Linearly Converging Quasi Branch and Bound Algorithms for Global Rigid Registration

Nadav Dym
Duke University
nadavd@math.duke.edu

Shahar Ziv Kovalsky
Duke University
shaharko@math.duke.edu

Abstract

In recent years, several branch-and-bound (BnB) algorithms have been proposed to globally optimize rigid registration problems. In this paper, we suggest a general framework to improve upon the BnB approach, which we name Quasi BnB. *Quasi BnB replaces the linear lower bounds used in BnB algorithms with quadratic quasi-lower bounds which are based on the quadratic behavior of the energy in the vicinity of the global minimum. While quasi-lower bounds are not truly lower bounds, the Quasi-BnB algorithm is globally optimal. In fact we prove that it exhibits linear convergence – it achieves ϵ-accuracy in $O(\log(1/\epsilon))$ time while the time complexity of other rigid registration BnB algorithms is polynomial in $1/\epsilon$. Our experiments verify that Quasi-BnB is significantly more efficient than state-of-the-art BnB algorithms, especially for problems where high accuracy is desired.*

1. Introduction

Rigid registration is a fundamental problem in computer vision and related fields. The input to this problem are two similar shapes and the goal is to find the rigid motion that best aligns the two shapes, as well as a good mapping between the aligned shapes. There are several different formulations of the rigid registration problem. In this paper we focus on two popular formulations, the rigid closest point *(rigid-CP)* [6] problem which is appropriate for partial matching problems (*e.g*. where a partial point cloud obtained from a scanning device is mapped to a reconstructed model), and the *rigid-bijective* problem [28] which is applicable to full matching problems where it naturally defines a distance between shapes [3].

Both of these rigid registration formulations optimize an appropriate energy $E(g, \pi)$ that depends on the chosen rigid motion g and a correspondence π between the two given point clouds. This energy is typically non-convex, however it is *conditionally tractable* in that there exist efficient algorithms for globally optimizing for either of its variables when the other is fixed. This is put to good use by the well-known *iterative closest point* (ICP) algorithm [6] that optimizes E using a very efficient alternating approach. It converges, however, to a local minimum and thus it strongly depends on a good initialization.

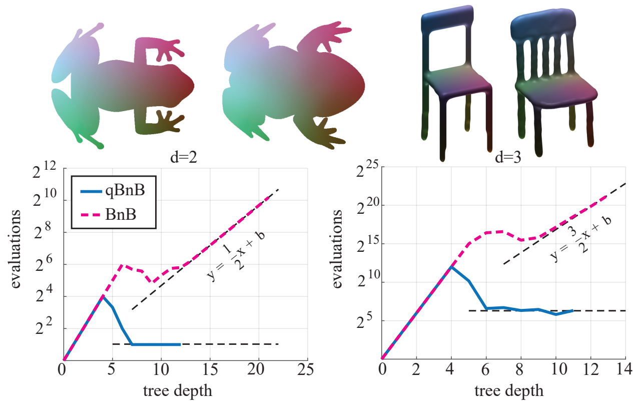


Figure 1: Comparing the proposed qBnB algorithm and the BnB algorithm of [21] for rigid bijective problems. The colored shapes illustrate the alignment and correspondence maps obtained for a pair of 2D and 3D shapes. The graphs demonstrate the preferable complexity of our qBnB approach, dashed lines illustrating the asymptotic behavior derived in our analysis (Theorem 2).

Conditional tractability can also be useful for *global* optimization algorithms. For a fixed rigid motion g it is possible to compute

$$F(g) = \min_{\pi} E(g, \pi)$$

and so to optimize E globally it suffices to optimize F. Figure 2(a) shows an example function F derived from a 2D rigid registration problem. The function F is non-convex and non-differentiable, and the complexity of global optimization algorithms for optimizing F is typically exponential in the dimension D of the rigid motion space. However, in many cases of interest D is a small constant and so these algorithms are in fact tractable. Thus, while high dimen-

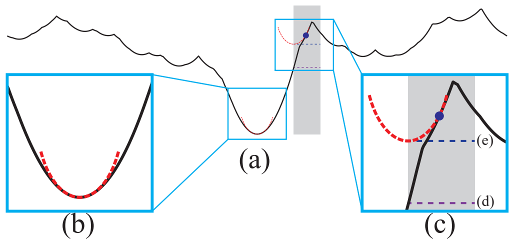


Figure 2: (a) Illustration of the function F_{CP} defined in Subsection 3.4 for a 2D registration problem as a function of rotation angle. (b) Quadratic behavior of F_{CP} at a minimum. (c) The true minimum (d) on the gray interval and a quasi-lower bound (e) computed using the quadratic behavior near the global minimum.

sional rigid registration problems can be computationally hard [13], they are fixed parameter tractable [11].

Branch and bound (BnB) algorithms [10, 33, 25, 21] are perhaps the only algorithms with a deterministic guarantee to globally optimize the rigid registration problem. The key ingredient in BnB algorithms is the ability to give a lower bound for the value of F in a certain region, based on a single evaluation of F.

In this paper we introduce the term *quasi-lower bound*, which is a lower bound for the value of the function, over a given region, under the *possibly-false* assumption that the region contains a global minimum. While quasi-lower bounds are not true bounds, the global optimality of qBnB algorithms is not compromised if lower bounds are replaced by quasi-lower bounds - leading to algorithms that we name quasi-BnB *(qBnB)*.

