

# Finger design automation for industrial robot grippers: A review

M. Honarpardaz<sup>a,\*</sup>, M. Tarkian<sup>b</sup>, J. Ölvander<sup>b</sup>, X. Feng<sup>a</sup>

<sup>a</sup> ABB, Corporate Research Center, Forskargränd 7, Västerås 721 78, Sweden

<sup>b</sup> Linköping University, Division of Machine Design, Linköping 581 83, Sweden

## HIGHLIGHTS

- An exhaustive literature review in finger design automation relevant research fields is carried out by filtering down thousands of articles to a manageable number of articles.
- Key processes for successfully achieving automatic finger design are identified and research contributions in each key process are critically reviewed.
- The proposed approaches in each key process are analyzed, verified and benchmarked.
- The most promising methods to accomplish finger design automation are highlighted and presented.

## ARTICLE INFO

### Article history:

Received 22 December 2015

Accepted 4 October 2016

Available online 18 October 2016

### Keywords:

Grippers fingers  
Design automation  
Grasp synthesis  
Grasp analysis

## ABSTRACT

Designing robust end-effector plays a crucial role in performance of a robot workcell. Design automation of industrial grippers' fingers/jaws is therefore of the highest interest in the robot industry. This paper systematically reviews the enormous studies performed in relevant research areas for finger design automation. Key processes for successfully achieving automatic finger design are identified and research contributions in each key process are critically reviewed. The proposed approaches in each key process are analyzed, verified and benchmarked. The most promising methods to accomplish finger design automation are highlighted and presented.

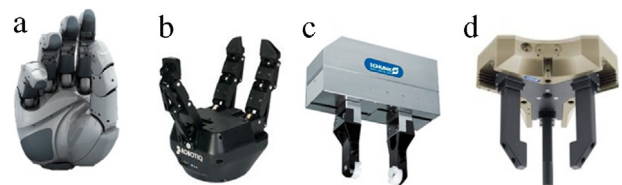
© 2016 Elsevier B.V. All rights reserved.

## 1. Introduction

This paper aims to review research contributions relevant for finger design automation. Designing functional fingers of robots is one of the most complex and sensitive criteria in robotics. The current iterative and time consuming procedure of designing fingers cannot fulfill the demands of agile manufacturing. Therefore, finger design automation plays a significant role in the competitiveness of the robotic automation in the agile market.

Designing gripper fingers properly can increase the workcell throughput, overcome robot inaccuracy and enhance overall system performance. Moreover, a poor finger design can damage the expensive robot hardware or the workpiece and consequently reduce the workcell throughput and reliability [1].

Industrial robots are designed in various sizes and payloads to fulfill a wide range of tasks. Some robots are dedicated to specific tasks, for instance welding, painting and cutting. These robots require special end-effectors called tools. Others are designed based on the task and operation environment to execute general operations, for instance *assembly* and *pick and place*. These end-effectors are the fundament of the review presented in this paper.



**Fig. 1.** (a) DLR hand HIT; (b) ROBOTIQ Adaptive 3-finger; (c) SCHUNK WSG; (d) SCHUNK PZH-plus.

An end-effector is the only interface between the robot and the working environment. Therefore, the overall performance of a robot highly depends on its end-effector and research in this area is of high importance for the industry.

The present work addresses two major categories of robotic end-effectors; *hands* and *grippers*. Hands are multi-finger end-effectors with more than one degree of freedom (DOF) per finger (see Fig. 1(a) and (b)) and grippers usually have 2 or 3 fingers with one DOF, as shown in Fig. 1(c) and (d). Hands are designed for general purpose grasps while grippers are designed for more specific tasks.

\* Corresponding author.

E-mail address: [mohammadali.honarpardaz@se.abb.com](mailto:mohammadali.honarpardaz@se.abb.com) (M. Honarpardaz).

The most common grippers in industry are parallel-jaw which are typically actuated by pneumatic systems. In these grippers, two opposing parallel axis fingers (or jaws) move toward and away from each other by using a linear motion system that keeps both gripper attachments parallel and collinear [2].

Today, a sophisticated process is required to design fingers that accomplish assembly tasks. An example of essential finger design processes for a given gripper is presented in Fig. 2. This figure is a result of the paper review, however it is presented early in order to easier explain the finger design automation process.

The process in Fig. 2 starts with defining the robot task which can be pick and place or assembly. In the case of an assembly task, external data regarding the assembly process are required to be imported to the system. In the next step, the 3D model or 3D vision of the workpiece are imported based on the availability of knowledge about the workpiece. Then, the type of grasp (i.e. form and force closure) is defined based on the geometrical and physical properties of the workpiece and the fingers. In the grasp synthesis and analysis process, the most suitable grasp contact sets for accomplishing the task are listed. Then fingers are designed based on the contact set. In the collision detection process, the feasibility of the designed fingers are verified by detecting possible unwanted collisions. In the occurrence of an unwanted collision, another contact set is selected. Otherwise, the fingers are experimentally verified and if they were promising the design process successfully terminates.

The presented processes in Fig. 2 are categorized into groups of *grasp*, *finger design* and *experimental verification* (see Fig. 3).

Completing the presented design process in Fig. 2, which consists of an iterative trial and error procedure, may take several weeks for an expert, while the agile market demands a quick design process [3]. As a result, automated finger design is of the highest importance for the robotic industry in order to comply with agile manufacturing [4].

In this paper, each key process is critically reviewed and the most efficient methods with respect to lead-time, accuracy and accessibility are highlighted. The employed methodology for the reviewing process is comprehensively described in Appendix. Results of the review are presented in Section 3. Section 4 discusses the research gaps and future work.

## 2. Results

This section presents the state-of-the-art papers collected in the review process. Based on the pre-review performed on the finger design research area (see Appendix), key processes that are essential to be able to automate the finger design procedure are identified. Fig. 2 presents an example of the superimposing of the finger design methods found in the pre-review phase. As shown in Fig. 3, the key processes are divided into three groups of *grasp*, *finger design* and *experimental verification*. In the following subsections, the detailed review of each key process is presented.

### 2.1. Grasp

Grasp in the context of robotics means holding the workpiece to accomplish the defined task. A task failure may cause expensive consequences, such as damaging the robot or the gripper. Thus, securing a robust grasp is essential in finger design.

As shown in Fig. 4, the two most important classes of grasps are *form-closure* and *force-closure* [5]. Reuleaux [6] first used these terms in his machine design studies and later [7] applied them to robotic grasping. The definition of form-closure and force-closure has been repeated in almost every work in robotic grasp and in most cases different understanding has been extracted from them. As a consequence, these terms are also defined in this work to

avoid misunderstanding and confusion. In this review we follow the definitions presented by Nguyen [5] and Bicchi [8].

**Force-closure definition:** A grasp is in *force-closure* if the fingers can resist the motion of the workpiece by applying arbitrary wrenches (with the set of contacts) on the object [5].

**Form-closure definition:** A grasp is in *form-closure* if only the location of the fingers contacts on the workpiece fix the position of the object in space [8].

A contact is defined as the connection point between the finger and the object. Accordingly, a contact model defines the forces (and moments) that can be transmitted through the contact to the object. Based on the geometry of the contact surface and the material properties (i.e. friction coefficient), three type of contact models are defined: point contact model with and without friction and the soft-finger contact model [7]. The point contact model without friction is used in form-closure grasps and the point contact model with friction (also called hard-finger contact model) and the soft-finger contact model are used in force-closure grasps.

The process of finding suitable grasps for an object involves searching for stable grasp locations, which is called *grasp synthesis* or *grasp planning*, and measuring the quality of the grasp, which is called *grasp analysis*. Fig. 5 presents the grasp categorization.

#### 2.1.1. Grasp synthesis (planning)

Grasp synthesizing can be defined as a search process in the space of possible grasp locations aimed at identifying closed grasps.

In general, the grasp synthesis research area is noticeably wide and, in some sense, unstructured. None of the proposed approaches have been widely used and considered as the reference for comparison. Liu and Carpin [9] pointed out two of the most important reasons for this issue. First, there is agreement on neither the definition of standard objects for grasping nor the amount of available preliminary knowledge for grasp planning. The second reason is due to the fact that most of the authors do not make their implementation available for others. Therefore, benchmarking and experimental comparison between different methods is limited to a small portion of research.

In the last two decades, several surveys [10–16] have been carried out in order to somehow categorize and organize the research done in this area. Sahbani et al. [15] present a clear overview of the methods and algorithms that are proposed until 2012. They divide the research in grasp synthesis into *analytical* approaches and *data-driven* (also known as *empirical* or *knowledge-based*) approaches. This paper follows the same definition and categorization.

#### Analytical Approaches

Analytical approaches consist of methods that use geometric, kinematic and dynamic formulations in order to verify a grasp. Bicchi and Kumar [13] have properly reviewed these analytical formulations. In this work, analytical approaches are classified into: *limited contact methods for polyhedral objects*, *unlimited contact methods for polyhedral objects*, *heuristic methods* and *methods for general object shape*.

#### Limited Contact Methods for Polyhedral Objects

The most basic approaches for grasp synthesis consider the minimum number of required contact points for force-closure of objects with polyhedral geometry. Nguyen [17] proves that non-marginal equilibrium grasps are force-closure grasps if each grasp has at least two soft-finger contacts or three hard-finger contacts in 3D. He proves that a grasp with 2 soft-finger contacts is force-closure if and only if the contact points connection line is located in the contacts friction cones. Also a grasp with at least three distinct hard-finger contacts is force-closure if it is in equilibrium, with contact forces pointing strictly within the friction cones at

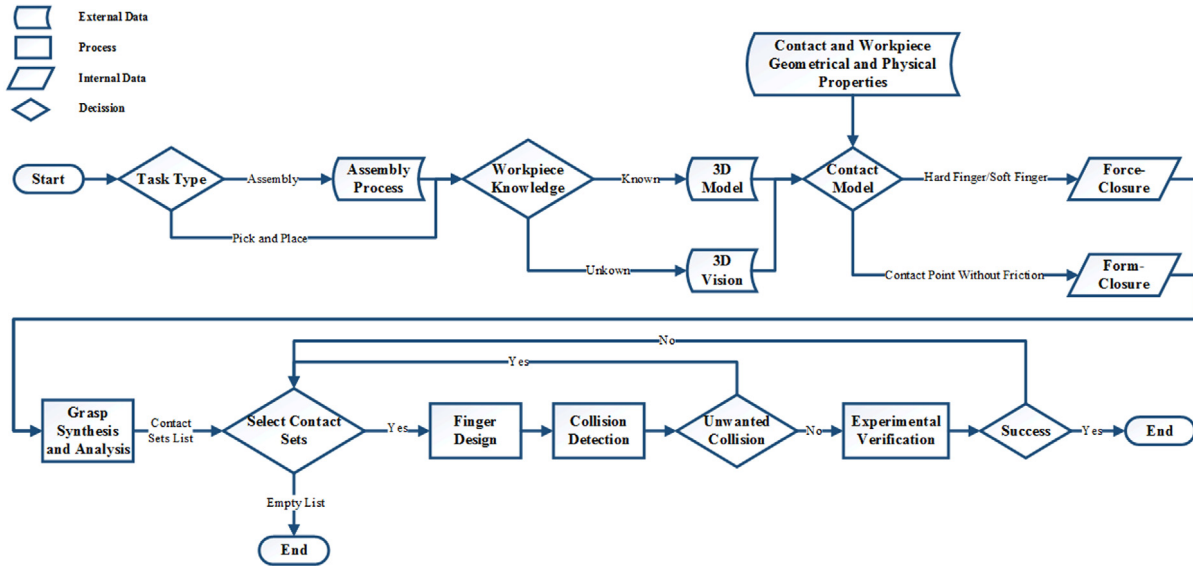


Fig. 2. Flowchart of an example of finger design process for a known gripper.

