Determining Force-Closure Grasps Reachable by a Given Hand ⋆

Fidel Gilart and Raúl Suárez

Institut d'Organització i Control de Sistemes Industrials (IOC)
Universitat Politècnica de Catalunya (UPC), Barcelona, Spain
e-mails: fidel.gilart@upc.edu, raul.suarez@upc.edu

Abstract: The paper presents an approach to find contact points on an object surface that are reachable by a given hand and such that the resulting grasp satisfies the force-closure condition. This is a very common problem that still requires a practical solution. The proposed method is based on the computation of a set of independent contact regions on the object boundary such that a finger contact on each region produce a force-closure grasp, and then this set of regions is iteratively recomputed while looking for a set of contact points that are reachable by a given hand. The search is done guided by a cost function that indicates the proximity of the hand fingertips to a candidate set of grasping contact points. The approach has been implemented for the Schunk Anthropomorphic Hand and planar objects, and application examples are included to illustrate its performance.

Keywords: Grasping, Multifingered Hands, Dexterous Manipulation.

1. INTRODUCTION

Object grasping and manipulation by multi-finger hands has received considerable attention in the last years. It has become an important area of great interest in robotics mainly because it increases the flexibility and versatility of the robotic arms, allowing the use of a single end effector for grasping and complex manipulation of a large variety of objects.

In order to perform suitable manipulation tasks with a given hand, two main conditions must be satisfied to accomplish a valid grasp. First, the grasping contact points must allow the application of forces on the object that achieve the object equilibrium, or to fully restrain the object to resist external disturbances. This is accomplished by satisfying form or force-closure conditions (Bicchi, 1995). Second, it must be ensured that the contact points on the object are reachable by the fingers.

The first condition has been widely used in the synthesis of precision grasps (i.e. when only the fingertips touch the object) for 2D (Liu, 2000; Niparnan and Sudsang, 2006; Cornellà and Suárez, 2009) and 3D objects (Ponce et al., 1997; Zhu and Wang, 2003; Roa and Suárez, 2009b). On another hand, in order to provide robustness to the grasp in front of finger positioning errors, the concept of set of independent contact regions (*ICRS*) on the object boundary was introduced (Nguyen, 1988). The positioning of a finger in each independent contact region ICR_i assures a force-closure (FC) grasp, independently of the exact position of each finger. The computation of sets *ICRS* has been solved for 2D polygonal (Cornellà and Suárez, 2005b) and 3D polyhedral objects (Ponce et al., 1997; Ponce and Faverjon, 1995), as well as for objects of arbitrary shape described by a mesh with large number of points, for 2D (Cornellà and Suárez, 2005a) and 3D (Pollard, 2004; Roa and Suárez, 2009b) discrete objects, and with frictional and frictionless contacts.

⋆ This work was partially supported by the Spanish Government through the projects DPI2010-15446 and DPI2011-22471. F. Gilart is supported by Cooperation Spanish Agency for International Development (AECID).

The second condition is verified by solving the inverse kinematics of the robot hand. This problem has been tackled with different local convergence methods (Borst et al., 2002; Rosell et al., 2005), and more recently (Claret and Suárez, 2011; Rosales et al., 2008). The method presented in (Rosales et al., 2008) is able to find all possible configurations that reach specified contact points, in contrast to local methods that only provide one solution to the problem, even if many are possible.

Solving the contact points on the object satisfying FC conditions and at the same time satisfying that the contact points are reachable by a specific hand is still an open problem. A previous work tackling this problem presents a method to seek feasible grasps on an object for a given hand (Zheng and Qian, 2008). The method is valid for convex objects, then for non-convex objects it is necessary a decomposition into convex parts that is done manually. Another drawback of the approach is that it depends on the position and orientation of the wrist, that must be given in each execution of the algorithm. Lippiello et al. (2010) present a method for visual grasp of unknown objects that includes an object surface reconstruction and a local grasp planner. The desired final grasp is characterized by having all the contact points lying on the same grasping plane in an equilateral configuration and the force-closure condition is not always assured. Another recent work presents an approach to obtain reachable independent contact regions taking into account the hand kinematics (Roa et al., 2011); a solution is found relatively fast, but it has the requirement that the initial pose of the hand with respect to the object must be given manually using a haptic device.