The advantage of quasi-lower bounds is that for conditionally smooth energies (as defined in Definition 1) F exhibits quadratic behavior near the global minimum, and this can be leveraged to define quadratic quasi-lower bounds, in contrast with lower bounds that are typically linear. This leads to much improved bounds, as illustrated in Figure 2, where the quasi-lower bound (e) is able to provide a bound that is higher than the true minimal value (d) in the gray interval, and hence tighter than any lower bound.

The tightness of quasi-lower bounds yields substantial benefits in terms of complexity –we show that the complexity of achieving an ϵ-optimal solution using BnB is at best $\sim \epsilon^{-D/2}$. In contrast, under some weak assumptions, the complexity of the qBnB algorithm is $\sim \log(1/\epsilon)$, that is, it achieves a *linear convergence rate* (in the sense of [7]). Our experiments (*e.g.* Figure 1) verify these theoretical results, and show that qBnB can be considerably more efficient than state-of-the-art BnB algorithms, especially in the presence of high noise or when high accuracy is required. The code used for producing the results in this paper will be made available online.

2. Related Work

We focus on global optimization methods for rigid registration. For other aspects of the rigid registration problem we refer the reader to surveys such as [30].

A variety of methods from the global optimization toolbox have been used to address the rigid registration problem, including Monte Carlo [19], RANSAC based [2], particle filtering [29], particle swarming [32], convex relaxations [22, 20], and graduated non-convexity [35]. Unlike BnB algorithms, none of these algorithms come with a deterministic guarantee for global convergence.

In [25, 23, 9] BnB algorithms for rigid matching in 2D are suggested. In [16] an algorithm for rigid matching in 3D is proposed, based on a combination of matching rotation-invariant descriptors and BnB. In [21] a BnB algorithm for a modified bijective matching objective in 3D is suggested, based on Lipschitz optimization.

In [33, 34] the Go-ICP algorithm for rigid matching in 3D is suggested. Their algorithm searches the 6D transformation space very efficiently in the low noise regime, although it can be slow in the high noise regime. A branch and bound procedure for the rigid matching energy of [9] in 3D is proposed in [10, 24]. For this energy they suggest various speedups that are not available for the classic maximum likelihood objective considered in ICP.

The qBnB approach we propose here can boost the convergence of any of the methods above which consider conditionally smooth objective, such as [33, 25, 21], by simply replacing their first order lower bound with an appropriate second order quasi-lower bound. However it is not directly applicable for optimizing objectives which are not conditionally smooth such as the geometric matching objective of [10, 9].

BnB algorithms with quadratic lower bounds have been proposed for smooth unconstrained optimization [15], and for camera pose estimation [18]. The latter approach obtains quadratic lower bounds by jointly optimizing over a linear approximation of a rotation, and the remaining variables; This approach does not straightforwardly apply to our scenario. In contrast, our qBnB approach only requires solving optimization problems with a fixed rotation.

[26] show that ICP achieves linear converge rate, and suggest alternative second order methods with super-linear convergence rate. These algorithms only converge to a local minimum, whereas our algorithm is shown to exhibit linear convergence to the *global minimum*.

3. Method

3.1. Problem statement

We consider the application of the quasi BnB framework as described in Section 1 to the problem of rigid alignment of point clouds. This problem has several variants, and in this paper we will focus on two of them that we will refer to as the rigid-closest-point (*rigid-CP*) problem and the *rigid-bijective* problem. We also wish to focus on the common structure of problems for which our quasi BnB framework is applicable, which we will name in this paper *D-quasi-optimizable*. We begin by defining D-quasi-optimizable problems.

Consider optimization problems of the form

$$\min_{x\in\mathbb{R}^D, y\in\mathcal{Y}} E(x,y), \tag{1}$$

where $\mathcal{Y}$ is a finite set. Denote the cube centered at x with half-edge length h by $\mathcal{C}_h(x)$.

Definition 1. *We say that an optimization problem of the form* (1) *is D-quasi-optimizable in a cube $\mathcal{C}^0 = \mathcal{C}_{h_0}(x_0)$ if it satisfies the following conditions:*

1. Existence of a minimizer: *There exists a minimizer (x_*, y_*) of E such that $x_* \in \mathcal{C}^0$.*
2. D-tractability*: For fixed x, minimizing $E(x,\cdot)$ over $\mathcal{Y}$ can be performed in polynomial time.*
3. Conditional smoothness *For fixed $y\in\mathcal{Y}$, the function $E(\cdot,y)$ is in $C^\infty(\mathbb{R}^d)$.*

We now turn to defining the rigid-CP and rigid-bijective problems and explaining why they are D-quasi-optimizable.

The input to the rigid-CP problem are two point clouds $\mathcal{P} = \{p_1,\dots,p_n\}$ and $\mathcal{Q} = \{q_1,\dots,q_m\}$ in $\mathbb{R}^d$ where we assume $d=3$ or $d=2$. We assume the point clouds are normalized to be in $[-1,1]^d$ and have zero mean. Our goal is to find a rigid motion that aligns the points as-well-as-possible. Namely, denoting the group of (orientation preserving) rigid motions $\mathrm{SO}(d)\times\mathbb{R}^d$ by G_{CP}, and denoting by Π_{CP} the collection of all functions $\pi:\mathcal{P}\to\mathcal{Q}$, our goal is to solve the minimization problem

$$\min_{\substack{(R,t)\in G_{\mathrm{CP}}\\ \pi\in\Pi_{\mathrm{CP}}}} E_{\mathrm{CP}}(R,t,\pi) = \frac{1}{n}\sum_{i=1}^{n}\|Rp_i + t - q_{\pi(i)}\|^2 \tag{2}$$

