Abstract

Modern adiabatic quantum computers (AQC) are already used to solve difficult combinatorial optimisation problems in various domains of science. Currently, only a few applications of AQC in computer vision have been demonstrated. We review modern AQC and derive the first algorithm for transformation estimation and point set alignment suitable for AQC. Our algorithm has a subquadratic computational complexity of state preparation. We perform a systematic experimental analysis of the proposed approach and show several examples of successful point set alignment by simulated sampling. With this paper, we hope to boost the research on AQC for computer vision.

1. Introduction

Since their proposal in the early eighties [8, 43, 27], quantum computers have attracted much attention of physicists and computer scientists. Impressive advances both in quantum computing hardware and algorithms have been demonstrated over the last thirty years [40, 30, 61, 58, 42, 19, 25, 49, 65, 48]. Quantum computers are not universally faster than conventional machines, but they can natively execute algorithms relying on quantum parallelism, i.e., the ability to perform operations on exponentially many superimposed memory states simultaneously [59].

To harness the advantages, carefully designed algorithms are required. Nowadays, the motivation to take advantage of quantum effects in computing is also facilitated by the classical computing paradigm approaching its limits, since the quantum effects are becoming non-neglectable while manufacturing and using conventional CPUs. As a result, alternative paradigms such as massively parallel computing devices have been brought into being.

While universal gate quantum computer technology has not yet reached the maturity, modern adiabatic quantum annealers (AQA) are already capable of solving difficult real-world combinatorial optimisation problems [15, 14, 23, 49]. The primary difference of universal gate quantum computing and AQA is that the latter can address objectives formulated as quadratic unconstrained binary optimisation problems (QUBOP) defined as

\[
\arg \min_{q \in B^n} q^T P q,
\]

where \( q \) is a set of \( n \) binary variables, and \( P \) is a symmetric matrix of weights between the variables. The operational principle of AQA is grounded on the adiabatic theorem of quantum mechanics [17] which states that

if a quantum-mechanical system is in the ground state of a time-dependent Hamiltonian and parameters of this Hamiltonian are changing gradually enough, the system will continue to remain in the ground state during the evolution (see Table 1 for quantum notions).

In their seminal paper, Farhi et al. [26] have shown that the adiabatic principle (2) can be used for solving \( \mathsf{NP} \)-complete optimisation problems and laid the foundation for adiabatic quantum computing. Several years later, Aharonov et al. [2] theoretically showed the equivalence between classical quantum computing and quantum annealing models. As of 2019, general-purpose quantum computers accessible for research purposes and applications contain up to 20 qubits [19]. In contrast, the latest quantum annealers support up to \( 2^{10} \) qubits [21]\(^1\). Nevertheless, due to design

\(^1\)the amount of logical qubits which can be used on this system is an order of magnitude lower, since most qubits are used for error correction

Figure 1: Different 2D point sets — fish [47], qubit, kanji and composer — aligned with our QA approach. For every pair of point sets, the initial misalignment is shown on the left, and the registration is shown on the right. QA is the first transformation estimation and point set alignment method which can be potentially executed on adiabatic quantum computers.
and practical restrictions, quantum algorithms for the gates model such as Shor’s prime number factorisation algorithm [61] cannot be implemented on current quantum annealers.

**Motivation and Contributions.** Considering recent successful applications of AQA in several fields of computational science [42, 49, 65], we are motivated to investigate how useful AQA can be for computer vision and which problems can be potentially solved on the new hardware. The vast majority of available materials about quantum annealers are either oriented to physicists or lack technical details and clarity. *Our goal is to fill this gap, introduce the reader into the modern AQA and provide all notions and background to understand, analyse, simulate and design quantum algorithms for computer vision which can potentially run on modern AQA, as well as interpret the results.*

We consider correspondence problems on point sets which have various applications in computer vision. They consist in finding an optimal rigid transformation between inputs [34, 13, 47, 66]. While transformation estimation assumes known matches, point set alignment is more general and targets, in addition, the recovery of correspondences. We consider two inputs, i.e., a fixed reference point set and a template undergoing a rigid transformation. Thus, our goal is to design a quantum approach for point set alignment which can potentially run on AQA and show that it offers advantages compared to the classical counterparts.

Therefore, we adapt the recent progress in rigid point set alignment which allows to formulate a globally multiply-linked energy functional which does not require any intermediate correspondence updates [29]. In gravitational approach (GA) [29], the optimal alignment is achieved when the gravitational potential energy (GPE) of the system with two interacting particle swarms is locally minimal. Proceeding from GA, we build the weight matrix $P$ for the associated QUBOP (1) which is unalterably valid in the course of the optimisation. Along with that, we are targeting at a method which is implementable on classical hardware and can solve real-world problems. *cf. Fig. 1.* To summarise, the main contributions of this paper are:

- The first self-contained and detailed introduction into modern quantum annealers for computer vision problems, including notions from quantum physics and computing (Sec. 2), modern adiabatic quantum annealers (Sec. 3) including D-WAVE (Sec. 3.2), and previous and related works from quantum computing (Sec. 4).

- The first approach to transformation estimation (Sec. 5) and point set alignment (Sec. 6) which can potentially run on the upcoming quantum annealers (Sec. 6.2).

- Experimental analysis of the proposed method in a simulated environment on several datasets (Sec. 7).

