

# Planning Grasp Strategies That Exploit Environmental Constraints

Clemens Eppner

Oliver Brock

**Abstract**—There is strong evidence that robustness in human and robotic grasping can be achieved through the deliberate exploitation of contact with the environment. In contrast to this, traditional grasp planners generally disregard the opportunity to interact with the environment during grasping. In this paper, we propose a novel view of grasp planning that centers on the exploitation of environmental contact. In this view, grasps are sequences of constraint exploitations, i.e. consecutive motions constrained by features in the environment, ending in a grasp. To be able to generate such grasp plans, it becomes necessary to consider planning, perception, and control as tightly integrated components. As a result, each of these components can be simplified while still yielding reliable grasping performance. We propose a first implementation of a grasp planner based on this view and demonstrate in real-world experiments the robustness and versatility of the resulting grasp plans.

## I. INTRODUCTION

Many recent advances in robotic grasping and manipulation can be explained by a simple insight: contact with the environment can improve performance! For example, under-actuated, soft hands greatly benefit from the interactions that occur naturally between hand, object, and environment [1]–[3]. Furthermore, the robustness of grasping can be increased through the use of contact with support surfaces [4], [5]. And the dexterity of simple, rigid hands is increased drastically through deliberate contact with the environment [6]. In addition, human graspers routinely and amply employ environmental contact, especially in difficult grasping problems [4]. Human studies also provide impressive anecdotal evidence, such as this annotated video<sup>1</sup>, of a human cook cutting potatoes. Given this broad support for the importance of contact in grasping and manipulation, it is surprising that recent grasp and manipulation planners generally regard the environment as an obstacle, rather than as an opportunity.

We introduce a grasp planner that generates robust grasping strategies based on the exploitation of contact constraints available in the environment (see Fig. 1). We define an *environmental constraint* (EC) as a feature of the environment that enables replacing aspects of control and/or perception with interaction between hand and environment. To plan the exploitation of EC, we must eliminate the existing separation between perception, planning, and control. Instead, we tightly integrate perception and action by realizing each to satisfy the others’ requirements and to account for its limitations.

Both authors are with the Robotics and Biology Laboratory, Technische Universität Berlin, Germany.

We gratefully acknowledge financial support by the Alexander von Humboldt foundation through an Alexander von Humboldt professorship (funded by the German Federal Ministry of Education and Research).

<sup>1</sup><http://youtu.be/Vjq5P24AkW>

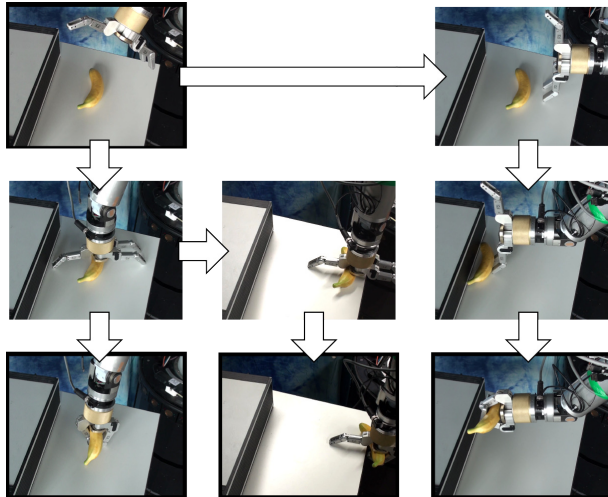


Fig. 1. Our planner generates grasping strategies that make extensive use of contact with the environment. Thus, control and perception can be simplified. In this example, three different grasping strategies for a banana are shown. They exploit the table surface, edge, and the side panel of an object sitting on top of the table.

The main contribution of this paper is this novel (albeit heavily informed by previous contributions) perspective on the grasping problem. We present a formulation of the grasping problem and transfer it into a specific planning algorithm. This algorithm is capable of generating grasp plans that achieve robustness through the exploitation of environmental constraints. This, at least in principle, is already possible with traditional planning methods. However, these depend on exact geometric models of the world. In contrast, we demonstrate grasp planning in real-world experiments based on a single depth image of a scene. From this single image, the planner generates robust, environmentally-constrained grasps. We believe that grasp and manipulation planning in this novel view will lead to increased robustness and increased capabilities for robotic manipulation systems.

### Planning Concept

As the proposed planner shares characteristics with several different branches of work in the robotics literature, it is helpful to provide a brief overview of the basic idea, before discussing related work. The key challenge is that exact planning in the combined state space of hand, object, and environment is too difficult. To follow the rest of this section, please refer to Fig. 2.