Following [33, 21] we simplify the domain of (2) by using the exponential map: We use s to denote the intrinsic dimension of $\mathrm{SO}(d)$. For a vector $r\in\mathbb{R}^s$, we define $[r]$ to be the unique $d\times d$ skew-symmetric matrix ($[r]^T$=$-[r]$) whose entries under the diagonal are given by r. By applying the matrix exponential to $[r]$ we get a matrix

$$R_r = \exp([r]) \in \mathrm{SO}(d).$$

Every rotation can be represented as R_r for some r in the closed ball $B_\pi(0)$ and so (2) can be reduced to the problem

$$\min_{(r,t)\in\mathbb{R}^s\times\mathbb{R}^d, \pi\in\Pi_{\mathrm{CP}}} E_{\mathrm{CP}}(R_r,t,\pi). \tag{3}$$

Our next step is to identify a cube $\mathcal{C}^0$ in $\mathbb{R}^s\times\mathbb{R}^d$ in which E_{CP} must have a global minimum. For the rotation component we can take the cube $\mathcal{C}_\pi(0)$ that bounds the ball $B_\pi(0)$. For the translation component, we note that if (R_*,π_*,t_*) is a minimizer of E, then the optimal translation is the difference between the average of R_*p_i, which is zero by assumption, and the average of $q_{\pi(i)}$, which will be in the unit cube. Thus there exists a minimizer (r_*,t_*,π_*) such that (r_*,t_*) is in $\mathcal{C}_\pi(0)\times\mathcal{C}_1(0)$. This shows that (3) satisfies the first condition for being D-quasi-optimizable, where $x=(r,t)$, $y=\pi$ and $D=s+d$ (*i.e.*, $D=6$ or 3 for 3D and 2D problems respectively).

D-tractability follows from the fact that the optimal mappings π minimizing $E_{\mathrm{CP}}(R,t,\cdot)$ for fixed R,t are just the mappings that take each rigidly transformed point Rp_i+t to its closest point in $\mathcal{Q}$, hence the term rigid-CP. Conditional smoothness is obvious.

The rigid bijective problem is similar to rigid-CP, but focuses on the case $n=m$, and only allows mappings between $\mathcal{P}$ and $\mathcal{Q}$ that are bijective (*i.e.*, permutations). We denote this set of mappings by Π_{bi}. In this scenario optimizing for π while holding the rigid transformation component fixed is tractable, but more computationally intensive than rigid-CP since it requires solving a linear assignment problem. On the other hand, in this scenario the optimal translation is always $t_*=0$, since the mean of the points p_i and $q_{\pi(i)}$ is zero. Therefore we can reduce our problem to lower-dimensional optimization over $G_{\mathrm{bi}}=\mathrm{SO}(d)$. As before, we use the exponential map to reparameterize the problem as

$$\min_{r\in\mathbb{R}^s, \pi\in\Pi_{\mathrm{bi}}} E_{\mathrm{bi}}(R_r,\pi) = \frac{1}{n}\sum_i \|R_r p_i - q_{\pi(i)}\|^2 \tag{4}$$

It follows from our discussion above that this optimization problem is D-quasi-optimizable over $\mathcal{C}_\pi(0)$ with $x=r, y=\pi$ and $D=s$ (*i.e.*, $D=3$ or 1 for 3D and 2D problems respectively).

3.2. Optimizing quasi-optimizable functions

BnB algorithms. D-tractability implies that we can reduce (1) to the equivalent problem of minimizing the D-dimensional function

$$F(x) = \min_{y\in\mathcal{Y}} E(x,y). \tag{5}$$

BnB algorithms for rigid registration and related problems are typically based on the ability to show that for $\delta>0$ and $x_1,x_2\in\mathcal{C}^0$ satisfying $\|x_1-x_2\|\le\delta$,

$$F(x_1) - F(x_2) \le \Delta(\delta) \tag{6}$$

where $\Delta(\cdot)$ is a function of the form

$$\Delta(\delta) = L\delta + O(\delta^2). \tag{7}$$

Using a bound of this form F can be bounded from below *in a cube* $\mathcal{C}_{h_i}(x_i)$ by evaluating $F(x_i)$ and noting that the distance of any x in the cube from x_i is at most $\sqrt{D}h_i$, and thus (6) implies that $F(x)$ is larger than

$$\text{lb}_i \equiv F(x_i) - \Delta(\sqrt{D}h_i). \tag{8}$$

In the appendix we show that any F arising from a D-quasi-optimizable optimization problem is Lipschitz and thus a bound of the form (7) always exists. We note that to apply a BnB algorithm to a specific problem it is necessary to explicitly compute a valid bounding function $\Delta(\cdot)$.

Based on the lower bound (8), a simple breadth-first-search (BFS) BnB algorithm starts from a coarse partitioning of $\mathcal{C}^0$ into sub-cubes $\mathcal{C}_{h_i}(x_i)$, and then evaluates F at each of the x_i. Each such evaluation gives a global upper bound $\text{ub}_i = F(x_i)$ for the minimum F^*, and a local lower bound lb_i for the minimum *in the cube* as defined in (8). At each step the algorithm keeps track of the best global upper bound found so far, which we denote by ub, and on the global lower bound lb that is defined as the smallest local lower bound found in this partition. Now for every i, If $\text{lb}_i > \text{ub}$ then F is not minimized in $\mathcal{C}_{h_i}(x_i)$ and this cube can be excluded from the search. The BnB algorithm then refines the partition into smaller cubes for cubes that have not yet been eliminated, and this process is continued until $\text{ub} - \text{lb} < \epsilon$. This is the BnB strategy used in [33, 25, 21, 23, 9] for rigid registration problems, though these papers vary in the search strategy they employ, the rigid energy the consider, the bound (6) they compute, and other aspects.