<table>
<thead>
<tr>
<th>quantum notion</th>
<th>classical counterpart</th>
</tr>
</thead>
<tbody>
<tr>
<td>qubit (states $0$ and $1$)</td>
<td>bit (states 0 and 1)</td>
</tr>
<tr>
<td>(time-dependent) Hamiltonian eigenstate</td>
<td>energy functional</td>
</tr>
<tr>
<td>ground state</td>
<td>some energy state</td>
</tr>
<tr>
<td>quantum system evolution</td>
<td>globally optimal energy state</td>
</tr>
<tr>
<td>quantum annealing [26]</td>
<td>optimisation process</td>
</tr>
<tr>
<td>spectral gap</td>
<td>simulated annealing [39]</td>
</tr>
<tr>
<td></td>
<td>difference in neighbouring energies</td>
</tr>
</tbody>
</table>

**Table 1:** Quantum notions and their counterparts in computer vision.

2. **Preliminaries, Definitions and Notations**

In this section, we introduce the reader into the basics of quantum computing. See Table 1 for a lookup of notions specific to AQA which have counterparts and interpretation in the classical optimisation theory for computer vision.

**Qubit.** Quantum computing encompasses tasks which can be performed on quantum-mechanical systems [53]. Quantum superposition and entanglement are two forms of parallelism evidenced in quantum computers. A qubit is a quantum-mechanical equivalent of a classical bit. A qubit $|\phi\rangle$ can be in the state $|0\rangle$, $|1\rangle$ or an arbitrary superposition of both states denoted by $|\phi\rangle = \alpha|0\rangle + \beta|1\rangle$, where $\alpha$ and $\beta$ are the (generally, complex) probability amplitudes satisfying $|\alpha|^2 + |\beta|^2 = 1$. In quantum computing, the state $|0\rangle + |1\rangle$ denoted by $|\pm\rangle$ is often used for initialisation of a qubit register. The state of a qubit remains hidden during the whole computation and is only revealed when measured. If qubits are entangled, measuring one of them influences the measurement outcome of the other one [59]. During the measurement, the qubit’s state irreversibly collapses to one of the basis states $|0\rangle$ or $|1\rangle$. For an efficient physical realisation of a qubit, very low temperatures are needed. Otherwise, thermal fluctuations will destroy the state and lead to arbitrary changes of the measured qubit state.

One possible physical implementation of a qubit is an electron which possesses a spin, i.e., its intrinsic magnetic moment [53, 63]. The spin of an electron can be manipulated and brought to the state spin down, spin up, or a superposition of both. A concrete experimentally realised scheme that uses this property is represented by an atom of phosphorus $^{31}\text{P}$ embedded into a $^{28}\text{Si}$ silicon lattice attached to a transistor [37, 46, 67]. The nucleus of $^{31}\text{P}$ has a positive charge compensated by electrons. The bundle of electrons in the transistor is filled up to the energetic level between the energy of spin-down and spin-up state of $^{31}\text{P}$. To change a state of a $^{31}\text{P}^{28}\text{Si}$ qubit, a microwave pulse of the frequency — which is equal to the resonance frequency of the atom — is applied to it. The new state $|\phi\rangle$ depends on the duration of the exposure. A transistor is used to measure a state of the $^{31}\text{P}^{28}\text{Si}$ qubit. If the extra electron of $^{31}\text{P}$ tunnels into the electron bundle, a positive charge is measured in the transistor indicating the spin-up state (e.g., $|1\rangle$).

Fig. 2-(a) visualises a qubit with a so-called Bloch sphere. Every qubit can be both in a superposition and en-
tangled with other qubits. Thus, quantum superposition is the property that calculations are performed on all possible inputs simultaneously which can result in exponential parallelism in the number of qubits. When entangled, states of qubits cannot be described independently from each other.

**Schrödinger Equation.** In the universal or gates model, transformations are described by a series of unitary transformations applied to qubits. This is a useful practical simplification while the evolution of every quantum-mechanical system can be described more precisely by continuous Schrödinger equation, which in common notation reads:

$$-i \frac{d}{dt} |\phi(t)\rangle = \hat{H}(t) |\phi(t)\rangle.$$  \hspace{1cm} (3)

For the simplicity, we denote here by $|\phi\rangle$ a state of $n$ qubits at time $t$, and $\hat{H}(t)$ is a Hamiltonian which is, in this case, a $2^n \times 2^n$ Hermitian matrix. Thus, a discrete time evolution of the quantum system is given by a unitary transformation.

**Hamiltonian.** Hamiltonian $\hat{H}$ is an energy operator of a system of $n$ qubits. It defines the energy spectrum of a system or, in our case, the space of all possible solutions. The ground state of the system is its lowest energy eigenstate. Finding a ground state of a Hamiltonian is equivalent to finding an optimal solution to the problem. The expectation value of Hamiltonian $\langle \hat{H} \rangle$ provides an instantaneous energy of a given qubit configuration. In correspondence problems, $\langle \hat{H} \rangle$ is a quantitative characteristic of point set alignment. We denote by $\Delta(\hat{H})$ the spectral gap of $\hat{H}$, i.e., the difference between the energies of the ground state and the second lowest eigenstate. The spectral gap influences the annealing rate and is considerable for algorithm design and evaluation in quantum annealing.

**Pauli Matrices.** An arbitrary Hamiltonian of a $n$-qubit system can be expressed by a linear combination of tensor products of Pauli matrices denoted by:

$$\sigma^x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma^y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \sigma^z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \hspace{1cm} (4)$$

The Pauli matrices are $2 \times 2$ Hermitian and unitary. Together with the identity $I_2$, they form a basis for $\mathbb{C}^{2 \times 2}$. $\sigma^x$ flips the probabilities to measure $|0\rangle$ and $|1\rangle$, whereas $\sigma^z|0\rangle = |0\rangle$, and $\sigma^z|1\rangle = -|1\rangle$.

**Pseudo-Boolean Functions.** A pseudo-boolean function is a real vector-valued function of $n$ boolean variables denoted by $x$ of the form $F(x) : B^n \rightarrow \mathbb{R}^M$, where $M$ is the number of real-valued outputs.