Environmental constraints implicitly divide the state space into separate regions, i.e. regions that correspond to one particular type of EC exploitation. Samples from these re-

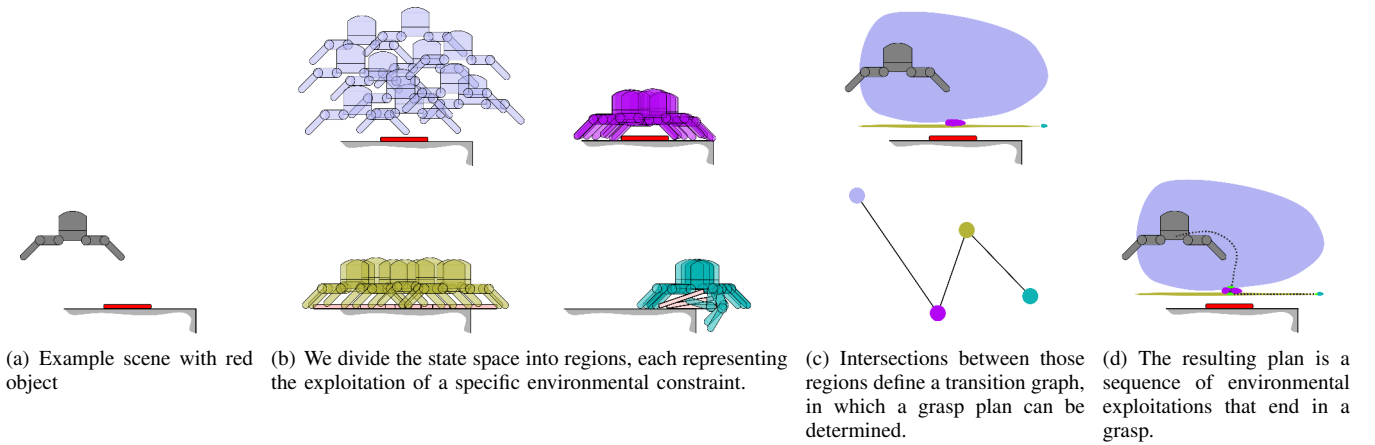


Fig. 2. Illustration of planning environmental constraint exploitations for grasping.

gions projected into the workspace are shown in Fig. 2(b). As we will see later, rather than computing these regions explicitly, which would be computationally challenging, we determine an approximate representation of these regions directly from sensor data. By determining intersections of these approximated state space regions in a graph (Fig. 2(c)), we obtain a representation of possible transitions between different EC. Given this information, we can search for a feasible sequence of EC exploitations that leads from the current state to a successful grasp (see Fig. 2(d)).

By approximating the EC-relevant state space regions from sensor data, we have circumvented complex, high-dimensional c-space planning problems. As we will see, these approximations still permit effective EC planning.

## II. RELATED WORK

The idea of exploiting the environment and using contact via compliance can at least be traced back to the influential approach by Lozano-Pérez et al. from 1984 [7]. This work introduced the central concept of the *pre-image*—a description of all positions that reach the goal given the same action—to enable the generation of fine-motion plans. Later, this idea was extended to the concept of *backprojections*, representing a weaker form of a pre-image, by separating goal reachability from goal recognizability [8]. By restricting plans to temporal termination predicates of motion primitives, manipulation tasks can even be solved in an open-loop, sensorless fashion [9]. Subsequently, the binary representation of a backprojection was given a probabilistic upgrade [10]. These approaches depend on exact geometric models of the environment.

The geometric reasoning we apply is similar to classical work in assembly planning [11]. The way we define pose constraints is similar to the task space regions [12] used for motion planning. And the concept of contact-state graphs [13] is reflected in our representation of environmental constraint transitions. However, in addition to contact states, our algorithm requires spatial information derived from sensor data. And again, in contrast to the mentioned approaches, we do so entirely based on sensor information.

Closely related to environmental constraint exploitations are *guarded moves*. A guarded move is “a move until some expected sensory event occurs” [14]. Plans of guarded moves can include branches based on sensor events to compensate for uncertainty in world modeling [15]. Our work includes this concept but in addition is concerned with the automated generation of such plans from sensor data.

Pre-grasp manipulation refers to the contact-drive modification of the environment to facilitate a subsequent manipulation action. These actions involve, for example, rotations due to payload limits [16], or sliding flat objects on table surfaces [17], [18]. All these works realized specific, pre-programmed actions. In addition, pushing or sweeping can be considered as pre-grasp action [19]. There, the environment is designed to present challenges to the planner, rather than opportunities.

