2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 23-27, 2022, Kyoto, Japan

2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) | 978-1-6654-7927-1/22/$31.00 ©2022 IEEE | DOI: 10.1109/IROS47612.2022.9981569

Extrinsic Dexterous Manipulation with a Direct-drive Hand: A Case Study

Arnav Gupta[1*], Yuemin Mao[1*], Ankit Bhatia[1*],
Xianyi Cheng[1], Jonathan King[1], Yifan Hou, and Matthew T. Mason[12]

Abstract— **This paper explores a novel approach to dexterous manipulation, aimed at levels of speed, precision, robustness, and simplicity suitable for practical deployment. The enabling technology is a Direct-drive Hand (DDHand) comprising two fingers, two DOFs each, that exhibit high speed and a light touch. The test application is the dexterous manipulation of three small and irregular parts, moving them to a grasp suitable for a subsequent assembly operation, regardless of initial presentation. We employed four primitive behaviors that use ground contact as a "third finger", prior to or during the grasp process: pushing, pivoting, toppling, and squeeze-grasping. In our experiments, each part was presented from 30 to 90 times randomly positioned in each stable pose. Success rates varied from 83% to 100%. The time to manipulate and grasp was 6.32 seconds on average, varying from 2.07 to 16 seconds. In some cases, performance was robust, precise, and fast enough for practical applications, but in other cases, pose uncertainty required time-consuming vision and arm motions. The paper concludes with a discussion of further improvements required to make the primitives robust, eliminate uncertainty, and reduce this dependence on vision and arm motion.**

I. Introduction

The *dexterous manipulation* problem traditionally is to shift a grasped object from one pose to another, perhaps to a pose more suitable for downstream processes. For example, it is impossible to wrap your fingers around a pencil lying flat on a table, so you grasp the pencil first, and then you use dexterous finger motions to shift the grasped pencil to a more suitable pose for writing, and you might use dexterous manipulation again to reverse the pencil and use the eraser. Over the years the definition has been broadened to include dexterity prior to grasping, during the grasp, or during placement or other operations.

Dexterous manipulation research has been advancing for forty years, yet still seems far from practical use, lacking in some key areas:

- *Breadth of scope.* It must be suitable to a broad range of items.
- *Precision.* The item must be moved to the desired pose in the hand, precisely enough for downstream processes such as product assembly.
- *Robustness.* The processes must be robust to variations in initial pose, object shape, deformation, materials, ambient lighting, and other variables.

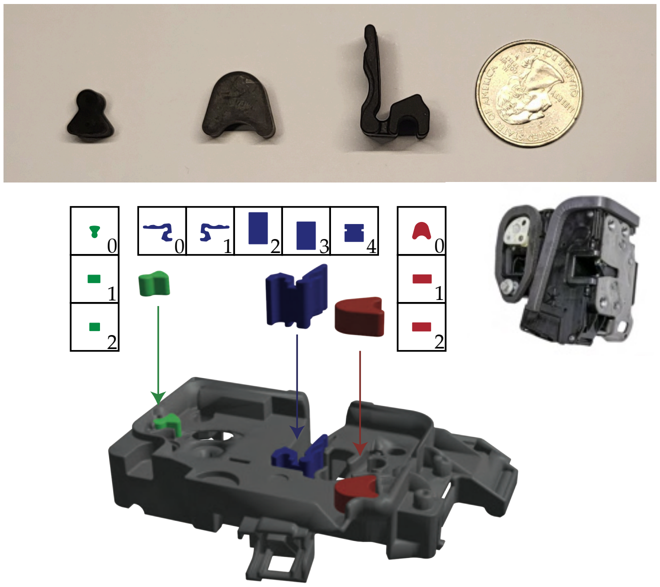


Fig. 1: Top: (left to right) Small Bumper, Medium Bumper, Long Bumper, and a US quarter for scale. Bottom: The assembly task discussed in this work. Three rubber parts need to be inserted into the housing (black): small bumper (green), medium bumper (red) and long bumper (blue.) The top view of the stable poses is shown in the figure. Pose 0 for all objects is considered to be the assemblable pose.

- *Speed.* Whether in industrial applications or domestic service, speed is essential. Human dexterous motions may take place in under a second. Industry typically demands cycle times of just a few seconds, leaving very little time for dexterous motions.

This paper presents a novel approach to dexterous manipulation which closes the gap in all four areas. The key elements of the approach are:

1) *Direct-drive hand.* Direct-drive is well-explored in manipulator design, but has only recently become practical for fingers.
2) *4DOF hand with "intermittently planar" kinematics.* Considerably simpler than most previous dexterous systems, the hand kinematics comprises two 2DOF fingers in a co-planar arrangement. We hypothesize that almost every required 3D manipulation can be mapped to a planar manipulation by placing the operating plane correctly.

