On the Efficient Computation of Independent Contact Regions for Force Closure Grasps

Robert Krug, Dimitar Dimitrov, Krzysztof Charusta and Boyko Iliev
AASS Research Center, Örebro University
Email: {robert.krug, dimitar.dimitrov, krzysztof.charusta, boyko.iliev}@oru.se

***Abstract*— Since the introduction of independent contact regions in order to compensate for shortcomings in the positioning accuracy of robotic hands, alternative methods for their generation have been proposed. Due to the fact that (in general) such regions are not unique, the computation methods used usually reflect the envisioned application and/or underlying assumptions made. This paper introduces a parallelizable algorithm for the efficient computation of independent contact regions, under the assumption that a user input in the form of initial guess for the grasping points is readily available. The proposed approach works on discretized 3D-objects with any number of contacts and can be used with any of the following models: frictionless point contact, point contact with friction and soft finger contact. An example of the computation of independent contact regions comprising a non-trivial task wrench space is given.**

I. Introduction

Evaluating the "goodness" of a given multifingered grasp while accounting for the capabilities of the grasping device is an important issue in dexterous manipulation. For a large class of grasps the force closure property is desirable. Loosely speaking, force closure means the ability of the grasp to immobilize the grasped object influenced by an arbitrary external disturbance, if the manipulator is capable of exerting sufficiently large contact forces on the object [1]. Contact force vectors and resulting torque vectors are commonly concatenated to wrench vectors. Mishra et al. [2] showed that a grasp is force closure, if the convex hull spanned by the contact wrenches contains a neighborhood of the origin.

However, in many cases force closure is just a necessary, and not a sufficient requirement. Usually it is desirable to specify additional conditions in order to evaluate a grasp. There are many quality measures proposed in the literature (see [3] for a survey). A good grasp should be able to efficiently withstand forces, which are likely to occur during the performed task. If nothing about the task is known, a common measure is the radius of the largest origin-centered insphere of the *Grasp Wrench Space* (GWS), which was proposed by Kirkpatrick et al. [4]. The GWS is defined as the convex hull over the set of all wrenches that the manipulator can exert on the object for a given grasp. In this definition it is presumed that the sum of the magnitude of the grasping forces is bounded. Ferrari and Canny [5] introduced the physically more relevant convex hull over the Minkowski sum of the grasp wrenches. This implies that no more than a force of a given magnitude is applied at each grasping point. A way to incorporate the whole object geometry into the grasp assessment was suggested by Pollard [6]. She introduces the *Object Wrench Space* (OWS) (which represents the best possible grasp), and formulates a quality measure as the scale of the largest OWS that fits entirely in the GWS. Several works have integrated disturbance forces on the object geometry in the grasp evaluation [6][7][8].

From the viewpoint of a mechanic manipulator, not only the ability to resist disturbances, but also the robustness of a grasp is an important factor. Grasps which are less sensitive to modeling and positioning errors are desirable. In this context, the notion of *Independent Contact Regions* (ICR) was suggested by Nguyen [9]. He defined the set of optimal independent regions with the largest minimal radius, which yield a force closure-grasp if each finger is placed anywhere within its respective region. The concept was extended to the computation of independent regions for three-finger grasps on planar objects [10], and four-finger grasps of polyhedral objects by Ponce et al. [11]. The latter approach has a number of drawbacks: (i) three conditions for force closure are presented, however, two are disregarded in the later analysis due to their nonlinear structure, leading to the possibility of excluding viable candidate regions; (ii) it is not clear, how to compute ICR given a bound on the possible disturbance wrenches; (iii) it is unclear, how the approach could be extended to five or more fingers. The above problems were addressed by Pollard in [12], where the synthesis of grasps on 3-D objects with a large number of contacts is discussed. Furthermore, a task related quality measure is incorporated in the evaluation of ICR. The computation is based on geometric reasoning in the wrench space and requires the solution of a Linear Programming problem (LP). Still, a detailed discussion of an efficient algorithm for the generation of ICR is not presented. Roa and Suárez [13] suggested an algorithm, which grows independent regions for precision grasps on discretized objects. However, their method is very sensitive to the choice of friction coefficient and more restrictive than the approach presented in [12].

In this work, an in-depth analysis about the geometric relations in the context of independent contact regions is provided, along with an extension of the approaches presented in [12] and [13]. We introduce an efficient parallelizable algorithm for determining ICR for a given fixed set of contact points on discretized 3D-objects with any number N of contacts which satisfy the force closure condition. The algorithm is capable of determining the regions for a non-trivial disturbance wrench set and can be used with:

frictionless point contact, point contact with friction and soft finger contact models.

The assumptions and required background are provided in Section II. In Section III we present our efficient algorithm for computing independent contact regions and finally Section IV contains a numerical evaluation.

II. Background

A. Nomenclature

N	number of contact points in a grasp,
S	number of points on the surface of an object,
L	number of wrenches used in a contact model,
H	number of hyperplanes bounding a convex hull,
n	index used for points in a given grasp,
s	index used for points on the surface of the object,
l	index used for wrenches in a contact model,
h	index used for hyperplanes.

Bold letters are used to denote matrices and vectors. The i^{th} element of a set $\mathcal{C}$ is denoted by $\mathcal{C}^{(i)}$.

