Hierarchical Fingertip Space: A Unified Framework for Grasp Planning and In-Hand Grasp Adaptation

Kaiyu Hang, Miao Li, Johannes A. Stork, Yasemin Bekiroglu, Florian T. Pokorny,
Aude Billard and Danica Kragic

***Abstract*—We present a unified framework for grasp planning and in-hand grasp adaptation using visual, tactile and proprioceptive feedback. The main objective of the proposed framework is to enable fingertip grasping and address problems of changed weight of the object, slippage and external disturbances. For this purpose, we introduce Hierarchical Fingertip Space (HFTS) concept as a representation enabling optimization for efficient grasp synthesis. Grasp synthesis is followed by a grasp adaptation step that consists of both grasp force adaptation through impedance control and regrasping/finger gaiting when the former is not sufficient. Experimental evaluation is conducted on an Allegro hand mounted on a Kuka LWR arm.**

***Index Terms*—Fingertip grasping, Hierarchical Fingertip Space, grasp synthesis, grasp adaptation**

I. Introduction

GRASP planning and in-hand grasp adaptation are two complex problems that have commonly been studied separately. Lots of contributions to these problems have been made during the past two decades considering stability modeling and estimation, task based grasping, object representation, grasping synergies and grasp adaptation [1]–[10].

In this paper, we present a framework for fingertip grasping considering an integrated approach to grasp planning and in-hand grasp adaptation. The main objective of the framework is to address the problem of grasp instability due to problems such as changed weight of the object, slippage or external disturbances such as collision. The framework implements both grasp force adaptation through impedance control and regrasping/finger gaiting when the former is not sufficient. We integrate our previous work on Hierarchical Fingertip Space (HFTS) based grasp synthesis [11] and grasp adaptation [9]. The approach consists of i) a pre-grasping phase executing grasp synthesis on an efficient representation including both object and hand properties, ii) grasp execution, and iii) a post-grasping phase where tactile feedback and experience are used for in-hand grasp adaptation, see Fig. 1 and Fig. 2.

In the pre-grasping phase, grasp synthesis is formulated as a combinatorial optimization problem considering grasp stability, contact locations and finger gaiting in an integrated manner. In the post-grasping phase, tactile feedback provides information of the stability of the executed grasp. An offline learned probabilistic model is used to assess the grasp stability and initiate an adaptation of grasp forces, followed by finger gaiting if needed. To the best of our knowledge, this is so far the only system that accomplishes grasp synthesis, stability estimation, online replanning and in-hand adaptation in a unified framework and evaluated on a real physical system.

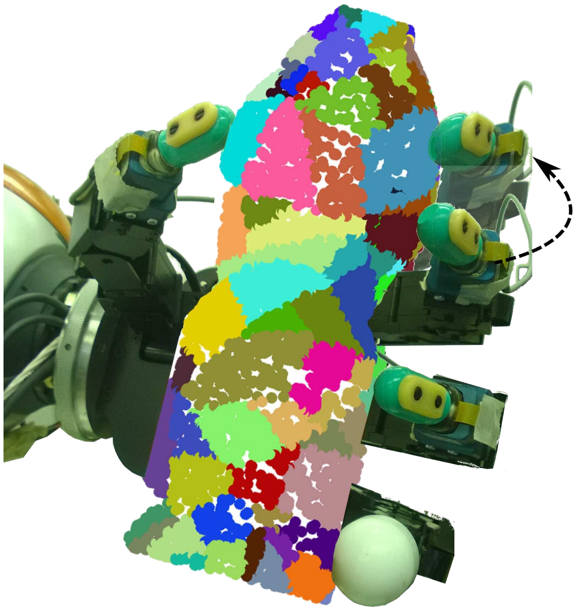


Fig. 1. A visualization of the proposed Hierarchical Fingertip Space concept: Initial fingertip locations are determined by optimizing grasp stability and adaptability using a hierarchical discretization of the object surface and an impedance controller is used to balance grasping forces. If a large disturbance occurs, the grasp is adapted by fingertip gaiting to maintain grasp stability. The new fingertip location is computed using an optimization in the HFTS.

Compared to the state of the art and our previous work in [11] and [9], our integrated system:

- closes the loop between grasp planning and control through stability estimation and finger gaiting;
- introduces a Gaussian Process-based cluster analysis for the Hierarchical Fingertip Space representation that is considering surface properties of objects and viable contact locations;
- optimizes grasp adaptability and demonstrates informed finger gaiting by considering viable hand configurations and object shape knowledge.

We review the related work in Sec. II and present the methodology in Sec. III, Sec. IV and Sec. V. We evaluate in Sec. VI and then conclude in Sec. VII.

K. Hang, J. A. Stork, Y. Bekiroglu, F. T. Pokorny and D. Kragic are with KTH Royal Institute of Technology, Stockholm, Sweden, as members of the Computer Vision & Active Perception Lab., Centre for Autonomous Systems {kaiyuh, jastork, yaseminb, fpokorny, dani}@kth.se.

M. Li and A. Billard are with Learning Algorithms and Systems Laboratory (LASA) at École Polytechnique Fédérale de Lausanne (EPFL), Switzerland {miao.li, aude.billard}@epfl.ch.

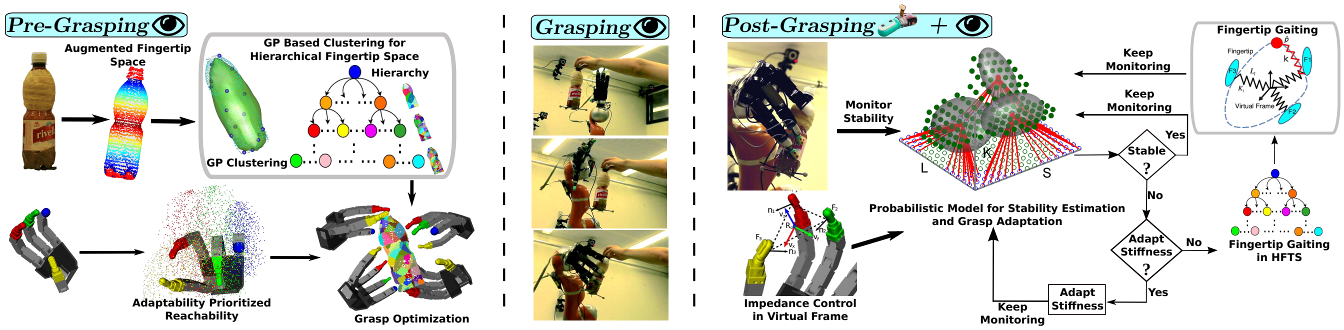

Pre-Grasping
Augmented Fingertip Space
GP Based Clustering for Hierarchical Fingertip Space
Hierarchy
GP Clustering
Adaptability Prioritized Reachability
Grasp Optimization
Grasping
Post-Grasping +
Monitor Stability
L
K
S
Probabilistic Model for Stability Estimation and Grasp Adaptation
Impedance Control in Virtual Frame
Keep Monitoring
Keep Monitoring
Fingertip Gaiting
Stable ?
Yes
No
Adapt Stiffness ?
No
Yes
Fingertip Gaiting in HFTS
Adapt Stiffness
Keep Monitoring

Fig. 2. Schematic overview of the system: *Pre-grasping*: After the Hierarchical Fingertip Space is generated, grasps are synthesized by a multi-level refinement strategy. *Grasping*: The synthesized hand configuration is used to execute the grasp. *Post-grasping*: Once tactile feedback is available, grasp stability is monitored by a learned probabilistic model. If a grasp is estimated as unstable, the stability is maintained through force adaptation or finger gaiting.

II. Related Work

The area of object grasping covers several problems: grasp stability analysis, data-driven grasp synthesis and planning, object and task representation, grasp adaptation [6], [10], [12] etc. Although each of these have been studied extensively during the past couple of decades there are rather few systems that have addressed grasp synthesis and in-hand grasp adaptation in an integrated manner.

In terms of object representation, there are many examples of works that rely on encoding shape properties of objects: Reeb Graph [13], Medial Axis [14], [15], hierarchical box decomposition [16], super-quadrics [17]–[20]. More recent work demonstrates topological analysis of shape for grasping and caging [21], [22]. Our HFTS proposes a hierarchical method for shape representation that encodes both the global and local shape of the object.

Classical work formulates contact-level grasp synthesis as an optimization problem [8], [12], [23]–[27] for which the objective—grasp stability—is commonly performed using force analysis in the contact wrench space [28]. The problem of estimating feasible hand configurations has also been addressed in this context [2], [29]. To account for uncertainties in physical properties of objects, grasp friction sensitivity [30] and independent contact regions [31] have been investigated. Our approach formulates fingertip grasping as an optimization problem considering grasp stability, adaptability and hand reachability to prepare a grasp for future adaptive execution against physical uncertainties.

Approaches to force based grasp control range from geometry based analytic methods [32]–[34] to learning-based frameworks for force optimization [35], [36]. In-hand manipulation has been addressed as finger gaiting with a rolling contact model and quasi-static assumption [37], [38]. Hybrid position and force control has also been addressed [39]–[42] as well as impedance control [43]–[46]. Our approach allows for grasp stabilization through both contact force adaptation and finger gaiting planned in real-time using tactile feedback and the proposed Hierarchical Fingertip Space.

In realistic tasks, the ability to retain a stable grasp on an object is an integral property of robust systems. A grasp originally estimated as stable may be perturbed while performing a manipulation with the held object. This is also valid for cases where some properties of the object change - weight can change if a glass held by the robot gets filled, environmental changes can affect friction coefficients, collision may cause slippage, etc. In addition, many of these properties may not be exactly known to start with. Thus, in-hand grasp adaptation may be needed after a grasp has been applied on an object. For this purpose, relying on visual feedback is not sufficient and many of the recent approaches facilitate haptic and proprioceptive information [7], [47]–[53]. Finger gaiting may be further required when applying higher grasping force does not suffice [9], [54]. Our work here builds upon [9], [54] and additionally allows for replanning during grasp execution.