*The first three authors contributed equally to this work.

[1] These authors are with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA

[2] This author is also with Berkshire Grey.

978-1-6654-7927-1/22/$31.00 ©2022 IEEE

3) *Pre-grasp whole-world manipulation.* Putting the dexterity before the grasp, or as part of the grasp, means the support surface is available as a third finger. The drawback is the additional time required. We hypothesize that the system will be fast enough to perform the manipulation and grasp in just a few seconds, fast enough for industrial overall cycle times as short as six seconds.
4) *Non-prehensile primitives*—pushing, squeezing, toppling, and pivoting—combined when possible with positive location, designed for open-loop precision and robustness.

To test the approach we focus on the dexterous manipulation of three small parts of an automobile door-latch subassembly (Fig. 1). The parts must be grasped in a pose suitable for an assembly operation, which usually requires dexterous manipulation. The parts are presented singulated on a light table, in any stable orientation. The presenting pose is reported by a vision system, our system sequences motion primitives to grasp the part in the assemblable pose, and the resulting plan is executed by a dexterous direct drive hand.

Note that this work doesn't explore autonomous generation or sequencing of the motion plans. Instead, we show a method well within the reach of industrial deployment needing minimal human design, and amenable to future work on automatic planning.

We report the results of over 400 trials– at least 30 trials for each presenting pose of each part. The success rate varies from 83% for some part poses to 100% for others. We show that speed is one advantage of this system — the required execution time was four seconds or less for two of the parts, but ran up to sixteen seconds for the third part. However, these times do not include the time for vision due to variations in some primitives' outcomes. To further speed up the process, we suggest using additional robustifying primitives to reduce object pose uncertainty, as presented in the discussion section.

A. Contributions

The main contributions of this work are:

- Novel hand kinematics employing two 2DOF co-planar fingers.
- Design and fabrication of a new hand design, the Dexterous Direct-drive Hand.
- Novel approach to 3D manipulation via intermittently planar hand motions, interleaved with 3D arm motion.
- Novel combination of pre-grasp and in-grasp primitives using whole-world manipulation.
- Experimental testing of the system.

B. Limitations

We test the approach on three parts of similar scale. It will take a considerable amount of further work to characterize the scope of the approach. In particular the scope of the specific primitives is unclear. Future work might explore the scope of the primitives and develop additional primitives.

This paper is entirely experimental, although some quasistatic analysis of the primitives might be straightforward, and might address the scope of the primitives, guide work on automatic planning, and guide refinement of the primitives or development of additional primitives.

The high-level programming was manual. While this isn't unusual in current commercial practice, it is unclear whether the programming will be straightforward enough for commercial deployment, or whether additional off-line programming tools and automatic planning are needed.

There isn't an experimental comparison with competing approaches. Implementation of a competing approach would be challenging or impossible, which is partly what motivated this work. The most likely competing approach would be to use all place-and-pick regrasps, which would require slow arm motions for every action, and could not possibly meet typical time constraints. (Compare Figures 2 and 3).

II. Related Work

The simplest form of manipulation is place-and-pick, where the hand is only used for gripping, and the object moves only when rigidly gripped by the hand. When a required grasp pose is inaccessible as presented, a *regrasp* aka *place-and-pick regrasp* might work. The object might be picked, then placed in an intermediate pose, then picked again in the required pose. Examples of the place-and-pick regrasp are seen in the Instant Insanity demo [1] and the Handey System [2], [3]. Unfortunately, the place-and-pick regrasp tends to be slow, requiring multiple grasps and large arm motions.

The "dexterous hand" approach, first described in Salisbury's PhD thesis [4], uses three fingers, each with 3 DOFs, to perform in-hand manipulation of a grasped object. A considerable body of work followed Salisbury's lead and variations [5].

Manipulation before the grasp or as part of the grasp, such as pushing and squeezing, can be observed in early blocks-world work, and was the subject of author Mason's PhD thesis in 1982 [6]. Since that time there have been numerous studies of pushing, squeezing, levering up, and related behaviors [7], [8], [9], [10], [11], [12], [13], [14].

Hand design is central to dexterous manipulation, including the trade-offs between simplicity and complexity. Salisbury's *dexterous hand* had three fingers, with three motors per finger, all actuated. Other designs have employed anthropomorphic grippers with even more fingers and motors. At the other end of the spectrum, simple hands are typically focused on gripping [15], [16], but it is possible to use simple hands for dexterous manipulation, with as few as a single motor at the extreme [17]. The tradeoff is that a simpler mechanism might require more complex behaviors. This paper proposes a hand design of intermediate complexity, employing four motors. We employ the concepts described as *extrinsic dexterity* and *shared grasping* [17], [18], while avoiding the more complex or less robust behaviors.

