Available online at www.sciencedirect.com

SCIENCE DIRECT®

Mechanism and Machine Theory 41 (2006) 1243–1257

Mechanism and Machine Theory

www.elsevier.com/locate/mechmt

Limiting and minimizing the contact forces in multifingered grasping

Yu Zheng *, Wen-Han Qian

Robotics Institute, Shanghai Jiao Tong University, Shanghai 200030, PR China

Received 8 July 2005; received in revised form 25 October 2005; accepted 11 November 2005
Available online 15 March 2006

Abstract

Without any limit to the contact force magnitude, the traditional definition of force-closure caused controversy. As an essential modification to it, in the early 1990s two criteria for goodness were proposed without preference. The grasp quality is thus measured by the largest resultant wrench in the worst direction generated by the contact forces with the normal components equal to unity at most. The difference between the two criteria lies in computing the resultant wrench by either the convex combination or the Minkowski sum. If the required resultant wrench is known, then we pursue the least contact forces. Correspondingly, the difference becomes minimizing either the sum or the maximum of the normal components. Following the former a number of important papers on optimal grasp planning (OGP) and dynamic force distribution (DFD) appeared in recent years converging into the mainstream, but none referred to the latter. Differing from the current trend, this paper reveals that the former is not so good to evaluate the grasp goodness, whereas the latter is a good criterion indeed. For the first time the algorithms for computing the latter and furthermore for OGP and DFD are formulated. With these results as a standard, the exactness of the former when applied to OGP and DFD is estimated.

Keywords: Force-closure; Optimal grasp planning (OGP); Dynamic force distribution (DFD); Multifingered grasping; Minimax criterion

1. Introduction

Illuminated by the creative work of Salisbury and Roth [1], optimal grasp planning (OGP) and dynamic force distribution (DFD) have been hotly studied more than two decades. The former topic is to determine the optimal contact positions for achieving the best grasp according to some grasp quality criterion(-ria). The latter is to find the optimal contact forces for equilibrating the external wrench subject to some force optimality criterion(-ria).

Closure properties, including form-closure and force-closure, are prerequisite to stable grasping and primarily required for OGP. These properties mean the capability of a grasp to restrain any motion and to equilibrate any external wrench on the grasped object. So far several force-closure conditions have been deduced

* Corresponding author.

0094-114X/$ - see front matter © 2006 Elsevier Ltd. All rights reserved.
doi:10.1016/j.mechmachtheory.2005.11.001

[1–7], which lead to the closure test [8–11] and lay the foundation for constructing force-closure grasps [12–21]. However, the traditional force-closure definition [1–3] permits the contact forces to be arbitrarily large as long as they conform to the contact constraint. Certain grasps must exert very large contact forces (even exceeding the material strength and/or the actuator power) to equilibrate a small external wrench. Therefore, the force magnitude must be limited and the grasp quality criteria regarding it should be considered. Being another vital problem, DFD always concerns the contact force magnitude, which is taken as an objective to be minimized.

In multifingered grasping, there are several contacts, so we need a quantity for indicating the overall contact force magnitude. The previous literature proposed two measures: the *sum* of the normal force components and their *maximum*. Limiting the sum to unity, Kirkpatrick et al. [22], Ferrari and Canny [23] suggested the largest resultant wrench in the worst direction generated by the feasible contact forces as a grasp quality criterion. It is geometrically interpreted as the minimum distance from the origin of the wrench space to the boundary of the convex hull of the primitive wrenches. Zhu and his colleagues [24,25] utilized the Q distance to compute the criterion easier but probably lost a little precision. Liu et al. [9,21] computed a distance between the origin and the boundary of the convex hull in a certain direction with high efficiency, but it is usually not the minimum. Limiting the maximum, Ferrari and Canny [23] recommended an alternative quality criterion without comparing them, but no algorithm for OGP according to the latter was put forward until now. In DFD, since a linear programming (LP) algorithm [26] was raised to minimize the sum, a compact-dual LP algorithm [27], a quadratic programming (QP) algorithm [28], two gradient algorithms [29,30], an interior point algorithm [31], and a Newton algorithm [32] appeared one after another. Liu et al. [33,34] proved that some of the algorithms [29–32] are quadratically convergent. Linearizing the Coulomb friction cone, Liu [9] developed a ray-shooting based algorithm using the duality of polytopes. DFD algorithms can be also found in [35–38]. Particularly, Zheng and Qian minimized the maximum [38] rather than the sum [26–37].

Since summation is a linear operation and much easier, the *sum* of the normal force components is adopted as the mainstream measure of the overall contact force magnitude [9,21–36], whereas their *maximum* is laid aside as an alternative after its proposal [23,31]. On the contrary, this paper claims the advantages of the latter. More generally, we limit each normal force component by an upper bound, which may be different for different contacts. To the magnitude of each contact force, we prefer the ratio of the normal force component to its upper bound. Accordingly the overall value is measured by the sum or the maximum of the ratios. The extreme contact force is defined as a contact force reaching the friction cone boundary in inclination and the force upper bound in magnitude. Its image under mapping of the grasp matrix into the wrench space is called a primitive wrench. Following [23], when the sum is limited to unity, the grasp quality is assessed by the distance between the origin and the convex hull $\mathscr{W}^u_{c-}$ of the union of the primitive wrenches. To limit only the maximum, the convex hull $\mathscr{W}^M_{c-}$ of the Minkowski sum is used instead of $\mathscr{W}^u_{c-}$. A method of precisely computing the two criteria is derived. Detailed comparison reveals that the former is not so proper for evaluating the grasp goodness, while the latter is really competent. An OGP algorithm is developed with either criterion. Due to the distinct descent directions of the criteria, the resulted optimal grasps may differ. Furthermore, we propose a unified DFD algorithm for minimizing the sum or the maximum. The difference lies in that the algorithm is based on $\mathscr{W}^u_{c-}$ or $\mathscr{W}^M_{c-}$. Their computation results are often conspicuously apart.

2. Preliminary studies

Consider an m-fingered hand grasping an object, fixed with a right-handed coordinate frame. Assume that each finger contacts the object at a regular point with the Coulomb friction. Set a local right-handed coordinate frame at each contact point with the unit inward normal $\boldsymbol{n}_i$ and the unit tangent vector(s) $\boldsymbol{t}_i$ (and $\boldsymbol{o}_i$). The contact force $\boldsymbol{f}_i$ $(i = 1, 2, \ldots, m)$ can be expressed in the local coordinate frame by

$$\boldsymbol{f}_i = [f_{in} \quad f_{it}]^{\mathrm{T}} \in \mathbb{R}^2 \text{ (2D grasps) or } \boldsymbol{f}_i = [f_{in} \quad f_{io} \quad f_{it}]^{\mathrm{T}} \in \mathbb{R}^3 \text{ (3D grasps)}, \tag{1}$$

where f_{in}, f_{io}, and f_{it} are the force components along $\boldsymbol{n}_i$, $\boldsymbol{o}_i$, and $\boldsymbol{t}_i$, respectively. To avoid separation and slip at contact, $\boldsymbol{f}_i$ should comply with the *contact constraint*:

$$f_{in} \geqslant 0, \text{ and } -\mu_i f_{in} \leqslant f_{it} \leqslant \mu_i f_{in} \text{ (2D) or } \sqrt{f_{io}^2 + f_{it}^2} \leqslant \mu_i f_{in} \text{ (3D)}, \tag{2}$$

where μ_i is the Coulomb friction coefficient. A contact force under the contact constraint is said to be *feasible*. Let $\boldsymbol{s}_{i,h}$, $h = 1, 2, \ldots, l$ be the *extreme contact forces*, which characterize or linearize the friction cone:

$$\boldsymbol{s}_{i,1} = [f_i^{\mathrm{U}} \quad \mu_i f_i^{\mathrm{U}}]^{\mathrm{T}}, \quad \boldsymbol{s}_{i,2} = [f_i^{\mathrm{U}} \quad -\mu_i f_i^{\mathrm{U}}]^{\mathrm{T}} \text{ (2D) or } \boldsymbol{s}_{i,h} = f_i^{\mathrm{U}}[1 \quad \mu_i \cos(2h\pi/l) \quad \mu_i \sin(2h\pi/l)]^{\mathrm{T}} \text{ (3D)}, \tag{3}$$

where f_i^{U} is the upper bound of f_{in}, bounded by the material strength and/or the actuator power. Then a feasible $\boldsymbol{f}_i$ under f_i^{U} can be represented by

$$\boldsymbol{f}_i = \sum_{h=1}^{l} \lambda_{i,h}\boldsymbol{s}_{i,h}, \quad \lambda_{i,h} \geqslant 0 \text{ for } h = 1, 2, \ldots, l \text{ and } \sum_{h=1}^{l} \lambda_{i,h} \leqslant 1. \tag{4}$$

Moreover, combining (1), (3) and (4) leads to

$$f_{in} = f_i^{\mathrm{U}} \sum_{h=1}^{l} \lambda_{i,h}. \tag{5}$$

The overall contact force magnitude covering all contacts takes either of

$$\sigma^u = \sum_{i=1}^{m} \frac{f_{in}}{f_i^{\mathrm{U}}} = \sum_{i=1}^{m} \sum_{h=1}^{l} \lambda_{i,h}, \tag{6}$$

$$\sigma^M = \max_{1 \leqslant i \leqslant m} \left(\frac{f_{in}}{f_i^{\mathrm{U}}}\right) = \max_{1 \leqslant i \leqslant m} \left(\sum_{h=1}^{l} \lambda_{i,h}\right). \tag{7}$$

Then the wrench $\boldsymbol{w}_i \in \mathbb{R}^d$ generated by contact i can be calculated by

$$\boldsymbol{w}_i = \boldsymbol{G}_i\boldsymbol{f}_i = \sum_{h=1}^{l} \lambda_{i,h}\boldsymbol{G}_i\boldsymbol{s}_{i,h} = \sum_{h=1}^{l} \lambda_{i,h}\boldsymbol{w}_{i,h} = \boldsymbol{W}_i\lambda_i, \tag{8}$$

where $d = 3$ and $d = 6$ for 2D and 3D grasps, respectively; $\boldsymbol{G}_i$ is the grasp matrix; $\boldsymbol{w}_{i,h} = \boldsymbol{G}_i\boldsymbol{s}_{i,h}$ is called a *primitive wrench*; $\boldsymbol{W}_i = [\boldsymbol{w}_{i,1} \quad \boldsymbol{w}_{i,2} \quad \cdots \quad \boldsymbol{w}_{i,l}] \in \mathbb{R}^{d\times l}$ and $\boldsymbol{\lambda}_i = [\lambda_{i,1} \quad \lambda_{i,2} \quad \cdots \quad \lambda_{i,l}]^{\mathrm{T}} \in \mathbb{R}^l$. Let $\mathscr{W}_i$ and $\mathscr{W}_{ic}$ be the set of and the convex hull of $\boldsymbol{w}_{i,1}, \boldsymbol{w}_{i,2}, \ldots, \boldsymbol{w}_{i,l}$ together with the origin $\boldsymbol{0}$ of the wrench space

$$\mathscr{W}_i = \{\boldsymbol{0}, \boldsymbol{w}_{i,1}, \boldsymbol{w}_{i,2}, \ldots, \boldsymbol{w}_{i,l}\}, \tag{9}$$

$$\mathscr{W}_{ic} = \mathrm{conv}\{\boldsymbol{0}, \boldsymbol{w}_{i,1}, \boldsymbol{w}_{i,2}, \ldots, \boldsymbol{w}_{i,l}\} = \mathrm{conv}\,\mathscr{W}_i, \tag{10}$$

where conv(·) denotes the convex hull of a set. The set $\mathscr{W}_{ic}$ consists of all the wrenches that can be generated by feasible $\boldsymbol{f}_i$ under f_i^{U}. For 2D grasps, $\mathscr{W}_{ic}$ is a triangle with a vertex at $\boldsymbol{0}$ of the 3D wrench space. For 3D grasps, $\mathscr{W}_{ic}$ is a 3D polytope with a vertex at $\boldsymbol{0}$ of the 6D wrench space.

