

Western University

Scholarship@Western

Electrical and Computer Engineering
Publications

Electrical and Computer Engineering
Department

Winter 2-1-2022

A New Approach for Grasp Quality Calculation using Continuous Boundary Formulation of Grasp Wrench Space

Shuwei Qiu

Shuwei Qiu Ph.D., P.Eng.

Western University, mkermani@eng.uwo.ca

Follow this and additional works at: <https://ir.lib.uwo.ca/electricalpub>



Part of the [Computer Engineering Commons](#), and the [Electrical and Computer Engineering Commons](#)

Citation of this paper:

Qiu, Shuwei and Qiu, Shuwei Ph.D., P.Eng., "A New Approach for Grasp Quality Calculation using Continuous Boundary Formulation of Grasp Wrench Space" (2022). *Electrical and Computer Engineering Publications*. 555.

<https://ir.lib.uwo.ca/electricalpub/555>

A New Approach for Grasp Quality Calculation using Continuous Boundary Formulation of Grasp Wrench Space

Shuwei Qiu*, Mehrdad R. Kermani

Electrical and Computer Engineering, Western University, London, Ontario, Canada

Abstract

In this paper, we aim to use a continuous formulation to efficiently calculate the well-known wrench-based grasp metric proposed by Ferrari and Canny which is the minimum distance from the wrench space origin to the boundary of the grasp wrench space. Considering the L_∞ metric and the nonlinear friction cone model, the challenge of calculating this metric is to determine the boundary of the grasp wrench space. Instead of relying on convex hull construction, we propose to formulate the boundary of the grasp wrench space as continuous functions. By doing so, the problem of grasp quality calculation can be efficiently solved as typical least-square problems and it can be easily implemented by employing off-the-shelf optimization algorithms. Numerical tests will demonstrate the advantages of the proposed formulation compared to the conventional convex hull-based methods.

Keywords: Grasp Quality Calculation; Boundary of Grasp Wrench Space; Continuous Formulation; L_∞ Metric; Nonlinear Friction Cone Model

1. Introduction

In the field of robotic grasping, grasp planning is a fundamental topic. Given a target object to be grasped, the purpose of grasp planning is to appropriately plan the contact points and the contact normals on the object's surface to balance contact forces and other external forces such as gravity. The topic of grasp planning contains many problems, for instance, the problem of finding a suitable set of contact points given the target object and some grasp constraints (i.e., the grasp synthesis problem), the problem of determining if a given grasp is force/form closure (i.e., the force/form closure determination problem), the problem of quantifying the goodness of a given grasp (i.e., the grasp quality calculation problem), and so on. The focus of this paper is on the problem of grasp quality calculation. We consider the hard-finger contact model in this paper, but the proposed method can be extended to the soft-finger contact model as well.

For a target object, there are usually numerous possibilities to grasp it [1]. To find the desired grasp among many potentials, it is necessary to evaluate and rank the quality of these grasps. To define the quality of a given grasp, many metrics from different perspectives have been proposed. For example, there are metrics related to the grasp matrix (G), such as the minimum singular value of G and the grasp isotropy index. There are also metrics considering the geometric relationships in the grasp, such as the shape and the area of the grasp polygon. The robot configurations can be used as a grasp quality measure. The distance to singular configurations [2] and the volume of the manipulability ellipsoid [3] are examples of such measures.

*This research was supported in part through funds received from the Natural Sciences and Engineering Research Council of Canada, Canada Foundation for Innovation, and Ontario Centres of Excellence.

*Corresponding author

Email addresses: sqiu47@uwo.ca (Shuwei Qiu), mkerman2@uwo.ca (Mehrdad R. Kermani)

By combining different grasp quality measures, one can also quantify the goodness of a grasp in a global scale [4]. Due to limited space, we refer the readers to [5] for a comprehensive overview of grasp metrics.

Among different grasp metrics, those related to the grasp wrench space (GWS) are undoubtedly among the most popular ones. The seminal work of GWS-based grasp metrics was proposed by Ferrari and Canny [6], which is often referred to as “ Q -distance”. Q -distance quantifies the goodness of force closure grasps. Since at least three contact points are required to achieve force closure in spatial problems, Q -distance is a suitable measure for multi-fingered robotic hands and precision grasping. Q -distance is conceptualized by following an intuitive approach which is to calculate the ratio between the magnitude of the maximum wrench this given grasp can resist along with all directions and the magnitude of the applied finger forces [6]. A wrench (denoted by $\vec{w} \in \mathbb{R}^6$) is a vector obtained by stacking a force vector $\vec{f} \in \mathbb{R}^3$ and a torque vector $\vec{t} \in \mathbb{R}^3$,

$$\vec{w} = \begin{bmatrix} \vec{f}^\top, \vec{t}^\top \end{bmatrix}^\top \quad (1)$$

whose magnitude is defined as

$$\|\vec{w}\| = \sqrt{\|\vec{f}\|^2 + \lambda\|\vec{t}\|^2} \quad (2)$$

where λ is a scaling factor. In this work, we use the L_2 metric for $\|\vec{w}\|$ (i.e., $\lambda = 1$). λ can also be defined differently for other purposes, for instance, for unifying force and torque units [7] and for removing the reference frame dependence [8]. To ensure a stable grasp, the contact force on each contact point must lie in the Coulomb friction cone (denoted by FC) expressed in the contact frame,

$$FC = \left\{ [f_1, f_2, f_n]^\top \mid f_n \geq 0, \sqrt{f_1^2 + f_2^2} \leq \mu f_n \right\} \quad (3)$$

where f_1 and f_2 are the tangential force components, f_n is the normal force component along the contact normal direction, and μ is the tangential friction coefficient. A contact force applied to the i -th contact point (denoted by \vec{f}_i) can be converted into a wrench by the corresponding grasp matrix (denoted by G_i) as $\vec{w}_i = G_i \vec{f}_i$. All possible wrenches that can be applied through the i -th contact point (i.e., the i -th wrench set, denoted by W_i) is then formulated as,

$$W_i = \left\{ G_i \vec{f}_i \mid \vec{f}_i \in FC \right\} \quad (4)$$

To find the set of wrenches that can be exerted on the object, Ferrari and Canny [6] introduced two criterias. One limits the maximum magnitude of each normal contact force to 1, i.e., the L_∞ metric. Using the L_∞ metric, the set of all possible wrenches acting on the object (i.e., the grasp wrench space) is,

$$W_{L_\infty} = \bigoplus_{i=1}^{n_c} W_i \quad (5)$$

where \bigoplus is the Minkowski summation and n_c is the number of contact points. The other criteria limits the summation of the magnitudes of all contact normal forces to 1, i.e., the L_1 metric. Using the L_1 metric, the grasp wrench space is,

$$W_{L_1} = \bigcup_{i=1}^{n_c} W_i \quad (6)$$

where \bigcup is the union operation. The value of Q -distance in a geometrical sense is the distance from the origin of the 6-D wrench space (i.e., $\vec{0}_{6 \times 1}$) to the boundary of the grasp wrench space. Ferrari and Canny also proposed to construct the convex hull of the grasp wrench space to calculate the value of Q -distance as follows. First, the primitive wrench sets are generated by approximating the friction cone as an m -sided

pyramid. Second, the grasp wrench space (W_{L_∞} or W_{L_1}) is approximated by constructing the convex hull. Third, the minimum distance between the origin and the facets of the convex hull is obtained.

Calculating the value of Q -distance is challenging in practice, especially with the L_∞ metric. The reason is twofold. First, the convex hull construction only works for finite sets. Second, the convex hull construction and the calculation of the distances from the origin to the convex hull’s facets are computationally expensive [9]. As a consequence, there are two common practices in the calculation of Q -distance. First, the Coulomb friction cone is linearized by an m -sided pyramid [10, 11]. Second, the L_1 metric is considered rather than the L_∞ metric. However, these two common practices have their own drawbacks. The linear friction cone model has two major disadvantages [12]. The solution obtained from the linear model may conflict with the one obtained from the nonlinear model. Also, the computation time will be significantly increased when increasing the number of the pyramid’s facets in the linear model. As for the L_1 metric, it only suits the robotic hands with under-actuated fingers [13]. If the L_1 metric is used for the robotic hands whose fingers are empowered by independent actuators, the actual capabilities of the grasp would be severely underestimated which may mislead the further operations and analysis [14].

In this paper, we study how to efficiently calculate the value of Q -distance considering both the L_∞ metric and the nonlinear friction cone model. To this end, we formulate the boundary of the grasp wrench space with continuous functions. By doing so, the value of Q -distance can be solved as a least-square problem and it can be easily implemented by employing off-the-shelf optimization algorithms.

The contributions of this paper are listed as follows.

- We mathematically derive the exact expression of the boundary of grasp wrench space.
- Based on the derived boundary formulation, we propose a continuous formulation for the calculation of Q -distance [6] considering the L_∞ metric and the nonlinear model of the Coulomb friction cone with the hard finger contact model.
- We show that it is easy to implement the proposed formulation by employing existing optimization algorithms.

The rest of this paper is structured as follows. [Section 2](#) reviews previous works related to Q -distance calculation. [Section 3](#) mathematically derives the continuous boundary formulation of grasp wrench space. [Section 4](#) presents the numerical results. Finally, [Section 5](#) concludes this work.