Quasi-BnB. The qBnB algorithm we suggest in this paper is based on two observations: The first is that though non-differentiable, F resembles a smooth function in that its behavior near minimizers is quadratic, and in fact F can be bounded in the cube $\mathcal{C}^0$ by a quadratic function centered at a global minimizer.

Lemma 1.

If $x_* \in \mathcal{C}^0$ is a global minimizer of F, then there exists $C > 0$ such that

$$F(x) - F(x_*) \le C\|x - x_*\|^2, \forall x \in \mathcal{C}^0 \tag{9}$$

Proof. Assume (x_*, y_*) is a global minimizer of E, Let $N_{x,y}$ be the operator norm of the Hessian of the function $E(\cdot, y)$ at a point x, and let C be the maximum of $2N_{x,y}$ over $\mathcal{C}^0 \times \mathcal{Y}$. Since the first order approximation of $E(\cdot, y_*)$ at x_* vanishes, we obtain

$$\begin{aligned} F(x) - F(x_*) &\le E(x, y_*) - E(x_*, y_*) \\ &\le C\|x - x_*\|^2 \end{aligned} \tag{10}$$

□

Let us assume that we have an explicit quadratic bound of the general form (10), up to possibly allowing higher order terms. That is, if x_* is a global minimizer, and $x, x_* \in \mathcal{C}^0$ satisfy $\|x - x_*\| \le \delta$, then

$$F(x) - F(x_*) \le \Delta_*(\delta) \tag{11}$$

where Δ_* is a function of the form

$$\Delta_*(\delta) = C\delta^2 + O(\delta^3) \tag{12}$$

In Subsection 3.4 we give an explicit computation of Δ_* for rigid-bijective and rigid-CP problems.

Using a bound of the form (11), F can be bounded from below in a cube $\mathcal{C}_{h_i}(x_i)$ *which contains a global minimizer* x_* by evaluating $F(x_i)$ and noting that since x_* is in the cube, its distance from x_i is at most $\sqrt{D}h_i$, and thus (11) implies that the minimum $F(x_*)$ is larger than

$$\text{qlb}_i \equiv F(x_i) - \Delta_*(\sqrt{D}h_i). \tag{13}$$

Note that qlb_i is well defined for any cube $\mathcal{C}_{h_i}(x_i)$, even if the cube does not contain a minimizer. However it is only guaranteed to be a true lower bound for the value of F in $\mathcal{C}_{h_i}(x_i)$ if the cube contains a global minimizer. For this reason we call qlb_i a *quasi-lower bound*.

The second observation the qBnB algorithm is based on is that the linear lower bounds (8) in the BnB algorithm can be replaced by quadratic quasi lower bounds (13) to obtain what we refer to as the qBnB algorithm (qBnB), without compromising the global optimality of the algorithm. To see this is indeed true we need to convince ourselves that if a given cube $\mathcal{C}_{h_i}(x_i)$ contains a global minimum x_*, then the cube will not be discarded by the qBnB algorithm. Indeed if $\mathcal{C}_{h_i}(x_i)$ contains a global minimum then qlb_i is a true lower bound for the minimal value of F in the cube, which in this case is exactly the minimum of the function, so for any given upper bound ub,

$$\text{qlb}_i \le F(x_*) \le \text{ub}$$

and so the cube will not be removed. For more details on global optimality we refer the interested reader to the proof of Theorem 2 which includes a formal proof of global optimality.

A simple BFS qBnB algorithm for optimizing D-quasi-optimizable functions, for a given quasi-lower bound $\Delta_*(\delta)$, is provided in Algorithm 1. We note that this algorithm also exploits the fact that for a collection of cubes L_g partitioning $\mathcal{C}^0$,

$$\text{lb} = \min\{\text{qlb}_i | \mathcal{C}_{h_i}(x_i) \in L_g\}$$

is a true lower bound for the global minimum, since one of the cubes in L_g must contain a global minimizer and hence the quasi-lower bound for this cube will be lower than the global minimum.

```
input  : Required accuracy ϵ
output: ϵ-optimal solution x_*
ub ← ∞, lb ← −∞, g ← 0;
Put C^0 = C_{h_0}(x_0) into the list L_g ;
while ub − lb > ϵ do
    forall C_{h_i}(x_i) ∈ L_g do
        Compute F(x_i);
        ub_i ← F(x_i), qlb_i ← F(x_i) − Δ_*(√D h_i);
        if ub_i < ub then
            ub ← ub_i;
            x_* ← x_i;
        end
    end
    lb = min{qlb_i | C_{h_i}(x_i) ∈ L_g} ;
    forall C_{h_i}(x_i) ∈ L_g do
        if qlb_i ≤ ub then
            subdivide C_{h_i}(x_i) into 2^D sub-cubes
             with half-edge length h_i/2 and insert
             into L_{g+1}
        end
    end
    g ← g + 1;
end
```

Algorithm 1: BFS qBnB

3.3. Complexity

Theorem 2 (below, proof in appendix) provides complexity bounds for the qBnB algorithm with a quadratic $\Delta_*(\delta)$, as in Algorithm 1, and for a similar BnB algorithm, where Δ_* is replaced by a Lipschitz bound $\Delta(\delta)$. We denote the number of F-evaluations in the algorithms by n_{qBnB} and n_{BnB} respectively. We consider the limit where the prescribed accuracy ϵ tends to zero and all other parameters of the problem are held fixed:

Theorem 2. *There exist positive constants $C_1, \dots, C_4$, such that*

$$C_1 \epsilon^{-D/2} \le n_{\mathrm{BnB}} \le C_2 \epsilon^{-D}. \tag{14}$$

$$n_{\mathrm{qBnB}} \le C_3 \epsilon^{-D/2}. \tag{15}$$

Furthermore, if E has a finite number of minimizers $(x_\ell^*, y_\ell^*)_{\ell=1}^N$, and the Hessian of $E(\cdot, y_\ell^*)$ is strictly positive definite for all ℓ, then

$$n_{\mathrm{qBnB}} \le C_4 \log_2(1/\epsilon). \tag{16}$$

3.4. Computing quasi-lower bounds

To conclude our discussion we now need to state explicit quadratic quasi-lower bounds for the rigid-CP and rigid-bijective problems. We denote by F_{CP} and F_{bi} the functions obtained by applying (5) to E_{CP} and E_{bi}.