To equilibrate an external wrench $\boldsymbol{w}_{\mathrm{ext}}$, the resultant wrench $\boldsymbol{w}$ applied by the hand should be

$$\boldsymbol{w} = \sum_{i=1}^{m} \boldsymbol{G}_i\boldsymbol{f}_i = \sum_{i=1}^{m} \boldsymbol{w}_i = \sum_{i=1}^{m} \boldsymbol{W}_i\boldsymbol{\lambda}_i = \boldsymbol{W}\boldsymbol{\lambda} = -\boldsymbol{w}_{\mathrm{ext}}, \tag{11}$$

where $\boldsymbol{W} = [\boldsymbol{W}_1 \quad \boldsymbol{W}_2 \quad \cdots \quad \boldsymbol{W}_m] \in \mathbb{R}^{d\times ml}$ and $\boldsymbol{\lambda} = \left[\boldsymbol{\lambda}_1^{\mathrm{T}} \quad \boldsymbol{\lambda}_2^{\mathrm{T}} \quad \cdots \quad \boldsymbol{\lambda}_m^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbb{R}^{ml}$.

3. Optimal grasp planning

3.1. Two grasp quality criteria

A grasp is said to be force-closure if any $\boldsymbol{w}_{\mathrm{ext}} \in \mathbb{R}^d$ can be equilibrated [2–25,31]. Without other conditions, certain grasps have to exert the contact forces exceeding their upper bounds. This implies that only the force-closure requirement is insufficient for OGP, and the grasp quality criteria regarding the force magnitude must be considered. For this the largest resultant wrench in the worst direction generated by the feasible contact forces with limited magnitude is suggested [22,23]. Since the contact force magnitude can be measured by σ^u or σ^M, limiting either of them, we obtain a criterion. First consider the former.

Let σ^u be within unity. From (6) and (11), the resultant wrenches that can be generated by the grasp lie in the convex hull $\mathscr{W}_c^u$ of the *union* of $\mathscr{W}_{ic}$, $i = 1, 2, \ldots, m$:

$$\mathscr{W}_c^u = \mathrm{conv}\left(\bigcup_{i=1}^m \mathscr{W}_{ic}\right) = \mathrm{conv}\left(\bigcup_{i=1}^m \mathrm{conv}\mathscr{W}_i\right) = \mathrm{conv}\left(\bigcup_{i=1}^m \mathscr{W}_i\right) = \mathrm{conv}\mathscr{W}^u, \tag{12}$$

where

$$\mathscr{W}^u = \bigcup_{i=1}^m \mathscr{W}_i. \tag{13}$$

Let $\mathscr{W}_-^u$ be the subset of $\mathscr{W}^u$ except $\mathbf{0}$ and $\mathscr{W}_{c-}^u$ the convex hull of $\mathscr{W}_-^u$:

$$\mathscr{W}_-^u = \mathscr{W}^u \setminus \mathbf{0} = \left(\bigcup_{i=1}^m \mathscr{W}_i\right) \setminus \mathbf{0}, \tag{14}$$

$$\mathscr{W}_{c-}^u = \mathrm{conv}\mathscr{W}_-^u = \mathrm{conv}\left(\left(\bigcup_{i=1}^m \mathscr{W}_i\right) \setminus \mathbf{0}\right). \tag{15}$$

The L_2 distance $\rho(\mathbf{0}, \mathscr{W}_{c-}^u)$ between $\mathbf{0}$ and $\mathscr{W}_{c-}^u$ can be taken as a grasp quality criterion.

Proposition 1. *The following statements are true*:

1. *If* $\rho(\mathbf{0}, \mathscr{W}_{c-}^u) > 0$, *then* $\mathbf{0} \notin \mathscr{W}_{c-}^u$. *The grasp cannot apply resultant wrenches in any two opposite directions* (*neither force-closure nor partial force-closure*).
2. *If* $\rho(\mathbf{0}, \mathscr{W}_{c-}^u) = 0$, *then* $\mathbf{0} \in \mathscr{W}_{c-}^u$ *with* $\mathbf{0} \notin \mathrm{int}\,\mathscr{W}_{c-}^u$. *The grasp may apply resultant wrenches in certain opposite directions but not in all directions* (*partial force-closure*).
3. *If* $\rho(\mathbf{0}, \mathscr{W}_{c-}^u) < 0$, *then* $\mathbf{0} \in \mathrm{int}\,\mathscr{W}_{c-}^u$. *The grasp can apply resultant wrenches in all directions* (*force-closure*). *Moreover*, $-\rho(\mathbf{0}, \mathscr{W}_{c-}^u)$ *indicates the magnitude of the largest resultant wrench in the worst direction generated by feasible contact forces with* σ^u *equal to unity.*

On the other hand, the last two paragraphs hold true if the word *union* is replaced by *Minkowski sum* and all the superscripts u by M. After the substitution, *Proposition 1* is renumbered as *Proposition 2*. Both criteria $\rho(\mathbf{0}, \mathscr{W}_{c-}^u)$ and $\rho(\mathbf{0}, \mathscr{W}_{c-}^M)$ are the-less-the-better. Hereinafter they are shortened as ρ^u and ρ^M, respectively.

3.2. Precise computation of the criteria

Although the above two criteria were advanced long ago [23], no precise computational method was available until now. From Theorem 1 in Appendix A it follows that

$$\rho^u = -\min_{\|z\|=1} p_{\mathscr{W}_{c-}^u}(z), \tag{16}$$

$$\rho^M = -\min_{\|z\|=1} p_{\mathscr{W}_{c-}^M}(z), \tag{17}$$

where $z \in \mathbb{R}^d$ is a variable; $p_{\mathscr{W}_{c-}^u}(z)$ and $p_{\mathscr{W}_{c-}^M}(z)$ are the support functions of $\mathscr{W}_{c-}^u$ and $\mathscr{W}_{c-}^M$, respectively. From $\mathscr{W}_{c-}^u = \mathrm{conv}\mathscr{W}_-^u$ it follows that $p_{\mathscr{W}_{c-}^u}(z) = p_{\mathscr{W}_-^u}(z)$, where $p_{\mathscr{W}_-^u}(z)$ is the support function of $\mathscr{W}_-^u$. By the same reason, we obtain $p_{\mathscr{W}_{c-}^M}(z) = p_{\mathscr{W}_-^M}(z)$, where $p_{\mathscr{W}_-^M}(z)$ is the support function of $\mathscr{W}_-^M$. Thus

$$\rho^u = -\min_{\|z\|=1} p_{\mathscr{W}_-^u}(z), \tag{18}$$

$$\rho^M = -\min_{\|z\|=1} p_{\mathscr{W}_-^M}(z). \tag{19}$$

From (9) and (14), $p_{\mathscr{W}^u_-}(z) = \max(w^T_{1,1}z, w^T_{1,2}z, \ldots, w^T_{m,l}z)$ and ρ^u can be found by

$$\begin{cases} \text{Maximize} & -\max(w^T_{1,1}z, w^T_{1,2}z, \ldots, w^T_{m,l}z) \\ \text{subject to} & \|z\| = 1. \end{cases} \tag{20}$$

Similarly, from Algorithm 1 (see below), $p_{\mathscr{W}^M_-}(z) = \max(w^T_1 z, w^T_2 z, \ldots, w^T_n z)$ and ρ^M can be obtained by

$$\begin{cases} \text{Maximize} & -\max(w^T_1 z, w^T_2 z, \ldots, w^T_n z) \\ \text{subject to} & \|z\| = 1. \end{cases} \tag{21}$$

Algorithm 1. This algorithm computes the elements of $\mathscr{W}^M_-$, denoted by w_j. A vector $q_j \in \mathbb{R}^{ml}$ is employed to note down $w_{i,h}$, $h = 1, 2, \ldots, l$ and $i = 1, 2, \ldots, m$ yielding w_j, i.e., $w_j = \sum_{i=1}^{m}\sum_{h=1}^{l} q_{l(i-1)+h,j} w_{l(i-1)+h} = Wq_j$, where $q_{l(i-1)+h,j}$ is the component of q_j.

Let $s_{l(i-1)+h} = s_{i,h}$ and $w_{l\,(i-1)+h} = w_{i,h}$ for $h = 1, 2, \ldots, l$ and $i = 1, 2, \ldots, m$.
$n = 0$;
For $i = 1, 2, \ldots, m$
  For $h = 1, 2, \ldots, l$
    For $j = 1, 2, \ldots, n$

$$w_{nh+j} = w_j + w_{l(i-1)+h};\ \ q_{nh+j} = q_j;\ \ q_{l(i-1)+h,nh+j} = 1;$$

    End
  End
  For $h = 1, 2, \ldots, l$

$$w_{n(l+1)+h} = w_{l(i-1)+h};\ \ q_{n(l+1)+h} = 0;\ \ q_{l(i-1)+h,n(l+1)+h} = 1;$$

  End
  $n = n(l + 1) + l$;
End

Let us make the following remarks:

1. $\mathscr{W}^M_- = \{w_1, w_2, \ldots, w_n\}$, where $n = (l + 1)^m - 1$.
2. Let $P = [\, w_1 \quad w_2 \quad \cdots \quad w_n \,] \in \mathbb{R}^{d\times n}$ and $Q = [\, q_1 \quad q_2 \quad \cdots \quad q_n \,] \in \mathbb{R}^{ml\times n}$. Then $P = WQ$.
3. q_j consists of 0 and 1. At most one of $q_{l(i-1)+h,j}$, $h = 1, 2, \ldots, l$ is 1, which implies that $w_{i,h}$, $h = 1, 2, \ldots, l$ from the same contact i cannot be simultaneously used to yield w_j. Moreover, at least one of $q_{l(i-1)+h,j}$, $h = 1, 2, \ldots, l$ and $j = 1, 2, \ldots, n$ is 1. Totally there are $ml(l + 1)^{m-1}$ entries of Q are 1's.
4. Let $\lambda = Q\alpha$, where each component α_j of $\alpha \in \mathbb{R}^n$ is nonnegative and $\lambda_{i,h}$ denotes a component of $\lambda \in \mathbb{R}^{ml}$. From Remark 3 it follows that $\sum_{h=1}^{l}\lambda_{i,h} \leqslant \sum_{j=1}^{n}\alpha_j$ and $\sum_{i=1}^{m}\sum_{h=1}^{l}\lambda_{i,h} \geqslant \sum_{j=1}^{n}\alpha_j$. This property together with Remark 2 plays an important role in deriving the DFD algorithm in Section 4.