III. Hierarchical Fingertip Space and Grasp Optimization

In the *pre-grasping* phase, we formulate fingertip grasp synthesis as an optimization problem considering each object represented as a point cloud $\mathcal{P} = \{p_i \in \mathbb{R}^3 \mid i \in \{1, ..., n_p\}\}$. We seek m contact locations, $C_g = \{c_1, ..., c_m | c_i \in \mathcal{P}\}$, on the object surface and a hand configuration, $\mathcal{J}_g \in \mathbb{R}^d$ where d are controlled joint angles.

There are two concepts we define: *Fingertip Space* and *Hierarchical Fingertip Space (HFTS)*. *Fingertip Space* represents a finite set of contacts on an object surface that are locally flat and large enough for a fingertip [11]. We denote the *Fingertip Space* as $\Phi(\mathcal{P}) = \{\phi_1, ..., \phi_{n_f}\} \subset \mathcal{P}$ and an element of this set ϕ_i is called a *Fingertip Unit*. Fingertip Space Φ is parametrized by locations and normals of Fingertip Units. We extract the $\Phi(\mathcal{P})$ from $\mathcal{P}$ based on the estimated surface curvature from a set of points within one fingertip size[1] r, around the potential contact, $N^r(p_i) \subset \mathcal{P}$. The fingertip space of $\mathcal{P}$ is given by

$$\Phi(\mathcal{P}) = \{\phi_i \mid \mathcal{K}(N^r(\phi_i)) \leq \kappa, \phi_i \in \mathcal{P}\} \qquad (1)$$

where $\mathcal{K}(N^r(p_i))$ is the local surface curvature estimated from $N^r(p_i)$ and $\kappa \in \mathbb{R}$ is the empirically determined curvature threshold. In the rest of this paper, we write Φ instead of $\Phi(\mathcal{P})$. Fig. 3(left) shows an example of *Fingertip Space*. To enable finger gaiting, we want our Fingertip Space to encode the space around each Fingertip Unit in an efficient manner. To achieve this, we put a penalty term on admissible regions

[1] For the SynTouch sensor used in this work, the fingertip size r is $14mm$, http://www.syntouchllc.com/

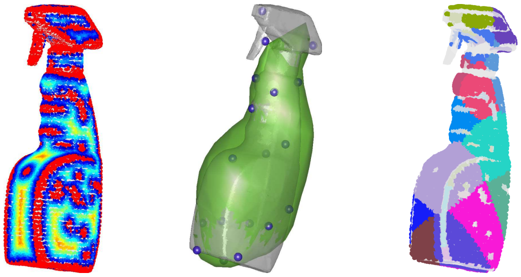


Fig. 3. Left: Fingertip Space with attached penalties rendered by jet colormap, in red are points that have been filtered out by Eq. (1), note that there are perception noise in planar areas. Middle: GP for the spray model represented by 20 cluster centers. Right: partitions of fingertip units rendered with different colors.

using a logistic function. Let $c(\phi_i) \in \mathcal{P} \setminus \Phi$ be the closest point to Fingertip Unit ϕ_i that has been rejected by Eq. (1), the penalty w_i is computed as:

$$w_i = \frac{1}{1 + e^{-\gamma \| \phi_i - c(\phi_i) \|}} \tag{2}$$

where $\gamma \in \mathbb{R}^+$ is an elasticity factor, see Fig. 3(left).

A. Multilevel refinement of Fingertip Space

Given the large number of Fingertip Units per object, formalizing grasp optimization on all combinations of these is computationally impractical. A more feasible strategy is to apply Surrogate Models or multilevel refinement [55], [56], that recursively approximate the original optimization problem in a hierarchy of simpler, more tractable problems i.e. surrogate models. We first explain grasp synthesis for a single fingertip contact optimization (gaiting) and then continue with optimization for multiple fingertip contacts.

Surrogate approximation of Φ is constructed by recursively grouping Fingertip Units by cluster analysis using geometric properties. For the optimization of a single contact in Φ we construct a hierarchy of surrogate approximations of Φ (see Fig. 4) as a similarity-based graph $G_\Phi = (E_\Phi, V_\Phi)$, with the hierarchy levels $i \in \{0, \ldots, l-1\}$ representing different scales of surrogate approximations. Φ is recursively partitioned into smaller sets of fingertip units, denoted as $\hat{\phi}_{i,j} \subset \Phi$, and is represented as a node $\phi_{i,j} \in V_\Phi$ in graph G_Φ, where i is the level of $\phi_{i,j}$ in the hierarchy and j is the index of the partition in level i. We partition the set Φ in a top-down manner, with parent $\phi_{i,j}$ nodes split into children nodes if $|\hat{\phi}_{i,j}| > 1$. Ultimately, the bottom level of G_Φ consists of nodes representing single fingertip units, $|\hat{\phi}_{0,j}| = 1$. Experimentally and as shown in Fig. 4, the number of partitioning centers for the second top level is set to 20 and in the remaining levels to 4 children similar to [8].

Given real sensor data, there is noise associated with the computation of surface normals. To address this, we employ a *Gaussian process* based filter with *Thin Plate Spline kernel* [57]. Higher sampling frequency for GP centers is used in areas of higher curvature, see Fig. 3(middle). The distribution of centers captures the geometric similarities (locations and normals) and therefore relate to the similarities in the grasp wrench space [58]. GP partitioning is regulated using a threshold T_p, so that if $|\hat{\phi}_{i,j}| \leq T_p$, a node is not further divided by GP partitioning but it is split up into all its fingertip units. Nodes consisting of single fingertip units are copied to the next level as long as some other nodes can be partitioned. This guarantees a balanced partitioning tree, and hence a valid surrogate approximation for every level in the hierarchy. As discrete optimization relies on relevant neighbors in the solution space, we introduce connectivity by introducing additional edges between nodes and their siblings into E_Φ. More precisely, the extended edge set consists of parent-child edges and sibling-edges $E_\Phi = E_\Phi^P \cup E_\Phi^S$ which are given as:

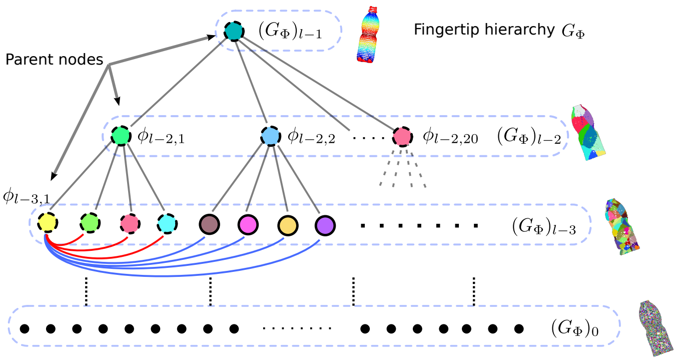


Fig. 4. Surrogate models represented as a graph. Fingertip unit partitions are represented as nodes in different levels of the hierarchy and the connectivities in this graph are represented by edges. Extra connectivities defined by Eq. (3) are exemplified in level $l-3$ for $\phi_{l-3,1}$, with red edges for 2 hops and blue edges for 4 hops.

$$\begin{aligned} E_\Phi^P &= \{\{\phi_{i,j}, \phi_{i-1,k}\} \in V_\Phi \times V_\Phi \mid \hat{\phi}_{i-1,k} \subseteq \hat{\phi}_{i,j}\} \\ E_\Phi^S &= \{\{\phi_{i,j}, \phi_{i,k}\} \in V_\Phi \times V_\Phi \mid hop(\phi_{i,j}, \phi_{i,k}) \leq h\} \end{aligned} \tag{3}$$

The function $hop(\phi_{i,j}, \phi_{i,k})$ denotes the hop distance between $\phi_{i,j}$ and $\phi_{i,k}$ along edges in E_Φ^P. The hop limit $h \in \mathbb{N}$ defines the size of the neighborhood and is set to 4 in our experiments, resulting in neighborhoods of size e.g., ca $4cm$ in the second top level. Using the definitions above, we can now define the i-th surrogate approximation of the Fingertip Space Φ as:

$$\begin{aligned} (G_\Phi)_i &= ((V_\Phi)_i, (E_\Phi)_i) \\ (V_\Phi)_i &= \bigcup_j \{\phi_{i,j}\} \\ (E_\Phi)_i &= \{\{\phi_{i,j}, \phi_{i,k}\} \mid \{\phi_{i,j}, \phi_{i,k}\} \in E_\Phi\} \end{aligned} \tag{4}$$

which is a subgraph of G_Φ and an approximation at i-th resolution level.

We define the mean location and orientation of the set of fingertip units contained in the partition $\hat{\phi}_{i,j}$ as $\mathbf{p}(\hat{\phi}_{i,j}) \in \mathbb{R}^3$ and $\mathbf{n}(\hat{\phi}_{i,j}) \in \mathbb{R}^3$, this will be used later for stability analysis. In terms of Eq. (2), the penalty assigned to a node $\phi_{i,j}$ is defined as:

$$w_{i,j} = \frac{1}{|\hat{\phi}_{i,j}|} \sum_{\phi_k \in \hat{\phi}_{i,j}} w_k \tag{5}$$

Given the hierarchy G_Φ of surrogate approximation models, we can optimize a fingertip location in a top-down manner. By optimizing the contact in a coarse to fine fashion, a final contact will be found in the bottom of the hierarchy. Next, we will investigate the grasp synthesis with multiple contacts.

