2012 IEEE International Conference on
Automation Science and Engineering (CASE)
Seoul, Korea. August 20-24, 2012

A Common 3-finger grasp Search Algorithm for a Set of Planar Objects

Avishai Sintov[1], Srinivas Raghothama[3], Roland Menassa[4] and Amir Shapiro[1]

Abstract— **This work proposes an algorithm for designing a simple End Effector configuration for a robotic arm which is able to grasp a given set of objects. The algorithm searches for a common 3-finger grasp over a set of objects. The search algorithm maps all possible grasps for each object which satisfy a quality criterion and takes into account an external wrench (force and torque) applied to the object. The mapped grasps are represented by feature vectors in a high-dimensional space. This feature vector describes the shape of the gripper. We then generate a database of all possible grasps for each object represented as points in the feature vector space. Then we use another search algorithm for intersecting all points over the entire sets and finding common points suitable for all objects. Each point (feature vector) is the grasp configuration for a group of objects, which implies for the end-effector design. The final step classifies the grasps found to subsets of the objects, according to the common points found, this with preference to find one grasp to all the objects. The algorithm will be useful for assembly line robots in reducing end-effector design time, end-effector manufacturing time and final product cost.**

I. INTRODUCTION

Today, the mass production in assembly lines and manufacturing plants is done by robotics and automation lines. The need of the plant, the large demand and the higher quality requirements, requires precise and high quality production while meeting very high production rate. The plants' robots move and handle components as needed, with the use of an arm and an end-effector designed specifically for this operation and specific part to handle. Many end-effectors look similar to one another; they are optimized, designed and built for a specific action and for a specific part. The design, build and testing phase of a typical end-effector consumes a considerable amount of engineering time and adds extra cost to the final product.

The purpose of this work is to develop an algorithm which will find a common grasp configuration and imply for the design of a simple universal end-effector for a set of objects. Such algorithm will contribute to the standardization of end-effectors and their components in assembly lines. With given a set of all CAD models of the parts, the idea is to design an end-effector that is universal in the sense of being able to hold a wide set of components for multiple manipulation tasks. The algorithm will be able to characterize the objects geometries and find a configuration for grasping them. As we discuss the design of an industrial end effector, the final configuration has to be simple and low cost. The configuration must be able to reach equilibrium even under the application of external wrenches due to the task being done, i.e., we require force-closure grasp.

This paper presents an algorithm for parameterization of the force-closure grasps of each object and using it for classification of the objects to possible grasps. In the first stage of the algorithm a *Force Closure Grasp Set* (FCGS) is constructed for every object by sampling all force-closure grasps which has a grasp quality measure lower then a predefined value, and representing the possible grasps as feature vectors in the high-dimensional space. Each feature vector injectively defines the grasp and implies for the end-effectors' design.

The next step of the algorithm is the similarity join, for finding pairs of common feature vectors in the FCGS of all objects. The similarity search is based on nearest-neighbor search algorithm for each two FCGS sets, all pairs of vectors which are within a predefined small distance. Such vectors are considered to be the same and inserted into a registry set with association to the respective FCGS. Finally, classification is done in order to find the minimal feature vectors which cover the whole set of objects.

The paper is organized as follows. The next section is a review of related work. Section III gives a background overview of grasping fundamentals used in this work. The structure of the grasp feature vector and the FCGS generation algorithm is described in section IV. Section V presents the main algorithm for similarity join and classification of the common feature vectors. Finally, Section VII contains a summary of the work and proposes future work.

II. RELATED WORK

This work uses the methods of force-closure and quality measure as criteria for determining and quantifying feasible grasps. Using the notion of wrenches (combining forces and torques), in [1], [2], [3] force-closure criterion is well defined and several algorithms for synthesis of a frictional [4] and frictionless [5], grasps were presented. Several grasp optimization methods using different grasp quality measures have been presented in literature; Ferrari and Canny [6] and Li and Sastry [7] introduced a quality measure based on the external wrench to be resisted, where the first introduced a general measure based on the largest wrench that the grasp can resist; the second uses task oriented quality defined by

*This work was supported by General Motors Ltd. under PO No. IME-US-NA-UNIV-11-025.