Toussaint et al. provide an interesting approach to formalize the advantages of contact exploitation [20]. They optimize plans so that they minimize uncertainty by contact with the environment. In our approach, the assumption that contact during manipulation is beneficial is directly encoded by focusing on contact-based actions.

## III. SEQUENCING ENVIRONMENTAL CONSTRAINT EXPLOITATIONS

EC exploitations represent structured parts of the state space that can be easily recognized and acted on. We define a single EC exploitation as a contiguous subset of all possible hand and object poses and the exerted forces onto the hand:

$$X_{ECE} \subset \mathcal{C}^{\text{hand}} \times \mathcal{C}^{\text{object}} \times \mathcal{W}^{\text{hand}},$$

where  $\mathcal{C}^{\text{hand}} = \mathcal{C}^{\text{object}} = SE(3)$  and  $\mathcal{W}^{\text{hand}}$  is the 6D wrench space. To plan among EC exploitations we need to look at their connectivity. This is defined by their intersections, i.e. we can transit between two arbitrary EC exploitations  $X_{ECE_i}$  and  $X_{ECE_j}$  if  $X_{ECE_i} \cap X_{ECE_j} \neq \emptyset$ .

Due to its high complexity, we need to approximate  $X_{ECE}$ . The corresponding hand poses are given as an oriented bounding box and orientations are represented by discretizing

all possible rotations. We use a single 6D vector to represent the contact wrench exerted onto the hand, since this contact property is constant within a single EC exploitation. To describe the relationship between hand and object, we use the predicates {away, caged, grasped}. The resulting parametrization used by the planner is:

$$\tilde{X}_{ECE_i}(obb^{hand}, R^{hand}, C^{obj}, w^{hand}) = \left\{ \left( \begin{array}{c} c^{hand} \\ c^{obj} \\ w^{hand} \end{array} \right) \left| \begin{array}{l} c_{position}^{hand} \in obb^{hand} \\ c_{rotation}^{hand} \in R^{hand} \\ c^{obj} \in C^{obj} \\ w^{hand} \in \mathbb{R}^6 \end{array} \right. \right\}, \quad (1)$$

where  $obb^{hand}$  is the 3D oriented bounding box,  $R^{hand}$  is the set of orientations, and  $C^{obj} \in \mathcal{P}(\{\text{away, caged, grasped}\})$ . The wrench  $w^{hand}$  describes the forces and torques acting onto the hand within that particular EC exploitation.

To find sequential exploitations of environmental constraints we need to define the aforementioned connectivity check between two arbitrary EC exploitations. We use their spatial and contact properties to decide this:

$$\begin{aligned} \tilde{X}_{ECE_i} \mapsto \tilde{X}_{ECE_j} &\iff obb_i \cap obb_j \neq \emptyset \wedge R_i \cap R_j \neq \emptyset \\ &\wedge w_j^{hand} - w_i^{hand} \neq 0 \\ &\wedge C_i^{obj} \cap C_j^{obj} \neq \emptyset \\ &\wedge C_j^{obj} \setminus C_i^{obj} \neq \{\text{away}\} \end{aligned}$$

This means that the hand poses described by both EC exploitation need to overlap, there needs to be a change in the force exerted by the environment onto the hand, and they need to share at least one object-hand mode. Note, that the sequence operator is not symmetric (that's why we use ' $\mapsto$ ' instead of ' $\cap$ '), because of the last term which excludes the retraction of the hand. Thus, all sequences undergo the same object-hand mode changes: away  $\mapsto$  caged  $\mapsto$  grasped.

Given the above parametrization we propose the following hierarchical planning scheme to find sequences of EC exploitations:

- 1) EC exploitations are extracted from a single depth image of the scene according to the definitions given above. Each one is inserted as a vertex into a graph.
- 2) All nodes  $\tilde{X}_{ECE_i}$  for which (grasped)  $\in C_i^{obj}$  are marked as goal nodes. The node that contains the current robot/object state is the starting node.
- 3) The  $\mapsto$ -operator is used to calculate transitions between all pairs of vertices. Each transition is added as a directed edge between the corresponding vertices.
- 4) Search the graph for all paths that connect the initial and goal nodes. All paths that lead from start to goal are sequences of EC exploitations.

This is very similar to the the concept of manipulation graphs [21].

We are finally looking for controllers to execute the sequence on a real robotic platform. Each EC exploitation can be seen as a controller with desired spatial and contact profiles and a termination predicate defining the switching condition. Given a planned sequence of EC exploitations

( $\tilde{X}_{ECE_1} \mapsto \tilde{X}_{ECE_2} \mapsto \dots \mapsto \tilde{X}_{ECE_n}$ ), we construct a hybrid automaton by using hybrid position/force controllers. Their termination predicates are defined as the poses given by  $(obb_i^{hand}, R_i^{hand}) \cap (obb_j^{hand}, R_j^{hand})$  and the wrenches given by  $w_j^{hand} - w_i^{hand}$ . Note, that here we use simple linear interpolation to generate trajectories but more advanced motion planning algorithms could be applied.

