

Velvet Fingers: Grasp Planning and Execution for an Underactuated Gripper with Active Surfaces

Robert Krug^{*}, Todor Stoyanov^{*}, Manuel Bonilla[†], Vinicio Tincani[†], Narunas Vaskevicius[‡],
Gualtiero Fantoni[†], Andreas Birk[‡], Achim Lilienthal^{*} and Antonio Bicchi[†]

Abstract—In this work we tackle the problem of planning grasps for an underactuated gripper which enable it to retrieve target objects from a cluttered environment. Furthermore, we investigate how additional manipulation capabilities of the gripping device, provided by active surfaces on the inside of the fingers, can lead to performance improvement in the grasp execution process. To this end, we employ a simple strategy, in which the target object is ‘pulled-in’ towards the palm during grasping which results in firm enveloping grasps. We show the effectiveness of the suggested methods by means of experiments conducted in a real-world scenario.

I. INTRODUCTION

Prior work in the field of robotic grasping has produced a multitude of different gripper designs in an attempt to achieve reliable grasps of various target objects. One line of research has focused on creating devices mimicking the mechanical structure of the human hand in order to allow grasping/manipulation of objects with a wide range of shapes and sizes [1]. This, however, results in complex designs and control schemes. An alternative approach is to simplify the design and/or control and tailor it to preserve some specific desired grasping or manipulation features [2].

Another recently investigated way to achieve more dexterous grippers is the addition of active surfaces to otherwise simple mechanical structures. Active surfaces have been used to regulate the adhesion between fingers and target object [3], or to augment the mechanical structure with conveyor belts to control the tangential push exerted on the target object. Currently, the three most advanced implementations of such grippers are: the Roll-on gripper [4], the Traction gripper [5] and the Velvet Fingers gripper [6], [7], the latter of which was utilized in this work and is depicted in Fig. 1. The Velvet Fingers gripper combines underactuation and active surfaces in the form of conveyor belts on the finger phalanges. The mechanical design (discussed in more detail in Section IV-A) features one actuated degree of freedom for opening and closing and two for the belt movements. If, during grasping, the proximal phalanges are blocked by the object, the gripper’s distal phalanges continue to ‘wrap-around’ the object and envelope it in a firm grasp.

In this work, we investigate two questions which arise when using such a device in a grasping scenario where a



Fig. 1. *Platform and test objects*: The platform utilized in the test runs in Section IV comprises the Velvet Fingers gripper with three actuated DoF (one for open/close movement and one per finger for conveyor belt actuation) and a seven-DoF KUKA lightweight arm. Perception is done with an ASUS Xtion structured light camera mounted on the gripper. Also shown are the five test objects used in the experiments - two different boxes, a beer barrel, a ball and a drum.

cluttered scene containing multiple objects is to be cleared by a robot:

- How to account for an underactuated gripper structure in the grasp synthesis?
- How can active surfaces contribute to the grasp execution process?

Grasp synthesis is the process of determining the gripper’s joint configuration and wrist pose with respect to the target object, such that a successful grasp execution is ensured. In this work, we opt for a data-driven approach to grasp synthesis, where a knowledge base is populated offline with target object models which are associated to pre-planned grasps. During execution, the database is used to retrieve grasps and rank them according to the expected quality in the current scene.

The contributions of this work are two-fold: first, we address the grasp planning problem by adapting the well known optimization based planning scheme introduced by Ciocarlie and Allen [8] to the specifics of underactuated gripper designs. Second, we investigate a method to improve the success of the grasp execution process by using the gripper’s active surfaces. We employ a simple ‘pull-in’ strategy where the target object is simultaneously manipulated and grasped by using the belts to pull the object towards the gripper’s palm while closing the fingers.

^{*} AASS Research Center; Örebro University; Fakultetsgatan 1, 70182 Örebro, Sweden.

[†] Interdepart. Research Center “E. Piaggio”; University of Pisa, Via Diotisalvi 2, 56100 Pisa, Italy.

[‡] Robotics Group, School of Engineering and Science; Jacobs University Bremen; Campus Ring 1, 28725 Bremen, Germany.

This article is organized as follows: below, we briefly review related work before we introduce our grasp planning- and execution scheme in Section III. In Section IV we describe the experimental setup, the target scenarios and the obtained results before we draw conclusions in Section V.

II. RELATED WORK

One way of categorizing grasp planning methodologies is to separate them into *analytic* and *data-driven* approaches. Analytic methods commonly construct geometrically stable force-closure grasps [9], [10] by defining the precise fingertip locations on the target object. Often, they rely on precise knowledge of hand kinematics, object geometry and the relative pose of hand and object (see [11] for a review). On the other hand, data-driven approaches generate grasp hypotheses for a given object in a knowledge database and usually provide a less specific grasp definition (*e.g.* only the approach vector [12]) which, combined with appropriate heuristics for grasp execution, is often more robust to the uncertainties inherent in a robotic system. To simplify the grasp synthesis and subsequent retrieval from the database, it has been suggested to approximate the target object with primitives or superquadrics [13], [14], [15]. We refer to Bohg et. al. [16] for a more complete recent review on data-driven grasp synthesis approaches.