B. Assumptions & Problem Description

A sufficiently discretized representation of the target objects surface, given as a polygonal mesh of points $\boldsymbol{p}_s$ $(s = 1, ..., S)$ with corresponding inward-pointing unit normals $\hat{\boldsymbol{n}}_s$ is required. The reference frame is fixed in the Center of Mass (CoM) of the object. Each point $\boldsymbol{p}_s$ has associated neighboring points, defined as the ones connected to $\boldsymbol{p}_s$ by an edge of the mesh. We presume, that a "reasonable" set of tasks $\mathcal{T}_t$ $(t = 1, ..., T)$ is specified as sets of disturbance wrenches, which needs to be resisted by the grasp. An initial force closure grasp, able to withstand the Minkowski sum of the given sets of disturbance wrenches $\mathcal{T}_t$ is provided. Such a grasp is defined as a set of N contact points on the objects surface $\mathcal{G} = \{\boldsymbol{p}_1, \cdots, \boldsymbol{p}_N\}$. The necessary starting grasp could be acquired by means of human demonstration or by one of the algorithms proposed for the synthesis of force closure grasps [13][14]. Furthermore, quasi-static conditions are assumed.

We are interested in the computation of ICR, defined as the N independent regions $\mathcal{C}_n$, each one associated with a contact point $\boldsymbol{p}_n$ of the original force closure grasp. The sets $\mathcal{C}_n$ contain points on the target objects surface, each of which can replace $\boldsymbol{p}_n$ in $\mathcal{G}$. Any grasp composed of N contact points, where one point is picked from each region $\mathcal{C}_n$, will be force closure and preserve the task requirements. An example of ICR for a four-fingered frictional grasp on the model of a cup is shown in Fig. 1.

C. Application

If expected disturbances are represented as a meaningful set of tasks, the size of the independent regions can be directly related to the required positioning accuracy. If each finger "aims" at the center of its respective region, larger ICR provide increased robustness to finger positioning errors. Kim et al. [15] formalized this notion by introducing the *Uncertainty Grasp Index*, which is described as the sum of the distances between the grasping points and the center of the corresponding independent region.

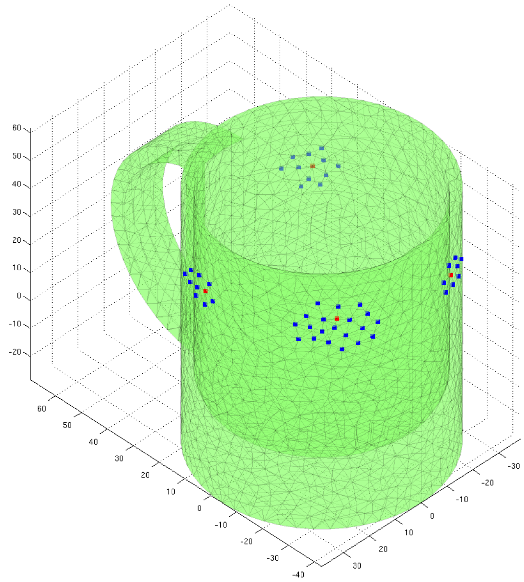

Fig. 1. *Independent Contact Regions:* Red squares represent the original grasping points, blue squares the independent contact regions. The regions are computed considering possible disturbances specified in Section IV-B, utilizing frictional hard-finger contacts.

D. Contact Models

We first consider **frictional point contacts** between the target object and the fingers of the gripper. The friction-coefficient according the Coulomb friction model is denoted as μ. In order to prevent slipping, a force $\boldsymbol{f}_s$ applied at a point $\boldsymbol{p}_s$ has to fulfill the following constraint:

$$||\boldsymbol{f}_s - (\boldsymbol{f}_s \cdot \hat{\boldsymbol{n}}_s)\hat{\boldsymbol{n}}_s|| \leq \mu(\hat{\boldsymbol{n}}_s \cdot \boldsymbol{f}_s). \tag{1}$$

This describes a nonlinear friction cone, which can be approximated by a L-sided convex polyhedron. The set of forces with magnitude F_G along the L edges of the discretized cone located at contact point $\boldsymbol{p}_s$ is denoted in matrix notation as $\boldsymbol{F}_s = [\boldsymbol{f}_1(\boldsymbol{p}_s), \cdots, \boldsymbol{f}_L(\boldsymbol{p}_s)]$. Thus, the grasping force $\boldsymbol{f}_s$ is given by:

$$\boldsymbol{f}_s = \boldsymbol{F}_s\boldsymbol{\alpha}_s, \quad \boldsymbol{\alpha}_s \geq \boldsymbol{0}, \quad ||\boldsymbol{\alpha}_s||_{L_1} \leq 1. \tag{2}$$

The force $\boldsymbol{f}_s$ creates a torque $\boldsymbol{\tau}_s = (\boldsymbol{p}_s \times \boldsymbol{f}_s)$. Force and torque vectors can be concatenated to a wrench vector $\boldsymbol{w}_s$:

$$\boldsymbol{w}_s = \begin{pmatrix} \boldsymbol{f}_s \\ \boldsymbol{\tau}_s/\lambda \end{pmatrix}, \quad \lambda = \max_s(||\boldsymbol{p}_s||). \tag{3}$$

Dividing the torque parts by the largest possible torque arm λ guarantees scale invariance [6]. The wrenches generated by forces $\boldsymbol{f}_l$ along an edge of the discretized friction cone are referred to as *primitive wrenches*. For a given contact point $\boldsymbol{p}_s$, the set of primitive wrenches is defined as:

$$\mathcal{W}_s = \{\boldsymbol{w}_1(\boldsymbol{p}_s), \cdots, \boldsymbol{w}_L(\boldsymbol{p}_s)\}. \tag{4}$$

The **soft finger contact** model according to [16] allows for additional torsional moments around the local contact normal $\hat{\boldsymbol{n}}_s$. Here, the set of primitive wrenches in Equation 4 needs to be supplemented by the according wrenches. In the soft finger contact model, scaling the wrench vectors by the largest possible torque arm λ does not grant scale-invariance any more. This is due to the fact, that the additional wrenches do not depend on the object geometry. Still, scaling imparts invariance to the chosen units of length.