#### IV. INTEGRATING PERCEPTION AND ACTION

In the following, we will explain three grasping strategies and three non-prehensile manipulations that make explicit use of ECs. For each of the six actions, we will devise a sensor model that is used to visually recognize the EC exploitation according to the parametrization given previously 1:  $\tilde{X}_{ECE}(obb^{hand}, R^{hand}, C^{obj}, w^{hand})$ . Sensory input is assumed to be in the form of a depth image of the scene. The object to be grasped is represented as a bounding box whose parameters are assumed to be known.

*Surface-Constrained Grasp:* This grasping strategy can be applied whenever the object is placed on a flat support surface and partly caged by the hand. During finger closing, a compliant wrist position along the support surface normal guarantees slip along the environmental surface while contact locations between object and fingers remain stable. The region  $obb^{hand}$  is given by the known object bounding box. The orientations  $R^{hand}$  are normal to the objects surrounding surface [22]. The wrench  $w^{hand}$  consists of a force component directed towards the palm of the hand. The surface-constrained grasp assumes the object to be already caged and closing the fingers will fixate the object inside the hand, i.e.  $C^{obj} = \{\text{caged, grasped}\}$ .

*Wall-Constrained Grasp:* This grasping strategy exploits two surfaces that form a concavity. The open hand pushes the object along its support surface towards the second surface – the *wall*. While the object is caged between wall and palm, the fingers can slip underneath the object (see Fig. 3). We detect wall-constrained grasps by finding concave edges in the environment. We have tested this strategy under varying inclination angles between the two surfaces. These tests revealed successful grasps for angles between  $\sim 90^\circ$  and  $\sim 140^\circ$ . The poses in which the hand can exploit the surface during grasping is defined by two intersecting planar segments. Accordingly,  $obb^{hand}$  is set along the edge and orientations  $R^{hand}$  are normal to the wall surface. Again, the wrench  $w^{hand}$  consists only of a force component directed towards the palm of the hand and  $C^{obj} = \{\text{caged, grasped}\}$ .

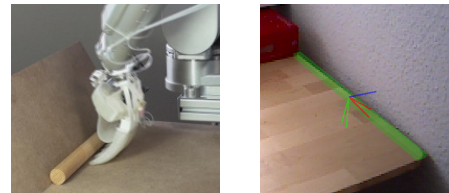


Fig. 3. *Left:* The wall-constrained grasp. *Right:* Visualization of its spatial properties  $obb^{hand}$  (green) and  $R^{hand}$  (RGB axes) extracted from depth measurements.



*Edge-Constrained Grasp:* Here, we assume that the object is close to a convex edge, partially protruding it. The open hand wraps its fingers around the exposed object surface to grasp it. In contrast to the two strategies explained above, the environment is not used to reduce the object’s DOF during grasping. Instead the edge is a particular part of space that allows the hand to easily take over the DOF-constraining function of the environment. The spatial parameters  $obb^{hand}$  and  $R^{hand}$  are computed based on the presence of edges in the scene. We extract edges by searching along the boundaries of planar segments in the depth image, see Fig. 4. Convexity is determined by the local curvature along an edge. The wrench parameter  $w^{hand}$  represents the torque components that is exerted onto the hand due to the fact that some fingers still touch the surface while half of the hand already passed the edge. The relationship between hand and object is described by  $C^{obj} = \{\text{caged, grasped}\}$ .

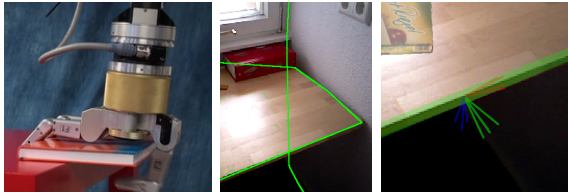


Fig. 4. *Left:* The edge-constrained grasp. *Right:* Visualization of candidate edges from planar segmentation and the resulting spatial properties  $obb^{hand}$  (green) and  $R^{hand}$  (RGB axes).

