, SIC

Abstract

*We present **QuantumSync**, the first quantum algorithm for solving a synchronization problem in the context of computer vision. In particular, we focus on permutation synchronization which involves solving a non-convex optimization problem in discrete variables. We start by formulating synchronization into a quadratic unconstrained binary optimization problem (QUBO). While such formulation respects the binary nature of the problem, ensuring that the result is a set of permutations requires extra care. Hence, we: (i) show how to insert permutation constraints into a QUBO problem and (ii) solve the constrained QUBO problem on the current generation of the adiabatic quantum computers D-Wave. Thanks to the quantum annealing, we guarantee global optimality with high probability while sampling the energy landscape to yield confidence estimates. Our proof-of-concepts realization on the adiabatic D-Wave computer demonstrates that quantum machines offer a promising way to solve the prevalent yet difficult synchronization problems.*

1. Introduction

Computer vision literature accommodates a myriad of efficient and effective methods for processing rich information from widely available 2D and 3D cameras. Many algorithms at our disposal excel at processing single frames or sequence of images such as videos [83, 34, 13]. Oftentimes, a scene can be observed by multiple cameras, from different, usually unknown viewpoints. To this date it remains an open question how to *consistently* combine the multi-view cues acquired without respecting any particular order [44].

Synchronization [92, 42, 93] is one of the proposed solutions to the aforementioned problem. On an abstract note, it involves distributing the discrepancies over the graph connecting multiple viewpoints such that the estimates are consistent across all considered nodes. To this end, synchronization simultaneously *averages* the pairwise local information into a global one [49, 55, 4, 12]. This procedure is a fundamental piece of most state-of-the-art multi-view reconstruction and multi-shape analysis pipelines [86, 23, 25] because it heavy-lifts the global constraint satisfaction while respecting the geometry of the parameters. In fact, most of the multiview-consistent inference problems can be expressed as some form of a synchronization [108, 15].

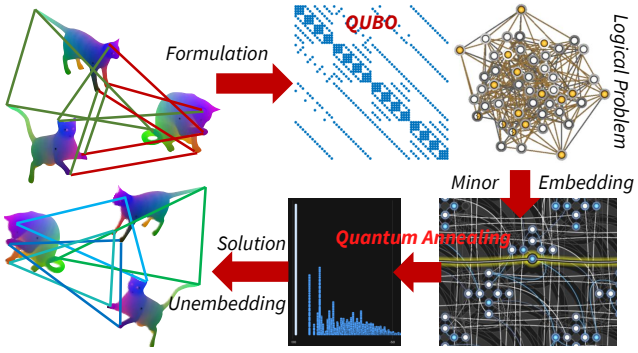

Figure 1. Overview of **QuantumSync**. QuantumSync formulates permutation synchronization as a QUBO and embeds its logical instance on a quantum computer. After running multiple anneals, it selects the lowest energy solution as the global optimum.

In this paper, our focus is *permutation synchronization*, where the edges of the graph are labeled by permutation matrices denoting the correspondences either between two 2D images or two 3D shapes. Specifically, we seek to find an absolute ordering for each point of each frame which sorts all the corresponding points into the same bin. Unfortunately, this problem by definition involves a combinatorial non-convex optimization, for which attaining the global minimum is intractable under standard formulations targeting classical von Neumann computers. This difficulty have encouraged scholars to seek *continuous relaxations* for which either a good local optimum [14, 16] or a closed form solution [6, 58, 5] could be found. The solution to this approximately equivalent relaxed problem should then be rounded back to the permutation matrices to report a valid result. Ideally, we would like to avoid such approximations and work with the original set of constraints at hand. This is exactly what we propose to do in this work. Because on a classical computer we cannot speak of a global optimality guarantee of our discrete problem, we turn our attention to a new family of processors, *i.e., quantum computers*.

⋆ both authors contributed equally to this work

A quantum computer is a computing machine which takes advantage of quantum effects such as *quantum superposition*, *entanglement*, *tunnelling* and *contextuality* to solve problems notoriously difficult (*i.e.,* $\mathcal{NP}$-hard) on a classical computer. Numerous quantum algorithms have demonstrated improved computational complexity finally reaching the desired *supremacy* [7]. Currently, we see the first practical uses of quantum computers thanks to the programmable quantum processing units (QPU) available to the research community such as D-Wave [2] or IBM [1]. As this new generation of computers offers a completely different computing paradigm, directly porting classical problem formulations is far from being trivial. In fact, oftentimes we are required to revise the problem at hand altogether [50].

This paper shows how to formulate the classical permutation synchronization in terms of a quadratic unconstrained binary optimization (QUBO) that is quantum computer friendly. QUBO optimizes for *binary* variables and not permutations, and, hence, we are required to ensure that permutation constraints are respected during optimization. To this end, we turn all constraints into linear ones and incorporate them into QUBO. Finally, we embed our problem to a real quantum computer and show that it is highly likely to achieve the lowest energy solution, *the global optimum* (see Fig. 1). Our **contributions** are as follows:

1. The first, to the best of our knowledge, formulation of the classical synchronization in a form consumable by an adiabatic quantum computer (AQC);
2. We show how to introduce permutations as linear constraints into the QUBO problem;
3. We numerically verify the validity of our formulation in simulated experiments as well as on a real AQC (for the first time, on D-Wave Advantage 1.1);
4. We perform extensive ablation studies giving insights into this new way of computing.

We obtain highly probable global optima up to either of: (i) eight views and three points per view, (ii) seven views with four points, or (iii) five points with three views. We experiment for the first time on an AQC with $5k$ qubits and perform extensive evaluations and comparisons to classical methods. Our approach can also be classified as the first method for quantum matching of multiple point sets. While we are the first to implement synchronization on quantum hardware, our evaluations and tests are proof of concepts as truly practical quantum computing is still a leap away.

2. Related Work

Synchronization. The art of consistently recovering absolute quantities from a collection of ratios, *synchronization*, is now the de-facto choice when it comes to bringing functions on image/shape collections into unison [86, 23, 25]. The problem is now very well studied, enjoying a rich set of algorithms. Primarily, there exists a plethora of works on group structures arising in different applications [49, 48, 16, 5, 4, 57, 55, 4, 101, 28, 97, 99, 6, 9]. Some of the proposed solutions are closed form [6, 5, 73, 4] or minimize robust losses [54, 27]. Others address certifiability [84] and global optimality [21]. Bayesian treatment or uncertainty quantification is also considered [98, 14, 12, 16] as well as low rank matrix factorizations [10]. Recent algorithms tend to incorporate synchronization into deep learning [59, 43]. This work concerns with synchronizing correspondence sets, otherwise known as *permutation synchronization* (PS) [82]. This sub-field also attracted a descent amount of attention: low-rank formulations [105, 102], convex programming [56], multi-graph matching[87], distributed optimization [56] or Riemannian optimization [16].

All of these approaches try to cope with the intrinsic non-convexity of the synchronization problem one way or another. Unfortunately, solving our problem on a classical computer is notoriously difficult. To the best of our knowledge, we are the firsts to address this problem through the lens of a new paradigm, *adiabatic quantum computing*.

Quantum computing. Since its motivation in the 1980s [72, 39], quantum computing has become an active research area, both from the hardware [30, 62, 36, 103, 74, 112, 66, 69, 67] and algorithmic side [90, 91, 17, 50, 38, 52, 100, 53, 31, 3, 71, 88, 76, 37, 109, 95]. Quantum methods offering speedup compared to the classical counterparts have been demonstrated for domains such as applied number theory [90, 17, 100, 37], linear algebra [52, 53, 68, 96, 95], machine learning [3, 71, 109] and simulation of physical systems [107, 80, 11, 63, 60], among others.

Quantum annealing (QA) [61, 22] and the adiabatic quantum evolution algorithm by Farhi *et al.* [38] have triggered the development of adiabatic quantum computers (AQC). In the last decade, the technology has matured and became accessible remotely for test and research purposes with the help of D-Wave [32]. A recent benchmarking of D-Wave AQC [33] has shown that for energy landscapes with large and tall barriers, quantum annealing can achieve speed-ups of up to eight orders of magnitude compared to simulated annealing [64] running on a single core. AQC has also been successfully applied to traffic flow optimization while outperforming classical methods [76].

Quantum computer vision. Several quantum methods for computer vision problems have been proposed in the literature including algorithms for image recognition [77], classification [20, 79] and facial feature learning by low-rank matrix factorization [81]. Recently, D-Wave has been applied to redundant object removal in object detection [70]. While a QUBO formulation of non-maximum suppression was already known in the literature [85], it has been improved in [70] for D-Wave 2X with solutions outperforming several classical methods. Quantum approach for point

set alignment [45, 46] solves a related problem to ours. It approximates rotation matrices in the affine space by basis elements which can be summed up according to the measured bitstring. However, this formulation cannot be easily extended to the multi-view case and does not support permutation constraints. In contrast, our method solves a multi-way matching problem, and we successfully deploy it on a real AQC Advantage 1.1. Our use of assignment constraints is also new. Concurrently to us, a similar form of permutation matrix constraints was proposed by Benkner *et al*. [89] for matching two graphs on AQC. Several previous works formulate similar penalties in different ways. Stollenwerk *et al*. [94] address the flight assignment problem on AQC. Formulation of the graph isomorphism problem can include permutations [111, 41]. In [111], an individual variable for every vertex pair of the same degree in a graph is allocated, which subsequently leads to a QUBO. Permutations can also be converted to a table with binary entries added as a penalty term to the target Hamiltonian [41].

3. Preliminaries and Technical Background

3.1. Synchronization

We model the multi-view configuration as a connected undirected graph $\mathcal{G} = (\mathcal{V} = \{1, 2, \dots, n\}, \mathcal{E} \subset [n] \times [n])$ where $|\mathcal{E}| = m$ and if $(i, j) \in \mathcal{E}$ then $(j, i) \in \mathcal{E}$. Each vertex $v_i \in \mathcal{V}$ (*e.g.,* image) is associated a domain $\mathcal{D}_i$ (*e.g.,* ordered points). Each edge $(i, j) \in \mathcal{E}$ is labeled with a function $f_{i,j} : \mathcal{D}_i \mapsto \mathcal{D}_j$ (*e.g.,* correspondence). We will refer to the edge-related entities as *relative* and node- (vertex-) related entities as *absolute*. Thus, we aptly call $\{f_{i,j}\}_{i,j}$ as *relative maps*. In this section, we define and explain the necessary notions following Fig. 2. The proofs of the theorems given in this section can be found in our supplementary material.

Definition 1 (Path, Cycle and Null Cycle)**.** *We define a* ***path*** *to be the ordered, unique index sequence* $p = \{(i_1, i_2), (i_2, i_3), \cdots, (i_{n-1}, i_n)\} \in \mathcal{P}$ *along* $\mathcal{G}$ *connecting* v_{i_1} *to* v_{i_n}*. It is called a* ***cycle****, if additionally the path traces back to the starting node. Finally, following [4], we denote a cycle* $c \in \mathcal{C}$ *to be a* ***null-cycle*** *of* $\mathcal{G}$ *if the composition of functions along* c *leads to the identity transformation:*

$$f_c = f_{1,2} \circ f_{2,3} \cdots \circ f_{(n-1),n} \circ f_{n,1} = f_{\varnothing}, \qquad (1)$$

where $f_{\varnothing}(\mathbf{x}) = \mathbf{x}, \forall \mathbf{x} \in \mathcal{D}_1$ *is the identity map and* f_c *denotes the composite function. Intuitively,* c *is a non-empty path in which the only repeated vertices are the first and last vertices. Note that the direction of the action of* $\circ$ *matters (e.g.,* $f_{1,2} \circ f_{2,3} \neq f_{2,3} \circ f_{1,2}$*) and is up to the convention.*

Definition 2 (k-cycle)**.** *We refer to a function mapping a vertex to itself as the* ***1-cycle****:* $f_{i,i} = f_{\varnothing}$*. Similarly, a 2-cycle would be* $f_c = f_{i,j} \circ f_{j,i}$ *and so on [78, 58].*

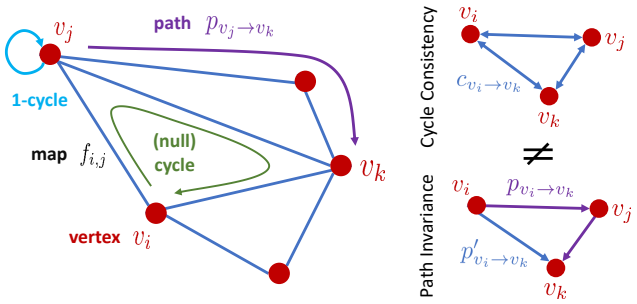

Figure 2. **(left)** Notation and illustration of the general setting we consider, **(right)** Cycle consistency *vs* path invariance.

