Fast Grasp Tool Design: From Force to Form Closure

Mohammadali Honarpardaz, Martin Meier and Robert Haschke

The self-archived postprint version of this journal article is available at Linköping University Institutional Repository (DiVA):
http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-143804

N.B.: When citing this work, cite the original publication.
https://doi.org/10.1109/COASE.2017.8256199

Original publication available at:
https://doi.org/10.1109/COASE.2017.8256199
Copyright: IEEE
http://www.ieee.org/
©2017 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.
Abstract—We present a novel technique which integrates automatic, part centered design of customized fingertips with a grasp planning stage for arbitrary parts of an assembly process. Starting with a set of CAD models of parts in an assembly sequence, force closure grasps are generated for each part in a sampling process. By employing the sampled force closure grasp positions in the second stage, the fingertip shape model is adapted to obtain form closure grasps at the specific grasp locations on the part. This approach significantly reduces the process time of designing fingers. Furthermore, the method increases the robustness of the fingers grasp for precise assemblies. The proposed fast generic automated finger design (FGAFD) method is able to design fingers for various geometrically complex workpieces. The designed fingers are experimentally verified. The results are discussed in detail and benchmarked against existing approaches.

I. INTRODUCTION

Design automation of robot fingers is one of the most interesting topics for robot industries. Fingers for grippers play a significant role in the overall performance and throughput of robot workcells. Currently fingers are manually designed in robot industries and the process contains several time-consuming trial-and-error iterations in the design, manufacturing and verification of fingers.

Since the manual finger design procedure is incapable to respond to the high demand of agile manufacturing, design automation of fingers is a promising approach that enables robot industries to comply with the growing agile market [1]. SARAFun [2] and Factory-in-a-day [3] are two example of European Commission projects which are defined to enable robot industries to respond to the demands of agile markets.

In our earlier comprehensive review [4] in finger design automation research area, we have shown that the very few proposed methods suffer two significant problems. First, the methods that are unable to handle objects with complex geometry. Second, the methods that are not reliable due to omission of some key processes for designing robust fingers in their procedure.

Recently, the Generic Automated Finger Design (GAFD) method is proposed to handle workpieces with complex geometry while taking all key processes into account [5]. However, the design process time (DPT) of this method is considerably long compared to existing methods. This is due to the time-consuming grasp planning stage of the algorithm which consists of an exhaustive search for a globally optimal grasp set.

This paper proposes a method to reduce the process time of the GAFD method for design automation of customized fingers of industrial grippers for robots. Further, the presented method is compared with the earlier version of the GAFD method and evaluated on a real robot.

The remainder of this paper is organized as follows: Section II reviews the relevant work and Section III explains the employed methodology. Results are presented and discussed in Section IV.

II. RELEVANT WORK

A. Grasp Generation

The process of generating a stable grasp for an unknown part by a robotic manipulator is essential for assembly tasks. Different strategies for grasp generation exist, for example analytical methods [6], leveraging large datasets for machine learning techniques [7] or sampling based approaches [8]. In the present work, we focus on sampling based techniques, since we strive to achieve fast generation of good grasps with a limited amount of data. We can neither assume to have a physically correct model of the assembly parts, nor to record the require amounts of data for machine learning approaches.

The authors in [8] used a sampling approach for stable grasp generation. They emphasize on the importance of sampling and apply perturbations during the sampling process to account for uncertainties which occur in factory settings. In [9], the authors employ a probabilistic approach by modeling the object surface with Gaussian Processes (GP) to account for uncertainties in the object shape before generating grasps. A Monte Carlo based approach was used in [10], to take care of uncertainties during grasp execution in the context of evaluating different grasp quality measures. To avoid explicit modeling of objects, the authors in [11] utilized a probabilistic model to find good pre-grasp positions before using an actuated gripper for force closure grasping. For a more comprehensive overview, the reader is referred to [12] and [13].