In the case of the **frictionless point contact** model, the friction coefficient μ is zero and $\boldsymbol{f}_s$ acts along the surface normal. In this case, the set $\mathcal{W}_s$ just holds one wrench generated by the respective normal force. Given a Grasp $\mathcal{G}$, the discrete GWS is described by the convex hull over the union of the N primitive wrench sets belonging to the grasping points $\boldsymbol{p}_n$:

$$\text{GWS} = \mathcal{CH}\left(\bigcup\{\mathcal{W}_1, \cdots, \mathcal{W}_N\}\right). \tag{5}$$

Equation (5) characterizes the space of wrenches, which can be exerted to the grasped object when the sum of the magnitudes of all finger forces is bounded by a value F_G. Since the applied forces are proportional to the current in the actuators, this can be seen as a limitation due to a common power source [5].

E. Task Model

Tasks are represented as sets of disturbance wrenches which needs to be resisted. Given T tasks $\mathcal{T}_t$, we denote the *Task Wrench Space* (TWS) as the convex hull over the Minkowski sum of the tasks:

$$\text{TWS} = \mathcal{CH}\left(\bigoplus\{\mathcal{T}_1, \cdots, \mathcal{T}_T\}\right). \tag{6}$$

One frequently used representation of the TWS is the largest origin-centered insphere of the GWS. Yet, this gives only weak protection against disturbance forces on the extreme parts of the object geometry and might pose unnecessary restrictions by protecting against disturbances which are unlikely to occur. A physically better motivated way to describe a task $\mathcal{T}_t$, is by wrenches resulting from a maximum number of S possible disturbance forces, which can act on any point $\boldsymbol{p}_s$ on the objects surface. The sum of the magnitudes of all disturbance forces is denoted as F_D, which has to be smaller or equal F_G. This way of modeling a task is equivalent to a scaled OWS [6] and shall be denoted as OWS_D. It is presumed, that the disturbance forces are caused by frictionless point contacts of the object with the environment. Assuming a sufficiently high discretization of the object, wrenches resulting from frictional contact may be contained in OWS_D nevertheless. Otherwise they can easily be added [6].

Combining multiple independent tasks usually involves the computationally expensive Minkowski sum according to Equation (6). However, Borst et al. [8] have shown, that disturbances caused by the gravitational force F_{grav} can easily be incorporated in the OWS_D. If the CoM is used as torque origin, the OWS_D as well as the gravitational forces are tightly enclosed by a sphere in the force domain. Thus, it is possible to simply scale the force domain of the OWS_D by a factor $(1+F_{grav}/F_D)$, in order to consider disturbances caused by gravity as well.

III. Independent Contact Regions

Let the $\mathcal{H}$-representation of the convex hull defined in (5) be given as $(\boldsymbol{A}, \boldsymbol{b})$, where $\boldsymbol{A} = [\boldsymbol{n}_1, ..., \boldsymbol{n}_H]^T \in \mathbb{R}^{H\times K}$ is a matrix containing the inward-pointing unit normals to the bounding hyperplanes. The vector $\boldsymbol{b} = [b_1, ..., b_H]^T \in \mathbb{R}^H$ contains the distances to the origin. $K = 3$ if the object to be grasped is planar, and $K = 6$ when the object is three dimensional. From our assumptions it follows that the convex hull associated with the TWS will be contained in the GWS of $\mathcal{G}$. Hence, for all disturbance wrenches $\boldsymbol{w}_d \in$ TWS, $\boldsymbol{A}\boldsymbol{w}_d + \boldsymbol{b} \geq \boldsymbol{0}$. We define $b_h - \epsilon_h$ as the distance from the h^{th} hyperplane to the TWS, $i.e.$ the hyperplane defined by $(\boldsymbol{n}_h, \epsilon_h)$ is tangent to the TWS. The distances ϵ_h are combined in the vector $\boldsymbol{\epsilon} = [\epsilon_1, ..., \epsilon_H]^T \in \mathbb{R}^H$.

In addition to the $\mathcal{H}$-representation of the GWS, we need to define sets of indices $\varrho_{n,v}$, one for each $\boldsymbol{w}_l(\boldsymbol{p}_n)$ that is a vertex in the GWS. Let $\mathcal{V}_n$ be a set containing the indices of the vertices (in the GWS) associated with $\boldsymbol{p}_n$. Clearly, the number of elements in $\mathcal{V}_n$ is smaller or equal to L.

$$\varrho_{n,v} = \{h \,:\, \boldsymbol{n}_h^T \boldsymbol{w}_v(\boldsymbol{p}_n) + b_h = 0\,,\, v \in \mathcal{V}_n\}. \tag{7}$$