**Quantum Annealing.** Quantum annealing is a heuristic combinatorial optimisation method for finding global optima which relies on quantum effects (superposition, entanglement and tunnelling) \[45, 39\]. In particular, it is used to find a ground state of an Ising Hamiltonian \[35, 57\], which encodes the target computational problem, see Fig. 2-(b).

Quantum annealing is the quantum counterpart of simulated annealing \[45, 39\]. Starting from the superposition state $|+\rangle^\otimes n$ (this is a shorthand notation for $n$ qubits in the state $|+\rangle$ (9)), the system evolves according to (3) under an external time-dependent magnetic field (a transverse field). When the external field is faded away, the system reaches the ground state of an Ising model \[35\]. According to (2), if an external magnetic field is changing gradually enough, the system remains near the ground state with high probability throughout the optimisation. Quantum annealing systems taking advantage of (2) are called adiabatic quantum computers (AQC). QUBOP is the most common problem form which can be mapped to current realisations of AQC.

### 3. Modern Adiabatic Quantum Computation

Adiabatic quantum computation is a form of quantum annealing which relies on the adiabatic theorem of quantum mechanics \[2\] \[17\]. Starting from a ground state of an initial default Hamiltonian $\hat{H}_I$, an AQC system adiabatically evolves into the ground state of a problem Hamiltonian $\hat{H}_P$ which encodes a solution to a problem \[26\]. In the case of adiabatic quantum annealing (AQA), the problem Hamiltonian $\hat{H}_P$ is given by the Ising model \[35\] :

$$\hat{H}_P = \sum_{j \in V} h_j \sigma_j^z + \sum_{(j, k) \in E_P} J_{j,k} \sigma_j^z \otimes \sigma_k^z,$$  \hspace{1cm} (5)

where the Kronecker product $\otimes$, $h_j$ denoting exterior local magnetic fields and $J_{i,j}$ standing for the pairwise connections between the particles. $V$ is a set of particles, and $E_P$ is a set of edges (intra-particle links) of the graph. Eq. (5) is written in a notation common in physics. The first term of (5) on the right side in the explicit notation reads

$$\hat{H}_{p}^{j \in V} = \begin{pmatrix} \sigma^z \otimes I \otimes \ldots \otimes I \\ I \otimes \sigma^z \otimes \ldots \otimes I \\ I \otimes I \otimes \ldots \otimes \sigma^z \end{pmatrix}_{2^n \times 2^{n-1}}$$

To simplify the notation, the first term of (5) on the right side in the explicit notation reads

$$\hat{H}_{p}^{j \in V} = \begin{pmatrix} h_1 I_{2^n \times 2^n} \\ h_2 I_{2^n \times 2^n} \\ \vdots \\ h_n I_{2^n \times 2^n} \end{pmatrix}_{n2^n \times 2^n}$$

\[6\]
where $\mathbf{I}$ without a subscript is a $2 \times 2$ identity matrix. The second term $\mathcal{H}_P^{(j,k)}$ of (5) can be expressed in a similar manner, involving pairs of $\sigma^z$ in the tensor product depending on the connectivity of the lattice.

Theoretically, each particle can interact with any other particle from the whole set of qubits. In practice, the couplings are restricted to local neighbourhoods (see Sec. 3.2). Thus, (5) describes a system of $N$ interacting spin-$\frac{1}{2}$ particles under the influence of distributed magnetic forces, and in the expanded form, $\mathcal{H}_P$ is a $2^n \times 2^n$ matrix. Finding a ground state of an Ising model is an $NP$-hard problem [7]. In the ground state, a spin configuration of all particles minimises Ising energy $E_{\text{Ising}}$ given by:

$$E_{\text{Ising}} = \sum_i h_i s_i + \sum_{i,j} J_{i,j} s_i s_j,$$

where $s_i \in \{1, -1\}$ denotes two possible spin measurement outcomes of a spin-$\frac{1}{2}$ particle.

### 3.1. Quantum System Evolution

Solving an $NP$-hard problems such as QUBOP on a classical computer requires exponential time in the size of the input. The main idea of the AQC is that a QUBOP (1) can be mapped to the Ising model (5) and optimised by allowing the system to evolve according to the adiabatic principle (2). Once annealing is finished, the qubit register will represent the solution to the programmed problem with a high probability [26] (cf. App. A on the annealing rate criterion). The initial Hamiltonian of the system is always initialised in the state

$$\mathcal{H}_I = -\sum_{j \in V} B_x \sigma^x_j,$$

where $B_x > 0$ stands for a magnetic field pointing in the $x$ direction. The ground state of (8) is a symmetrised superposition with equal normalised probability amplitudes for the states $|0\rangle$ and $|1\rangle$ for all qubits, i.e.,

$$|\rangle = \left(\frac{|0\rangle + |1\rangle}{\sqrt{2^n}}\right)^\otimes n.$$

This initial state (9) is comparably easy to construct by radiating a microwave of the same duration and wavelength to all qubits. In mathematical terms, (9) is obtained by applying a Hadamard transform $H = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$ to $n|0\rangle$ qubits.

The lowest energy $E_{\text{GS}} = -nB_x$ of (8) is achieved when all qubits are in the system point in the anti-parallel direction of the magnetic field, so that $\sigma^x_j|s_j\rangle = |s_j\rangle$. During AQC, the initial Hamiltonian $\mathcal{H}_I$ is evolving into the problem Hamiltonian $\mathcal{H}_P$, with a high probability of reaching the ground state of $\mathcal{H}_P$ [26]. The interpolation between the Hamiltonians can be written as

$$\mathcal{H} = [1-s] \mathcal{H}_I + s \mathcal{H}_P,$$ with $s \in [0; 1]$ being the time in relative units from the start of annealing at $s = 0$ until reaching the ground state of $\mathcal{H}_P$ at $s = 1$. The problem Hamiltonian and the final state of the system depend on the objective function $f(x)$ or the matrix of weights between the qubits $P$ in (1). After the annealing is accomplished, the state of each qubit is measured, and the result corresponds to the solution of the programmed problem with a high probability. At this stage, the states of all binary variables are classical, not quantum anymore.