B. Finger Generation

In [4], we have highlighted that the current approaches in the field of finger design research may be classified as modular design, re-configurable design and customized design. According to the conclusion, customized design
method are the most suitable approach for finger design automation due to their genericity. Velasco et al. [14] have proposed a method which begins by defining an initial block of fingers, then subtracts the geometry of the workpiece from the finger blocks. The remaining geometry of the finger is considered as the customized designed finger [1,15]. A recently launched tool by SCHUNK (SCHUNK GmbH & Co. KG, Lauffen am Neckar, Germany), eGrip [16], follows the same methodology as [14] proposes. The process begins by manually importing the Computer Assisted Design (CAD) model and then defining mass property of the workpiece. After selecting the type of the gripper (from SCHUNK products), the orientation of the workpiece is defined and afterwards the finger design is initiated. The finger design algorithm generates fingers based on the method presented by [14].

While these methods are able to handle complex workpieces and have a short design process time, they are neither able to provide a solution for axisymmetric objects, nor internal grasps. As a result, recently the Generic Automated Finger Design (GAFD) [5] method was introduced to overcome flaws of the existing customized design methods. This method starts with an analysis of the 3D model of the workpiece and creates a point cloud model of the object by meshing the geometry. In the next step, all possible grasp sets are determined and then the quality of each grasp set is measured using the ball criteria. The grasp set with the highest quality is used to create fingers for parallel-jaw grippers. This method is capable of handling workpieces with complex geometries, yet it has a time consuming design process due to exhaustive search for the globally optimal grasp. In the following section, we introduce a new approach that combines the state of the art finger generation technique GAFD with an efficient grasp sampling method to facilitate the demands of rapid prototyping environments.

III. Method

In order to properly design functional fingers, two essential key processes are employed and integrated in the design process as shown in Fig. 1. The proposed method begins by importing the 3D model of the workpiece to the system. In the grasp planning module, the best grasp set according to a gripper specific quality measure is obtained and exported to the finger design module. Then, the grasp set is used in the finger design module to generate customized fingers. In the next step, the feasibility of the designed fingers is verified by employing a collision detection process to check for unwanted collisions with the environment. Once feasible fingers are obtained, they are produced using additive manufacturing methods. Details of sub-processes in the grasp planning and finger design modules as well as the integration approach of these modules are described in the following sections.

A. Grasp Generation

To facilitate an efficient sampling of grasp poses, the sampling process starts with an analysis of the supplied CAD model of the part, as demonstrated in Fig. 2.

To estimate the volume and orientation of a supplied part, the first step is to obtain data-points on the surface of it. This is achieved by tessellating the CAD model of the part into a mesh to obtain 3D points and wireframe representation, as depicted in Fig. 3 (a).

An additional step is the compute the surface normals at the center of each facet of the mesh via a standard Delaunay triangulation. To facilitate a fast access to these normal vectors during the later grasp evaluation stage, a pair of facet center and the corresponding normal vector is stored in a kd-tree [17] which allows the search for a data-point in logarithmic time. In our case, we have a 3D kd-tree of facet center positions with associated normal direction as additional data assigned to the 3D location. A visualization of these facet normals is shown in Fig. 3 (b), where the xyz direction of a normal is mapped to a corresponding RGB value.
robot may experiences in the working environment. If the end effector pose, to account for uncertainties the real Gaussian noise to the positional and orientational component position is perturbed a fixed number of times by adding with the object. Additionally at each point in this "grid", the kinematically until each end effector, or finger, is in contact acquired, simulated grasps are executed at each collision free. When this reduced set of grasp candidate positions is obtain a fine grained resolution of possible grasp points.

The available 3D positions of the facet centers are also used to estimate the orientation and bounding volume of the part, to perform sampling only in regions of the working space where the part is located. The initial analysis is done by performing a Principal Component Analysis (PCA) [18] over the 3D data. PCA was originally developed as a dimensionality reduction technique. Given a set of n-dimensional data-points, PCA calculates an n-dimensional orthogonal coordinate system, where the length of the axis represents the variance of the data along that axis. The axes are sorted with respect to their variance in a way that the first axis captures the highest variance along the data distribution, the second captures the second largest variance and so forth. This allows a dimensionality reduction by omitting the data along the smallest, i.e. least significant axes. If no axes are omitted, the PCA space represents an oriented bounding volume, similar to an ellipsoid, of a data distribution in an n-dimensional space in which the principal axes span a coordinate system, or local transformation of this volume, which is relative to the origin of the system.