[1]A. Sintov and A. Shapiro are with the Department of Mechanical Engineering, Ben-Gurion University of the Negev, Beer-Sheva 84105, Israel (Corresponding author: e-mail sintova@post.bgu.ac.il, phone +972-54-5562555). A. Shapiro is also affiliated with the Pearlstone center for aeronautical research and with the Paul Ivanir Center for robotics research and production management.

[3]S. Raghothama is with Global General Motors R&D, India Science Lab. Bangalore 560066, India.

[4]R. Menassa is with Global General Motors R&D, Manufacturing Systems Research Lab, Warren 48090, MI, USA.

978-1-4673-0428-3/12/$31.00 ©2012 IEEE

the specific wrenches applied during execution. The first method is the common one and is embedded in this work.

To the best of our knowledge, no previous work has been done for searching common grasps or end-effectors design for a set of objects. However, much work has been done in the area of 3D shape similarity comparison algorithms, such as [8], [9], local storage search, face recognition, image processing or parts grasping in assembly lines. However, such methods deals with geometry parameterization of the objects and cannot be applied for grasping. The work of Li and Pollard, [10], is based on shape matching for finding the best grasp for a set of objects. The best grasp is found by matching hand poses from a database to each object. This is being done by using a predefined parameterization of the object surface and the hand poses, a method which inspired this work. The work done in GraspIt [11] has similarities to this work in the field of force closure and quality measure analysis.

III. Background

A. Grasping Model

Forces and torques can be represented as a wrench vector in the wrench space. A wrench is a k-dimensional vector where k equals 3 for the planar case and 6 for a spatial object case, and is denoted as $w = \begin{pmatrix} f & \tau \end{pmatrix}^T$ where f is a force vector and τ is a torque vector. Furthermore, A wrench applied at the contact point, c_i, can be described as $w_i = \begin{pmatrix} f_i & p_i \times f_i \end{pmatrix}^T$ where p_i is the location of contact point c_i represented in the object coordinate frame (Fig. 1).

Friction exists between the fingertips of the End-Effector and the object's surface and can be represented by the simple Coulomb friction model $|f_i^T| \leq \mu f_i^N$ where f_i^N and f_i^T are the normal and tangential forces at the contact point, respectively, and μ is the coefficient of friction. According to this model, forces exerted in the contact point must lie within a cone centered about the surface normal. This is known as the *Friction Cone* (FC) and in the planar case can be defined by a linear combination of f_i^+ and f_i^-, which are unit vectors on the edges of the friction cone. The angle between them equals to $2\tan^{-1}\mu$. If the force lies within the FC, the force can be represented as a linear combination given by

$$f_i = \alpha_i^+ f_i^+ + \alpha_i^- f_i^- \tag{1}$$

where α_i^+,α_i^- are nonnegative constants [4]. The associated wrenches can be expressed by the primitive forces as

$$w_i^+ = \begin{pmatrix} f_i^+ \\ p_i \times f_i^+ \end{pmatrix}, w_i^- = \begin{pmatrix} f_i^- \\ p_i \times f_i^- \end{pmatrix} \tag{2}$$

An n-finger grasp can be represented by the position vectors of all contact points $P = (p_1, ..., p_n)$. Equivalently, we can represent the grasp using the matching wrenches applied at the contact points represented in the object coordinate frame $W = (w_1, ..., w_n)$. In frictional grasps, the wrench set can be expressed by the primitive wrenches $W = (w_1^+, w_1^- ..., w_n^+, w_n^-)$.

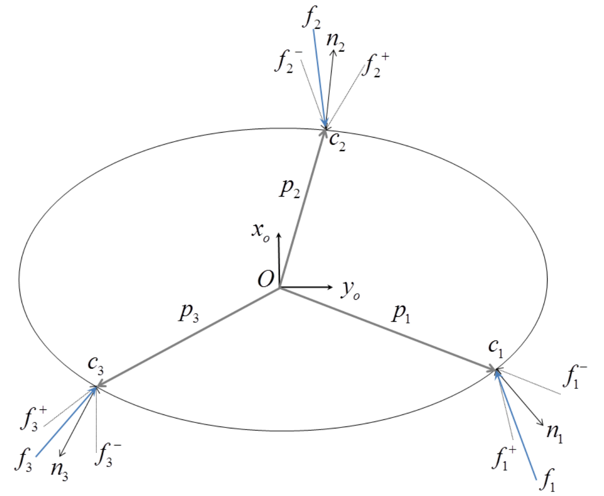

Fig. 1. An ellipse planar object.