Rigid bijective quasi-lower bounds. For $k \in \mathbb{N}$, let ψ_k denote the truncated Taylor expansion of e^x,

$$\psi_k(x) = e^x - \sum_{j=0}^{k-1} \frac{x^j}{j!} \tag{17}$$

Theorem 3. *Let $\delta > 0, r \in \mathbb{R}^D$ and r_* be a global minimizer of F_{bi}, and assume $\|r - r_*\| \le \delta$. Let $\sigma_{\mathcal{P}}, \sigma_{\mathcal{Q}}$ denote the Frobenius norm of the matrices whose columns are the points in $\mathcal{P}$ and $\mathcal{Q}$ respectively. Then $\Delta_*(\delta)$ is given by*

$$F_{\mathrm{bi}}(r) - F_{\mathrm{bi}}(r_*) \le \Delta_*(\delta) \equiv \frac{2}{n} \sigma_{\mathcal{P}} \sigma_{\mathcal{Q}}\, \psi_2(\delta) \tag{18}$$

Proof. Let us consider the case where $r_* = 0$, so (I_d, π_*) minimizes E_{bi} for some appropriate π_*. Note that for all R, π,

$$E_{\mathrm{bi}}(R, \pi) = \frac{1}{n}\left[\sigma_{\mathcal{P}}^2 + \sigma_{\mathcal{Q}}^2 - 2\sum_i \langle Rp_i, q_{\pi(i)}\rangle\right] \tag{19}$$

it follows that

$$\begin{aligned}
F_{\mathrm{bi}}(r) - F_{\mathrm{bi}}(0) &\le E_{\mathrm{bi}}(R_r, \pi_*) - E_{\mathrm{bi}}(I_d, \pi_*) \\
&\overset{(*)}{=} -\frac{2}{n}\sum_{i=1}^{n}\sum_{k=2}^{\infty}\langle \frac{1}{k!}[r]^k p_i, q_{\pi(i)}\rangle \\
&\le \frac{2}{n}\sum_{i=1}^{n}\sum_{k=2}^{\infty}\frac{1}{k!}\|[r]\|_{\mathrm{op}}^k \|p_i\|\|q_{\pi_*(i)}\| \\
&\le \frac{2}{n}\psi_2(\|[r]\|_{\mathrm{op}})\sigma_{\mathcal{P}}\sigma_{\mathcal{Q}}.
\end{aligned}$$

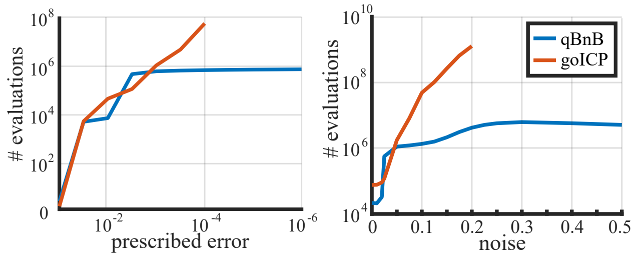
To obtain the equality $(*)$ note that $r_* = 0$ minimizes $E_{\mathrm{bi}}(\cdot, \pi_*)$ and so the first order approximation of this function vanishes. In the appendix we show that $\|[r]\|_{\mathrm{op}} \le \|r\|$ which concludes the proof for the case $r_* = 0$, and then use this case to prove the theorem for general r_*. □

Based on this quasi-lower bound, our algorithm for optimizing the rigid-bijective problem is just Algorithm 1, where as stated in Subsection 3.1 we take $x = r, y = \pi, \mathcal{C}_{h_0}(x_0) = \mathcal{C}_\pi(0)$, and we use Δ_* defined in (18).

Rigid CP quasi-lower bounds. For the rigid-CP problem we obtain the following quasi-lower bound that we prove in the appendix.

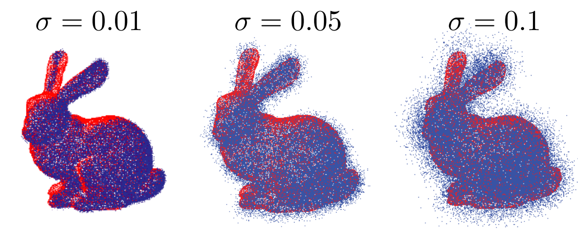
Theorem 4. *Let (r_*, t_*) be a minimizer of F_{CP}, and let $(r, t) \in \mathbb{R}^s \times \mathbb{R}^d$, and $\delta_1, \delta_2 > 0$ which satisfy $\|r - r_*\| \le \delta_1$and$\|t - t_*\| \le \delta_2$. Let f_* be some upper bound for the global minimum of F_{CP}. Then*

$$F_{\mathrm{CP}}(r, t) - F_{\mathrm{CP}}(r_*, t_*) \le \Delta_*(\delta_1, \delta_2) \tag{20}$$



(a) (b)

Figure 3: Comparison of the dependence of qBnB and Go-ICP on the error tolerance (a) and noise level (b).