2. Related Works

In this section, we briefly review the works related to the calculation of Q -distance [6], which are summarized in [Table 1](#). Following the suggested approach of Ferrari and Canny [6], Miller and Allen [15] provided examples of calculating Q -distance with the L_1 metric by employing the open-sourced Qhull algorithm [16]. They invoked the Qhull algorithm to construct the convex hull of the grasp wrench space and calculated the distance between all facets of the convex hull to the 6-D origin. The approach was computationally expensive, so to expedite the computational speed, some improvements were proposed. Borst *et al.* [17] proposed to incrementally construct the convex hull of the grasp wrench space. In this approach, the convex hull was constructed starting from a coarse approximation of the friction cone and the nearest facet of this convex hull to the origin was attained. To compensate for the approximation error, additional wrenches with the largest possible distance to the previously obtained facet were incrementally added to the wrench set spanning the convex hull. The nearest facet to the origin and the quality measure were then updated. The procedure was repeated until the value of the quality measure reached below a preset threshold. Zhu and Wang [18] provided an approximation of Q -distance by considering the distance between the origin and the boundary of grasp wrench space along with finite directions instead of all directions. Based on the Qhull algorithm, Liu and Carpin [19] proposed a solution to calculate Q -distance while constructing the convex hull to avoid building the entire convex hull of the grasp wrench space. Dai *et al.* [20] showed that the value of Q -distance with the L_1 metric can be approximated by solving a semi-definite programming problem. With the L_1 metric, Pokorny and Kragic [21] investigated some theoretical properties of Q -distance and proposed an algorithm to compute an upper bound of Q -distance which can be used to efficiently reject

Table 1: Related Works on Q -Distance Calculation

| Methods | Friction Cone Model | Normal Force Constraint | Highlights |
|---------------------------|---------------------|-------------------------|---------------------------------------------------------------------------------------------------------------------------|
| Miller and Allen [15] | Linear | L_1 metric | Invoke the Qhull algorithm [16] to calculate Q -distance |
| Borst <i>et al.</i> [17] | Linear | L_∞ metric | Incrementally construct the convex hull of the grasp wrench space |
| Zhu and Wang [18] | Linear | L_∞ metric | Consider the distance between the origin and the boundary of GWS along with finite directions |
| Liu and Carpin [19] | Linear | L_1 and L_∞ | Intertwine Q -distance calculation with convex hull construction |
| Dai <i>et al.</i> [20] | Linear | L_1 metric | Approximate Q -distance by semi-definite programming |
| Pokorny and Kragic [21] | Linear | L_1 metric | Investigate theoretical properties of Q -distance and compute an upper bound |
| Harada <i>et al.</i> [22] | Ellipsoidal | L_∞ metric | Evaluate the grasp stability under gravity with soft finger contact model |
| Krug <i>et al.</i> [14] | Linear | L_1 and L_∞ | Investigate the influence of the L_1 and L_∞ metric to a fully actuated robotic hand |
| Zheng and Qian [8] | Nonlinear | L_1 and L_∞ | Cast the problem of Q -distance calculation as a nonlinear optimization problem |
| Zheng [23] | Nonlinear | L_∞ metric | Iteratively enlarge a polytope inside GWS and contains the origin by calculating its support function and support mapping |

unstable grasps. Using an ellipsoidal approximation of the friction cone, Harada *et al.* [22] took the soft finger contact model into account and evaluated the stability of a given grasp under gravity. Krug *et al.* [14] investigated the influence of the L_1 and L_∞ metric to wrench-based grasp quality indexes. They showed that the capability of a grasp executed by a fully actuated robotic hand would be severely underestimated if the L_1 metric is applied.

The above-mentioned works [14, 15, 17–22] have a common limitation of simplifying the nonlinear friction cone with either an m -sided pyramid or an ellipsoid. Many works have been proposed to consider the nonlinear friction cone model. Zheng and Qian [8] formulated the problem of Q -distance calculation as a nonlinear optimization problem by means of the concept of support function. To improve the computational efficiency, Zheng [23] proposed an improvement of Borst’s method [17] to calculate Q -distance with the nonlinear friction cone model. Starting from a polytope in the grasp wrench space containing the origin, Zheng’s method [23] iteratively enlarged this polytope by calculating its support function and support mapping. By doing so, the minimum distance between the origin and the polytope boundary could quickly converge to the value of Q -distance. Zheng’s method [23] only works when the origin is contained within the grasp wrench space and is complex to be implemented.

In this work, we calculate the value of Q -distance using a geometric approach. We regard all forces and torques that can be applied at each contact point as two solid objects in 3D geometric space and the grasp wrench space (GWS) as a 6D convex object. We then parameterize the boundary of GWS and calculate Q -distance as typical least-square problems. The details of the proposed method are explained next.

3. Continuous Boundary Formulation of Grasp Wrench Space

3.1. Problem Formulation

The problem we target is to calculate the wrench-based grasp quality metric proposed by Ferrari and Canny [6] (the Q -distance) with the L_∞ metric and the nonlinear friction cone. We propose to solve this problem based on the geometric interpretation of Q -distance. Q -distance is geometrically interpreted as the

shortest distance from the six-dimensional origin ($\vec{0}_{6 \times 1}$) to the boundary of the wrench space composed of all possible wrenches that can be generated by the given grasp (i.e., the grasp wrench space). Assuming a grasp with n_c contact points, the corresponding grasp matrices with a hard finger model (denoted by G_i , $i = 1, \dots, n_c$), and the friction coefficient (denoted by μ), one can formulate the problem of Q -distance calculation as,

$$\min_{\vec{w} \in bd(W_{L_\infty})} \frac{1}{2} \vec{w}^\top \vec{w} \quad (7)$$

where W_{L_∞} is the grasp wrench space with the L_∞ metric and $bd(W_{L_\infty})$ is the boundary of W_{L_∞} . Assuming μ is invariant over the object's surface, the challenge of solving Eq. (7) is the determination of $bd(W_{L_\infty})$.

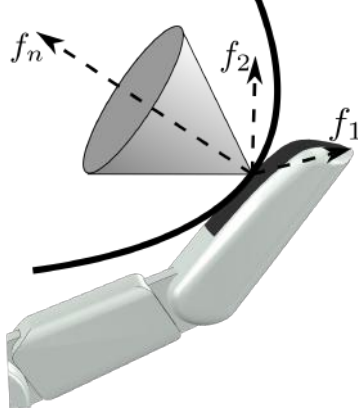


Figure 1: 3D convex cone expressed in the local contact frame

We consider the determination of $bd(W_{L_\infty})$ as a geometric problem. With the hard finger model, the friction cone constraint defines a second-order cone expressed in the local contact frame (see Fig. 1) as,

$$FC = \left\{ [f_1, f_2, f_n]^\top \mid f_n \in [0, h], \sqrt{f_1^2 + f_2^2} \leq \mu f_n \right\} \quad (8)$$

where h is the height of this cone acting as the maximum value of the normal force component. All forces that can be applied at the i -th contact point (i.e., the i -th force set, denoted by F_i) constitute a 3D convex cone expressed in the inertial frame,

$$F_i = \left\{ R_i \vec{f}_i \mid \vec{f}_i \in FC \right\} = \left\{ R_i [f_{i1}, f_{i2}, f_{in}]^\top \mid f_{in} \in [0, h], \sqrt{f_{i1}^2 + f_{i2}^2} \leq \mu f_{in} \right\}, \quad i = 1, \dots, n_c \quad (9)$$

where $R_i \in \mathbb{R}^{3 \times 3}$ is the orientation of the i -th contact frame with respect to the inertial frame. All torques that can be applied at the i -th contact point (i.e., the i -th torque set, denoted by T_i) constitute a convex object obtained from FC after a linear transformation,

$$T_i = \left\{ S_i R_i \vec{t}_i \mid \vec{t}_i \in FC \right\} = \left\{ S_i R_i [t_{i1}, t_{i2}, t_{in}]^\top \mid t_{in} \in [0, h], \sqrt{t_{i1}^2 + t_{i2}^2} \leq \mu t_{in} \right\}, \quad i = 1, \dots, n_c \quad (10)$$

where $S_i = \text{Skew}(\vec{p}_i - \vec{c})$ is the cross product matrix, in that \vec{p}_i represents the i -th contact point and \vec{c} represents the object's center of mass. Based on the definition of the wrench vector as per Eq. (1), all wrenches that can be applied at the i -th contact point (i.e., the i -th wrench set, denoted by W_i) constitute a 6D convex object which is the Cartesian product of F_i (a 3D convex cone) and T_i (a 3D convex object),

$$\begin{aligned} W_i &= F_i \times T_i = \left\{ \begin{bmatrix} \vec{f} \\ \vec{t} \end{bmatrix} \mid \vec{f} \in F_i, \vec{t} \in T_i \right\}, \quad i = 1, \dots, n_c \\ &= \left\{ \begin{bmatrix} R_i [f_{i1}, f_{i2}, f_{in}]^\top \\ S_i R_i [t_{i1}, t_{i2}, t_{in}]^\top \end{bmatrix} \mid f_{in}, t_{in} \in [0, h], \sqrt{f_{i1}^2 + f_{i2}^2} \leq \mu f_{in}, \sqrt{t_{i1}^2 + t_{i2}^2} \leq \mu t_{in} \right\} \end{aligned} \quad (11)$$

With this geometric interpretation, W_{L_∞} is a 6D convex object obtained as the Minkowski sum of multiple 6D convex objects (W_i) as $W_{L_\infty} = \bigoplus_{i=1}^{n_c} W_i$. In what follows, the proposed continuous formulation of $bd(W_{L_\infty})$ is derived.

3.2. *Boundary Decomposition:* $bd(W_{L_\infty}) = bd(W_{L_\infty})_1 \cup bd(W_{L_\infty})_2$

To reduce the difficulty of formulating $bd(W_{L_\infty})$, we first decompose $bd(W_{L_\infty})$ into two components. Regarding F_i and T_i as independent convex objects and substituting $W_i = F_i \times T_i$ in $W_{L_\infty} = \bigoplus_{i=1}^{n_c} W_i$, we obtain,

$$W_{L_\infty} = \bigoplus_{i=1}^{n_c} W_i = \bigoplus_{i=1}^{n_c} (F_i \times T_i) = \bigoplus_{i=1}^{n_c} F_i \times \bigoplus_{i=1}^{n_c} T_i \quad (12)$$

where we use the fact that the Minkowski sum is distributive over Cartesian product [24, 25]. Since F_i and T_i are both closed convex objects in 3D geometric space, $\bigoplus_{i=1}^{n_c} F_i$ and $\bigoplus_{i=1}^{n_c} T_i$ are also 3D closed objects [26]. An example of the Minkowski sum of two 3D cones is shown in Fig. 2.

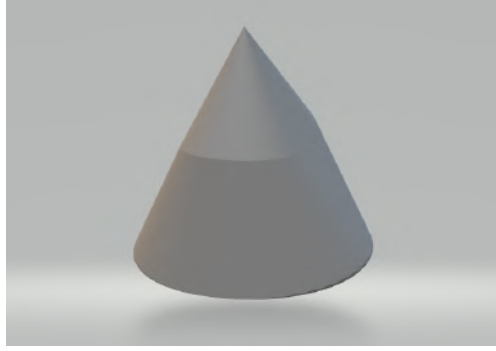


Figure 2: An example of the Minkowski sum of two 3D cones [27]

From Eq. (12), $bd(W_{L_\infty})$ is obtained as,

$$bd(W_{L_\infty}) = bd\left(\bigoplus_{i=1}^{n_c} F_i \times \bigoplus_{i=1}^{n_c} T_i\right) = \underbrace{\left[bd\left(\bigoplus_{i=1}^{n_c} F_i\right) \times \bigoplus_{i=1}^{n_c} T_i\right]}_{bd(W_{L_\infty})_1} \cup \underbrace{\left[\bigoplus_{i=1}^{n_c} F_i \times bd\left(\bigoplus_{i=1}^{n_c} T_i\right)\right]}_{bd(W_{L_\infty})_2} \quad (13)$$

where we apply the product rule for the boundary of the Cartesian product of closed sets (see Appendix A for proof). A 3D example of Eq. (13) is shown in Fig. 3. Geometrically, a solid cylinder is the Cartesian product of a disk and a line segment (see Fig. 3(a)). The boundary of a cylinder can be obtained by applying the product rule (see Fig. 3(b)).