3.3. Comparison of their physical meanings

Both criteria search for the largest resultant wrench in the worst direction. The only difference is that ρ^u comes from σ^u, while ρ^M comes from σ^M.

From (4) and (11), the contact forces generating $\mathscr{W}^u_{c-}$ can be expressed by

$$f_i = \sum_{h=1}^{l} \lambda_{i,h} s_{i,h}, i = 1, 2, \ldots, m \text{ satisfying} \sum_{i=1}^{m}\sum_{h=1}^{l} \lambda_{i,h} = 1 \text{ with all } \lambda_{i,h} \geqslant 0$$

Thus $\mathscr{W}^u_{c-}$ can be rewritten as

$$\mathscr{W}^u_{c-} = \left\{ \sum_{i=1}^{m} \boldsymbol{G}_i \boldsymbol{f}_i | \boldsymbol{f}_i \text{ satisfies (4) for } i = 1, 2, \ldots, m \text{ with } \sigma^u = 1 \right\}. \tag{22}$$

From (4) and Remark 2 on Algorithm 1, the contact forces generating $\mathscr{W}^M_{c-}$ can be expressed by

$$\boldsymbol{f}_i = \sum_{h=1}^{l} \lambda_{i,h} \boldsymbol{s}_{i,h}, i = 1, 2, \ldots, m \text{ satisfying } \lambda = \boldsymbol{Q\alpha} \text{ and } \sum_{j=1}^{n} \alpha_j = 1 \text{ with all } \alpha_j \geqslant 0$$

where $\alpha = [\alpha_1\ \alpha_2\ \cdots\ \alpha_n]^{\mathrm{T}} \in \mathbb{R}^n$. From Remark 4, we further obtain $\sum_{h=1}^{l} \lambda_{i,h} \leqslant \sum_{j=1}^{n} \alpha_j = 1$ and $\sum_{i=1}^{m} \sum_{h=1}^{l} \lambda_{i,h} \geqslant \sum_{j=1}^{n} \alpha_j = 1$. Thus $\mathscr{W}^M_{c-}$ can be rewritten as

$$\mathscr{W}^M_{c-} = \left\{ \sum_{i=1}^{m} \boldsymbol{G}_i \boldsymbol{f}_i | \boldsymbol{f}_i \text{ satisfies (4) for } i = 1, 2, \ldots, m \text{ with } \sigma^u \geqslant 1 \text{ and } \sigma^M \leqslant 1 \right\}. \tag{23}$$

Comparing (22) with (23), we see that the constraint on the contact forces in $\mathscr{W}^u_{c-}$ is excessive and much stronger than that in $\mathscr{W}^M_{c-}$; thus $\mathscr{W}^u_{c-}$ is no more than a subset of $\mathscr{W}^M_{c-}$. Fig. 1(a) gives a planar example, where the friction coefficients are 0.2 and the force upper bounds are 1 at each contact. Fig. 1(b) illustrates that the corresponding $\mathscr{W}^u_{c-}$ is contained in and much smaller than $\mathscr{W}^M_{c-}$. Consequently, for a force-closure grasp, ρ^u is usually greater than ρ^M (note that both are negative and their absolute values indicate the resultant wrenches). Fig. 1(c) together with (a) shows that ρ^u does not change when the contacts increase. Actually, since the resultant wrenches in $\mathscr{W}^u_{c-}$ are obtained by the convex combination (a weighted average) of $\mathscr{W}_{ic}$, $i = 1, 2, \ldots, m$, ρ^u computes the average function of the contacts. As a result, ρ^u is insensitive to the contact number. On the contrary, $\mathscr{W}^M_{c-}$ is obtained by the Minkowski sum of $\mathscr{W}_{ic}$, $i = 1, 2, \ldots, m$, so that the functions of all the contacts are added up. We would say, ρ^M reflects the load capacity of a grasp. Therefore the two criteria are not equivalent. For evaluating the goodness of various grasps, of course the latter is better.

3.4. Description of the OGP algorithm

OGP problem: Given an object, determine the contact positions of a grasp such that a quality criterion attains a minimum.

OGP algorithm (*Refer to Fig. 2*). For easy understanding, we start with the 2D case. The contact position $\boldsymbol{r}_i$ can be expressed by a function of one variable ϕ_i. ρ^M is taken as the optimization criterion, which changes with ϕ_i, $i = 1, 2, \ldots, m$. Let ϕ_i^{L} and ϕ_i^{U} denote the lower and upper bounds on ϕ_i, i.e., $\phi_i^{\mathrm{L}} \leqslant \phi_i \leqslant \phi_i^{\mathrm{U}}$.

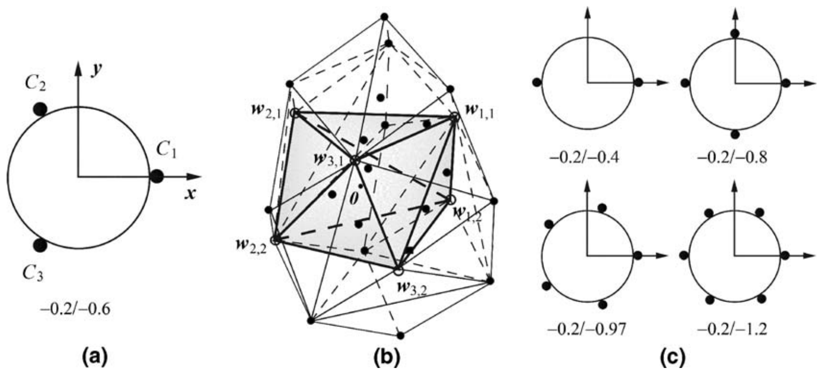

Fig. 1. $\mathscr{W}^u_{c-}$ adopted in previous literature compared with $\mathscr{W}^M_{c-}$ recommended by this paper. The data at the bottom of each grasp are ρ^u/ρ^M, which are the distances from $\boldsymbol{0}$ to $\mathscr{W}^u_{c-}$ and $\mathscr{W}^M_{c-}$, respectively. ρ^u (resp. ρ^M) is negative if and only if $\boldsymbol{0}$ lies in the interior of $\mathscr{W}^u_{c-}$ (resp. $\mathscr{W}^M_{c-}$), and is the-less-the-better. (a) A three-finger grasp holding a unit circle. (b) $\mathscr{W}^u_{c-}$ in the thick lines and $\mathscr{W}^M_{c-}$ in the thin lines of (a). (c) A variety of other grasps. Two phenomena are worthy of note: (i) $\mathscr{W}^u_{c-}$ is rather smaller than $\mathscr{W}^M_{c-}$. (ii) With more contacts, the grasp is certainly better, but ρ^u keeps the same.

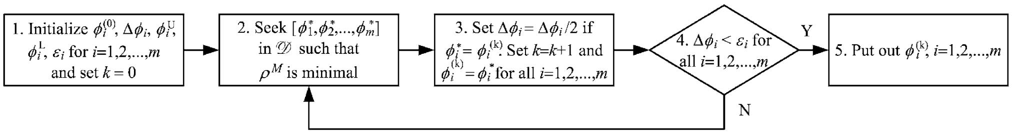

Fig. 2. Flowchart of the optimal grasp planning (OGP) algorithm. $\mathscr{W}^{u}_{c-}$ can be used instead of $\mathscr{W}^{M}_{c-}$, but the results might be different.

Step 1. Choose the initial values $\phi_i^{(0)}$, the initial step $\Delta\phi_i$ ($\Delta\phi_i > 0$), the lower bounds ϕ_i^{L} and the upper bounds ϕ_i^{U} of ϕ_i, and the termination tolerances ε_i ($\varepsilon_i > 0$) of $\Delta\phi_i$, $i = 1, 2, \ldots, m$. Set $k = 0$.
Step 2. Search the domain $\mathscr{D} = \prod_{i=1}^{m}\{\max(\phi_i^{\mathrm{L}}, \phi_i^{(k)} - \Delta\phi_i), \phi_i^{(k)}, \min(\phi_i^{(k)} + \Delta\phi_i, \phi_i^{\mathrm{U}})\}$ for $[\phi_1^*\phi_2^*\cdots\phi_m^*]$, where the value of ρ^M is minimal.
Step 3. Check whether $\phi_i^* = \phi_i^{(k)}$ for $i = 1, 2, \ldots, m$. If $\phi_i^* = \phi_i^{(k)}$, set $\Delta\phi_i = \Delta\phi_i/2$. Let $\phi_i^{(k+1)} = \phi_i^*$ and $k = k + 1$.
Step 4. If $\Delta\phi_i \leqslant \varepsilon_i$ for all $i = 1, 2, \ldots, m$, then go to Step 5; otherwise, go to Step 2.
Step 5. $\phi_i^{(k)}$, $i = 1, 2, \ldots, m$ give the optimal grasp configuration. The algorithm terminates.

This algorithm searches the gradient descent directions so as to decrease ρ^M most rapidly. The quality of the computed grasp depends on the initial values and the initial steps. Instead of the globally optimal solution, often a local minimum is attained, like [24]. Striving for the global minimum, one may try several initial values and steps. The running time is determined by the initial steps and the termination tolerances. For 3D grasps, it needs only to add a parameter ψ_i to specify the contact position $\boldsymbol{r}_i$ together with ϕ_i and treat it like ϕ_i. If ρ^M is replaced by ρ^u, the results may differ because their descent directions are different.

4. Dynamic force distribution

4.1. Principle of the DFD algorithm

DFD problem: Given a force-closure grasp and a dynamic external wrench $\boldsymbol{w}_{\mathrm{ext}}$, find the contact forces $\boldsymbol{f}_i$, $i = 1, 2, \ldots, m$ satisfying (4) and (11) such that the contact force magnitude becomes minimum.