The approach presented in this work try to solve this problem with a fully automatic procedure by iteratively computing a new *ICRS* that, eventually, includes regions ICR_i are reachable by the fingers of a given hand. This is done by displacing the regions ICR_i in a certain direction on the surface of the object. This direction is determined using as criterion the minimization of the distance between the fingertips and the regions ICR_i on the object. This criterion is formally defined later in this work as a reachability cost function. Since there exist procedures to efficiently compute *ICRS*, the approach presented here uses them to search sets of FC grasps for reachable grasp. Considering the possibility of doing it in the reverse way, the authors are not aware of any procedure that allows an efficient determination of sets of reachable grasps that can be then searched for FC grasps. A typical approach is the search of a FC grasp using heuristics to determinate the wrist position and orientation, then closing the hand (in simulation) to obtain a reachable grasp and then evaluating whether it is FC or not (Miller and Allen, 2004); one drawback of this approach is that there is not any control on the resulting contact points. It must be remarked that in the proposed approach it is not necessary to provide the initial wrist position and orientation, being the search systematically done using only the geometric model of the object and the kinematic model of the hand.

The rest of the paper is organized as follows. Section II provides basic concepts and background on FC grasps, grasp space and *ICRS*. Section III describes the approach proposed and presents the algorithms to find reachable grasp for a given hand. Section IV shows four examples to illustrate the approach, and Section V summarizes the work and presents some future research.

2. BASIC CONCEPTS AND BACKGROUND

2.1 Object Model

The approach is intended to be valid for rigid objects of any shape. It is assumed that: the object surface is discretized with a large enough set of points represented by N position vectors $\mathbf{p}_i$ measured with respect to a reference system located at the center of mass of the object; the normal direction $\hat{\mathbf{n}}_i$ pointing toward the interior of the object at $\mathbf{p}_i$ is known; and each point of the object surface defined by $\mathbf{p}_i$ is considered as a potential contact point.

2.2 Contact and Force Model

The contacts between the fingertips and the object are assumed to be punctual and frictional. Friction is modeled according to Coulomb's law, which states that in order to avoid slippage the force $\mathbf{f}_i$ applied at $\mathbf{p}_i$ must lie inside the friction cone defined by $\mathbf{f}_i^t \leq \mu \mathbf{f}_i^n$, where μ is the friction coefficient and $\mathbf{f}_i^t$ and $\mathbf{f}_i^n$ are, respectively, the modules of the tangential and normal components of $\mathbf{f}_i$.

The force $\mathbf{f}_i$ applied on the object at $\mathbf{p}_i$ generates a torque $\boldsymbol{\tau}_i = \mathbf{p}_i \times \mathbf{f}_i$ with respect to the center of mass of the object. $\mathbf{f}_i$ and $\boldsymbol{\tau}_i$ are grouped together in a wrench vector given by $\mathbf{w}_i = (\mathbf{f}_i^T \; \boldsymbol{\tau}_i^T)^T$.

The wrenches produced by forces on the two boundaries of the friction cone for 2D objects or by m forces on the boundary of the friction cone for 3D objects (such that the m forces define a polyhedral convex cone that approximates the real friction cone) are called primitive wrenches.

2.3 Force-Closure

An important property of a grasp is the ability to balance external wrenches by applying appropriate finger wrenches at the contact points. A grasp that can resist wrenches with any direction is called force-closure (FC) grasp. A necessary and sufficient condition for the existence of a FC grasp is that the origin O of the wrench space lies strictly inside the convex hull of the primitive wrenches (Murray et al., 1994). Several FC tests based on this condition have been proposed, for instance, solving linear optimization problems (Zhu and Wang, 2003), or using collision checks (Zhu et al., 2004). Another approach is based on linear matrix inequalities that efficiently deal with frictional constraints, thus avoiding the linearization of the friction cone (Han et al., 2000).

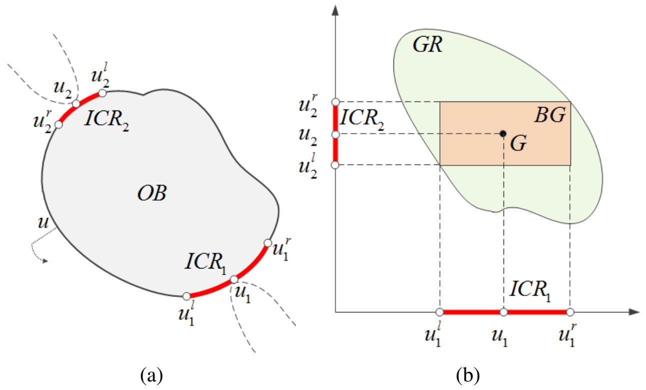


Fig. 1. Example of *ICRS* and associate box *BG* for an arbitrary object *OB*: a) *ICRS* represented in physical space, b) the corresponding box *BG* represented in the grasp space *GS*.