To remain in the ground state during the system evolution, the annealing rate has to be carefully chosen. The condition of adiabaticity (2) is derived from the time-dependent perturbation theory of quantum systems. It is achieved when the average energy pumped into the system per time interval $T$ is smaller than the minimal energy difference between the ground state and the first excited state. This statement was quantified in [4] which generalises the original adiabatic theorem [17] for periodic driving, see App. A.

### 3.2. Quantum Annealer D-WAVE

D-WAVE relies on the adiabatic criterion in its specified form and currently supports up to ≈2000 qubits [22]. It reflects the state-of-the-art in physical realisation of quantum processors. It is relatively inexpensive to bring the system in the superposition state, and every computation on D-WAVE starts with the problem-independent $\mathcal{H}_I$ (8). Qubits can interact with a restricted number of other qubits, and it is possible to define qubit equality and entanglement constraints [22]. Possible interactions can be seen from the chimera graph which schematically depicts the layout of the quantum processor [22, 16]. At the same time, the physically realised connectivity can model QUBOP with arbitrary connectivities through an internal conversion [16]. The drawback is that in the worst case, a quadratic increase in the number of variables is required. A fully connected graph of layers with $N$ qubits would require $N^2$ qubits for processing. Some QUBOP cannot be mapped to the chimera graph, and some problems can be mapped in multiple ways [55].

### 4. Previous and Related Work

**Universal Quantum Computers.** The paradigm of the universal quantum computer originates in the attempts to gain control over individual quantum systems in the early eighties [62, 53]. Later, extending the control to multiple quantum systems has attracted the interest of physicists, promising to facilitate discoveries in quantum physics [53]. By that time, it was noticed that simulating a quantum-mechanical system on a classical computer requires exponential time in the number of simulated elements [43, 27]. “Can you do it \(^2\) with a new kind of computer – a quantum computer?” [27] is a famous quote by R. Feynman which has
triggered research on quantum computers in the subsequent years. The so-called no-cloning theorem [56, 64] belongs to the first discoveries strongly influenced quantum information theory and quantum computations. Nowadays, quantum computers can be used not only to fulfil their primary goal, i.e., to simulate quantum-mechanical systems for different branches of science, but also to solve other computational problems — such as balanced function decision problem [24], quantum Turing machines for complexity analysis [12], prime number factorisation and discrete logarithms [61], database search [30], graph matching [3], data classification [58] and principal component analysis [42] — faster than on classical machines. The related field of quantum communication and quantum key distribution has already found broad practical use nowadays [10, 9, 60].

**Classical Methods using Quantum Analogies.** Quantum-mechanical effects inspired multiple techniques for conventional computers including variants of genetic and evolutionary algorithms [31, 32], non-rigid mesh analysis [5] and image segmentation [6], among others.

**Quantum Annealers in Computer Vision.** Only a few theoretical results and applications of AQC to image processing, machine learning and computer vision are known. Neven et al. [51] have shown how image recognition can be formulated as QUBOP. Image classification on 12 × 12 images with AQC was addressed in [52]. The method of O’Malley et al. can learn facial features and reproduce facial image collections [54]. Boyda et al. [18] propose AQC approach to detect areas with trees from aerial images. Several methods target classification, dimensionality reduction and training of deep neural networks [50, 38, 1]. Not all theoretical findings of these works are possible to test on the real AQC hardware yet. Nonetheless, we believe that it is essential to explore the theory and highlight the advantages of the upcoming hardware for computer vision tasks.

### 5. Quantum Transformation Estimation

In this section, we introduce our solution to transformation estimation. The inputs are a reference point set \([x\_n] \in X \in \mathbb{R}^{D \times N}\) and a template point set \([y\_n] \in Y \in \mathbb{R}^{D \times N}\), \(n \in \{1, \ldots, N\}\). \(N\) is the number of points in both point sets and \(D\) is the dimensionality of the points. We assume that translation is resolved, the centroids of the point sets coincide, and points are in correspondence. In Secs. 5.1 and 5.2, we discuss the 2D and 3D cases, respectively. In both cases, we show how transformation estimation can be mapped to (1) and solved on AQC.

#### 5.1. Transformation Estimation in 2D

To obtain an advantage in solving transformation estimation on a quantum annealer we should avoid uniform sampling of rotations applied to \(Y\). Elements of the rotation group are non-commutative, and it is not possible to formulate multiplication of basis rotations as QUBOP. Instead, we propose to assemble the transformation matrix as a linear combination of basis elements.

In the case of rotation, an assembly of a transformation matrix by addition is not as straightforward. Recall that for any rotation matrix, \(R^{-1} = R^T\). Rotation in 2D consists of four elements, i.e., \(R = \begin{pmatrix} r_{1,2} & r_{2,2} \\ r_{2,1} & r_{1,2} \end{pmatrix}\). Additively, we can create a basis for all possible values of \(R\) and encode the influence of the additive elements as binary variables.