Go-ICP	4.1K	5.2×10^6	1×10^8
Q-BnB	2K	1.6×10^6	1.4×10^6

Figure 4: Registration of a noisy scan of the Stanford bunny to the full 3D model at three different noise levels. The bottom row shows the number of F_{CP} evaluations Go-ICP and qBnB required to perform the registrations.

where

$$\Delta_*(\delta_1, \delta_2) = \frac{1}{n}\Big[2\psi_2(\delta_1)(\sigma_{\mathcal{P}}^2 + \sigma_{\mathcal{P}}\sqrt{nf_*}) + 2\delta_2\psi_1(\delta_1)\sum_i \|p_i\| + n\delta_2^2\Big] \quad (21)$$

Using this quasi-lower bound the most straightforward way to construct a qBnB algorithm is by using Algorithm 1 setting $x = (r, t), y = \pi$ and using the initial cube $\mathcal{C}_\pi(0) \times \mathcal{C}_1(0)$ as described in Subsection 3.1 and the bound $\Delta_*(\delta_1, \delta_2)$ from (21). However, to enable simple comparison with the Go-ICP algorithm [33] we use their BnB architecture, wherein two nested BnB are used – an outer BnB for the rotation component and a separate inner BnB for the translation component. The quadratic quasi-lower bounds for these BnBs can be computed by setting $\delta_2 = 0$ or $\delta_1 = 0$, respectively. Further details are provided in the supplementary material.

4. Results

Evaluation of qBnB for rigid-CP. We evaluate the performance of the proposed qBnB algorithm for the rigid-CP problem. We compare with Go-ICP [33], a state-of-the-art approach for global minimization of this problem. Our implementation is based on a modified version of the Go-ICP code, which utilizes a nested BnB (see section 3.4) and an efficient approximate CP computation (see implementation details below).

We ran both algorithms on synthetic rigid-CP problems that were generated by uniformly sampling n points on the unit cube to form $\mathcal{P}$, and then applying a random rigid transformation and Gaussian noise with std σ to form $\mathcal{Q}$. Finally $\mathcal{P}$ and $\mathcal{Q}$ are translated so that they have zero mean and then scaled by $\min_{i,j}\{\|p_i\|_\infty^{-1}, \|q_j\|_\infty^{-1}\}$ so that they both reside in the unit cube.

Figure 3 shows comparisons performed with varying (a) prescribed accuracy and (b) noise. Both algorithms are comparable in the low noise/low accuracy regime. This is consistent with [33] that report better performance in the regime where the optimal energy E^* is lower than the error tolerance ϵ. In contrast, when accuracy or noise are increased, the complexity of the Go-ICP algorithm grows rapidly, while the complexity of qBnB stabilizes at an almost constant number of $\approx 10^6$ function evaluations. This results is consistent with the complexity analysis of Theorem 2.

In Figure 3(a) we took $n = 50$, $\sigma = 0.05$, and prescribed accuracy ϵ varying between 10^{-1} and 10^{-6}, and in Figure 3(b) we took $n = 100$, $\epsilon = 10^{-3}$ and σ varying between 0 and 0.5. Each data point in the graph represents an average over 100 instances with the same parameters. We exclude results for Go-ICP with $\epsilon < 10^{-4}$ or $\sigma > 0.2$ due to excessive run-time (over six hours). The maximal run-time of our algorithm over all experiments presented was 9 minutes.

Figure 4 exemplifies the applicability of our algorithm for solving larger rigid-CP problems. We used our algorithm and Go-ICP to register 500 points sampled from the Stanford bunny [31] to a full 3D model consisting of nearly 36K vertices. Both methods are global, thus they obtained the same results up to small indiscernible differences; therefore, we only show our alignments, as well as the evaluation counts of both algorithms. As in Figure 3(b), qBnB is more efficient than Go-ICP, especially in the high noise regime. For noise level of $\sigma = 0.1$ our algorithm required under 7 minutes, whereas Go-ICP took over 6 hours to complete.

Evaluation of qBnB for rigid-bijective. We compare our qBnB algorithm for the rigid bijective problem with the BFS version of the BnB algorithm proposed in [21]. This algorithm modifies the standard ℓ^2 energy we study here, and globally optimizes it over the same feasible set $G_{\text{bi}} \times \Pi_{\text{bi}}$. in Figure 1 we examined the performance of both algorithms on the problem of registering 2D frog silhouettes [12] and 3D chair meshes [17], sampled at 50 points, with accuracy set at $\epsilon = 10^{-6}$. Both algorithms returned identical results, which are visualized on the top of Figure 1. However qBnB took 0.5 and 6.5 seconds to solve the 2D and 3D problems

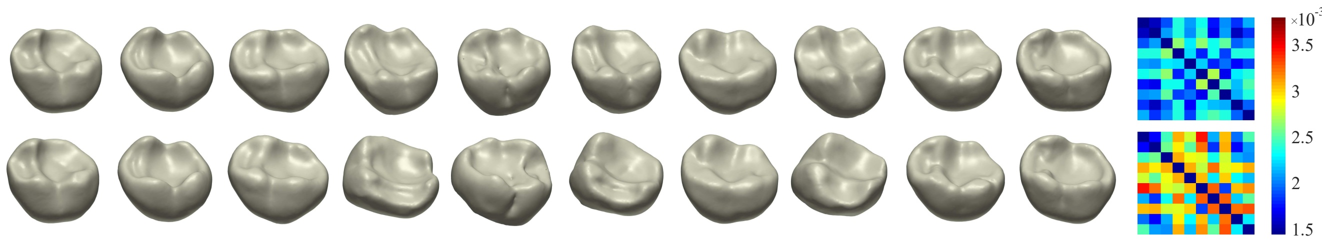


Figure 5: The top row shows the alignment computed by qBnB between the first tooth on the left and all remaining teeth. The matrix on the right shows all pairwise distances computed for the teeth. The bottom row shows the analogous result for Auto3dgm [8], a local alignment algorithm used in biological morphology.