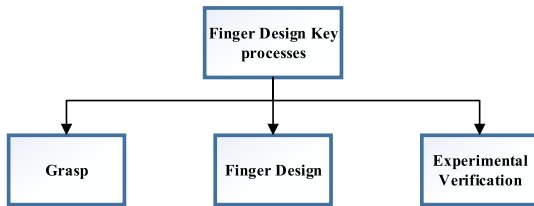


Fig. 3. Finger design key processes categorization.

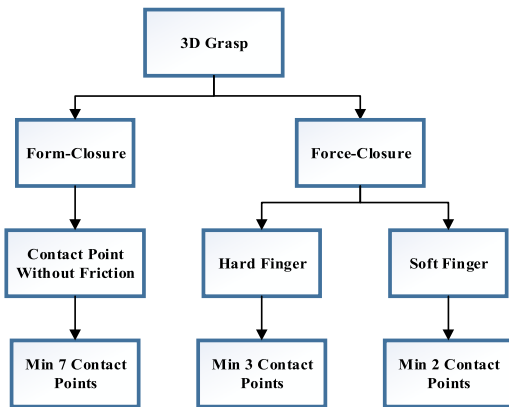


Fig. 4. Manipulator grasp classification.

the respective points of contact. Nguyen also presents that at least 7 contact points are required to achieve from-closure with frictionless contacts (see Fig. 4).

Brown and Brost [18] propose a method that imports the disturbance wrenches to be applied to the object and tests grasps for force-closure. While their method has the benefit of computing the minimum required jaw closing force using linear programming, it would be difficult for the user to know all of the disturbance wrenches. Balan and Bone [19] proposed a closed-form algorithm for testing non-marginal equilibrium grasps [17]. In comparison to [18], this method has the advantage that the closed-form solution has a much lower computational complexity. Still it has the disadvantage that the assumptions employed are conservative, which means it may reject some valid force closure grasps. Li et al.

[20] propose a method for grasp synthesizing three-finger grippers with hard-finger contacts. Their algorithm computes the force-closure grasp of 3D workpieces based on the intersection of the three contact point's triangle area and each contact's friction cone.

**Pros:** The searching method in ches enables global contact sets to be found.

**Cons:** These methods are applicable to grasps with a very limited number of contact points. Thus, more general methods are required for synthesizing grasps regardless of number of contact points.

Fig. 6 represents the flowchart of an example of limited contact methods for polyhedral objects.

#### Unlimited Contact Methods for Polyhedral Objects

As in some cases number of contacts may exceed the minimum of requirements, Han et al. [21] present a qualitative test based on linear matrix inequalities that deal efficiently with frictional constraints to avoid the linearization of the friction cones by considering the fact that a necessary and sufficient condition for force closure grasps is that the origin of the wrench space lies precisely inside the convex hull of the primitive contact wrenches resulted by the contact forces at the fingers [7,22]. Another method is proposed by Liu and Wang [23] that tests the force closure condition by using a ray-shooting problem and solves as a linear programming problem. Based on the force-closure tests presented in [21,23], 3D grasp synthesizer algorithms for unlimited number of contacts are proposed. Liu [24] propose a grasp synthesis method for  $n$  contacts when  $n - 1$  contacts have fixed positions and the grasp with the  $n - 1$  contacts is not force-closure. Utilizing the linear parametrization of a point on a facet, the algorithm searches locations on that face for the  $n$ th contact that secures force-closure.

**Pros:** These methods are applicable to any number of contacts.

**Cons:** The approaches suffer from an exhaustive search procedure.

The flowchart of an example of unlimited contact methods for polyhedral objects is demonstrated in Fig. 7.

#### Heuristic Methods

Heuristic methods start by generating a large number of grasp candidates randomly [25] based on a predefined procedure [26]

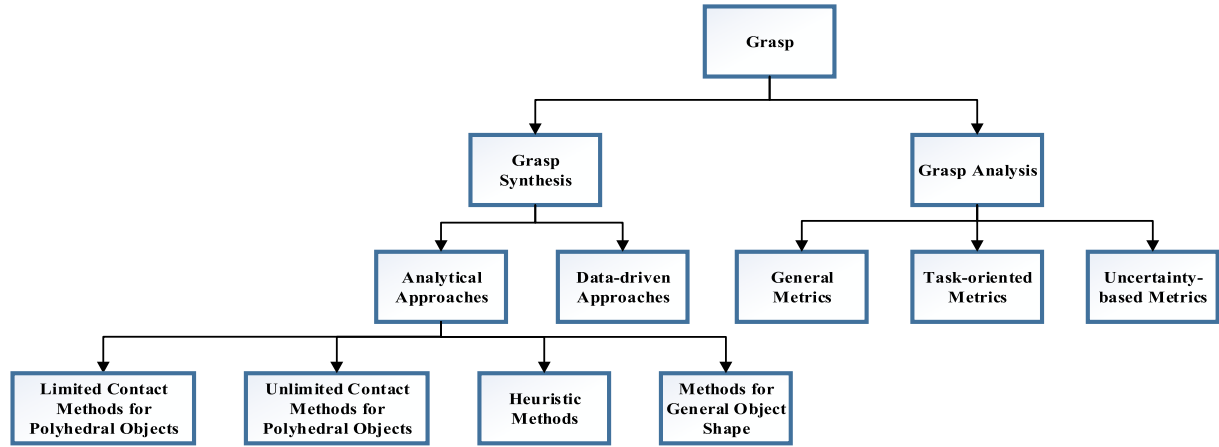


Fig. 5. Grasp categorization.

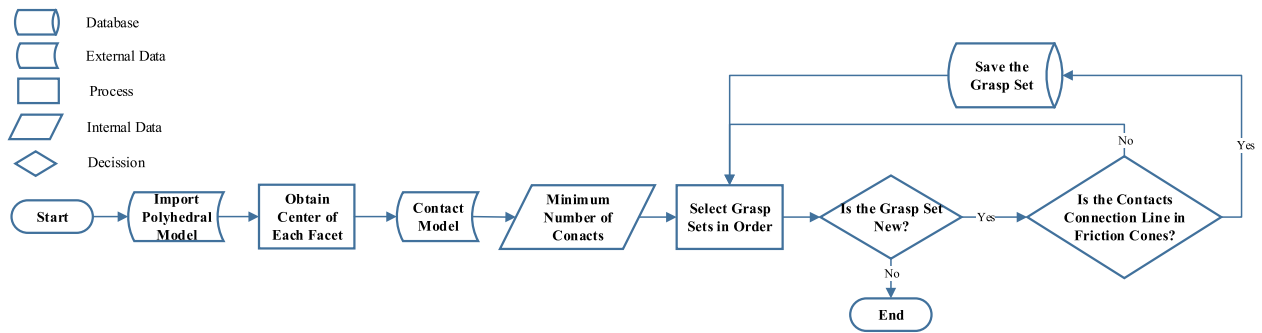


Fig. 6. Flowchart of an example of limited contact methods for polyhedral objects.

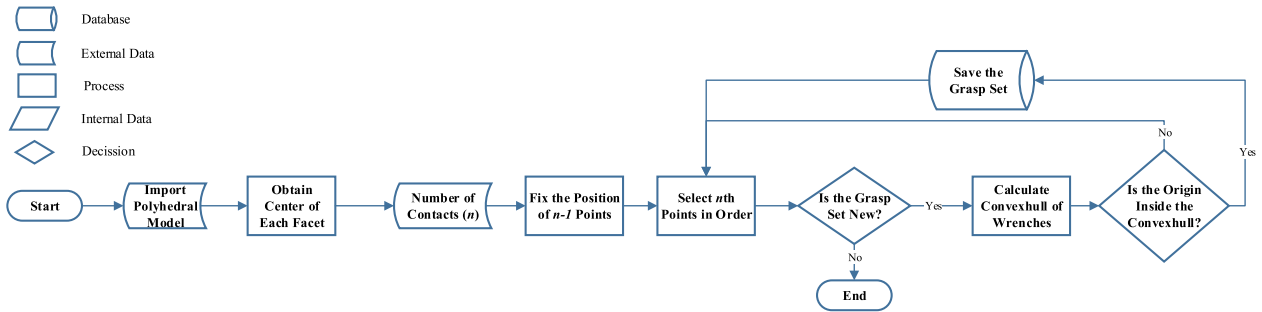


Fig. 7. Flowchart of an example of unlimited contact methods for polyhedral objects.

or by defining a set of rules that can be tested on the workpiece model [27,28]. Then unfeasible grasp candidates that do not fulfill the force-closure condition are filtered out. Ding et al. [29] present an algorithm to compute the positions for  $n$  contact points to form a force-closure grasp from an initial random grasp. The algorithm starts with selecting randomly a grasp on a given facet of the polyhedral workpiece. If the origin of the wrench space lies outside the initial wrenches convex hull, the algorithm moves each contact position by using the linear parametrization of a point on an object face. The method moves the contact point location so that the convex hull moves toward the origin until the force-closure property is fulfilled.

**Pros:** These approaches can be utilized for unlimited number of contacts low computational effort in compare with *unlimited contact methods for polyhedral objects*.

**Cons:** Since these methods do not take all the possible grasp sets into account in their searching procedures, they are limited to determining the local optima.

Fig. 8 shows the flowchart of an example of heuristic methods for grasp synthesis.

#### Methods for General Object Shape

Despite the fact that the methods proposed for polyhedral objects can be implemented for any number of contact points, they are only computationally efficient for very simple objects with few facets. As a result, new general techniques are required for grasp synthesis that are first, independent of object model constraints; and second, do not require such a large computation time. To overcome these issues, researchers propose heuristic approaches with modeling workpieces as a cloud of 3D points or triangular mesh. Ding et al. [30] present an algorithm to synthesize force-closure grasps with 7 frictionless contacts. In their method the workpiece



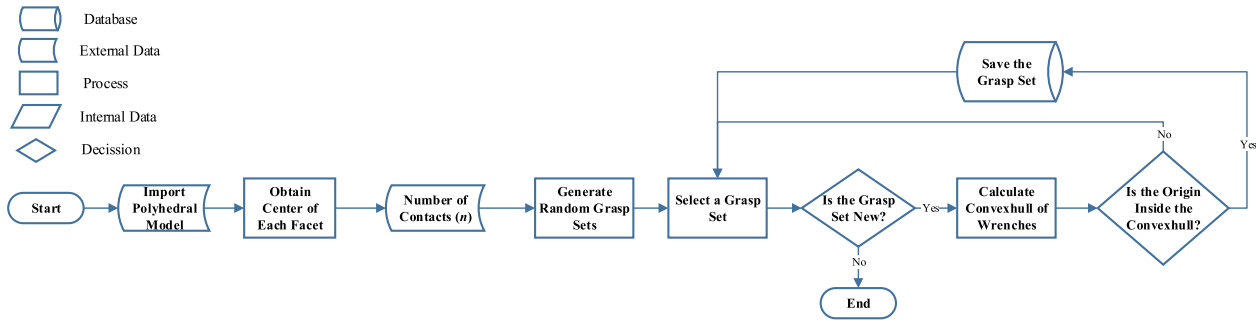


Fig. 8. Flowchart of an example of heuristic methods.

is discretized to a point cloud. It has been assumed the point normals are available so a large compilation of contact wrenches can be determined. The algorithm begins with a primitive set of seven contacts randomly selected among the set of points. The algorithm terminates when the chosen grasp is in force-closure. Otherwise, the initial contacts are iteratively replaced with other grasp candidates until a force-closure grasp is achieved. Liu et al. [31] extend this algorithm for contacts with friction. The recent work from Liu and Carpin [9] introduces a new grasp synthesizer using triangular meshes. Their method relies on the ability of CAD software to represent the workpiece with triangular meshes. The proposed algorithm starts by randomly placing contact points on the workpiece and then iteratively modifying the contact positions while locally searching for better possible grasp locations. There are two interesting features that differentiate this method from other optimization methods. First, the searching method does not require initial force-closure grasps; and second, it allows the contact points to cross boundaries (e.g. sharp edges). These features make their method to be flexible and independent of size and complexity of the objects geometry.