To minimize the contact forces, the objective function can be taken as $\sigma^u = \sum_{i=1}^{m}(f_{in}/f_i^{\mathrm{U}})$ for fast computation or $\sigma^M = \max_{1\leqslant i\leqslant m}(f_{in}/f_i^{\mathrm{U}})$ for better solution. Now we transform the DFD problems with the two objectives into the same convex analysis problem, which has been solved in Appendix B.

From (4), (6), and (11), DFD with σ^u minimized can be transformed into computing $\lambda_{i,h}$, $i = 1, 2, \ldots, m$ and $h = 1, 2, \ldots, l$ satisfying (11) with minimum $\sum_{i=1}^{m}\sum_{h=1}^{l}\lambda_{i,h}$. On the other hand, $-\boldsymbol{w}_{\mathrm{ext}}$ can be expressed by

$$-\boldsymbol{w}_{\mathrm{ext}} = \sum_{j=1}^{n}\alpha_j\boldsymbol{w}_j = \boldsymbol{P\alpha}, \quad \alpha_j \geqslant 0 \text{ for } j = 1, 2, \ldots, n. \tag{24}$$

From Remark 2, we obtain

$$-\boldsymbol{w}_{\mathrm{ext}} = \boldsymbol{W\lambda} \text{ with } \boldsymbol{\lambda} = \boldsymbol{Q\alpha}. \tag{25}$$

From (5) and Remark 4, $f_{in} \leqslant f_i^{\mathrm{U}}\sum_{j=1}^{n}\alpha_j$ for $i = 1, 2, \ldots, m$, which implies that $\max_{1\leqslant i\leqslant m}(f_{in}/f_i^{\mathrm{U}}) \leqslant \sum_{j=1}^{n}\alpha_j$. Thus DFD aiming to minimize σ^M can be transformed into computing α_j, $j = 1, 2, \ldots, n$ satisfying (24) with minimum $\sum_{j=1}^{n}\alpha_j$.

Referring to Appendix B, to compute $\lambda_{i,h}$, $i = 1, 2, \ldots, m$ and $h = 1, 2, \ldots, l$ such that σ^u is minimum, we first determine $\boldsymbol{z}_b$ by

$$\begin{cases} \text{Maximize } & -\boldsymbol{w}_{\mathrm{ext}}^{\mathrm{T}}\boldsymbol{z} \\ \text{subject to } & \boldsymbol{w}^{\mathrm{T}}\boldsymbol{z} \leqslant 1, \boldsymbol{w} \in \mathscr{W}^{u}_{-}. \end{cases} \tag{26}$$

Let $\boldsymbol{W}$ be partitioned into $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$, where $\boldsymbol{W}_1$ consists of $\boldsymbol{w} \in \mathscr{W}^{u}_{-}$ satisfying $\boldsymbol{z}_b^{\mathrm{T}}\boldsymbol{w} = 1$ and $\boldsymbol{W}_2$ consists of the others. Correspondingly, $\boldsymbol{\lambda}$ is be partitioned into $\boldsymbol{\lambda}_1$ and $\boldsymbol{\lambda}_2$. Then

$$\boldsymbol{\lambda}_1 = -\boldsymbol{W}_1^{+}\boldsymbol{w}_{\mathrm{ext}} \quad \text{and} \quad \boldsymbol{\lambda}_2 = \boldsymbol{0}. \tag{27}$$

To compute $\lambda_{i,h}$, $i = 1, 2, \ldots, m$ and $h = 1, 2, \ldots, l$ such that σ^M is minimum, just substituting $\mathscr{W}_-^M$ for $\mathscr{W}_-^u$ and $\boldsymbol{P}$ for $\boldsymbol{W}$ in the above derivation, we obtain

$$\boldsymbol{\alpha}_1 = -\boldsymbol{P}_1^+ \boldsymbol{w}_{\text{ext}} \quad \text{and} \quad \boldsymbol{\alpha}_2 = \boldsymbol{0}. \tag{28}$$

Partition $\boldsymbol{Q}$ into $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_2$ according to $\boldsymbol{\alpha}_1$ and $\boldsymbol{\alpha}_2$. Combining (25) and (28) leads to

$$\boldsymbol{\lambda} = \boldsymbol{Q}_1\boldsymbol{\alpha}_1 + \boldsymbol{Q}_2\boldsymbol{\alpha}_2 = -\boldsymbol{Q}_1\boldsymbol{P}_1^+ \boldsymbol{w}_{\text{ext}}. \tag{29}$$

Furthermore, $\sigma^u_{\min}$ and $\sigma^M_{\min}$ are the optimal objective values of (26) w.r.t. $\mathscr{W}_-^u$ and $\mathscr{W}_-^M$, respectively.

4.2. Enhancements of DFD based on $\mathscr{W}_{c-}^M$

Solving problem (26) w.r.t. $\mathscr{W}_-^M$ needs O(n) time, where $n = (l+1)^m - 1$ is large for 3D grasps. This encumbers the real-time application of the DFD algorithm. Section 4.1 implies that only a few elements of $\mathscr{W}_-^M$, namely those being the columns of $\boldsymbol{P}_1$, are required for $\boldsymbol{w}_{\text{ext}}$ at a time point. Let $\mathscr{W}_t^M$ be a set of them. Then $\bigcup_t \mathscr{W}_t^M$ is sufficient for the whole $\boldsymbol{w}_{\text{ext}}$. Notice that $\mathscr{W}_t^M$ and $\mathscr{W}_{t+\Delta t}^M$ are identical when Δt is small. Thus we can pick out $\bigcup_t \mathscr{W}_t^M$ at some discrete time points. Moreover, as the assumption in Appendix B, it should be satisfied that $\boldsymbol{0} \in \text{int}(\text{conv}(\bigcup_t \mathscr{W}_t^M))$; otherwise randomly add more elements, since they will not participate in computing the contact forces. This process can be automatically executed by computers in advance.

In addition, the matrix $\boldsymbol{Q} \in \mathbb{R}^{ml\times n}$ is a large-scale sparse matrix, which can be substituted by two matrices $\boldsymbol{A} \in \mathbb{R}^{2\times n}$ and $\boldsymbol{B} \in \mathbb{R}^{2\times ml(l+1)^{m-1}}$. The entries $b_{1,k}$ and $b_{2,k}$ of $\boldsymbol{B}$ keep the values of i and h, while the entries $a_{1,j}$ and $a_{2,j}$ of $\boldsymbol{A}$ note down the starting and end addresses of $\boldsymbol{b}_k$ corresponding to $\boldsymbol{w}_j$:

$$\boldsymbol{w}_j = \sum_{k=a_{1,j}}^{a_{2,j}} \boldsymbol{w}_{b_{1,k},b_{2,k}} = \sum_{k=a_{1,j}}^{a_{2,j}} \boldsymbol{G}_{b_{1,k}} \boldsymbol{s}_{b_{1,k},b_{2,k}}. \tag{30}$$

This process can be easily integrated into Algorithm 1. Then $\boldsymbol{f}_i = \boldsymbol{f}_{b_{1,k}}$ can be computed by the following loop

$$\boldsymbol{f}_{b_{1,k}} = \boldsymbol{f}_{b_{1,k}} + \alpha_j \boldsymbol{s}_{b_{1,k},b_{2,k}} \text{ for } k = a_{1,j}, a_{1,j}+1, \ldots, a_{2,j} \text{ and } j = 1, 2, \ldots, n. \tag{31}$$

Moreover, by this means, computation of $\boldsymbol{\lambda}$ can be skipped.

4.3. Description of the DFD algorithm

DFD algorithm (*Refer to Fig. 3*). The algorithm is implemented in two phases.

Offline phase.

Step 1. Calculate $\boldsymbol{w}_{i,h}$, $h = 1, 2, \ldots, l$ and $i = 1, 2, \ldots, m$.
Step 2. To minimize $\sigma^M = \max_{1\leqslant i\leqslant m}(f_{in}/f_i^{\text{U}})$, compute $\boldsymbol{w}_j$, $j = 1, 2, \ldots, n$ by Algorithm 1, reserve the required $\boldsymbol{w}_j$ by the method in Section 4.2, and go to Step 7. To minimize $\sigma^u = \sum_{i=1}^m (f_{in}/f_i^{\text{U}})$, go to Step 3.

Online phase.

Step 3. Determine $\boldsymbol{z}_b$ by (26) w.r.t. $\mathscr{W}_-^u$.
Step 4. Construct $\boldsymbol{W}_1$ by the elements $\boldsymbol{w}_{i,h}$ of $\mathscr{W}_-^u$ satisfying $\boldsymbol{z}_b^{\text{T}}\boldsymbol{w}_{i,h} = 1$.

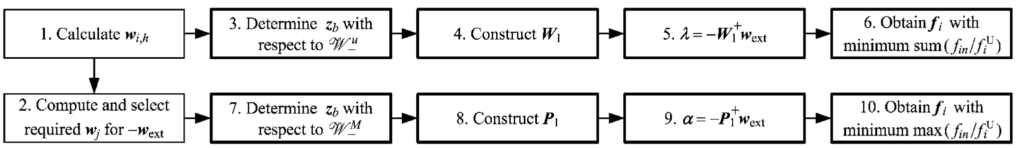

Fig. 3. Flowchart of the dynamic force distribution (DFD) algorithm. Steps 1 and 2 are performed offline. The online phase starts from Step 3 or 7 for minimizing $\sum_{i=1}^m (f_{in}/f_i^{\text{U}})$ or $\max_{1\leqslant i\leqslant m}(f_{in}/f_i^{\text{U}})$.

Step 5. Compute $\boldsymbol{\lambda}$ by (27).
Step 6. The contact forces $\boldsymbol{f}_i$, $i = 1, 2, \ldots, m$ with minimum σ^u is computed by (4). The algorithm ends.
Step 7. Determine $\boldsymbol{z}_b$ by (26) w.r.t. $\mathscr{W}_-^M$.
Step 8. Construct $\boldsymbol{P}_1$ by the elements $\boldsymbol{w}_j$ of $\mathscr{W}_-^M$ satisfying $\boldsymbol{z}_b^{\mathrm{T}} \boldsymbol{w}_j = 1$.
Step 9. Compute $\boldsymbol{\alpha}$ by (28).
Step 10. The contact forces $\boldsymbol{f}_i$, $i = 1, 2, \ldots, m$ with minimum σ^M is computed by (31). The algorithm ends.