*Visually-Constrained Positioning:* Inside the visible workspace, the hand can be constraint visually, i.e. by visual servoing. This strategy is helpful whenever positioning the hand in free space is hard due to missing external calibrations or poor sensor models (e.g. encoders that ignore cable stretch). Although we constrain the hand by model-based tracking in 3D using a depth sensor, more complex schemes are possible that use lower-level features like edges and RGB-cameras [23]. To compute the spatial extent  $obb^{hand}$ , we expand a box in visible free space starting from the current hand pose until collision with a depth measurement. The set  $R^{hand}$  includes all possible orientations. The wrench parameter  $w^{hand} = 0$  and the object is outside the hand, i.e.  $C^{obj} = \{\text{away}\}$ .

*Object-Constrained Caging:* This action simply gets the hand/palm close to the object. It can be applied whenever parts of the object are freely accessible. The pre-grasp configurations based on fitting geometric primitives to the object [22] could be used. Since the bounding box of the object is given, we use only the box pre-grasp to determine  $obb^{hand}$  and  $R^{hand}$ . The wrench  $w^{hand}$  exerted onto the hand is zero since object and hand are not directly in contact. Still,  $C^{obj} = \{\text{away, caged}\}$  holds since this EC exploitation is exactly at the boundary of caging the object.

*Surface-Constrained Sliding:* This strategy assumes that the hand is caging the object or that the hand is isolated from the object. During sliding the hand and object’s motion are restricted by a support surface exposing only three DOFs. We extract sliding constraints by segmenting the depth image

with a flood-fill algorithm. It clusters regions with low curvature and small changes of surface normals (see Fig. 5). To generate the corresponding region  $obb^{hand}$  each planar segment is turned into the maximum inscribing rectangle and off-setted by object size. The orientations  $R^{hand}$  includes all rotations around the surface normal of the segment. The wrench  $w^{hand}$  is set with a force that points inside the hand’s palm;  $C^{obj} = \{\text{caged}\}$  and in the case of hand/palm interaction  $C^{obj} = \{\text{away}\}$ .



Fig. 5. *Left:* The surface-constrained slide. *Right:* Visualization of the planar segmentation from depth data and the resulting spatial properties  $obb^{hand}$  (green) and  $R^{hand}$  (RGB axes).

## V. EXPERIMENTS

Our experiments need to show that the grasp strategies produced by the planner can be executed reliably on a physical platform. Additionally, we need to show that the proposed environmental constraints can be found and exploited in a significant amount of everyday scenarios where a robot will encounter grasping tasks.

### A. Evaluation of various real-world scenes

To evaluate the applicability of our planner, we used 30 indoor scenes from a clutter dataset [24]. They depict office desks, book shelves, and kitchen environments and contain a significant amount of clutter. The scenes are encoded as polygonal meshes but we feed our algorithm with a ray-traced depth image of a single viewpoint in which most of the mesh is visible. For each scene, we position a box-like object at a random location that we assume to be statically stable. In total, our planner generated 218 grasping sequences, with at least one sequence per scene (average of  $\sim 3.6$  per scene). Among the planned strategies most ended in an edge-constrained grasp (64%), followed by surface-constrained (27%), and wall-constrained (9%) ones. The most prominent problems the planner encountered were wrongly recognized edge-constraint grasps. To assess the quality of the generated plans, they were visually inspected and categorized according to their feasibility. In total, 62% of the plans were deemed feasible. The majority among the infeasible plans ended in edge-constrained grasps (92%). This had two reasons: Because of the nature of the dataset, point clouds were often incomplete, increasing the occurrence of shadow edges (see Fig. 6). The dataset also contains a significant amount of clutter making support surfaces unnavigable. Example plans showing the different failure cases and successes can be seen in Fig. 6. The results indicate that the environmental constraints are general features that can be exploited for grasping in a wide variety of human environments.