B. Hierarchical Fingertip Space

In the previous section, we introduced the similarity-based graph G_Φ for a single fingertip. For m fingertips, we define the product graph $\Lambda_\Phi = (V_\Lambda, E_\Lambda)$ named *Hierarchical Fingertip Space (HFTS)* as in Eq.(6). Thus, nodes in V_Λ represent combinations of m contacts, $\lambda_{i,j} = (\phi_{i,j_1}, ..., \phi_{i,j_m})$, and the graph-distance between nodes in the same level reflects the similarity of the individual contacts. Formally, the *HFTS* is defined as:

$$\begin{aligned} \Lambda_\Phi &= G_\Phi^1 \times \cdots \times G_\Phi^m \\ V_\Lambda = \{\lambda_{i,j} &= (\phi_{i,j_1}, ..., \phi_{i,j_m}) \mid \lambda_{i,j} \in V_\Phi^1 \times \ldots \times V_\Phi^m\} \end{aligned} \tag{6}$$

where $G_\Phi^k = (V_\Phi^k, E_\Phi^k)$ is the surrogate hierarchy for the k-th fingertip. The penalty value for a set of contacts is defined as minimum of all individual contact penalties:

$$w_{i,j}^* = \min\{w_{i,j_1}, \ldots, w_{i,j_m}\} \tag{7}$$

Optimization on Λ_Φ requires definition of neighborhoods and we define two types of edges for E_Λ: *1)* Edges between nodes and their parent, E_Λ^P, such that Λ_Φ inherits the hierarchy levels from the individual G_Φ^k, and *2)* edges between nodes of the same level, E_Λ^S, for which the individual contacts are identical or siblings in their graph G_Φ^k, respectively. Formally, we obtain $E_\Lambda = E_\Lambda^P \cup E_\Lambda^S$:

$$\begin{aligned} E_\Lambda^P &= \{\{\lambda_{i,j_1}, \lambda_{i-1,j_2}\} \mid \forall k \colon \{\lambda_{i,j_1}^{(k)}, \lambda_{i-1,j_2}^{(k)}\} \in E_\Phi^P\} \\ E_\Lambda^S &= \{\{\lambda_{i,j_1}, \lambda_{i,j_2}\} \mid \forall k \colon \{\lambda_{i,j_1}^{(k)}, \lambda_{i,j_2}^{(k)}\} \in E_\Phi^S\} \end{aligned} \tag{8}$$

where $\lambda_{i,j}^{(k)} \in V_\Phi^k$ is the k-th item of tuple $\lambda_{i,j}$. Similarly to the surrogate models for a single fingertip contact, we define the i-th surrogate approximation of multiple fingertip grasping in *HFTS* as:

$$\begin{aligned} (\Lambda_\Phi)_i &= ((V_\Lambda)_i, (E_\Lambda)_i) \\ (V_\Lambda)_i &= \bigcup_j \{\lambda_{i,j}\} \\ (E_\Lambda)_i &= \{\{\lambda_{i,j_1}, \lambda_{i,j_2}\} \mid \{\lambda_{i,j_1}, \lambda_{i,j_2}\} \in E_\Lambda\} \end{aligned} \tag{9}$$

C. Grasp Optimization in HFTS

So far, we described the solution space for grasp synthesis using nodes $\lambda_g \in \Lambda_\Phi$ from different levels, which are combinations of contacts on the object surface. However, to realize the contacts with a robot hand, we additionally need the joint angles $\mathcal{J}_g \in \mathbb{R}^d$. A valid *grasp solution*, $g = (\lambda_g, \mathcal{J}_g)$, is a combination of contact positions and joint angles.

1) Grasp Stability: During the pre-grasping phase, when we synthesize a grasp, only visual information of object is available and we need to evaluate or predict grasp stability without feedback. This can be done using contact based force closure analysis [12], [28]: Given a grasp solution g, the grasp quality measure $Q(\lambda_g) \in \mathbb{R}$ is the minimum offset between the origin of the wrench space and the convex hull spanned by friction cones of contacts parametrized by positions and normals [23]. The value is positive when the grasp is force closed and larger for more stable grasps.

2) Grasp Reachability: Not all combinations of contacts λ_g can be realized by a given robotic hand and we can classify contacts into reachable or unreachable using a function $R^* \colon V_\Lambda \to \{0, 1\}$ so that the optimization can be constrained to reachable grasps with $R^*(\lambda_g) = 0$. Since a robotic hand can have many degrees of freedom and complicated coupled kinematics, and we seek a computation time on the scale of milliseconds during grasp optimization, it can be too costly to analytically compute $R^*(\lambda_g)$ in each optimization step. For this, various forms of constraints have been formulated [59], [60]. To achieve required speed and precision, we linearly relax it as a measure of dissimilarity between λ_g and the closest known reachable contacts λ_g^* of grasp solution $g^* = (\lambda_{g^*}, \mathcal{J}_{g^*})$. The reachability measure of m contacts λ_g is then reformulated as a residual $R(\lambda_g) \in \mathbb{R}^+$:

$$R(\lambda_g) = \left\| C(\lambda_g) - C(\lambda_{g^*}) \right\| \tag{10}$$

where $C(\cdot) \in \mathbb{R}^{6 \times (m-2)}$ is an affine invariant encoding of m contacts in terms of its contact locations and normals [11].

To generate a collection of viable grasps, we randomly sample hand configurations and save the encoded contacts and corresponding hand configuration $\mathcal{J}_g$ into a k-d tree like data structure T offline with the query time $\mathcal{O}(n \log n)$. Using T, we can compute the residual by lookup and find the hand configuration for realizing the contacts if the residual was small.

$$T \colon \lambda_g \mapsto (\mathcal{J}_{g^*}, R(\lambda_g)) \tag{11}$$

3) Grasp Adaptability: We use *grasp adaptability* to enable finger gaiting already in the grasp synthesis stage. By decomposing the hand Jacobian and calculating the manipulability [61] of a hand configuration in the tangential plane of contacts, we measure the *adaptability* of a grasp, denoted as $A(\mathcal{J}_g) \in \mathbb{R}^+$. Concretely, given the Jacobian $J_f(\mathcal{J}_g) \in \mathbb{R}^{3 \times n}$ and the normal $\mathbf{n}_f \in \mathbb{R}^3$ of fingertip f, the Jacobian can be rotated by $R_f \in \mathbb{R}^{3 \times 3}$ such that the last row of $\hat{J}_f(\mathcal{J}_g) = R_f J_f(\mathcal{J}_g)$ corresponds to the movement of fingertip in the direction of $\mathbf{n}_f$. The first two rows of $\hat{J}_f$, denoted by $\widetilde{J_f}(\mathcal{J}_g) \in \mathbb{R}^{2 \times n}$, are then the projection of J_f in the tangential plane of the fingertip normal.

$$A(\mathcal{J}_g) = \sum_f \sqrt{det \widetilde{J_f}(\mathcal{J}_g) \widetilde{J_f}^T(\mathcal{J}_g)} \tag{12}$$

Note that we can assume that the fingertip normal (on the robot hand) and the fingertip unit normal will be similar when the grasp is realized if $R(\lambda_g)$ is small. An example of grasp adaptability measure is shown in Fig. 5. Since this measure is hand configuration based, it is affine invariant, and hence grasp pose independent.

In order to capture grasp stability, reachability and adaptability in the grasp optimization, the optimization objective is defined as follows:

$$\text{Priority 1: Maximize} \quad \theta(g) \tag{13}$$

$$\text{Priority 2: Maximize} \quad A(\mathcal{J}_g) \tag{14}$$

with

$$\theta(g) = Q(\lambda_g) - \alpha R(\lambda_g), \quad \alpha \in \mathbb{R}^+ \tag{15}$$

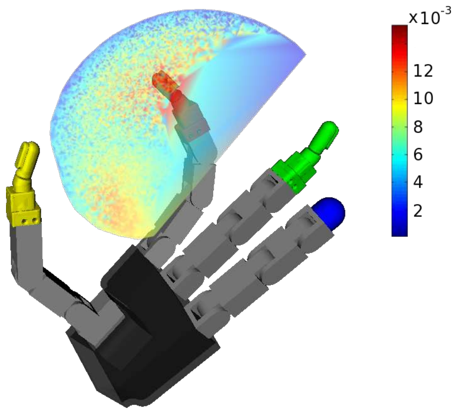


Fig. 5. Grasp Adaptability for fingertip 1. Adaptability is computed for fingertip positions sampled in joint space. The colored volume shows fingertip positions and their adaptability values.

Algorithm 1 Surrogate-Based Optimization in *HFTS*

```
Input: $stopCondition$, $\Lambda_\Phi$, $maxIter$
Output: grasp $g = (\lambda_g, \mathcal{J}_g)$
 1: for $i = l - 1$ to $0$ do
 2:     if $i = l - 1$ then                               ▷ Initialization
 3:         $\lambda_g \leftarrow$ random from $(\Lambda_\Phi)_i$
 4:     else                                              ▷ Extend to Lower Surrogate
 5:         $\lambda_g \leftarrow$ random child of $\lambda_g$
 6:     end if
 7:     $(\mathcal{J}_g, R(\lambda_g)) \leftarrow T(\lambda_g)$
 8:     for $1$ to $maxIter$ do                           ▷ Optimize Surrogate
 9:         $\lambda_{g'} \leftarrow$ random neighbor of $\lambda_g \in (\Lambda_\Phi)_i$
10:         $(\mathcal{J}_{g'}, R(\lambda_{g'})) \leftarrow T(\lambda_{g'})$
11:         if $Pr(g, g') \geq \text{rand}(0, 1)$ then
12:             $g \leftarrow g'$
13:         end if
14:         if $stopCondition(g)$ then                    ▷ Good Solution
15:             break
16:         end if
17:     end for
18: end for
```

where α is a weighing factor to account for the hand size. To optimize the second objective, we use a sorted lookup table for $R(\lambda_g)$ which returns the most adaptable joint configuration in the area of the best grasp according to $A(\lambda_g)$ [62], [63] when querying reachability residuals (line 7 and 10 in Alg. 1). As we can see in Fig. 6, for the same contact locations, there can be multiple hand configurations for realizing it, however, our prioritized lookup table will always return the hand configuration with the best adaptability.

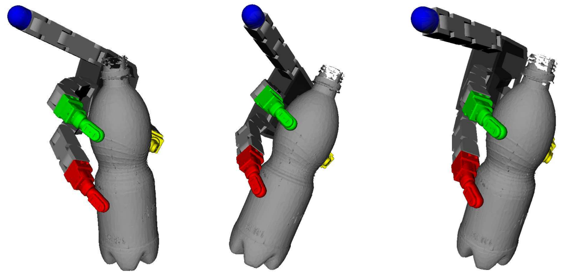


Fig. 6. Grasps with same contacts and different adaptabilities: the left grasp has the highest adaptability.