B. Force Closure

A grasp is said to be force-closure if it is possible to apply wrenches at the contacts such that any external forces and torques acting on the object can be counter-balanced. A system of wrenches can achieve force-closure when any external load can be balanced by a non-negative combination of the wrenches [1]. Therefore, the notion of the *Convex-Hull* (CH) is introduced. The Convex Hull of set W of wrench vectors is the set of all convex combinations of the subsets of vectors from W. In other words, the CH is the minimal convex set containing W. With system W of wrenches $w_1, ..., w_n$, the $CH(W)$ is defined as

$$\begin{aligned} CH(W) = \{\textstyle\sum_{i=1}^n (a_i^+ w_i^+ + a_i^- w_i^-) : \\ w_i^+, w_i^- \in W, \textstyle\sum_{i=1}^n (a_i^+ + a_i^-) = 1, a_i^+, a_i^- \geq 0\} \end{aligned} \tag{3}$$

The convex hull of the system of contact wrenches is denoted as the *Grasp Wrench Set* (GWS). A necessary and sufficient condition for a system of n wrenches $w_1, ..., w_n$ to be force-closure is that the origin of $\mathbb{R}^k$ lies in the interior of the convex hull of the contact wrenches ([12], [1]),

$$O \in interior(CH(W)). \tag{4}$$

C. Grasp quality measure

The grasp quality measure quantifies the quality of the grasp. In other words, it measures how much a grasp can resist an external wrench without fingers loosing contact or sliding [13]. Increasing the quality of a grasp reduces the magnitude of the contact forces which are required to counter-balance an external wrench. Thus, a higher quality measure reduces object deformations and actuators resources. The quality measure will be used as a grasp criterion for the algorithm presented later in this paper. There are several quality measures; the most common one is the largest ball criterion which is used in this work. The measure is based on a general *Task Wrench Set* (TWS). The TWS is a wrench set of all external wrenches needed to be applied during execution of a prescribed task and is equivalent to the radius of the largest ball centered at the origin of the GWS and fully contained in the Convex-Hull of W [5]. In other words, the grasp quality measure is the distance from the origin of the GWS to the closest facet of $CH(W)$. Mathematically we can say that the quality measure is defined as

$$Q = \min_{w \in \partial W} \|w\| \tag{5}$$

where ∂W is the boundary of $CH(W)$ [6].

Fig. 2 illustrates the GWS and the largest ball contained. The quality measure Q is the radius of the ball. This means that large contact forces would have to be applied if an external wrench will be applied in the weakest direction defined by the vector from the origin to the point where the Q sized ball is tangent to the boundary of $CH(W)$.

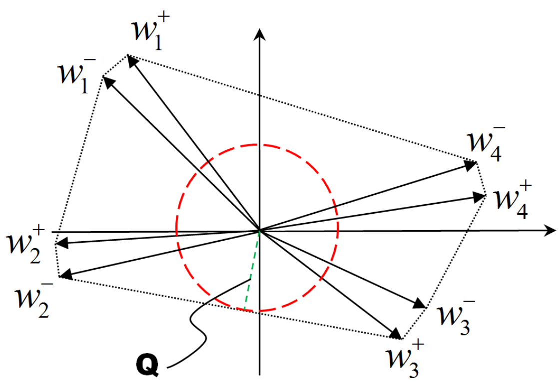

Fig. 2. The Grasp Wrench Space (GWS) and the Task Wrench Space (TWS).

IV. FCGS GENERATION ALGORITHM

We generate the set of all possible grasps for each object. The grasps are represented as feature vectors in the *Force Closure Grasp Set* (FCGS). This section presents the proposed structure of the grasp feature vector and the method for constructing the FCGS.