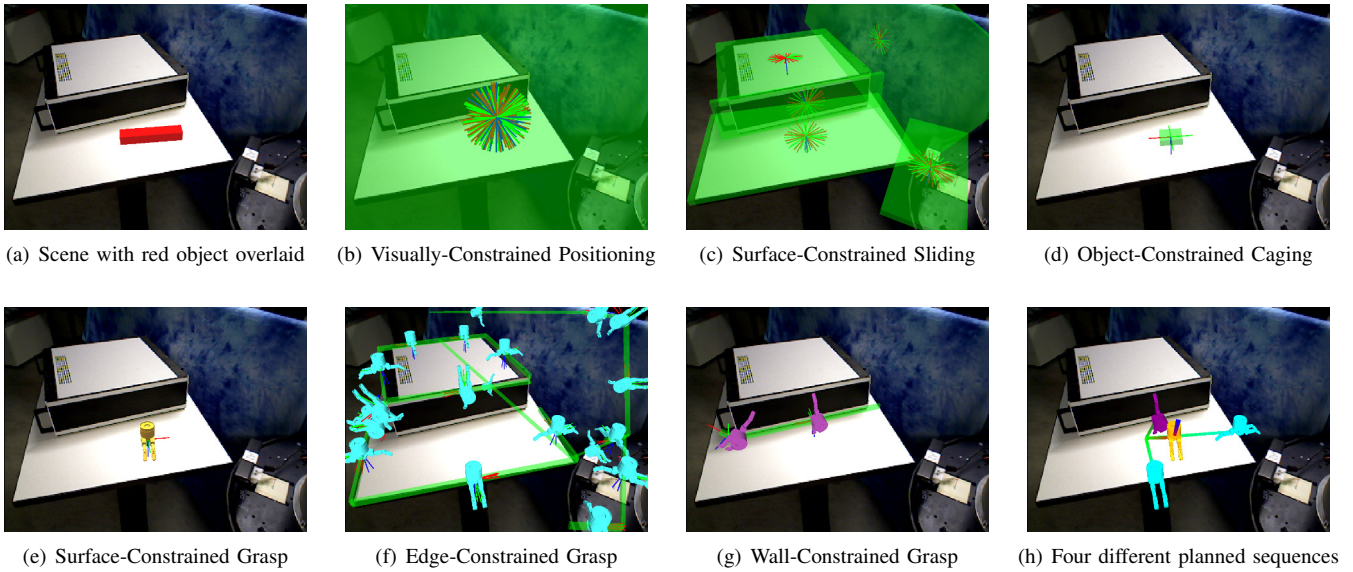


Fig. 7. Experimental banana scene: For each of the six EC exploitations, the extracted spatial properties  $obb^{hand}$  and  $R^{hand}$  are shown. The final planned grasping sequences are shown in Fig. 7(h).

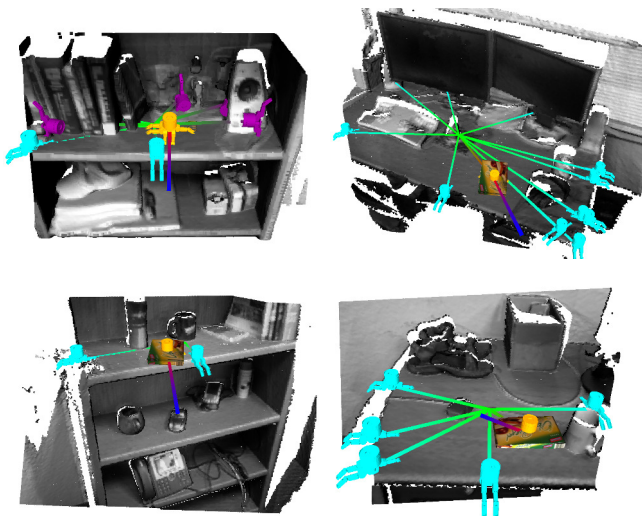


Fig. 6. Evaluated environments from a clutter dataset [24]. The resulting grasp plans are overlaid in color. Lines connect consecutive exploitations. The final prehensile exploitation for each sequence is plotted with a hand model (*orange*: surface-constrained, *cyan*: edge-constrained, *purple*: wall-constrained grasp). The dataset revealed a lot of shadow edges which produced false positives among the edge-constrained grasps.

### B. Execution of Plans on a Real Robot

To further evaluate the feasibility of the plans, we executed some of them on a real robotic platform. We used a Barrett WAM with seven DOFs, an ATI Industrial Automation multi-axis force-torque sensor, and a Barrett hand BH-262 with four DOFs. We chose a natural scenario in which multiple grasping strategies would be possible. It contained a banana placed onto a table surface with a big electronic amplifier next to it (Fig. 7(a)). The scene was measured with an Asus Xtion Live depth sensor at QVGA resolution from a single static viewpoint. Color information is not used at any stage

of the algorithm. Location and dimensions of a bounding box describing the banana were given to the algorithm.

Fig. 7(b)-7(d) show the extracted EC exploitations for the non-prehensile actions. Slidable surfaces were found on the table, at the side panels of the electronic amplifier, at the curtain and on top of the robot base. Fig. 7(e)-7(g) display the detected EC exploitations that refer to grasping actions. Two possible wall-constrained grasps were found between the table and the electronic amplifier 7(g). Much more false positives were among the recognized edge-constrained grasps 7(f): E.g. the lower part of the curtain was shadowed by the table and not a real edge due to depth discontinuities. In total, 28 EC exploitations were found. Their connectivity is depicted in the graph in Fig. 8. The graph also shows the four paths the algorithm finally found from the current unconstrained hand pose to one of the three types of prehensile EC exploitations. Fig. 7(h) visualizes the four sequences in the scene. For execution, the sequences are converted to multiple hybrid position/force controllers. Switching between them is governed by contact with a surface. Instances of the executed sequences are shown in Fig. 9.