Having defined the objective function, we can now proceed to grasp optimization. Using a surrogate-based optimization metaheuristic, we need to find solutions on each of the surrogate approximations and extend them to the next model. For optimization in each model, we adopt stochastic hill climbing which can escape from local optima by means of randomness. Switching between two solutions g and g' is determined by the probabilistic function in Eq. (16):

$$Pr(g, g') = \left(1 + \exp \frac{w_g\theta(g) - w_{g'}\theta(g')}{\zeta}\right)^{-1} \tag{16}$$

where w_g is the penalty assigned to a tuple of contacts defined by Eq. (7). The randomness in the optimization is determined by ζ, it makes the optimization more random when a large value is chosen, while it behaves more like pure hill climbing if a small value is applied. The grasp optimization algorithm is shown in Alg. 1.

For realizing the grasp, we can transform the hand base to the pose where the fingertips meet the contact locations. In cases when the final reachability residual $R(\lambda_g) \neq 0$, a local optimization of joint configuration by linear interpolation [64] is required to realize desired contacts. To avoid too small and time consuming incremental improvements at each level, we utilize a *stopCondition*. It can be set to *false* if we want to explore the space until convergence or we control the number of iterations by setting a threshold for the optimization function in Eq. (15).

IV. Grasp adaptation

A synthesized grasp is executed using a simple position control [9]. When contacts are made, tactile readings are available and an impedance controller [3] is used to regulate grasp stiffness. The controller is formulated in a virtual frame (VF) defined in terms of fingertip locations as

$$\begin{aligned} R_o &= [\mathbf{v}_x, \mathbf{v}_y, \mathbf{v}_z] \in \text{SO}(3) \\ \mathbf{v}_x &= \frac{\mathbf{p}_3 - \mathbf{p}_1}{\|\mathbf{p}_3 - \mathbf{p}_1\|} \\ \mathbf{v}_z &= \frac{(\mathbf{p}_2 - \mathbf{p}_1) \times \mathbf{v}_x}{\|(\mathbf{p}_2 - \mathbf{p}_1) \times \mathbf{v}_x\|} \\ \mathbf{v}_y &= \mathbf{v}_z \times \mathbf{v}_x \end{aligned} \tag{17}$$

where $\mathbf{p}_1, \mathbf{p}_2$ and $\mathbf{p}_3 \in \mathbb{R}^3$ are locations of the fingertips, see Fig. 7.

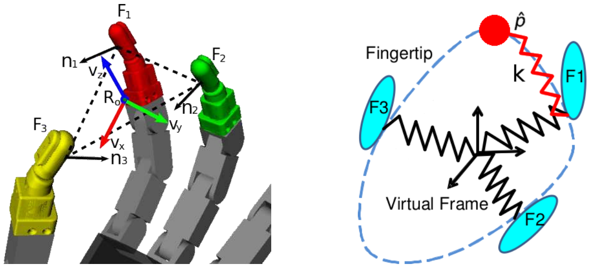


Fig. 7. Left: Virtual frame R_o, v_x, v_y and v_z defined by fingertip locations. Right: Virtual springs used by the impedance controller. A virtual spring (red) is superimposed on the impedance controller (between fingertip and the new location $\hat{p}$) when a fingertip gaiting is requested.

A grasp in the VF is denoted $\hat{g} = (K, L, S)$ where $K = (K_x, K_y, K_z) \in \mathbb{R}^3$ is grasp stiffness and $L = (L_1, L_2, L_3) \in$

$\mathbb{R}^3$ is grasp rest length, i.e. the distance between each fingertip and the center of VF. $S = (S_1, S_2, S_3) \in \mathbb{R}^{57}$ represents the tactile readings, in our case from SynTouch sensors.

Grasp stability is monitored using a probabilistic representation relying on a Gaussian Mixture Model Θ that is trained offline, see Fig. 8. As described in detail in our previous work [9], Θ is trained over K, L, S parameters for a variety of objects. Given Θ, grasp stability is estimated by

$$p(\hat{g}|\Theta) = \sum_{i=1}^{n_g} \pi_i \mathcal{N}(\hat{g}|\mu_i, \Sigma_i) \tag{18}$$

where n_g is the number of Gaussian components, each of which has a prior π_i. $\mathcal{N}(\hat{g}|\mu_i, \Sigma_i)$ is the Gaussian distribution with mean μ_i and covariance Σ_i.

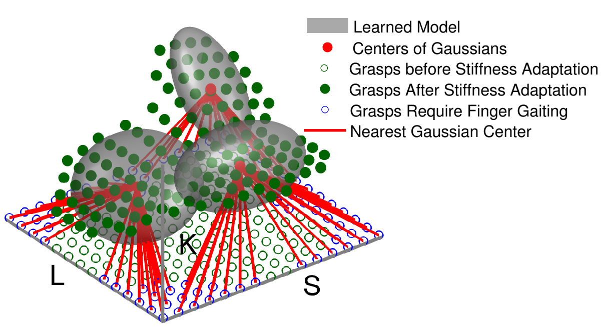


Fig. 8. GMM model Θ for grasp stability estimation and decision making for grasp adaptation. The gray ellipsoids depict the Gaussian components of Θ, dots and circles show grasps, represented by K, L, S in different stages, and the red lines illustrate how grasps are switched between different stages by grasp stiffness adaptation and fingertip gaiting.

A grasp $\hat{g}$ is considered unstable if the log likelihood of Eq. (18) is smaller than a predefined threshold, which is set to -100 as in our previous work [9]. If a grasp $\hat{g}$ is unstable, we compute its Mahalanobis distance to each component in Θ and denote the minimum distance as m_d. If m_d is within two standard deviations, we apply force adaptation by changing stiffness K to the value obtained by computing the maximum expectation of K conditioned on grasp rest length L and tactile readings S. The details of this process have been described in detail in our previous work [9]. Otherwise, a finger gaiting strategy is initiated as explained in detail in next section.

V. Regrasping by Finger Gaiting

Stiffness adaptation is not enough in cases when there is an upper bound on the force exerted by the hand. Thus, to stabilize a grasp, the system initiates finger gaiting. Finger gaiting is defined as an optimization problem based on the current rest length L represented in VF:

$$\theta^*(\lambda_g) = \|L - L^*\| + \beta R(\lambda_g) \tag{19}$$

where $R(\lambda_g)$ is the reachability defined in Eq. (10), $\beta \in \mathbb{R}^+$ is a weighing factor to account for the hand size and L^* is the desired rest length obtained from the closest Gaussian center $\hat{g}^* = (K^*, L^*, S^*)$ in terms of m_d. The reasoning above is to find the closest stable and reachable grasp to the current one, taking into account the current tactile readings.

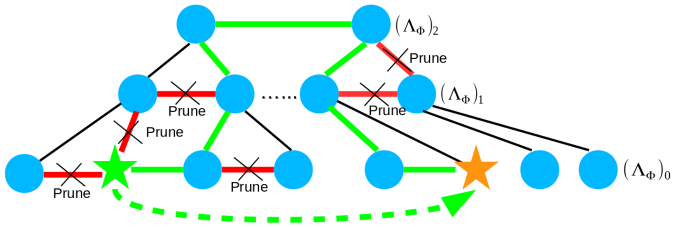


Fig. 9. Breadth-first search in *HFTS* for fingertip gaiting optimization. The green path shows how the search fringe evolves, and the red edges show the pruned path due to the 2 criteria defined in Alg. 2.

For the robot hand we use in this work, we can only relocate fingertip $F1$ or $F2$, as shown in Fig. 7, since relocating the thumb $F3$ leaves the grasp without contacts on opposite sides of the object. Our strategy of choosing between $F1$ and $F2$ is straightforward: we compute the optimization for $F1$ and $F2$ in parallel for choosing objective value from Eq. (19). Our optimization procedure employs breadth-first search in Λ_Φ starting from the initial contact. The search is terminated in each branch when the reachability measure grows beyond a predefined threshold ϵ_R. Since we move only one finger, we need an additional rule:

$$Prune(\lambda_g, \lambda_{g'}, f_o) = \begin{cases} True, & \exists i : i \neq f_o \wedge \lambda_g^{(i)} \neq \lambda_{g'}^{(i)} \\ False, & \text{otherwise.} \end{cases} \tag{20}$$

where f_o is the fingertip to be relocated, λ_g is the node that represents the current grasp contacts, and $\lambda_{g'}$ is the new solution. Since the search fringe can go upwards in the hierarchy graph Λ_Φ, this rule asserts that only a single fingertip is moved while the remaining two are kept fixed. The main idea is sketched in Fig. 9 and the procedure summarized in Alg. 2. Note that it includes the penalty factor from Eq. (7).

Algorithm 2 Fingertip Gaiting by Optimization in Λ_Φ

```
Input: $\Lambda_\Phi$, $\lambda_g$, $\epsilon_R$, $f_o$
Output: $\hat{p}$                                          ▷ New Location
$R_o = R(\lambda_g)$
$\lambda^* \leftarrow \lambda_g$
$Queue.push$(neighbors of $\lambda_g$)
while $Queue$ is not empty do
    $\lambda_{g'} \leftarrow Queue.pop()$
    if $R(\lambda_{g'}) > R_o + \epsilon_R$ or $Prune(\lambda_g, \lambda_{g'}, f_o)$ then
        continue                                           ▷ Pruning
    end if
    if $\frac{1}{w_{g'}}\theta^*(\lambda_{g'}) < \frac{1}{w_{\lambda^*}}\theta^*(\lambda^*)$ then
        $\lambda^* \leftarrow \lambda_{g'}$
    end if
    $Queue.push$(neighbors of $\lambda_{g'}$)              ▷ Breadth-First
end while
$\hat{p} \leftarrow \lambda^{*(f_o)}$                      ▷ New Location for $f_o$
```

A. Fingertip Gaiting in Practice

When grasp stability changes rapidly, the system would need to switch between impedance and position control frequently. To avoid this, we stay in impedance control mode during finger gaiting by sliding the finger to the desired position. To allow this, we formulate fingertip gaiting using impedance controller defined in VF. A virtual spring with stiffness k is defined to connect the current location of the moved fingertip and $\hat{p}$, which is equivalent to a fingertip impedance controller superimposed on the original grasp controller. An example of fingertip $F1$ gaiting is depicted as in Fig. 7.