Thus, $h \in \varrho_{n,v}$ and $v \in \mathcal{V}_n$ imply, that the wrench $\boldsymbol{w}_v(\boldsymbol{p}_n)$ is a vertex and lies on the h^{th} hyperplane ($\mathcal{H}_h$). Let us denote the independent contact region associated with $\boldsymbol{p}_n$ by $\mathcal{C}_n$. By definition, $\mathcal{C}_n$ will contain points each of which can replace $\boldsymbol{p}_n$ in $\mathcal{G}$ and still preserve the task requirements. This implies that the convex hull spanned by the wrenches associated with any point in $\mathcal{C}_n$, combined with the wrenches associated with $N-1$ points, each chosen from one of the other $N-1$ independent regions, will contain the task disturbances. Adding a new point $\boldsymbol{p}_s$ to $\mathcal{C}_n$ by using a brute-force approach and testing whether $b_h - \epsilon_h \geq 0, \forall h$ (which requires the re-computation of (5)), for all possible grasps with points already in the other independent regions is not feasible. Instead, by defining search regions directly in the wrench space, Pollard [12] presented an easy to evaluate criterion for adding points in a given independent contact region. Figure 2 illustrates the core idea, which is based on geometric reasoning. It shows the convex hull $\mathcal{CH}(\mathcal{X})$, spanned by vectors $\boldsymbol{x}_i$ ($\mathcal{X} = \{\boldsymbol{x}_1 \cdots \boldsymbol{x}_I\}$), containing the origin. By convexity, $\mathcal{CH}(\mathcal{X})$ is fully contained in one of the half-spaces defined by the hyperplane $\mathcal{H}_f$, corresponding to facet f. Facet f is said to belong to the *visible region* of a point $\hat{\boldsymbol{x}}_i$ if that point lies in the half-space of $\mathcal{H}_f$ not including the origin ($i.e.$ $\hat{\boldsymbol{x}}_i$ "sees" f) [17]. Let $\mathcal{S}_i$ be the intersection of all half-spaces defined by hyperplanes corresponding to facets which contain $\boldsymbol{x}_i$, so that $\mathcal{S}_i$ does not contain the origin.

Proposition 1:

(a) $\mathcal{CH}(\mathcal{X}) \subseteq \mathcal{CH}(\{\mathcal{X}\backslash\boldsymbol{x}_i, \hat{\boldsymbol{x}}_i\})$ *if all visible facets from* $\boldsymbol{x}_i$ *are visible from* $\hat{\boldsymbol{x}}_i$.

(b) *The convex hull of multiple sets containing* $\mathcal{CH}(\mathcal{X})$ *contains* $\mathcal{CH}(\mathcal{X})$.

Proposition 1-a states, that the convex hull resulting from replacing a vertex $\boldsymbol{x}_i$ with a point $\hat{\boldsymbol{x}}_i$ will fully contain

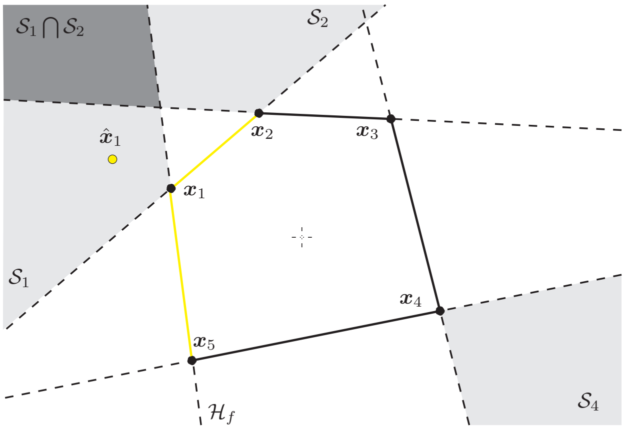

Fig. 2. *Visible Region:* The yellow facets denote the visible region from the point $\hat{\boldsymbol{x}}_1$ on $\mathcal{CH}(\mathcal{X})$. According to Proposition 1, point $\boldsymbol{x}_1$ can safely be substituted by $\hat{\boldsymbol{x}}_1$. Points $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ can simultaneously replaced by a point lying in the intersection of search regions $\mathcal{S}_1$ and $\mathcal{S}_2$.

$\mathcal{CH}(\mathcal{X})$, if the visible region of $\boldsymbol{x}_i$ on $\mathcal{CH}(\mathcal{X})$ is seen by $\hat{\boldsymbol{x}}_i$ as well. This is the case for any $\hat{\boldsymbol{x}}_i \in \mathcal{S}_i$ [17]. Note that it is possible for $\hat{\boldsymbol{x}}_i$ to see more facets of $\mathcal{CH}(\mathcal{X})$ than $\boldsymbol{x}_i$. Proposition 1-b is a direct consequence of convexity. According to the above Proposition, point $\hat{\boldsymbol{x}}_1$ in Fig. 2 can safely substitute $\boldsymbol{x}_1$ while preserving $\mathcal{CH}(\mathcal{X})$. One point in the intersection $\mathcal{S}_1 \bigcap \mathcal{S}_2$ is sufficient to replace $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ simultaneously.

In this light, the requirements for a point on a target objects surface to be included in one of the independent regions are illustrated in Fig. 3. Shown are the convex hulls of a three-fingered frictional grasp and a respective task in a hypothetical two-dimensional wrench space. In order for a candidate point $\boldsymbol{p}_s$ to qualify as a member of $\mathcal{C}_n$, the TWS has to be fully contained in the GWS resulting from replacing the primitive wrenches $\boldsymbol{w}_v(\boldsymbol{p}_n)$ with the primitive wrenches corresponding to $\boldsymbol{p}_s$. For example, the condition for a point $\boldsymbol{p}_s$ to be included in the independent region $\mathcal{C}_1$ is that there have to exist possible convex combinations of the primitive wrenches $\boldsymbol{w}_l(\boldsymbol{p}_s)$ inside both search regions $\mathcal{S}_{\varrho_{1,1}}$ and $\mathcal{S}_{\varrho_{1,2}}$. If this condition is satisfied, $\boldsymbol{p}_s$ can replace the original grasping point $\boldsymbol{p}_1$ according to Proposition 1. The search region $\mathcal{S}_{\varrho_{1,1}}$ is built by the intersection of the half-spaces defined by hyperplanes parallel to facets containing $\boldsymbol{w}_1(\boldsymbol{p}_1)$ and tangent to the TWS, so that $\mathcal{S}_{\varrho_{1,1}}$ does not contain the origin ($\mathcal{S}_{\varrho_{1,2}}$ is defined accordingly). Note that Proposition 1 is also satisfied if there exists a convex combination of the primitive wrenches $\boldsymbol{w}_l(\boldsymbol{p}_s)$ in the intersection $\mathcal{S}_{\varrho_{1,1}} \bigcap \mathcal{S}_{\varrho_{1,2}}$. The general definition of search spaces $\mathcal{S}_{\varrho_{n,v}}$ is as follows:

$$\mathcal{S}_{\varrho_{n,v}} = \{\boldsymbol{w} \in \mathbb{R}^K \;:\; \boldsymbol{A}_{\varrho_{n,v}} \boldsymbol{w} + \boldsymbol{\epsilon}_{\varrho_{n,v}} \leq \boldsymbol{0}\,,\, v \in \mathcal{V}_n\}. \quad (8)$$

$\boldsymbol{A}_{\varrho_{n,v}}$ above denotes the $\varrho_{n,v}$ rows of $\boldsymbol{A}$, likewise for $\boldsymbol{\epsilon}_{\varrho_{n,v}}$. Let $\boldsymbol{W}_s$ be the matrix corresponding to $\mathcal{W}_s$. A contact point $\boldsymbol{p}_s$ qualifies as a member of the independent contact region $\mathcal{C}_n$, if there exist convex combinations of primitive wrenches $\boldsymbol{W}_s$ inside each region $\mathcal{S}_{\varrho_{n,v}}$, or formally:

$$\begin{aligned}\mathcal{C}_n = \{\boldsymbol{p}_s \;:\; \exists \boldsymbol{\alpha}_v \in \mathbb{R}^L \; s.t.\; (\boldsymbol{W}_s \boldsymbol{\alpha}_v) \in \mathcal{S}_{\varrho_{n,v}} \\ v \in \mathcal{V}_n, \quad \boldsymbol{\alpha}_v \geq \boldsymbol{0}, \quad ||\boldsymbol{\alpha}_v||_{L_1} = 1\}. \end{aligned} \quad (9)$$

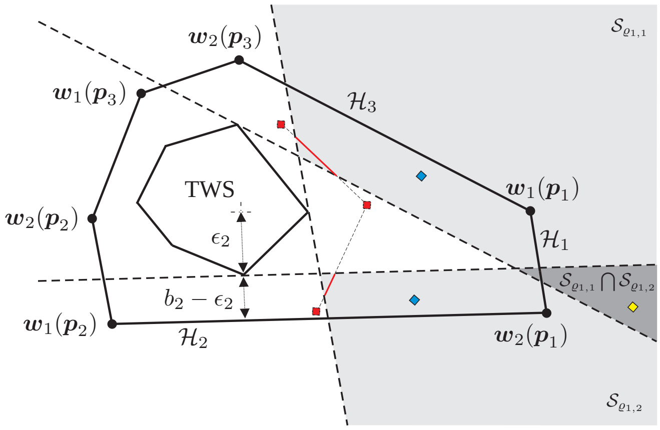

Fig. 3. *Search regions for* $\mathcal{C}_1$: Abstract 2-dimensional GWS, showing valid locations for primitive wrenches, so that the associated point in task space $\in \mathcal{C}_1$. A friction cone discretization of $L = 2$ is assumed. The red lines denote valid convex combinations of the primitive wrenches, which are shown as red squares. Contact points associated with the primitive wrenches depicted as blue squares, as well as the primitive wrench illustrated as a yellow square also can replace $\boldsymbol{p}_1$ without violating the TWS. Note that all primitive wrenches (with the possible exception of those stemming from torsional moments in case of soft finger contact) lie on the boundary of the OWS, which cannot be adequately represented in two dimensions. Furthermore, the primitive wrenches $\boldsymbol{w}_v(\boldsymbol{p}_n)$ corresponding to contact points $\boldsymbol{p}_n$ are connected by ridges of the convex hull, which are depicted in the same line style as facets.

A. Previous approaches

Here, we want to provide a brief discussion of the suggestions presented by Pollard [12] and Roa and Suárez [13] and compare it to our approach. Pollard provides the idea of spanning search spaces belonging to primitive wrenches of the GWS. However, no algorithm for the computation of independent regions is provided, and search spaces corresponding to *every* wrench $\boldsymbol{w}_l(\boldsymbol{p}_n)$ are defined. Compared to the search regions formulated in Eq. (8), this is disadvantageous from a computational point of view, because not necessarily every $\boldsymbol{w}_l(\boldsymbol{p}_n)$ is a vertex of the GWS since some may lie on the boundary or, in case of the soft finger contact model, inside the GWS. Hence, the approach in [12] can produce more search regions than necessary, which have to be evaluated.

Roa and Suárez [13] simplify the search problem, by exclusively checking primitive wrenches $\boldsymbol{w}_l(\boldsymbol{p}_s)$ for the inclusion in the respective search regions, instead of their convex combination. This might lead to the exclusion of some viable contact points, but is computationally more efficient. However, they define only one search region associated with each $\boldsymbol{p}_n$ as the following intersection of half-spaces: $\bigcap_{v \in \mathcal{V}_n} \mathcal{S}_{\varrho_{n,v}}$ (i.e. $\mathcal{S}_{\varrho_{1,1}} \bigcap \mathcal{S}_{\varrho_{1,2}}$ in Fig. 3). In this formulation a point $\boldsymbol{p}_s$ qualifies as a member of $\mathcal{C}_n$, if at least one of its primitive wrenches $\boldsymbol{w}_l(\boldsymbol{p}_s)$ lies in this intersection. This makes no difference in the frictionless case. However, increasing the friction coefficient μ causes this intersection to "move away" from the OWS and can result in smaller or even empty contact regions $\mathcal{C}_n$. To illustrate the influence of the choice of search regions, an example of a four-fingered grasp on a discretized ellipse is shown in Fig. 4 (the example is