**Pros:** These approaches noticeably reduce the computation effort on searching for grasp solution.

**Cons:** Like *heuristic methods*, these methods suffer from the local optima problems.

The flowchart of an example of methods for general object shape is illustrated in Fig. 9.

#### Data-driven Approaches

In data-driven approaches, grasp candidates are sampled for the workpiece and then are ranked based on a specific metric. This methodology normally relies on some existing grasp experience that can be heuristic or is generated in simulation or on a real robot [15]. Bohg et al. [16] review the studies done on *data-driven* approaches in the context grasp synthesis context. Their review concludes that most of the studies in this area are developed for robot hands which are not interesting for industrial applications. Among few approaches that are applicable to grippers as well, Kim et al. [32] introduce a new algorithm to automate generating possible grasp sets through a simulation-based method. Their approach is different from most other studies as it focuses on the grasp process instead of the final grasp configuration. The proposed method starts with defining a grasp with combination of the relative pose of the hand (or gripper) and grasping approach. Then the algorithm selects stable and realistic grasp sets which are more suitable for power grasps. Wolniakowski et al. [33] recently propose a data-driven method for industrial parallel-jaw grippers. Their algorithm starts with generating a grasp database by sampling on nearly parallel surfaces. Then a dynamics based grasp simulator is used for evaluating grasps. If the gripper can be placed in the sample grasping pose without colliding with the workpiece, the sample is moved to another database, called *targets*. In the final step of their

algorithm, the grasp sets in the targets database are measured by three defined quality metrics.

**Pros:** These methods in comparison to analytical approaches are computational cheap.

**Cons:** The proposed approaches require a great deal of manual input and therefore are not applicable for finger design automation.

#### 2.1.2. Grasp analysis

Once the grasp planner and the contact models are established, a grasp analyzing method is required to test the grasp's resistance against disturbances. Although the criteria presented in Section 2.1.1 guarantees the force-closure grasp, they do not provide any information about the quality of the selected grasp. Therefore in order to analyze grasp candidates, some quality metrics are need to be defined. Several quality metrics are introduced in the grasping literature which are extensively reviewed by Roa and Suárez [34]. They categorize the quality measures into two major groups of *contact position based metrics* and *hand/gripper configuration based metrics* that are based on the utilized methodologies. In this review, the quality metrics are grouped as *general metrics* and *task-oriented metrics* based on types of the measurements.

##### General Metrics:

General metrics are the quality measures that do not take the (assembly) task into account. As these metrics provide a general information regarding stability of the grasp, they are more suitable for simple pick and place tasks. Ferrari and Canny [35] introduce the most popular methods for measuring grasp quality. They propose using the largest disturbance wrench that can be resisted in all directions by the contacts as a grasp quality metric. From a geometrical perspective, this resembles the radius  $\varepsilon$  of the largest ball centered at the origin, which is inscribed in the convex hull of the unit contact wrench (see Fig. 10). To determine this metric, computation of a convex hull in six dimensions is required. The freely available QuickHull [36] implementation [37] is widely used to solve this problem. Computing the convex hull for every set of grasp is computationally expensive.

Liu and Carpin [38] have shown that for the force-closure grasps most of the computation of the convex hull is unnecessary based on the definition of the quality metric presents by Ferrari and Canny [35]. Thus, Liu and Carpin [38] propose the Partial Quick Hull (PQH) algorithm to efficiently compute the metric. PQH determines, on the fly, when the computation can be ended as the quality metric can be already computed before completely determining the convex hull. This method noticeably reduces computation time when using this metric. Zheng [39] presents a different algorithm for the grasp quality measure proposed by Ferrari and Canny [35]. Zheng's method iteratively grows a polytope containing the origin in the grasp wrench space (GWS) as though the minimum distance between the origin and the facets of the polytope quickly converges

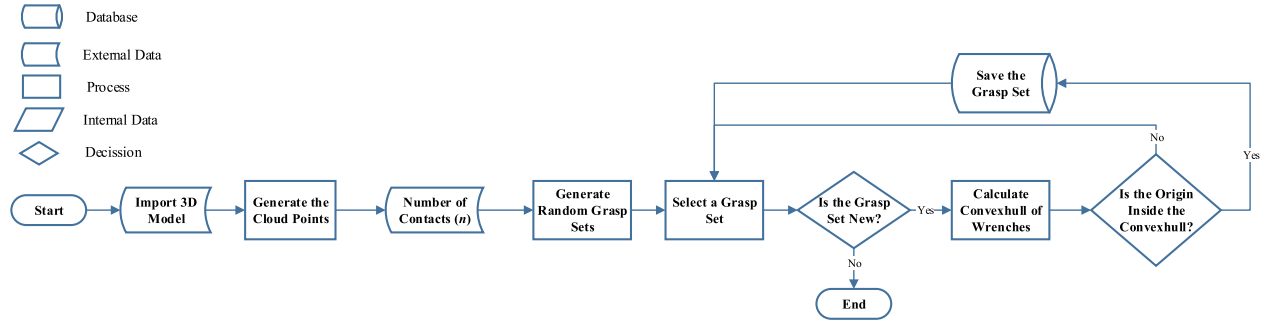


Fig. 9. Flowchart of an example of methods for general object shape.

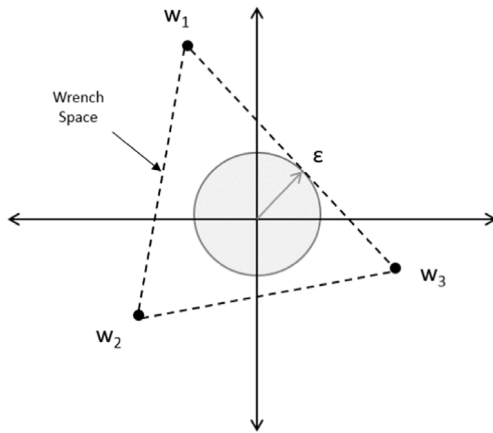


Fig. 10. Bidimensional example of the grasp quality [35] for three contact points.

to the quality metric value. This method is more efficient than others in the way that it does not require computation of the Quickhull algorithm. Besides, there is no need for approximating the friction cone hence it provides more accurate results. Another widely used quality measure that is similar to Ferrari and Canny [35] is presented by Mirtich and Canny [40]. They propose a method that first collects the grasp sets with the best resistance to disturbance forces and from these selects those which resist external torques better.

While the studies reviewed so far are considering contact wrenches, some researchers focused on introducing quality metrics based on the geometric properties of the workpiece. Chinellato et al. [41] propose a metric based on the area of the grasp polygon formed by the contact points on the workpiece. Chinellato et al. propose that maximizing the area of the triangle formed by three contact points on the workpiece increases the stability of the grasp.

Minimizing the effects of internal and gravitational forces is another essential factor in obtaining a stable grasp. Thus, the distance between the center of mass of the object and the centroid of the polyhedron formed by the contact points has been proposed as grasp quality metric [42–44]. On one hand, the geometric based metrics are computationally cheap and simple to implement. On the other hand, they all suffer from a bold assumption that an unlimited amount of force can be applied by the fingers. In other words, the fingers may have to exert extremely large forces for resisting small disturbances, which is not realistic.

**Pros:** The proposed *general metrics* are generic and can be implemented to any robot application.

**Cons:** Since these approaches do not take the task wrenches into account, they cannot very accurately predict the grasp quality in reality.

#### Task-oriented Metrics:

The quality measures presented so far are task-independent, in which evenly distributed wrenches in all directions are assumed. In many assembly tasks, a task-oriented grasp quality measure can be utilized to find grasps that are most suitable to fulfill that specific task's requirements. A task can be defined as a set of wrenches that should be exerted on the workpiece to accomplish a given objective or a set of expected manipulation disturbance wrenches that the object should resist. These wrenches form the polytope formed by these wrenches is called *task wrench space* (TWS). Li and Sastry [45] present the TWS as a 6D ellipsoid. This representation helps to find grasps that are more suitable for that specific task. Fig. 11 schematically illustrates the difference between the general ball metric [35] and the task-oriented ellipsoid [45] for a given grasp wrenches ( $w_1 - w_4$ ) and task wrenches ( $tw_1 - tw_6$ ). As the general ball metric does not take the task wrenches into account, it considers the exemplified grasp set as a low quality grasp while the ellipsoid task-oriented metric presents the grasp as a better grasp for the task.

A popular task-oriented grasp metric is to choose an appropriate TWS and then measure how well it can be inscribed in a grasp wrench space (GWS) [21,45,46]. Among the studies done in this area, just a few have considered the task information in the grasp synthesis and grasp analysis due to difficulty of modeling the task. In reality, high resolution sensors are required to measure the TWS by help of human demonstrations [47] which is not applicable in industries yet.

**Pros:** These metrics accurately measure the grasp quality when information regarding robot task is available.

**Cons:** The proposed metrics require many manual inputs that inconvenience finger design automation.

#### Uncertainty-based Metrics:

None of the methods presented so far have considered unavoidable errors in manipulation. In reality, there are three major positioning errors that cause uncertainties on the contact point location. First, error on the contact surface due to workpiece modeling errors. Second, error on the object location. The last but not the least, error on the gripper robot and gripper kinematic and dynamic model. Bicchi and Kumar [13] highlight a gap in approaches toward grasp analysis that can handle positioning errors.

Balan and Bone [19] propose a metric that measures the sensitivity of a grasp to the positional error between the grasp locations and the actual finger locations. In their method, a circular area of expected error is defined at each contact point. The area of the intersection of the error circle and the contact surface provide the quality measure. The grasps with small intersection areas are more likely to fail in practice. A recent study by Caldas et al. [48] proposes a new quality metric that considers the errors on workpiece geometry. They define a new wrench space, called Reachable Wrench

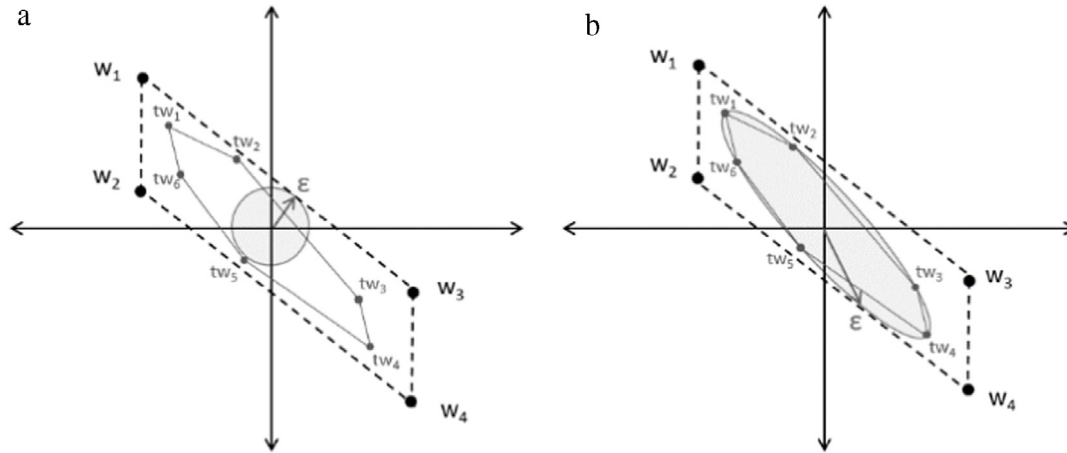


Fig. 11. Bidimensional example of (a) classic general quality metric (b) task-oriented quality metric.