The stiffness k of the virtual spring is determined by the distance $d_{\hat{p}} \in \mathbb{R}$ between the fingertip's current location $\hat{p}$ and an empirical parameter $\Gamma \in \mathbb{R}$ as: $k = d_{\hat{p}}\Gamma$. In this way, the fingertip will be slided towards $\hat{p}$ while keeping the contact on the object. Since $\hat{p}$ is computed in the *HFTS*, we ensure that the desired position is on the object surface. If a new goal position is requested during finger gaiting, the system will either continue to the new position if the same fingertip is concerned or stop the current gaiting and initiate gaiting with another fingertip. Another example situation is depicted in Fig. 10 where fingertip $F2$ stopped moving before the desired position is reached, since the grasp was estimated as stable on the way.

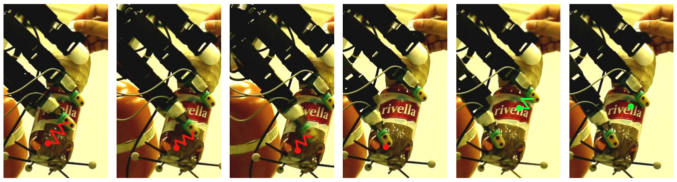


Fig. 10. The rivella bottle is grasped by the Allegro hand and a human is applying random perturbations on top of it. The red and green points are showing the new locations for fingertip $F1$ and $F2$ computed by Alg. 2 with virtual springs in the virtual frame.

VI. Experimental Evaluation

We perform experimental evaluation with an Allegro hand mounted on a Kuka LWR arm. The hand is equipped with SynTouch[2] tactile sensors on three fingertips. The systems performance is evaluated using six objects shown in Fig. 12, which are tracked using the OptiTrack[3] real-time motion tracking system. The evaluation presented below demonstrates the performance of the grasp synthesis system alone as well as the integrated system for grasp adaptation.

A. Grasp Synthesis

Grasp synthesis is performed on a point cloud representation of objects generated offline. We also generated a reachability table with 10^6 hand configurations using rejection sampling: grasps are first uniformly sampled in the hand joint space and those with adaptabilities larger than 0.02 are kept. The motivation for choosing the adaptabilities larger than 0.02 is shown in Fig. 11 (upper), showing the distribution of the adaptability.

Alg. 1 generates both contact locations and hand configurations. Simple position based control is used to execute a grasp [9]. A few examples are shown in Fig. 12 and Fig. 13.

For evaluating the performance of the grasp planner, we repeat the grasp optimization according to Alg. 1 for each test object. In order to keep an equal number of iterations for each repetition of the algorithm, we set $maxIter = 100$ and $stopCondition(g) = false$. For each object, we run the algorithm with random initialization until we achieve 100 stable and collision free grasps. Evaluation results are summarized into a table shown in Fig. 14.

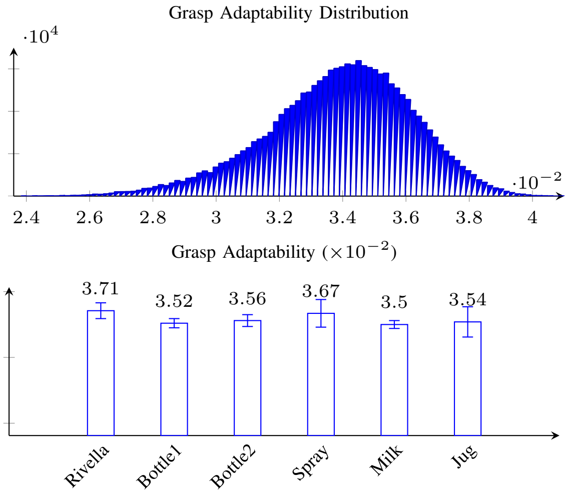


Fig. 11. *Upper:* Grasp adaptability distribution of 10^6 hand configurations in the reachability lookup table. *Lower:* Average grasp adaptabilities for the 100 grasps generated in the evaluations for all objects.

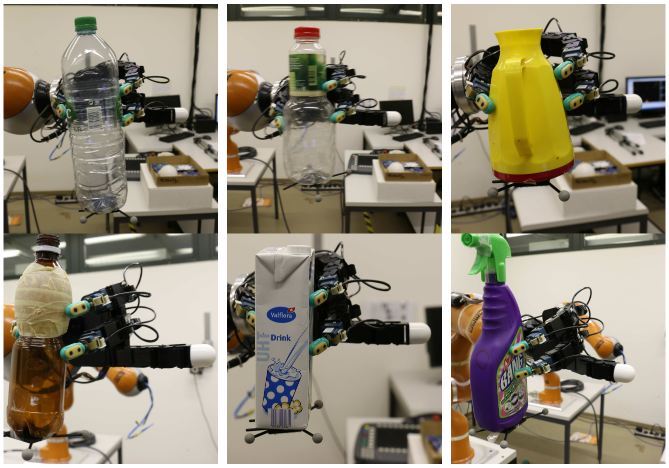


Fig. 12. Six example objects used in the evaluation: there is both variation in global geometry as well as local surface properties.

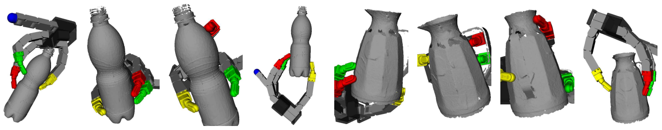


Fig. 13. Example grasps generated by Alg. 1 with $stopCondition(g)$ that, as soon as the grasp is stable and the reachability residual is smaller than 0.006, we stop optimizing on the current level of Λ_Φ and continue on the next level.

First, Fig. 14 shows that the number of levels of the graph G_Φ are between 4 and 5 when $T_p = 10$, or between 3 and 4 when $T_p = 40$. This indicates that our system produces similar depth of the *HFTS* independent of the shape of the object. However, the shape of the object affects the number of

[2] http://www.syntouchllc.com/

[3] http://www.naturalpoint.com/optitrack/

Object(#Units)	T_p	#Levels	#Nodes	Time(s)	SR(%)	$\theta(\times 10^{-2})$
bottle1(2736)	10	6	5672	1.81	96.15	5.23
bottle2(3102)		6	6321	1.77	93.46	4.74
jug(2671)		5	3227	1.64	89.29	4.96
rivella(2273)		6	5124	1.54	98.03	3.29
milk(2696)		5	3204	0.96	97.09	4.05
spray(3207)		6	6926	2.04	94.34	3.71
bottle1(2736)	40	3	2839	0.45	94.34	5.01
bottle2(3102)		3	3203	0.64	94.34	4.56
jug(2671)		4	3002	0.81	87.72	4.99
rivella(2273)		3	2399	0.44	99.01	3.17
milk(2696)		4	2912	0.73	96.15	4.12
spray(3207)		3	3310	0.62	92.59	3.62

Fig. 14. Evaluation of Alg. 1: #Units: number of fingertip units in $\Phi(\mathcal{P})$, #Levels: number of levels in graph G_Φ (including the top level with only one node), #Nodes: number of nodes in graph G_Φ, Time(s): average time in seconds for one run of the algorithm, SR: success rate of synthesizing a stable grasp. θ: optimized objective value defined in Eq. (15). The evaluations were implemented in Python on a machine with Ubuntu 12.04 running on Intel Core i7-2820QM 2.30GHz processors.

nodes at each level, given that some branches are terminated for objects of simpler geometry, such as the milk package. Regarding the success rate (SR), it shows the number of runs needed to generate 100 stable and collision free grasps on each object. We can see that SR lies at approximately 90%. Fig. 11 shows average adaptabilities for the 100 stable grasps for each object. Average adaptability values, computed by Eq. (12), are large showing that our methodology considers the adaptability effectively.

B. Grasp Adaptation

Once a grasp is executed and contacts established, the system will enter the post-grasping phase and start monitoring the stability based on tactile feedback. Now, the impedance controller is used to control the grasp using GMM based model Θ. Our tactile based grasp stability estimator is trained in the same way as in [9]. The log likelihood threshold for Eq. (18) is set to -100 for stable grasps as we found empirically that the object drops at a value below -130. For the force control of the hand, we set the initial grasp stiffness $K = (12, 2, 2)$ and use it for the execution of all grasps, as described in Sec. IV.

For the evaluation, we run two sets of experiments: 1) We continuously increase the objects' weight by filling them to evaluate the maximum weight each grasp can withstand, and 2) we shake the grasped objects by linearly increasing acceleration in different directions to evaluate the maximum acceleration each grasp can withstand. For comparison, we conduct the same experiments without any grasp adaptation and on the system proposed in [9] which does not consider object shape information when relocating fingertips.

1) Testing maximum weight: For each object, we execute the best out of 100 grasps generated in Sec. VI-A and align the object with vertical axis as shown in Fig. 18. We then gradually fill object with black pepper beans and record the maximum weight the grasp can withstand. The maximum weight is reached when the stability estimator predicts unstable grasp for more than 2 seconds or if the object drops. We repeat this test for each grasp 5 times and summarize the results in Fig. 15. For comparison, we perform the same tests for the system proposed in [9] and a system without adaptation.

Object	Weight	Without	With [9]	Improved
bottle1	34	55.1 ± 7.11	153.1 ± 12.31	165.3 ± 13.27
bottle2	39	62.8 ± 6.63	102.3 ± 13.38	121.3 ± 9.91
jug	112	125.3 ± 14.90	147.4 ± 9.62	162.1 ± 13.12
rivella	24	36.0 ± 6.96	76.5 ± 9.4	92.7 ± 7.45
milk	34	63.5 ± 8.20	151.8 ± 7.24	157.4 ± 8.35
spray	63	75.7 ± 7.21	102.2 ± 6.02	121.6 ± 7.15

Fig. 15. The comparison of the supported object weights(Unit:gram). **without**: without grasp adaptation; **with [9]**: with grasp adaptation in [9]; **improved**: the new adaptation approach in this paper.