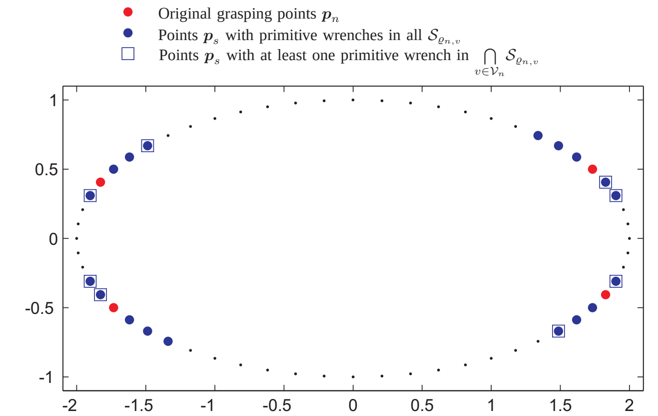

Fig. 4. *ICR's for a planar grasp on an ellipse:* The ellipse is discretized with 60 points. Each contact friction cone is approximated by two edges, the friction coefficient $\mu = 0.2$.

adapted from [13], Section IV-A). In fact, choosing a friction coefficient of $\mu \geq 0.27$ is causing empty regions $\mathcal{C}_n$ for the given example if search regions as $\bigcap_{v\in\mathcal{V}_n} \mathcal{S}_{\varrho_{n,v}}$ are utilized.

B. Computation algorithm

Here we give an efficient algorithm for the computation of ICR based on equation (9). Note that the sequence in which candidate points are evaluated does not matter, since the regions $\mathcal{C}_n$ are computed independently. The general structure is presented in Algorithm 1, while two options for the inclusion test in search regions $\mathcal{S}_{\varrho_{n,v}}$ are presented in Algorithms 2 and 3. Algorithm 1 starts by evaluating the GWS and $\varrho_{n,v}$ from equation (7). Since we defined $\varrho_{n,v}$ only with respect to vertex points, it can be formed simply by using the indices of points that comprise the facets of the convex hull. In case of a non-trivial TWS, in line 3, ϵ_h is computed by forming the dot products of all disturbance wrenches with $\boldsymbol{n}_h$ and setting ϵ_h equal to the largest one. If the TWS is a sphere, the distances $\epsilon_h(h = 1, \ldots, H)$ are set to be equal to the radius of the sphere. Starting with the grasping points $\boldsymbol{p}_n$ of the original grasp, neighboring points are evaluated according to equation (9). In order to keep track of already explored points, we define the N sets $\mathcal{E}_n$:

$$\mathcal{E}_n^{(s)} = \begin{cases} 1 & \text{if } \boldsymbol{p}_s \text{ has been explored for inclusion in } \mathcal{C}_n \\ 0 & \text{if } \boldsymbol{p}_s \text{ has not been explored for inclusion in } \mathcal{C}_n \end{cases}$$

Note that in the inclusion test given in Algorithm 3, in step 4, we do not need to carry out the whole matrix vector product, since if the product of one row of $\boldsymbol{A}_{\varrho_{n,v}}$ and $\boldsymbol{w}_l(\boldsymbol{p}_g)$ turns out to be positive, the rest of the computation can be truncated for the current iterate l.

IV. Numerically Evaluated Results

The Algorithm was implemented in Matlab and tested on a PC comprising a Core 2 Duo 2.9-GHz processor. As a test object, the model of a cup in Fig 1 was used. It is sampled with a number of $S = 2911$ vertices, which are meshed by 5822 triangles. The "GNU Linear Programming Kit" [18] was used to solve the linear program in Algorithm 2, convex hulls were computed using the "Qhull"-package [19].

Algorithm 1: *ICR computation*

1 Compute the GWS using equation (5)
2 Generate the $\mathcal{H}$-representation of the GWS
3 Define $\varrho_{n,v}$, $\forall v \in \mathcal{V}_n$
4 Determine ϵ_h, $h = 1, \ldots, H$
5 **for** $n \leftarrow 1$ *to* N **do** /* *i.e.* for each contact point $\in \mathcal{G}$ */
6 Initialize: $\mathcal{E}_n^{(s)} \leftarrow 0$ for $s = 1, \ldots, S$, set $i \leftarrow 1$, $j \leftarrow 1$
7 $\mathcal{C}_n^{(i)} \leftarrow \boldsymbol{p}_n$ (include $\boldsymbol{p}_n$ in $\mathcal{C}_n$)
8 **while** $i \leq j$ **do**
9 **for** *all neighbors of* $\mathcal{C}_n^{(i)}$ **do**
10 $g \leftarrow$ index of a neighbor of $\mathcal{C}_n^{(i)}$
11 **if** $\boldsymbol{p}_g$ *is not explored (i.e.* $\mathcal{E}_n^{(g)} = 0$*)* **then**
12 $\mathcal{E}_n^{(g)} \leftarrow 1$ /* set the current point as explored */
13 **if** *InclusionTest* $(\boldsymbol{p}_g)$ **then**
14 $j \leftarrow j + 1$
15 $\mathcal{C}_n^{(j)} \leftarrow \boldsymbol{p}_g$
16 $i \leftarrow i + 1$