In the online phase, Step 3 can be solved in O(ml) time. Step 4 takes O(ml) time to find $\boldsymbol{w}_j$, which are commonly much less than ml. In Step 5, $\boldsymbol{\lambda}$ is obtained in O(1) time. In Step 6, $\boldsymbol{f}_i$, $i = 1, 2, \ldots, m$ are computed in O(ml) time. Totally, the time complexity of the online phase for minimizing σ^u is O(ml) and *linear* with the number of the elements of $\mathscr{W}_-^u$, essentially faster than the *polynomial* time complexity of Han et al. algorithm [31] and the *quadratic* time complexity of Helmke et al. algorithm [32]. Similarly, its time complexity for minimizing σ^M is O(n_1), where n_1 is the number of the elements reserved in $\mathscr{W}_-^M$ after Step 2.

5. Numerical examples

We implement the algorithms using the optimization toolbox of MATLAB on a Pentium-M notebook.

Example 1. It is required to grip a racket, as depicted in Fig. 4. First determine a grasp with three contacts (see Fig. 5(a)–(e)). The contact positions and the physical conditions are as follows:

$$\boldsymbol{r}_1 = [\,0.13\cos\phi_1 \quad 0.16\sin\phi_1\,]^{\mathrm{T}}\ \mathrm{m}, \quad 0 \leqslant \phi_1 \leqslant \pi,\ \mu_1 = 0.1,\ f_1^{\mathrm{U}} = 30\ \mathrm{N},$$

$$\boldsymbol{r}_2 = [\,-0.68\phi_2^3 - 0.015 \quad 0.8\phi_2^2 - 0.315\,]^{\mathrm{T}}\ \mathrm{m}, \quad 0 \leqslant \phi_2 \leqslant 0.5,\ \mu_2 = 0.1,\ f_2^{\mathrm{U}} = 60\ \mathrm{N},$$

$$\boldsymbol{r}_3 = [\,0.68\phi_3^3 + 0.015 \quad 0.8\phi_3^2 - 0.315\,]^{\mathrm{T}}\ \mathrm{m}, \quad 0 \leqslant \phi_3 \leqslant 0.5,\ \mu_3 = 0.1,\ f_3^{\mathrm{U}} = 60\ \mathrm{N}.$$

Let $\phi = [\phi_1, \phi_2, \phi_3]$ denote the grasp configuration. Taking the initial configuration $\phi^{(0)} = [0, 0.5, 0.5]$, the initial step $\Delta\phi = [\pi/4, 0.2, 0.2]$, and the termination tolerance $\varepsilon = [\pi/8, 0.1, 0.1]$, the proposed OGP algorithm based on $\mathscr{W}_{c-}^M$ turns out the optimal grasp $\phi^{(4)} = [\pi/4, 0.45, 0]$ in four iterations with the CPU time of

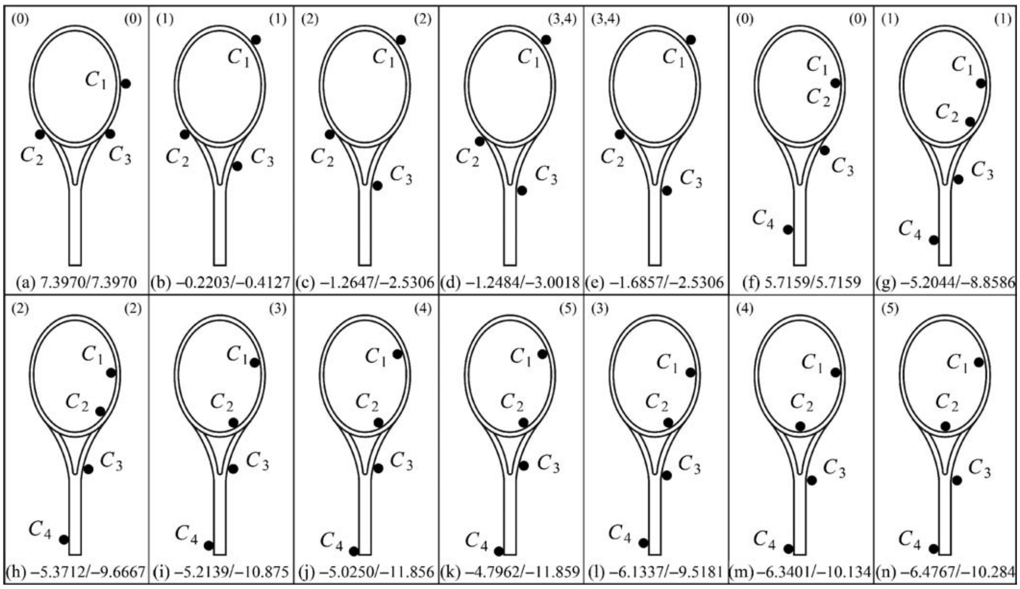

Fig. 4. Optimal grasp planning for a racket with (a)–(e) three contacts or (f)–(l) four contacts. The data at the bottom of each grasp are ρ^u/ρ^M. The numbers in the top left and top right corners are the iteration numbers of the OGP algorithm based on ρ^u and ρ^M, respectively.

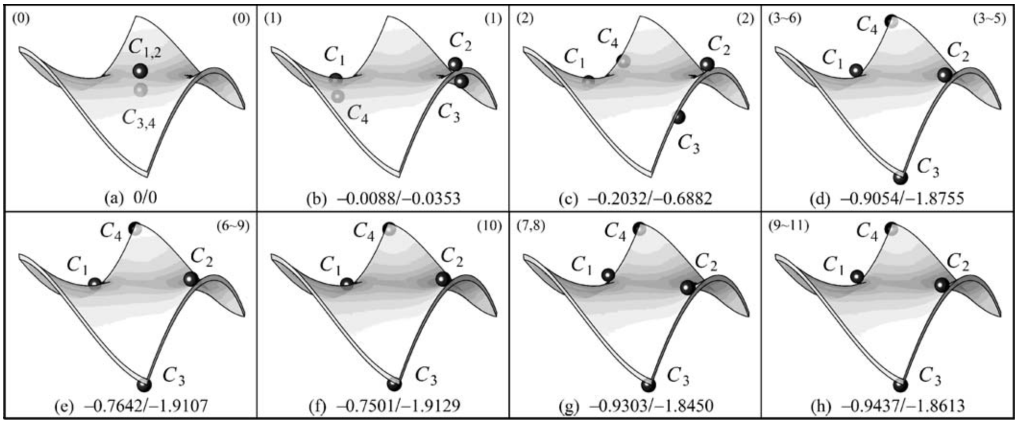

Fig. 5. Various grasps of a tile. The values of ρ^u/ρ^M are shown at the bottom. The iteration numbers of the OGP algorithm based on ρ^u and ρ^M are given in the top left and top right corners, respectively.

129.15 s. Fig. 4(a)–(d) describe the grasp configurations in process of the optimization. During the last two iterations, the grasp configuration keeps unchanged. Only the steps get reduced. Using $\mathscr{W}^u_{c-}$ instead of $\mathscr{W}^M_{c-}$, the OGP algorithm terminates at $\phi^{(4)} = [\pi/4, 0.5, 0]$ with the CPU time of 165.97 s (see Fig. 4(a)–(c) and (e)).

Next we construct a four-finger grasp, as described in Fig. 4(f)–(l), with contacts at

$$\begin{aligned} \boldsymbol{r}_1 &= [0.12\cos\phi_1 \quad 0.15\sin\phi_1]^{\mathrm{T}}\ \mathrm{m}, \quad 0 \leqslant \phi_1 \leqslant \pi,\ \mu_1 = 0.1,\ f_1^{\mathrm{U}} = 30\ \mathrm{N}, \\ \boldsymbol{r}_2 &= [0.12\cos\phi_2 \quad 0.17\sin\phi_2]^{\mathrm{T}}\ \mathrm{m}, \quad -\pi \leqslant \phi_2 \leqslant 0,\ \mu_2 = 0.1,\ f_2^{\mathrm{U}} = 30\ \mathrm{N}, \\ \boldsymbol{r}_3 &= [0.68\phi_3^3 + 0.015 \quad 0.8\phi_3^2 - 0.315]^{\mathrm{T}}\ \mathrm{m}, \quad 0 \leqslant \phi_3 \leqslant 0.5,\ \mu_3 = 0.1,\ f_3^{\mathrm{U}} = 60\ \mathrm{N}, \\ \boldsymbol{r}_4 &= [-0.015 \quad \phi_4]^{\mathrm{T}}\ \mathrm{m}, \quad -0.5 \leqslant \phi_4 \leqslant -0.32,\ \mu_4 = 0.2,\ f_4^{\mathrm{U}} = 100\ \mathrm{N}. \end{aligned}$$

The grasp configuration is given by $\phi = [\phi_1, \phi_2, \phi_3, \phi_4]$. The initial step and the termination tolerance are set to $\Delta\phi = [\pi/4, \pi/4, 0.2, 0.02]$ and $\varepsilon = [\pi/8, \pi/8, 0.1, 0.01]$. With $\phi^{(0)} = [0, 0, 0.4, -0.42]$, the OGP algorithm puts out $\phi^{(5)} = [\pi/8, -3\pi/8, 0.225, -0.5]$ based on $\mathscr{W}^M_{c-}$ (Fig. 4(f)–(k)) and $\phi^{(5)} = [\pi/16, -\pi/2, 0, -0.5]$ based on $\mathscr{W}^u_{c-}$ (Fig. 4(f)–(h) and (l)–(n)). The CPU times are 434.58 s and 479.67 s, respectively.

Fig. 4 also displays the values of ρ^u and ρ^M. For force-closure grasps, the former are much greater than the latter, such as (d) and (k). More valuably, while ρ^M decreases in iteration, ρ^u sometimes keeps unchanged or even increases, such as (h)–(k). This implies that the descent directions of ρ^u and ρ^M are distinct; thus OGP using the two criteria could result in different grasps, such as (d), (e) and (k), (n).

To test the DFD algorithm, assume that an external wrench $\boldsymbol{w}_{\mathrm{ext}} = [-8\ -10\ 6]^{\mathrm{T}}$ (N or N · m) exerts on the above grasps (Fig. 4(d), (e), (k) and (n)). Table 1 lists the results of the DFD algorithm based on $\mathscr{W}^u_{c-}$ and $\mathscr{W}^M_{c-}$. For the three-finger grasps (d) and (e), the two kinds of optimal contact forces are identical, while for the four-finger grasps (k) and (n), they are not. In particular, for grasp (n), the contact forces with minimum σ^u are over f_i^{U}, whereas those with minimum σ^M are all below f_i^{U}. (d) satisfies the traditional condition of force-closure, but actually does not work because $\sigma^M_{\min} > 1$.