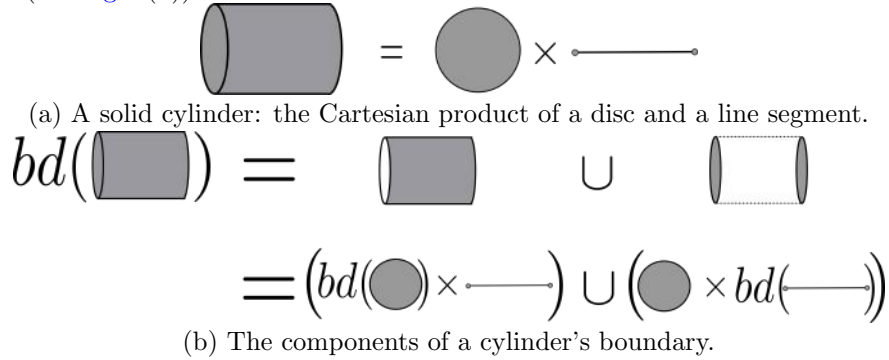


Figure 3: The boundary of a cylinder: a 3D example of Eq. (13).

As seen in Eq. (13), $bd(W_{L_\infty})$ is decomposed into two components, $bd(W_{L_\infty})_1$ and $bd(W_{L_\infty})_2$. Note that $bd(W_{L_\infty})_1$ and $bd(W_{L_\infty})_2$ are not fully disjoint as they both contain the component $[bd(\bigoplus_{i=1}^{n_c} F_i) \times bd(\bigoplus_{i=1}^{n_c} T_i)]$. But decomposing $bd(W_{L_\infty})$ as in Eq. (13), we can make use of the second-order cone definition of the friction cone Eq. (8) to formulate $\bigoplus_{i=1}^{n_c} T_i$ and $\bigoplus_{i=1}^{n_c} F_i$ as shown in Eq. (22) and Eq. (24), respectively. Using Eq. (13), Eq. (7) can be re-written as,

$$\begin{aligned} \min_{\vec{w} \in bd(W_{L_\infty})} \frac{1}{2} \vec{w}^\top \vec{w} &\Leftrightarrow \min(d_1, d_2) \\ d_1 &= \min_{\vec{w}_1 \in bd(W_{L_\infty})_1} \frac{1}{2} \vec{w}_1^\top \vec{w}_1 \\ d_2 &= \min_{\vec{w}_2 \in bd(W_{L_\infty})_2} \frac{1}{2} \vec{w}_2^\top \vec{w}_2 \end{aligned} \quad (14)$$

In what follows, we will explain the formulation of $bd(W_{L_\infty})_1$ and $bd(W_{L_\infty})_2$.

3.3. Formulation of $bd(W_{L_\infty})_1$

In this section, $bd(W_{L_\infty})_1$ is formulated. Since $bd(W_{L_\infty})_1 = bd(\bigoplus_{i=1}^{n_c} F_i) \times \bigoplus_{i=1}^{n_c} T_i$ as per Eq. (13), we need to formulate $bd(\bigoplus_{i=1}^{n_c} F_i)$ (i.e., the boundary of the Minkowski sum of force sets) and $\bigoplus_{i=1}^{n_c} T_i$ (i.e., the Minkowski sum of torque sets).

Geometrically, the boundary of the Minkowski sum of convex objects is obtained by only adding up the points on each object's boundary with the same outward normal directions [28, 29]. A 2D example of this principle is shown in Fig. 4 in that the boundary of two squares' Minkowski sum is obtained by combining the sides with the same outward normal directions. Following this principle, $bd(\bigoplus_{i=1}^{n_c} F_i)$ is formulated as,

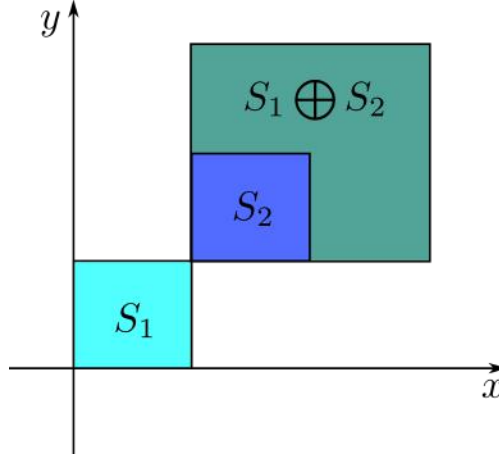


Figure 4: The Minkowski sum of two squares

$$bd\left(\bigoplus_{i=1}^{n_c} F_i\right) = \left\{ \sum_{i=1}^{n_c} \vec{b}_{F_i} \mid \vec{b}_{F_i} \in bd(F_i), \vec{n}_{b_{F_1}} = \dots = \vec{n}_{b_{F_{n_c}}} \right\} \quad (15)$$

where $bd(F_i)$ is the boundary of F_i and $\vec{n}_{b_{F_i}}$ is the unit outward normal vector on $bd(F_i)$ ($i = 1, \dots, n_c$). Since $F_i = \{R_i \vec{f}_i \mid \vec{f}_i \in FC\}$, $bd(F_i)$ is obtained as,

$$bd(F_i) = \left\{ R_i \vec{b}_{FC} \mid \vec{b}_{FC} \in bd(FC) \right\} \quad (16)$$

where $bd(FC)$ (the boundary of the local friction cone) is determined by the boundary condition of the Coulomb friction law (i.e., $\sqrt{f_1^2 + f_2^2} = \mu f_n$) and is parameterized as,

$$bd(FC) = \left\{ \begin{bmatrix} \mu f_n \cos \theta \\ \mu f_n \sin \theta \\ f_n \end{bmatrix} \mid f_n \in [0, h], \theta \in [0, 2\pi) \right\} \quad (17)$$

To attain \vec{n}_{bF_i} ($i = 1, \dots, n_c$), we first obtain the unit outward normal vector on $bd(FC)$ (denoted by \vec{v}) as [30, Chapter 3],

$$\begin{aligned}\vec{v} &= \left(\frac{\partial bd(FC)}{\partial f_n} \times \frac{\partial bd(FC)}{\partial \theta} \right) / \left\| \frac{\partial bd(FC)}{\partial f_n} \times \frac{\partial bd(FC)}{\partial \theta} \right\| \\ &= \frac{1}{\mu f_n \sqrt{\mu^2 + 1}} \begin{bmatrix} \mu f_n \cos \theta \\ \mu f_n \sin \theta \\ -\mu^2 f_n \end{bmatrix} = \frac{1}{\sqrt{\mu^2 + 1}} \begin{bmatrix} \cos \theta \\ \sin \theta \\ -\mu \end{bmatrix}\end{aligned}\quad (18)$$

Alternatively, the same expression of \vec{v} can be obtained by,

$$\begin{aligned}BF(f_1, f_2, f_n) &:= f_1^2 + f_2^2 - \mu^2 f_n^2 = 0 \\ \vec{v} &= \frac{Grad(BF)}{\|Grad(BF)\|} \Big|_{\substack{f_1 = \mu f_n \cos \theta \\ f_2 = \mu f_n \sin \theta}}\end{aligned}$$

where $BF(f_1, f_2, f_n)$ is defined as an implicit function of the local friction cone's boundary whose gradient is denoted by $Grad(BF)$. Since $bd(F_i) = \left\{ R_i \vec{b}_{FC} \mid \vec{b}_{FC} \in bd(FC) \right\}$ ($i = 1, \dots, n_c$), we characterize \vec{n}_{bF_i} from \vec{v} using different parameters (θ_{bF_i}) as,

$$\vec{n}_{bF_i} = R_i \vec{v}(\theta_{bF_i}) = \frac{R_i}{\sqrt{\mu^2 + 1}} \begin{bmatrix} \cos \theta_{bF_i} \\ \sin \theta_{bF_i} \\ -\mu \end{bmatrix}\quad (19)$$

Furthermore, from the condition that \vec{n}_{bF_1} and \vec{n}_{bF_i} pointing in the same direction, we can obtain,

$$\begin{aligned}\vec{n}_{bF_i} &= \vec{n}_{bF_1}, \quad i = 2, \dots, n_c \\ \Leftrightarrow \frac{R_i}{\sqrt{\mu^2 + 1}} \begin{bmatrix} \cos \theta_{bF_i} \\ \sin \theta_{bF_i} \\ -\mu \end{bmatrix} &= \frac{R_1}{\sqrt{\mu^2 + 1}} \begin{bmatrix} \cos \theta_{bF_1} \\ \sin \theta_{bF_1} \\ -\mu \end{bmatrix} \\ \Leftrightarrow \begin{bmatrix} \cos \theta_{bF_i} \\ \sin \theta_{bF_i} \\ -\mu \end{bmatrix} &= R_i^\top R_1 \begin{bmatrix} \cos \theta_{bF_1} \\ \sin \theta_{bF_1} \\ -\mu \end{bmatrix} \\ \Rightarrow \begin{cases} \cos \theta_{bF_i} = \left({}^i r_{11} - \frac{{}^i r_{12} {}^i r_{31}}{{}^i r_{32}} \right) \cos \theta_{bF_1} + \left(\frac{{}^i r_{12} ({}^i r_{33} - 1)}{{}^i r_{32}} - {}^i r_{13} \right) \mu \\ \sin \theta_{bF_i} = \left({}^i r_{21} - \frac{{}^i r_{22} {}^i r_{31}}{{}^i r_{32}} \right) \cos \theta_{bF_1} + \left(\frac{{}^i r_{22} ({}^i r_{33} - 1)}{{}^i r_{32}} - {}^i r_{23} \right) \mu \end{cases}\end{aligned}\quad (20)$$

where ${}^i r_{jk}$ ($j, k = 1, 2, 3$) is the entry of the matrix $R_i^\top R_1$ at j -th row and k -th column.