Definition 3 (Cycle Consistency)**.** *We call the graph* $\mathcal{G}$ *to be cycle-consistent on* $\mathcal{C}$ *if* $f_c = f_{\varnothing}\ \forall c \in \mathcal{C}$*, where* $\mathcal{C}$ *is the set of all cycles [51].*

The notion of cycle consistency for *directed* graphs is known as *path invariance* [108] and differs from cycle consistency as shown in Fig. 2. In this paper, we further assume the maps belong to the *general linear group* and are *isomorphisms*, *i.e.,* $f_{ji} = f_{ij}^{-1}$.

Remark 1. *Depending on the graph topology, the number of cycles may be exponential in the number of vertices. Hence, naively ensuring the consistency of large graphs according to Dfn. 3 quickly becomes intractable. Algorithms such as Guibas et al. [51] aim to satisfy the consistency of a subset of cycles* $\bar{\mathcal{C}} \subset \mathcal{C}$ *(bases), where enforcing consistency along these cycles induces consistency along all cycles of the input graph* $\mathcal{G}$*. However, as efficient selection of these* cycle-consistency bases *[106] is a problem under investigation, we instead use the available group structure:*

Theorem 1 (Cycle Consistency by Construction)**.** *A* ***consistent vertex labeling*** $\{f_i : \mathcal{V} \mapsto \mathcal{D}_i\}_i$ *where* $(\mathcal{D}, \circ)$ *forms a* ***group*** *with operation* $\circ$*, can be constructed by satisfying the following constraint for all* i *and* j*:*

$$f_{i,j} = f_i \circ f_j^{-1}. \qquad (2)$$

Hence, Eq (2) is called the *cycle consistency constraint*.

Definition 4 (Synchronization)**.** *Synchronization is the procedure of finding a consistent labeling of* $\mathcal{G}$ *given a collection of ratios* $\{f_{i,j}\}_{i,j}$ *ensuring the cycle consistency of* $\mathcal{G}$*:*

$$\underset{\{f_k\}_k}{\arg\min} \sum_{(i,j)\in\mathcal{E}} d(\hat{f}_{i,j}, f_{i,j}). \qquad (3)$$

$\hat{f}_{i,j} = f_i \circ f_j^{-1}$ *are the estimated ratios and* $d(\cdot)$ *is a group-specific distance metric.*

Remark 2. *A closer look to the problem reveals that it is* ***non-convex*** *due to the composition, but* ***convex when*** *the vertices* f_j *are fixed during optimization of* f_i*. In fact, if*

f_i is considered to be fixed, this problem resembles an averaging under the metric $d(\cdot)$. As for different i we have different averages to compute, synchronization is often referred as ***multiple averaging*** *[40, 55, 26, 54].*

Theorem 2 (Gauge Freedom)**.** *The problem in Dfn. 4 is subject to a freedom in the choice of the reference or the* gauge *[8, 26]. In other words, the solution set to Eq (3) can be transformed arbitrarily by a common f_g i.e. $f_i \leftarrow f_i \circ f_g$ while still satisfying the consistency constraint.*

In practice, a gauge is fixed by setting one of the vertex labels to identity: $f_1 = f_\varnothing$.

Definition 5 (Permutation Matrix)**.** *A permutation matrix is defined as a sparse, square binary matrix, where each column or row contains only a single non-zero entry:*

$$\mathcal{P}_n := \{\mathbf{P} \in \{0,1\}^{n\times n} : \mathbf{P}\mathbf{1}_n = \mathbf{1}_n \,,\, \mathbf{1}_n^\top \mathbf{P} = \mathbf{1}_n^\top\}. \quad (4)$$

where $\mathbf{1}_n$ denotes a n-dimensional ones vector. Every $\mathbf{P} \in \mathcal{P}_n$ is a total permutation matrix and $P_{ij} = 1$ implies that point i is mapped to element j. Note, $\mathbf{P}^\top = \mathbf{P}^{-1}$. We also denote the ***product manifold*** *of m permutations as $\mathcal{P}_n^m$.*

Definition 6 (Relative Permutation)**.** *We define a permutation matrix to be a* ***relative map*** *if it is the ratio (or difference) of two group elements $(i \to j)$: $\mathbf{P}_{ij} = \mathbf{P}_i \mathbf{P}_j^\top$.*

Definition 7 (Permutation Synchronization Problem)**.** *For a redundant set of measures of ratios $\{\mathbf{P}_{ij}\}$, the permutation synchronization [82] seeks to recover $\{\mathbf{P}_i\}$ for $i = 1, \ldots, N$ such that Eq (2) is satisfied: $\mathbf{P}_{ij} = \mathbf{P}_i \mathbf{P}_j^{-1}$.*

If the input data is noise-corrupted, this *consistency* will not hold and to recover the *absolute permutations* $\{\mathbf{P}_i\}$, some form of a *consistency error* is minimized.

3.2. Adiabatic Quantum Computation

Contemporary AQC can solve QUBO over a set of pseudo-Boolean functions of the following form:

$$\arg\min_{\mathbf{x}\in\mathcal{B}^n} \mathbf{x}^\top \mathbf{Q}\mathbf{x} + \mathbf{s}^\top \mathbf{x}, \quad (5)$$

where $\mathcal{B}^n$ denotes the set of binary vectors of length n, $\mathbf{Q} \in \mathbb{R}^{n\times n}$ is a real symmetric matrix and $\mathbf{s}$ is a real n-dimensional vector. (5) is a frequent problem that is known to be $\mathcal{NP}$-hard on a classical computer. AQA operates with *qubits* obeying the laws of quantum mechanics. In contrast to a classical bit, a qubit $|\phi\rangle$ can continuously transition between the states $|0\rangle$ and $|1\rangle$ (the equivalents of classical states 0 and 1) fulfilling the equation $|\phi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle$, with probability amplitudes satisfying $|\alpha|^2 + |\beta|^2 = 1$.

AQA algorithms for problems in a non-QUBO form first have to *encode the problem* as QUBO (5), which defines the former in terms of *logical* qubits and weights ($\mathbf{s}$ and $\mathbf{Q}$) between them. The logical problem is then *minor-embedded* to the AQA's physical qubits graph, with methods such as [24]. AQA interprets $\mathbf{s}$ and $\mathbf{Q}$ as qubit biases and couplings, respectively, and converts them to local magnetic fields on a QPU imposed on the qubits during *anneallings*, *i.e.,* free evolutions of the quantum-mechanical computing system. The search of the optimal $\mathbf{x}$ is performed by optimizing over the *hidden*[1] states of n qubits and consequently *unembedding* and *measuring* them as classical bitstrings. The latter are then passed to the *solution interpretation* step, which decodes them in the context of the original problem.

Before annealing, all n qubits are initialized in ground states of an initial *Hamiltonian* $\mathcal{H}_I$ that is easy to achieve as a superposition with equal probabilities of measuring $|0\rangle$ or $|1\rangle$ for every qubit. A Hamiltonian is an operator defining the energy spectrum of the system and— interpreted for computational problems— the space of all possible solutions. AQA performs a series of annealings, during which $\mathcal{H}_I$ continuously alters towards the problem Hamiltonian $\mathcal{H}_P$ under the influence of the local magnetic fields. This *instantaneous* Hamiltonian $\mathcal{H}$ can be expressed as:

$$\mathcal{H}(\tau) = [1-\tau]\,\mathcal{H}_I + \tau\,\mathcal{H}_P, \quad (6)$$

with the local time variable $\tau \in [0;1]$ transitioning from 0 to 1 during the annealing time t (*e.g.,* 20 μs). According to the adiabatic theorem of quantum mechanics [19], the system will likely remain in the ground state of $\mathcal{H}_P$ by the end of the anneal, assuming a *sufficiently long* t and despite a highly non-convex energy landscape of the problem. This is due to the quantum effects of superposition, entanglement and tunnelling. *Superposition* enables the optimization to be performed on all possible qubit states simultaneously; it allows to operate on a 2^n-dimensional state space spanned by n qubits. During quantum computation, *entangled* states are created, *i.e.,* the states of multiple qubits which cannot be described independently from each other. *Tunnelling* enables transition through the barriers (in geometric interpretation). For a more comprehensive overview of the AQC foundations, see [61, 38, 2] as well as Secs. 2 and 3 of [45].

4. Quantum Synchronization

Suppose a multi-view configuration where we are given n points in each of the m views. Points in view i relate to the ones in view j via a permutation $\mathbf{P}_{ij}$ (see Sec. 3.1). $\mathbf{P}_{ij}$ is usually obtained independently for each pair in the edge set $\mathcal{E}$ and hence is *noisy*. Following [16], we see the permutation synchronization as a probabilistic inference problem, where we will be interested in the following quantities:

1. Maximum a-posteriori (MAP):

$$\mathbf{X}^\star = \arg\max_{\mathbf{X}\in\mathcal{P}_n^m} \log p(\mathbf{X}|\mathbf{P}) \quad (7)$$

[1] in the sense that α, β cannot be revealed (measurement postulate) and non-destructive copying of $|\phi\rangle$ is not possible (no-cloning theorem [104])

where $\log p(\mathbf{X}|\mathbf{P}) =^{+} -\beta \sum_{(i,j)\in\mathcal{E}} \|\mathbf{P}_{ij} - \mathbf{X}_i\mathbf{X}_j^\top\|_{\mathrm{F}}^2$, and $=^{+}$ denotes equality up to an additive constant.

2. The full posterior distribution: $p(\mathbf{X}|\mathbf{P}) \propto p(\mathbf{P}, \mathbf{X})$.

Here, $\mathbf{X}^\star$ denotes the entirety of the sought permutations, and, similarly, $\mathbf{P}$ is the collection of all specified pairwise permutations. The MAP estimate is often easier to obtain and useful in practice. On the other hand, samples from the full posterior can provide important additional information, such as *uncertainty*. Not surprisingly, the latter is a much harder task, especially considering the discrete nature of our problem. Each AQC algorithm includes problem encoding, minor embedding, sampling and solution interpretation, as described in Sec. 3.2. In what follows, we will describe the problem encoding and the preparation of $\mathbf{Q}$ in Eq (5).

4.1. Permutation Synchronization as QUBO

We first re-write the synchronization loss in Eq (7) as a quadratic assignment problem (QAP) that is more friendly for adiabatic optimization, and later insert the permutations as linear constraints into the formulation.

Proposition 1. *Permutation synchronization under the Frobenius norm can be written in terms of a QUBO:*

$$\underset{\{\mathbf{X}_i\in\mathcal{P}_n\}}{\arg\min} \sum_{(i,j)\in\mathcal{E}} \|\mathbf{P}_{ij} - \mathbf{X}_i\mathbf{X}_j^\top\|_{\mathrm{F}}^2 = \underset{\{\mathbf{X}_i\in\mathcal{P}_n\}}{\arg\min}\ \mathbf{x}^\top\mathbf{Q}'\mathbf{x}.$$

Here, $\mathbf{x} = [\cdots \mathbf{x}_i^\top \cdots]^\top$ *and* $\mathbf{x}_i = \mathrm{vec}(\mathbf{X}_i)$ *where* $\mathrm{vec}(\cdot)$ *acts as a vectorizer.* $\mathbf{Q}'$ *is then a matrix of the form:*

$$\mathbf{Q}' = -\begin{bmatrix} \mathbf{I}\otimes\mathbf{P}_{11} & \mathbf{I}\otimes\mathbf{P}_{12} & \cdots & \mathbf{I}\otimes\mathbf{P}_{1m} \\ \mathbf{I}\otimes\mathbf{P}_{21} & \mathbf{I}\otimes\mathbf{P}_{22} & \cdots & \mathbf{I}\otimes\mathbf{P}_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{I}\otimes\mathbf{P}_{m1} & \mathbf{I}\otimes\mathbf{P}_{m2} & \cdots & \mathbf{I}\otimes\mathbf{P}_{mm} \end{bmatrix}. \quad (8)$$

Proof. The steps are intuitive to follow and an expanded proof is included in the supplementary material:

$$\begin{aligned} \mathbf{X}^\star &= \underset{\mathbf{X}\in\mathcal{P}_n^m}{\arg\min} \sum_{(i,j)\in\mathcal{E}} \|\mathbf{P}_{ij} - \mathbf{X}_i\mathbf{X}_j^\top\|_{\mathrm{F}}^2 && (9)\\ &= \underset{\mathbf{X}\in\mathcal{P}_n^m}{\arg\min}\ 2N^2n - 2\sum_{(i,j)\in\mathcal{E}} \mathrm{tr}(\mathbf{X}_j\mathbf{X}_i^\top\mathbf{P}_{ij}) && (10)\\ &= \underset{\mathbf{X}\in\mathcal{P}_n^m}{\arg\min} -\sum_{(i,j)\in\mathcal{E}} \mathrm{vec}(\mathbf{X}_i)^\top(\mathbf{I}\otimes\mathbf{P}_{ij})\mathrm{vec}(\mathbf{X}_j) \\ &= \underset{\mathbf{X}\in\mathcal{P}_n^m}{\arg\min}\ \mathbf{x}^\top\mathbf{Q}'\mathbf{x}. && (11) \end{aligned}$$

The Hessian of this problem is given by $\mathbf{Q}'$ itself. Hence, the problem is only convex and solvable by algorithms such as *interior-point* or *trust-region* methods when $\mathbf{Q}'$ is *positive definite*. The indefinite problems can be solved via *active-set* methods if the variables are relaxed to the set of reals. For discrete variables and when $\mathbf{Q}'$ is not positive definite, this is a variant of $\mathcal{NP}$-hard *integer quadratic problem*. We will instead show how to use the recent quantum computers to obtain the global minimum.