Space under Uncertainties (RWSU), containing all the possible wrenches that the grasp can exert regardless of uncertainties. Two algorithms are provided to estimate the upper and lower bounds of the RWSU. They illustrate that the combination of these algorithms can provide a very good approximation of the RWSU.

**Pros:** These metrics are able to predict the quality of robot grasp in reality.

**Cons:** Similar to general metrics, these metrics do not take the task wrenches into consideration. Thus, they cannot accurately predict the grasp quality in reality.

The metrics that have been proposed to the grasp analysis research field have either limitations on the application area or defined for a specific criteria. As a result, some researchers [19,49] have utilized several metrics, both in parallel and series, in their algorithm to ensure the accuracy of the quality measurement of the grasp.

Some researchers have developed simulators for grasp synthesis and grasp analysis that are freely available to the public. Most popular open-source simulators are GraspIt! [50], OpenGRASP [51] and SynGrasp [52]. GraspIt! is the result of Millers Ph.D. thesis on developing a grasping simulator for robot hands [53]. In this free access tool, the user can create, simulate and analyze grasps of a given 3D model of the object and an articulated hand/gripper model. The system allows the user to test the quality of a grasp configuration and compare it with other grasps. Their novel visualization method helps the user to understand the weak points and strong points of a grasp. The system uses the Coulomb friction model and the grasp quality metric presented by Ferrari and Canny [35] to measure the quality of grasps. The detailed description of the analyzing method can be found in [28,50].

Another existing and publicly available software framework is OpenGRASP which is a result of the European Commission founded project GRASP [51]. This simulation toolkit is an improved version of OpenRAVE [54] with focus on robot grasping and manipulation. The main design principles of OpenGRASP are extensibility, interoperability and public availability [55]. León et al. [51] have proven the ability of their own developed tool, OpenGRASP, in grasping known and unknown objects. These open-source simulators help the research community to use them as references for evaluating and comparing their own approaches.

## 2.2. Finger design

By knowing the suitable grasp locations from grasp synthesis and grasp analysis processes, the next step in finger design automation is defining a method that automatically designs the

structure of the fingers. While much work has been done on design automation, little research has been focused on automation of finger design. Most of the finger design studies in the literature do not have the design automation capability. The studies on finger design automation can be divided into three categories: modular finger design, re-configurable fingers design and customized finger design (see Fig. 12).

### Modular Designs

In most modular design approaches [56,57], the shape of the workpiece is initially simplified to basic geometries (e.g. cylinder, sphere, cube, etc.) and then, from a finger library, suitable pairs of fingers are selected to grasp the workpiece [57]. However, Sanfilippo et al. [58] introduce a different technique. They propose a modular finger design to find a trade-off between a simple gripper and complex human like hands. Their work focuses on a modular grasping device that adapts itself to the characteristics of the workpiece. They use a one DoF mechanism as the basic element for the modular fingers of the gripper. An iterative procedure is proposed to obtain the minimum amount of possible modules that are required to execute the task requirement. The designed gripper can potentially have multi fingers, similar to a human hand.

**Pros:** The proposed algorithms are general and flexible so that they can be utilized for designing gripper fingers regardless of the workpiece shape complexity.

**Cons:** The methods go through all the finger design procedures to verify each finger design iteration. Thus, these approaches are computationally expensive.

Fig. 13 represents the flowchart of an example of modular finger design approaches.

### Re-configurable Design

Some researchers [18,19,59] approach gripper designs with the possibility of re-configuring fingers (pins) locations. The work of [19] is focused on three-finger parallel jaw grippers. The fingers configuration is defined to have two fixed cylindrical fingers in one jaw and the third finger is fixed on the perpendicular bisector of the centerline on the other fingers. The pair of fingers on the same jaw are the variables for changing configuration and the single finger is constantly fixed. The developed algorithm automatically obtains the fingers configurations.

**Pros:** These approaches provide quick and simple solutions for industrial applications.

**Cons:** The proposed algorithms require many technical inputs which cause only experts in this area be able to use the methods.

The flowchart of an example of re-configurable finger design approaches is demonstrated in Fig. 14.

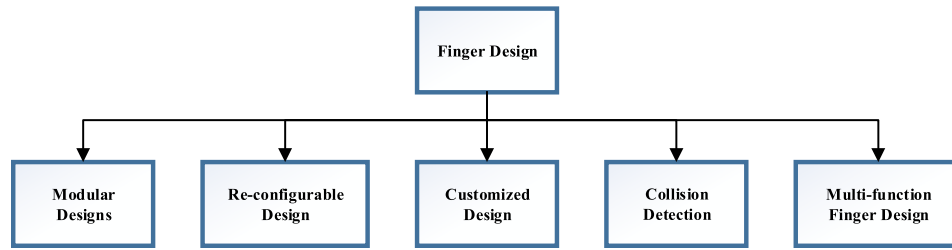


Fig. 12. Finger design categorization.

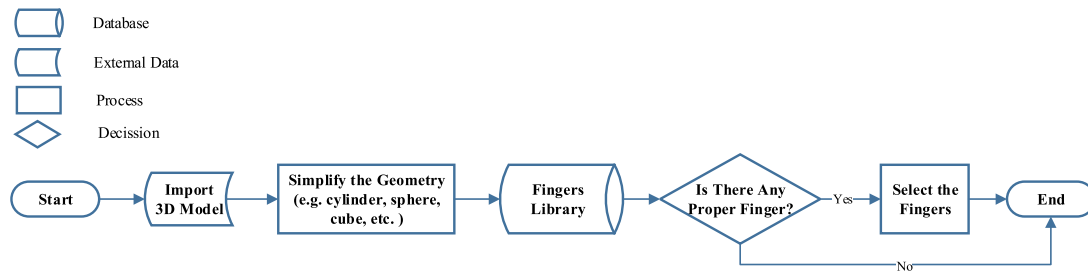


Fig. 13. Flowchart of an example of modular finger design.

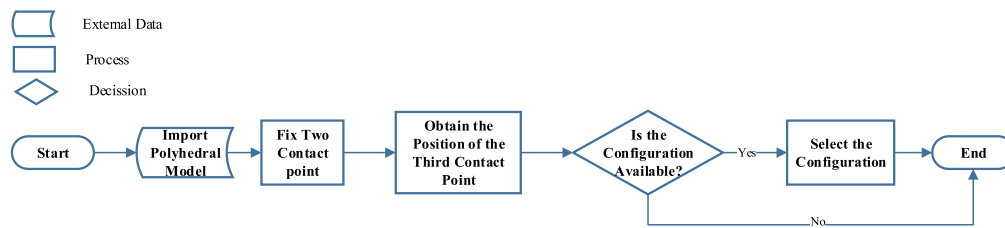


Fig. 14. Flowchart of an example of re-configurable finger design.

### Customized Design

Unlike modular and reconfigurable designs using general fingers for grasping an object, some studies design customized fingers for each workpiece [4,56,60,61]. Velasco and Newman [4] propose a quick technique for automated design and production of parallel-jaw gripper fingers. In support of agile manufacturing demands, they consider rapid prototyping in their approach for designing fingers. The proposed method assumes the fingers to be solid uniform volumes. By subtracting the geometry of the workpiece from the fingers block, they try to achieve a customized finger design. By this approach, the workpiece will be constrained in almost fully enveloped (form-closure) grasp.

Recently a commercial web-based automated finger design tool called *eGrip* is launched by SCHUNK [62] that uses a similar approach as [4]. The method implemented in the tool is patented [63]. In the tool, the user first imports the workpiece CAD model (STereolithography (STL)) and then selects a gripper from SCHUNK's gripper library. Then the planned grasp approach is required to be defined by the user. By providing these inputs, the tool generates fingers using form-closure grasp. This method suffers the same problem as [4] and cannot guarantee form-closure grasp solution for every object.

Pedrazzoli et al. [56] present a method which is the combination of modular and customized designs. To design fingers for a parallel-jaw gripper, the method begins by approximating the workpiece to simple geometries, such as sphere, cylinder and cube. Then it selects a set of fingers for approximated geometry from a finger library. In the last step, selected fingers are customized to match the dimension of the workpiece such that fingertips fit the objects surface contour.

**Pros:** These methods are simple, quick and require few inputs from the user.

**Cons:** These approaches do not provide a solution for every shape, e.g. axisymmetric shapes.

Fig. 15 shows the flowchart of an example of customized finger design approaches.

#### 2.2.1. Collision detection

The key process after designing the fingers structures is collision detection. This process is crucial to ensure the designed fingers would not collide with the workpiece while approaching for grasping. In other words, this process checks the accessibility of the chosen grasp locations.

In the last decades, many collision detection modules have been developed for various applications. RAPID [64], I-COLLIDE [65] and V-COLLIDE [66] are some of collision detection modules freely available to public. Among these, V-COLLIDE is more widely used in robotic research [50]. However, some researchers developed their own algorithms for detecting collision. Smith et al. [49] propose an algorithm for 2D collision detection for parallel-jaw grippers. Balan and Bone [19] extend [49] for 3D objects. Their proposed method starts by projecting the workpiece edges onto the gripper access plane. Then using shooting-ray method to simulate the fingers closing motions. If any intersection between the rays and edges occurs, the grasp is considered as inaccessible. Despite the fact that the method is quick and simple to implement, it is suitable only for very simple polyhedral objects and has difficulty handling complex shapes.

Miller [53] utilizes a bounding volume hierarchy (BVH) based method, called Proximity Query Package (PQP) [67], for detecting



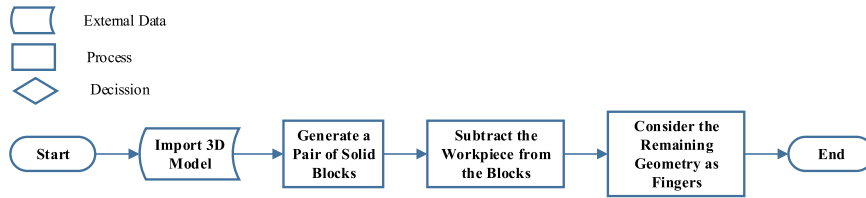


Fig. 15. Flowchart of an example of customized finger design.

collision of manipulating objects. This collision detection module starts with approximating the object surfaces to triangular facets. The collection of the facets for each object are imported to the PQP algorithm to check for collision. In PQP, each object is modeled as a hierarchy tree (called BVH) so that each leaf represents a facet. BVH trees are used as parents to generate a tree that consists all possible combinations of facets of objects. The latter tree is called bounding volume traversal tree (BVTT) [67]. The algorithm starts to check the interference for each facets combination (leaf of BVTT) so it detects which facets of the object have interference. In this method, like many other graphic-based methods, there is a trade-off between accuracy and the speed of the detection operation.

Saut and Sidobre [68] propose a method for detection collision for multi-finger robot hands. Their algorithm first discretizes the workspace of fingers to a point cloud. Then the point cloud is converted to a volumetric approximation as a set of spheres. In the last step, the algorithm checks the collision of each sphere with the workpiece, the hand (or gripper) and other fingers. This study presents a fast collision detection method yet it does not provide information regarding exact collision location.