In view of Eqs. (15), (16) and (17), we parameterize $bd\left(\bigoplus_{i=1}^{n_c} F_i\right)$ with θ_{bF_1} and f_{i_n} ($i = 1, \dots, n_c$) as,

$$bd\left(\bigoplus_{i=1}^{n_c} F_i\right) = \left\{ R_1 \begin{bmatrix} \mu f_{1_n} \cos \theta_{bF_1} \\ \mu f_{1_n} \sin \theta_{bF_1} \\ f_{1_n} \end{bmatrix} + \sum_{i=2}^{n_c} R_i \begin{bmatrix} \mu f_{i_n} \cos \theta_{bF_i} \\ \mu f_{i_n} \sin \theta_{bF_i} \\ f_{i_n} \end{bmatrix} \mid f_{1_n}, f_{i_n} \in [0, h], \theta_{bF_1} \in [0, 2\pi] \right\}\quad (21)$$

where $\cos \theta_{bF_i}$ and $\sin \theta_{bF_i}$ ($i = 2, \dots, n_c$) are functions about $\cos \theta_{bF_1}$ as per Eq. (20).

To formulate $bd(W_{L_\infty})_1$, we also need to formulate $\bigoplus_{i=1}^{n_c} T_i$. Based on Eq. (10), $\bigoplus_{i=1}^{n_c} T_i$ is formulated as,

$$\bigoplus_{i=1}^{n_c} T_i = \left\{ \sum_{i=1}^{n_c} S_i R_i \vec{t}_i \mid \vec{t}_i \in FC \right\} = \left\{ \sum_{i=1}^{n_c} S_i R_i \begin{bmatrix} t_{i1} \\ t_{i2} \\ t_{in} \end{bmatrix} \mid t_{in} \in [0, h], \sqrt{t_{i1}^2 + t_{i2}^2} \leq \mu t_{in} \right\}\quad (22)$$

where t_{i_1}, t_{i_2} , and t_{i_n} ($i = 1, \dots, n_c$) are the tangent and normal components expressed in the i -th contact frame, respectively.

Having formulated $bd(\bigoplus_{i=1}^{n_c} F_i)$ and $\bigoplus_{i=1}^{n_c} T_i$, $bd(W_{L_\infty})_1$ is formulated as,

$$bd(W_{L_\infty})_1 = \left\{ \begin{bmatrix} \vec{f} \\ \vec{t} \end{bmatrix} \mid \vec{f} \in bd\left(\bigoplus_{i=1}^{n_c} F_i\right), \vec{t} \in \bigoplus_{i=1}^{n_c} T_i \right\} \quad (23)$$

where the expressions of $bd(\bigoplus_{i=1}^{n_c} F_i)$ and $\bigoplus_{i=1}^{n_c} T_i$ are formulated as per Eqs. (21) and (22), respectively.

3.4. Formulation of $bd(W_{L_\infty})_2$

In this section, $bd(W_{L_\infty})_2$ is formulated. The procedure of formulating $bd(W_{L_\infty})_2$ is similar with that of $bd(W_{L_\infty})_1$. To formulate $bd(W_{L_\infty})_2$, we need to formulate $\bigoplus_{i=1}^{n_c} F_i$ (i.e., the Minkowski sum of force sets) and $bd(\bigoplus_{i=1}^{n_c} T_i)$ (i.e., the boundary of the Minkowski sum of torque sets) since $bd(W_{L_\infty})_2 = \bigoplus_{i=1}^{n_c} F_i \times bd(\bigoplus_{i=1}^{n_c} T_i)$ as per Eq. (13).

From Eq. (9), $\bigoplus_{i=1}^{n_c} F_i$ is formulated as,

$$\bigoplus_{i=1}^{n_c} F_i = \left\{ \sum_{i=1}^{n_c} R_i \vec{f}_i \mid \vec{f}_i \in FC \right\} = \left\{ \sum_{i=1}^{n_c} R_i \begin{bmatrix} f_{i_1} \\ f_{i_2} \\ f_{i_n} \end{bmatrix} \mid f_{i_n} \in [0, h], \sqrt{f_{i_1}^2 + f_{i_2}^2} \leq \mu f_{i_n} \right\} \quad (24)$$

Analogous to $bd(\bigoplus_{i=1}^{n_c} F_i)$, $bd(\bigoplus_{i=1}^{n_c} T_i)$ is also obtained by only adding up the points on the boundary of torque sets (denoted by $bd(T_i)$, $i = 1, \dots, n_c$) with the same outward normal directions (\vec{n}_{bT_i}) [28, 29] (see Fig. 4 for a graphical example),

$$bd\left(\bigoplus_{i=1}^{n_c} T_i\right) = \left\{ \sum_{i=1}^{n_c} \vec{b}_{T_i} \mid \vec{b}_{T_i} \in bd(T_i), \vec{n}_{bT_1} = \dots = \vec{n}_{bT_{n_c}} \right\} \quad (25)$$

where

$$bd(T_i) = \left\{ S_i R_i \vec{b}_{FC} \mid \vec{b}_{FC} \in bd(FC) \right\} \quad (26)$$

And \vec{n}_{bT_i} is obtained from \vec{v} Eq. (18) as,

$$\vec{n}_{bT_i} = S_i R_i \vec{v}(\theta_{bT_i}) = \frac{S_i R_i}{\sqrt{\mu^2 + 1}} \begin{bmatrix} \cos \theta_{bT_i} \\ \sin \theta_{bT_i} \\ -\mu \end{bmatrix} \quad (27)$$

From \vec{n}_{bT_1} and \vec{n}_{bT_i} ($i = 2, \dots, n_c$) pointing in the same direction, we can obtain,

$$\begin{aligned} & \vec{n}_{bT_i} = \vec{n}_{bT_1}, \quad i = 2, \dots, n_c \\ & \Leftrightarrow \frac{S_i R_i}{\sqrt{\mu^2 + 1}} \begin{bmatrix} \cos \theta_{bT_i} \\ \sin \theta_{bT_i} \\ -\mu \end{bmatrix} = \frac{S_1 R_1}{\sqrt{\mu^2 + 1}} \begin{bmatrix} \cos \theta_{bT_1} \\ \sin \theta_{bT_1} \\ -\mu \end{bmatrix} \\ & \Leftrightarrow \begin{bmatrix} \cos \theta_{bT_i} \\ \sin \theta_{bT_i} \\ -\mu \end{bmatrix} = (S_i R_i)^{-1} S_1 R_1 \begin{bmatrix} \cos \theta_{bT_1} \\ \sin \theta_{bT_1} \\ -\mu \end{bmatrix} \\ & \Rightarrow \begin{cases} \cos \theta_{bT_i} = \left({}^i s_{11} - \frac{{}^i s_{12} {}^i s_{31}}{{}^i s_{32}} \right) \cos \theta_{bT_1} + \left(\frac{{}^i s_{12} ({}^i s_{33} - 1)}{{}^i s_{32}} - {}^i s_{13} \right) \mu \\ \sin \theta_{bT_i} = \left({}^i s_{21} - \frac{{}^i s_{22} {}^i s_{31}}{{}^i s_{32}} \right) \cos \theta_{bT_1} + \left(\frac{{}^i s_{22} ({}^i s_{33} - 1)}{{}^i s_{32}} - {}^i s_{23} \right) \mu \end{cases} \quad (28) \end{aligned}$$

where ${}^i s_{jk}$ ($j, k = 1, 2, 3$) is the entry of the matrix $(S_i R_i)^{-1} S_1 R_1$ at j -th row and k -th column.

In view of Eqs. (17), (25) and (26), $bd(\bigoplus_{i=1}^{n_c} T_i)$ is parameterized with θ_{bT_1} and t_{i_n} ($i = 1, \dots, n_c$) as,

$$bd\left(\bigoplus_{i=1}^{n_c} T_i\right) = \left\{ S_1 R_1 \begin{bmatrix} \mu t_{1_n} \cos \theta_{bT_1} \\ \mu t_{1_n} \sin \theta_{bT_1} \\ t_{1_n} \end{bmatrix} + \sum_{i=2}^{n_c} S_i R_i \begin{bmatrix} \mu t_{i_n} \cos \theta_{bT_i} \\ \mu t_{i_n} \sin \theta_{bT_i} \\ t_{i_n} \end{bmatrix} \mid t_{1_n}, t_{i_n} \in [0, h], \theta_{bT_1} \in [0, 2\pi) \right\} \quad (29)$$

where $\cos \theta_{bT_i}$ and $\sin \theta_{bT_i}$ ($i = 2, \dots, n_c$) are functions of $\cos \theta_{bT_1}$ as per Eq. (28).

Having formulated $\bigoplus_{i=1}^{n_c} F_i$ and $bd(\bigoplus_{i=1}^{n_c} T_i)$, $bd(W_{L_\infty})_2$ is formulated as,

$$bd(W_{L_\infty})_2 = \left\{ \begin{bmatrix} \vec{f} \\ \vec{t} \end{bmatrix} \mid \vec{f} \in \bigoplus_{i=1}^{n_c} F_i, \vec{t} \in bd\left(\bigoplus_{i=1}^{n_c} T_i\right) \right\} \quad (30)$$

where $\bigoplus_{i=1}^{n_c} F_i$ is as per Eq. (24) and $bd(\bigoplus_{i=1}^{n_c} T_i)$ is as per Eq. (29).