A commonly used strategy to compute grasp hypotheses is to sample the target object’s surface or bounding-box normals [17], [18] and to use them as approach vectors in a simulation where the fingers are closed once the gripper’s palm contacts the object. A wrench-based geometric quality criterion, such as the one introduced by Ferrari and Canny [19], is usually used to rank the grasps. Alternatively, suitable pre-grasps can be created by minimizing an appropriate energy function as demonstrated in [8]. Again, the final grasp quality evaluation is usually done after auto-closure of the fingers in a static simulation (*i.e.*, using a spatially fixed object and only performing forward kinematics and collision checks while ignoring interaction forces) [8], [20]. However, for underactuated simple grippers this strategy is unsuitable because it fails to accurately predict the final grasp configuration which depends on the interaction between gripper and object.

Grasp planning and execution in cluttered scenes is subjected to intrinsic difficulties since many pre-planned grasps are not reachable in such environments. Berenson et. al. [21] address this problem by online computation of a grasp score based on heuristics. In [22], pushing actions are used to manipulate otherwise ungraspable objects. Saxena et. al. [23] present a vision-based approach which accounts for uncertainty in the target object’s location during planning and grasp selection.

The use of active surfaces regulating adhesion for gripping devices has so far been only explored on the micro/nano scale [3]. Some designs of grippers utilizing belts have been employed in industrial settings but, to the best of our knowledge, no attempt has been made so far to develop

a programmatic strategy of utilizing active surfaces during grasp execution.

III. GRASP PLANNING AND EXECUTION

Underactuated grippers can only control their active joints and rely on physical interaction with the environment for reconfiguring the passive ones. Thus, accurately accounting for the interaction between such a gripper and a target object in the grasp synthesis process would require a detailed dynamic simulation of the grasp procedure. Even then, grasps pre-planned in this fashion might fail when applied in cluttered environments, since a physical simulation of the current scene would be necessary to guarantee successful target object retrieval.

Here, we chose a different strategy aimed at exploiting the underactuated structure and the active surfaces of the Velvet Fingers gripper to simplify grasp synthesis. Therefore, for planning purposes, we assume extended distal links and a fully actuated mechanical structure comprising only one proximal joint α connecting the fingers to the palm (see Section IV-A for an overview of the actual kinematic structure). These assumptions conform to the natural configurations of the gripper during approach (see Fig. 1 for an exemplary approach configuration) when no mechanical interaction with the environment occurs. During grasp execution, we then rely on the specific underactuated kinematic and transmission design to envelope the object in a stable grasp or use the belts to pull the object into such an enveloping grasp as described in Section IV-C. Hence, the goal of the presented grasp planning methodology is to find appropriate gripper wrist poses and opening angles α .

Our grasp planning methodology is built upon the work by Ciocarlie and Allen [8], who generate pre-grasps in low-dimensional hand joint subspaces by minimizing an energy function based on distance/alignment between pre-defined contact locations on the hand and the target object. The corresponding non-convex optimization problem is solved via a simulated annealing algorithm which has been shown to work well for synthesizing enveloping grasps for multi-fingered hands [24], [20]. The Velvet Gripper’s relatively large size, and the fact that target objects can only be approached with extended distal links, makes many grasps with inappropriate wrist poses infeasible since they would result in collisions with the environment. Consequently, we additionally consider the following two observations in our planning framework:

- i) It has been shown, that most successful grasps usually approach along a surface normal of the object [21], [25].
- ii) Preliminary experiments showed that in many successful grasps the gripper’s lateral axis (the x -axis in Fig. 2) is normal to one of the principal component directions of the object.

In accordance with the first observation, we constrain the approach direction (the y -axis in Fig. 2) to be along a surface normal \mathbf{n} of the object and plan wrist poses $\mathbf{P} = \mathbf{P}(d, \phi, \psi) \in \mathbb{R}^3$ over the approach distance d and wrist

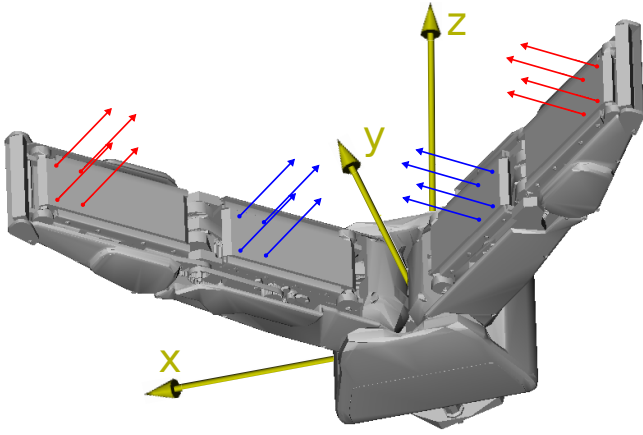


Fig. 2. *Preset contacts*: Shown are the preset contact locations and corresponding surface normals $\hat{\mathbf{n}}$ on the Velvet Fingers gripper. Contacts depicted in blue result in enveloping grasps, for the synthesis of fingertip grasps the set of contacts shown in red were used.