Consider instead the power series of the rotation matrix in 2D. Every such matrix has a corresponding skew-symmetric matrix of the form

\[
S = \theta \mathbf{M}, \quad \mathbf{M} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix},
\]

with a real number \(\theta\). According to the Cayley-Hamilton theorem, \(S^2 + \theta^2 \mathbf{I} = 0\) which leads to the following exponential map for \(R\) with power series:

\[
R = \exp(S) = \cos(\theta) \mathbf{I} + \left(\frac{\sin(\theta)}{\theta}\right) S = \cos(\theta) \mathbf{I} + \sin(\theta) \mathbf{M}.
\]

From (12) we see that \(R\) is composed of an identity weighted by \(\cos(\theta)\) and \(\mathbf{M}\) weighted by \(\sin(\theta)\). If the basis would resemble additive elements \(\mathbf{I}\) and \(\mathbf{M}\) of the exponential map, we can stronger constrain the assembly of \(R\). We see that \(r_{1,1}\) is entangled with \(r_{2,2}\), and \(r_{1,2}\) is entangled with \(r_{2,1}\). Eventually, we need fewer basis elements, the optimisation will finish faster and the method can be also implemented and tested on a classical computer. Thus, our basis \(Q = \{Q_k\}\) for \(R\) is a compound of \(K = 20\) elements:

\[
\{Q_k = q \mathbf{C} \in \mathbb{R}^{2 \times 2}, \forall q \in \{0.5, 0.2, 0.1, 0.1, 0.05\}, \forall \mathbf{C} \in \{\mathbf{I}, \mathbf{M}, -\mathbf{I}, -\mathbf{M}\}\}.
\]

We then multiply each template point \(y\_n\) with each of the \(Q_k\) basis elements and stack the result into the \(\Phi\) matrix:

\[
\Phi = \begin{bmatrix}
    x^1 & x^2 & \cdots & x^N \\
    -[\mathbf{Q}_1y_1]^T & -[\mathbf{Q}_2y_2]^T & \cdots & -[\mathbf{Q}_Ny_N]^T \\
    -[\mathbf{Q}_2y_1]^T & -[\mathbf{Q}_2y_2]^T & \cdots & -[\mathbf{Q}_Ny_N]^T \\
    \vdots & \vdots & \ddots & \vdots \\
    -[\mathbf{Q}_Ny_1]^T & -[\mathbf{Q}_Ny_2]^T & \cdots & -[\mathbf{Q}_Ny_N]^T
\end{bmatrix}.
\]

Next, we set the weight matrix in (1) as

\[
\mathbf{P} = \Phi \Phi^T,
\]

and the final QUBOP reads

\[
\arg \min_{\mathbf{q} \in \mathbb{R}^{2^T}} \mathbf{q}^T \Phi \Phi^T \mathbf{q}.
\]

In total, 21 qubits are required to resolve the transformation on AQC in 2D, with the first qubit of \(\mathbf{q}\) being fixed to \(|1\rangle\).
5.2. Transformation Estimation in 3D

In 3D, a skew-symmetric matrix can be represented as

\[
S = \theta M, \quad M = \begin{bmatrix}
    m_{1,1} & m_{1,2} & m_{1,3} \\
    m_{2,1} & m_{2,2} & m_{2,3} \\
    m_{3,1} & m_{3,2} & m_{3,3}
\end{bmatrix} = \begin{bmatrix}
    0 & a & b \\
    -a & 0 & c \\
    -b & -c & 0
\end{bmatrix},
\]

where \(\theta, a, b, c\) are real numbers, and \(a^2 + b^2 + c^2 = 1\). In the 3D case, the Cayley-Hamilton theorem states that \(-S - \theta^2 S = 0\). The exponential map for \(R\) in 3D with power series reads

\[
R = \exp(S) = I + \frac{(\sin \theta)}{\theta} S + \frac{(1 - \cos \theta)}{\theta^2} S^2 = I + \sin \theta M + (1 - \cos \theta) M^2.
\]

Next, \(M\) can be decomposed as follows:

\[
M = a \begin{bmatrix}
    0 & 1 & 0 \\
    -1 & 0 & 0 \\
    0 & 0 & 0
\end{bmatrix} + b \begin{bmatrix}
    0 & 0 & 1 \\
    0 & 0 & 0 \\
    -1 & 0 & 0
\end{bmatrix} + c \begin{bmatrix}
    0 & 0 & 0 \\
    0 & 0 & 1 \\
    0 & -1 & 0
\end{bmatrix}.
\]

Regarding \(M\), we see that

- \(\{m_{1,2}; m_{2,1}\}, \{m_{1,3}; m_{3,1}\}\) and \(\{m_{2,3}; m_{3,2}\}\) are mutually dependent or entangled,
- \(m_{i,j} \in [-1; 1]\),
- \(M = -M^T\), \(M_a = -M_a^T\), \(M_b = -M_b^T\) and \(M_c = -M_c^T\), i.e., they are anti-symmetric, and
- \(M^2 = \begin{bmatrix}
    v_{11} & d & e \\
    d & v_{22} & f \\
    e & f & v_{33}
\end{bmatrix}\) is symmetric negative semi-definite, with \(\{v_{11}, v_{22}, v_{33}\} \in \mathbb{R}^+, \{d, e, f\} \in \mathbb{R}\).