Formulating the QUBO synchronization. The problem in Eq (11) has an added difficulty of being a discrete combinatorial optimization problem over the *product manifold of permutations*. Replacing permutations with different choices of matrices lead to different relaxations, for example: (i) positivity allows for semi-definite programming [58], (ii) orthonormality allows for spectral solutions [73], and (iii) doubly stochastic relaxation can allow for Riemannian optimization [16]. However, all of these methods have to be followed by a projection step onto the discrete Permutohedron, often cast as an assignment problem and solved via the celebrated Hungarian algorithm. QUBO enables us to solve this problem *without continuous relaxations* in a globally optimal manner by rephrasing Eq (11) in terms of binary variables $\mathcal{B}$ at our disposal:

$$\underset{\mathbf{x}\in\mathcal{B}}{\arg\min}\ \mathbf{x}^\top\mathbf{Q}'\mathbf{x}. \quad (12)$$

Permutations as linear constraints. The binary variables $\mathbf{x}\in\mathcal{B}$ are a superset of the product-permutations. In other words, the solution to Eq (11) need not result in permutations once the matrices corresponding to each node are extracted. We propose to encourage the solution towards a set of permutations by introducing linear constraints $\mathbf{A}\mathbf{x} = \mathbf{b}$ such that the optimization adheres to the definition of a permutation: rows and columns sum to one as in Eq (4). Given $\mathbf{x}_i = \mathrm{vec}(\mathbf{X}_i)$, this amounts to having $\mathbf{b}_i = \mathbf{1}$ and

$$\mathbf{A}_i = \begin{bmatrix} \mathbf{I}\otimes\mathbf{1}^\top \\ \mathbf{1}^\top\otimes\mathbf{I} \end{bmatrix}. \quad (13)$$

Put simply, the matrix $\mathbf{A}_i$ is assembled as follows: in row j with $1 \le j \le n$, the ones are placed in columns $(j-1)\cdot n + 1$ to $(j)\cdot n$. In a row j with $j > n$, ones will be placed at $(j-n) + p\cdot n$ for $p \in \{0, ..., n-1\}$. To enforce the permutation-ness of all the individual $\mathbf{x}_i$ that make up $\mathbf{x}\in\mathbb{R}^{n^2\times m}$, we construct a $n^2 \times 2n$ block-diagonal matrix $\mathbf{A} = \mathrm{diag}(\mathbf{A}_1, \mathbf{A}_2, \ldots, \mathbf{A}_m)$.

Introducing linear constraints into QUBO. We now extend our formulation by introducing the equality constraints $\mathbf{A}\mathbf{x} = \mathbf{b}$ into the optimization.

Proposition 2. *The constrained minimization:*

$$\underset{\mathbf{x}\in\mathcal{B}}{\arg\min}\ \mathbf{x}^\top\mathbf{Q}'\mathbf{x} \quad \textit{s.t.} \quad \mathbf{A}\mathbf{x} = \mathbf{b} \quad (14)$$

can be turned into an (unconstrained) QUBO

$$\underset{\mathbf{x}\in\mathcal{B}}{\arg\min}\ \mathbf{x}^\top\mathbf{Q}\mathbf{x} + \mathbf{s}^\top\mathbf{x}, \quad (15)$$

where $\mathbf{Q} = \mathbf{Q}' + \lambda\mathbf{A}^\top\mathbf{A}$ *and* $\mathbf{s} = -2\lambda\mathbf{A}^\top\mathbf{b}$.

Table 1. Evaluations on Willow Dataset.

	Car	Duck	Motorbike	Winebottle	Average
Exhaustive	**0.84 ± 0.104**	**0.91 ± 0.115**	**0.82 ± 0.10**	**0.95 ± 0.096**	**0.88 ± 0.104**
EIG	0.81 ± 0.083	0.86 ± 0.102	0.77 ± 0.059	0.87 ± 0.107	0.83 ± 0.088
ALS	**0.84 ± 0.095**	0.90 ± 0.102	0.81 ± 0.078	0.94 ± 0.092	0.87 ± 0.092
LIFT	**0.84 ± 0.102**	0.90 ± 0.103	0.81 ± 0.078	0.94 ± 0.092	0.87 ± 0.094
Birkhoff	**0.84 ± 0.094**	0.90 ± 0.107	0.81 ± 0.079	0.94 ± 0.093	0.87 ± 0.093
D-Wave(Ours)	**0.84 ± 0.104**	0.90 ± 0.104	0.81 ± 0.080	**0.93 ± 0.095**	0.87 ± 0.096

Both $\mathbf{Q}'$ and $\mathbf{Q}$ are sparse matrices. We provide their sparsity patterns and the proof of Prop. 2 in our supplement. We finally map this modified QUBO $(\mathbf{Q}, \mathbf{s})$ onto D-Wave.

Encoding, Sampling and Interpretation. We fix the gauge by letting the first matrix to be identity: $\mathbf{X}_1 = \mathbf{I}$. Once *QuantumSync* terminates, the measured bitstring can be directly interpreted as the solution permutations after reordering into $m-1$ matrices of dimension $n \times n$. To explain the posterior landscape, we propose to sample from many low-energy states using the same quantum annealing. Further details are presented in the supplement.

5. Experiments and Evaluations

Real dataset. To showcase that quantum computers offer a promising way to solve the challenging multi-view matching problems, we extract four categories (*duck*, *car*, *winebottle*, *motorbike*) of Willow Object Classes [29] composed of 40 RGB images each, acquired *in the wild* (see Fig. 3). The images suffer from significant pose, lighting and environment variation. Hence, ten keypoints are manually annotated on each image. We use the first four of these keypoints and create 35 small problems for each category by creating a fully connected graph composed of all four consecutive frames. We follow [102] and extract local features from a set of 227×227 patches centered around the annotated landmarks, using Alexnet [65] pretrained on ImageNet [35]. The feature map responses of *Conv4* and *Conv5* layers are then matched by the Hungarian algorithm [75] to initialize the synchronization. As the data is manually annotated, the ground-truth relative maps are known.

Synthetic dataset. For a controlled evaluation of our method, similar to [16], we generate synthetic datasets composed of graphs with $m = |\mathcal{V}|$ nodes and $|\mathcal{E}| = m(m-1)$ edges. At each node i we generate n points mappable by a permutation $\mathbf{P}_{ij}$ to n other points at node j. Hence, all the permutation matrices $\mathbf{P}_i$ or $\mathbf{P}_{ij}$ are total *i.e.,* of size $n \times n$. While the graph is by default fully connected, in certain experiments we randomly drop certain edges to have a completeness C where $0.5 < C < 1$. For instance, $C = 0.75$ means that only 75% of the edges are actively present. We optionally perturb the relative permutations by swapping a percentage of the rows and columns. We call this the *swap ratio* and denote it as σ, where $0 \leq \sigma \leq 0.25$.

Evaluation methodology. We implement our algorithm on D-Wave Advantage 1.1 and compare it against the state-of-the-art methods MatchEIG [73], MatchALS [110], MatchLift [58], MatchBirkhoff [16] as well as to the *exhaustive* solution obtained by enumerating all possible permutations. In all of our evaluations we report the number of bits correctly detected and call this metric *accuracy*. For synthetic evaluations involving noise, we generate seven random problems with the same σ and average the results.

Figure 3. A random *car* example from Willow Object Classes [29].

5.1. Evaluations on D-Wave Advantage

In this section, we describe our experiments on D-Wave Advantage system 1.1, which is an AQC with 5436 qubits arranged on a graph of cells with eight qubits each. Every qubit operates under $\approx 15.8mK$ temperature and is connected to 15 other qubits from the same or other cells which enables compact minor embeddings, *i.e.,* mappings of target QUBOs to the processor topology with shorter chains of physical qubits compared to the previous $2000Q$ [18]. To map logical qubits which are connected to more than 15 other logical qubits, chaining of physical qubits is necessary, *i.e.,* entanglement between qubit states. These chains are maintained by auxiliary magnetic fields and can break during annealings. The mechanism of resolving broken chains is majority voting. Minor embeddings are performed automatically via [24], and each annealing takes $20\mu s$ – apart from AQC problem transmission overheads which can sum up to 0.1 sec, minor embedding time (which, however, can be pre-computed for multiple problem sizes) and AQC waiting time. We access D-Wave machines remotely through *Leap2* [32]. The total AQC runtime spent in the experiments amounts to ≈ 15.5 min. ($> 5 \cdot 10^5$ annealings in total). We show exemplary embeddings in our supplement.

Evaluations on the real dataset. We begin by putting D-Wave to test on synchronizing real data. Tab. 1 shows the accuracy of the best (lowest energy) solution we attain for different categories, as well as averaged over all classes. It is seen that while quantum solution is not the top-performer, it is certainly on par with the well engineered approaches of

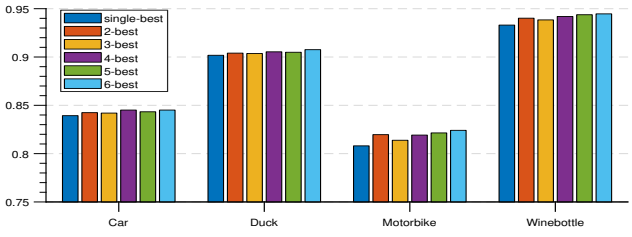

Figure 4. Samples from the quantum annealer can be helpful in improving the solution quality or reporting uncertainty.

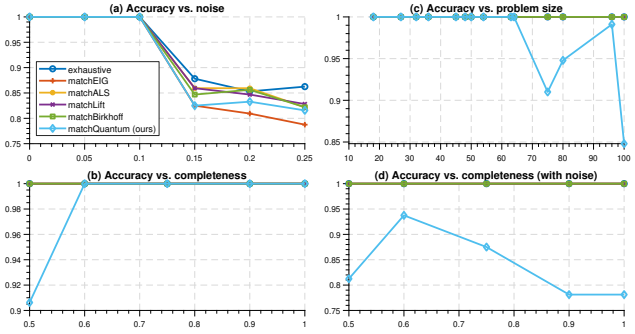

Figure 5. Evaluations on the synthetic dataset (n = 4, m = 4).

the state of the art. Moreover, on a theoretical note, it enjoys polynomial speedup. Note that the exhaustive solution performs the best on this dataset. This further encourages research on making better quantum computers reducing the gap between the true and the machine-computed optima.

Explaining the posterior. While not being a true Bayesian inference, quantum annealing can provide samples from the energy landscape or the Hamiltonian. The samples could be used either, for instance, in associating a confidence to solutions or maybe in scenarios like active learning. To assess the usability of the samples, instead of looking into the lowest energy solution, we look at k-lowest energy solutions of the previous real data evaluation. We process them jointly and correct the erroneous bits by replacing them with the most frequent ones over all the samples. Fig. 4 shows on the real benchmark that for increasing k values, the accuracy also increases. This validates that different samples from a quantum annealer can be informative in exploring the energy landscape *i.e.,* posterior defined in Sec. 4.