A. Grasp feature vector

A 3-finger grasp of an object B can be defined by a set of contact points, $P = (p_1, p_2, p_3)$, on the object, and the normal to the object surface at each point (Fig. 1). This grasp definition can be mapped into a set of parameters injectively representing the grasp. A parameters set for grasp j can be written as a d-dimensional feature vector $e_j = (u_1, ..., u_d)^T$.

In this paper only planar objects are discussed. Fig. 3 presents an option for defining a 3-finger grasp. The grasp can be represented by a triangle where the contact points are the vertices. The position of the 3 fingers relative to themselves can be injectively represented as a triangle by two angles γ_1, γ_2 and the edge length between them d_1, given by

$$\cos\gamma_1 = \frac{(p_3 - p_1) \cdot (p_2 - p_1)}{\|(p_3 - p_1)\| \|(p_2 - p_1)\|} \tag{6}$$

$$\cos\gamma_2 = \frac{(p_1 - p_2) \cdot (p_3 - p_2)}{\|(p_1 - p_2)\| \|(p_3 - p_2)\|} \tag{7}$$

$$d_1 = \|p_2 - p_1\| \tag{8}$$

the angles of the normals relative to the triangle edges are given by

$$\theta_i = \begin{cases} \cos^{-1}\left[\frac{n_i \cdot (p_n - p_1)}{\|p_n - p_1\|}\right], & i = 1 \\ \cos^{-1}\left[\frac{n_i \cdot (p_{i-1} - p_i)}{\|p_{i-1} - p_i\|}\right], & i = 2, 3 \end{cases} \tag{9}$$

Therefore, the 3-finger grasp can be injectively defined by a 6-dimensional feature vector e_j,

$$e_j = (\gamma_1 \ \gamma_2 \ d_1 \ \theta_1 \ \theta_2 \ \theta_3)^T \tag{10}$$

where the parameters of the feature vector are given by equations (6)-(9). It should be mentioned that there are three different representations for a triangle, depending on which edge is picked for defining d_1. Therefore, the longest edge of the triangle is always picked to represent d_1. This ensures that the algorithm will always observe the triangles from the same point of view.

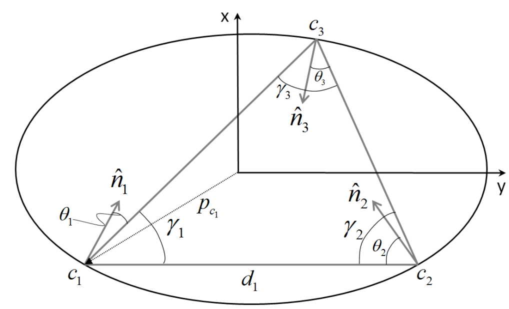

Fig. 3. 6 parameters representing a 3-finger grasp.

The advantage of this feature vector representation is that it is frame invariant. The feature vector defines a grasp independent of the object and its coordinate frame, and therefore can be compared to other feature vectors of other objects.

B. Force closure check

In section III the force-closure criterion was presented. After sampling and generating all grasp feature vectors of a certain object (up to mesh size), force-closure check has to be done to each possible grasp in order to omit inappropriate ones. The force-closure criterion checks whether the grasp satisfies condition (4) and whether the grasp quality measure Q of a feature vector is greater than a predefined value Q_d.

C. Grasp space generation

Generation of the FCGS is done by going over all possible grasp feature vectors of object B, up to mesh size, omitting those which are not force-closure (as described in subsection IV-B). Algorithm 1 receives as an input a mesh of a CAD model. It samples possible grasp feature vectors for the object and computes a set $E \in \mathbb{R}^6$ representing all combinations of 3-points (up to mesh size) which achieve force-closure under the frictional contact constraint. Thus, the output of the algorithm will be a set of feature vectors

$E = (e_1, ..., e_v)$ representing the set of force closure grasps of object B. The algorithm is based on the one proposed in [14].