2.4 Grasp Space

An n-finger grasp G is described by the set of parameters u_i that determine the contact positions $\mathbf{p}_i$ of the fingers on the grasped object, i.e. $G = \{u_1, \ldots, u_\alpha\}$, with $\alpha = n$ for 2D objects and $\alpha = 2n$ for 3D objects. The α-dimensional space representing the positions of the possible contact points defined by $u_1, ..., u_\alpha$ is called grasp space *GS* (also known as grasp configuration space or contact space (Ponce and Faverjon, 1995)). In this work $\alpha = n$ because 2D objects are considered and therefore G is defined indistinctly as $G = \{\mathbf{p}_1, \ldots, \mathbf{p}_n\}$ or $G = \{u_1, \ldots, u_n\}$.

Given an object *OB* and a hand *H*, the grasp space *GS* for these two elements contains the following regions of interest:

- Graspable Region, *GR*, that includes all the points of *GS* that represent FC grasps. *GR* may be composed of I different subregions GR_i, i.e $GR = \{GR_1, \ldots, GR_I\}$.
- Reachable Region, *RR*, that includes all the points of *GS* that represent grasps that are reachable by the hand *H*. *RR* may be composed of J different subregions RR_i, i.e $RR = \{RR_1, \ldots, RR_J\}$.

both I and J may be unknown a priori.

2.5 Independent Contact Regions

In a real world application, the actual and the planned grasp may differ due to finger positioning errors. In order to provide robustness to the grasp despite these errors, Nguyen (1988) introduced the concept of independent contact regions $ICRS = \{ICR_1, ..., ICR_n\}$, where each ICR_i is a regions on the object surface such that a finger contact on each ICR_i assures a FC grasp with independence of the exact position of the fingers. In this work we compute the sets *ICRS* using the procedure proposed by Roa and Suárez (2009a). The representation of all the possible grasps allowed by a given set *ICRS* is an axis-aligned box region *BG* in *GS*. The projection of *BG* onto the

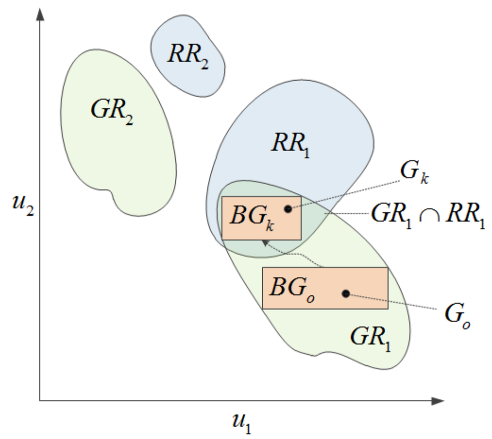


Fig. 2. Hypothetical 2-dimensional grasp space GS with graspable region $GR = \{GR_1, GR_2\}$ and reachable region $RR = \{RR_1, RR_2\}$. Starting from a grasp $G_o \notin GR_1 \cap RR_1$ and cost $R(G_o) \neq 0$, the procedure would find a grasp $G_k \in GR_1 \cap RR_1$, i.e. with cost $R(G_k) = 0$.

axis i of GS correspond to ICR_i. Each box BG can be stored with 2α parameters, representing lower and upper limits along each axis of GS.

An $ICRS$ and the corresponding box BG are conceptually illustrated in Fig. 1 using an arbitrary 2D object OB grasped with 2 fingers (fingers represented as dotted lines) producing a FC grasp G. A parameter u determines the contact positions on the object surface. Fig. 1a shows the finger positions defining $G = \{u_1, u_2\}$. Hypothetical ICR_1 and ICR_2, obtained from G, are represented respectively with regions defined by left and right limits $[u_1^l, u_1^r]$ and $[u_2^l, u_2^r]$. Fig. 1b shows the hypothetical 2-dimensional grasp space GS corresponding to OB with the representation of: G, a graspable region GR, and the box BG defined by the limits of ICR_1 and ICR_2, satisfying $BG \subset GR$.

3. PROPOSED APPROACH

3.1 Approach Overview

The problem tackled in this work is formulated as: given a specific object OB with associated grasp space GS, and a specific hand H with n fingers, the objective is to find a set of contact points on the object surface $\{p_1, \ldots, p_n\}$ that allows a FC grasp G (i.e. $G \in GR$) and that at the same time G were reachable by H (i.e. $G \in RR$); then, the problem to be solved is summarized as finding a grasp $G \in GR \cap RR$.

Finding $G \in GR \cap RR$ in a systematic way for any object and any hand is a complicated task that has a high computational cost, and therefore it may be not of practical application. The approach presented in this work deals with this problem by computing sets $ICRS$ that eventually are reachable by the given hand. This is done by recomputing the regions ICR_i displacing them on the surface of the object in a direction determined by a cost function R, which indicates the proximity of the hand fingertips to the contact points of an analyzed grasp G.