Table 1
The results of the DFD algorithm based on $\mathscr{W}^u_{c-}$ and $\mathscr{W}^M_{c-}$ in Example 1

Grasp	$\mathscr{W}^u_{c-}$			$\mathscr{W}^M_{c-}$		
	$\mathrm{Max}(f_{in}/f_i^u)$	$\sum(f_{in}/f_i^{\mathrm{U}})$	CPU time (ms)	$\mathrm{Max}(f_{in}/f_i^u)$	$\sum(f_{in}/f_i^{\mathrm{U}})$	CPU time (ms)
(d)	1.3868	3.1099	26.58	1.3868	3.1099	29.34
(e)	0.9521	2.0623	25.15	0.9521	2.0623	28.76
(k)	0.8658	1.2280	24.23	0.8119	1.4159	28.04
(n)	1.1336	1.6728	27.54	0.9904	1.6945	28.94

Example 2. The object to be grasped is a glazed tile for modern roofs and mosaics. Its upper and lower surfaces are two parallel "monkey saddles", given by

$$\boldsymbol{r}_u = [\,\phi \quad \psi \quad 10(\phi^3 - 3\psi^2\phi) + 0.005\,]^{\mathrm{T}}\ \mathrm{m}, \qquad \boldsymbol{r}_l = [\,\phi \quad \psi \quad 10(\phi^3 - 3\psi^2\phi) - 0.005\,]^{\mathrm{T}}\ \mathrm{m}$$

where $-0.2 \leqslant \phi \leqslant 0.2$ and $-0.2 \leqslant \psi \leqslant 0.2$. Note that it is square and symmetric w.r.t. the x-axis, because the third entries of the above vectors remain unchanged if ψ is substituted by $-\psi$. Therefore each side of the tile can be connected smoothly with the proper side of another tile. In this way, the area covered seamlessly by such tiles can be extended in the x and y directions to an arbitrary rectangular polygon whatever we like. We put two contacts (C_1 and C_2) on the tile and two (C_3 and C_4) beneath. Their positions are specified by $[\phi_i, \psi_i]$, $i = 1, 2, 3, 4$ with constraints $\phi_2 = -\phi_1$, $\psi_2 = -\psi_1$, $\phi_4 = -\phi_3$, and $\psi_4 = -\psi_3$. Thus the grasp configuration is expressed by $c = [\phi_1, \psi_1, \phi_3, \psi_3]$. Assume that $\mu = 0.2$ and $f^{\mathrm{U}} = 10$ N at each contact. Take the initial steps to be 0.2 m and the termination tolerances to be 0.001 m. Each friction cone is linearized into a 10-side polyhedral cone, i.e., $l = 10$ in (3). Using $c^{(0)} = [0,0,0,0]$, the OGP algorithm returns $c^{(10)} = [-0.2, 0.0375, 0.1992, -0.2]$ based on $\mathscr{W}^M_{c-}$ (Fig. 5(a)–(f)) and $c^{(11)} = [-0.2, 0.0531, -0.2, 0.2]$ based on $\mathscr{W}^u_{c-}$ (Fig. 5(a)–(d), (g), and (h)). The CPU times are 112.38 min and 53.67 min, respectively. Due to the different descent directions of ρ^u and ρ^M, the latter result is just a value in the former iterations.

Next, let grasp (*f*) equilibrate a dynamic external wrench $\boldsymbol{w}_{\mathrm{ext}} = [\boldsymbol{f}^{\mathrm{T}}_{\mathrm{ext}}\ \boldsymbol{m}^{\mathrm{T}}_{\mathrm{ext}}]^{\mathrm{T}}$, where

$$\boldsymbol{f}_{\mathrm{ext}} = [\,-0.7\cos 0.2\pi t - 0.4\sin 0.2\pi t - 7.8 \quad 0.2\cos 0.2\pi t - 1.4\sin 0.2\pi t + 2.2 \quad -2.7\cos 0.2\pi t + 2\,]^{\mathrm{T}},$$
$$\boldsymbol{m}_{\mathrm{ext}} = [\,-0.7\cos 0.2\pi t - 1.1\sin 0.2\pi t + 0.8 \quad \cos 0.2\pi t - 0.8\sin 0.2\pi t - 1.2 \quad 0.7\cos 0.2\pi t + 2.5\,]^{\mathrm{T}}.$$

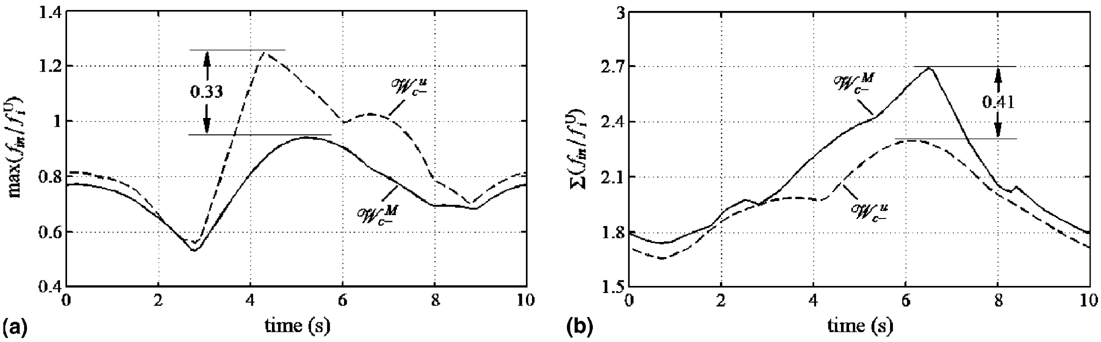

Fig. 6. Comparison of the DFD algorithm results based on $\mathscr{W}^u_{c-}$ (dashed curve) and $\mathscr{W}^M_{c-}$ (solid curve) for a certain external wrench. Naturally the former aiming to minimize $\sum_{i=1}^{m}(f_{in}/f_i^{\mathrm{U}})$ is above the latter aiming to minimizing $\max_{1\leqslant i\leqslant m}(f_{in}/f_i^{\mathrm{U}})$ in (a) and under the latter in (b). Note that the former exceeds the upper bound 1, while the latter goes well.

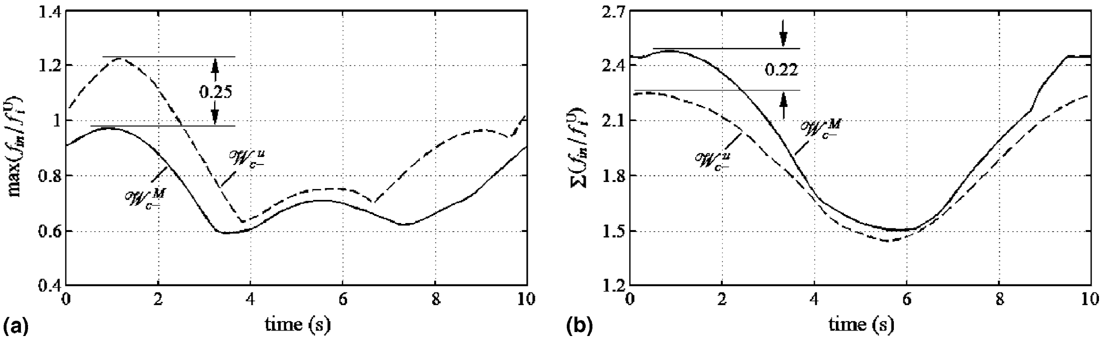

Fig. 7. Comparison of the DFD algorithm results based on $\mathscr{W}^u_{c-}$ (dashed curve) and $\mathscr{W}^M_{c-}$ (solid curve) for another external wrench. The situations are similar to Fig. 6.

Computing DFD based on $\mathscr{W}^{u}_{c-}$ and based on $\mathscr{W}^{M}_{c-}$, respectively, we obtain Fig. 6, which shows that the former contact forces exceed the upper bounds somewhere but the latter are always below. Originally, $\mathscr{W}^{M}_{c-}$ has 14 640 elements. Then the online phase takes over 2 s at a point. Using the method given in Section 4.2, we pick out 275 elements with $\boldsymbol{w}_{\text{ext}}(t)$ at $t = 1, 2, \ldots, 10$ and $-\sum_{t=1}^{10}\boldsymbol{w}_{\text{ext}}(t)$. The CPU time for a point is reduced to 62.10 ms. We also try another external wrench as follow:

$$\boldsymbol{f}_{\text{ext}} = [-\cos 0.2\pi t - 0.1\sin 0.2\pi t - 6.6 \;\; 0.1\cos 0.2\pi t - 1.2\sin 0.2\pi t + 0.5 \;\; -2.2\cos 0.2\pi t + 2.8]^{\mathrm{T}},$$
$$\boldsymbol{m}_{\text{ext}} = [-\cos 0.2\pi t + 0.3\sin 0.2\pi t - 1.3 \;\; 0.2\cos 0.2\pi t + 1.2\sin 0.2\pi t + 0.3 \;\; 0.7\cos 0.2\pi t - 2]^{\mathrm{T}}.$$

The results of the DFD algorithm based on $\mathscr{W}^{u}_{c-}$ and based on $\mathscr{W}^{M}_{c-}$ are depicted in Fig. 7. The optimal contact forces still differ significantly. The CPU times for computing them are 46.20 ms and 65.64 ms, respectively.

6. Conclusions

When a system involves a number of the-less-the-better quantities of the same attribute, normally its overall quality is determined by the maximum quantity, which we ought to limit and minimize. However, achieving such a goal is often complicated and difficult. As an easier but approximate way, people may limit and minimize their sum instead.

The multifingered grasp involving the contact forces is a typical system of this kind. In the early 1990s, taking the sum of the normal force components or their maximum as a measure of the overall force magnitude, two grasp quality criteria were proposed without preference [22,23]. Since not the sum but the maximum is directly related to the actuator power as well as the strength of the grasped object and the grasping mechanism, of course the latter is the reasonable choice. Nevertheless, the previous literatures on OGP and DFD always adopted the former as the optimization criterion [9,21–36]. Only [38] is an exception. Its approach is totally different from the others.

This paper points out that the former cannot assess the grasp goodness appropriately. The convex hull $\mathscr{W}^{u}_{c-}$ (resp. $\mathscr{W}^{M}_{c-}$) of the *union* (resp. *Minkowski sum*) of the primitive wrenches contains all the feasible resultant wrenches under the limitation of the *sum* (resp. *maximum*). Due to the excessive restraint on the contact force magnitude, the distance between the origin and $\mathscr{W}^{u}_{c-}$ is not suitable for evaluating the grasp goodness. Contrarily, the distance between the origin and $\mathscr{W}^{M}_{c-}$ properly reflects the load capacity of a grasp and is an ideal criterion. A precise and unified computational method is put forward. Because the different criteria have different descent directions, the proposed OGP algorithm with the same initial conditions may bring about different optimal grasps. Moreover, we show that DFD with the sum (resp. maximum) minimized can be transformed into a convex analysis problem based on $\mathscr{W}^{u}_{c-}$ (resp. $\mathscr{W}^{M}_{c-}$). By solving the problem, a unified DFD algorithm for minimizing the sum or the maximum is developed. It is demonstrated that the two kinds of optimal contact forces are often quite different, especially for 3D multifingered grasps.