Algorithm 1 FCGS generation.

```
Input: Mesh of object B to be grasped.
Output: FCGS E = (e_1, ..., e_v) of object B.
 1: Generate grasp j defined by P_j = (p_1, p_2, p_3).
 2: if P_j is force-closure then
 3:     Map grasp j to feature vector e_j = (u_1 ... u_6)^T.
 4:     Label e_j as force-closure.
 5:     Store link between e_j and P_j.
 6: else
 7:     Label e_j as force-closure.
 8:     Remove e_j from E.
 9: end if
10: if E = (e_1, ..., e_v) is not fully labeled then
11:     go to step 1.
12: else
13:     return grasp set E = (e_1, ..., e_v).
14: end if
```

V. MULTI-SETS COMMON VECTORS SEARCH

Given t sets of points in a high-dimensional space $E_1, ..., E_t \in \mathbb{R}^6$ representing the FCGS of each object and a set of parameters tolerances for each dimension $\varepsilon_1, ..., \varepsilon_6 \in \mathbb{R}$. A similarity algorithm will take the sets and output a set of vectors $Z \in \mathbb{R}^6$ that are common to two or more sets of $E_1, ..., E_t$.

A. Similarity join

The main idea of the search algorithm is for each two sets E_i, E_j, to find feature vectors that are common to the two sets. Every vector that is found is checked whether it is already in the registry set Z. If yes, it is labeled to be in E_i and E_j. If not, it is added to Z and then labeled to exist in E_i and E_j.

Let U_t be a t-dimensional vector set of size v consisting binary vectors, i.e., vector with t components consisting of 0's and 1's. Each common vector added to Z is to be a high-dimensional vector $v_i \in \mathbb{R}^6$, denoting the position of the vector in the 6-dimensional space. A compatible vector $\tilde{v}_i \in \mathbb{R}^6$ is coupled to v_i. Component k of vector $\tilde{v}_i$ denote whether the point is within set E_k if labeled "1" and "0" if not.

The $JoinFCGS$ (Algorithm 2) function receives all FCGS data sets $E_1, ..., E_t \in \mathbb{R}^6$ and for each two sets finds pairs of vectors, using Nearest Neighbor (NN) algorithm (implemented using the MATLAB[1] knnsearch function). Each pair is composed of a vector from one set to the nearest one in the other set. After all the pairs were found, further inspection of whether they are close enough to be considered as the same grasp should be done. Therefore, function $PDist(a, b)$ (Algorithm 4) calculates the projection distances between the two vectors along the major coordinates axes and checks whether they are both inside an hyper-rectangle with edge lengths of the predefined tolerances $\varepsilon_1, ..., \varepsilon_6$. If they are, then the vectors are considered to be the same.

[1]Matlab® is a registered trademark of The Mathworks, Inc.

Algorithm 2 Function $JoinFCGS(E_1, ..., E_n)$

```
Input: FCGS of each object E_1, ..., E_n.
Output: Registry set Z.
for i = 0 → n − 1 do
    for j = i + 1 → n do
        [a, b] = NearestNeighbor(E_i, E_j).
        for k = 1 → size(a) do
            if PDdist(a_k, b_k) = true then
                Z = InsertToZ(a_k, b_k, i, j)
            end if
        end for
    end for
end for
return Z.
```

Algorithm 3 Function $InsertToZ(a, b, i, j)$

```
Input: Vector a in FCGS i and vector b in FCGS j.
Output: Updates registry set Z to contain a, b.
v = ((a_1+b_1)/2 ... (a_6+b_6)/2)^T
u = NearestNeighbor(Z, v) */Find NN to v.
if PDdist(v, u) = true then
    ũ_i = 1
    ũ_j = 1
else
    Add point v to Z.
    ṽ_i = 1
    ṽ_j = 1
end if
```

The pair found is the input of procedure $InsertToZ$ (Algorithm 3), which takes the mean point of the pair and checks whether it exists in the registry set Z. As mentioned, the set Z is a 6-dimensional database of the vectors that are common in two or more sets of $E_1, ..., E_n$.

Algorithm 4 Function $PDdist(a, b)$

```
Input: Vectors a and b.
Output: Are the two vectors close enough (true/false).
for i = 1 → 6 do
    if |x(i) − y(i)| > ε_i then
        return false
    end if
end for
```

After constructing the FCGS and finding similar vectors, a registry, high-dimensional vector set Z is generated. This registry is a cluster of vectors constructing the set of common grasp feature vectors that exist in two or more FCGS sets. Each feature vector within Z is marked in which primitive set it exist. Thus, enabling classification described as followed.