Today, most commercial CAD software has in-built collision detecting features which enhance the collecting information regarding the location of the collision [69,70]. Brown and Brost [18] use such features to detect collisions in their 3D modular finger design tool.

### 2.2.2. Multi-function (Multipurpose) finger design

In assembly applications, multi-function fingers play a very significant role in decreasing tool-change operation and consequently increasing throughput of the workcell. Designing the gripper fingers in such a way that they can grasp more than one part can enormously reduce the cycle time for executing an assembly task when handling various object shapes [1]. Therefore considering this information in finger design is essential. However, designing multi-functional fingers demands a high level of skill and experience in finger design area. Therefore, adding this functionality to the automated finger design tool can extremely increase its value of it. Unfortunately, few researchers mention the importance of multi-functional fingers in their work and only a handful of them propose methods for design automation of such fingers.

Pham et al. [57] propose a trial and error based method in their design algorithm. Their algorithm starts to design a set of fingers for the workpiece A. Then the algorithm checks whether the workpiece B can be grasped with the designed fingers. If it can, there is no need to design a new set of fingers. Otherwise the same design process is applied to workpiece B and the designed fingers are checked if they can grasp the workpiece A or not. If not, a separate set of fingers is required for each workpiece. While the proposed method is simple to implement, it is only applicable to geometries having similar shape.

Velasco and Newman [4] propose two methods for automatically designing multi-functional fingers. The first method uses the geometric superposition of the workpieces to design gripper fingers. The second method translates and/or rotates one or more of the workpieces in different orientations, then uses the first method to check for a solution. As the authors also mention, the first

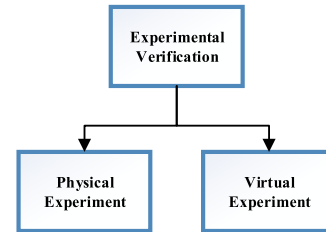


Fig. 16. Experiment categorization.

method fails for cases that workpieces with similar geometry but different dimensions, for instance two spheres with different radii. Similarly, the second method does not guarantee a feasible solution as it approaches random translation and rotation of workpieces.

### 2.3. Experimental verification

The last key process in the finger design procedure is executing experiments to verify the proposed methods. This process is essential in finger design automation to ensure the fingers can accomplish the task. As shown in Fig. 16, experimental verifications presented in the finger design and grasp literature can be divided into *virtual* and *physical* groups.

#### 2.3.1. Virtual experiment

Virtual experiments are done using virtual workpiece models instead of physical ones. This type of experiment is very popular in literature due to its simplicity and low cost. Using publicly available grasp simulators have helped many researchers to verify their approach virtually [47,71]. Studies using virtual experiments usually verified their methods on examples [19,33,44,48,72–74] and only a few research compared their proposed approach with others [9,38,39]. As mentioned in Section 3.1.1, the main reason of limited comparison in literature is the reluctance of the authors to publish the implemented methods.

#### 2.3.2. Physical experiment

Unlike the virtual experiments, very few studies have physically verified their approaches. The main reason is that not every researcher has access to the expensive robotic hardware (e.g. robot, gripper, sensor, etc.). Besides, setting up a physical verification system requires experimental skills. For instance, choosing an experiment task that is not reliable makes it difficult to figure out whether an error that occurred was due to a system problem or task problem. In general, physical experiments are crucial to test an approach in reality. Recently, several works [32,75,76] demonstrate that classic grasp quality metrics do not have a good prediction capability in reality. Balasubramanian et al. [75] experimentally validate the grasps considered stable by virtual grasp synthesizers (e.g. Graspl!). Their results show around 30% of the virtually approved grasps failed in physical experiments.

Kim et al. [32] also illustrate the importance of physical experiments. They physically test the widely used grasp quality measure presented by Ferrari and Canny [35] and the results show the metric poorly evaluates planned grasps.

While physical experiments are essential in the finger design process, most of the studies that present physical verifications in their work have not provided detailed information regarding the process of the experiment [3,4,18,32,77,78]. Therefore, it is not possible to repeat their experiments and consequently to compare them. Among the works reviewed in this paper, only [1,53] give the details of their experiments. Some works [47,71,79] verified their method by both virtual and physical experiments. Only [79] provide the virtual experiment platform available to the public.

### 3. Discussion

In this section, a quick review on strengths and weaknesses of approaches proposed in each process (see Fig. 2) is presented. Unfortunately, a fair quantitative comparison between different approaches is to date not possible because of two major reasons. First, due to the absence of a mainstream method or group of criteria for verification of proposed approaches, quantitative results presented by researchers are not comparable as they are validated by different methods and criteria. Second reason is the lack of benchmarks and the permanent inability to distinguish different solutions as most authors do not make their implementations available to public [9]. As a result, approaches presented in this paper are qualitatively compared. By considering the main purpose of the finger design automation procedure, the most efficient methods with respect to the defined criteria are pointed out. A combination of the selected approaches should potentially aid the user to automatically design fully functional gripper fingers.

#### 3.1. Discussion on grasp

The methods proposed for grasp synthesis and grasp analysis are discussed in this section and advantages and disadvantages are highlighted.

##### Grasp Synthesis

The benchmark of the following approaches are based on four criteria:

- The first criterion takes the flexibility of the approaches on taking number of contact points into account. Approaches that use a limited number of contacts are considered to have low flexibility and approaches with high flexibility are applicable on any number on contacts.
- The second criterion represents the flexibility of the approaches on handling workpiece geometry. Approaches with low flexibility can only be implemented on polyhedral geometries and approaches with high flexibility are applicable to any geometry.
- The third criterion qualitatively benchmarks the computational effort required for each approach. Determining the convex hull is computationally expensive as well as searching for global optima. Therefore, approaches that utilize both convex hull and global optima require high computational effort. Approaches that use one of them are considered to have medium computational efficiency and the approaches that do not use any of these methods are highly efficient.
- The last criterion considers the capability of the approaches in the context of automation. Approaches are compared according to the amount of required input. Approaches with high capability only require the 3D model or 3D vision of the workpiece while approaches with low capability require more inputs which are usually manually inputs to the systems.

**Limited Contact Methods for Polyhedral Objects:** These approaches consider the minimum number of required contact points to verify closure of objects with polyhedral geometry [17–20]. In the proposed analytical approaches, methods for limited number of contact points need medium computational effort due to their global optima searching method and are highly suitable for automation. However, they are limited to 2 or 3 contact points and can only be implemented on polyhedral objects. Thus, these methods are suitable for simple pick and place operations with small disturbance wrenches.

**Unlimited Contact Methods for Polyhedral Objects:** Methods for handling unlimited number of contacts which overcome the limitation of number of contact points by determining convex hull of wrenches [21,23,24]. While these approaches have high flexibility on the number of contacts and automation capability, they are computationally expensive and applicable only to polyhedral objects.

**Heuristic Methods:** These methods tackle the global optima problem by randomly selecting possible grasp contact points [25,27–29]. These methods reduce the computational searching effort, but require convex hull computation. Therefore heuristic methods have high flexibility on the number of contacts and automation capability. However, they demand medium computational effort and are applicable only to polyhedral objects.

**Methods for General Object Shape:** Methods that model the workpiece as a point cloud or triangular meshes in order to get the better of the limitation problem of heuristic methods on polygonal objects [9,30,31]. These methods require medium computational effort to provide high flexibility on number of contacts, object geometry and automation capability.

**Data-driven Approaches:** Despite the fact that the analytical approaches rely on complex computation of geometric, kinematic and dynamic formulations, data-driven approaches are introduced to reduce computational effort by sampling possible grasps, utilizing simulation methods or human demonstrations [16,32,33]. These methods are computationally inexpensive and applicable to unlimited numbers of contacts and workpieces with any shape. In spite of this, they suffer from complexity of automation. These methods require operation (task) procedure simulations that are mostly manual procedures. In order to automate this method, a huge computational effort is required to recognize the assembly process and to simulate every task. This cancels out the main advantage of these methods in comparison to analytical approaches. Therefore the efficiency of these approaches drops dramatically.

Based on the benchmark of methods presented in Table 1, *methods for general object shape* are the most appropriate approaches for finger design automation. Among the grasp synthesis methods for general object shape presented, Liu and Carpin [9] method is the most suitable for finger design automation due to the following reasons:

- This method can be combined with any metric to measure the quality of grasp so it can be easily integrated with any grasp analyzer.
- The method can be easily integrated with commercial CAD software that is accessible to most design engineers and manufacturers. This advantage reduces the number of required tools and ease the data transfer.
- The implemented method is freely available which has a huge benefit in compare to other methods as it ensures the practicability of the method.

##### Grasp Analysis

To measure the quality of a grasp, three groups of metrics are presented in this paper.

**Table 1**  
Qualitative benchmark of grasp synthesis approaches.

		Flexibility on number of contacts (fingers)	Flexibility of the object geometry	Computational efficiency	Automation & integration possibilities
Analytical approaches	Limited contact methods for polyhedral objects	Low	Low	Medium	High
	Unlimited contact methods for polyhedral objects	High	Low	Low	High
	Heuristic methods	High	Low	Medium	High
	Methods for general object shape	High	High	Medium	High
Data-driven approaches	Simulation-based methods	High	High	High	Low

**General Metrics:** Metrics that measure the grasps quality from a global perspective and does not take the robot task into account [35,38–41]. These metrics are easy to implement and suitable for both simple pick and place applications and more complicated tasks.

**Task-oriented Metrics:** Unlike the general metrics, these metrics measure the quality of grasps based on the specific task that is defined for the robot to execute [21,45–47]. These metrics provide more accurate measurements for comparing possible grasp sets. However, automatically modeling the tasks require tremendous amount of effort and computation.

**Uncertainty-based Metrics:** This group of metrics considers the uncertainties when measuring the quality of grasps [19,48]. These metrics predict the stability of grasps much closer to the reality in comparison to others.

Since each group of metrics takes a specific and individualistic measurement factor into consideration, benchmarking them is neither realistic nor appropriate. To improve the accuracy of the grasp quality measures, combining the metrics is the best solution for an automated finger design system. This enables the user to select metrics based on the available task information.

### 3.2. Discussion on finger design

This section benchmarks the finger design approaches using four criteria.

- *Design generalness* which compares the methods based on the closure types (i.e. force and form closures) that designed fingers can achieve. Design approaches that can achieve both force-closure and form-closure grasps are considered more generic than the methods obtaining only one type of closure.
- *Design flexibility* which takes the ability of the design method on handling objects with complex shapes. Methods that are applicable to simple geometries, polyhedral and complex geometry are respectively considered to have low, medium and high design flexibility.
- *Design reliability* that benchmarks the ability of the methods to design fingers that fit the surface of the object. This ability compensates the position uncertainties of the workpiece and increases the contact and friction area. As a consequence, most reliable designed fingers fully match the contact surface contour of the workpiece regardless of the shape complexity and are of high design reliability. Methods that design fingers to fit workpiece surface contour with simple shapes are considered to have medium design reliability. The methods that do not take the contour of the workpiece contact surface into account have low design reliability.
- *Cost efficiency* which compares the overall cost of finger design methods solutions for accomplishing a task. Obviously, the lower the cost of the design solution, the more efficient the method.

The studies done on finger design are categorized into modular designs, reconfigurable designs and customized designs.

**Modular design-based approaches** usually simplify the shape of the workpiece to basic geometries (e.g. cylinder, sphere, cube and etc.) and then, from a finger library, suitable pairs of fingers are selected [56–58]. While the modular design-based methods do not require fingers to be produced for each workpiece, they need an initial investment on producing libraries of fingers. Therefore they are considered to be medium in cost efficiency. These methods have difficulty handling complex objects and can achieve only force-closure, so they respectively have low design flexibility.