3.5. The Proposed Q -Distance Calculation Method

Having formulated the exact expression of $bd(W_{L_\infty})_1$ and $bd(W_{L_\infty})_2$, we finalize the formulation of Q -distance calculation based on Eq. (14). To obtain meaningful values of the Q -distance and keep the generality of the proposed solution, we set the height of the local friction cone as 1 (i.e., $h = 1$). The calculation of Q -distance is formulated as,

$$Q = \sqrt{2d_{min}}, \quad d_{min} = \min(d_1, d_2) \quad (31)$$

$$d_1 = \min_{\vec{w}_1 \in bd(W_{L_\infty})_1} \frac{1}{2} \vec{w}_1^\top \vec{w}_1 \quad (32)$$

$$d_2 = \min_{\vec{w}_2 \in bd(W_{L_\infty})_2} \frac{1}{2} \vec{w}_2^\top \vec{w}_2 \quad (33)$$

Since $bd(W_{L_\infty})_1 = bd(\bigoplus_{i=1}^{n_c} F_i) \times \bigoplus_{i=1}^{n_c} T_i$ and $bd(W_{L_\infty})_2 = \bigoplus_{i=1}^{n_c} F_i \times bd(\bigoplus_{i=1}^{n_c} T_i)$ as per Eq. (13), we can expand Eqs. (32) and (33) as,

$$d_1 = \min_{\substack{\vec{f}_1 \in bd(\bigoplus_{i=1}^{n_c} F_i) \\ \vec{t}_1 \in \bigoplus_{i=1}^{n_c} T_i}} \frac{1}{2} \begin{bmatrix} \vec{f}_1^\top & \vec{t}_1^\top \end{bmatrix} \begin{bmatrix} \vec{f}_1 \\ \vec{t}_1 \end{bmatrix} = \underbrace{\min_{\vec{f}_1 \in bd(\bigoplus_{i=1}^{n_c} F_i)} \frac{1}{2} \vec{f}_1^\top \vec{f}_1}_{d_{1f}} + \underbrace{\min_{\vec{t}_1 \in \bigoplus_{i=1}^{n_c} T_i} \frac{1}{2} \vec{t}_1^\top \vec{t}_1}_{d_{1t}} \quad (34)$$

$$d_2 = \min_{\substack{\vec{f}_2 \in \bigoplus_{i=1}^{n_c} F_i \\ \vec{t}_2 \in bd(\bigoplus_{i=1}^{n_c} T_i)}} \frac{1}{2} \begin{bmatrix} \vec{f}_2^\top & \vec{t}_2^\top \end{bmatrix} \begin{bmatrix} \vec{f}_2 \\ \vec{t}_2 \end{bmatrix} = \underbrace{\min_{\vec{f}_2 \in \bigoplus_{i=1}^{n_c} F_i} \frac{1}{2} \vec{f}_2^\top \vec{f}_2}_{d_{2f}} + \underbrace{\min_{\vec{t}_2 \in bd(\bigoplus_{i=1}^{n_c} T_i)} \frac{1}{2} \vec{t}_2^\top \vec{t}_2}_{d_{2t}} \quad (35)$$

As observed, the calculation of Q -distance is now formulated as four independent least-square problems which are detailed as follows.

Based on the formulation of $bd(\bigoplus_{i=1}^{n_c} F_i)$ as per Eq. (21), the minimization problem for solving the value of d_{1f} is formulated as,

$$d_{1f} = \min \frac{1}{2} \vec{f}_1^\top \vec{f}_1 \quad (36)$$

$$\vec{f}_1 = f_{11} R_1 \begin{bmatrix} \mu \cos \theta_{bF_1} \\ \mu \sin \theta_{bF_1} \\ 1 \end{bmatrix} + \sum_{i=2}^{n_c} f_{1i} R_i \begin{bmatrix} \mu \cos \theta_{bF_i} \\ \mu \sin \theta_{bF_i} \\ 1 \end{bmatrix}$$

$$\cos \theta_{bF_i} = \left({}^i r_{11} - \frac{{}^i r_{12} {}^i r_{31}}{{}^i r_{32}} \right) \cos \theta_{bF_1} + \left(\frac{{}^i r_{12} ({}^i r_{33} - 1)}{{}^i r_{32}} - {}^i r_{13} \right) \mu, \text{ for } i > 1$$

$$\sin \theta_{bF_i} = \left({}^i r_{21} - \frac{{}^i r_{22} {}^i r_{31}}{{}^i r_{32}} \right) \cos \theta_{bF_1} + \left(\frac{{}^i r_{22} ({}^i r_{33} - 1)}{{}^i r_{32}} - {}^i r_{23} \right) \mu, \text{ for } i > 1$$

$$\text{s.t. } \theta_{bF_i} \in [0, 2\pi), f_{1i} \in [0, 1], i = 1, \dots, n_c$$

where θ_{bF_1} and f_{1i} ($i = 1, \dots, n_c$) are the decision variables, where $R_i \in \mathbb{R}^{3 \times 3}$ is the orientation of the i -th contact frame with respect to the inertial frame, ${}^i r_{jk}$ ($j, k = 1, 2, 3$) is the entry of the matrix $R_i^\top R_1$ at j -th row and k -th column, and μ is the tangential friction coefficient.

Based on the formulation of $\bigoplus_{i=1}^{n_c} T_i$ as per Eq. (22), the minimization problem for solving the value of d_{1t} is formulated as,

$$\begin{aligned} d_{1t} &= \min \frac{1}{2} \vec{t}_1^\top \vec{t}_1 \\ \vec{t}_1 &= \sum_{i=1}^{n_c} S_i R_i \begin{bmatrix} t_{1i_1} \\ t_{1i_2} \\ t_{1i_n} \end{bmatrix} \\ \text{s.t. } t_{1i_n} &\in [0, 1], \sqrt{t_{1i_1}^2 + t_{1i_2}^2} \leq \mu t_{1i_n}, \quad i = 1, \dots, n_c \end{aligned} \quad (37)$$

where t_{1i_1} , t_{1i_2} and t_{1i_n} ($i = 1, \dots, n_c$) are the decision variables, and $S_i = \text{Skew}(\vec{p}_i - \vec{c})$ is the cross product matrix, in that \vec{p}_i represents the i -th contact point and \vec{c} represents the object's center of mass.

Based on the formulation of $\bigoplus_{i=1}^{n_c} F_i$ as per Eq. (24), the minimization problem for solving the value of d_{2f} is formulated as,

$$\begin{aligned} d_{2f} &= \min \frac{1}{2} \vec{f}_2^\top \vec{f}_2 \\ \vec{f}_2 &= \sum_{i=1}^{n_c} R_i \begin{bmatrix} f_{2i_1} \\ f_{2i_2} \\ f_{2i_n} \end{bmatrix} \\ \text{s.t. } f_{2i_n} &\in [0, 1], \sqrt{f_{2i_1}^2 + f_{2i_2}^2} \leq \mu f_{2i_n}, \quad i = 1, \dots, n_c \end{aligned} \quad (38)$$

where f_{2i_1} , f_{2i_2} and f_{2i_n} ($i = 1, \dots, n_c$) are the decision variables.

Based on the formulation of $bd(\bigoplus_{i=1}^{n_c} T_i)$ as per Eq. (29), the minimization problem for solving the value of d_{2t} is formulated as,

$$\begin{aligned} d_{2t} &= \min \frac{1}{2} \vec{t}_2^\top \vec{t}_2 \\ \vec{t}_2 &= t_{21} S_1 R_1 \begin{bmatrix} \mu \cos \theta_{bT_1} \\ \mu \sin \theta_{bT_1} \\ 1 \end{bmatrix} + \sum_{i=2}^{n_c} t_{2i} S_i R_i \begin{bmatrix} \mu \cos \theta_{bT_i} \\ \mu \sin \theta_{bT_i} \\ 1 \end{bmatrix} \\ \cos \theta_{bT_i} &= \left({}^i s_{11} - \frac{{}^i s_{12} {}^i s_{31}}{{}^i s_{32}} \right) \cos \theta_{bT_1} + \left(\frac{{}^i s_{12} ({}^i s_{33} - 1)}{{}^i s_{32}} - {}^i s_{13} \right) \mu, \quad \text{for } i > 1 \\ \sin \theta_{bT_i} &= \left({}^i s_{21} - \frac{{}^i s_{22} {}^i s_{31}}{{}^i s_{32}} \right) \cos \theta_{bT_1} + \left(\frac{{}^i s_{22} ({}^i s_{33} - 1)}{{}^i s_{32}} - {}^i s_{23} \right) \mu, \quad \text{for } i > 1 \\ \text{s.t. } \theta_{bT_1} &\in [0, 2\pi), \quad t_{2i} \in [0, 1], \quad i = 1, \dots, n_c \end{aligned} \quad (39)$$

where θ_{bT_1} and t_{2i} ($i = 1, \dots, n_c$) are the decision variables, and ${}^i s_{jk}$ ($j, k = 1, 2, 3$) is the entry of the matrix $(S_i R_i)^{-1} S_1 R_1$ at j -th row and k -th column.

Remark. Note that trivial solutions exist in the above minimization problems (i.e., $f_{1i} = 0$ for Eq. (36), $t_{1i_n} = 0$ for Eq. (37), $f_{2i_n} = 0$ for Eq. (38), and $t_{2i} = 0$ for Eq. (39) ($i = 1, \dots, n_c$)). To exclude the trivial solutions in these minimization problems, one should use a small value (e.g., 1×10^{-3}), instead of 0, as the lower bound for the decision variables.

The calculation of d_{1f} in Eq. (36) and d_{2t} in Eq. (39) are the minimization of trigonometric polynomials, which can be solved by constrained nonlinear optimization algorithms. The calculation of d_{1t} in Eq. (37)

and d_{2f} in Eq. (38) are second-order cone programming problems, which can be efficiently solved by second-order cone programming algorithms. In addition, since these four least-square problems are independent, the computational speed can be further increased by implementing parallel computing techniques if available. The algorithm for calculating the value of Q -distance using the above formulation is provided in Algorithm 1 in the form of pseudo-code.

Algorithm 1: Q -Distance Calculation with Continuous Boundary Formulation

Input : contact points ($\vec{p}_i, i = 1, \dots, n_c$), contact normals (\vec{n}_i), center of mass (\vec{c}), and the friction coefficient (μ)