B. Classification

A high-dimensional registry set Z of vectors $v_1, ..., v_k \in Z$ is obtained. Each position vector v_i is coupled to a binary vector $\tilde{v}_i$ which denote its existence in any of the primitive sets $E_1, ..., E_t$. The next step is to classify all vectors in set Z to classes. The main objective is to find the minimum subset of points $H \subseteq Z$ which exist in all of the primitive vector sets, such subset is said to cover the whole primitive sets $E_1, ..., E_t$. This is a standard case of the set-cover problem [15], solved using optimization methods. However, as we deal with binary vectors, solution using union of the vectors is enough.

Let a vector $u_i \in Z$, then its compatible vector $\tilde{u}_i \in U_i$ is a binary vector consisting one's or zero's. It can be said that the a subset $H \subseteq Z$ where $u_1, ..., u_p \in H$, covers $E_1, ..., E_t$ if

$$\left. \bigcup_{i=1}^{p} \tilde{u}_i \right|_{min(p)} = (\vec{1})_{n\times 1} \tag{11}$$

As mentioned we want the size p of the subset H to be minimal and seeking for 1s. The class search algorithm will start by searching for a single vector u_1 within Z that covers the whole primitive sets, i.e., all components of the compatible vector $\tilde{u}_1$ equals to 1. If fails to do so, it will search two vectors that covers the FCGS sets, the algorithm will continue to do so until succeeds. In the end of the process, a minimal set H of feature vectors is found, where each one of them is in fact a grasp configuration of a matched subset of the objects.

VI. Simulations and Performance

For simulations of the proposed method, the algorithm was implemented in MATLAB on an Intel-Core i7-2620M 2.5GHz laptop computer with 8GB of RAM. The following simulations present an example of the algorithm operation on 3-finger frictional grasps of four planar objects.

A. Implementation and results

The performance of the proposed algorithm is illustrated using the four 2D shapes shown in Fig. 4. The shapes are described with a mesh of $k = 156$ points uniformly distributed along their boundary. According to algorithm 1, the FCGS is generated for each shape. Each candidate grasp is checked for force-closure and only grasps with quality measure greater than 0.1 are approved (for convenience, the quality measure Q is normalized by the maximum quality measure of all sampled grasps and therefore bounded to be $0 < Q \leq 1$). Fig. 5 shows one generated FCGS of the ellipse shape. Due to the high-dimensionality of the space, for illustration, each space is shown as projections on two 3-dimensional spaces.

The edge lengths of the similarity search's hyper-rectangle $\varepsilon_1, \varepsilon_2, \varepsilon_3 [mm]$ were chosen in such way that each of the edges of the triangle will not extend or shorten more than 6% of their original length due to changes in the triangles angles. $\varepsilon_4, \varepsilon_5, \varepsilon_6 [^\circ]$ are defined to be 70% of the friction cones angle (the friction coefficient was chosen to be 0.6). For these

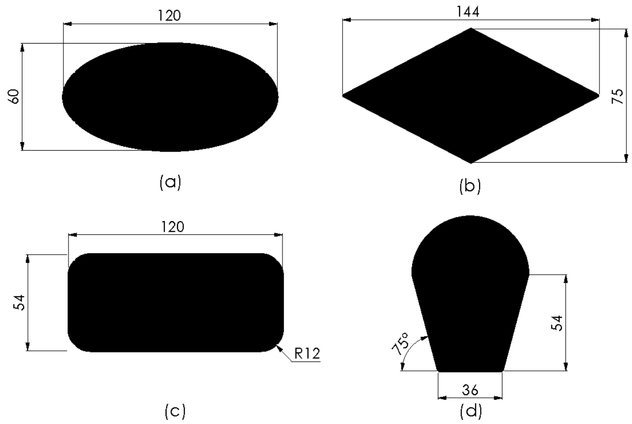

Fig. 4. Four planar objects to be grasped: (a) ellipse, (b) dalton, (c) rectangle and (d) an ice cream cone.

conditions, the algorithms outputs are 8 solutions of common grasps for all shapes. Fig. 6 shows 16171 points/grasps which are common to 2 or more shapes (the blue dots). The 8 point solutions which are marked with red squares are the common grasps for all shapes.