**Re-configurable design approaches** are based on algorithms that automatically find a configuration of fingers (cylindrical pins) locations to grasp the workpiece and accomplish the task [18,19,59]. Despite the fact that these methods are the most cost efficient, they have medium design flexibility due to handling only polyhedral objects. Besides, the re-configurable methods have low design genericity and reliability due to their ability to achieve only force-closure and not considering the workpiece surface contour in design process.

**Customized design approaches**, different from the previous ones, provide a specific solution for each workpiece [4,60–62]. These methods are considered highly generic due to their capability to achieve both form and force closures. However, none of the existing customized design methods have the ability of achieving both form and force closure. The customized design methods can also handle objects with any shape complexity and the designed fingers fully match the shape of the workpiece. Thus, the customized design methods have high design flexibility and reliability. However, designing and producing separate fingers for every workpiece results in low cost-efficiency for these methods.

**Table 2** visualizes the benchmark of finger design methods. Based on this table, customized design methods are deemed most suitable for finger design automation. Among these methods, Velasco and Newman [4] approach has the advantage that can be integrated with the grasp synthesizer and analyzer in a CAD software. While the method designs fingers to fully encompass the workpiece and does not rely on friction, this is often infeasible in reality. For instance, this method is unable to provide solutions with frictionless contacts that fully fix the position of axisymmetric workpieces in space. As a result, the research community needs to present a new method that considers both form and force closure for designing gripper fingers.

#### Collision Detection

Many collision detection methods have been proposed for a vast range of applications [49,53,64–66,68]. Some collision detection algorithms are free as modules. While many studies have used these open-source modules, a number of researchers developed their own algorithms that are more or less impossible to verify. Another option to detect a collision is utilizing built-in collision detectors in commercial CAD software. The latter method is the most suitable one for finger design automation for two main reasons. First, it can be integrated with grasp synthesis, grasp analysis and finger design in a CAD tool. Second, by utilizing CAD, more details on the

**Table 2**  
Qualitative benchmark of finger design approaches.

	Design genericity	Cost efficiency	Design flexibility	Design reliability
Modular designs	Low	Medium	Low	Medium
Re-configurable design	Low	High	Medium	Low
Customized design	High	Low	High	High

collision location can be extracted which can be helpful in grasp synthesizing and analyzing.

#### Multi-function Finger Design

The review in the literature illustrates that the significance of multi-functional design of fingers has been missed out in the finger design research area. Except for a few research results that consider this issue in their design process [1,4,57], no remarkable study has been able to propose a practical method for multi-function fingers design.

### 3.3. Discussion on experimental verification

#### Virtual Experiment

Publicly available grasp simulators have enhanced the finger design research area so the user can fully simulate and control the workcell environment and its attributes. The users have the capability of performing a large number of experiments without having access to costly robotic hardware. While the currently available virtual experiments are easy to carry out, studies have shown that the results of virtual experiments noticeably differ from physical experiments. As a result these experiments should only be used for preliminary and conceptual checks on the feasibility of the designed fingers.

#### Physical Experiment

In manufacturing industries, functional gripper fingers play a significant role on overall performance of the robot cell. Therefore, the physical experiments are crucial to examine the practicability of the designed fingers in reality. However, the gripper fingers design research area lacks a general experiment method for physically examining the designed fingers.

## 4. Conclusion

This paper reviews the enabler studies for finger design automation and the key processes to achieve this goal.

An exhaustive literature review is carried out by filtering down thousands of articles to a manageable number of articles. These articles are used to categorize the key processes in finger design automation, i.e. *grasp*, *finger design* and *experimental verification*. Furthermore, each key process is divided into sub-categories and the most promising method for each category is identified. Extraction of this review and the results are presented as follows.

- **Grasp:** Robotic grasp can be divided into two wide research areas of grasp synthesis and grasp analysis.
  - **Grasp Synthesis:** The proposed methods for grasp synthesis are categorized as analytical and data-driven approaches. According to the results, analytical approaches are more appropriate for finger design automation than data-driven.  
*Suggested method:* Along with analytical approaches, the method proposed by Liu and Carpin [9] is suitable for implementing on finger design automation.
  - **Grasp Analysis:** The proposed analyzing methods for measuring the quality of possible grasps are grouped as general metric, task-oriented metrics and uncertainty-based metrics.

*Suggested method:* Since each grasp analysis group measures an independent quality factor, the combination of metrics is the best way in order to ensure more accurate grasp quality prediction.

- **Finger Design:** Limited research is done in gripper finger design. The section is divided to modular design, re-configurable design and customized design. Among these approaches, customized design methods are considered most practical for automation of the finger design procedure.

*Suggested method:* The customized design method presented by Velasco and Newman [4] is suggested for finger design automation due to ability of integration with CAD tools.

- **Collision Detection:** Many collision detection tools and algorithms have been generated in the literatures to fulfill various purposes that can be implemented in finger design automation.

*Suggested method:* The most relevant approach for finger design automation is utilizing built-in collision detection tools in CAD software.

- **Multi-function Finger Design:** It is shown that designing multi-function fingers may remarkably increase the workcell performance yet it has not been studied properly due to the inherent complexities of this task.  
*Suggested method:* While handful researches have considered the significance of multi-function finger design, the literature lacks a practical method to tackle this issue.

- **Experimental Verification:** Verifying the designed fingers by executing experiments are crucial in the automated finger design procedure. Virtual and physical experiment are two methods to examine the functionality of the fingers.

- **Virtual Experiment:** This type of experiment is based on simulating the robot operation in a virtual environment. Virtual experiments are popular among researchers due to its controllability, simplicity and low cost.

*Suggested method:* Public available simulators are the best option for performing virtual experiments because of their large gripper libraries, flexibility on designing environment and simplicity to integrate with other algorithms.

- **Physical Experiment:** Even though the virtual experiments are simple and cheap, studies shown that they do not cover all unknowns that are unfolded in real experiment. Therefore, physical experiments are essential to ensure the practicability of the designed fingers.

*Suggested method:* While the experimental methods plays a significant role in finger design automation, no research study presents a concrete method for executing physical experiments.

According to the results of this review, the handful developed methods for finger design automation suffer from two substantial problems. Some methods are limited to handling polyhedral workpieces and inapplicable for complex geometries. Others miss out some key processes in their design procedure which leads into uncertainty in practicability of the designed fingers.



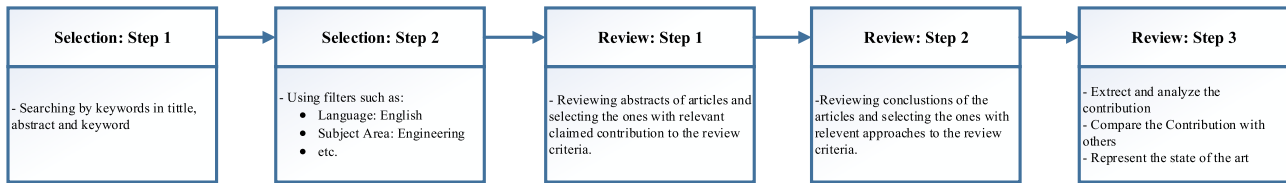


Fig. A.1. Flowchart of the sequential review process.

Table A.1

Details of the review process.

Key processes	Review process							
	Selection: Step 1			Section: Step 2		Review: Step 1	Review: Step 2	Review: Step 3
	Source	Keywords	Results	Filter	Results	Results	Results	Results
Grasp synthesis	Google Scholar SCOPUS	grasp synthesis OR plan* method	1180	Language : English Subject Area: Engineering Publish Year: 2012–2015	230	84	57	22
Grasp analysis	Google Scholar SCOPUS	grasp analysis method	1368	Language : English Subject Area: Engineering Publish Year: 2013–2015	537	74	51	20
Finger design	Google Scholar SCOPUS	finger OR jaw OR gripper design	978	Language : English Subject Area: Engineering	565	63	31	8
Multi-function finger design	Google Scholar SCOPUS	multi part OR purpose OR function grasp OR finger OR jaw	229	Language : English Subject Area: Engineering	103	26	14	1
Collision detection	Google Scholar SCOPUS	robot* collision detect* method	749	Language : English Subject Area: Engineering	461	63	37	9
Experiment	Google Scholar SCOPUS	robot* experiment method	185	Language : English Subject Area: Engineering	126	18	7	2

By connecting all key processes for finger design automation presented in this review, one can quantify the capabilities of automated finger design in compare to other related finger design approaches.

As a result, finger design automation literature lacks, besides the points already made above, a study that benchmarks the performance of the fingers designed by considering all key processes with fingers design by existing approaches that partially consider key processes.

## Acknowledgments

The research leading to these results has received funding from the European Community's Framework Programme Horizon 2020 – under Grant Agreement No. 644938 – SARAfun. Emelie Jin, Jonas Larsson and Daniel Sirkett at ABB are gratefully acknowledged for technical discussions and various supports.

## Appendix. Review method

In this appendix, the review process is presented with the goal of systematically selecting and reviewing the published papers in the finger design research area. The purpose of this paper is design of end-effectors for pick & place and assembly operations of industrial robots. Therefore robot hands are excluded. As illustrated in Fig. A.1, the review process in this work consists of three parts, a pre-review phase, a selection phase and a review phase.

### A.1. Pre-reviewing phase

In order to be able to perform a structured review, the various fields within finger design automation need to be quantified.

Therefore an unstructured review is done mainly on high citation papers in this field. The result of this pre-review phase is the quantified fields presented in Fig. 2. These fields are then individually reviewed as described in the next section.

### A.2. Selection phase

The following review process is implemented on each of the key process steps shown in Fig. 2, i.e. *grasp*, *finger design* and *experimental verification*.

In the selection phase, major databases in this research field (e.g. SCOPUS and Google Scholar) are utilized to search for relevant articles. This phase consists of the following steps.

**Step 1:** Hot keywords for each topic are used to search in titles, abstracts and keywords. Duplicated papers are eliminated from the results, e.g. only the latest paper covering a specific field, presented by a specific research group is taken into consideration.

**Step 2:** The results are limited to English language articles within the subject area of engineering. For example the word *finger* is widely used in *medicine* and *neuroscience* subject areas which are not relevant to the sought after research area of this paper.

For grasp analysis and synthesis processes, only the papers published between 2012 and 2015 are included in the selection and review phases of this paper. Papers before 2012 have been extensively reviewed in numerous review publications of which the results are taken into consideration in this paper.

### A.3. Review phase

In this stage, the results from the selection phase are reviewed and critically analyzed. The selected papers are classified as *review papers* and *methodology papers* based on their contribution types. The criteria for review papers are:

- Review of the *key processes* research fields
- Review must be relevant to finger design automation.

The criteria for methodology papers are:

- New methodology is proposed that is relevant to the key processes
- Improvements of a previously proposed methodology.

In order to minimize the reviewing time, the process is divided into three steps.

**Step 1:** The abstract of each article from the selection phase is reviewed and the papers with claimed contributions relevant to the review criteria are collected.

**Step 2:** Conclusions of the results from the first step are reviewed. The papers with proposed approaches relevant to review criteria are collected.

**Step 3:** The entire paper is reviewed. The claimed contributions are verified and approaches of each research are analyzed and compared with similar contributions. The result of this stage is the state of the art in the research field and is presented in the following section.

Implementing this literature review process aided 4689 identified papers in step 1 of the selection phase to be filtered down to 62 reviewed papers which are reviewed. Detail of each phase implementation is presented in [Table A.1](#).