Evaluations on the synthetic dataset. We now interrogate various characteristics of our quantum approach, *QuantumSync*. Our results are plotted in Fig. 5. First, we inspect the behaviour under increasing noise. Fig. 5(a) shows that all the methods can handle low noise regimes. The increasing noise similarly impacts the methods we test. Our approach while not being superior to any, is on par. Fig. 5(b) shows that *QuantumSync* is significantly impacted from the increased problem sizes (mn^2). This shows perhaps the most important limitation of current quantum computers, *i.e.,* we can only reliably handle the problems of size < 64. Note that, this is also the size of our real sub-Willow dataset. The rest of the plots (c,d) show the impact of connectivity (graph completeness) on the solution quality. With noise ($\sigma = 0.1$ for this case) sparse graphs cause significant problems.

Parameter selection. We now evaluate, with the help of our synthetic data, how the solution and its probability changes w.r.t. the chain strength χ and λ. Both parameters strongly influence the probability to measure optimal solutions. χ being too high keeps the chains unbroken during annealings, but adversely affects the solutions. Too low χ

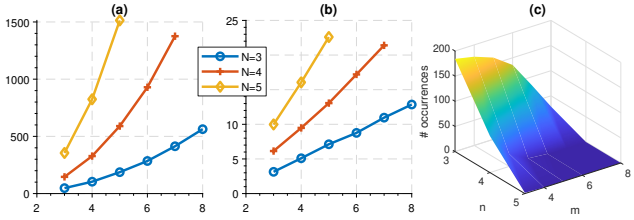

Figure 6. For different n and an increasing number of views m, (**a**) plots the number of qubits required to map a problem; and (**b**) at $\chi = 3.0$, shows the required maximum chain length required to embed the problem on Advantage 1.1. (**c**) plots the average number of measured optimal solutions in 200 samples, for different pairs of n and m (averaged over 50 repetitions).

lead to broken chains which, however, in many cases can be resolved with majority voting. Initial trial experiments help us to identify $\chi = 3.0$ and $\lambda = 2.5$ as optimal parameters which are kept fixed in our experiments. Note that the ablative study for λ on a classical computer in Sec. 5.2 also suggests $\lambda > 2.0$ as a suitable value for a range of (n, m). Tab. 2 highlights the effect of varying χ in the tests with $n = 3, m = 3$ and $n = 4, m = 4$ on Advantage 1.1. We report the number of logical qubits of the target problem n_l, the number of physical qubits required to embed the problem on AQC n_{ph}, the average maximum chain length of the embedding l and the average number of annealings leading to the optimal solution out of 200 samples. Each test for each χ is repeated 50 times.

Table 2. The table summarises the effect of varying chain strength χ on the number of measured optimal solutions out of 200, for $n = 3, m = 3$ (first row) and $n = 4, m = 4$ (second row).

n_l / n_{ph} / l	$\xi = 1$	$\xi = 2$	$\xi = 3$	$\xi = 4$	$\xi = 5$
18 / 48 / 3.2	120.9 ± 48.8	187.8 ± 11.5	180.9 ± 11.2	150.3 ± 19.9	85.3 ± 23.6
48 / 325 / 9.5	0.1 ± 0.30	9.9 ± 10.08	17.3 ± 14.0	6.1 ± 7.77	1.42 ± 2.72

Different problem sizes and minor embedding. Next, we systematically analyze which problem sizes can be successfully embedded on Pegasus topology of Advantage system 1.1 and solved globally optimally with high probability over 200 samples. In each configuration of n and m which can be successfully solved on D-Wave, we report the average number of measurements corresponding to the global optimum and its standard deviation over 50 repetitions with 200 annealings each. We also analyze minor embedding in the same experiments and report the average number of physical qubits used in the embedding, the maximum chain lengths along with their standard deviations over 50 runs. Fig. 6 visualises the experimental outcomes. We see that with the increasing number of logical qubits, the number of physical qubits required for the embedding as well as the maximum chain length increase (Fig. 6-(a),(b)). For $n = 3$ and $m = 3$, the ratio is $c = \frac{n_{ph}}{n_l} \approx 2.65$. It increases to $c \approx 14.5$ for $n = 5$ and $m = 5$. This is not surprising, since longer qubit chains increase the probability of chain

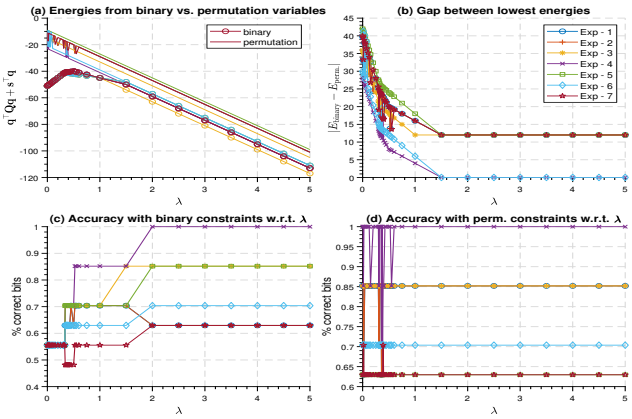

Figure 7. Studying the problem by exhaustive solutions on a classical computer. We solve seven random synthetic problems with $n = 3$, $m = |\mathcal{V}| = 3$ and $\sigma = 0.2$ by searching over **binary variables** (binary) or **permutation matrices** (perm.) exhaustively for the global optimum. We show: **(a)** the energies attained for both cases, **(b)** the gap (absolute difference) between the lowest energies in (a), **(c)** the ratio of correctly guessed bits as a function of the regularizer (λ) for the binary constraints, **(d)** same plot in (c) but for permutation matrices. Note, due to significant noise, the global optimum is not always the same as ground truth.

breaks and, hence, decrease the overall probability to measure optimal solution. Fig. 6-(c) shows the number of optimal measurements out of 200, for varying n and m. We see that our *QuantumSync* can solve the cases $n = 3, m = 8$, with probability to measure optimal solution in a single annealing $\rho = 4.39\%$; $n = 4, m = 4$, with $\rho = 12.5\%$, and $n = 5, m = 4$, with $\rho < 1\%$. At the same time, for problems with $n = 3$ and $m < 7$, $\rho > 62\%$. Note that certain problems are unmappable to D-Wave due to their size and thus we cannot report the statistics for those.

5.2. Ablation Studies on a Classical Computer

We now study the global minima of our problem on a classical computer and we design seven random problem instances that are globally solvable on standard hardware. Hence, we choose $n = 3$, $m = |\mathcal{V}| = 3$ and $C = 1$ (fully connected graph). For such a small size, we could exhaustively search for the global optimum both over binary variables and permutation matrices. Note that while the latter is the reasonable (actual) search space, an AQC can only optimize over the former. Hence, we are interested in quantifying the gap between the two and verify that our formulation indeed allows a QUBO-solver to achieve the global optimum for the problem at hand.

How can we choose the regularization coefficient λ? The coefficient λ is one of the most important hyper-parameters of our algorithm as it balances the data term *vs* permutation penalty. Hence, we investigate its behavior. Fig. 7 shows that over those seven experiments where $\sigma = 0.2$, while the lowest energies between the two solutions can differ (**a** and **b**), for a wide variety of λ-choices the accuracy attained by binary optimization and permutation optimization can be very comparable (**c** *vs* **d**). As long as λ is not small (*e.g.,* < 2), we observe almost identical performance. This positive result has motivated us to settle for a single value $\lambda = 2.5$ for all of our evaluations (including Sec. 5.1).

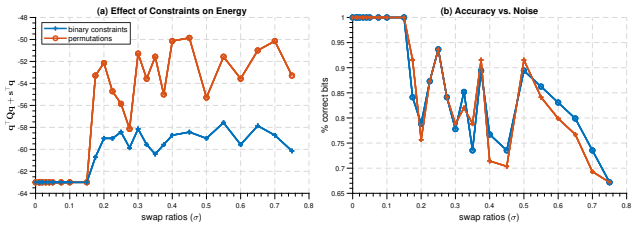

Figure 8. Effect of noise on the accuracy of the solution obtained by optimizing either over binary variables or permutation matrices. **(a)** Energy levels when the solution is restricted to binary variables or permutation matrices. **(b)** Accuracy attained by these two restricted solutions.

Are permutation constraints effective? As D-Wave cannot search over the permutations but only over binary variables, it is of interest to see whether our permutation-ness regularization really works. To investigate that, we design random experiments ($n = 3, m = 3$) with increasing noise (swap ratio) where the GT is known. We then form the constrained $\mathbf{Q}$ matrix and solve it via exhaustive search on a classical computer. Averaged over seven experiments, Fig. 8 shows that: (i) for low noise regime, optimizing over general binary variables $\mathcal{B}$ or over permutations $\mathcal{P}$ are indifferent and global optimum can always be found, (ii) for higher noise levels, while the energies attained seem to differ, the final accuracy is very similar. Hence, we conclude that injecting permutation constraints into $\mathbf{Q}$ as proposed is useful and makes it possible to use binary variables instead of permutations. This justifies why an adiabatic computer such as D-Wave could obtain global optima.

6. Conclusion

We presented *QuantumSync*, the first quantum approach to synchronization. We specifically focused on the group of permutations and showed how to formulate such problems for an adiabatic computer. We then used the cutting-edge quantum hardware to solve real-world problems with global guarantees. Our forward-looking experiments demonstrate that quantum computing hardware has reached the level that it can be applied to real-world problems with high potential to improve upon the known classical methods. We believe that our technique can inspire new generations of better algorithms for related and other computer vision problems, and we expect to see more work in the field in near future.

Acknowledgements. The work was supported by the ERC Grant 4DReply (770784), a Vannevar Bush Faculty fellowship, a grant from the Stanford-Ford Alliance and gifts from Amazon AWS and Snap, Inc.

References

[1] Ibm q experience. https://quantum-computing.ibm.com. Accessed on the 17.01.2021. 2

[2] Technical description of the d-wave quantum processing unit. https://docs.dwavesys.com/docs/latest/doc_qpu.html. Accessed on the 17.01.2021. 2, 4

[3] Esma Aïmeur, Gilles Brassard, and Sébastien Gambs. Quantum speed-up for unsupervised learning. *Machine Learning*, 90(2):261–287, 2013. 2

[4] Federica Arrigoni and Andrea Fusiello. Synchronization problems in computer vision with closed-form solutions. *International Journal of Computer Vision (IJCV)*, 2019. 1, 2, 3

[5] Federica Arrigoni, Eleonora Maset, and Andrea Fusiello. Synchronization in the symmetric inverse semigroup. In *International Conference on Image Analysis and Processing (ICIAP)*, pages 70–81, 2017. 1, 2

[6] Federica Arrigoni, Beatrice Rossi, and Andrea Fusiello. Spectral synchronization of multiple views in se (3). *SIAM Journal on Imaging Sciences*, 9(4):1963–1990, 2016. 1, 2

[7] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C. Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando G. S. L. Brandao, David A. Buell, et al. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, 2019. 2

[8] Vadim Belov. On geometry and symmetries in classical and quantum theories of gauge gravity. *arXiv:1905.06931*, 2019. 4

[9] Florian Bernard, Johan Thunberg, Peter Gemmar, Frank Hertel, Andreas Husch, and Jorge Goncalves. A solution for multi-alignment by transformation synchronisation. In *Computer Vision and Pattern Recognition (CVPR)*, pages 2161–2169, 2015. 2

[10] Florian Bernard, Johan Thunberg, Jorge Goncalves, and Christian Theobalt. Synchronisation of partial multi-matchings via non-negative factorisations. *Pattern Recognition*, 92:146 – 155, 2019. 2

[11] Dominic W. Berry, Graeme Ahokas, Richard Cleve, and Barry C. Sanders. Efficient quantum algorithms for simulating sparse hamiltonians. *Communications in Mathematical Physics*, 270:359–371, 2007. 2

[12] Tolga Birdal, Michael Arbel, Umut Simsekli, and Leonidas J Guibas. Synchronizing probability measures on rotations via optimal transport. In *Computer Vision and Pattern Recognition (CVPR)*, pages 1569–1579, 2020. 1, 2

[13] Tolga Birdal, Benjamin Busam, Nassir Navab, Slobodan Ilic, and Peter Sturm. Generic primitive detection in point clouds using novel minimal quadric fits. *IEEE transactions on pattern analysis and machine intelligence*, 42(6):1333–1347, 2019. 1

[14] Tolga Birdal, Umut Şimşekli, M. Onur Eken, and Slobodan Ilic. Bayesian Pose Graph Optimization via Bingham Distributions and Tempered Geodesic MCMC. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018. 1, 2

[15] Tolga Birdal and Slobodan Ilic. Cad priors for accurate and flexible instance reconstruction. In *International Conference on Computer Vision (ICCV)*, pages 133–142, 2017. 1