The approach is conceptually illustrated in Fig. 2 with a 2-dimensional grasp space GS corresponding to a 2-finger grasp $G = \{u_1, u_2\}$. The figure shows the graspable region $GR = \{GR_1, GR_2\}$ and the reachable region $RR = \{RR_1, RR_2\}$. The objective is to find $G \in GR \cap RR$, which in this case means $G \in GR_1 \cap RR_1$. Using already published procedures (Roa and Suárez, 2009b), we can find a starting grasp $G_o \in GR$; let's assume that $G_o \in GR_1$ and $G_o \notin RR$. Starting from G_o a set $ICRS_o$ is computed, which is represented in GS by a box BG_o (see Subsection 2.5); then, the cost function R is evaluated for some selected grasps $G_i \in BG_o$, if none of then is reachable with H (i.e. none of them produces $R = 0$) a new set $ICRS_1$ is computed starting from the selected G_i with minimum cost R. This procedure is iteratively repeated until a region $BG_k \cap GR_1 \cap RR_1 \neq \emptyset$ is found and a selected grasp G_k satisfies $G_k \in GR_1 \cap RR_1$. In the case that $G_o \in GR_2$, since $GR_2 \cap RR = \emptyset$ the procedure will return failure after completely exploring GR_2, and it is necessary to look for another starting grasp $G'_o \in GR_1$, which can be done using existing published procedures.

This paper is focused on the iterative algorithm to explore the regions GR_i, independently of whether a solution exists or not in the explored region; if a solution is not found in the explored region the procedure will automatically explore another one. The kinematic model of the given hand H is used in the computation of the cost function, as it is explained in next subsection.

3.2 Cost Function

Given a hand H with n fingers, each fingertip position in the physical space is represented by a vector $\mathbf{q}_i$. Then, the configuration of the hand H can be represented as the set $HC = \{\mathbf{q}_1, \ldots, \mathbf{q}_n\}$.

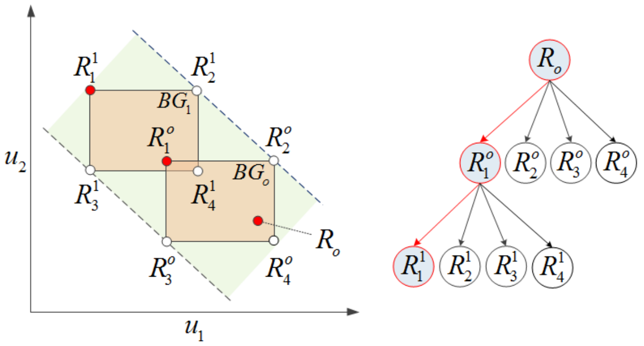
In order to verify whether a given grasp $G \in GR$ is reachable by the hand H, the inverse kinematic of H is solve trying to put the fingertips $\{\mathbf{q}_1, \ldots, \mathbf{q}_n\}$ at the contact points $\{\mathbf{p}_1, \ldots, \mathbf{p}_n\}$ defining G. If there is no solution to the inverse kinematic of the hand then $G \notin RR$ and the inverse kinematic algorithm returns the closest reachable hand configuration (Claret and Suárez, 2011). In this case, the cost function $R(G)$ associated with the returned configuration is given by the summation of the distances from each fingertip $\mathbf{q}_i$ to the corresponding contact point $\mathbf{p}_i$ on the object, which is formally expressed as

$$R(G) = \sum_{i=1}^{n} \|\mathbf{p}_i - \mathbf{q}_i\| \tag{1}$$

3.3 Main Algorithm

The main algorithm of the proposed approach performs the iterative computation of new $ICRS$ sweeping a GR and looks for a grasp that also belongs to RR.

The iterative computation of new $ICRS$ is done as a targeted search in a dynamic tree, as illustrated conceptually in Fig. 3. The tree starts with the initial grasp G_o with cost $R(G_o)$ and its associate FC region BG_o. The children of a node in the iteration k are γ selected grasps $G_{i,k} \in BG_k$, with $i = 1, \ldots, \gamma$. In this work we use as children the grasps defining the vertices of BG_k (the number of vertices for planar objects grasped with n fingers gives $\gamma = 2^n$). The nodes of the tree are ordered in a list L_r according to their cost $R(G_{i,k})$, and the expansion of new children is done starting from the first node in L_r, i.e. the node with smallest associated cost. Before a node is expanded a predefined number B of samples $G_{aux} \in BG_k$ are checked to verified whether they belong to RR and therefore G_{aux} is already a solution (note that by construction $G_{aux} \in GR$). In the



(a) (b)

Fig. 3. Iterative computation of *ICRS* according to the cost R. a) 2-dimensional grasp space with: an initial grasp with cost R_o, the box BG_o generated from it with the costs of the grasps at the corners $R_1^o, R_2^o, R_3^o, R_4^o$, and a box BG_1 generated from the vertex of BG_o with minimum cost; b) The tree from the starting FC grasp expanding the child with minimum cost, e.g. R_1^o in the first iteration.

implemented algorithm we select the B samples G_{aux} following a regular pattern in BG_k.