Following the part analysis, the actual grasp sampling is executed in two stages. During the first stage, the fully opened gripper is moved along the PCA axes of the supplied part to check for unwanted collisions with the environment, for example fixtures or tables in the workcell. This movement is constrained to keep the tip of the manipulator within the bounding volume computed by the PCA. We set the sampling step width along the 3D axes of the simulated manipulator to be the same as the repeatability accuracy of the real robot, to obtain a fine grained resolution of possible grasp points.

When this reduced set of grasp candidate positions is acquired, simulated grasps are executed at each collision free position from this set. To this end, the gripper is closed kinematically until each end effector, or finger, is in contact with the object. Additionally at each point in this "grid", the position is perturbed a fixed number of times by adding Gaussian noise to the positional and orientational component of the end effector pose, to account for uncertainties the real robot may experiences in the working environment.

For each contact location between end effector and part, the contact normal is calculated and compared to the closest surface normal of the part, which is precomputed and stored in the kd-tree mentioned earlier. By calculating the absolute value of the cosine between surface and contact normal, we obtain a simple quality of how good the grasp is in terms of force closure. If the value is one, the normals are parallel and the force exerted onto the object is maximal, if the value is zero, the normals are perpendicular and an applied force will perturb the object. In this way, we obtain a contact quality as follows

\[
q_i(n_i, n_j) = \frac{n_i \cdot n_j}{||n_i|| \cdot ||n_j||} \quad \text{Eq. (1)}
\]

where \(n_i\) is the normal of the facet and \(n_j\) is the normal of the contact between finger and part. Assuming \(N\) end effectors, the force closure can be calculated based on the distribution of contacts around the part. In the following evaluation, we utilize a robot equipped with a parallel gripper, so the two fingers are always on opposite sides of the part. To this end, the grasp quality at an end effector pose can be computed as Eq. (2) but can be easily extend to end effectors with more than two fingers by employing e.g. a ball criterion.

\[
q = \frac{1}{2} (q_1 + q_2) \quad \text{Eq. (2)}
\]

Each of the end effector positions that facilitates a successful grasp is stored together with the local contact position of each finger for computing a set of best grasp positions. Since we cannot assume to have any inertia information of the part, the quality \((q)\) of successful grasps is additionally weighted by their distance \((d)\) to the geometric center of the part.

Finally, the set of successful grasps is sorted according to the joint quality value \((q/d)\) and a set of good grasp candidates can be estimated in various ways, e.g. enforcing a minimal distance between different grasp candidates to achieve a high variability between grasp candidates or for example constraint the grasp candidates in a way that only grasps close to the border of the part are taken as successful, to allow a better manipulability with the grasped part for later assembly tasks [19].

B. Finger Generation

In the second stage of GAFD, finger generation, the grasp set with the highest quality is selected to design the fingertips (see Fig. 4). The fingertips are constructed in order to imitate the contour of the surface of the handling objects at the contact point, as demonstrated in Fig 5. This increases the contact area and reliability of the fingers in comparison to simple fingertips such as plane and V-groove contacts. Besides, customized fingertips increases the

![Figure 3. Visualization of the preprocessing of a part, the left image a) shows the result of the tessellation stage as a wireframe model. In b) the normal of each facet is visualized by mapping the x, y and z component of the normal vector to the red, green and blue color channel, respectively.](image)

![Figure 4. Flowchart of processes in the finger design of FGAFD](image)
robustness of the grasp by self-adjustment of the workpiece in position uncertainties.

After generating the fingertips, the bodies of the fingers are designed based on the gripper properties and the grasp types (i.e. internal and external grasp). The finger-bodies are completely parametrized to have full control on the topology of the fingers as well as enable the user to modify the fingers geometry in case of need. Once the finger-bodies are generated, the finger-bases are designed based on the gripper types. As shown in Fig. 6 (a), finger-bases are constructed to connect the finger-bodies to the gripper jaws.