roll and pitch angles ϕ and ψ . To this end, we minimize the following energy function which was introduced in [8]

$$E(\mathbf{P}, \alpha) = \sum_{c=1}^{n_C} \left(1 - \frac{\hat{\mathbf{n}}_c^T \mathbf{o}_c}{\|\mathbf{o}_c\|_2} + \frac{\|\mathbf{o}_c\|_2}{s} \right), \quad (1)$$

where n_C indicates the number of preset target contact locations on the phalanges of the gripping device (see Fig. 2), $\hat{\mathbf{n}}_c$ denotes the outward-pointing surface unit normal at a preset contact, \mathbf{o}_c is the vector from contact c on the hand to the closest point \mathbf{p} on the object and s is a scaling parameter (for clarity, the implicit dependence of \mathbf{o}_c and $\hat{\mathbf{n}}_c$ on \mathbf{P} and α was omitted in the notation). The first term in Eq. (1) captures the alignments between the object and target contact locations on the hand, the second term indicates the distances of the target contacts to the object. As in [8], the grasp energy in (1) is minimized using the simulated annealing algorithm which is able to escape local minima through possible “uphill moves” during optimization via generating random neighbors.

Given a vertex \mathbf{p} with associated vertex normal \mathbf{n} on the target object’s discretized surface, we account for the second observation by providing the solver with appropriate initial conditions. To this end, the gripper is positioned at a fixed offset distance from \mathbf{p} along the negative approach direction which coincides with the normal \mathbf{n} . The initial wrist rotation is determined such that the gripper’s lateral axis is normal to the object’s principal component direction with the largest eigenvalue which is not parallel to the approach vector. This rotation is then offset by an initial roll angle ϕ_0 . During optimization, box constraints are enforced to ensure that the variables do not diverge too far from the initial condition. The proposed method allows to put bounds on the resulting grasp poses while, opposed to approaches solely relying on heuristic sampling [21], [18], still retains the flexibility to adjust them to the specific object geometry.

Our grasp planning methodology is summarized in Algorithm 1 and was integrated in the GraspIt! [24] simulator, using GraspIt!’s simulated annealing solver. We iteratively

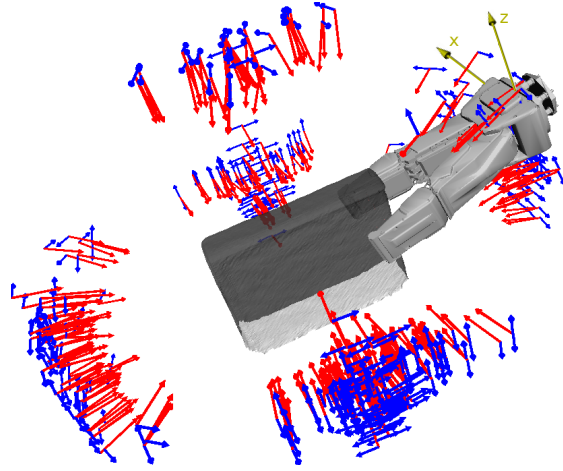


Fig. 3. *Planning example*: Shown is a set of fingertip grasps as computed by the planner. Wrist poses are indicated by the approach direction (y -axis) in red and the x -axis in blue. As intended, the resulting wrist orientations are roughly normal to principal component directions of the target object.

Algorithm 1: Grasp planning

Input: Discretized object represented as a set of vertex/vertex normal tuples $\{(\mathbf{p}, \mathbf{n})\}$, Initial wrist rotation discretization n_J , Desired number of grasps n_G , Maximum grasp energy E_{max}

Output: Grasp poses $\{\mathbf{P}\}$, opening angles $\{\alpha\}$

```

g = 0
while g < n_G do
  /* Sample a vertex and associated vertex normal
  without replacement */
  (p, n) = randomSample({(p, n)})
  /* Initialize the solver from n_J initial states */
  for j = 1, ..., n_J do
    /* Compute the initial roll angle phi_0 */
    phi_0 = (j - 1)2pi/n_J
    orientGripper(p, n, phi_0)
    minimize E in (1)
      P, alpha
    subject to box constraints on all variables

    /* Add the solution to the output set if it
    satisfies the energy threshold */
    if E <= E_max then
      add P to {P} and alpha to {alpha}
  g = g + 1

```

sample vertex/vertex normal tuples from a discretized target object representation. Subsequently, the solver is restarted with initial states as described above each time a grasp is found. In Algorithm 1, n_J grasp hypotheses are computed for each approach direction. To obtain grasps conforming to other principal directions and to obtain symmetrical grasp hypotheses, $n_J = 4$ is typically chosen since this advances the initial roll angle ϕ_0 in steps of size $\pi/2$. Our optimization