The algorithm is formalized as follows:

Algorithm Search of $G \in GR_i \cap RR$

(1) Find a starting FC grasp $G_o = \{\mathbf{p}_1, ..., \mathbf{p}_n\}$ (e.g. using the algorithm presented by Roa and Suárez (2009b)).
(2) Assign each fingertip to a contact point of G_o, i.e. associated fingertip points $\mathbf{q}_j$ with contact points $\mathbf{p}_i$.
(3) Compute $R(G_o)$.
(4) IF $R(G_o) = 0$ THEN RETURN(G_o)
ELSE create the list L_r with the element G_o
(5) WHILE L_r is not empty, DO
  (a) Select as G the first element of L_r.
  (b) Compute *ICRS* and the corresponding BG for G
  (c) FOR $i = 1$ to B DO
    Obtain a sample G_{aux} from BG
    Compute $R(G_{aux})$.
    IF $R(G_{aux}) = 0$ THEN RETURN(G_{aux})
  (d) FOR each vertex G_{vertex} of BG DO
    Compute $R(G_{vertex})$.
    IF $R(G_{vertex}) = 0$ THEN RETURN(G_{vertex})
    ELSE IF $G_{vertex} \notin L_r$ THEN add G_{vertex} to L_r
  (e) Remove G from L_r
  (f) Order L_r in ascendent order according to the cost R of each element
(6) RETURN(failure)

Note that in Step 2 there is as assignment of the fingertips to the contact points on the object. Then, for a grasp with n fingers there are $n!$ possible combinations, depending on the particular kinematic structure of the hand (that could have symmetries) all the possible combinations may produce different results, so if there is no solution for a combination the others should be tested. For simplicity, this evident loop was not included in the algorithm above.

The algorithm is computationally simple and although it has a heuristic nature, some remarks about the computational cost can be made. Step 5b computes the set *ICRS* which requires the computation of a convex hull in a λ-dimensional space, with $\lambda = 3$ for 2D objects and $\lambda = 6$ for 3D objects. This work uses the *qhull* implementation, which has a worst case complexity $O(NlogN)$ for $\lambda \leq 3$ and $O(N^{\lfloor \lambda/2 \rfloor} / \lfloor \lambda/2 \rfloor !)$ for $\lambda \geq 4$, which yields $O(N^3/6)$ for $\lambda = 6$ (Barber et al., 1996). Step 5c is quite relevant for the algorithm performance, since in the worst case it requires to solve B times the inverse kinematics of the hand to compute the reachability cost $R(G_{aux})$ for the B sampled grasps G_{aux} in the region BG. The algorithm is complete if all the grasps in BG are checked, but this is not done in practical applications where just a reduced set of samples provides a good (practical) idea of the current situation and allows a faster search of a solution; therefore there is always a tradeoff in the selection of B. Selection of an optimal value for B is an interesting issue for future work. It must be also remarked that the areas in GR_i not covered by a region BG and therefore not analyzed in one run of the algorithm are potential targets for a new starting FC grasp if the current run returns failure (i.e. they could be considered as new unexplored regions GR_i).

4. EXPERIMENTS

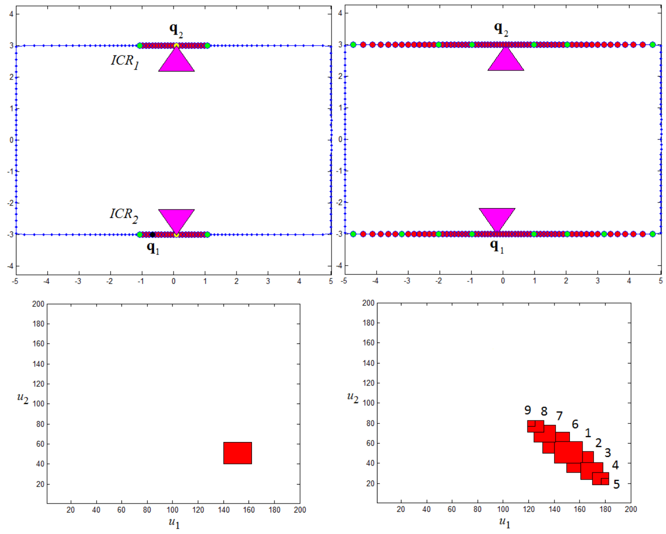
The proposed approach has been implemented on a Core2Duo 2 GHz PC for the Schunk Anthropomorphic Hand grasping planar objects. The performance of the algorithm is first illustrated with simulated basic examples (implemented in MATLAB) using only two fingers, allowing therefore a graphical representation of the grasp space *GS*, and then with real experiments (implemented in C++) using the four fingers of the hand, wich produce a 4-dimensional space *GS*.