### C. Limitations

Though our method proves to be a powerful way of generating robust grasping behavior, there are limitations that require future work. As mentioned earlier, a planned sequence does never contain multiple contact-making/breaking events between hand and object contact establishing phases. For most grasping strategies this is a reasonable assumption. Additionally, the shape of the object is only represented by a bounding box, more complex kinematic relations between object and environment such as rolling contacts are missing. Presently, the algorithm does not use any intrinsic object properties during planning (e.g. friction, mass). The linear trajectories within one environmental constraint exploitation



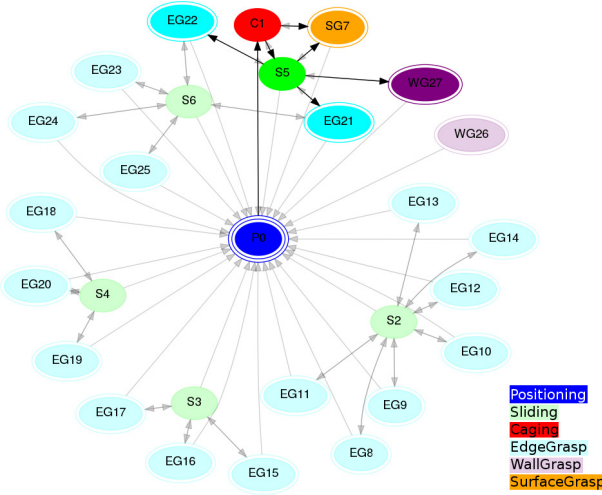


Fig. 8. Graph showing the extracted EC exploitations and connectivity in the example scene 7. Nodes are color-coded according to EC exploitation (see Fig. 7). Framed nodes are goal nodes, start node is double-framed.

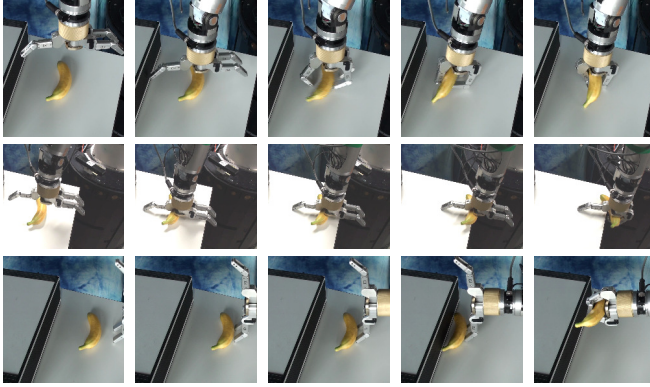


Fig. 9. Sequences of executing three plans found in the banana scene 7. (First to last row: surface-constrained, edge-constrained, and wall-constrained grasping strategies)

makes the approach prone to obstacle collisions. Here, more sophisticated schemes such as trajectory optimization could be applied.

## VI. CONCLUSION

We presented a grasp planning algorithm to synthesize and execute grasping strategies that exploit environmental constraints. Recent results from the grasping literature lead to the conclusion that such exploitation plays an important role in achieving robust grasping performance. Our grasp planner leverages this insight and sequences constraint exploitations into versatile and robust grasp plans. The algorithm tightly couples planning, perception, and control, thereby enabling grasp planning from real-world sensor data in the absence of prior information about the world. We demonstrated the effectiveness of the planner in experiments on a real robot platform and illustrated the generality of the planner by generating grasp plans in a great variety of environments. We believe that the exploitation of environmental constraints is a promising route leading towards robust grasping and

manipulation with weak a priori object and world models.