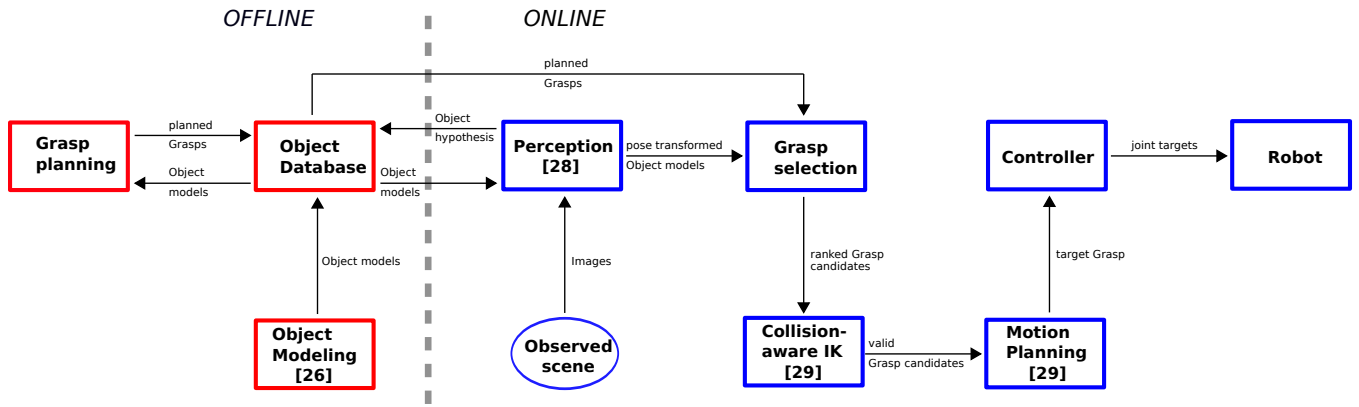


Fig. 4. *Grasping pipeline*: The pipeline uses an offline created database storing object models together with grasps provided by the grasp planner. The perception module creates object hypotheses from the observed scene and matches them to the models in the database. Found models are forwarded to the grasp selection module which ranks the associated grasps according to their score in the current scene. Subsequently, inverse kinematics and goal pose collision checks are performed on a chosen grasp before a motion plan is generated which is executed by the controller.



Fig. 5. *Pull-in grasping strategy*: Depicted is a sequence of intermediate grasp states where the gripper’s belts are used to pull the object towards the palm which results in a transition from a fingertip to an enveloping grasp.

scheme only operates on four decision variables which results in fast planning times. As a grasp quality metric for ranking, we use the value of the energy function in (1). The use of wrench-based metrics is avoided, since they have shown to be rather fragile in practice [18] especially when planning on imperfect object models reconstructed from sensor data as the ones we use in this work as detailed in Section III-B.

In Section IV we evaluate our planner with two different sets of contact references as illustrated in Fig. 2. The choice of reference locations on the gripper provides an easy way to control the resulting grasp configurations. Locations on the proximal phalanges result in enveloping grasps, which are potentially robust. However, in cluttered scenes many enveloping grasps are not achievable without collisions and fingertip grasps are preferable, especially when active surfaces are available to aid the grasp execution process as discussed below. An example for the planners output is depicted in Fig. 3.

A. Simultaneous Manipulation and Grasping

An interesting possibility offered by the active surfaces of the considered gripper is to manipulate the object while grasping. One idea to improve the grasp execution success in the presence of object pose and gripper positioning uncertainties is to employ a ‘pull-in’ strategy. Here, the belts move the object towards the gripper’s palm while the phalanges squeeze the object as illustrated in Fig. 5. In section IV-C, we provide a proof-of-concept verification of this strategy in combination with fingertip grasps in cluttered scenes. Here, enveloping grasps are often infeasible and the ‘pull-

in’ strategy aids in obtaining firm grasps which would not be achievable without the active surfaces.

B. Grasping Pipeline

To carry out the experiments in Section IV we employ the grasping pipeline illustrated in Fig. 4. Our approach employs an offline and an online stage. In the offline stage, we acquire 3D models of the target objects, train a perception module and compute grasps. The first step in the offline stage is to acquire accurate models of the objects of interest. The model acquisition approach follows the work of Mihalyi et al. [26]. First, a number of augmented reality markers are placed in the scene and a set of training RGB-D images is collected. The training set is used to estimate a graph of marker positions and orientations. Next, we successively place each target object in the scene and acquire a set of RGB-D images, which are subsequently registered in a common reference frame using the marker graph. We use the registered depth images to reconstruct a Truncated Signed Distance Field (TSDF) representation of the object from which triangular meshes are extracted using the marching tetrahedrons algorithm. The TSDF tracker algorithm in [27] is employed to correct for local inaccuracies in the RGB-D data registration. By fusing consecutive views of the object in the TSDF representation we effectively eliminate a large portion of the sensor noise, leveraging on the changing sensor viewpoints. The smoothed triangle meshes are then stored in the database and used by the previously described grasp planner to generate grasps. Thus, contrary to prior approaches that use ground truth geometric models, we train grasps on models reconstructed from noisy sensor observations. The final component of the offline stage uses the RGB-D images of each object to train the recognition modules of the perception system. The Perception module in the pipeline is based on previous work by Vaskevicius et al. [28]. During the offline stage, this system extracts local visual features from the RGB data component, references them based on the depth component and stores the resulting feature graph in the database.