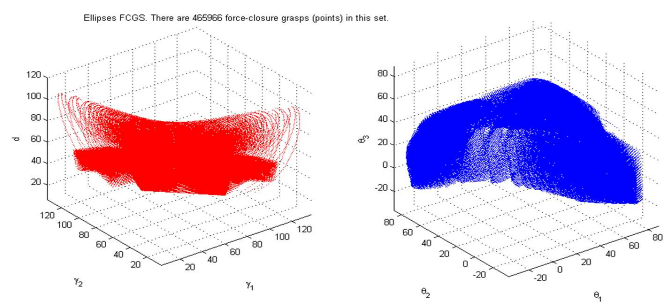

Fig. 5. FCGS for the ellipse object.

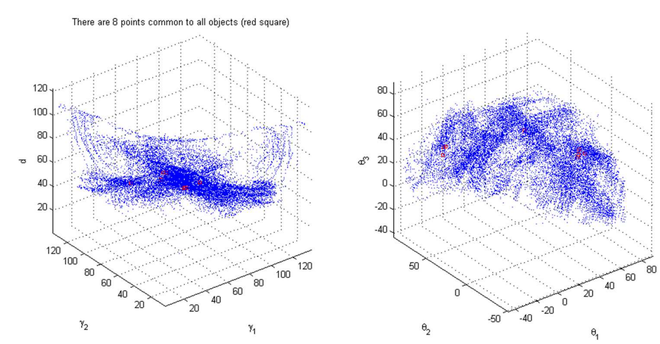

Fig. 6. Registry set Z and the 8 solutions (red squares).

The highest quality measure solution (Q = 0.49) is shown in Fig. 7.

B. Performance

Fig. 8 and Fig. 9 show performance measures as function of the mesh size chosen. Fig. 8 is the time required for the common search algorithm operation. Fig. 9 is the number of solutions found, meaning the number of common grasps for all shapes.

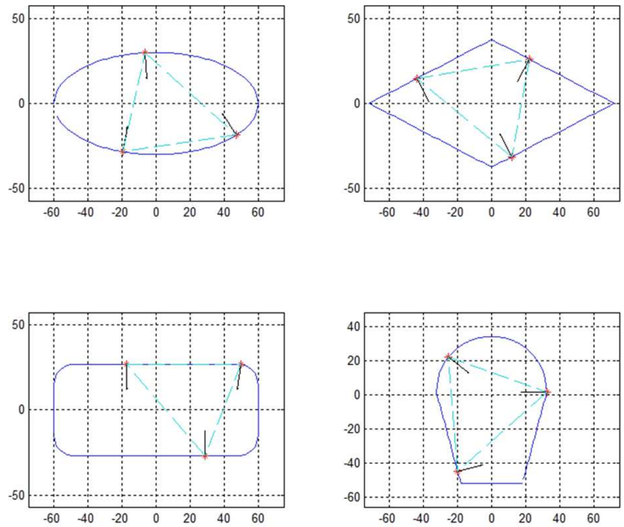

Fig. 7. One of the grasps solutions.

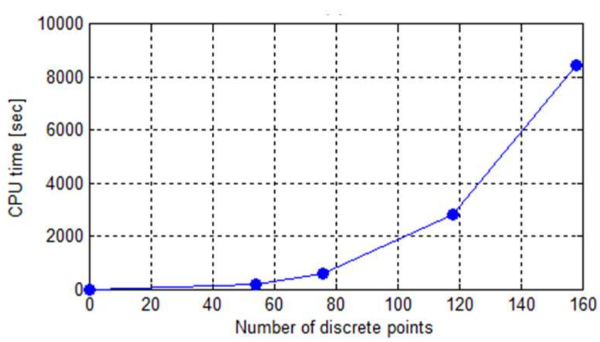

Fig. 8. Solution time relative to the mesh size.

The simulations present the implementation of the algorithm proposed, finding 8 common 3-finger grasps of four objects in about 2 hours (for mesh of 156 points). Of the 8 solutions, we choose the common grasp design with the highest quality measure which is the optimal one for the four objects.