Acknowledgement

This work was supported by the National Natural Science Foundation of China under Grant 59685004.

Appendix A. The L_2 distance between the origin and a nonempty compact convex set

Let $\boldsymbol{0}$ and S be the origin and a nonempty compact convex subset of $\mathbb{R}^d$, respectively. The L_2 *distance* between $\boldsymbol{0}$ and S, denoted by $\rho(\boldsymbol{0}, S)$, is defined by

$$\rho(\boldsymbol{0}, S) = \begin{cases} \min\limits_{x_b \in S} \|\boldsymbol{x}_b\|, & \text{if } \boldsymbol{0} \notin \operatorname{int} S, \\ -\min\limits_{x_b \in \operatorname{bd} S} \|\boldsymbol{x}_b\|, & \text{if } \boldsymbol{0} \in \operatorname{int} S, \end{cases}$$

where $\|\cdot\|$, int(·) and bd(·) denote the L_2 norm of a vector, the interior and the boundary of a set. When $\boldsymbol{0} \notin S$, the value $\rho(\boldsymbol{0}, S)$ is positive and equal to the radius of the largest open ball centered at $\boldsymbol{0}$ without intersecting S.

When $\boldsymbol{0} \in \operatorname{bd} S$ or $\boldsymbol{0} \in S$ with $\operatorname{int} S = \emptyset$, $\rho(\boldsymbol{0}, S) = 0$. When $\boldsymbol{0} \in \operatorname{int} S$, $\rho(\boldsymbol{0}, S)$ is negative and its additive inverse equals the radius of the largest closed ball centered at $\boldsymbol{0}$ contained in S. In what follows, we derive a general and efficient computational method of $\rho(\boldsymbol{0}, S)$.

Lemma 1. *Suppose that B is a convex subsets of* $\mathbb{R}^d$. *If* $\operatorname{int} B \neq \emptyset$ *and* $S \cap \operatorname{int} B = \emptyset$, *then there exists a hyperplane H separating S and B.*

Proof. See [39], p. 36. □

Lemma 2. *Through each boundary point of S there is a hyperplane supporting S.*

Proof. See [39], p. 41. □

Lemma 3. *There exists a point* $\boldsymbol{x}_b \in S$ (*if* $\boldsymbol{0} \notin \operatorname{int} S$) *or* $\boldsymbol{x}_b \in bd S$ (*if* $\boldsymbol{0} \in int S$) *such that* $\rho(\boldsymbol{0}, S) = \|\boldsymbol{x}_b\|$ *or* $\rho(\boldsymbol{0}, S) = -\|\boldsymbol{x}_b\|$. *Moreover, there is a hyperplane of the form* $H = \{\boldsymbol{x} \in \mathbb{R}^d | \boldsymbol{x}^{\mathrm{T}}\boldsymbol{z} = -\rho(\boldsymbol{0}, S)\}$ *supporting S at* $\boldsymbol{x}_b$ *such that* $\boldsymbol{x}^T\boldsymbol{z} \leqslant -\rho(\boldsymbol{0}, S)$ *for all* $\boldsymbol{x} \in S$, *where* $\boldsymbol{z}$ *is a unit normal to H.*

Proof. The compactness of S and the continuity of the L_2 norm imply the existence of such a point $\boldsymbol{x}_b$.

Case 1 ($\boldsymbol{0} \notin \operatorname{int} S$; then $\rho(\boldsymbol{0}, S) = \|\boldsymbol{x}_b\|$). If $\boldsymbol{x}_b \neq \boldsymbol{0}$, then there is a closed ball B centered at $\boldsymbol{0}$ such that $S \cap B = \boldsymbol{x}_b$, $\operatorname{int} B \neq \emptyset$, and $S \cap \operatorname{int} B = \emptyset$. From Lemma 1, there is a hyperplane H separating S and B; thus H supports S and B at $\boldsymbol{x}_b$. The supporting hyperplane of B at $\boldsymbol{x}_b$ can be written as $H = \{\boldsymbol{x} \in \mathbb{R}^d | \boldsymbol{x}^{\mathrm{T}}(-\boldsymbol{x}_b/\|\boldsymbol{x}_b\|) = -\|\boldsymbol{x}_b\|\}$, and obviously $\boldsymbol{x}^{\mathrm{T}}(-\boldsymbol{x}_b/\|\boldsymbol{x}_b\|) \leqslant -\|\boldsymbol{x}_b\|$ for all $\boldsymbol{x} \in S$. If $\boldsymbol{x}_b = \boldsymbol{0}$, then $\boldsymbol{0} \in \operatorname{bd} S$ or $\boldsymbol{0} \in S$ with $\operatorname{int} S = \emptyset$. From Lemma 2, there is a hyperplane supporting S at $\boldsymbol{0}$, or containing S; anyway, there is a supporting hyperplane of S at $\boldsymbol{0}$. Properly selecting its normal $\boldsymbol{z}$, we can also obtain $\boldsymbol{x}^{\mathrm{T}}\boldsymbol{z} \leqslant 0$ for all $\boldsymbol{x} \in S$.

Case 2 ($\boldsymbol{0} \in \operatorname{int} S$; then $\rho(\boldsymbol{0}, S) = -\|\boldsymbol{x}_b\|$). The closed ball B of radius $\|\boldsymbol{x}_b\|$ centered at $\boldsymbol{0}$ is contained in S such that $\boldsymbol{x}_b \in \operatorname{bd} S \cap \operatorname{bd} B$. From Lemma 2, there is a hyperplane H supporting S at $\boldsymbol{x}_b$, which in turn supports B at $\boldsymbol{x}_b$. The hyperplane can be written as $H = \{\boldsymbol{x} \in \mathbb{R}^d | \boldsymbol{x}^{\mathrm{T}}(\boldsymbol{x}_b/\|\boldsymbol{x}_b\|) = \|\boldsymbol{x}_b\|\}$, and clearly $\boldsymbol{x}^{\mathrm{T}}(\boldsymbol{x}_b/\|\boldsymbol{x}_b\|) \leqslant \|\boldsymbol{x}_b\|$ for all $\boldsymbol{x} \in S$. □

Let p_S denote the *support function* of S, which is the real-valued function defined by

$$p_S(\boldsymbol{z}) = \sup_{x \in S} \boldsymbol{x}^{\mathrm{T}}\boldsymbol{z}.$$

Lemma 4. *Let* $\boldsymbol{z}$ *be a unit point of* $\mathbb{R}^d$ *and B the ball of radius* $|p_S(\boldsymbol{z})|$ *centered at the origin* $\boldsymbol{0}$ *of* $\mathbb{R}^d$. *Then the hyperplane* $H = \{\boldsymbol{x} \in \mathbb{R}^d | \boldsymbol{x}^{\mathrm{T}}\boldsymbol{z} = p_S(\boldsymbol{z})\}$ *supports S and B. Moreover, If* $p_S(\boldsymbol{z}) < 0$, *then H separates S and B. If* $p_S(\boldsymbol{z}) \geqslant 0$, *then S and B lie in the same closed half-space determined by H.*

Proof. Referring to [39], p. 206, the proof is straightforward. □

Theorem 1

$$\rho(\boldsymbol{0}, S) = -\min_{\|\boldsymbol{z}\|=1} p_S(\boldsymbol{z})$$

Proof. From Lemma 3, there is a hyperplane $H = \{\boldsymbol{x} \in \mathbb{R}^d | \boldsymbol{x}^{\mathrm{T}}\boldsymbol{z} = -\rho(\boldsymbol{0}, S)\}$ supporting S at $\boldsymbol{x}_b$ such that $\boldsymbol{x}^{\mathrm{T}}\boldsymbol{z} \leqslant -\rho(\boldsymbol{0}, S)$ for all $\boldsymbol{x} \in S$. Thus $p_S(\boldsymbol{z}) = -\rho(\boldsymbol{0}, S)$ and $\min_{|\boldsymbol{z}|=1} p_S(\boldsymbol{z}) \leqslant -\rho(\boldsymbol{0}, S)$. Assume $p_S(\boldsymbol{z}') < -\rho(\boldsymbol{0}, S)$ for some unit $\boldsymbol{z}'$. Let B' be the ball of radius $|p_S(\boldsymbol{z}')|$ centered at $\boldsymbol{0}$. If $\boldsymbol{0} \notin \operatorname{int} S$, then $\rho(\boldsymbol{0}, S) = \|\boldsymbol{x}_b\| < |p_S(\boldsymbol{z}')|$, and $\boldsymbol{x}_b \in \operatorname{int} B'$. From $\boldsymbol{x}_b \in S$ it next follows that S and B' cannot be separated. But from Lemma 4 there is a hyperplane H' separating S and B', which is a contradiction. If $\boldsymbol{0} \in \operatorname{int} S$, then $|p_S(\boldsymbol{z}')| < -\rho(\boldsymbol{0}, S) = \|\boldsymbol{x}_b\|$, and $B' \subset \operatorname{int} S$, which implies that S and B' cannot be supported by the same hyperplane. But from Lemma 4 there is a hyperplane H' supporting S and B', which is also a contradiction. Therefore, $\min_{|\boldsymbol{z}|=1} p_S(\boldsymbol{z}) = -\rho(\boldsymbol{0}, S)$. □

Appendix B. On computing the coefficients with the minimum sum in positive combination

Let S be a finite set of points $\boldsymbol{x}_j$, $j = 1, 2, \ldots, n$ of $\mathbb{R}^d$ and $\boldsymbol{w}$ a point other than the origin. Assume that the convex hull S_c of S contains the origin as an interior point. Then the point $\boldsymbol{w}$ can be expressed by

$$\boldsymbol{w} = \sum_{j=1}^{n} c_j \boldsymbol{x}_j = \boldsymbol{A}\boldsymbol{c}, \quad c_j \geqslant 0 \text{ for } j = 1, 2, \ldots, n,$$

where $\boldsymbol{A} = [\,\boldsymbol{x}_1 \quad \boldsymbol{x}_2 \quad \cdots \quad \boldsymbol{x}_n\,] \in \mathbb{R}^{d\times n}$ and $\boldsymbol{c} = [\,c_1 \quad c_2 \quad \cdots \quad c_n\,]^{\mathrm{T}} \in \mathbb{R}^n$. The problem is to find $c_j, j = 1,2,\ldots,n$ with minimum $\sigma = \sum_{j=1}^{n} c_j$.