[16] Tolga Birdal and Umut Simsekli. Probabilistic permutation synchronization using the riemannian structure of the birkhoff polytope. In *Computer Vision and Pattern Recognition (CVPR)*, pages 11105–11116, 2019. 1, 2, 4, 5, 6

[17] Dan Boneh and Richard J. Lipton. Quantum cryptanalysis of hidden linear functions. In *Advances in Cryptology (CRYPTO)*, 1995. 2

[18] Kelly Boothby, Paul Bunyk, Jack Raymond, and Aidan Roy. Next-Generation Topology of D-Wave Quantum Processors. *arXiv e-prints*, 2020. 6

[19] Max Born and Vladimir Fock. Beweis des adiabatensatzes. *Zeitschrift für Physik*, 51(3):165–180, 1928. 4

[20] Edward Boyda, Saikat Basu, Sangram Ganguly, Andrew Michaelis, Supratik Mukhopadhyay, and Ramakrishna R. Nemani. Deploying a quantum annealing processor to detect tree cover in aerial imagery of california. *PLoS ONE*, 12, 2017. 2

[21] Jesus Briales and Javier Gonzalez-Jimenez. Cartan-sync: Fast and global se (d)-synchronization. *IEEE Robotics and Automation Letters*, 2(4):2127–2134, 2017. 2

[22] Justin J. Brooke, David Bitko, Thomas F. Rosenbaum, and Gabriel Aeppli. Quantum annealing of a disordered magnet. *Science*, 284(5415):779–781, 1999. 2

[23] Cesar Cadena, Luca Carlone, Henry Carrillo, Yasir Latif, Davide Scaramuzza, José Neira, Ian Reid, and John J. Leonard. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Transactions on robotics*, 32(6):1309–1332, 2016. 1, 2

[24] Jun Cai, William G. Macready, and Aidan Roy. A practical heuristic for finding graph minors. *arXiv e-prints*, 2014. 4, 6

[25] Luca Carlone, Roberto Tron, Kostas Daniilidis, and Frank Dellaert. Initialization techniques for 3d slam: a survey on rotation estimation and its use in pose graph optimization. In *International Conference on Robotics and Automation (ICRA)*, pages 4597–4604, 2015. 1, 2

[26] Avishek Chatterjee and Venu Madhav Govindu. Efficient and robust large-scale rotation averaging. In *International Conference on Computer Vision (ICCV)*, pages 521–528, 2013. 4

[27] Avishek Chatterjee and Venu Madhav Govindu. Robust relative rotation averaging. *IEEE transactions on pattern analysis and machine intelligence (TPAMI)*, 40(4):958–972, 2017. 2

[28] Kunal N. Chaudhury, Yuehaw Khoo, and Amit Singer. Global registration of multiple point clouds using semidefinite programming. *SIAM Journal on Optimization*, 25(1):468–501, 2015. 2

[29] Minsu Cho, Karteek Alahari, and Jean Ponce. Learning graphs to match. In *International Conference on Computer Vision (ICCV)*, pages 25–32, 2013. 6, 16, 17, 18, 19

[30] Isaac L. Chuang, Neil Gershenfeld, and Mark Kubinec. Experimental implementation of fast quantum searching. *Phys. Rev. Lett.*, 80:3408–3411, 1998. 2

[31] B. David Clader, Bryan C. Jacobs, and Chad R. Sprouse. Preconditioned quantum linear system algorithm. *Phys. Rev. Lett.*, 110, 2013. 2

[32] D-Wave Systems, Inc. Leap datasheet v10. https://www.dwavesys.com/sites/default/files/Leap_Datasheet_v10_0.pdf, 2020. online; latest access on the 8 November 2020. 2, 6

[33] Vasil S. Denchev, Sergio Boixo, Sergei V. Isakov, Nan Ding, Ryan Babbush, Vadim Smelyanskiy, John Martinis, and Hartmut Neven. What is the computational value of finite-range tunneling? *Phys. Rev. X*, 6:031015, 2016. 2

[34] Haowen Deng, Mai Bui, Nassir Navab, Leonidas Guibas, Slobodan Ilic, and Tolga Birdal. Deep bingham networks: Dealing with uncertainty and ambiguity in pose estimation. *arXiv preprint arXiv:2012.11002*, 2020. 1

[35] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition (CVPR)*, 2009. 6

[36] David P. DiVincenzo. The physical implementation of quantum computation. *Fortschritte der Physik*, 48(9-11):771–783, 2000. 2

[37] Alvaro Donis-Vela and Juan Carlos Garcia-Escartin. A quantum primality test with order finding. *Quantum Info. Comput.*, 18(13–14):1143–1151, 2018. 2

[38] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, Joshua Lapan, Andrew Lundgren, and Daniel Preda. A quantum adiabatic evolution algorithm applied to random instances of an np-complete problem. *Science*, 292(5516):472–475, 2001. 2, 4

[39] Richard P. Feynman. Simulating physics with computers. *International Journal of Theoretical Physics*, 21(6):467–488, 1982. 2

[40] Johan Fredriksson and Carl Olsson. Simultaneous multiple rotation averaging using lagrangian duality. In *Asian Conference on Computer Vision (ACCV)*, pages 245–258, 2012. 4

[41] Frank Gaitan and Lane Clark. Graph isomorphism and adiabatic quantum computing. *Physical Review A*, 89(2), 2014. 3

[42] Arvind Giridhar and Praveen R. Kumar. Distributed clock synchronization over wireless networks: Algorithms and analysis. In *IEEE Conference on Decision and Control (CDC)*, pages 4915–4920, 2006. 1

[43] Zan Gojcic, Caifa Zhou, Jan D Wegner, Leonidas J Guibas, and Tolga Birdal. Learning multiview 3d point cloud registration. In *Computer Vision and Pattern Recognition (CVPR)*, pages 1759–1769, 2020. 2

[44] Vladislav Golyanik, Soshi Shimada, and Christian Theobalt. Fast simultaneous gravitational alignment of multiple point sets. In *International Conference on 3D Vision (3DV)*, 2020. 1

[45] Vladislav Golyanik and Christian Theobalt. A quantum computational approach to correspondence problems on point sets. In *Computer Vision and Pattern Recognition (CVPR)*, 2020. 3, 4, 17

[46] Vladislav Golyanik and Christian Theobalt. A quantum computational approach to correspondence problems on point sets (supplementary material about the experiments on d-wave 2000q). In *Computer Vision and Pattern Recognition (CVPR)*, 2020. 3, 17

[47] Vladislav Golyanik, Christian Theobalt, and Didier Stricker. Accelerated gravitational point set alignment with altered physical laws. In *International Conference on Computer Vision (ICCV)*, 2019. 17

[48] Venu Madhav Govindu. Lie-algebraic averaging for globally consistent motion estimation. In *Computer Vision and Pattern Recognition (CVPR)*, 2004. 2

[49] Venu Madhav Govindu and A Pooja. On averaging multiview relations for 3d scan registration. *IEEE Transactions on Image Processing*, 23(3):1289–1302, 2014. 1, 2

[50] Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Annual ACM Symposium on Theory of Computing*, 1996. 2

[51] Leonidas J Guibas, Qixing Huang, and Zhenxiao Liang. A condition number for joint optimization of cycle-consistent networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1005–1015, 2019. 3

[52] Sean Hallgren. Fast quantum algorithms for computing the unit group and class group of a number field. In *Annual ACM Symposium on Theory of Computing*, page 468–474, 2005. 2

[53] Aram W. Harrow, Avinatan Hassidim, and Seth Lloyd. Quantum algorithm for linear systems of equations. *Phys. Rev. Lett.*, 103, 2009. 2

[54] Richard Hartley, Khurrum Aftab, and Jochen Trumpf. L1 rotation averaging using the weiszfeld algorithm. In *Computer Vision and Pattern Recognition (CVPR)*, pages 3041–3048, 2011. 2, 4

[55] Richard Hartley, Jochen Trumpf, Yuchao Dai, and Hongdong Li. Rotation averaging. *International Journal of Computer Vision (IJCV)*, 103(3), 2013. 1, 2, 4

[56] Nan Hu, Qixing Huang, Boris Thibert, and Leonidas Guibas. Distributable consistent multi-object matching. In *Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[57] Qixing Huang, Zhenxiao Liang, Haoyun Wang, Simiao Zuo, and Chandrajit Bajaj. Tensor maps for synchronizing heterogeneous shape collections. *ACM Transactions on Graphics (TOG)*, 38(4):106, 2019. 2

[58] Qi-Xing Huang and Leonidas Guibas. Consistent shape maps via semidefinite programming. In *Eurographics/ACM SIGGRAPH Symposium on Geometry Processing (SGP)*. Eurographics Association, 2013. 1, 3, 5, 6

[59] Xiangru Huang, Zhenxiao Liang, Xiaowei Zhou, Yao Xie, Leonidas J Guibas, and Qixing Huang. Learning transformation synchronization. In *Computer Vision and Pattern Recognition (CVPR)*, pages 8082–8091, 2019. 2

[60] Stephen P. Jordan, Keith S. M. Lee, and John Preskill. Quantum algorithms for quantum field theories. *Science*, 336(6085):1130–1133, 2012. 2

[61] Tadashi Kadowaki and Hidetoshi Nishimori. Quantum annealing in the transverse ising model. *Phys. Rev. E*, 58:5355–5363, 1998. 2, 4

[62] Bruce E. Kane. A silicon-based nuclear spin quantum computer. *Nature*, 393:133–137, 1998. 2

[63] Ivan Kassal, Stephen P. Jordan, Peter J. Love, Masoud Mohseni, and Alán Aspuru-Guzik. Polynomial-time quantum algorithm for the simulation of chemical dynamics. *Proceedings of the National Academy of Sciences*, 105(48):18681–18686, 2008. 2

[64] Scott Kirkpatrick, C. Daniel Gelatt, and Mario P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983. 2

[65] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems (NeurIPS)*, pages 1097–1105, 2012. 6

[66] Trevor Lanting et al. Entanglement in a quantum annealing processor. *Phys. Rev. X*, 4:021041, 2014. 2

[67] Trevor Lanting, Andrew D. King, Bram Evert, and Emile Hoskinson. Experimental demonstration of perturbative anticrossing mitigation using nonuniform driver hamiltonians. *Phys. Rev. A*, 96:042322, Oct 2017. 2

[68] Francois Le Gall. Improved output-sensitive quantum algorithms for boolean matrix multiplication. In *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1464–1476, 2012. 2

[69] Bjoern Lekitsch, Sebastian Weidt, Austin G. Fowler, Klaus Mølmer, Simon J. Devitt, Christof Wunderlich, and Winfried K. Hensinger. Blueprint for a microwave trapped ion quantum computer. *Science Advances*, 3(2), 2017. 2

[70] Junde Li and Swaroop Ghosh. Quantum-soft qubo suppression for accurate object detection. In *European Conference on Computer Vision (ECCV)*, 2020. 2, 17

[71] Seth Lloyd, Masoud Mohseni, and Patrick Rebentrost. Quantum principal component analysis. *Nature Physics*, 10:631–633, 2014. 2

[72] Yuri Manin. *Computable and Noncomputable*. Sov. Radio., 1980. 2

[73] Eleonora Maset, Federica Arrigoni, and Andrea Fusiello. Practical and efficient multi-view matching. In *International Conference on Computer Vision (ICCV)*, 2017. 2, 5, 6

[74] Andrea Morello, Jarryd J. Pla, Floris A. Zwanenburg, Kok W. Chan, Kuan Y. Tan, Hans Huebl, Mikko Möttönen, Christopher D. Nugroho, Changyi Yang, Jessica A. van Donkelaar, Andrew D. C. Alves, David N. Jamieson, Christopher C. Escott, Lloyd C. L. Hollenberg, Robert G. Clark, and Andrew S. Dzurak. Single-shot readout of an electron spin in silicon. *Nature*, 467:687–691, 2010. 2

[75] James Munkres. Algorithms for the assignment and transportation problems. *Journal of the society for industrial and applied mathematics*, 5(1):32–38, 1957. 6

[76] Florian Neukart, Gabriele Compostella, Christian Seidel, David von Dollen, Sheir Yarkoni, and Bob Parney. Traffic flow optimization using a quantum annealer. *Frontiers in ICT*, 4:29, 2017. 2

[77] Hartmut Neven, Vasil S. Denchev, Geordie Rose, and William G. Macready. Qboost: Large scale classifier training with adiabatic quantum optimization. In *Asian Conference on Machine Learning (ACML)*, 2012. 2