VII. CONCLUSIONS

The main idea of the search algorithm proposed, is for searching high dimensional sets containing feature vectors describing all force-closure grasps of each object. A method for generating the FCGS containing a cluster of points representing grasp feature vectors was presented. Moreover, an efficient similarity join algorithm search was described and a simple classification followed. The feasibility of the algorithm was validated through presented simulations. A minimum set of grasp feature vectors was found that exist in all of the FCGS sets. This means, a set of grasp configurations was obtained, where each grasp can hold the whole set of objects. Every grasp found is in fact a configuration of an end-effector used to grasp the set of objects. The search algorithm for 3-finger common grasps has an overall runtime of the order of $O(n^3)$.

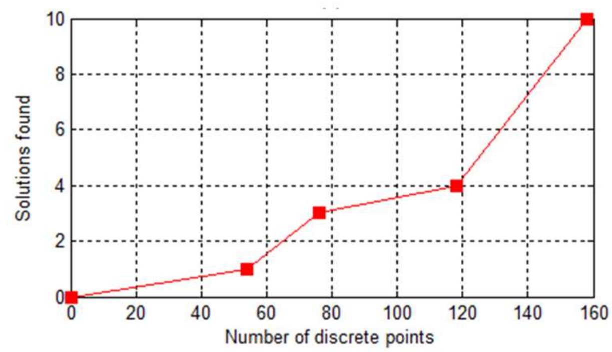

Fig. 9. Number of solutions found relative to mesh size.

Future work will extend the algorithm for the general case of g-finger grasps. Moreover, we would like to modify the definition of the feature vector to accommodate it for the grasp of 3D objects. Finally, we are currently working on experimental verification of the this algorithm and the results presented in this paper.

REFERENCES

[1] J. Ponce and B. Faverjon, "On computing three-finger force-closure grasps of polygonal objects," in *Advanced Robotics, 1991. 'Robots in Unstructured Environments', 91 ICAR., Fifth International Conference on*, pp. 1018 –1023 vol.2, jun 1991.

[2] A. Shapiro, E. Rimon, and S. Shoval, "On the passive force closure set of planar grasps and fixtures," *The International Journal of Robotics Research*, vol. 29, no. 11, pp. 1435–1454, 2010.

[3] N. Niparnan and A. Sudsang, "Computing all force-closure grasps of 2d objects from contact point set.," in *IROS'06*, pp. 1599–1604, 2006.

[4] Y. H. Liu, "Computing n-Finger Form-Closure grasps on polygonal objects," *The International Journal of Robotics Research*, vol. 19, pp. 149–158, Feb. 2000.

[5] M. Roa and R. Suarez, "Geometrical approach for grasp synthesis on discretized 3d objects," in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3283–3288, IEEE, Oct. 2007.

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

[7] Z. Li and S. Sastry, "Task oriented optimal grasping by multifingered robot hands," in *Robotics and Automation. Proceedings. 1987 IEEE International Conference on*, vol. 4, pp. 389–394, Mar. 1987.

[8] R. Ohbuchi, T. Otagiri, M. Ibato, and T. Takei, "Shape-similarity search of three-dimensional models using parameterized statistics," pp. 265–274, 2002.

[9] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin, "Shape distributions," *ACM Transactions on Graphics*, vol. 21, pp. 807–832, Oct. 2002.

[10] Y. Li and N. S. Pollard, "A shape matching algorithm for synthesizing humanlike enveloping grasps," in *Humanoid Robots, 2005 5th IEEE-RAS International Conference on*, pp. 442–449, 2005.

[11] A. T. Miller and P. K. Allen, "Graspit! a versatile simulator for robotic grasping," *Robotics & Automation Magazine, IEEE*, vol. 11, no. 4, pp. 110–122, 2004.

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

[13] C. Borst, M. Fischer, and G. Hirzinger, "A fast and robust grasp planner for arbitrary 3d objects," in *Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on*, vol. 3, pp. 1890 –1896 vol.3, 1999.

[14] M. Roa, R. Suarez, and J. Rosell, "Grasp space generation using sampling and computation of independent regions," in *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pp. 2258 –2263, sept. 2008.

[15] U. Feige, "A threshold of ln n for approximating set cover," *J. ACM*, vol. 45, pp. 634–652, July 1998.