The basis for rotation in 3D is comprised of the identity matrix \(I\), \(M_a\), \(M_b\), \(M_c\) as well as the basis for \(M^2\):

\[
M_d = \begin{bmatrix}
    1 & 0 & 0 \\
    0 & 0 & 0 \\
    0 & 0 & 0
\end{bmatrix}, \quad M_e = \begin{bmatrix}
    0 & 0 & 1 \\
    0 & 0 & 0 \\
    1 & 0 & 0
\end{bmatrix}, \quad M_f = \begin{bmatrix}
    0 & 0 & 0 \\
    0 & 0 & 1 \\
    1 & 0 & 0
\end{bmatrix}.
\]

Thus, our basis \(Q^{3D} = \{Q_k^{3D}\}\) for \(R\) in 3D is a compound of \(K = 80\) elements:

\[
\{Q_k^{3D} = q \in \mathbb{R}^{3 \times 3}, \forall q \in \{0.5, 0.2, 0.1, 0.1, 0.05\},
\forall C \in \{I, -I, M_a, -M_b, M_b, -M_c, -M_c, -M_c, -M_c\},
M_{d}, -M_{d}, -M_{e}, -M_{f}, -M_{f}\}\}.
\]

The final QUBOP for the 3D case is obtained similarly to Eqs. (14)–(16), with \(q \in B^{31}\) (\(q_0\) remains fixed to \(1\)).

6. Quantum Point Set Registration

In point set registration, the input point sets are of different cardinalities, and correspondences between points are, generally, not known, i.e., \(\{x_m\} \in X \in \mathbb{R}^{D \times N}\) and \(\{y_m\} \in Y \in \mathbb{R}^{D \times M}\), \(m \in \{1, \ldots, M\}\). \(N\) and \(M\) are the numbers of points in the reference and template, respectively, while \(D\) is the point dimensionality. The objective of point set alignment is to recover rotation \(R = R^{-1} = R^T\), translation \(t\) aligning \(Y\) to \(X\). We assume that the translation is resolved in the pre-processing step by bringing the point set centroids into coincidence.

Point set alignment can be alternatingly solved on AQC by finding some point matches and estimating the transformation with the given correspondences in the ICP fashion [12]. This would result in a sequence of QUBOP of the form (16). To express alignment as a single QUBOP, we have to find an energy functional which is correspondence-free and which, when minimised in one shot on AQC, would result in an optimal alignment. The desired form of the energy functional has been recently shown in the literature [29].

6.1. Particle Dynamics Based Alignment

Barnes-Hut Rigid Gravitational Approach (BHRGA) [29] is a recent point set alignment method with a single energy functional which remains unchanged during the entire optimisation. BHRGA is a globally multiply-linked approach, i.e., all \(y_m\) interact with all \(x_n\). In [29], point sets are aligned by minimising the mutual gravitational potential energy (GPE) \(E\) of the corresponding system of particles in the force field induced by \(X\):

\[
E(R, t) = \sum_m \sum_n \mu_{y_m} \mu_{x_n} \|R y_m + t - x_n\|_2^2,
\]

where \(\mu_{y_m}\) and \(\mu_{x_n}\) denote masses of the template and reference points, respectively. With no imposed boundary conditions, particles are initialised with unit masses. In [29], (6.1) is optimised with the Levenberg-Marquardt algorithm [41, 44], and the optimum is achieved when the system’s GPE is locally minimal. Without acceleration by a \(2^D\)-tree, the method has quadratic complexity and involves all possible interactions between the template and reference points.

We can now derive a QUBOP in the similar fashion as in Sec. 5 for the transformation estimation. The difference to the former problem is that now we have to consider all possible point interactions since the numbers of points in \(X\) and \(Y\) are different and no matches are known in advance. Note, however, that the bases (13) and (21) allow for scaling. Thus, implicitly, we would optimise

\[
E(R, t) = \sum_m \sum_n \mu_{y_m} \mu_{x_n} \|R y_m s + t - x_n\|_2^2,
\]

where the scalar \(s\) is the scaling of the template. As proven in [28], allowing for scale in globally multiply-linked point
set alignment results in the shrinkage of the template to a single point with a very high probability. To remedy the problem, either prior correspondences can be used, or point interactions can be restricted to local vicinities [28]. In our QA, we opt for the second solution which allows to use the rotational bases (13) and (21) elaborated in Sec. 5. Eventually, the \( \Phi \in \mathbb{R}^{(K+1) \times (L+1) \times (L+1) + \ldots + (L+1) \times N} \) matrix encoding point interactions for point set alignment reads

\[
\Phi = [\Phi_1 \Phi_2 \ldots \Phi_N],
\]

with \( \Phi_n, n \in \{1, \ldots, N\} \), of the form

\[
\begin{bmatrix}
    x_1^n & \cdots & x_L^n \\
    -Q_1 y_1^n & \cdots & -Q_L y_1^n \\
    \vdots & \ddots & \vdots \\
    -Q_K y_1^n & \cdots & -Q_K y_L^n 
\end{bmatrix}.
\]

(25)

\( \Phi_1, \Phi_2 \) and \( \Phi_N \) encode point interactions between every \( x_n \) and corresponding \( L(n) \ll M \) points of the template denoted by superscripted \( y_1^n, y_2^n, \ldots, y_{L(n)}^n \). Note that \( y_1^n, y_2^n, \ldots, y_{L(n)}^n \) build \( N \) subsets of \( y_1, y_2, \ldots, y_M \) of different cardinalities \( L(n), \hat{L} \) on average. If different \( x_n \) interact with the same \( y_m \), the corresponding sub-columns \( [Q_1 y_m^1]T [Q_2 y_m^2]T \ldots [Q_K y_m^K]T \) of \( \Phi_n \) can be computed only once and reused. The final QUBOP for point set alignment with \( \Phi_n \) as in (25) reads

\[
\arg \min_{q \in \mathbb{S}^N} q^T \Phi^T \Phi q.
\]

(26)

In total, 21 and 81 qubits are required to align points sets on AQC in the 2D and 3D case, respectively. Both transformation estimation and point set alignment need the same number of qubits in the same dimensions, and the difference lies in the complexity to construct \( \Phi \). Note that if the same template has to be aligned to multiple references, the corresponding \( \Phi \) can be obtained by reusing \( [Q_1 y_m^1]T [Q_2 y_m^2]T \ldots [Q_K y_m^K]T \) (which has to be computed only once). The first qubit of \( q \) has to be fixed to \( |1\rangle \), since the first element of every column contains a reference point which has to be active during the entire optimisation.