4.1 Basic illustrative examples

Since the cost function R depends on the hand configuration returned by the inverse kinematic procedure (either being a solution or not), in order to illustrate the method we consider grasps with 2 fingers considering as kinematic constraint that the distance between the fingertips $\mathbf{q}_1$ (thumb) and $\mathbf{q}_2$ (index finger) has a maximum value L, i.e. $\|\mathbf{q}_1 - \mathbf{q}_2\| \leq L$.

Using this simple constraint on the fingertip positions, the inverse kinematics of the hand is solved by making $\mathbf{q}_1 = \mathbf{p}_1$ and positioning $\mathbf{q}_2$ such that $\|\mathbf{q}_2 - \mathbf{p}_2\|$ is minimized subject to the kinematic constraint. Then, the cost of a hand configuration associated to a grasp G is directly obtained as $R(G) = \|\mathbf{q}_2 - \mathbf{p}_2\|$. The objects to be grasped were discretized with 200 points.

Example 1: Rectangle grasped with 2 fingers. The object is a rectangle with the closest parallel faces at a distance of 6 units. First, the allowed maximum distance between the fingertips is $L = 6.1$. In this case a set of reachable points was found in the initial box BG in 2.1 s. The obtained results are shown in Fig. 4a. Top picture shows the discretized rectangle, the set *ICRS* (containing 2 regions ICR_i, one for each finger), the fingertip positions $\mathbf{q}_1$ and $\mathbf{q}_2$ for the initial FC grasp (that is no-reachable) and the corresponding friction cones for the final grasp (reachable). Bottom picture shows the representation of the box BG in the grasp space. Then, the allowed maximum distance between the fingertips was reduced to $L = 2$, so the rectangle cannot be grasped. The graspable region corresponding to one finger in each of the opposite faces was fully explored in 9 iterations (i.e. 9 sets *ICRS* were computed) in 14.1 s before returning "failure". The results are shown in Fig. 4b. Top picture shows the discretized rectangle,



(a) (b)

Fig. 4. Grasping a rectangle with 2 fingers and different maximum finger opening L: a) $L = 6.1$ is large enough and a solution is found; b) $L = 2$ is not large enough and no solution is found after exploring the whole graspable region.

the 9 sets *ICRS* (that partially overlap themselves and fully cover the rectangle edges), the fingertips $\mathbf{q}_1$ and $\mathbf{q}_2$ for the initial FC grasp (that is no-reachable) and the corresponding friction cones. Bottom picture shows the 9 boxes BG_i, $i = 1,\ldots,9$ representing the 9 sets *ICRS* in the grasp space following the order in which they were generated.

Example 2: Trapezium grasped with 2 fingers. The relative position of the non parallel edges of the trapezium makes that 2 fingers with maximum opening $L = 6.1$ can grasp the object only contacting these edges on the side they are closer. The solution was found in 9 iterations in 9.5 s. Fig. 5 (top) shows the trapezium, the multiple generated *ICRS* (that overlap themselves covering a portion of the edges), the fingertip positions $\mathbf{q}_1$ and $\mathbf{q}_2$ for the original non reachable grasp (#1) and for the final reachable one (#9), and the corresponding friction cones. Fig. 5 (bottom) shows the 9 boxes BG_i, $i = 1,\ldots,9$ representing the 9 sets *ICRS* in the grasp space following the order in which they were generated.

4.2 Real experiments

Real experiments were performed using the Schunk Anthropomorphic Hand (SAH). This robotic hand has four fingers, each finger has 4 joints and 3 independent degrees of freedom (the two outer joints are coupled), and the thumb base has an extra degree of freedom (Center, 2007).

The inverse kinematics of the hand is solved using the method presented by Claret and Suárez (2011). The method combines an iterative algorithm with an off-line analysis that allows significant reductions of the execution time. Given a grasp G, the inverse kinematics procedure returns a final configuration of the hand (either being a solution or not) that is used to compute the cost function $R(G)$.