Output: the value of Q -distance (Q)

```

/* Step 1: Compute grasp matrices */
1  $[G_i, R_i, S_i] \leftarrow$  Get Grasp Matrices( $\vec{p}_i, \vec{n}_i, \vec{c}$ ), ( $i = 1, \dots, n_c$ )
/* Step 2: Define the minimization problem for solving  $d_{1f}$  */
2  $[\theta_{bF_1}, f_{1i}] \leftarrow$  Define Decision Variables, ( $i = 1, \dots, n_c$ )
3  $C_1 = [0 \leq \theta_{bF_1} < 2\pi, 0 \leq f_{1i} \leq 1]$ , ( $i = 1, \dots, n_c$ )// Define constraints
4  $CF_1 \leftarrow$  Define Cost Function// see Eq. (36)
/* Step 3: Define the minimization problem for solving  $d_{1t}$  */
5  $[t_{1i_1}, t_{1i_2}, t_{1i_n}] \leftarrow$  Define Decision Variables, ( $i = 1, \dots, n_c$ )
6  $C_2 = [0 \leq t_{1i_n} \leq 1, \sqrt{t_{1i_1}^2 + t_{1i_2}^2} \leq \mu t_{1i_n}]$ , ( $i = 1, \dots, n_c$ )// Define constraints
7  $CF_2 \leftarrow$  Define Cost Function// see Eq. (37)
/* Step 4: Define the minimization problem for solving  $d_{2f}$  */
8  $[f_{2i_1}, f_{2i_2}, f_{2i_n}] \leftarrow$  Define Decision Variables, ( $i = 1, \dots, n_c$ )
9  $C_3 = [0 \leq f_{1i_n} \leq 1, \sqrt{f_{1i_1}^2 + f_{1i_2}^2} \leq \mu f_{1i_n}]$ , ( $i = 1, \dots, n_c$ )// Define constraints
10  $CF_3 \leftarrow$  Define Cost Function// see Eq. (38)
/* Step 5: Define the minimization problem for solving  $d_{2t}$  */
11  $[\theta_{bT_1}, t_{2i}] \leftarrow$  Define Decision Variables, ( $i = 1, \dots, n_c$ )
12  $C_4 = [0 \leq \theta_{bT_1} < 2\pi, 0 \leq t_{2i} \leq 1]$ , ( $i = 1, \dots, n_c$ )// Define constraints
13  $CF_4 \leftarrow$  Define Cost Function// see Eq. (39)
/* Step 6: Solve least-square problems */
14  $d_{1f} \leftarrow$  Nonlinear Minimization( $CF_1, C_1$ )
15  $d_{1t} \leftarrow$  Second-order Cone Programming( $CF_2, C_2$ )
16  $d_{2f} \leftarrow$  Second-order Cone Programming( $CF_3, C_3$ )
17  $d_{2t} \leftarrow$  Nonlinear Minimization( $CF_4, C_4$ )
/* Step 7: Calculate the value of  $Q$ -distance */
18  $d_1 = d_{1f} + d_{1t}, d_2 = d_{2f} + d_{2t}$ 
19  $d_{min} = \min(d_1, d_2), Q = \sqrt{2d_{min}}$ 
20 Return  $Q$ 

```

4. Numerical Results

This section outlines the numerical tests designed to demonstrate the performance of the proposed solution in comparison with the conventional convex hull construction method implemented with the Qhull algorithm [16].

4.1. Implementation Details

Before presenting the results of numerical tests, we provide the details of the implemented methods. For the proposed solution explained in Section 3.5, the least-square problems Eq. (37) and Eq. (38) (i.e., the calculation of d_{1t} and d_{2f}) were solved by the “SeDuMi” algorithm [31] implemented with “YALMIP” [32] in MATLAB. The other two least-square problems Eq. (36) and Eq. (39) (i.e., the calculation of d_{1f} and d_{2t}) were solved by the MATLAB built-in function “fmincon” implemented with “YALMIP”. Both algorithms were using their default settings in “YALMIP”. Since YALMIP is not compatible with the Parallel Computing Toolbox of MATLAB, these least-square problems were solved in sequence rather than in parallel. For the conventional Qhull-based method, the calculation of Q -distance was performed in three steps. (1) Given unit

normal forces, the primitive wrench vectors at a contact point were generated by linearizing the friction cone as an m -sided pyramid. With different values of m , the Qhull-based methods are referred to as “Qhull- m ” hereafter. (2) Considering all combinations among primitive wrenches (i.e., the Minkowski sum of primitive wrenches), the convex hull of the grasp wrench space was constructed by invoking the Qhull program. (3) The distance from the origin to the facets of the constructed convex hull was calculated and the minimum value is regarded as the value of Q -distance.

4.2. Numerical Tests

All numerical tests were conducted using MATLAB r2019b on a laptop computer powered by an i5-5200U CPU @2.20GHz with 12GB RAM. The friction coefficient is assumed to be $\mu = 0.3$ in all tests. We consider the case of grasping a banana, a power drill, and a cleanser bottle with three, four, and five contact points, respectively. 100 force-closure grasps were randomly selected from the point cloud of each object from the famous YCB dataset [33] (see Fig. 5). The centroid of the point cloud was regarded as the center of mass in all tests. To calculate the value of Q -distance for these grasps, we implemented the proposed method

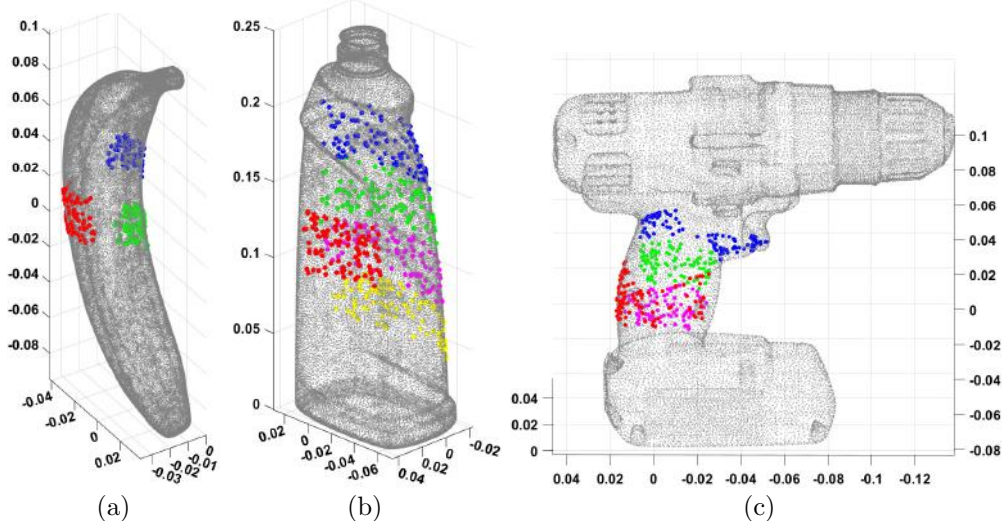


Figure 5: Different grasps on different objects (unit: meter). (a) 3-contact grasps on a banana. (b) 5-contact grasps on a cleanser bottle. (c) 4-contact grasps on a power drill. Red, blue, green, magenta, and yellow dots are the contact points for Finger 1, 2, 3, 4, and 5, respectively.

in comparison with Qhull-5 (i.e., the Qhull-based method using a 5-sided pyramid to linearize the friction cone), Qhull-8, Qhull-9, Qhull-10, Qhull-13, Qhull-15, and Qhull-17.

4.3. Results

The results are listed in Tables 2, 3 and 4, where SD stands for standard deviation. The proposed method outperformed Qhull-based methods dramatically in terms of computational speed. Efficient grasp quality evaluation is a critical component in the process of grasp planning, especially for real-time applications [34–36]. In the process of grasp planning, it is ordinary to synthesize and evaluate numerous grasp configurations using grasp quality measure(s) in an iterative process. A decreased evaluation time of a single grasp configuration is an important factor that can significantly improve the overall grasp planning efficiency. The proposed continuous boundary formulation enhances the grasp quality evaluation in two aspects. On the one hand, the proposed continuous boundary formulation makes it possible to utilize the efficiency of existing solvers and mathematical programming techniques. On the other hand, since the proposed calculation method is composed of four independent minimization problems, it enables the use of parallel computing techniques on individual processing units to further enhance the efficiency of the grasp quality evaluation. As for the values of Q -distance, the results from Qhull-based methods fluctuate significantly. For example, among the same potential 3-contact grasps for grasping a banana as shown in Fig. 5(a), the maximum value of Q -distance calculated by Qhull-9 (7.40×10^{-5}) is less than the minimum Q -distance value calculated by

Table 2: Results for 3-contact grasps

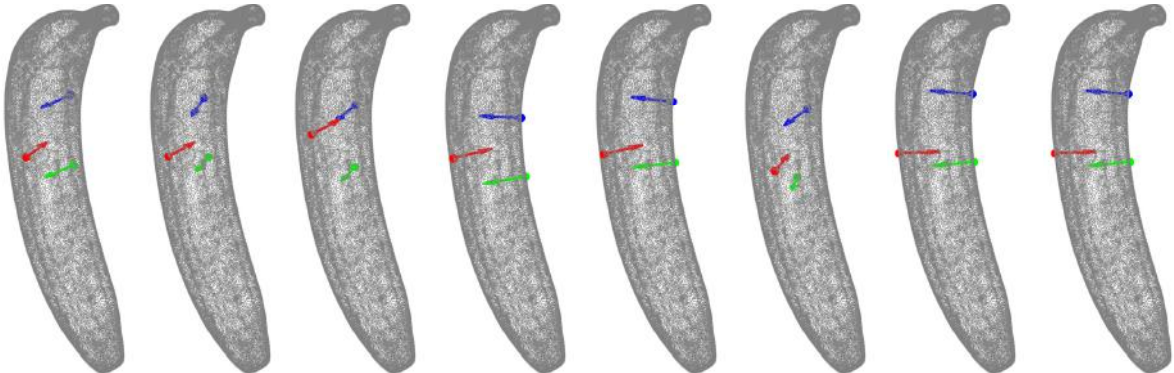
| | Q-distance values | | Computation time (s) | |
|----------|-----------------------|-----------------------|----------------------|--------|
| | Min | Max | Mean | SD |
| Proposed | 1.18×10^{-7} | 1.16×10^{-3} | 1.1091 | 0.6660 |
| Qhull-5 | 4.21×10^{-2} | 8.93×10^{-2} | 6.5833 | 0.6923 |
| Qhull-8 | 5.00×10^{-7} | 1.01×10^{-2} | 8.4916 | 1.4610 |
| Qhull-9 | 4.85×10^{-7} | 7.40×10^{-5} | 10.9036 | 1.1889 |
| Qhull-10 | 9.54×10^{-5} | 2.75×10^{-3} | 15.9925 | 3.0494 |
| Qhull-13 | 2.50×10^{-2} | 3.90×10^{-2} | 20.4811 | 4.0401 |
| Qhull-15 | 5.35×10^{-7} | 2.72×10^{-5} | 25.1361 | 5.1627 |
| Qhull-17 | 1.95×10^{-2} | 3.26×10^{-2} | 22.3091 | 4.8169 |

Table 3: Results for 4-contact grasps

| | Q-distance values | | Computation time (s) | |
|----------|-----------------------|-----------------------|----------------------|---------|
| | Min | Max | Mean | SD |
| Proposed | 1.41×10^{-7} | 1.26×10^{-3} | 1.2441 | 0.8372 |
| Qhull-5 | 2.19×10^{-2} | 1.40×10^{-1} | 10.8608 | 3.0631 |
| Qhull-8 | 5.01×10^{-7} | 2.30×10^{-2} | 16.8790 | 2.3686 |
| Qhull-9 | 5.33×10^{-7} | 2.82×10^{-4} | 17.8227 | 3.3504 |
| Qhull-10 | 7.38×10^{-7} | 1.58×10^{-2} | 28.6167 | 4.5378 |
| Qhull-13 | 1.31×10^{-2} | 7.58×10^{-2} | 19.6621 | 5.2444 |
| Qhull-15 | 1.31×10^{-2} | 7.58×10^{-2} | 43.3528 | 13.3126 |
| Qhull-17 | 1.04×10^{-2} | 5.62×10^{-2} | 24.7275 | 5.7986 |