### 6.2. Complexity to Prepare \( \Phi = \Phi \Phi^T \)

To prepare \( \Phi \), \( \mathcal{O}(KDN \xi) \) and \( \mathcal{O}(KDN \xi L \xi) \) operations are required for the transformation estimation and point set alignment, respectively. \( \xi \) denotes the number of operations for multiplying \( y_m \) with one element of the additive basis \( Q_\xi \). To obtain the final \( \Phi \), we need to transpose \( \Phi \) and multiply \( \Phi \) with \( \Phi^T \) which, in the worst case, takes \( \mathcal{O}(K^2DN) \) operations for the transformation estimation and \( \mathcal{O}(K^2DNL) \) operations for the point set alignment. There are also slightly faster algorithms for matrix multiplication compared to the naïve way [20].

<table>
<thead>
<tr>
<th>TE</th>
<th>10</th>
<th>20</th>
<th>30</th>
<th>40</th>
<th>50</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \epsilon_{2D} )</td>
<td>0.023</td>
<td>0.026</td>
<td>0.041</td>
<td>0.078</td>
<td>0.17</td>
</tr>
<tr>
<td>( \sigma_{2D} )</td>
<td>0.012</td>
<td>0.013</td>
<td>0.012</td>
<td>0.012</td>
<td>0.012</td>
</tr>
<tr>
<td>( \epsilon_{R} )</td>
<td>0.058</td>
<td>0.062</td>
<td>0.083</td>
<td>0.22</td>
<td>0.47</td>
</tr>
<tr>
<td>( \sigma_{R} )</td>
<td>0.041</td>
<td>0.044</td>
<td>0.041</td>
<td>0.036</td>
<td>0.031</td>
</tr>
</tbody>
</table>

Table 2: The accuracy of QA under random initial misalignments, for the transformation estimation (“TE”) and point set alignment \( (K > 1) \).

### 7. Experimental Evaluation

We perform various experiments with 2D point sets. The current generation of D-WAVE annealers does not support the precision of weights in \( P \) necessary for our method [22]. It is foreseeable that future generations of AQC will enable a higher accuracy for couplings. Thus, we implement and test QA with an AQC sampler on a conventional computer (Intel i7-6700K with 32GB RAM, no specialised hardware used). All quantitative tests are performed with 21 binary variables corresponding to the size of the Q basis in 2D.

We report two error metrics, i.e., the alignment error \( \epsilon_{2D} \) and the transformation discrepancy \( \epsilon_{R} \), together with their standard deviations denoted by \( \sigma_{2D} \) and \( \sigma_{R} \), respectively. The alignment error \( \epsilon_{2D} = \| \text{HS} \|_{\mathcal{H}_R} \) denoted the Hilbert-Schmidt norm measures how accurately the aligned shape coincides with the reference and requires ground truth correspondences. The transformation discrepancy is defined as \( \epsilon_{R} = \| I - RR^T \|_{\mathcal{H}_R} \) where \( R \) is the recovered rotation. It measures how closely the recovered transformation resembles a valid rigid transformation. The usage of two complementary metrics is necessary because a low \( \epsilon_{R} \) does not automatically imply an accurate registration. On the other hand, a low \( \epsilon_{2D} \) does not quantify how rigid the recovered transformation is.

#### Datasets and Proof of Concept

We use four datasets, i.e., fish [47], qubit, kanji and composer with cardinalities varying from 91 (fish) to 7676 (composer). Qualitative registration results are shown in Fig. 1. For point sets with up to a few thousand points, the simulation time \( T_P < 1 \) sec. For \( \sim 7.7k \), \( \tau_P \) grows to 20.178 sec (by a factor of \( \sim 10^4 \)). Simulation with \( n = 30 \) takes already \( \sim 2.5 \) days. More binary variables allow for more elements in the basis \( Q \) resulting in more accurate alignment. Note that even with 80 qubits, i.e., for problems with \( n = 80 \), annealing on AQC takes around 100 ms. A simulation with \( n = 80 \) is not possible on a conventional supercomputer in a reasonable time.

#### Initial Misalignment and Point Linking

We test how accurately our method recovers the transformation under the random angle of initial misalignment \( \theta \) and the different size of the point linking region. We generate 500 random transformations in the range \( \theta \in [0; 2\pi] \) of the fish dataset and resolve them with QA, for each \( K \in \{1, 10, 20, 30\} \). The results are summarised in Table 2. We see that \( \epsilon_{2D} \) cor-
the transformation assembly caused by interactions for one exemplary problem. In Fig. 4, we plot the sensitivity between the energy of the ground state and the second-lowest eigenstate. Each problem has an intrinsic and unique \( \Delta(\hat{H}) \). Even though a rigorous analysis of the spectral gap is out of the scope of this paper, we make several qualitative observations about the energy landscape of QA, the difference in the energy values and the corresponding registrations for one exemplary problem. In Fig. 4, we plot the sequences of energy-decreasing transitions together with the energy values in the experiment with \( fish \), for three \( \theta \) values. We notice that some solutions have very small differences in the energies and are qualitatively indistinguishable from each other. This is accounted for by the choice of the additive basis, i.e., that the same alignment can be encoded in different ways. In contrast, we see significant differences in the energy values of the qualitatively different solutions (orders of magnitudes larger in the analysed experiment).

We conclude that even though \( \Delta(\hat{H}) \) is small, the alignments corresponding to several few lowest eigenstates are qualitatively similar. This suggests that our selection of the basis leads to problems with sufficient spectral gaps.