The models of the objects to be grasped were discretized with 1000 points, the friction coefficient was $\mu = 0.08$, and $B = 20$ grasps were sampled in each region BG.

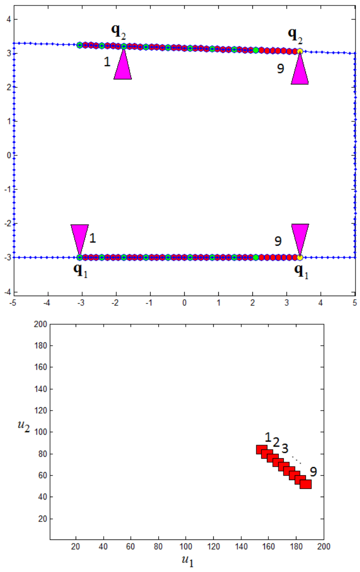


Fig. 5. Grasping a trapezium with 2 fingers and a maximum finger opening $L = 6.1$. Starting from the grasp #1 the final reachable grasp #9 was found in 9 iterations.

Example 3: Grasping a rectangle with the SAH. A valid solution was found in 467 s with 32 iterations. Simulation result is shown in Fig. 6a. The ICR_i obtained for the thumb is very small (only 2 points) and it is not appreciated. The real execution of this grasp is shown in Fig. 7, the hand was moved to the determined grasping configuration and then the fingertips were moved 1 mm inside the object along the surface normal direction at the respective nominal positions $\mathbf{q}_i$.

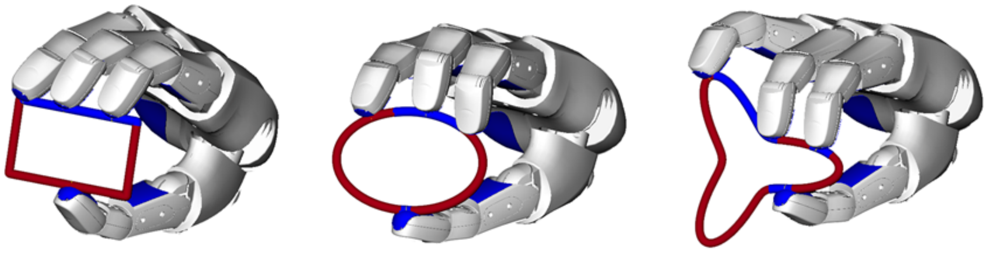
Example 4: Grasping an ellipse with the SAH. In this case a solution was found in 124 s with 56 iterations. Simulation result is shown in Fig. 6b. The ICR_i obtained for the index and ring fingers are very small (1 and 2 points respectively) and therefore they are not appreciated.

Example 5: Grasping a three-point star with the SAH. This example uses an object defined by a closed parametric curve (Jia, 2004). In this case a solution was found in 221 s with 65 iteration and the resulting grasp is shown in Fig. 6c. The ICR_i for ring finger is very small (2 points) an therefore it is not appreciated.

5. SUMMARY AND FUTURE WORKS

This paper presents an approach to find contact points on an object that allow a force-closure grasp and that at the same time are reachable by a specific hand. The algorithms has been implemented for the Schunk Anthropomorphic Hand and planar objects and some examples were included to illustrate how they work. Basically, the approach uses a cost function to perform the search of a reachable grasp G within a set GR of force-closure ones.

Topics for future research include looking for potential improvements of the exploration of the graspable regions when looking for a reachable grasp. Another topic is that, besides the kinematic constrains of the hand, the accessibility to the contact points must be considered. Finally, the approach is formally valid for 3D objects and the complete implementation for these objects is still to be developed.



(a) (b) (c)

Fig. 6. Simulation results using the SAH to grasp (ICR_i for each finger are in blue): a) a rectangle; b) a ellipse; c) a 3-point star.

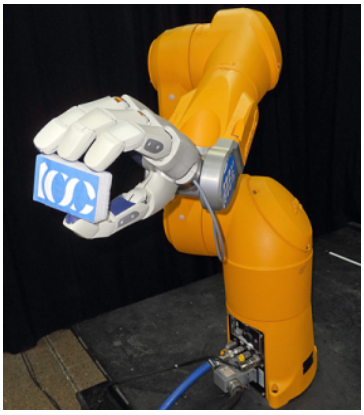


Fig. 7. Real execution of the example 3 shown in Fig. 6a.

REFERENCES

Barber, C., Dobkin, D., and Huhdanpaa, H. (1996). The quickhull algorithm for convex hulls. *ACM Trans. Math. Softw*, 22(4), 469–483.