Table 4: Results for 5-contact grasps

| | Q-distance values | | Computation time (s) | |
|----------|-----------------------|-----------------------|----------------------|----------|
| | Min | Max | Mean | SD |
| Proposed | 1.27×10^{-7} | 1.30×10^{-3} | 1.5835 | 1.1145 |
| Qhull-5 | 1.16×10^{-1} | 2.09×10^{-1} | 8.5327 | 2.3327 |
| Qhull-8 | 5.00×10^{-7} | 1.19×10^{-2} | 20.7646 | 5.0158 |
| Qhull-9 | 4.83×10^{-7} | 6.62×10^{-4} | 37.7040 | 10.8483 |
| Qhull-10 | 7.22×10^{-7} | 3.50×10^{-3} | 55.4412 | 20.7376 |
| Qhull-13 | 4.45×10^{-2} | 8.72×10^{-2} | 26.3182 | 9.1274 |
| Qhull-15 | 5.23×10^{-7} | 1.91×10^{-4} | 464.3587 | 484.6158 |
| Qhull-17 | 3.66×10^{-2} | 8.12×10^{-2} | 53.8731 | 24.1987 |



(a) Qhull-5 (b) Qhull-8 (c) Qhull-9 (d) Qhull-10 (e) Qhull-13 (f) Qhull-15 (g) Qhull-17 (h) Proposed

Figure 6: The best 3-contact grasp found by different methods. Red, blue, and green dots (arrows) are the contact points (normals) for Finger 1, 2, and 3, respectively.

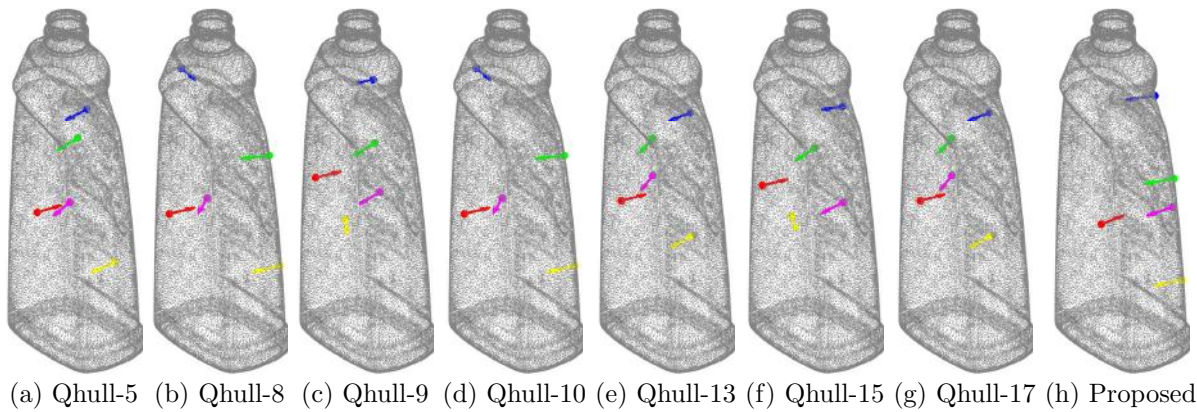


Figure 7: The best 5-contact grasp found by different methods. Red, blue, green, magenta, and yellow dots (arrows) are the contact points (normals) for Finger 1, 2, 3, 4, and 5, respectively.

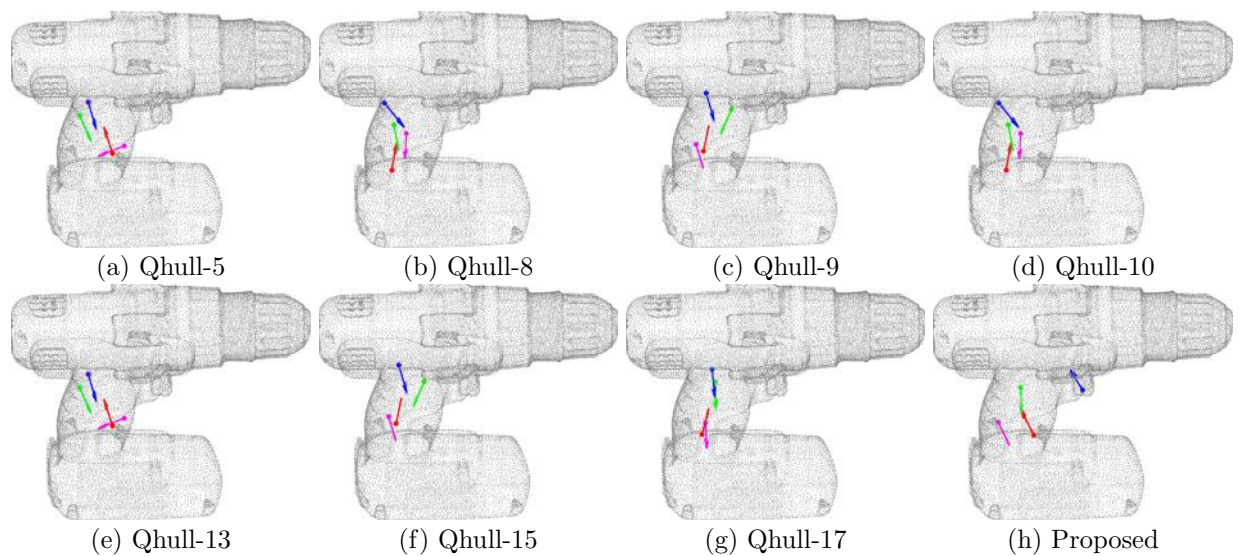


Figure 8: The best 4-contact grasp found by different methods. Red, blue, green, and magenta dots (arrows) are the contact points (normals) for Finger 1, 2, 3, and 4, respectively.

Qhull-10 (9.54×10^{-5}). We also exhibit the best grasp found by different methods (i.e., the grasps corresponding to the maximum Q -distance value calculated by different methods) in Figs. 6, 7 and 8. As seen, the best grasps found by different methods are not consistent. Thus, in the absence of ground truth solution and without the consideration of the hand structure in the Q -distance definition, it remains debatable which grasp configuration is better than the others. The stability of a planned grasp needs to be assessed when Q -distance is used for a specific task, due to its task-independent definition. This can be achieved using some benchmarks [33, 37, 38]. These results show that the number of pyramid’s sides (i.e., m) sways the result obtained from convex hull-based methods, and different choices of m affect not only the computation time but also the decision of grasp planning. The problem of selecting m stems from the sampling nature of convex hull construction. The input point set to the operation of convex hull construction has to be finite. As a consequence, the boundary of the wrench set at the i -th contact point (i.e., $bd(W_i)$, $i = 1, \dots, n_c$) can only be approximated by a limited number of points. Mathematically speaking, $bd(W_i)$ can be decomposed as,

$$\begin{aligned} bd(W_i) &= bd(F_i \times T_i) = [bd(F_i) \times T_i] \cup [F_i \times bd(T_i)] \\ &= \{bd(F_i) \times [int(T_i) \cup bd(T_i)]\} \cup \{[int(F_i) \cup bd(F_i)] \times bd(T_i)\} \\ &= [bd(F_i) \times int(T_i)] \cup [int(F_i) \times bd(T_i)] \cup [bd(F_i) \times bd(T_i)] \end{aligned} \quad (40)$$

where $int(F_i)$ and $int(T_i)$ denote the interior of F_i and T_i , respectively. As seen, $bd(W_i)$ is decomposed into three disjoint components. In convex hull-based methods, it is common to approximate $bd(W_i)$ using the primitive wrench set (denoted by PW_i) which is obtained by mapping the primitive force set (denoted by PF_i) using the grasp matrix (G_i) as,

$$PW_i = \left\{ G_i \vec{f} \mid \vec{f} \in PF_i \right\} = \left\{ \begin{bmatrix} R_i \\ S_i R_i \end{bmatrix} \vec{f} \mid \vec{f} \in PF_i \right\} \quad (41)$$

Since PF_i is sampled from the friction cone boundary (i.e., $PF_i \subset bd(FC)$), we can obtain,

$$\left\{ R_i \vec{f} \mid \vec{f} \in PF_i \right\} \subset bd(F_i), \quad \left\{ S_i R_i \vec{f} \mid \vec{f} \in PF_i \right\} \subset bd(T_i) \quad (42)$$

As a result, $PW_i \subset [bd(F_i) \times bd(T_i)]$ and PW_i only approximates a part of $[bd(F_i) \times bd(T_i)]$ which is only one component of $bd(W_i)$ as in Eq. (40), in that there is no consideration about the points in the other two components of $bd(W_i)$ (i.e., $[bd(F_i) \times int(T_i)]$ and $[int(F_i) \times bd(T_i)]$). Consequently, one can only conclude that in convex hull-based methods, the grasp wrench space boundary has been obtained using the points sampled from one component of $bd(W_i)$. As a comparison, we consider all the portions of $bd(W_i)$ ($i = 1, \dots, n$) in the proposed method by regarding F_i and T_i as solid objects in the 3D geometric space, and parameterize $bd(W_i)$ and $bd(W_{L_\infty})$ following geometric principles. Therefore, the proposed formulation of $bd(W_{L_\infty})$ is more concrete from a geometric perspective.

It is noteworthy to discuss a limitation of the calculation method presented in Section 3.5. In the proposed Q -distance calculation method, two nonlinear minimization problems are involved (i.e., Eqs. (36) and (39)) which were solved by a generic constrained nonlinear optimization algorithm (i.e., the “fmincon” function in MATLAB) in the current work. Consequently, it is not guaranteed to find the global minimum of these two problems (Eqs. (36) and (39)). However, the proposed calculation method can benefit from the future development of mathematical programming techniques since it does not require any specific algorithms or special-designed heuristic procedures.

5. Conclusions

In this paper, we formulated the boundary of grasp wrench space with continuous functions considering the L_∞ metric and the nonlinear friction cone model. With this new continuous formulation, the wrench-based grasp quality (Q -distance) [6] is calculated much more efficiently as typical least-square problems. The proposed method can be easily implemented by employing off-the-shelf optimization algorithms. In addition,

by regarding the force sets ($F_i, i = 1, \dots, n_c$) and the torque sets (T_i) as solid objects in 3D geometric space, this paper provides an entirely new and more concrete formulation for Q -distance calculation than the convex hull-based methods from a geometric perspective.