Naturally, the system without any adaptation performs the worst and the integrated system outperforms the system from [9]. This is since our system: i) takes into account grasp reachability during the exploration, and ii) the new location is computed in the *HFTS*, thus ensuring it is valid, avoiding problems shown in Fig. 16, and iii) considers two fingers for gaiting, resulting in increased flexibility.

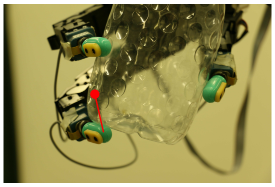


Fig. 16. The risk of moving a fingertip to an non-existing position present in [9] is addressed by using our HFTS representation. The red point shows the fingertip position before gaiting.

A quantitative evaluation of the proposed system and the system in [9] has been conducted with respect to optimization residual. We first execute the grasp in simulation and then trigger the fingertip gaiting by sending desired rest lengths randomly sampled around the current values within a ball of radius $20mm$. The result is shown in Fig. 17. Due to the object shape constraint, the systems cannot provide zero residuals. Our system performs much better for non-planar objects given that *HFTS* representation considers shape in an effective way.

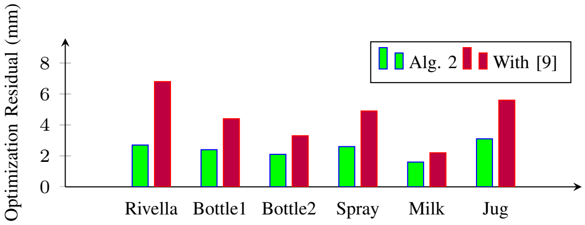


Fig. 17. The results of fingertip gaiting optimization residual. 10^5 desired rest lengths are randomly sampled around the current rest length within a ball of radius $20mm$.

An example of the supported weight test for the rivella bottle is shown in Fig. 18. In the beginning when the object is not too heavy, the likelihood $p(\hat{g}|\Theta)$ is larger than -100 and the grasp stiffness K is constant. As the weight increases, grasp becomes unstable and stiffness adaptation is initiated. Stiffness changes rapidly when the weight increases, and when the force adaptation is not able to handle the current weight, a finger gaiting is triggered and fingertip $F2$ is relocated. After finger gaiting, grasp stiffness is decreased since the new grasp requires less force to be stable. As the weight increases again, the whole process is repeated, resulting in $F1$ finger gaiting.

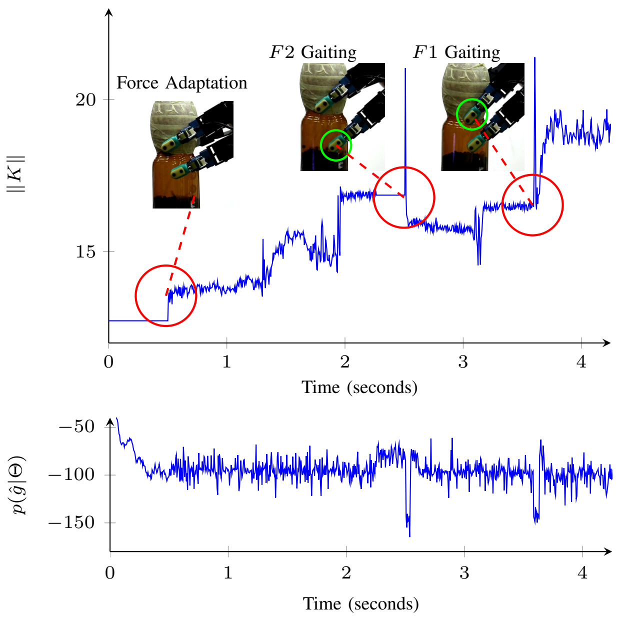


Fig. 18. A record of supported weight test of a grasp on rivella bottle. Upper: The norm of grasp stiffness and fingertip gaiting. Lower: Likelihood for grasp stability estimation defined in Eq. (18).

$J0$	$J1$	$J2$	$J3$	$J4$	$J5$	$J6$
$-30°$	$30°$	$2°$	$-60°$	$-20°$	$0°$	$-60°$

Init. $K = (K_x, K_y, K_z)$	Horizontal Acc.	Vertical Acc.
$(12, 2, 2)$	$2m/s^2$ – $8m/s^2$	$2m/s^2$ – $8m/s^2$

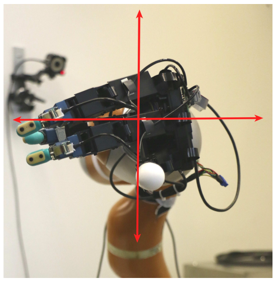


Fig. 19. The setup of grasp shaking test, in which the arm shakes each grasp in horizontal and vertical directions. $J0$ to $J6$ are the joint values for the initial pose of shaking test. When shaking horizontally, the shaking direction is fixed to be perpendicular to the palm.

2) Shaking Test: External disturbances, such as collision, may occur once a grasp has been executed. To evaluate the proposed system, we designed a shaking test. We first execute the best out of the 100 generated grasps for each object according to Eq. (15), and then rotate the arm to the configuration shown in Fig. 19. Thereafter, we start to shake the arm in either vertical or horizontal directions while linearly increasing the acceleration from $2m/s^2$ to $8m/s^2$. The shaking magnitude is limited to $10cm$ in either directions, which means that the hand is accelerating in the first $5cm$ and decelerating in the second $5cm$. After every period of shaking, we increase the acceleration by $1m/s^2$ and therefore have 14 shakes for every test.

Similarly to the supported weight test, we evaluate each grasp by measuring the maximum acceleration it can withstand. The criterion is similar: the maximum acceleration is recorded when the grasp is predicted as unstable for more than 2 seconds or if the object drops. The shaking test is conducted in both directions separately and on each object by filling it with $10g$, $20g$, $30g$, $40g$ and $50g$ black pepper beans. Each test is repeated 5 times. For comparison, we also conduct the same test for a system without adaptation and the system proposed in [9].

Experimental results are summarized in Fig. 20. If the maximum acceleration rate is $8m/s^2$, it means that the grasp has been kept stable during the test. On the other hand, if the maximum acceleration rate is $2m/s^2$, it means that the grasp could not withstand any shaking. We can see that our system outperforms both the system without the adaptation and the system proposed in [9]. The strength of our approach is that we ensure that the finger gaiting has resulted in an actual contact with the object which is not the case in [9]. In addition, the flexibility of gaiting two fingers provides additional strength.

Additional quantitative results are shown in Fig. 21. We can see that the average computing time of Alg. 2 is between $20ms$ and $40ms$. The average number of explored nodes shows that the pruning is efficient since less than 5% of all nodes in G_Φ are considered. Note that the computation time and number of explored nodes are heavily dependent on the connectivity of graph G_Φ: less nodes in the graph does not mean less computing time. Therefore, the connectivity in G_Φ indirectly measures how complex an object is in the context of this system.

VII. Conclusion

We have presented a unified framework for grasp planning and in-hand grasp adaptation using visual, tactile and proprioceptive feedback. The proposed Hierarchical Fingertip Space defines a hierarchy of surrogate solution spaces of fingertip grasping enabling both planning and adaptation. By augmenting the fingertip space in terms of local geometry and spatial relations, as well as optimizing hand configurations with respect to grasp adaptability, we demonstrated efficient planning and adaptation. Moreover, the probabilistic model for grasp stability estimation and adaptation has shown its feasibility in closing the loop between grasp replanning and control. We have evaluated the performance of the proposed system quantitatively and shown that the proposed system significantly improves the robustness of grasp execution. It also outperforms our previous work reported in [9]. To the best of our knowledge, this is so far the only system that accomplishes grasp synthesis, stability estimation, online replanning and in-hand adaptation in a unified framework, as well as evaluating this on a real physical system.

However, as a basic drawback of most learning based approaches, our probabilistic model is experience based, and hence relying on the training data, i.e. limited number of objects and examples to generalize from. As a potential future work, we plan to design an active learning strategy to update this model iteratively using new experiences over time, so as