## REFERENCES

- [1] A. M. Dollar and R. D. Howe, "The highly adaptive sdm hand: Design and performance evaluation," *Int. J. Robotics Research*, vol. 29, no. 5, pp. 585–597, 2010.
- [2] R. Deimel and O. Brock, "A novel type of compliant, underactuated robotic hand for dexterous grasping," in *Proc. of Robotics: Science and Systems*, Berkeley, USA, July 2014.
- [3] M. G. Catalano, G. Grioli, E. Farnioli, A. Serio, c. Piazza, and A. Bicchi, "Adaptive synergies for the design and control of the Pisa/IIT SoftHand," *Int. J. Robotics Research*, vol. 33, no. 5, pp. 768–782, 2014.
- [4] R. Deimel, C. Eppner, J. Álvarez-Ruiz, M. Maertens, and O. Brock, "Exploitation of environmental constraints in human and robotic grasping," in *Int. Symposium on Robotics Research (ISRR)*, 2013.
- [5] M. Kazemi, J. Valois, J. A. Bagnell, and N. Pollard, "Human-inspired force compliant grasping primitives," *Autonomous Robots*, vol. 37, no. 2, pp. 209–225, 2014.
- [6] N. Chavan-Dafle, A. Rodriguez, R. Paolini, B. Tang, S. Srinivasa, M. Erdmann, M. T. Mason, I. Lundberg, H. Staab, and T. Fuhlbrigge, "Extrinsic dexterity: In-hand manipulation with external forces," in *Int. Conf. on Robotics and Automation (ICRA)*, 2014.
- [7] T. Lozano-Pérez, M. T. Mason, and R. H. Taylor, "Automatic synthesis of fine-motion strategies for robots," *Int. J. Robotics Research*, vol. 3, no. 1, pp. 3–24, 1984.
- [8] M. Erdmann, "Using backprojections for fine motion planning with uncertainty," *Int. J. Robotics Research*, vol. 5, no. 1, pp. 19–45, 1986.
- [9] M. A. Erdmann and M. T. Mason, "An exploration of sensorless manipulation," *IEEE Journal of Robotics and Automation*, vol. 4, no. 4, pp. 369–379, 1988.
- [10] R. C. Brost and A. D. Christiansen, "Probabilistic analysis of manipulation tasks: A conceptual framework," *Int. J. Robotics Research*, vol. 15, no. 1, pp. 1–23, 1996.
- [11] R. H. Wilson and J.-c. Latombe, "Geometric reasoning about mechanical assembly," *Artificial Intelligence*, vol. 71, pp. 371–396, 1994.
- [12] D. Berenson, S. S. Srinivasa, and J. Kuffner, "Task space regions: A framework for pose-constrained manipulation planning," *Int. J. Robotics Research*, 2011.
- [13] X. Ji and J. Xiao, "Automatic generation of high-level contact state space," *Int. J. Robotics Research*, vol. 20, pp. 238–244, 1999.
- [14] P. Will and D. Grossman, "An experimental system for computer controlled mechanical assembly," *IEEE Transactions on Computers*, vol. C-24, no. 9, pp. 879–888, 1975.
- [15] B. Finkemeyer, T. Kröger, and F. M. Wahl, "Executing assembly tasks specified by manipulation primitive nets," *Advanced Robotics*, vol. 19, no. 5, pp. 591–611, 2005.
- [16] L. Chang, S. Srinivasa, and N. Pollard, "Planning pre-grasp manipulation for transport tasks," in *Int. Conf. on Robotics and Automation (ICRA)*, 2010, pp. 2697–2704.
- [17] J. King, M. Klingensmith, C. Dellin, M. Dogar, P. Velagapudi, N. Pollard, and S. Srinivasa, "Pregrasp manipulation as trajectory optimization," in *Proc. of Robotics Science and Systems (RSS)*, 2013.
- [18] D. Kappler, L. Chang, M. Przybylski, N. Pollard, T. Asfour, and R. Dillmann, "Representation of pre-grasp strategies for object manipulation," in *Int. Conf. on Humanoid Robots*, 2010.
- [19] M. R. Dogar and S. S. Srinivasa, "A planning framework for non-prehensile manipulation under clutter and uncertainty," *Autonomous Robots*, vol. 33, no. 3, pp. 217–236, 2012.
- [20] M. Toussaint, N. Ratliff, J. Bohg, L. Righetti, P. Englert, and S. Schaal, "Dual execution of optimized contact interaction trajectories," in *Int. Conf. on Intelligent Robots and Systems (IROS)*, 2014.
- [21] T. Siméon, J. Cortés, A. Sahbani, and J.-P. Laumond, "A general manipulation task planner," in *Algorithmic Foundations of Robotics V*. Springer, 2004, pp. 311–328.
- [22] C. Eppner and O. Brock, "Grasping unknown objects by exploiting shape adaptability and environmental constraints," in *Int. Conf. on Intelligent Robots and Systems (IROS)*, 2013, pp. 4000–4006.
- [23] A. Fox and S. Hutchinson, "Exploiting visual constraints in the synthesis of uncertainty-tolerant motion plans," *IEEE T. Robotics and Automation*, vol. 11, no. 1, pp. 56–71, 1995.
- [24] A. Karpathy, S. Miller, and L. Fei-Fei, "Object discovery in 3d scenes via shape analysis," in *Int. Conf. on Robotics and Automation (ICRA)*, 2013.