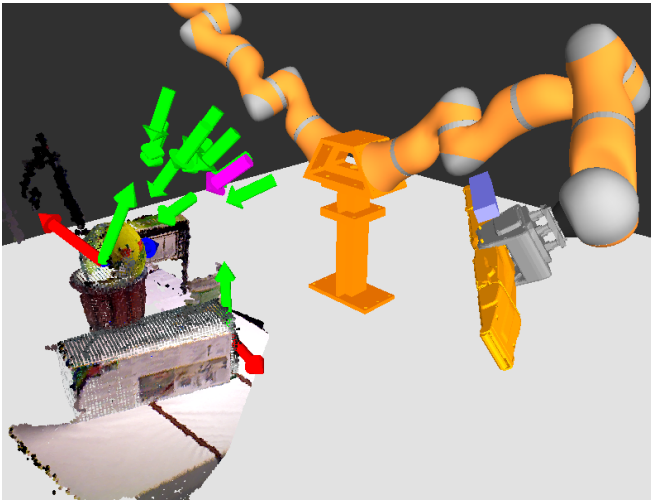


Fig. 6. *Grasp retrieval*: Shown is a scene as observed by the robot with two detected objects (ball and box). Also depicted are the feasible grasps \mathcal{G}^* for the ball which are indicated by their approach vectors in green, the purple vector signifies the grasp chosen by the grasp selection module.

The online stage of the proposed pipeline starts with the acquisition of an RGB-D image of the target scene. Following the work in [28], the image is then over-segmented in patches. Local visual features from each patch are then extracted and compared against the feature graphs stored in the database. If a candidate match to an object is detected, additional checks for consistency are performed by back-projecting the database object to the scene. Once the perception system obtains a list of detected objects, the grasp selection module is used to associate a set of grasps to each of the pose-transformed object candidates. For evaluation purposes, as described in Section IV-B, this module tests all grasps from the database for feasibility, using a fast RRT planner. An example is shown in Fig. 6. Given a high number of objects and possible grasps, this strategy may not be feasible for online operation. To this end, we employ a grasp ranking procedure and a first-feasible execution strategy. We follow an approach similar to the one outlined by Berenson et. al. [21] and compute, for the given environment, a composite score for each grasp by summing the following two parts:

- grasp energy score
- gripper-relative position score

Here, the grasp energy score is the value of the grasp energy in (1) for the current grasp configuration, normalized by the largest energy value over all grasps for an object. The gripper-relative position score captures the similarity of the grasp pose to the current gripper pose and is expressed as one minus the cosine of the angle between the current and target wrist orientation. All grasps are ranked by increasing scores and successively tested for feasibility as described above. Once a valid candidate is found, joint motion trajectories are generated and passed on to the controllers to execute the movement. In all experiments, inverse kinematics, collision checking and motion planning were carried out with the MoveIt! framework [29].

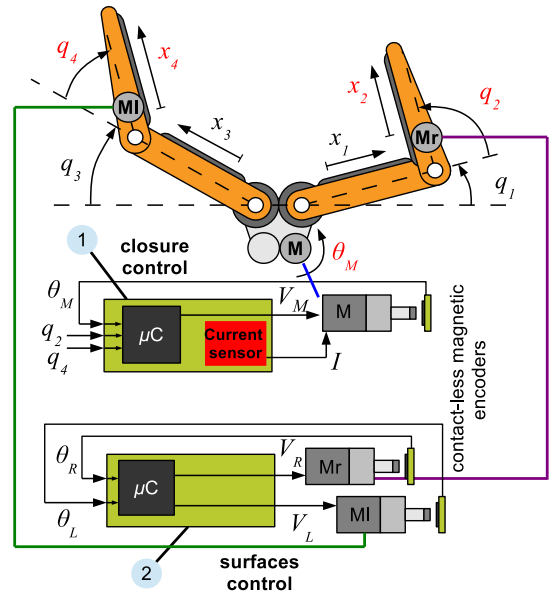


Fig. 7. *Kinematics and Control architecture of the Velvet Fingers gripper*: One electronic board controls the motor M for the opening and closing, reads the angular position of the second joints q_2 and q_4 and limits the current absorption of the actuator. The second board controls the motors M_r and M_l driving the belts.

IV. EVALUATION AND RESULTS

In this section, we outline the hardware setup and target scenarios, before proceeding with a discussion of the obtained results.

A. System configuration and Target Scenarios

In order to evaluate the suggested grasping pipeline, two sets of experimental evaluations were performed. All experiments were carried out on the platform depicted in Fig. 1, consisting of a Velvet Fingers gripper mounted on a joint impedance controlled KUKA lightweight robot arm. Each of the gripper's two fingers has a planar manipulator structure with two joints plus two coupled conveyor belts which implement the active surfaces. The kinematics and control scheme of the Velvet Fingers gripper is depicted in Fig. 7. All the actuators are controlled with simple PID control loops which are closed on the angle rotation of the motor shafts through magnetic encoders. A current sensor on the electronic board controlling the opening/closing actuator allows to set a threshold on the current absorption. This ensures a robust grasping behavior and, at the same time, enables safeguarding the entirety of the gripper (see [7] for more details). For all object recognition and collision detection tasks an ASUS Xtion structured light camera, which is mounted on the gripper, is used.