[78] Andy Nguyen, Mirela Ben-Chen, Katarzyna Welnicka, Yinyu Ye, and Leonidas Guibas. An optimization approach to improving collections of shape maps. In *Computer Graphics Forum*, pages 1481–1491, 2011. 3

[79] Nga T. T. Nguyen, Garrett T. Kenyon, and Boram Yoon. A regression algorithm for accelerated lattice qcd that exploits sparse inference on the d-wave quantum annealer. *Scientific Reports*, 10, 2020. 2

[80] Gerardo Ortiz, James E. Gubernatis, Emanuel Knill, and Raymond Laflamme. Quantum algorithms for fermionic simulations. *Phys. Rev. A*, 2001. 2

[81] Daniel O'Malley, Velimir V. Vesselinov, Boian S. Alexandrov, and Ludmil B. Alexandrov. Nonnegative/binary matrix factorization with a d-wave quantum annealer. *PLOS ONE*, 13(12), 12 2018. 2

[82] Deepti Pachauri, Risi Kondor, and Vikas Singh. Solving the multi-way matching problem by permutation synchronization. In *Advances in neural information processing systems*, pages 1860–1868, 2013. 2, 4

[83] Davis Rempe, Tolga Birdal, Yongheng Zhao, Zan Gojcic, Srinath Sridhar, and Leonidas J. Guibas. Caspr: Learning canonical spatiotemporal point cloud representations. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 1

[84] David M. Rosen, Luca Carlone, Afonso S. Bandeira, and John J. Leonard. Se-sync: A certifiably correct algorithm for synchronization over the special euclidean group. *The International Journal of Robotics Research*, 38(2-3):95–125, 2019. 2

[85] Sitapa Rujikietgumjorn and Robert T. Collins. Optimized pedestrian detection for multiple and occluded people. In *Computer Vision and Pattern Recognition (CVPR)*, 2013. 2

[86] Renato F. Salas-Moreno, Richard A. Newcombe, Hauke Strasdat, Paul H. J. Kelly, and Andrew J. Davison. Slam++: Simultaneous localisation and mapping at the level of objects. In *Computer Vision and Pattern Recognition (CVPR)*, pages 1352–1359, 2013. 1, 2

[87] Michele Schiavinato and Andrea Torsello. Synchronization over the birkhoff polytope for multi-graph matching. In *International Workshop on Graph-Based Representations in Pattern Recognition*, pages 266–275, 2017. 2

[88] Maria Schuld, Ilya Sinayskiy, and Francesco Petruccione. Prediction by linear regression on a quantum computer. *Phys. Rev. A*, 94, 2016. 2

[89] Marcel Seelbach Benkner, Vladislav Golyanik, Christian Theobalt, and Michael Moeller. Adiabatic quantum graph matching with permutation matrix constraints. In *International Conference on 3D Vision (3DV)*, 2020. 3, 17, 18

[90] Peter W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Annual Symposium on Foundations of Computer Science*, 1994. 2

[91] Daniel R. Simon. On the power of quantum computation. In *Annual Symposium on Foundations of Computer Science*, 1994. 2

[92] Barbara Simons. An overview of clock synchronization. In *Fault-Tolerant Distributed Computing*. Springer, 1990. 1

[93] Amit Singer. Angular synchronization by eigenvectors and semidefinite programming. *Applied and computational harmonic analysis*, 30(1):20–36, 2011. 1

[94] Tobias Stollenwerk, Elisabeth Lobe, and Martin Jung. Flight gate assignment with a quantum annealer. In *International Workshop on Quantum Technology and Optimization Problems*, 2019. 3

[95] Andriyan Suksmono and Yuichiro Minato. Finding hadamard matrices by a quantum annealing machine. *Scientific Reports*, 9, 12 2019. 2

[96] Amnon Ta-Shma. Inverting well conditioned matrices in quantum logspace. In *ACM Symposium on Theory of Computing*, 2013. 2

[97] Johan Thunberg, Florian Bernard, and Jorge Goncalves. Distributed methods for synchronization of orthogonal matrices over graphs. *Automatica*, 80:243–252, 2017. 2

[98] Roberto Tron and Kostas Daniilidis. Statistical pose averaging with non-isotropic and incomplete relative measurements. In *European Conference on Computer Vision (ECCV)*, 2014. 2

[99] Roberto Tron and Rene Vidal. Distributed 3-d localization of camera sensor networks from 2-d image measurements. *IEEE Transactions on Automatic Control*, 59(12):3325–3340, 2014. 2

[100] Wim van Dam and Igor E. Shparlinski. Classical and Quantum Algorithms for Exponential Congruences. *arXiv e-prints*, 2008. 2

[101] Lanhui Wang and Amit Singer. Exact and stable recovery of rotations for robust synchronization. *Information and Inference: A Journal of the IMA*, 2(2):145–193, 2013. 2

[102] Qianqian Wang, Xiaowei Zhou, and Kostas Daniilidis. Multi-image semantic matching by mining consistent features. In *Computer Vision and Pattern Recognition (CVPR)*, 2018. 2, 6, 16

[103] Janus H. Wesenberg, Arzhang Ardavan, G. Andrew D. Briggs, John J. L. Morton, Robert J. Schoelkopf, David I. Schuster, and Klaus Mølmer. Quantum computing with an electron spin ensemble. *Phys. Rev. Lett.*, 103, 2009. 2

[104] William K. Wootters and Wojciech H. Zurek. A single quantum cannot be cloned. *Nature*, 299(5886):802–803, 1982. 4

[105] Jin-Gang Yu, Gui-Song Xia, Ashok Samal, and Jinwen Tian. Globally consistent correspondence of multiple feature sets using proximal gauss–seidel relaxation. *Pattern Recognition*, 51:255–267, 2016. 2

[106] Christopher Zach, Manfred Klopschitz, and Marc Pollefeys. Disambiguating visual relations using loop constraints. In *Computer Vision and Pattern Recognition (CVPR)*, pages 1426–1433, 2010. 3

[107] Christof Zalka. Efficient simulation of quantum systems by quantum computers. *Fortschritte der Physik*, 46(6-8):877–879, 1998. 2

[108] Zaiwei Zhang, Zhenxiao Liang, Lemeng Wu, Xiaowei Zhou, and Qixing Huang. Path-invariant map networks. In *Computer Vision and Pattern Recognition (CVPR)*, pages 11084–11094, 2019. 1, 3

[109] Zhikuan Zhao, Alejandro Pozas-Kerstjens, Patrick Rebentrost, and Peter Wittek. Bayesian deep learning on a quantum computer. *Quantum Machine Intelligence*, pages 41–51, 2019. 2

[110] Xiaowei Zhou, Menglong Zhu, and Kostas Daniilidis. Multi-image matching via fast alternating minimization. In *International Conference on Computer Vision (ICCV)*, pages 4032–4040, 2015. 6

[111] Kenneth M. Zick, Omar Shehab, and Matthew French. Experimental quantum annealing: case study involving the graph isomorphism problem. *Scientific reports*, 5(1), 2015. 3

[112] Floris A. Zwanenburg, Andrew S. Dzurak, Andrea Morello, Michelle Y. Simmons, Lloyd C. L. Hollenberg, Gerhard Klimeck, Sven Rogge, Susan N. Coppersmith, and Mark A. Eriksson. Silicon quantum electronics. *Rev. Mod. Phys.*, 85:961–1019, 2013. 2

Quantum Permutation Synchronization —Supplementary Material—

This part supplements our main paper by providing (i) proofs of the theorems and propositions contained in the main paper; (ii) further insights into the constraints that we propose; (iii) visualizations of minor embeddings on D-Wave; (iv) details and illustrations of the synthetic data; and (v) further descriptions of our real dataset. The figures and tables introduced in this document are referenced using Roman numerals, whereas the original references to the main matter are preserved.

A. Proof of Theorems

Proof of Theorem 1. The proof follows from plugging Eq. (2) into the definition of a null-cycle in Eq. (1) of the main paper, obtaining:

$$
\begin{aligned}
f_c &= f_{1,2} \circ f_{2,3} \circ \cdots \circ f_{(n-1),n} \circ f_{n,1} && (16)\\
&= \left(f_1 \circ f_2^{-1}\right) \circ \left(f_2 \circ f_3^{-1}\right) \circ \cdots \circ \left(f_n \circ f_1^{-1}\right) && (17)\\
&= f_1 \circ \left(f_2^{-1} \circ f_2\right) \circ \left(f_3^{-1} \circ f_3\right) \circ \cdots \circ \left(f_n^{-1} \circ f_n\right) \circ f_1^{-1} && (18)\\
&= f_1 \circ f_1^{-1} = f_{null} \quad \forall c \in \mathcal{C}. && (19)
\end{aligned}
$$

□

Proof of Theorem 2. Without loss of generality, for an edge (i, j), we consider the transformed cycle consistency constraint:

$$
(f_i \circ f_g) \circ (f_j \circ f_g)^{-1} = (f_i \circ f_g) \circ (f_g^{-1} \circ f_j) = f_i \circ f_j^{-1}. \tag{20}
$$

This shows that consistency relation is unchanged under the action of an arbitrary element in the group f_g. □

B. Extended Proof of Proposition 1

Here we provide more steps to the derivation in Prop. 1 of the main paper:

$$
\begin{aligned}
\mathbf{X}^\star &= \underset{\{\mathbf{P}_i \in \mathcal{P}_n^N\}}{\arg\min} \sum_{(i,j)\in\mathcal{E}} \|\mathbf{P}_{ij} - \mathbf{P}_i \mathbf{P}_j^\top\|_{\mathrm{F}}^2 && (21)\\
&= \underset{\{\mathbf{P}_i \in \mathcal{P}_n^N\}}{\arg\min} \sum_{(i,j)\in\mathcal{E}} \|\mathbf{P}_{ij}\|_{\mathrm{F}}^2 - \|\mathbf{P}_i \mathbf{P}_j^\top\|_{\mathrm{F}}^2 - 2\mathrm{tr}(\mathbf{P}_j \mathbf{P}_i^\top \mathbf{P}_{ij})\\
&= \underset{\{\mathbf{P}_i \in \mathcal{P}_n^N\}}{\arg\min}\ 2N^2 n - 2 \sum_{(i,j)\in\mathcal{E}} \mathrm{tr}(\mathbf{P}_j \mathbf{P}_i^\top \mathbf{P}_{ij}) && (22)\\
&= \underset{\{\mathbf{P}_i \in \mathcal{P}_n^N\}}{\arg\min}\ - \sum_{(i,j)\in\mathcal{E}} \mathrm{tr}(\mathbf{P}_j \mathbf{P}_i^\top \mathbf{P}_{ij}) && (23)\\
&= \underset{\{\mathbf{P}_i \in \mathcal{P}_n^N\}}{\arg\min}\ - \sum_{(i,j)\in\mathcal{E}} \mathrm{vec}(\mathbf{P}_i)^\top (\mathbf{I} \otimes \mathbf{P}_{ij}) \mathrm{vec}(\mathbf{P}_j)\\
&= \underset{\{\mathbf{P}_i \in \mathcal{P}_n^N\}}{\arg\min}\ - \sum_{(i,j)\in\mathcal{E}} \mathbf{q}_i^\top (\mathbf{I} \otimes \mathbf{P}_{ij}) \mathbf{q}_j = \underset{\{\mathbf{P}_i \in \mathcal{P}_n^N\}}{\arg\min}\ \mathbf{q}^\top \mathbf{Q}' \mathbf{q}. && (24)
\end{aligned}
$$

C. Proof of Proposition 2

Linear constraints can be injected into a QUBO problem in the following manner:

$$
\begin{aligned}
\mathbf{x}^\star &= \underset{\mathbf{x}\in\mathcal{B}}{\arg\min}\ \mathbf{x}^\top \mathbf{Q}' \mathbf{x} + \lambda \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 && (25)\\
&= \underset{\mathbf{x}\in\mathcal{B}}{\arg\min}\ \mathbf{x}^\top \mathbf{Q}' \mathbf{x} + \lambda \big(\mathbf{A}\mathbf{x} - \mathbf{b}\big)^\top \big(\mathbf{A}\mathbf{x} - \mathbf{b}\big) && (26)\\
&= \underset{\mathbf{x}\in\mathcal{B}}{\arg\min}\ \mathbf{x}^\top \big(\mathbf{Q}' + \lambda \mathbf{A}^\top \mathbf{A}\big)\mathbf{x} - 2\lambda \mathbf{b}^\top \mathbf{A}\mathbf{x} && (27)\\
&= \underset{\mathbf{x}\in\mathcal{B}}{\arg\min}\ \mathbf{x}^\top \mathbf{Q}\mathbf{x} + \mathbf{s}^\top \mathbf{x}. && (28)
\end{aligned}
$$