Algorithm 2: *InclusionTest* with a linear program

1 **for** *all search regions* $\mathcal{S}_{\varrho_{n,l}}$ *associated with* $\boldsymbol{p}_n$ **do**
2 Solve the following linear program:

$$\begin{aligned} \underset{\boldsymbol{\alpha}_g\in\mathbb{R}^L,\, z\in\mathbb{R}}{\text{minimize}} \quad & z \\ \text{subject to} \quad & \boldsymbol{A}_{\varrho_{n,v}} \boldsymbol{W}_g \boldsymbol{\alpha}_g + \boldsymbol{\epsilon}_{\varrho_{n,v}} \leq z[1, \ldots, 1]^T \\ & ||\boldsymbol{\alpha}_g||_{L_1} = 1, \quad \boldsymbol{\alpha}_g \geq \boldsymbol{0} \end{aligned}$$

3 **if** $z > 0$ **then**
4 **return** false /* test for inclusion in $\mathcal{C}_n$ has failed */
5 **return** true /* test for inclusion in $\mathcal{C}_n$ has succeeded */

Algorithm 3: *InclusionTest* with primitive wrenches only

1 **for** *all search regions* $\mathcal{S}_{\varrho_{n,v}}$ *associated with* $\boldsymbol{p}_n$ **do**
2 set $l \leftarrow 1$, $f \leftarrow 1$
3 **while** $l \leq L$ **and** $f = 1$ **do**
4 $\boldsymbol{r} \leftarrow \boldsymbol{A}_{\varrho_{n,v}} \boldsymbol{w}_l(\boldsymbol{p}_g) + \boldsymbol{\epsilon}_{\varrho_{n,v}}$
5 **if** $\max(\boldsymbol{r}) \leq 0$ **then**
6 $f \leftarrow 0$
7 $l \leftarrow l + 1$
8 **if** $f = 1$ **then**
9 **return** false /* test for inclusion in $\mathcal{C}_n$ has failed */
10 **return** true /* test for inclusion in $\mathcal{C}_n$ has succeeded */

A. Benchmark

The benchmark was conducted by generating random 4-fingered frictional force closure grasps while varying the friction cone discretization $L \in \{6, 8, 10\}$ and the friction coefficient $\mu \in \{0.2, 0.5, 0.8\}$. As a TWS, the largest insphere of the GWS, scaled by a factor $\alpha = 0.75$ was used. We compared the performance of the inclusion tests utilizing the linear programming approach (LPA) in Algorithm 2 and the primitive wrench approach (PWA) in Algorithm 3, respectively. The results are summarized in Table I. For low friction coefficients μ, there is not much difference regarding the average number of total ICR-points. However, with increasing friction coefficient the LPA is able to detect significantly more points. Furthermore, the PWA is more sensitive to the chosen friction cone discretization L. The average computation times for the PWA are substantially

TABLE I

COMPARISON BETWEEN LP AND PW-APPROACH FOR 1000 RANDOMLY GENERATED 4-FINGER FORCE CLOSURE GRASPS

	LP	PW
	$L = 6$	
$\mu = 0.2$	$\overline{ICR}_{LP} = 18.74$	$\overline{ICR}_{PW} = 15.35$
	$\overline{t}_{LP} = 0.94s$	$\overline{t}_{PW} = 0.08s$
	$\max(t_{LP}) = 4.02s$	$\max(t_{PW}) = 0.12s$
$\mu = 0.5$	$\overline{ICR}_{LP} = 36.04$	$\overline{ICR}_{PW} = 26.03$
	$\overline{t}_{LP} = 1.47s$	$\overline{t}_{PW} = 0.08s$
	$\max(t_{LP}) = 4.77s$	$\max(t_{PW}) = 0.14s$
$\mu = 0.8$	$\overline{ICR}_{LP} = 47.78$	$\overline{ICR}_{PW} = 31.76$
	$\overline{t}_{LP} = 1.77s$	$\overline{t}_{PW} = 0.09$
	$\max(t_{LP}) = 5.86s$	$\max(t_{PW}) = 0.19s$
	$L = 8$	
$\mu = 0.2$	$\overline{ICR}_{LP} = 19.14$	$\overline{ICR}_{PW} = 16.53$
	$\overline{t}_{LP} = 1.61s$	$\overline{t}_{PW} = 0.15s$
	$\max(t_{LP}) = 6.48s$	$\max(t_{PW}) = 0.22s$
$\mu = 0.5$	$\overline{ICR}_{LP} = 40.52$	$\overline{ICR}_{PW} = 31.96$
	$\overline{t}_{LP} = 2.75s$	$\overline{t}_{PW} = 0.16s$
	$\max(t_{LP}) = 10.96s$	$\max(t_{PW}) = 0.27s$
$\mu = 0.8$	$\overline{ICR}_{LP} = 52.43$	$\overline{ICR}_{PW} = 39.74$
	$\overline{t}_{LP} = 3.68s$	$\overline{t}_{PW} = 0.17s$
	$\max(t_{LP}) = 12.10s$	$\max(t_{PW}) = 0.26s$
	$L = 10$	
$\mu = 0.2$	$\overline{ICR}_{LP} = 19.59$	$\overline{ICR}_{PW} = 17.49$
	$\overline{t}_{LP} = 2.36s$	$\overline{t}_{PW} = 0.25s$
	$\max(t_{LP}) = 8.89s$	$\max(t_{PW}) = 0.47s$
$\mu = 0.5$	$\overline{ICR}_{LP} = 41.61$	$\overline{ICR}_{PW} = 34.69$
	$\overline{t}_{LP} = 4.96s$	$\overline{t}_{PW} = 0.27s$
	$\max(t_{LP}) = 16.53s$	$\max(t_{PW}) = 0.43s$
$\mu = 0.8$	$\overline{ICR}_{LP} = 54.80$	$\overline{ICR}_{PW} = 44.00$
	$\overline{t}_{LP} = 7.15s$	$\overline{t}_{PW} = 0.29s$
	$\max(t_{LP}) = 22.48s$	$\max(t_{PW}) = 0.44s$

$\overline{ICR}_{LP/PW}$ – average number of overall ICR-points

$\overline{t}_{LP/PW}$ – average computation time

$\max(t_{LP/PW})$ – maximum computation time

lower than for the LPA. For the latter, especially the high maximal computation times are significant. It is evident, that choosing between LPA and PWA is a trade-off between accuracy and computational effort. However, utilizing the PWA with a high discretization L gives a good compromise.