In the experiments, we used a database containing the five target objects shown in Fig. 1. We opted for a two-staged evaluation. In the first set of experiments we investigate the quality of the proposed grasp planning approach under different environment configurations. We then proceed to analyze the performance of the system in a tabletop grasping scenario including a preliminary performance analysis of the proposed 'pull-in' strategy.

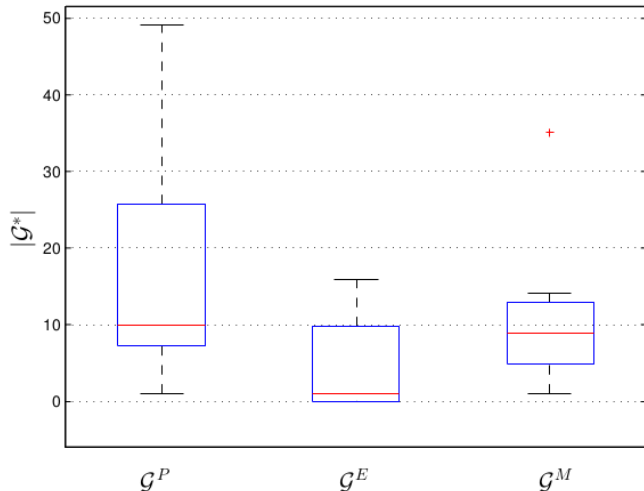


Fig. 8. *Planning results - isolated objects*: Boxplot showing the number of feasible grasps $|\mathcal{G}^*|$ per object (out of a total number of 400 pre-planned grasps) as a function of the chosen planning flavor (pinch, envelope or mixed grasps). The grasps were extracted from data set \mathcal{S}^I which contains observations of isolated target objects only.

B. Grasp Planning Evaluation

To evaluate the planning, we computed three sets of 400 grasps for each object, containing pinch grasps \mathcal{G}^P , enveloping grasps \mathcal{G}^E and an equal mixture of both \mathcal{G}^M respectively. Next, the experimental platform was used to collect two data sets of depth and color images. The first scene data set \mathcal{S}^I contains three observations of each target object in isolation (*i.e.*, only one object in the scene). The second data set \mathcal{S}^C contains observations of scenes with various amount of clutter (*i.e.*, multiple objects in the scene). Here, a total of fifteen different scenes with all five objects, which are incrementally cleared by a human removing one object at a time, were recorded resulting in a total of sixty observed scenes containing at least two objects each. For each observed scene we executed our grasping pipeline using the differently planned grasp sets. In these sets of experiments we evaluate all grasps up to the motion planning module, but do not select or execute a grasp since we aim to evaluate how many of the planned grasps are feasible (*i.e.*, reachable by a collision free path) under different conditions. The predefined parameters for the box constraints, wrist rotation discretization and energy threshold used in Algorithm 1 are summarized in Table I.

The outcome of the first experiment is visualized in Fig. 8. It is clear that pinch grasps are much more likely to be feasible, even if only a single target object is in the robot’s workspace. Many enveloping grasps are rejected because they necessitate large opening angles resulting in bulky gripper silhouettes for which no collision free approach trajectories can be found.

Table II shows the feasible grasps per object depending on the number of objects in the cluttered scene and the chosen planning flavor (pinch, envelope or mixed grasps). We note that two out of the 15 scenes containing all five objects yielded an exceptionally high number of feasible pinch grasps, which biased the according entry in Table II.

TABLE I

PLANNING PARAMETERS: *Predefined parameters for Algorithm 1*

$\Delta\phi$ [rad]	$\Delta\psi$ [rad]	Δd [m]	n_J	E_{max}
$\pm \pi/10$	$\pm \pi/10$	± 0.4	4	20

TABLE II

PLANNING RESULTS - CLUTTERED SCENES: *Mean and 1-STD values of the number of feasible grasps $|\mathcal{G}^*|$ per object depending on the planning flavor (pinch, envelope or mixed) and the number of objects in the scene.*

	$ \mathcal{G}^{*P} /\text{Obj}$	$ \mathcal{G}^{*E} /\text{Obj}$	$ \mathcal{G}^{*M} /\text{Obj}$
5 Obj	7.5 ± 12.3	1.3 ± 3.4	3.3 ± 8.0
4 Obj	4.7 ± 6.2	1.4 ± 3.2	2.1 ± 3.5
3 Obj	5.1 ± 8.9	1.5 ± 2.5	2.1 ± 4.2
2 Obj	8.4 ± 9.3	4.0 ± 5.7	4.6 ± 8.0

This is due to the fact, that the number of feasible grasps found also significantly depends on the location of the objects in the robot’s workspace which favored pinch grasps in these cases. Nevertheless, it is evident that the difficulty of finding feasible enveloping grasps increases with the amount of clutter, whereas it still possible to obtain pinch grasps with multiple objects in the scene. Also, the set of mixed grasps does not perform significantly better than enveloping grasps. These results provide a strong motivation to exclusively use dexterous initial pinch grasps, coupled with a strategy for subsequent robustness improvement such as using active surfaces to pull the object into a firm enveloping grasp.