8. Conclusions

In summary, this paper introduces adiabatic quantum computers (AQC) for the computer vision community and shows that fundamental low-level problems can be brought to a representation suitable for solving by AQC. We provide a detailed and thorough overview of the modern AQC technology and propose a new method for transformation estimation and point set alignment which can be directly mapped and solved on modern adiabatic quantum computers. In simulations on a classical computer and in a wide range of scenarios, QA is shown to successfully recover 2D transformations which are close approximations of globally optimal transformations. With the chosen basis of 20 elements, the estimated transformations result in low transformation discrepancy and alignment errors. Observations on how to avoid singularities and the spectral gap analysis complement the experimental section.

In future work, our technique can be extended to affine transformations. We hope to see more research on computer vision methods with quantum hardware in the next decades.
A. Appendix

In this additional section, we provide details on the selection of the annealing rate, analyse the structure of \( \mathbf{P} \) and formalise the unembedding procedure, i.e., the conversion of the solution to QUBOP (16) to the solution of the original alignment problem on point sets.

**Annealing Rate.** Suppose \( E_m(s) \) is the ground state of instantaneous Hamiltonian, \( E_m(0) \) is the initial state (ground state) of the system and \( E_m(s) \) is any other excited state of the instantaneous Hamiltonian. Let \( s = \frac{T}{t} \in [0; 1] \), where \( T \) is the overall time of interpolation and \( t \) is physical time. Then, according to [4], \( T \) has to be chosen so that

\[
T \gg \frac{|(E_m(s)|dH/ds|E_m(s))|}{E_{nm}(s)^2}, \quad \forall m \neq n, \quad (27)
\]

where \( dH/ds \) is the rate of change of Hamiltonian with respect to \( s \) and \( E_{nm} \) is the difference in the corresponding instantaneous energies.

**Analysis of \( \mathbf{P} \).** Fig. I visualises several exemplary weight matrices \( \mathbf{P} \) from the experiments with clean and noisy data (see Sec. 7). There are several observations. First, \( \mathbf{P} = \mathbf{\Phi} \mathbf{\Phi}^T \) is symmetric upon algorithm design. We also see that the columns of \( \mathbf{\Phi} \) can be arbitrarily reshuffled as long as the correspondences are preserved\(^3\). Second, \( \mathbf{P} \) contains regularly arranged zero submatrices, due to our choice of the basis. As soon as a row of \( \mathbf{\Phi} \) induced by \( q\mathbf{C}_1 \), where \( \mathbf{C}_1 = \{1, -1\} \), is multiplied by a column of \( \mathbf{\Phi}^T \), induced by \( q\mathbf{C}_M \), where \( \mathbf{C}_M = \{0, \mathbf{M}, -\mathbf{M}\} \), and vice versa, we obtain a zero entry in \( \mathbf{P} \). The reason is that

\[
\begin{bmatrix}
\mathbf{I} \sum_i y_i \\
-\mathbf{I} \sum_i y_i \\
\mathbf{M} \sum_j y_j \\
-\mathbf{M} \sum_j y_j
\end{bmatrix} = 0
\]

if \( \sum_i y_i = \sum_j y_j \), which holds in our case since each row of \( \mathbf{\Phi} \) except the first row includes all points of \( \mathbf{Y} \) multiplied by a single basis element \( \mathbf{Q}_k \) (see Fig. I-(top left) for \( \mathbf{C} \) pairs resulting in zero matrices). Third, the structure of \( \mathbf{P} \) reflects that its diagonal entries encode biases, and non-diagonal elements represent couplings between the qubits.

With the increasing \( K \), the span of the absolute energy values increases, due to the higher number of point interactions. As expected, \( \mathbf{P} \) depends on data and the angle of initial misalignment between the point sets. For all possible inputs and initial conditions — point sets of different cardinalities, \( K \) and \( \theta \) — the structure of \( \mathbf{P} \) is the same for the chosen basis. From \( \mathbf{P} \), we also recognise that the considered alignment problem is not purely combinatorial and requires high-precision weights \( J_{j,k} \) in (5).

\[^3\text{a reshuffling of rows requires changing the order of elements in } \mathbf{Q} \]

**Unembedding.** Unembedding is the decoding of the solution to QUBOP (16) to the solution of the original alignment problem. Upon the design, our QA method assembles the entries of the transformation matrix in the additive basis \( \mathbf{Q}_k \) (see Secs. (5.1)–(6.1)). Suppose \( \mathbf{q} \) is the measurement result of \( \mathbf{q} \), i.e., it is a classical bitstring with \( K + 1 \) elements. Recall that \( \mathbf{q}_1 \) is reserved for reference points and does not contribute to the assembly of the transformation. Once \( \mathbf{q} \) is measured and returned, we obtain the corresponding transformation \( \mathbf{R} \) by summing up \( \mathbf{Q}_k \) multiplied by \( \mathbf{q}_{k+1} \):

\[
\mathbf{R} = \sum_{k} \mathbf{q}_{k+1} \mathbf{Q}_k. \quad (29)
\]

The obtained \( \mathbf{R} \) is an affine transformation. If the solution has to represent a valid rotation matrix \( \mathbf{R}_t \), \( \mathbf{R} \) can be projected to the rotation group by solving the closest orthogonal approximation problem with constraints:

\[
\begin{align*}
\min & \quad \| \mathbf{R}_t - \mathbf{R} \|_H^2 \\
\text{s. t.} & \quad \mathbf{R}_{t}^{-1} = \mathbf{R}_t^T \text{ and } \det(\mathbf{R}_t) = 1
\end{align*} \quad (30)
\]

For a solution to (30) by singular value decomposition, see [33].
References