Bicchi, A. (1995). On the closure properties of robotic grasping. *Int. J. Robotics Research*, 14(4), 319–334.

Borst, C., Fischer, M., and Hirzinger, G. (2002). Calculating hand configurations for precision and pinch grasps. In *Proc. IEEE Int. Conf. Intelligent Robots and Systems*, 1553–1559. Lausanne.

Center, D.G.A. (2007). *Schunk Anthropomorphic Hand User Manual*, 1.2 edition.

Claret, J. and Suárez, R. (2011). Efficient and practical determination of grasping configurations for anthropomorphic hands. In *18th IFAC World Congress*, 14660–14666. Milano, Italy.

Cornellà, J. and Suárez, R. (2005a). Determining independent grasp regions on 2d discrete objects. In *18th IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2936–2941.

Cornellà, J. and Suárez, R. (2005b). Fast and flexible determination of force-closure independent regions to grasp polygonal objects. In *Proc. IEEE Int. Conf. Robotics and Automation*, 778–783.

Cornellà, J. and Suárez, R. (2009). Efficient determination of 4-point form-closure optimal constraints of polygonal objects. *IEEE Trans. Automation Science and Engineering*, 6(1), 121–130.

Han, L., Trinkle, J., and Li, Z. (2000). Grasp analysis as linear matrix inequality problems. *IEEE Trans. Robotics an Automation*, 16(6), 663–674.

Jia, Y. (2004). Computation on parametric curves with an application in grasping. *Int. J. Robotics Research*, 673–692.

Lippiello, V., Ruggiero, F., Siciliano, B., and Villani, L. (2010). Preshaped visual grasp of unknown objects with a multi-fingered hand. In *IEEE Int. Conf. Intelligent Robots and Systems*, 5894–5899. Taipei, Taiwan.

Liu, Y. (2000). Computing n-finger form-closure grasps on polygonal objects. *Int. J. Robotics Research*, 19(2), 149–158.

Miller, A.T. and Allen, P.K. (2004). Graspit! a versatile simulator for robotic grasping. *IEEE Trans. Robotics an Automation*, 11(4), 110–122.

Murray, R., Li, Z., and Sastry, S. (1994). *A Mathematical Introduction to Robotic Manipulation*. Boca Raton, FL:CRC.

Nguyen, V. (1988). Constructing force-closure grasps. *Int. J. Robotics Research*, 7(3), 3–16.

Niparnan, N. and Sudsang, A. (2006). Computing all force-closure grasps of 2d objects from contact point set. In *Proc. IEEE/RSJ*, 319–334.

Pollard, N. (2004). Closure and quality equivalence for efficient synthesis of grasps from examples. *Int. J. Robotics Research*, 23(6), 595–614.

Ponce, J. and Faverjon, B. (1995). On computing three-finger force-closure grasps of polygonal objects. *IEEE Trans. Robotics and Automation*, 11(6), 868–881.

Ponce, J., Sullivan, S., Sudsang, A., Boissonat, J., and Merlet, J. (1997). On computing four-finger equilibrium and force-closure grasps of polyhedral objects. *Int. J. Robotics Research*, 16(1), 11–35.

Roa, M., K.Hertkorn, Borst, C., and Hirzinger, G. (2011). Reachable independent contact regions for precision graps. In *Proc. IEEE ICRA*.

Roa, M. and Suárez, R. (2009a). Computation of independent contact regions for grasping 3d objects. *IEEE Trans. Robotics*, 25(4), 839–850.

Roa, M. and Suárez, R. (2009b). Finding locally optimum force-closure grasps. *Robotics and Computer-Integrated Manufacturing*, 25(3), 319–334.

Rosales, C., Porta, J.M., Suárez, R., and Ros, L. (2008). Finding all valid hand configurations for a given precision grasp. In *IEEE Int. Conf. Robotics and Automation*, 1634–1640. Pasadena, California, USA.

Rosell, J., Sierra, X., Palomo, L., and Suárez, R. (2005). Finding grasping configuration of a dextrous hand and an industrial robot. In *Proc. IEEE. Int. Conf. Robotics and Automation*, 1190–1195. Barcelona, Spain.

Zheng, Y. and Qian, W.H. (2008). On some weaknesses existing in optimal grasp planning. *Mechanism and Machine Theory*, 43(5), 576–590.

Zhu, X., Ding, H., and Tso, S. (2004). A pseudodistance function and its applications. *IEEE Trans. Robotics an Automation*, 20(2), 344–352.

Zhu, X. and Wang, J. (2003). Synthesis of force-closure grasps on 3-d objects based on the q distance. *IEEE Trans. Robotics and Automation*, 19(1), 669–679.