Future works include extending the proposed formulation to the soft finger contact model, implementing parallel computing techniques to further increase the computational speed, comparing the proposed method with more Q -distance calculation methods, and applying the proposed method in grasp planning applications with real robotic systems.

Appendix A. Boundary of Cartesian Product of Two Closed Sets

Given two closed sets (A and B) in two topological spaces (X and Y), we have $A \subseteq X$ and $B \subseteq Y$. The boundary of the Cartesian product of A and B , $bd(A \times B)$ is equal to $[bd(A) \times B] \cup [A \times bd(B)]$.

Proof. Since A and B are closed sets, their Cartesian product ($A \times B$) is also closed. One can obtain

$$bd(A \times B) = A \times B - int(A \times B) \quad (\text{A.1})$$

where $int(A \times B)$ denotes the interior points of $A \times B$. Since $int(A \times B) = intA \times intB$ where $intA$ and $intB$ are the interior of A and B , respectively, one can obtain

$$bd(A \times B) = A \times B - intA \times intB \quad (\text{A.2})$$

$$= (A \times B) \cap (intA \times intB)^c \quad (\text{A.3})$$

where we make use of the fact that $S_1 - S_2 = S_1 \cap (S_2)^c$ for two sets S_1 and S_2 , in that $(\cdot)^c$ is the complement of a set. Since $A \subseteq X$ and $B \subseteq Y$, one can obtain

$$(intA \times intB)^c = [(intA)^c \times Y] \cup [X \times (intB)^c] \quad (\text{A.4})$$

Substitute Eq. (A.4) into Eq. (A.3), one have

$$bd(A \times B) = (A \times B) \cap \{[(intA)^c \times Y] \cup [X \times (intB)^c]\} \quad (\text{A.5})$$

$$= \{(A \times B) \cap [(intA)^c \times Y]\} \cup \{(A \times B) \cap [X \times (intB)^c]\} \quad (\text{A.6})$$

Since $(S_1 \cap S_2) \times (S_3 \cap S_4) = (S_1 \times S_3) \cap (S_2 \times S_4)$ where S_i ($i = 1, 2, 3, 4$) are sets, one can finally obtain

$$bd(A \times B) = \{[A \cap (intA)^c] \times (B \cap Y)\} \cup \{(A \cap X) \times [B \cap (intB)^c]\} \quad (\text{A.7})$$

$$= [bd(A) \times B] \cup [A \times bd(B)] \quad (\text{A.8})$$

□

References

- [1] M. Abdeetetal and M. R. Kermani, "Grasp synthesis for purposeful fracturing of object," *Robotics and Autonomous Systems*, vol. 105, pp. 47–58, 2018.
- [2] C. A. Klein and B. E. Blaho, "Dexterity measures for the design and control of kinematically redundant manipulators," *The international journal of robotics research*, vol. 6, no. 2, pp. 72–83, 1987.
- [3] T. Yoshikawa, "Manipulability of robotic mechanisms," *The international journal of Robotics Research*, vol. 4, no. 2, pp. 3–9, 1985.
- [4] B. León, J. L. Sancho-Bru, N. J. Jarque-Bou, A. Morales, and M. A. Roa, "Evaluation of human prehension using grasp quality measures," *International Journal of Advanced Robotic Systems*, vol. 9, no. 4, p. 112, 2012.
- [5] M. A. Roa and R. Suárez, "Grasp quality measures: review and performance," *Autonomous robots*, vol. 38, no. 1, pp. 65–88, 2015.
- [6] C. Ferrari and J. F. Canny, "Planning optimal grasps." in *ICRA*, vol. 3, 1992, pp. 2290–2295.
- [7] M. A. Roa and R. Suárez, "Computation of independent contact regions for grasping 3-d objects," *IEEE Transactions on Robotics*, vol. 25, no. 4, pp. 839–850, 2009.
- [8] Y. Zheng and W.-H. Qian, "Improving grasp quality evaluation," *Robotics and Autonomous Systems*, vol. 57, no. 6-7, pp. 665–673, 2009.

- [9] H. Dong, C. Qiu, D. K. Prasad, Y. Pan, J. Dai, and I.-M. Chen, “Enabling grasp action: Generalized quality evaluation of grasp stability via contact stiffness from contact mechanics insight,” *Mechanism and Machine Theory*, vol. 134, pp. 625–644, 2019.
- [10] Y. Zheng and W.-H. Qian, “On some weaknesses existing in optimal grasp planning,” *Mechanism and Machine Theory*, vol. 43, no. 5, pp. 576–590, 2008.
- [11] G. Li, P. Xu, S. Qiao, and B. Li, “Stability analysis and optimal enveloping grasp planning of a deployable robotic hand,” *Mechanism and Machine Theory*, vol. 158, p. 104241, 2021.
- [12] L. Han, J. C. Trinkle, and Z. X. Li, “Grasp analysis as linear matrix inequality problems,” *IEEE Transactions on Robotics and Automation*, vol. 16, no. 6, pp. 663–674, 2000.
- [13] M. Abdeetetal and M. R. Kermani, “Grasp and stress analysis of an underactuated finger for proprioceptive tactile sensing,” *IEEE/ASME Transactions on Mechatronics*, vol. 23, no. 4, pp. 1619–1629, 2018.
- [14] R. Krug, Y. Bekiroglu, and M. A. Roa, “Grasp quality evaluation done right: How assumed contact force bounds affect wrench-based quality metrics,” in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 1595–1600.
- [15] A. T. Miller and P. K. Allen, “Examples of 3d grasp quality computations,” in *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C)*, vol. 2. IEEE, 1999, pp. 1240–1246.
- [16] C. B. Barber, D. P. Dobkin, and H. Huhdanpaa, “The quickhull algorithm for convex hulls,” *ACM Transactions on Mathematical Software (TOMS)*, vol. 22, no. 4, pp. 469–483, 1996.
- [17] C. Borst, M. Fischer, and G. Hirzinger, “A fast and robust grasp planner for arbitrary 3d objects,” in *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C)*, vol. 3. IEEE, 1999, pp. 1890–1896.
- [18] X. Zhu and J. Wang, “Synthesis of force-closure grasps on 3-d objects based on the q distance,” *IEEE Transactions on robotics and Automation*, vol. 19, no. 4, pp. 669–679, 2003.
- [19] S. Liu and S. Carpin, “Fast grasp quality evaluation with partial convex hull computation,” in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 4279–4285.
- [20] H. Dai, A. Majumdar, and R. Tedrake, “Synthesis and optimization of force closure grasps via sequential semidefinite programming,” in *Robotics Research*. Springer, 2018, pp. 285–305.
- [21] F. T. Pokorný and D. Kragic, “Classical grasp quality evaluation: New algorithms and theory,” in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 3493–3500.
- [22] K. Harada, T. Tsuji, S. Uto, N. Yamanobe, K. Nagata, and K. Kitagaki, “Stability of soft-finger grasp under gravity,” in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 883–888.
- [23] Y. Zheng, “An efficient algorithm for a grasp quality measure,” *IEEE Transactions on Robotics*, vol. 29, no. 2, pp. 579–585, 2012.
- [24] S. Liu and M. Zamani, “Compositional synthesis of almost maximally permissible safety controllers,” in *2019 American Control Conference (ACC)*. IEEE, 2019, pp. 1678–1683.
- [25] L. Gurvits, “A short proof, based on mixed volumes, of liggett’s theorem on the convolution of ultra-logconcave sequences,” *the electronic journal of combinatorics*, pp. N5–N5, 2009.
- [26] P. K. Ghosh, “A unified computational framework for minkowski operations,” *Computers & Graphics*, vol. 17, no. 4, pp. 357–378, 1993.
- [27] J.-M. Lien, “A simple method for computing minkowski sum boundary in 3d using collision detection,” in *Algorithmic foundation of robotics VIII*. Springer, 2009, pp. 401–415, available: <http://masc.cs.gmu.edu/wiki/SimpleMsum>.
- [28] M. Peternell and F. Manhart, “The convolution of a paraboloid and a parametrized surface,” *Journal for Geometry and Graphics*, vol. 7, no. 2, pp. 157–171, 2003.
- [29] M. Peternell and T. Steiner, “Minkowski sum boundary surfaces of 3d-objects,” *Graphical Models*, vol. 69, no. 3-4, pp. 180–190, 2007.
- [30] W. Kühnel, *Differential geometry*. American Mathematical Soc., 2015, vol. 77.
- [31] J. F. Sturm, “Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones,” *Optimization methods and software*, vol. 11, no. 1-4, pp. 625–653, 1999.
- [32] J. Lofberg, “Yalmip: A toolbox for modeling and optimization in matlab,” in *2004 IEEE international conference on robotics and automation (IEEE Cat. No. 04CH37508)*. IEEE, 2004, pp. 284–289.
- [33] B. Calli, A. Walsman, A. Singh, S. Srinivasa, P. Abbeel, and A. M. Dollar, “Benchmarking in manipulation research: The ycb object and model set and benchmarking protocols,” *arXiv preprint arXiv:1502.03143*, 2015.
- [34] S. Yao, M. Ceccarelli, G. Carbone, and Z. Dong, “Grasp configuration planning for a low-cost and easy-operation under-actuated three-fingered robot hand,” *Mechanism and Machine Theory*, vol. 129, pp. 51–69, 2018.
- [35] M. A. Roa, K. Hertkorn, C. Borst, and G. Hirzinger, “Reachable independent contact regions for precision grasps,” in *2011 IEEE International Conference on Robotics and Automation*. IEEE, 2011, pp. 5337–5343.
- [36] J. Shi and G. S. Koonjul, “Real-time grasping planning for robotic bin-picking and kitting applications,” *IEEE Transactions on Automation Science and Engineering*, vol. 14, no. 2, pp. 809–819, 2017.
- [37] Y. Bekiroglu, N. Marturi, M. A. Roa, K. J. M. Adjigble, T. Pardi, C. Grimm, R. Balasubramanian, K. Hang, and R. Stolkin, “Benchmarking protocol for grasp planning algorithms,” *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 315–322, 2019.
- [38] F. Bottarel, G. Vezzani, U. Pattacini, and L. Natale, “Graspa 1.0: Graspa is a robot arm grasping performance benchmark,” *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 836–843, 2020.