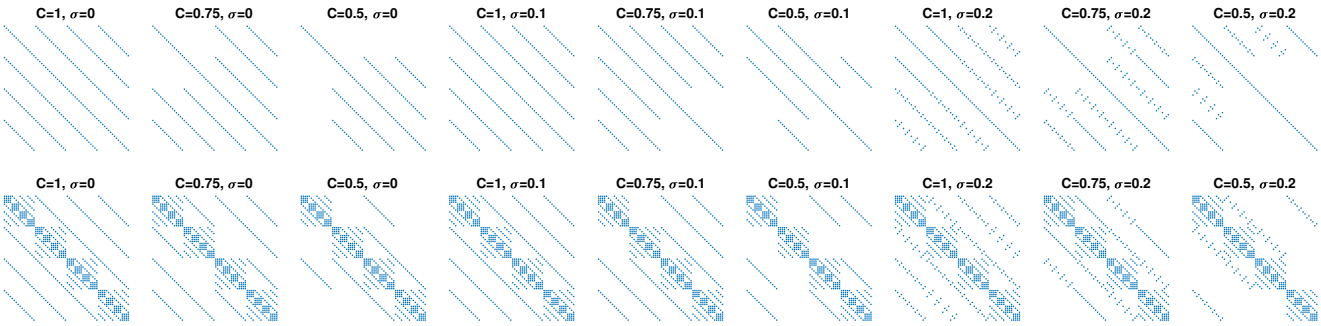

Figure 9. Sparsity patterns of the unconstrained (**top row**) and constrained (**bottom row**) $\mathbf{Q}$-matrices. For the former we use $\lambda = 0$ while the latter uses $\lambda = 2$. The matrix is constructed for $n = m = 4$ and thus has $mn^2 = 64$ elements per dimension. We visualize the matrices for a range of completeness C and swap-ratio σ values.

D. Example Constraint Matrices and Sparsity Patterns

We have shown in the main paper that the permutation constraints can be formulated as a set of linear systems $\{\mathbf{A}_i\mathbf{q}_i = \mathbf{b} \triangleq \mathbf{1}\}$. We now show concrete examples in two and three dimensions of these constraint matrices:

1. $n = 2$:

$$\mathbf{A}_i = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}, \quad \mathbf{b}_i = \mathbf{1} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}. \tag{29}$$

2. $n = 3$:

$$\mathbf{A}_i = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}, \mathbf{b}_i = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}. \tag{30}$$

We further analyze how our quadratic constraint matrix looks like when these linear constraints are added. For different random experiments, we plot in Fig. 9 these $\mathbf{Q}$-matrices for different values of completeness C and swap ratio σ when $n = m = 4$. Note that this is a typical setting for our synthetic evaluations.

E. Minor Embeddings and Sampling

In the default mode, *Leap2* allocates couplings between qubits in the minor embeddings even if they are equal to zero. If zero couplings are explicitly avoided, the total number of physical qubits required to minor-embed a logical problem, along with the maximum required chain length, decreases, and the probability of measuring the optimal solution in single annealing is increasing. See Fig. 10 for the variant of Fig. 6 of the main paper, without allocated zero qubit couplings. We observe empirically that more compact minor embeddings —in contrast to what one could presume— do not allow to solve larger problem instances.

In Fig. 11, we show three exemplary minor embeddings on the Pegasus architecture of Advantage system 1.1 for problems of different sizes. While the number of logical qubits for the shown embeddings ranges from 18 (A) to 63 (C), the number of the corresponding physical qubits ranges from 49 (A) to 550 (C). Note that the shown visualizations include zero couplings, and more compact minor embeddings are possible (*cf.* Fig. 10). With the increasing problem size, we also see that maximum lengths of qubit chains l in the embeddings, which encode the same logical qubit, increase as well. While $l = 4$ for $n = 3, m = 4$, it increases to 10 and 13 for the combinations $n = 4, m = 4$ and $n = 3, m = 8$, respectively. For the same problems, AQC of the previous generation 2000Q requires 108, 782 or 1378 physical qubits on average over 50 runs, respectively.

Selecting qubit biases. We empirically find that the qubit biases have to be set differently compared to as derived in (15),

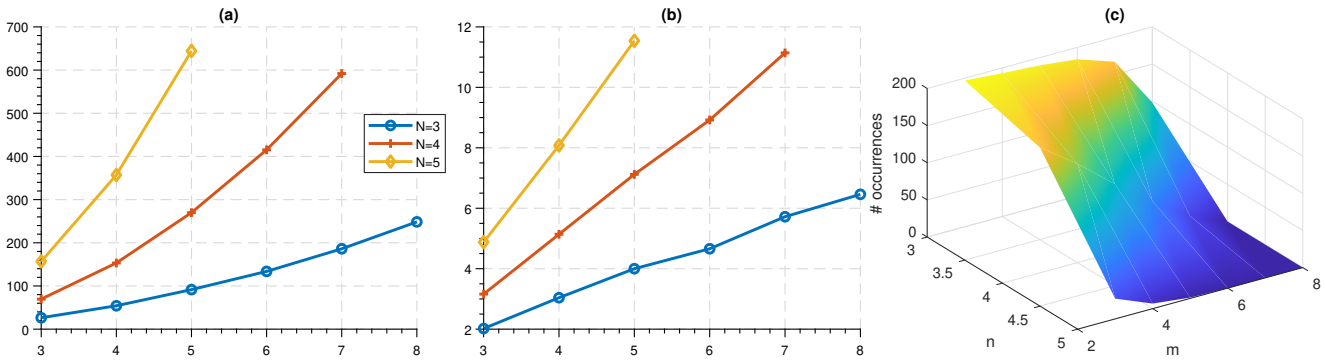

Figure 10. This figure supplements Fig. 6 of the main paper with the difference that the zero qubit couplings are excluded. For different n and an increasing number of views m, (**a**) plots the number of qubits required to map a problem; and (**b**) at $\chi = 3.0$, shows the required maximum chain length required to embed the problem on Advantage 1.1. (**c**) plots the average number of measured optimal solutions in 200 samples, for different pairs of n and m (averaged over 50 repetitions).

Table 3. The table summarises the average number (and the corresponding standard deviation) of measured optimal solutions out of 200, for $n = 4, m = 4$ and varying chain strength χ, strictly as derived in Prop. 2 (first row) and with adjusted weights as described in Sec. E, *i.e.,* $-\sqrt{|\mathbf{s}_k|}$ (second row). Each number is reported for 50 repetitions of the experiment.

	$\xi = 1$	$\xi = 2$	$\xi = 3$	$\xi = 4$	$\xi = 5$
derived	0.02 ± 0.14	1.26 ± 2.1	1.38 ± 1.75	0.08 ± 0.27	0.0 ± 0.0
adjusted	0.1 ± 0.30	9.9 ± 10.08	17.3 ± 14.0	6.1 ± 7.77	1.42 ± 2.72

see Tab. 3 for the summary of our experiment. Thus, $-\sqrt{|\mathbf{s}_k|}$ worked well (instead of the derived $-\operatorname{diag}(\mathbf{Q})_k - \mathbf{s}_k$)[2] in combination with the selected chain strengths and λ, see Sec. 5.1 for details on parameter selection. The possible reasons for that lie on the hardware side. First, the qubit biases and couplings are converted to magnetic fields acting on qubits, *i.e.,* the weight encoding and annealing are analogue physical processes. The embedded problem is predominantly defined in terms of qubit couplings ($\frac{k^2-k}{2}$ logical couplings at most in contrast to k logical biases at most for a problem with k logical qubits). Moreover, magnetic fields are imposed to keep chains of physical qubits intact. Last but not least, the range of real (floating-point) values which can be mapped to the native Ising format of D-Wave is limited (currently, it is $[-2; 2]$ for biases and $[-1; 1]$ for couplings on 2000Q and system 1.1), and analytically derived biases and couplings have to be scaled down to the supported ranges. These factors can lead to offsets in qubit biases which are difficult to predict theoretically, as QUBO formulations are often derived without consideration of the minor embedding to a real AQC.

On sampling. Note that, theoretically the samples coming from the quantum annealer can not be directly interpreted as the samples from the induced Bolzmann distribution characterized by β (*e.g.,* lower values of β result in samples less constrained to the lowest energy states), as the annealer samples a modified posterior. Optionally, one could steer the samples towards the local minima using a CPU-based descent algorithm. However, we found that in practice this hack, also suggested by D-Wave, does not work well for capturing a diverse set of modes. Nevertheless, as we show in Fig. 4 (in the main paper) using the original samples as alternative plausible solutions could still boost the accuracy albeit incrementally. While a true posterior adjustment is not yet implemented in D-Wave, we foresee that upcoming years would witness a leap on these fronts.

F. Further Details on Synthetic Experiments

We visualize in Fig. 12 some examples from our random synthetic dataset. While the common case of $n = 4$ and $m = 4$ can be visualized this way, the same does not hold true for other n, m combinations. That is why the points on the grid only assist the visualization, and are not used in practice. The important cues are the correspondences denoting permutations, whose rows might be randomly swapped to inject noise.

[2]Our experiments suggest that there is a broader range of smaller biases which work as well as $-\sqrt{|\mathbf{s}_k|}$. In several our test cases, $-0.5\mathbf{s}_k$ were leading to similar solution distributions. Further study is required on the differences between the derived weights and the weights which should be set in practice.

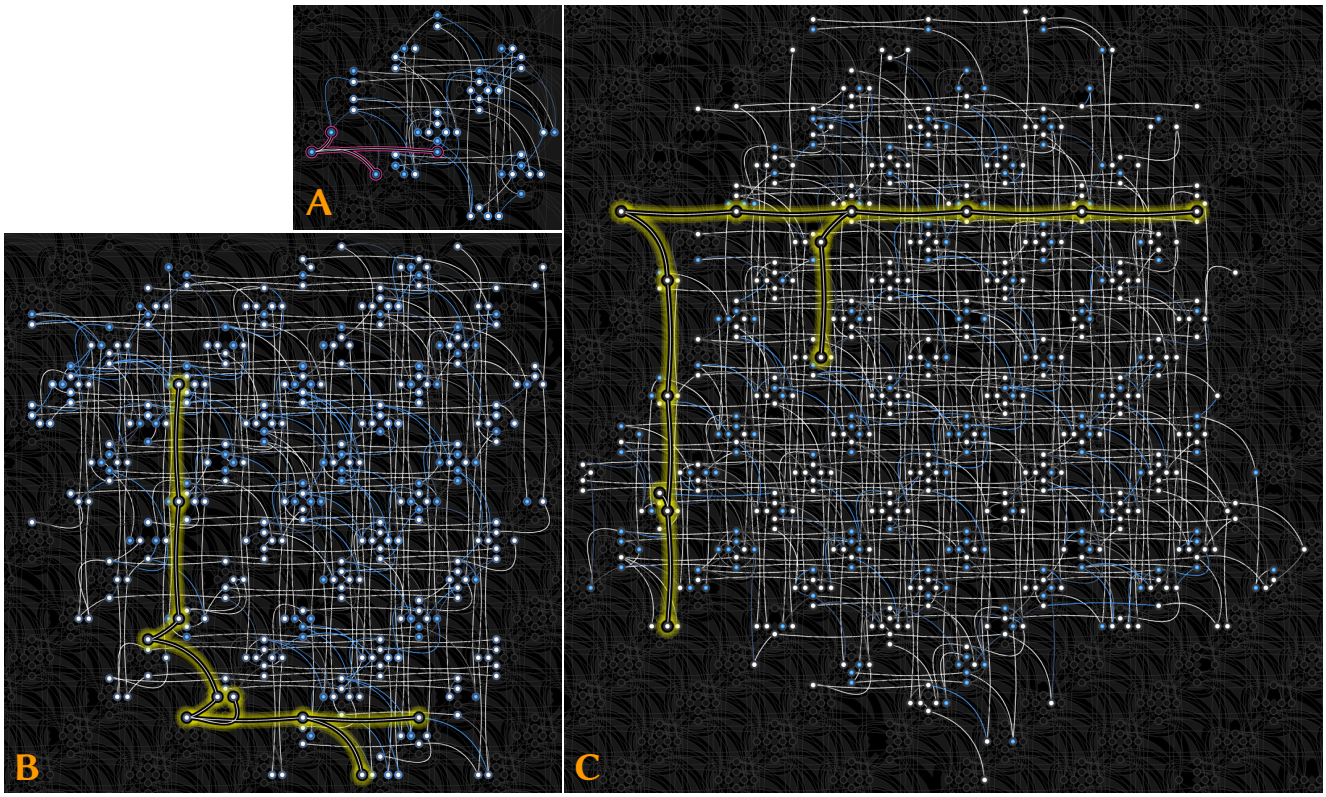

Figure 11. Exemplary minor embeddings in the experiments with $n = 3, m = 3$ (A, $n_{ph} = 49$), $n = 4, m = 4$ (B, $n_{ph} = 341$) and $n = 3, m = 8$ (C, $n_{ph} = 550$). In each case, we highlight qubit chains of the maximum chain length in the embedding, either in magenta (A, no warnings) or yellow (B and C, chain length warnings). Note that the shown minor embeddings include zero qubit couplings.