## References

- [1] G.C. Causey, Elements of agility in manufacturing, 1999.
- [2] T.R. Kurfess, *Robotics and Automation Handbook*, CRC Press, 2004.
- [3] M.T. Zhang, K. Goldberg, Designing robot grippers: optimal edge contacts for part alignment, *Robotica* 25 (2007) 341. <http://dx.doi.org/10.1017/S0263574706003134>.
- [4] J. Velasco, V.B., W.S. Newman, Computer-assisted gripper and fixture customization using rapid-prototyping technology, in: 1998 IEEE International Conference on Robotics and Automation, 1998. Proceedings. Presented at the 1998 IEEE International Conference on Robotics and Automation, 1998. Proceedings, vol. 4, 1998, pp. 3658–3664. <http://dx.doi.org/10.1109/ROBOT.1998.681393>.
- [5] V.-D. Nguyen, Constructing force-closure grasps, in: 1986 IEEE International Conference on Robotics and Automation. Proceedings. Presented at the 1986 IEEE International Conference on Robotics and Automation. Proceedings, 1986, pp. 1368–1373. <http://dx.doi.org/10.1109/ROBOT.1986.1087483>.
- [6] F. Reuleaux, *The Kinematics of Machinery: Outlines of a Theory of Machines*, Macmillan and Company, 1876.
- [7] J.K. Salisbury, Kinematic and Force Analysis of Articulate Hands, 1985.
- [8] A. Bicchi, On the closure properties of robotic grasping, *Int. J. Robot. Res.* 14 (1995) 319–334. <http://dx.doi.org/10.1177/027836499501400402>.
- [9] S. Liu, S. Carpin, Global grasp planning using triangular meshes, in: 2015 IEEE International Conference on Robotics and Automation, ICRA. Presented at the 2015 IEEE International Conference on Robotics and Automation, ICRA, 2015, pp. 4904–4910. <http://dx.doi.org/10.1109/ICRA.2015.7139880>.
- [10] E. Al-Gallaf, A. Allen, K. Warwick, A survey of multi-fingered robot hands: Issues and grasping achievements, *Mechatronics* 3 (1993) 465–491. [http://dx.doi.org/10.1016/0957-4158\(93\)90018-W](http://dx.doi.org/10.1016/0957-4158(93)90018-W).
- [11] K.B. Shimoga, Robot grasp synthesis algorithms: A survey, *Int. J. Robot. Res.* 15 (1996) 230–266. <http://dx.doi.org/10.1177/027836499601500302>.
- [12] J.I. Pons, R. Ceres, F. Pfeiffer, Multifingered dextrous robotics hand design and control: a review, *Robotica* (1999) 661–674.
- [13] A. Bicchi, V. Kumar, Robotic grasping and contact: a review, in: IEEE International Conference on Robotics and Automation, 2000. Proceedings. ICRA '00. Presented at the IEEE International Conference on Robotics and Automation, 2000. Proceedings. ICRA '00, vol. 1, 2000, pp. 348–353. <http://dx.doi.org/10.1109/ROBOT.2000.844081>.
- [14] S. Yao, Q. Zhan, M. Ceccarelli, G. Carbone, Z. Lu, Analysis and grasp strategy modeling for underactuated multi-fingered robot hand, in: International Conference on Mechatronics and Automation, 2009. ICMA 2009. Presented at the International Conference on Mechatronics and Automation, 2009. ICMA 2009, 2009, pp. 2817–2822. <http://dx.doi.org/10.1109/ICMA.2009.5246448>.
- [15] A. Sahbani, S. El-Khoury, P. Bidaud, An overview of 3D object grasp synthesis algorithms, *Robot. Auton. Syst., Autonomous Grasping* 60 (2012) 326–336. <http://dx.doi.org/10.1016/j.robot.2011.07.016>.
- [16] J. Bohg, A. Morales, T. Asfour, D. Kragic, Data-Driven Grasp Synthesis—A Survey, *ArXiv* 13092660, 2013.
- [17] V.-D. Nguyen, Constructing force-closure grasps in 3D, in: 1987 IEEE International Conference on Robotics and Automation. Proceedings. Presented at the 1987 IEEE International Conference on Robotics and Automation. Proceedings, 1987, pp. 240–245. <http://dx.doi.org/10.1109/ROBOT.1987.1088014>.
- [18] R.G. Brown, R.C. Brost, A 3D modular gripper design tool, in: 1997 IEEE International Conference on Robotics and Automation, 1997. Proceedings. Presented at the 1997 IEEE International Conference on Robotics and Automation, 1997. Proceedings, vol. 3, 1997, pp. 2332–2339. <http://dx.doi.org/10.1109/ROBOT.1997.619310>.
- [19] L. Balan, G.M. Bone, Automated gripper jaw design and grasp planning for sets of 3D objects, *J. Robot. Syst.* 20 (2003) 147–162. <http://dx.doi.org/10.1002/rob.10076>.
- [20] J.-W. Li, H. Liu, H. Cai, On computing three-finger force-closure grasps of 2-D and 3-D objects, *IEEE Trans. Robot. Autom.* 19 (2003) 155–161. <http://dx.doi.org/10.1109/TRA.2002.806774>.
- [21] L. Han, J.C. Trinkle, Z.X. Li, Grasp analysis as linear matrix inequality problems, *IEEE Trans. Robot. Autom.* 16 (2000) 663–674. <http://dx.doi.org/10.1109/70.897778>.
- [22] B. Mishra, J.T. Schwartz, M. Sharir, On the existence and synthesis of multifinger positive grips, *Algorithmica* 2 (1987) 541–558. <http://dx.doi.org/10.1007/BF01840373>.
- [23] Y.-H. Liu, M. Wang, Qualitative test and force optimization of 3D frictional force-closure grasps using linear programming, in: 1998 IEEE International Conference on Robotics and Automation, 1998. Proceedings. Presented at the 1998 IEEE International Conference on Robotics and Automation, 1998. Proceedings, vol. 4, 1998, pp. 3335–3340. <http://dx.doi.org/10.1109/ROBOT.1998.680953>.
- [24] Y.-H. Liu, Computing n-finger form-closure grasps on polygonal objects, *Int. J. Robot. Res.* 19 (2000) 149–158. <http://dx.doi.org/10.1177/02783640022066798>.
- [25] C. Borst, M. Fischer, G. Hirzinger, Grasping the dice by dicing the grasp, in: 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2003, IROS 2003. Proceedings. Presented at the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2003. (IROS 2003). Proceedings, vol. 3, 2003, pp. 3692–3697. <http://dx.doi.org/10.1109/IROS.2003.1249729>.
- [26] M. Fischer, G. Hirzinger, Fast planning of precision grasps for 3D objects, in: Proceedings of the 1997 IEEE/RSJ International Conference on Intelligent Robots and Systems, 1997. IROS '97. Presented at the, Proceedings of the 1997 IEEE/RSJ International Conference on Intelligent Robots and Systems, 1997. IROS'97, vol. 1, 1997, pp. 120–126. <http://dx.doi.org/10.1109/IROS.1997.649021>.
- [27] A.T. Miller, P.K. Allen, Examples of 3D grasp quality computations, in: 1999 IEEE International Conference on Robotics and Automation, 1999. Proceedings. Presented at the 1999 IEEE International Conference on Robotics and Automation, 1999. Proceedings, vol. 2, 1999, pp. 1240–1246. <http://dx.doi.org/10.1109/ROBOT.1999.772531>.
- [28] A.T. Miller, S. Knoop, H.I. Christensen, P.K. Allen, Automatic grasp planning using shape primitives, *IEEE* (2003) 1824–1829. <http://dx.doi.org/10.1109/ROBOT.2003.1241860>.
- [29] D. Ding, Y.-H. Liu, S. Wang, Computing 3-D optimal form-closure grasps, in: IEEE International Conference on Robotics and Automation, 2000. Proceedings. ICRA '00. Presented at the IEEE International Conference on Robotics and Automation, 2000. Proceedings. ICRA '00, vol. 4, 2000, pp. 3573–3578. <http://dx.doi.org/10.1109/ROBOT.2000.845288>.
- [30] D. Ding, Y.-H. Liu, M.Y. Wang, On computing immobilizing grasps of 3-D curved objects, in: Proceedings 2001 IEEE International Symposium on Computational Intelligence in Robotics and Automation, 2001. Presented at the Proceedings 2001 IEEE International Symposium on Computational Intelligence in Robotics and Automation, 2001, 2001, pp. 11–16. <http://dx.doi.org/10.1109/CIRA.2001.1013165>.
- [31] Y.-H. Liu, M. Lam, D. Ding, A complete and efficient algorithm for searching 3-D form-closure grasps in the discrete domain, *IEEE Trans. Robot.* 20 (2004) 805–816. <http://dx.doi.org/10.1109/TRO.2004.829500>.
- [32] J. Kim, K. Iwamoto, J.J. Kuffner, Y. Ota, N.S. Pollard, Physically based grasp quality evaluation under pose uncertainty, *Trans. Robot.* 29 (2013) 1424–1439. <http://dx.doi.org/10.1109/TRO.2013.2273846>.
- [33] A. Wolniakowski, K. Miatliuk, N. Krüger, J.A. Rytz, Automatic evaluation of task-focused parallel jaw gripper design, in: D. Brugali, J.A. Broenink, T. Kroeger, B.A. MacDonald (Eds.), *Simulation, Modeling, and Programming for Autonomous Robots*, in: Lecture Notes in Computer Science, Springer International Publishing, 2014, pp. 450–461.