C. Grasp Execution and Active Surfaces

For a proof of concept evaluation of the suggested methods, we used the platform in Fig. 1 to incrementally clear one of the cluttered scenes using the described grasping pipeline utilizing the set of planned pinch grasps \mathcal{G}^P . Initial grasps were performed by thresholding the current absorption of the gripper’s closing actuator. Currently, no current feedback is available for the belts on the fingers. Therefore, after an initial pinch grasp was acquired, the belt movements responsible for pulling the object into an enveloping grasp were triggered by an operator. In this fashion, the robot was able to retrieve all five objects needing six attempts, one object was dropped during the lift phase. An example of the robot using the ‘pull-in’ strategy is depicted in Fig. 9.

V. CONCLUSION

The main concept in this work is to simplify grasp planning for an underactuated grasping device and to entrust the specific kinematic/transmission structure of such a device with robust grasp execution. To this end, we adapt a well-established optimization based planning scheme [8] to the specifics of underactuated grippers. Furthermore, we investigate how active surfaces in form of belts on the fingers can aid in the grasp execution process, by manipulating an object in order to pull it into a robust enveloping grasp starting from an initial dexterous pinch grasp. We provide a numerical evaluation of the proposed grasp planner in a

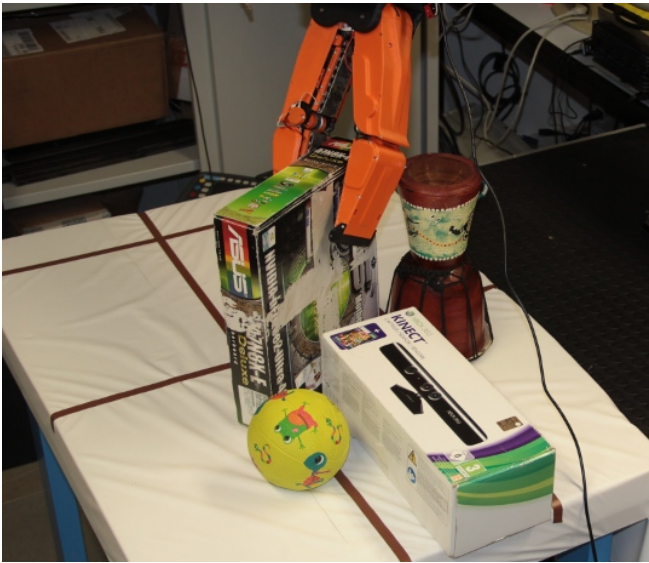


Fig. 9. Grasp execution with the ‘pull-in’ strategy: Shown is an example of the gripper using the active surfaces to retrieve an object via pulling it towards its palm.

real-world scenario and conduct proof-of-concept test runs on a robotic platform featuring an underactuated gripper.

Future work will be directed at an extensive experimental evaluation of the proposed grasp planning- and execution methodology. Therefore, the belt actuators will be augmented with current sensors to allow autonomous conduction of the pull in strategy by thresholding the corresponding motor currents. Furthermore, we will explore new possibilities of utilizing active surfaces in grasping such as using them to flip objects in order to ease grasp acquisition.

ACKNOWLEDGMENTS

This research has been partially supported by the ROBLOG project, funded by the European Community’s Seventh Framework Programme (FP7/2007-2013) under grant agreement ICT-270350.

REFERENCES

- [1] M. Grebenstein, A. Albu-Schaffer, T. Bahls, M. Chalon, O. Eiberger, W. Friedl, R. Gruber, S. Haddadin, U. Hagn, R. Haslinger, H. Hoppner, S. Jorg, M. Nickl, A. Nothhelfer, F. Petit, J. Reill, N. Seitz, T. Wimbock, S. Wolf, T. Wusthoff, and G. Hirzinger, “The DLR hand arm system,” in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2011, pp. 3175–3182.
- [2] A. M. Dollar and R. D. Howe, “The highly adaptive SDM hand: Design and performance evaluation.” *IJRR*, vol. 29, no. 5, pp. 585–597, 2010.
- [3] J. Dejeu, M. Bechelany, P. Rougeot, L. Philippe, and M. Gauthier, “Adhesion control for micro- and nanomanipulation,” *ACS Nano*, vol. 5, no. 6, pp. 4648–4657, 2011.
- [4] Fraunhofer-Institut fuer Materialfluss und Logistik IML. (2013) Roboter mit aufwaelzgreifer wird zum multitalent fuer die lagerlogistik. [Online]. Available: http://www.ipa.fraunhofer.de/fileadmin/www.ipa.fhg.de/Presse/Pressemitteilung/PR_Automatica.pdf
- [5] —. (2013) Traction gripper systeme - reibschluessiges greifen von stueckgut. [Online]. Available: <http://www.iml.fraunhofer.de/content/dam/iml/de/documents/OE%20140/Traction%20Gripper%20Systeme.pdf>