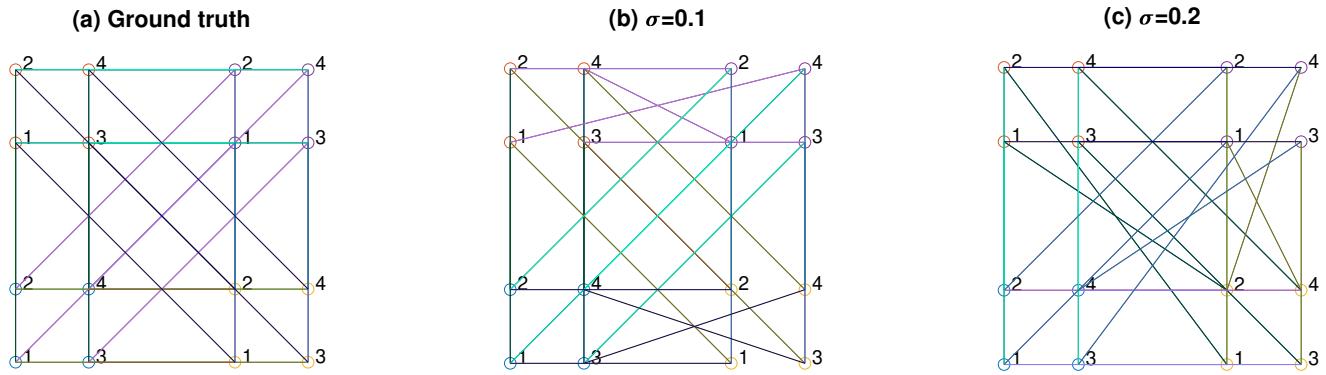

Figure 12. Samples from our fully connected synthetic dataset for different values of swap ratio σ. For sparser graphs, where completeness is less than one, it is possible to imagine edges being dropped from these graphs. In the figure, each group (indicated using differently colored points) corresponds to a *view* and each inter-group correspondence corresponds to a permutation that we optimize for. Note that the points are drawn as a grid to ease visual perception. Neither our algorithm nor the state-of-the-art methods we compare would use this information.

G. Evaluations on the Real Dataset Based on the Willow Object Classes [29]

Dataset description. Willow dataset is composed of 40 RGB images of four object classes: *duck*, *car*, *winebottle*, *motorbike* and *face*. These images are extracted either from *Caltech-256* or *PASCAL VOC* datasets. The *in-the-wild* nature of these images as shown in Figs. 13 and 15 makes simple template matching ill-suited for correspondence estimation. This is why, for our evaluations—as explained in the main paper—we followed the same design as Wang *et al.* [102] and benefited from the state-of-the-art-deep learning approaches. The authors of [29] have manually selected ten distinctive points on the target

object (category). These keypoints are annotated consistently across all the instances in each of the categories as we show in Fig. 13. To this end, a semi-automatic graph matching has been used as explained in [29]. In our paper, we have further extracted 35 subsets of this dataset composed of only the first four annotations ($n = 4$, as shown in Fig. 13) and four images which were consecutive ($m = 4$). Thus, we have 35 subset graphs. Due to our automated initial graph matching process, while the keypoint locations are always correct, the permutation matrices $\mathbf{P}_{ij}$ are contaminated with real noise (Fig. 15).

Figure 13. A random example from the *cars* class of Willow Object Classes dataset [29].

Extended plots of the evaluations. In the paper we have reported the average accuracy over all the images in our modified-willow dataset. However, it is also of interest to see how our algorithm performs on individual subsets. We plot in Fig. 14 the performance of all the approaches under test, over all the individual subgraphs. It is visible that all the methods perform similarly. While our algorithm has an overall advantage, it can fail on certain examples, despite the quest for the global minimum. This explains the slightly lower accuracy with respect to the exhaustive solution.

Evaluation metrics. In the main paper, we mention that we report the number of bits correctly detected. This so-called *accuracy* measure between the estimated permutations $\mathbf{X}$ and their ground truth $\bar{\mathbf{X}}$ is formally defined as:

$$\epsilon(\mathbf{X}, \bar{\mathbf{X}}) = 1 - \frac{1}{mn^2} \sum_{i=1}^{m} |\text{vec}(\mathbf{X}_i \oplus \bar{\mathbf{X}}_i)|_1, \tag{31}$$

where $\oplus$ is the *exclusive-or* operand. Intuitively, this is a Hamming similarity derived from the Hamming distance in which an error is made whenever two bits differ.

H. Detailed Analysis of Several Recent Works on Quantum Computer Vision

Algorithm	Problem	Processor	# qubits	total QPU time
QUBO Suppression [70]	non-maximum suppression in human tracking	2X	1000	n/a
QA [45, 46]	transformation estimation and point set alignment	2000Q	2048	60 sec.*
QGM [89]	graph matching (two graphs, up to four points)	2000Q	2048	$2 - 2.5$ min.
QuantumSync (ours)	permutation synchronization (multiple views, multiple points)	Adv. 1.1	5436	> 15 min.

Table 4. Overview of several recent quantum computer vision methods published at computer vision conferences and our *QuantumSync*. Note that the right-most column reports the overall experimental QPU runtime in the evaluation of the methods. "*": QA has been recently tested on D-Wave 2000Q; the results are reported in the supplementary document [46].

In this section, we analyze several recent works on quantum computer vision [45, 70, 89] and *QuantumSync*. We summarize the problems, the D-Wave processors and the experimental QPU time used in the experiments on real quantum hardware in Tab. 4. The QUBO suppression approach [70] solves an existing QUBO formulation of non-maximum suppression on D-Wave 2X with 1000 qubits. The experimental results show improvements in solving the target combinatorial optimization problem on quantum hardware, and the authors conclude that the age of quantum computing for human tracking has arrived.

Quantum Approach (QA) to correspondence problems on point sets, inspired by the altered gravitational model for point set alignment [47], was proposed in [45], where the cases with and without known correspondences between the points have been considered. In QA, final rigid transformations are approximated as a linear combination of basis elements which, in the general case, allows representing affine transforms. Thus, the method can be extended to affine transforms in a straightforward way. QA was confirmed on D-Wave 2000Q as reported in the supplementary material accompanying the paper [46]. QA can align two point sets at a time. Even though the dimensionality of the resulting matrix of couplings and biases does not

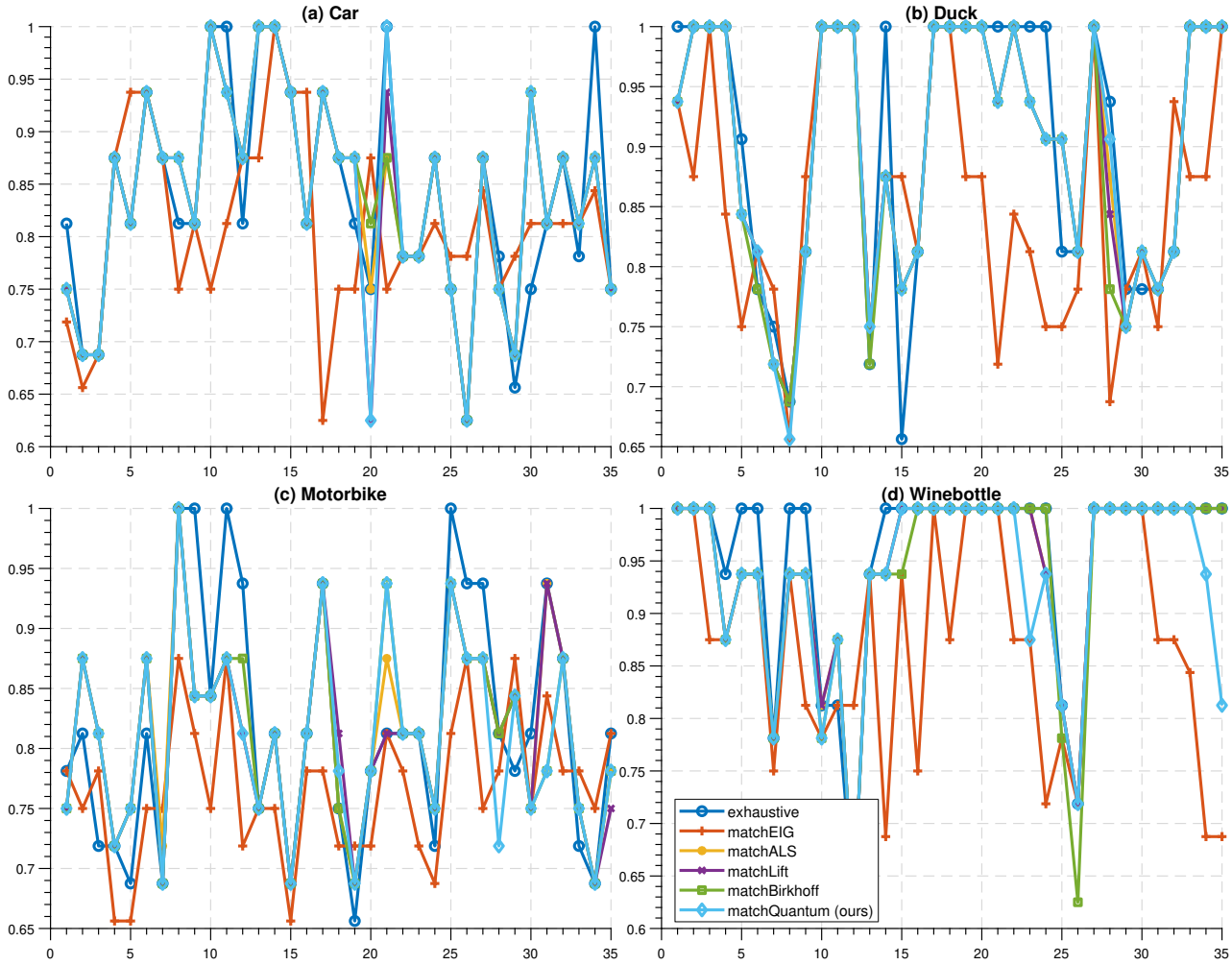

Figure 14. Detailed evaluations over all the subsets on the Willow Object Classes dataset [29].

depend on the number of points (the size of this matrix depends on the cardinality of the basis elements set), the super-linear complexity to prepare the state for large problems on classical hardware can result in large runtimes.

The method of Benkner *et al.* [89] for quantum graph matching (QGM) is concurrent to our work. While they propose a similar mechanism to impose permutation matrix constraints, there are multiple fundamental differences. First, QGM is designed for operation on two graphs, whereas we propose a permutation (map) synchronization algorithm for multiple views/scans with multiple points each. In our problem setting, initial (noisy) estimates of the pairwise permutations are given. Second, we successfully confirm our method for up to nine views with three points each, or seven views with four points each, whereas in [89], only the 3×3 case has been successfully solved on a real AQC. Another distinguishing aspect of our work is that we achieve the probability of over 90% to measure a correct optimal solution for the permutation synchronization problem for the 3×3 case, whereas [89] this probability (for a different problem though) only slightly exceeds the probability of randomly guessing a correct 3×3 permutation (*i.e.,* $\approx 16.7\%$). While the experiments in [89] are performed on $2000Q$ as opposed to the Advantage system 1.1 (the latest generation of D-Wave AQC) we use, the difference in the probabilities is unlikely to stem from the differences between the hardware architectures. We have tested our algorithm on $2000Q$ as well and even observed slightly higher probabilities for the 3×3 case, though larger problems either resulted in lower probabilities compared to Advantage system 1.1 or could not be embedded on the $2000Q$ QPU and solved on it at all.

Figure 15. Initial matches on random examples from Willow Object Classes dataset [29].