- [34] M.A. Roa, R. Suárez, Grasp quality measures: review and performance, *Auton. Robots* 38 (2014) 65–88. <http://dx.doi.org/10.1007/s10514-014-9402-3>.
- [35] C. Ferrari, J. Canny, Planning optimal grasps, in: 1992 IEEE International Conference on Robotics and Automation, 1992. Proceedings. Presented at the, 1992 IEEE International Conference on Robotics and Automation, 1992. Proceedings, vol. 3, 1992, pp. 2290–2295. <http://dx.doi.org/10.1109/ROBOT.1992.219918>.
- [36] C.B. Barber, D.P. Dobkin, H. Huhdanpaa, The Quickhull algorithm for convex hulls, *ACM Trans. Math. Softw.* 22 (1996) 469–483. <http://dx.doi.org/10.1145/235815.235821>.
- [37] Qhull [WWW Document], 2015. [www.qhull.org](http://www.qhull.org). (Accessed 12 August 2015).
- [38] S. Liu, S. Carpin, Fast grasp quality evaluation with partial convex hull computation, *Proc. IEEE Int. Conf. Robot. Autom.* (2015).
- [39] Y. Zheng, An efficient algorithm for a grasp quality measure, *IEEE Trans. Robot.* 29 (2013) 579–585. <http://dx.doi.org/10.1109/TRO.2012.2222274>.
- [40] B. Mirtich, J. Canny, Easily computable optimum grasps in 2-D and 3-D, in: IEEE International Conference on Robotics and Automation, 1994, pp. 739–747.
- [41] E. Chinellato, R.B. Fisher, A. Morales, A.P. del Pobil, Ranking planar grasp configurations for a three-finger hand, in: IEEE International Conference on Robotics and Automation, 2003. Proceedings. ICRA '03. Presented at the IEEE International Conference on Robotics and Automation, 2003. Proceedings. ICRA '03, vol. 1, 2003, pp. 1133–1138. <http://dx.doi.org/10.1109/ROBOT.2003.1241745>.
- [42] S.S. Jean Ponce, On computing four-finger equilibrium and force-closure grasps of polyhedral objects, *Int. J. Robot. Res.* 16 (1995). <http://dx.doi.org/10.1177/027836499701600102>.
- [43] I. Kamon, T. Flash, S. Edelman, Learning to grasp using visual information, in: 1996 IEEE International Conference on Robotics and Automation, 1996. Proceedings. Presented at the, 1996 IEEE International Conference on Robotics and Automation, 1996. Proceedings, vol. 3, 1996, pp. 2470–2476. <http://dx.doi.org/10.1109/ROBOT.1996.506534>.
- [44] D. Ding, S. Wang, Computation of 3-D form-closure grasps, *IEEE Trans. Robot. Autom.* (2001) 515–522. <http://dx.doi.org/10.1109/70.954765>.
- [45] Z. Li, S.S. Sastry, Task-oriented optimal grasping by multifingered robot hands, *IEEE J. Robot. Autom.* 4 (1988) 32–44. <http://dx.doi.org/10.1109/56.769>.
- [46] C. Borst, M. Fischer, G. Hirzinger, Grasp planning: how to choose a suitable task wrench space, in: 2004 IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. Presented at the 2004 IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04, vol. 1, 2004, pp. 319–325. <http://dx.doi.org/10.1109/ROBOT.2004.1307170>.
- [47] Y. Lin, Y. Sun, Grasp planning to maximize task coverage, *Int. J. Robot. Res.* (2015). <http://dx.doi.org/10.1177/0278364914583880>.
- [48] A. Caldas, A. Micaelli, M. Grossard, M. Makarov, P. Rodriguez-Ayerbe, D. Dumur, New metric for wrench space reachability of multifingered hand with contact uncertainties, in: 2014 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM). Presented at the 2014 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM), 2014, pp. 1236–1242. <http://dx.doi.org/10.1109/AIM.2014.6878251>.
- [49] G. Smith, E. Lee, K. Goldberg, K. Böhringer, J. Craig, Computing Parallel-Jaw Grip Points, 1999.
- [50] A.T. Miller, P.K. Allen, Graspt!: A versatile simulator for grasp analysis, in: *Proc. of the asme dynamic systems and control division*, 2000, pp. 1251–1258.
- [51] B. León, S. Ulbrich, R. Diankov, G. Puche, M. Przybylski, A. Morales, T. Asfour, S. Moio, J. Bohg, J. Kuffner, R. Dillmann, OpenGRASP: A toolkit for robot grasping simulation, in: N. Ando, S. Balakirsky, T. Hemker, M. Reggiani, O. von Stryk (Eds.), *Simulation, Modeling, and Programming for Autonomous Robots*, in: *Lecture Notes in Computer Science*, Springer, Berlin, Heidelberg, 2010, pp. 109–120.
- [52] Malvezzi G. Gioioso, G. Salvietti, D. Prattichizzo, A. Bicchi, A. Bicchi, SynGrasp: a MATLAB Toolbox for Grasp Analysis of Human and Robotic Hands [WWW Document], 2013.
- [53] A.T. Miller, Graspt!: A versatile simulator for robotic grasping, *IEEE Robot. Autom. Mag.* 11 (2004) 110–122.
- [54] OpenRAVE [WWW Document], 2015. <http://openrave.org/>. (Accessed 12 August 2015).
- [55] OpenGRASP [WWW Document], 2015. <http://opengrasp.sourceforge.net/>. (Accessed 12 August 2015).
- [56] P. Pedrazzoli, R. Rinaldi, C.R. Boer, A rule based approach to the gripper selection issue for the assembly process, in: Proceedings of the IEEE International Symposium on Assembly and Task Planning, 2001. Presented at the Proceedings of the IEEE International Symposium on Assembly and Task Planning, 2001, 2001, pp. 202–207. <http://dx.doi.org/10.1109/ISATP.2001.928990>.
- [57] D.T. Pham, N.S. Gourashi, E.E. Eldukhri, Automated configuration of gripper systems for assembly tasks, *Proc. Inst. Mech. Eng. B* 221 (2007) 1643–1649. <http://dx.doi.org/10.1243/09544054JEM878SC>.
- [58] F. Sanfilippo, G. Salvietti, H.X. Zhang, H.P. Hildre, D. Prattichizzo, Efficient modular grasping: An iterative approach, in: 2012 4th IEEE RAS EMBS International Conference on Biomedical Robotics and Biomechatronics, BioRob. Presented at the 2012 4th IEEE RAS EMBS International Conference on Biomedical Robotics and Biomechatronics, BioRob, 2012, pp. 1281–1286. <http://dx.doi.org/10.1109/BioRob.2012.6290693>.
- [59] R.C. Brost, R.R. Peters, Automatic design of 3-D fixtures and assembly pallets, *Int. J. Robot. Res.* 17 (1998) 1243–1281. <http://dx.doi.org/10.1177/027836499801701201>.
- [60] V.B. Velasco Jr., W.S. Newman, An Approach to Automated Gripper Customization Using Rapid Prototyping Technology, 1996.
- [61] V.B. Velasco, W.S. Newman, Y. Zheng, S. Choi, Automated Gripper and Fixture Customization via Rapid Prototyping, 1998.
- [62] eGrip, eGrip [WWW Document], 2015. <http://www.egrip.schunk.com/Account/LogOn?ReturnUrl=%2f>. (Accessed 12 August 2015).
- [63] A. Schuster, R. Becker, M. Poguntke, Method for the production of customer-specific components, 2014, WO2014060605 A1.
- [64] S. Gottschalk, M.C. Lin, D. Manocha, OBBTree: A hierarchical structure for rapid interference detection, in: Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH'96, ACM, New York, NY, USA, 1996, pp. 171–180. <http://dx.doi.org/10.1145/237170.237244>.
- [65] J.D. Cohen, M.C. Lin, D. Manocha, M. Ponamgi, I-COLLIDE: An interactive and exact collision detection system for large-scale environments, in: Proceedings of the 1995 Symposium on Interactive 3D Graphics, I3D '95, ACM, New York, NY, USA, 1995, p. 189. <http://dx.doi.org/10.1145/199404.199437>.
- [66] T.C. Hudson, M.C. Lin, J. Cohen, S. Gottschalk, D. Manocha, V-COLLIDE: Accelerated collision detection for VRML, in: Proceedings of the Second Symposium on Virtual Reality Modeling Language, VRML '97, ACM, New York, NY, USA, 1997, p. 117. <http://dx.doi.org/10.1145/253437.253472>.
- [67] E. Larsen, S. Gottschalk, M.C. Lin, D. Manocha, Fast proximity queries with swept sphere volumes, 1999.
- [68] J.-P. Saut, D. Sidobre, Efficient models for grasp planning with a multi-fingered hand, *Robot. Auton. Syst., Autonomous Grasping* 60 (2012) 347–357. <http://dx.doi.org/10.1016/j.robot.2011.07.019>.
- [69] CATIA [WWW Document], 2016. <http://www.3ds.com/>. (Accessed 2 February 2016).
- [70] SolidWorks [WWW Document], 2015. <http://www.solidworks.com/>. (Accessed 12 August 2015).
- [71] N. Daoud, J.P. Gazeau, S. Zeghloul, M. Arsicault, A fast grasp synthesis method for online manipulation, *Robot. Auton. Syst.* 59 (2011) 421–427. <http://dx.doi.org/10.1016/j.robot.2011.02.005>.
- [72] J.C. Trinkle, A quantitative test for form closure grasps, in: Proceedings of the 1992 IEEE/RSJ International Conference on Intelligent Robots and Systems, 1992. Presented at the, Proceedings of the 1992 IEEE/RSJ International Conference on Intelligent Robots and Systems, 1992, 1992, pp. 1670–1677. <http://dx.doi.org/10.1109/IROS.1992.594246>.
- [73] H. Bruyninckx, S. Demey, V. Kumar, Generalized stability of compliant grasps, in: Proceedings of 1998 International Conference on Robotics and Automation, 1998, pp. 2396–2402.
- [74] M.A. Roa, R. Suárez, Finding locally optimum force-closure grasps, *Robot. Comput.-Integr. Manuf.* 25 (2009) 536–544. <http://dx.doi.org/10.1016/j.rcim.2008.02.008>.
- [75] R. Balasubramanian, L. Xu, P.D. Brook, J.R. Smith, Y. Matsuoka, Physical human interactive guidance: Identifying grasping principles from human-planned grasps, *IEEE Trans. Robot.* 28 (2012) 899–910. <http://dx.doi.org/10.1109/TRO.2012.2189498>.
- [76] J. Weisz, P.K. Allen, Pose error robust grasping from contact wrench space metrics, in: 2012 IEEE International Conference on Robotics and Automation, ICRA. Presented at the 2012 IEEE International Conference on Robotics and Automation, ICRA, 2012, pp. 557–562. <http://dx.doi.org/10.1109/ICRA.2012.6224697>.
- [77] G.M. Bone, A. Lambert, M. Edwards, Automated modeling and robotic grasping of unknown three-dimensional objects, in: IEEE International Conference on Robotics and Automation, 2008. ICRA 2008. Presented at the IEEE International Conference on Robotics and Automation, 2008. ICRA 2008, 2008, pp. 292–298. <http://dx.doi.org/10.1109/ROBOT.2008.4543223>.
- [78] R. Ala, D.H. Kim, S.Y. Shin, C. Kim, S.-K. Park, A 3D-grasp synthesis algorithm to grasp unknown objects based on graspable boundary and convex segments, *Inf. Sci.* 295 (2015) 91–106. <http://dx.doi.org/10.1016/j.ins.2014.09.062>.
- [79] F. Sanfilippo, H. Zhang, K.Y. Pettersen, G. Salvietti, D. Prattichizzo, ModGrasp: An open-source rapid-prototyping framework for designing low-cost sensorised modular hands, in: Biomedical Robotics and Biomechatronics (2014 5th IEEE RAS EMBS International Conference on. Presented at the Biomedical Robotics and Biomechatronics (2014 5th IEEE RAS EMBS International Conference on, 2014, pp. 951–957. <http://dx.doi.org/10.1109/BIOROB.2014.6913903>.



**Mohammadali Honarpardaz** is Associate Scientist at ABB Corporate Research and studying industrial Ph.D. in the field of robot finger design automation at Linköping University. He received his bachelor's degree in Fluid Mechanics in 2008 and M.Sc. in Structural Mechanics from Blekinge Institute of Technology in 2011. He works at ABB Corporate Research since 2011 and currently is responsible for mechanical simulations and calculations of ABB products. His research interests are Design Automation and Robotics, specifically grippers.



**Johan Ölvander** is full professor in Engineering Design at Linköping University (LiU). He received his Ph.D. from LiU in 2001. Currently, he is division manager in Machine Design at Linköping University. His research interests are within the field of product design and development, and specifically how simulation and optimization techniques can be employed in order to support the development of complex engineering systems. Important aspects of this work include formulation of the optimization problem in general and formulation of objective functions in particular.



**Mehdi Tarkian** received an M.Sc. in Mechanical Engineering from Linköping University 2007, and a Ph.D.-degree in 2012. He is currently an Assistant Professor at the division of Machine Design in Linköping University. His key research interest are Design Automation and Multidisciplinary Optimization. The challenges have been to establish methods for knowledge capture, storage, execution and optimization. A close collaboration with Swedish companies such as ABB has enabled the design methodologies to be evaluated in real industrial context, resulting in practical and industry related results.



**Xiaolong Feng** is the global research area manager of mechanics at ABB and docent professor at Linköping University. He has a background in mechanical engineering with a bachelor's degree from Zhejiang University in China in 1983 and a master of The Petroleum University of China in 1988. He continued his scientific work at Stockholm University where he obtained a licentiate degree in 1996 and a doctorate in 1998, both in physical chemistry. Since 1998, he works at ABB Corporate Research, currently, he is Global Research Area Manager.