- [6] V. Tincani, M. Catalano, E. Farnioli, M. Garabini, G. Grioli, G. Fantoni, and A. Bicchi, “Velvet fingers: A dexterous gripper with active surfaces,” in *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2012, pp. 1257–1263.
- [7] V. Tincani, G. Grioli, M. G. Catalano, M. Garabini, S. Grechi, G. Fantoni, and A. Bicchi, “Implementation and control of the velvet fingers: a dexterous gripper with active surfaces,” in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2013, pp. 2744–2750.
- [8] M. T. Ciocarlie and P. K. Allen, “Hand posture subspaces for dexterous robotic grasping,” *IJRR*, vol. 28, no. 7, pp. 851–867, 2009.
- [9] C. Borst, M. Fischer, and G. Hirzinger, “Grasping the dice by dicing the grasp,” in *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2003, pp. 3692–3697.
- [10] —, “Grasp planning: how to choose a suitable task wrench space,” in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, vol. 1, 2004, pp. 319–325.
- [11] A. Bicchi and V. Kumar, “Robotic grasping and contact: a review,” in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, vol. 1, 2000, pp. 348–353.
- [12] S. Ekvall and D. Kragic, “Learning and evaluation of the approach vector for automatic grasp generation and planning,” in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2007, pp. 4715–4720.
- [13] A. Miller, S. Knoop, H. Christensen, and P. Allen, “Automatic grasp planning using shape primitives,” in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, vol. 2, 2003, pp. 1824–1829.
- [14] C. Goldfeder, P. Allen, C. Lackner, and R. Pelossof, “Grasp planning via decomposition trees,” in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2007, pp. 4679–4684.
- [15] K. Huebner and D. Kragic, “Selection of robot pre-grasps using box-based shape approximation,” in *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2008, pp. 1765–1770.
- [16] J. Bohg, A. Morales, T. Asfour, and D. Kragic, “Data-driven grasp synthesis—a survey,” *IEEE Transactions on Robotics*, to appear.
- [17] D. Berenson and S. Srinivasa, “Grasp synthesis in cluttered environments for dexterous hands,” in *Proc. of the IEEE/RAS Int. Conf. on Humanoid Robots*, 2008, pp. 189–196.
- [18] R. Diankov, “Automated construction of robotic manipulation programs,” Ph.D. dissertation, Carnegie Mellon University, Robotics Institute, 2010.
- [19] C. Ferrari and J. Canny, “Planning optimal grasps,” in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, vol. 3, 1992, pp. 2290–2295.
- [20] C. Goldfeder and K. Allen, “Data-driven grasping,” *Autonomous Robots*, vol. 31, no. 1, pp. 1–20, 2011.
- [21] D. Berenson, R. Diankov, K. Nishiwaki, S. Kagami, and J. Kuffner, “Grasp planning in complex scenes,” in *Proc. of the IEEE/RAS Int. Conf. on Humanoid Robots*, 2007, pp. 42–48.
- [22] M. R. Dogar and S. S. Srinivasa, “Push-grasping with dexterous hands: Mechanics and a method,” in *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2010, pp. 2123–2130.
- [23] A. Saxena, L. Wong, M. Quigley, and A. Y. Ng, “A vision-based system for grasping novel objects in cluttered environments,” in *Robotics Research*. Springer, 2011, pp. 337–348.
- [24] A. Miller and P. Allen, “Graspit! a versatile simulator for robotic grasping,” *IEEE Robotics Automation Magazine*, vol. 11, no. 4, pp. 110–122, 2004.
- [25] E. Rombokas, P. Brook, J. Smith, and Y. Matsuoka, “Biologically inspired grasp planning using only orthogonal approach angles,” in *Proc. of the IEEE/RAS-EMBS Int. Conf. on Biomedical Robotics and Biomechanics*, 2012, pp. 1656–1661.
- [26] R.-G. Mihalyi, K. Pathak, N. Vaskevicius, and A. Birk, “Uncertainty Estimation of AR-Marker Poses for Graph-SLAM Optimization in 3D Object Model Generation with RGBD Data,” in *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2013, pp. 1807–1813.
- [27] D. Canelhas, T. Stoyanov, and A. Lilienthal, “SDF tracker: A parallel algorithm for on-line pose estimation and scene reconstruction from depth images,” in *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2013, pp. 3671–3676.
- [28] N. Vaskevicius, K. Pathak, A. Ichim, and A. Birk, “The jacobs robotics approach to object recognition and localization in the context of the icra’11 solutions in perception challenge,” in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2012, pp. 3475–3481.
- [29] I. A. Sucas and S. Chitta. (2013) “Moveit!”. [Online]. Available: <http://moveit.ros.org/>