Let $\boldsymbol{x}_e = \boldsymbol{w}/\sigma$. Then $\boldsymbol{x}_e \in S_c$ and $\sigma = \|\boldsymbol{w}\|/\|\boldsymbol{x}_e\|$. Clearly, σ attains a minimum when $\boldsymbol{x}_e$ is on the boundary bd S_c of S_c. Furthermore, $\boldsymbol{w}$ can be restricted to a nonnegative combination of the elements of S that fall on a hyperplane supporting S_c at $\boldsymbol{x}_e$. Partition $\boldsymbol{A}$ into $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$ where $\boldsymbol{A}_1$ consists of such elements and $\boldsymbol{A}_2$ consists of the others. Correspondingly, $\boldsymbol{c}$ is partitioned into $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$. Thus $\boldsymbol{c}$ can be calculated by

$$\boldsymbol{c}_1 = \boldsymbol{A}_1^{+}\boldsymbol{w} \text{ and } \boldsymbol{c}_2 = \boldsymbol{0},$$

where $\boldsymbol{A}_1^{+}$ is the pseudoinverse of $\boldsymbol{A}_1$.

To find a hyperplane supporting S_c at $\boldsymbol{x}_e$, we turn to the polar set S_c^* of S_c and the support function $p_{S_c^*}$ of S_c^*, which are defined by

$$S_c^* = \{\boldsymbol{z} \in \mathbb{R}^d | \boldsymbol{x}^{\mathrm{T}}\boldsymbol{z} \leqslant 1 \text{ for all } \boldsymbol{x} \in S_c\},$$
$$p_{S_c^*}(\boldsymbol{w}) = \sup_{z\in S_c^*} \boldsymbol{w}^{\mathrm{T}}\boldsymbol{z}.$$

Theorem 2. *Let $\boldsymbol{z}_b$ be a point in S_c^* such that $p_{S_c^*}(\boldsymbol{w}) = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{z}_b$. The following statements are true*:

1. $p_{S_c^*}(\boldsymbol{w})^{-1}\boldsymbol{w} \in \mathrm{bd}\, S_c$.
2. *The hyperplane $H = \{\boldsymbol{x} \in \mathbb{R}^d | \boldsymbol{z}_b^{\mathrm{T}}\boldsymbol{x} = 1\}$ supports S at the point $p_{S_c^*}(\boldsymbol{w})^{-1}\boldsymbol{w}$.*

Proof. See [10,36]. □

Theorem 2, point (1) implies that $\boldsymbol{x}_e = p_{S_c^*}(\boldsymbol{w})^{-1}\boldsymbol{w}$ and $\sigma_{\min} = p_{S_c^*}(\boldsymbol{w})$. Point (2) indicates that H is just the hyperplane we are looking for. From $S_c = \mathrm{conv}S$ it follows that $S_c^* = S^*$ and $p_{S_c^*}(\boldsymbol{w}) = \sup_{z\in S^*} \boldsymbol{w}^{\mathrm{T}}\boldsymbol{z}$. Thus $\boldsymbol{z}_b$ is the optimal solution of the following problem:

$$\begin{cases} \text{Maximize } \boldsymbol{w}^{\mathrm{T}}\boldsymbol{z} \\ \text{subject to } \boldsymbol{x}^{\mathrm{T}}\boldsymbol{z} \leqslant 1, \boldsymbol{x} \in S. \end{cases}$$

References

[1] J.K. Salisbury, B. Roth, Kinematic and force analysis of articulated hands, ASME Journal of Mechanisms, Transmissions, and Automation in Design 105 (1) (1983) 35–41.
[2] B. Mishra, J.T. Schwarz, M. Sharir, On the existence and synthesis of multifingered positive grips, Algorithmica 2 (4) (1987) 541–558.
[3] R.M. Murray, Z.X. Li, S.S. Sastry, A Mathematical Introduction to Robotic Manipulation, CRC Press, Boca Raton, 1994, pp. 223–229.
[4] A. Bicchi, On the closure properties of robotics grasping, International Journal of Robotics Research 14 (4) (1995) 319–334.
[5] W.-H. Qian, H. Qiao, S.K. Tso, Synthesizing two-fingered grippers for positioning and identifying objects, IEEE Transactions on Systems, Man, and Cybernetics B 31 (4) (2001) 602–615.
[6] Y.-L. Xiong, H. Ding, M.Y. Wang, Quantitative analysis of inner force distribution and load capacity of grasps and fixtures, ASME Journal of Manufacturing Science and Engineering 124 (2) (2002) 444–455.
[7] Y. Zheng, W.-H. Qian, Coping with the grasping uncertainties in force-closure analysis, International Journal of Robotics Research 24 (4) (2005) 311–327.
[8] B.-R. Zuo, W.-H. Qian, A force-closure test for soft multi-fingered grasps, Science in China ser. E 41 (1) (1998) 62–69.

[9] Y.-H. Liu, Qualitative test and force optimization of 3-D frictional form-closure grasps using linear programming, IEEE Transactions on Robotics and Automation 15 (1) (1999) 163–173.
[10] Y. Zheng, W.-H. Qian, Simplification of the ray-shooting based algorithm for 3D force-closure test, IEEE Transactions on Robotics 21 (3) (2005) 470–473.
[11] X.-Y. Zhu, H. Ding, Y. Wang, A numerical test for the closure properties of 3D grasps, IEEE Transactions on Robotics and Automation 20 (3) (2004) 543–549.
[12] V.-D. Nguyen, Constructing force-closure grasps, International Journal of Robotics Research 7 (3) (1988) 3–16.
[13] Y.C. Park, G.P. Starr, Grasp synthesis of polygonal objects using a three-fingered robot hand, International Journal of Robotics Research 11 (3) (1992) 163–184.
[14] I.-M. Chen, J.W. Burdick, Finding antipodal point grasps on irregular shaped objects, IEEE Transactions on Robotics and Automation 9 (4) (1993) 507–512.
[15] C.P. Tung, A.C. Kak, Fast construction of force-closure grasps, IEEE Transactions on Robotics and Automation 12 (4) (1996) 615–626.
[16] J. Ponce, D. Stam, B. Faverjon, On computing two-fingered force-closure grasps of curved 2D objects, International Journal of Robotics Research 12 (3) (1993) 263–273.
[17] J. Ponce, B. Faverjon, On computing three-fingered force-closure grasps of polygonal objects, IEEE Transactions on Robotics and Automation 11 (6) (1995) 868–881.
[18] J. Ponce, S. Sullivan, A. Sudsang, J.D. Boissonnat, J.P. Merlet, On computing four-fingered equilibrium and force-closure grasps of polyhedral objects, International Journal of Robotics Research 16 (1) (1997) 11–35.
[19] Y.-H. Liu, Computing *n*-finger form-closure grasps on polygonal objects, International Journal of Robotics Research 19 (2) (2000) 149–158.
[20] J.-W. Li, H. Liu, H.-G. Cai, On computing three-finger force-closure grasps of 2-D and 3-D objects, IEEE Transactions on Robotics and Automation 19 (1) (2003) 155–161.
[21] Y.-H. Liu, M.-L. Lam, D. Ding, A complete and efficient algorithm for searching 3D form-closure grasps in the discrete domain, IEEE Transactions on Robotics and Automation 20 (5) (2004) 805–816.
[22] D. Kirkpatrick, B. Mishra, C. Yap, Quantitative Steinitz's theorem with applications to multi-fingered grasping, in: Proceedings of the twenty-second annual ACM Symposium on Theory of Computing, 1990, pp. 341–351.
[23] C. Ferrari, J. Canny, Planning optimal grasps, in: Proceedings of the IEEE International Conference on Robotics and Automation, 1992, pp. 2290–2295.
[24] X.-Y. Zhu, J. Wang, Synthesis of force-closure grasps on 3-D objects based on the Q distance, IEEE Transactions on Robotics and Automation 19 (4) (2003) 669–679.
[25] X.-Y. Zhu, H. Ding, J. Wang, Grasp analysis and synthesis based on a new quantitative measure, IEEE Transactions on Robotics and Automation 19 (6) (2003) 942–953.
[26] J. Kerr, B. Roth, Analysis of multifingered hands, International Journal of Robotics Research 4 (4) (1985) 3–17.
[27] F.T. Cheng, D.E. Orin, Efficient algorithm for optimal force distribution—the compact-dual LP method, IEEE Transactions on Robotics and Automation 6 (2) (1990) 178–187.
[28] M. Nahon, J. Angeles, Real-time force optimization in parallel kinematic chains under inequality constraints, IEEE Transactions on Robotics and Automation 8 (4) (1992) 439–450.
[29] M. Buss, H. Hashimoto, J.B. Moore, Dextrous hand grasping force optimization, IEEE Transactions on Robotics and Automation 12 (3) (1996) 406–417.
[30] M. Buss, L. Faybusovich, J.B. Moore, Dikin-type algorithms for dextrous grasping force optimization, International Journal of Robotics Research 17 (8) (1998) 831–839.
[31] L. Han, J.C. Trinkle, Z.X. Li, Grasp analysis as linear matrix inequality problems, IEEE Transactions on Robotics and Automation 16 (6) (2000) 663–674.
[32] U. Helmke, K. Hüper, J.B. Moore, Quadratically convergent algorithms for optimal dextrous hand grasping, IEEE Transactions on Robotics and Automation 18 (2) (2002) 138–146.
[33] G.F. Liu, Z.X. Li, Real-time grasping-force optimization for multifingered manipulation: theory and experiments, IEEE/ASME Transactions on Mechatronics 9 (1) (2004) 65–77.
[34] G.F. Liu, J.J. Xu, Z.X. Li, On geometric algorithms for real-time grasping force optimization, IEEE Transactions on Control Systems Technology 12 (6) (2004) 843–859.
[35] Y.S. Xia, J. Wang, L.-M. Fok, Grasping-force optimization for multifingered robotic hands using a recurrent neural network, IEEE Transactions on Robotics and Automation 20 (3) (2004) 549–554.
[36] Y. Zheng, W.-H. Qian, Linearizing the soft finger contact constraint with application to dynamic force distribution in multifingered grasping, Science in China ser. E 48 (2) (2005) 121–130.
[37] B.-R. Zuo, W.-H. Qian, A general dynamic force distribution algorithm for multifingered grasping, IEEE Transactions on Systems, Man, and Cybernetics B 30 (1) (2000) 185–192.
[38] Y. Zheng, W.-H. Qian, Dynamic force distribution in multifingered grasping by decomposition and positive combination, IEEE Transactions on Robotics 21 (4) (2005) 718–726.
[39] S.R. Lay, Convex Sets and Their Application, John Wiley & Sons, New York, 1982.