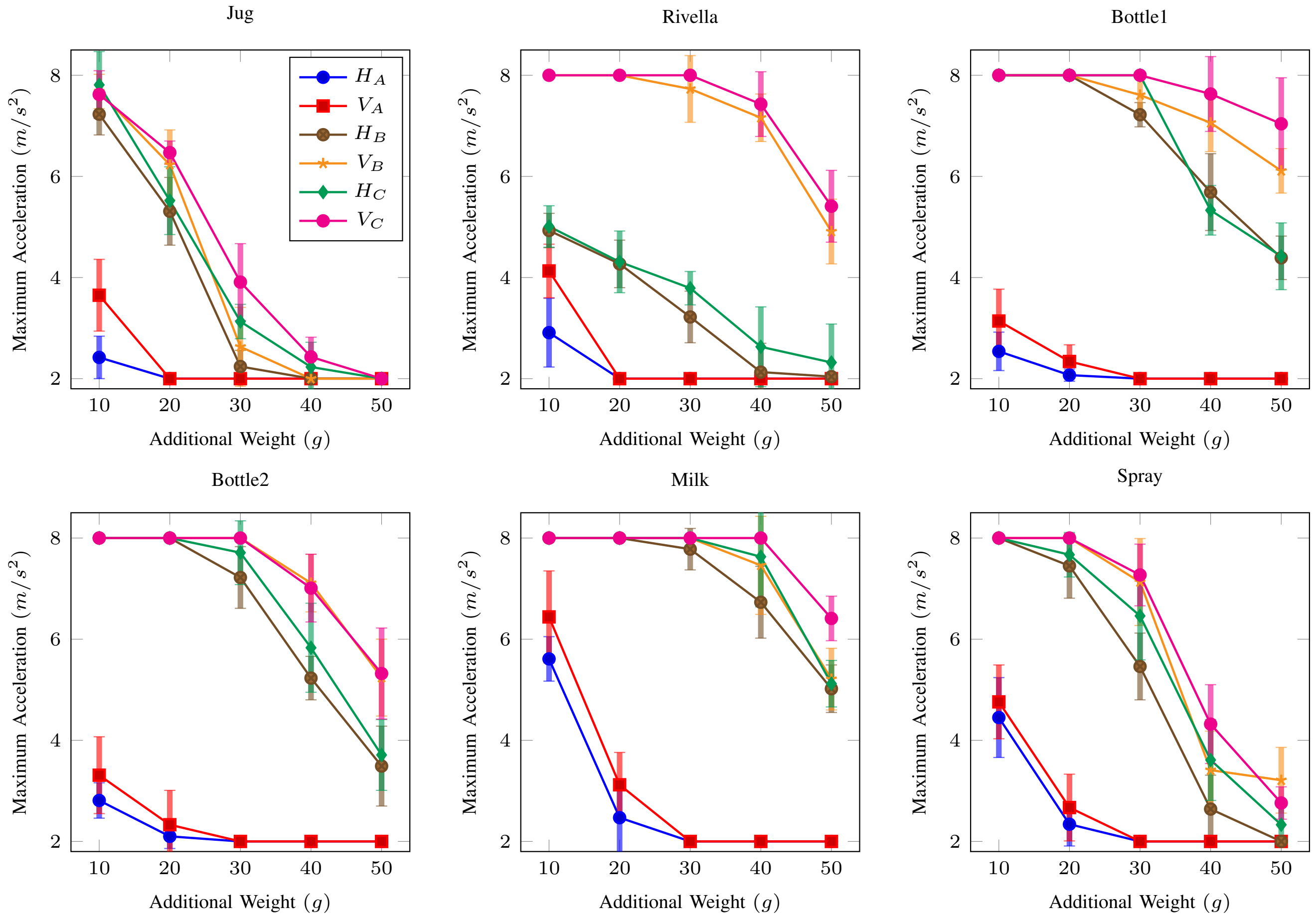


Fig. 20. Results of shaking tests on grasps shown in Fig. 12. In the legend, H and V refer to horizontal shaking test and vertical shaking test respectively. A, B and C refer to 3 grasp strategies: grasp without adaptation, grasp adaptation in [9] and the grasp adaptation proposed in this paper.

Object	Avg. Duration(ms)	Avg. Improvement	Avg. Comp. Time(ms)	Avg. Err.(m)	Avg. # Nodes	# Gaiting
bottle1	261.2	66.21	30.7	0.0074	279.2	6
bottle2	320.1	75.17	32.1	0.0062	221.5	4
jug	414.4	70.72	19.4	0.0042	140.4	5
rivella	447.9	52.39	38.6	0.0045	194.1	12
milk	392.6	47.11	24.9	0.0057	137.6	9
spray	502.7	57.26	29.2	0.0068	197.6	7

Fig. 21. Results for the horizontal shaking tests when the objects are filled with 20g of pepper beans (from left to right): average duration for one time of fingertip gaiting; Average stability likelihood improvement after fingertip gaiting; Average computation time of Alg. 2 for each computation; Average errors between achieved rest lengths and the rest lengths computed by Alg. 2; Average number of nodes explored in Alg. 2; Number of fingertip gaiting required during a shaking test with 14 shakes. The evaluations were implemented in C++ and run on a machine with Ubuntu 12.04 running on Intel Core i7-2820QM 2.30GHz processors.

to evolve the model in a long term to generalize it to a broader set of objects, without retraining the models from scratch.

Acknowledgement

This work was supported by the European Union Seventh Framework Program FLEXBOT - FP7-ERC-279933 and ROBOHOW - FP7-288533.

References

[1] K. Shimoga, "Robot grasp synthesis algorithms: A survey," *The International Journal of Robotics Research*, vol. 15, no. 3, pp. 230–266, 1996.

[2] C. Borst, M. Fischer, and G. Hirzinger, "Calculating hand configurations for precision and pinch grasps," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2002.

[3] K. Tahara, S. Arimoto, and M. Yoshida, "Dynamic object manipulation using a virtual frame by a triple soft-fingered robotic hand," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2010.

[4] T. Yoshikawa, "Multifingered robot hands: Control for grasping and manipulation," *Annual Reviews in Control*, vol. 34, no. 2, pp. 199 – 208, 2010.

[5] J. Romano, K. Hsiao, G. Niemeyer, S. Chitta, and K. Kuchenbecker, "Human-inspired robotic grasp control with tactile sensing," *IEEE Transactions on Robotics*, vol. 27, no. 6, pp. 1067–1079, Dec 2011.

[6] A. Sahbani, S. El-Khoury, and P. Bidaud, "An Overview of 3D Object Grasp Synthesis Algorithms," *Robot. Auton. Syst.*, vol. 60, no. 3, pp. 326–336, 2011.

[7] Y. Bekiroglu, D. Song, L. Wang, and D. Kragic, "A probabilistic framework for task-oriented grasp stability assessment," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2013.

[8] K. Hang, J. A. Stork, F. T. Pokorny, and D. Kragic, "Combinatorial optimization for hierarchical contact-level grasping," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2014.

[9] M. Li, Y. Bekiroglu, D. Kragic, and A. Billard, “Learning of grasp adaptation through experience and tactile sensing,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2014.

[10] J. Bohg, A. Morales, T. Asfour, and D. Kragic, “Data-driven grasp synthesis – a survey,” *IEEE Transactions on Robotics*, vol. 30, no. 2, pp. 289–309, 2014.

[11] K. Hang, J. A. Stork, and D. Kragic, “Hierarchical fingertip space for multi-fingered precision grasping,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2014.

[12] A. Bicchi and V. Kumar, “Robotic grasping and contact: A review,” in *Proc. IEEE Int. Conf. Robot. Autom.*, 2000.

[13] J. Aleotti and S. Caselli, “A 3d shape segmentation approach for robot grasping by parts,” *IEEE RAS*, vol. 60(3), pp. 358–366, 2012.

[14] M. Przybylski, T. Asfour, and R. Dillmann, “Planning grasps for robotic hands using a novel object representation based on the medial axis transform,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2011.

[15] N. Vahrenkamp, M. Przybylski, T. Asfour, and R. Dillmann, “Bimanual grasp planning,” in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2011.

[16] K. Huebner, “BADGrA toolbox for box-based approximation, decomposition and GRasping,” *Robot. Auton. Syst.*, vol. 60, no. 3, pp. 367 – 376, 2012.

[17] R. Pelossof, A. Miller, P. Allen, and T. Jebara, “An svm learning approach to robotic grasping,” in *Proc. IEEE Int. Conf. Robot. Autom.*, 2004.

[18] G. Biegelbauer and M. Vincze, “Efficient 3d object detection by fitting superquadrics to range image data for robot’s object manipulation,” in *Proc. IEEE Int. Conf. Robot. Autom.*, 2007.

[19] T. T. Cocias, S. M. Grigorescu, and F. Moldoveanu, “Multiple-superquadrics based object surface estimation for grasping in service robotics,” in *Optimization of Electrical and Electronic Equipment (OPTIM)*, 2012, pp. 1471–1477.

[20] D. Berenson, R. Diankov, K. Nishiwaki, S. Kagami, and J. Kuffner, “Grasp planning in complex scenes,” in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2007.

[21] J. A. Stork, F. T. Pokorny, and D. Kragic, “Integrated motion and clasp planning with virtual linking,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2013.

[22] F. T. Pokorny, J. A. Stork, and D. Kragic, “Grasping objects with holes: A topological approach,” in *Proc. IEEE Int. Conf. Robot. Autom.*, 2013.

[23] C. Borst, M. Fischer, and G. Hirzinger, “Grasping the dice by dicing the grasp,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2003.

[24] B. Mirtich and J. Canny, “Easily computable optimum grasps in 2-d and 3-d,” in *Proc. IEEE Int. Conf. Robot. Autom.*, 1994.

[25] J.-P. Saut and D. Sidobre, “Efficient models for grasp planning with a multi-fingered hand,” *Robot. Auton. Syst.*, vol. 60, no. 3, pp. 347 – 357, 2012.

[26] C. Rosales, J. Porta, and L. Ros, “Grasp optimization under specific contact constraints,” *IEEE Transactions on Robotics*, vol. 29, no. 3, pp. 746–757, June 2013.

[27] I. Gori, U. Pattacini, V. Tikhanoff, and G. Metta, “Three-finger precision grasp on incomplete 3d point clouds,” in *Proc. IEEE Int. Conf. Robot. Autom.*, 2014.

[28] C. Ferrari and J. Canny, “Planning optimal grasps,” in *Proc. IEEE Int. Conf. Robot. Autom.*, 1992.

[29] C. Rosales, J. Porta, R. Suarez, and L. Ros, “Finding all valid hand configurations for a given precision grasp,” in *Proc. IEEE Int. Conf. Robot. Autom.*, 2008.

[30] K. Hang, F. T. Pokorny, and D. Kragic, “Friction coefficients and grasp synthesis,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2013.

[31] M. Roa and R. Suarez, “Computation of independent contact regions for grasping 3-d objects,” *IEEE Transactions on Robotics*, vol. 25, no. 4, pp. 839–850, 2009.

[32] K. Hertkorn, M. Roa, and B. Ch, “Planning in-hand object manipulation with multifingered hands considering task constraints,” in *Proc. IEEE Int. Conf. Robot. Autom.*, 2013.

[33] C. Borst, M. Fischer, and G. Hirzinger, “Grasp planning: how to choose a suitable task wrench space,” in *Proc. IEEE Int. Conf. Robot. Autom.*, 2004.

[34] R. Platt, A. H. Fagg, and R. A. Grupen, “Null-space grasp control: Theory and experiments,” *IEEE Transactions on Robotics*, vol. 26, no. 2, Apr. 2010.

[35] C. Xiong and Y. Xiong, “Neural-network based force planning for multifingered grasp,” *Robot. Auton. Syst.*, vol. 21, no. 4, pp. 365 – 375, 1997.

[36] Y. Xia, J. Wang, and L.-M. Fok, “Grasping-force optimization for multifingered robotic hands using a recurrent neural network,” *IEEE Transactions on Robotics and Automation*, vol. 20, no. 3, pp. 549–554, June 2004.

[37] L. Han and J. Trinkle, “Dexterous manipulation by rolling and finger gaiting,” in *Proc. IEEE Int. Conf. Robot. Autom.*, 1998.