In the final step of the finger generation, the feasibility of the designed fingers is verified by investigating occurrence of unwanted collisions when fingers are approaching the workpiece, as illustrated in Fig. 6 (b). If the approach is collision-free, the designed fingers are considered as feasible and CAD models are exported for the production and experimental verification.

C. Integration and Implementation

The presented approach is implemented based on two specialized software packages. The finger generation component employs the professional software CATIA V5R21 for mesh generation, while the grasp sampling component is based on the open source motion generation framework MoveIT! [20] within a ROS environment.

The reason behind this decision is based on the exclusive features of the CATIA such as power copy which enables the designer to parameterize a template of the object and use it in different modules without needs for referencing on the one hand and the availability of a large set of ready to use robot models in a high performance simulation environment provided by MoveIT! and ROS on the other hand.

To facilitate a seamless communication between the components, we specified a communication protocol following the standard XML remote procedure calls (XML-RPC) [21]. Although more recent RPC protocols exists, we decided to use XML-RPC based on the widespread availability of implementations in nearly every programming language, which eases the adaptation of the developed components to other software frameworks. To account for the communication overhead of XML-RPC, we settled for a small set of high level procedure calls which only initiate the different processes from Fig. 1. The protocol includes the exchange of CAD models, end effector poses and contact location on the parts.

The concrete implementation of the interface is done in C++ for the grasp sampling component to facilitate a ROS integration and in Visual Basic to allow a seamless integration into the CATIA application programming interface (API). However, the method is generic and any CAD based tool that offers an API which can be used for the implementation of the algorithm. The generated fingers are automatically integrated into the open source motion planning framework MoveIT!, which allows fingertip optimization with respect to local contact properties while incorporating the whole kinematic chain of the robot and its environment, thus facilitating global optimization of the finger shape, for example in case of external collisions during motion planning.

IV. RESULTS AND DISCUSSIONS

To be able to fairly measure the performance of the proposed method with existing one, the FGAFD algorithm is implemented for 22 real industrial assembly applications. Table I presents six of the examined workpieces and the design process time of each workpiece using GAFD and FGAFD methods. Information regarding the remaining 16 workpieces, generated fingers and processing time can be found online at [22]. The design process time for each workpiece is measured from initiating the algorithm until feasible finger designs are obtained. A subset of the automatically generated fingers is 3D printed in acrylonitrile butadiene styrene (ABS) polymer and examined in a real grasping application. Fig. 7 (a, b and c) respectively represent execution of the PCB-and-Cover, Clip-and-Board, Lamp Base-and-Cap assemblies using a YuMi (IRB 14000) [23] robot as the platform.

After the preprocessing described in section III.A, the generation of the surface normal tree and the calculation of the sampling space, the grasp sampling process is initialized with a sampling step size of 0.25 (mm) for translational movements and 0.1 degree for rotational movements. At each position in the sampling grid, an additional Gaussian perturbation is added to the pose of the end effector with zero mean and $\sigma = 0.1$ for the translational and $\sigma = 0.05$ for the rotational component, respectively.

Fig. 8 displays two important properties for the sampling process of the used models, the bounding volume of the model and the number of facets. The median number of facets in the case studies is 4016 and the median volume is...
The raw sampling time per model, excluding the time for finger generation, is shown in Fig. 9, together with the sampled volume per second and the number of sampled facets per second. Compared to the model properties in Fig. 8, a clear relation between the bounding box volume and the sampling time can be seen. This behavior is evident, since the sampling is performed on a grid that is defined by the bounding volume, so an increase in the bounding volume accounts for a proportional increase in the sampling time. The number of facets on the other hand has a small impact on the sampling time, since the facets and corresponding surface normals are stored in an efficient data structure and can be queried in logarithmic time. This can also be seen in (FIG REF TIME PLOT). Although the hexnut model is a much higher resolution than the gripper platform, they are processed in nearly the same time, 3.7 (s) for the hexnut and 3.9 (s) for the gripper base.