respectively, while [18] required 11 seconds to solve the 2D problem, and did not converge to the required accuracy for the 3D problem after 10^7 evaluations (over 1 hour). The graphs in Figure 1 show the number of evaluations the algorithms performs in each depth level of the BFS. In both cases qBnB required significantly less evaluations than [21], and furthermore we note that asymptotically qBnB requires a constant number of evaluations at each depth, while [21] requires $\sim 2^{Dg/2}$ iterations at depth g, as shown by the log-scale asymptotes in the figure. This illustrates the theoretical results of Theorem 2: As for BnB algorithms $\epsilon \sim 2^{-g}$, they require $\sim \epsilon^{-D/2}$ evaluations. In contrast, the dependence of qBnB on $g \sim \log(1/\epsilon)$ is linear.

Application to biological morphology. Biological morphology is concerned with the geometry of anatomical shapes. In this context, Auto3dgm [8, 27] is a popular algorithm that seeks for a plausible, consistent alignment of a collection of scanned morphological shapes. It is based on solving the rigid bijective matching problem between pairs of shapes, and on a global step that synchronizes the rotations. The rigid bijective matching problem is solved using an alternating ICP-like algorithm, initialized from each of the 2^3 rotations that align the principal axes of the two shapes. This results in 8 possible solutions, from which the solution with best energy is selected. In this application, reflections are typically allowed, and our algorithm can be easily extended to this case as $\mathrm{O}(d)$ is just a disjoint union of two copies of $\mathrm{SO}(d)$.

Figure 5 compares the performance of qBnB to the pairwise matching algorithm of Auto3dgm for aligning a challenging collection of ten almost-symmetric teeth of spider monkeys (Ateles) from [1]. Details on the data for this experiment can be found in Appendix C. Each bone surface is sampled at 200 points using farthest point sampling. The pairwise energy between all pairs, and the alignment computed between the first tooth and all remaining teeth is shown for qBnB (top) and Auto3dgm (bottom). We see that qBnB obtains lower energy solutions and more plausible alignments which have been verified as semantically correct by biological experts. This comes at the cost of higher complexity ($2-5$ minutes as opposed to ~ 3 seconds).

Implementation details. The qBnB algorithm for rigid-CP was implemented in C++, based on the implementation of [33]. The algorithm for rigid-bijective matching was implemented in Matlab; We solved for the bijective mapping used the excellent MEX implementation of [4] for the auction algorithm [5]. Timings were measured using a Intel 3.10GHz CPU.

For the optimization of large scale rigid-CP with moderate accuracy $\epsilon = 10^{-3}$ as in Figure 3(b) and Figure 4, we followed the implementation of GoICP [33] and used the 3D Euclidean distance transform (DT) from [14] with a $300 \times 300 \times 300$ grid; this provides a very fast approximation of the closest point computation, compared with the standard (accurate) KD-tree based computation. For evaluating the complexity as a function of accuracy in Figure 3(a), we used both GoICP and qBnB without DT transform.

5. Future work and conclusions

We presented the qBnB framework for globally optimizing D-quasi-optimizable functions, and demonstrated theoretically and empirically the advantage of this framework over existing BnB algorithm. Future challenges include applying the qBnB framework to other rigid registration problems that can handle outliers better than the standard ℓ^2 energy, as well as using this framework for global optimization of D-quasi-optimizable functions in other knowledge domains.

Acknowledgments The authors would like to thank Ingrid Daubechies for several helpful discussions. This research was supported by Simons Math+X Investigators Award 400837, NSF CAREER Award BCS-1552848, and NSF DBI-1759839.

References

[1] The morphosource database, https://www.morphosource.org/. 7

[2] Dror Aiger, Niloy J. Mitra, and Daniel Cohen-Or. 4-points congruent sets for robust pairwise surface registration. In *ACM Transactions on Graphics (TOG)*, volume 27, page 85. Acm, 2008. 2

[3] Reema Al-Aifari, Ingrid Daubechies, and Yaron Lipman. Continuous procrustes distance between two surfaces. *Communications on Pure and Applied Mathematics*, 66(6):934–964, 2013. 1

[4] Florian Bernard, Nikos Vlassis, Peter Gemmar, Andreas Husch, Johan Thunberg, Jorge Goncalves, and Frank Hertel. Fast correspondences for statistical shape models of brain structures. In *Medical Imaging 2016: Image Processing*, volume 9784, page 97840R. International Society for Optics and Photonics, 2016. 7

[5] Dimitri P. Bertsekas. *Network optimization: continuous and discrete models*. Athena Scientific Belmont, 1998. 7

[6] Paul J. Besl and Neil D. McKay. Method for registration of 3-d shapes. In *Sensor Fusion IV: Control Paradigms and Data Structures*, volume 1611, pages 586–607. International Society for Optics and Photonics, 1992. 1

[7] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004. 2

[8] Doug M. Boyer, Jesus Puente, Justin T. Gladman, Chris Glynn, Sayan Mukherjee, Gabriel S. Yapuncich, and Ingrid Daubechies. A new fully automated approach for aligning and comparing shapes. *The Anatomical Record*, 298(1):249–276, 2015. 7