[38] J.-P. Saut, A. Sahbani, S. El-Khoury, and V. Perdereau, “Dexterous manipulation planning using probabilistic roadmaps in continuous grasp subspaces,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2007.

[39] Z. Li, P. Hsu, and S. Sastry, “Grasping and coordinated manipulation by a multifingered robot hand.” *The International Journal of Robotics Research*, vol. 8, no. 4, pp. 33–50, 1989.

[40] T. Yoshikawa and X.-Z. Zheng, “Coordinated dynamic hybrid position/force control for multiple robot manipulators handling one constrained object,” *The International Journal of Robotics Research*, vol. 12, no. 3, pp. 219–230, 1993.

[41] M. Buss, H. Hashimoto, and J. Moore, “Dextrous hand grasping force optimization,” *IEEE Transactions on Robotics and Automation*, vol. 12, no. 3, pp. 406–418, Jun 1996.

[42] T. Takahashi, T. Tsuboi, T. Kishida, Y. Kawanami, S. Shimizu, M. Iribe, T. Fukushima, and M. Fujita, “Adaptive grasping by multi fingered hand with tactile sensor based on robust force and position control,” in *Proc. IEEE Int. Conf. Robot. Autom.*, 2008.

[43] S. A. Schneider and R. H. Cannon, “Object impedance control for cooperative manipulation: theory and experimental results,” *IEEE Transactions on Robotics and Automation*, vol. 8, no. 3, pp. 383–394, 1992.

[44] H. Liu and G. Hirzinger, “Cartesian impedance control for the DLR hand,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 1999.

[45] K. Tahara, K. Maruta, A. Kawamura, and M. Yamamoto, “Externally sensorless dynamic regrasping and manipulation by a triple-fingered robotic hand with torsional fingertip joints,” in *Proc. IEEE Int. Conf. Robot. Autom.*, 2012.

[46] M. Li, H. Yin, K. Tahara, and A. Billard, “Learning object-level impedance control for robust grasping and dexterous manipulation,” in *Proc. IEEE Int. Conf. Robot. Autom.*, 2014.

[47] K. Hsiao, S. Chitta, M. Ciocarlie, and E. Jones, “Contact-reactive grasping of objects with partial shape information,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2010.

[48] K. Hsiao, L. Kaelbling, and T. Lozano-Perez, “Task-driven tactile exploration,” in *Proceedings of Robotics: Science and Systems*, 2010.

[49] A. Morales, M. Prats, P. Sanz, and A. P. Pobil, “An Experiment in the Use of Manipulation Primitives and Tactile Perception for Reactive Grasping,” in *Robotics: Science and Systems - Robot Manipulation: Sensing and Adapting to the Real World*, 2007.

[50] C. Zito, M. Kopicki, R. Stolkin, C. Borst, F. Schmidt, M. Roa, and J. Wyatt, “Sequential trajectory re-planning with tactile information gain for dexterous grasping under object-pose uncertainty,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, Nov 2013.

[51] H. Dang and P. K. Allen, “Grasp adjustment on novel objects using tactile experience from similar local geometry.” in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2013.

[52] H. Dang, J. Weisz, and P. Allen, “Blind grasping: Stable robotic grasping using tactile feedback and hand kinematics,” in *Proc. IEEE Int. Conf. Robot. Autom.*, 2011.

[53] Y. Bekiroglu, J. Laaksonen, J. A. Jorgensen, V. Kyrki, and D. Kragic, “Assessing grasp stability based on learning and haptic data,” *IEEE Transactions on Robotics*, vol. 27, no. 3, pp. 616–629, June 2011.

[54] E. L. Sauser, B. D. Argall, G. Metta, and A. G. Billard, “Iterative learning of grasp adaptation through human corrections,” *Robot. Auton. Syst.*, 2012.

[55] S. Koziel, D. Ciaurri, and L. Leifsson, “Surrogate-based methods,” in *Computational Optimization, Methods and Algorithms*, ser. Studies in Computational Intelligence. Springer Berlin Heidelberg, 2011, vol. 356, pp. 33–59.

[56] C. Walshaw, “Multilevel refinement for combinatorial optimisation problems,” *Annals of Operations Research*, vol. 131, no. 1-4, pp. 325–372, 2004.

[57] N. Sommer, L. Miao, and A. Billard, “Bimanual compliant tactile exploration for grasping unknown objects,” in *Proc. IEEE Int. Conf. Robot. Autom.*, 2014.

[58] F. T. Pokorny and D. Kragic, “Classical grasp quality evaluation: New theory and algorithms,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2013.

[59] S. El Khoury, M. Li, and A. Billard, “Bridging the gap: One shot grasp synthesis approach,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, Oct 2012, pp. 2027–2034.

[60] B. Huang, S. El-Khoury, M. Li, J. Bryson, and A. Billard, “Learning a real time grasping strategy,” in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2013, pp. 593–600.

[61] T. Yoshikawa, “Manipulability of robotic mechanisms,” *The International Journal of Robotics Research*, 1985.

[62] L. Guilamo, J. Kuffner, K. Nishiwaki, and S. Kagami, “Efficient prioritized inverse kinematic solutions for redundant manipulators,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2005.

[63] Y. Aydin and M. Nakajima, “Database guided computer animation of human grasping using forward and inverse kinematics.” *Computers & Graphics*, vol. 23, pp. 145–154, 1999.

[64] M. Powell, “A direct search optimization method that models the objective and constraint functions by linear interpolation,” in *Advances in Optimization and Numerical Analysis*, ser. Mathematics and Its Applications. Springer Netherlands, 1994, vol. 275, pp. 51–67.

Kaiyu Hang received the B.S. degree in information and communication engineering from Xi’An Jiaotong University, Xi’an, China in 2010 and the M.Sc. degree in communication systems from KTH Royal Institute of Technology, Stockholm, Sweden in 2012.

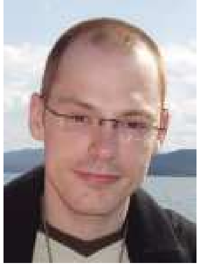
He is currently a Ph.D student at the Centre for Autonomous Systems and the Computer Vision and Active Perception Lab at KTH Royal Institute of Technology, Stockholm, Sweden. His research include robotic precision/fingertip grasping, object-hand integrated representation, adaptive grasping and grasp manifold learning.

Miao Li received the Bachelor and Masters degree both from the School of Mechanical Science and Engineering, Huazhong University of Science and Technology(HUST), China, in 2008 and 2011 respectively. From Sep. 2011, he joined thecurrent Department of Mechanical Engineer-ing at cole Polytechnique Fdrale de Lausanne (EPFL) in Switzerland, as a doctoral assistant. His current research interests mainly focus on robust robotic grasping, impedance control of dexterous manipulation using tactile sensing.

Johannes A. Stork received the B.S. degree in computer science from Albert Ludwigs University of Freiburg, Freiburg, Germany, in 2009 and the M.S. degree in computer science from Albert Ludwigs University of Freiburg, Freiburg, Germany, in 2012. He is currently pursuing the Ph.D. degree in computer science and robotics at the School of Computer Science and Communication, KTH Royal Institute of Technology, Stockholm, Sweden.

From 2009 to 2012, he was a Research Assistant with the Social Robotics Lab, Albert Ludwigs University of Freiburg, Freiburg, Germany. His research interest include representation learning, robotic in-hand manipulation, robotic grab synthesis.

Yasemin Bekiroglu received the M.Sc. degree in computer engineering from Karadeniz Technical University, Trabzon, Turkey, and the M.Sc. degree in applied artificial intelligence from Dalarna University, Dalarna, Sweden. She completed her Ph.D. studies in the Computer Vision and Active Perception Lab (CVAP), in the School of Computer Science and Communication, at the Royal Institute of Technology (KTH), in Stockholm, Sweden, in 2012. Currently, she is a postdoctoral researcher at CVAP. Her research interests include robotics, computer vision and machine learning.



Florian T. Pokorny is a Postdoctoral Researcher at KTH Royal Institute of Technology. He is interested in robotic manipulation and machine learning methods. He received his PhD in mathematics from the University of Edinburgh and, before that, he completed Part III of the mathematical tripos at the University of Cambridge and obtained a BSc (first class) in Mathematics from the University of Edinburgh.

Aude Billard is Professor of Micro and Mechanical Engineering, and the head of the LASA Laboratory at the School of Engineering at the Swiss Federal Institute of Technology in Lausanne. She received a M.Sc. in Physics from EPFL (1995), a MSc. in Knowledge-based Systems (1996) and a Ph.D. in Artificial Intelligence (1998) from the University of Edinburgh. She was the recipient of the Intel Corporation Teaching award, the Swiss National Science Foundation career award in 2002, the Outstanding Young Person in Science and Innovation from the Swiss Chamber of Commerce and the IEEE-RAS Best Reviewer award in 2012. Aude Billard served as an elected member of the Administrative Committee of the IEEE Robotics and Automation society (RAS) for two terms (2006-2008 and 2009-2011) and is the chair of the IEEE-RAS Technical Committee on Humanoid Robotics. Her research interests focus on machine learning tools to support robot learning through human guidance. This extends also to research on complementary topics, including machine vision and its use in human-robot interaction and computational neuroscience to develop models of motor learning in humans.

Danica Kragic (SM12) received the M.Sc. degree in mechanical engineering from the Technical University of Rijeka, Rijeka, Croatia, in 1995 and the Ph.D. degree in computer science from the Royal Institute of Technology (KTH), Stockholm, Sweden, in 2001. She is currently a Professor with the School of Computer Science and Communication, KTH. Her research include computer vision, object grasping and manipulation, and humanrobot interaction. Her recent work explores different learning methods for formalizing the models for integrated representation of objects and actions that can be applied on them. This work has demonstrated how robots can achieve scene understanding through active exploration and how full body tracking of humans can be made more efficient.

Dr. Kragic received the 2007 IEEE Robotics and Automation Society Early Academic Career Award. She is a member of the Swedish Royal Academy of Sciences and the Swedish Young Academy. She has chaired the IEEE Robotics and Automation Society (RAS) Technical Committee on Computer and Robot Vision and, since 2009, has served as an IEEE RAS Administrative Committee member.