According to the Table II, the design process the FGAFD method for 22 selected case studies is on average 20.67 times faster than the GAFD method. However, the difference in the design process time (DPT) of these methods highly depends on the size (volume) of the workpiece and also the complexity (i.e. number of facets) of the workpiece has a minor influence. In other words, the performance of the FGAFD method is less dependent to geometrical properties of the workpiece in comparison to the GAFD method.

Table I. Design process time (DPT) of industrial components using GAFD and FGAFD methods.

<table>
<thead>
<tr>
<th>Workpiece Name</th>
<th>Workpiece</th>
<th>DPTGAFD [s]</th>
<th>DPTFGAFD [s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCB</td>
<td></td>
<td>2653.84</td>
<td>96.42</td>
</tr>
<tr>
<td>Cover</td>
<td></td>
<td>2139.06</td>
<td>87.98</td>
</tr>
<tr>
<td>Clip</td>
<td></td>
<td>217.83</td>
<td>22.88</td>
</tr>
<tr>
<td>Board</td>
<td></td>
<td>3628.13</td>
<td>122.11</td>
</tr>
<tr>
<td>Lamp Base</td>
<td></td>
<td>3338.88</td>
<td>103.15</td>
</tr>
<tr>
<td>Lamp Cap</td>
<td></td>
<td>438.26</td>
<td>42.97</td>
</tr>
</tbody>
</table>

Figure 7. Experimental verification of fingers designed by FGAFD for a) PCB-and-Cover; b) Clip-and-Board; c) Lamp Base-and-Cap assemblies.
Figure 8. The figure visualizes two of the most important properties of the used CAD models in the evaluation, the physical volume of the model and the resolution in terms of facets. As shown in this figure, the employed dataset covers a large variety of different volumes and resolutions.

Figure 9. Plot of the time needed during the grasp sampling stage of the approach for each model (red) and two combined values to account for important object properties, the volume of the model and the resolution in term of the number of facets.

TABLE II. REDUCTION OF DESIGN PROCESS TIME FOR INDUSTRIAL COMPONENT EXAMPLES USING FGAFD METHOD IN COMPARISON TO GAFD.

<table>
<thead>
<tr>
<th>DPT_{GAFD} / DPT_{FGAFD}</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCB</td>
</tr>
<tr>
<td>Cover</td>
</tr>
<tr>
<td>Clip</td>
</tr>
<tr>
<td>Board</td>
</tr>
<tr>
<td>Lamp Base</td>
</tr>
<tr>
<td>Lamp Cap</td>
</tr>
</tbody>
</table>

V. CONCLUSION

This paper introduced the Fast Generic Automated Finger Design (FGAFD) as a quick method for the automation design of customized fingers for industrial grippers (e.g. parallel-jaw). The main reason of developing finger design automation methods is to reduce design process lead-time and complexity in order to enable robot industries to comply with the trending agile market. Since existing methods are either only capable of handling workpieces with specific geometries or not fast enough to fully satisfy the industry requirements, this paper proposes a fast and generic method to automate the finger design process. We evaluated the proposed method with a large variety of workpieces for which customized fingers have been generated. A subset of these fingers was evaluated on a real robot platform and successful, stable grasps have been shown. We additionally discussed the performance of the presented approach with respect to important properties of real workpieces and their CAD models, namely the volume of the workpiece and the resolution of the model. It has been shown that the presented method scales especially well with high resolution CAD models and also with models having a high physical volume. For complex CAD models, the presented approach is able to perform automatic finger generation within minutes, which meets the needs of rapid prototyping demands. Since the included grasp generation component is based on a constraint sampling method, next steps in the development will include a thorough comparison of the quality of generated fingers compared to a globally optimal grasp position, to compare the time saving aspect in a more extensive context.

ACKNOWLEDGMENT

Xiaolong Feng and Jonas Larsson at ABB and Mehdi Tarkian and Johan Ölvander at Linköping University are gratefully acknowledged for technical discussions and various supports.

REFERENCES