B. Example

We give an example of ICR computation for a 4-fingered frictional grasp of the cup in Fig. 1. The TWS was modeled as the OWS_D with scaled force domain in order to protect against gravity as well (see Section II-E). In order to determine ICR, the PWA with following parameters was utilized: $\mu = 0.8$, $L = 8$, $F_G = 10$, $F_D = 0.6$ and $F_{grav} = 1.5$. A total number of 52 ICR-points were found. The computation time evaluated to $0.41s$. This shows, that even for non-trivial task wrench spaces, independent regions can be efficiently computed with the proposed algorithm.

V. CONCLUSION

In this work, an efficient and parallelizable algorithm for the computation of independent regions on discretized 3-D objects is presented. The suggested method allows the incorporation of disturbance wrench sets, corresponding to a given task. Following contact models can be applied: frictionless point contact, point contact with friction and soft-finger contact. Furthermore, a geometrical analysis of search regions in wrench space, suitable for the computation of independent regions, is provided. The computational efficiency of the approach is shown by means of an example including non-trivial disturbances.

ACKNOWLEDGMENTS

This research has been partially supported by the HANDLE project, funded by the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement ICT 231640.

REFERENCES

[1] A. Bicchi, "On the closure properties of robotic grasping," *The Int. J. of Robotics Research*, vol. 14, no. 4, pp. 319–334, August 1995.

[2] B. Mishra, J. T. Schwartz, and M. Sharir, "On the existence and synthesis of multifinger positive grips," *Algorithmica*, vol. 2, pp. 541–558, 1987.

[3] R. Suárez, M. A. Roa, and J. Cornellà, "Grasp quality measures," Institute of Industrial and Control Engineering, Technical University of Catalonia, Tech. Rep., 2006.

[4] D. G. Kirkpatrick, B. Mishra, and C. K. Yap, "Quantitative steinitz's theorems with applications to multifingered grasping," in *STOC '90: Proceedings of the twenty-second annual ACM symposium on Theory of computing*, 1990, pp. 341–351.

[5] C. Ferrari and J. Canny, "Planning optimal grasps," in *Robotics and Automation, 1992. Proceedings., 1992 IEEE International Conference on*, May 1992, pp. 2290 –2295 vol.3.

[6] N. S. Pollard, "Parallel methods for synthesizing whole-hand grasps from generalized prototypes," Ph.D. dissertation, MIT, Dept. of Electrical Engineering and Computer Science, 1994.

[7] M. Strandberg, "A grasp evaluation procedure based on disturbance forces," in *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*, vol. 2, 2002, pp. 1699 – 1704.

[8] C. Borst, M. Fischer, and G. Hirzinger, "Grasp planning: how to choose a suitable task wrench space," in *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on*, vol. 1, 2004, pp. 319 – 325.

[9] V.-D. Nguyen, "Constructing force-closure grasps," in *Robotics and Automation. Proceedings. 1986 IEEE International Conference on*, vol. 3, Apr 1986, pp. 1368 – 1373.

[10] J. Ponce and B. Faverjon, "On computing three-finger force-closure grasps of polygonal objects," *Robotics and Automation, IEEE Transactions on*, vol. 11, no. 6, pp. 868 –881, dec 1995.

[11] J. Ponce, S. Sullivan, A. Sudsang, J.-D. Boissonnat, and J.-P. Merlet, "On computing four-finger equilibrium and force-closure grasps of polyhedral objects," *International Journal of Robotics Research*, vol. 16, no. 1, pp. 11–35, 1997.

[12] N. S. Pollard, "Closure and quality equivalence for efficient synthesis of grasps from examples," *International Journal of Robotics Research*, vol. 23, no. 6, pp. 595–614, 2004.

[13] M. A. Roa and R. Suárez, "Computation of independent contact regions for grasping 3-d objects," *IEEE Transactions on Robotics*, vol. 25, pp. 839–850, 2009.

[14] X. Zhu and J. Wang, "Synthesis of force-closure grasps on 3-d objects based on the q distance," *Robotics and Automation, IEEE Transactions on*, vol. 19, no. 4, pp. 669 – 679, Aug. 2003.

[15] B.-H. Kim, B.-J. Yi, S.-R. Oh, and I. H. Suh, "Non-dimensionalized performance indices based optimal grasping for multi-fingered hands," *Mechatronics*, vol. 14, no. 3, pp. 255 – 280, 2004.

[16] R. M. Murray, Z. Li, and S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation*, 1st ed. CRC, 1994.

[17] M. De Berg, O. Cheong, and M. van Kreveld, *Computational geometry: algorithms and applications*. Springer, 2008.

[18] "GNU Linear Programming Kit Version 4.42," http://www.gnu.org/software/glpk/.

[19] C. B. Barber, D. P. Dobkin, and H. Huhdanpaa, "The quickhull algorithm for convex hulls," *ACM Transactions on Mathematical Software*, vol. 22, no. 4, pp. 469–483, 1996.