[9] Thomas M. Breuel. Implementation techniques for geometric branch-and-bound matching methods. *Computer Vision and Image Understanding*, 90(3):258–294, 2003. 2, 4

[10] Álvaro Parra Bustos, Tat-Jun Chin, Anders Eriksson, Hongdong Li, and David Suter. Fast rotation search with stereographic projections for 3d registration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(11):2227–2240, 2016. 2

[11] Marek Cygan, Fedor V. Fomin, Łukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michał Pilipczuk, and Saket Saurabh. *Parameterized algorithms*, volume 4. Springer, 2015. 2

[12] Pedro de Siracusa. The phylopic database. http://phylopic.org/. 6

[13] Nadav Dym and Yaron Lipman. Exact recovery with symmetries for procrustes matching. *SIAM Journal on Optimization*, 27(3):1513–1530, 2017. 2

[14] Andrew W. Fitzgibbon. Robust registration of 2d and 3d point sets. *Image and vision computing*, 21(13-14):1145–1153, 2003. 7

[15] Jaroslav M. Fowkes, Nicholas I. M. Gould, and Chris L. Farmer. A branch and bound algorithm for the global optimization of hessian lipschitz continuous functions. *Journal of Global Optimization*, 56(4):1791–1815, 2013. 2

[16] Natasha Gelfand, Niloy J. Mitra, Leonidas J. Guibas, and Helmut Pottmann. Robust global registration. In *Symposium on geometry processing*, volume 2, page 5. Vienna, Austria, 2005. 2

[17] Daniela Giorgi, Silvia Biasotti, and Laura Paraboschi. Shape retrieval contest 2007: Watertight models track. *SHREC competition*, 8(7), 2007. 6

[18] Richard I. Hartley and Fredrik Kahl. Global optimization through rotation space search. *International Journal of Computer Vision*, 82(1):64–79, 2009. 2, 7, 13

[19] Sandy Irani and Prabhakar Raghavan. Combinatorial and experimental results for randomized point matching algorithms. *Computational Geometry*, 12(1-2):17–31, 1999. 2

[20] Yuehaw Khoo and Ankur Kapoor. Non-iterative rigid 2d/3d point-set registration using semidefinite programming. *IEEE Transactions on Image Processing*, 25(7):2956–2970, 2016. 2

[21] Hongdong Li and Richard Hartley. The 3d-3d registration problem revisited. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8. IEEE, 2007. 1, 2, 3, 4, 6, 7

[22] Haggai Maron, Nadav Dym, Itay Kezurer, Shahar Kovalsky, and Yaron Lipman. Point registration via efficient convex relaxation. *ACM Transactions on Graphics (TOG)*, 35(4):73, 2016. 2

[23] David M. Mount, Nathan S. Netanyahu, and Jacqueline Le Moigne. Efficient algorithms for robust feature matching. *Pattern recognition*, 32(1):17–38, 1999. 2, 4

[24] Alvaro Parra Bustos, Tat-Jun Chin, and David Suter. Fast rotation search with stereographic projections for 3d registration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3930–3937, 2014. 2

[25] Frank Pfeuffer, Michael Stiglmayr, and Kathrin Klamroth. Discrete and geometric branch and bound algorithms for medical image registration. *Annals of Operations Research*, 196(1):737–765, 2012. 2, 4

[26] Helmut Pottmann, Qi-Xing Huang, Yong-Liang Yang, and Shi-Min Hu. Geometry and convergence analysis of algorithms for registration of 3d shapes. *International Journal of Computer Vision*, 67(3):277–296, 2006. 2

[27] Jesus Puente. Distances and algorithms to compare sets of shapes for automated biological morphometrics. 2013. 7

[28] Anand Rangarajan, Haili Chui, and Fred L Bookstein. The softassign procrustes matching algorithm. In *Biennial International Conference on Information Processing in Medical Imaging*, pages 29–42. Springer, 1997. 1

[29] Romeil Sandhu, Samuel Dambreville, and Allen Tannenbaum. Point set registration via particle filtering and stochastic dynamics. *IEEE transactions on pattern analysis and machine intelligence*, 32(8):1459–1473, 2010. 2

[30] Gary KL Tam, Zhi-Quan Cheng, Yu-Kun Lai, Frank C. Langbein, Yonghuai Liu, David Marshall, Ralph R. Martin, Xian-Fang Sun, and Paul L. Rosin. Registration of 3d point clouds and meshes: a survey from rigid to nonrigid. *IEEE transactions on visualization and computer graphics*, 19(7):1199–1217, 2013. 2

[31] Greg Turk and Marc Levoy. The stanford 3d scanning repository. http://graphics.stanford.edu/data/3Dscanrep/. 6

[32] Mark P. Wachowiak, Renata Smolíková, Yufeng Zheng, Jacek M. Zurada, and Adel Said Elmaghraby. An approach to multimodal biomedical image registration utilizing particle swarm optimization. *IEEE Transactions on evolutionary computation*, 8(3):289–301, 2004. 2

[33] Jiaolong Yang, Hongdong Li, Dylan Campbell, and Yunde Jia. Go-icp: A globally optimal solution to 3d icp point-set registration. *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)*, 38(11):2241–2254, 2016. 2, 3, 4, 6, 7, 14

[34] Jiaolong Yang, Hongdong Li, and Yunde Jia. Go-icp: Solving 3d registration efficiently and globally optimally. In *Proceedings of the 14th International Conference on Computer Vision (ICCV)*, pages 1457–1464, 2013. 2

[35] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Fast global registration. In *European Conference on Computer Vision*, pages 766–782. Springer, 2016. 2