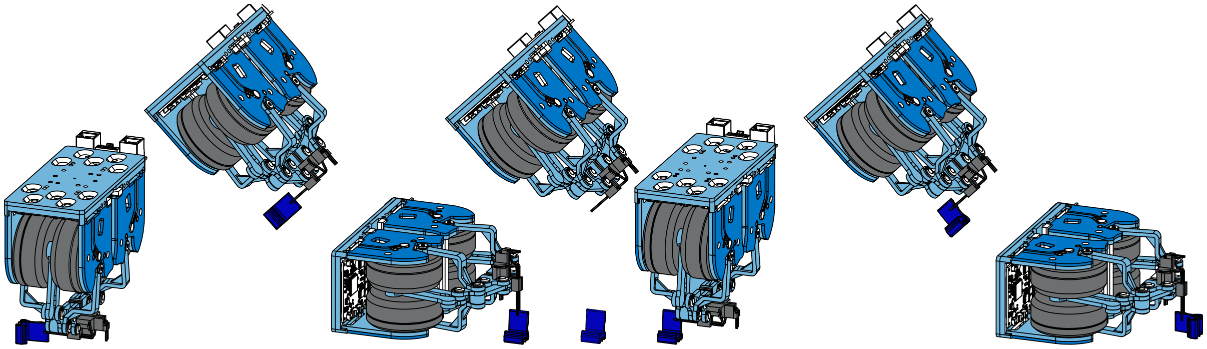


Fig. 2: The Dexterous Direct-drive Hand reorienting a long bumper using a traditional place-and-pick regrasp method.

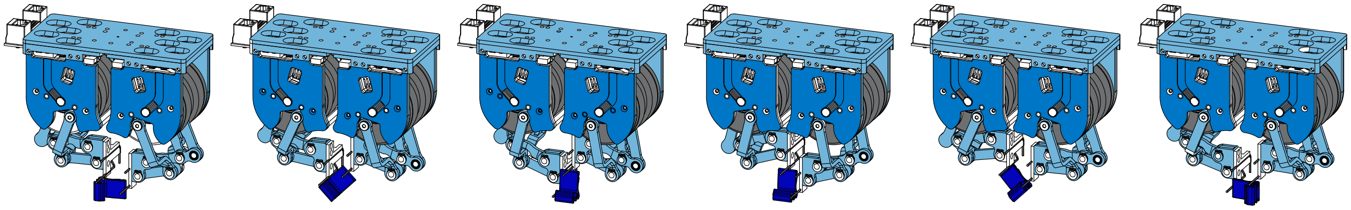


Fig. 3: The Dexterous Direct-drive Hand reorienting a long bumper using the extrinsic dexterity method developed in this work.

We design the Dexterous Direct-drive Hand (Dexterous DDHand), which is based on the earlier Direct-drive Hand (DDHand) [19]. Direct-drive actuation has been explored for manipulators for decades [20], but the use of direct-drive in hands is a recent development. The speed and natural compliance of the Dexterous DDHand are essential, allowing compliant manipulation to occur before the grasp, or as part of the grasp, without blowing the time budget.

Dexterous behaviors can be complex. Division of these complex behaviors into smaller repeating blocks or *primitives* is common in robotics. Motion primitives can be hand-designed [21], [22], generated through motion planning [11], learned from demonstrations [23] or through reinforcement learning. These motion primitives can be composed into behaviors either manually, or with some task-level planning frameworks.

III. Doorlatch Assembly Task

This paper deals with the challenge of the acquisition of parts when an immediate assemblable grasp is unavailable. The scenario is as follows. Consider the assembly of non-uniform parts which have been singulated on a surface (*e.g.* a table or conveyer belt.) The robot hand must acquire the parts in a specific assemblable pose from the surface and assemble the objects. Sometimes, the required grasp pose is not immediately feasible. It might place the hand in a collision with the ground or other environmental constraints, or the grasp pose may be out of the manipulator's workspace.

Specifically, we are concerned with the task of assembling three components into a housing that forms a subassembly for an automotive door latch, as shown in Fig. 1. These components were chosen to represent a practical use case in the industry as well as for their variations in shape and material. The small bumper and the medium bumper are made from high stiffness rubber, while the long bumper is made from a more pliable low stiffness rubber. The small and medium bumpers have three unique stable poses, while the long bumper has five. The top views of these poses are inset in Fig. 1. A complete manipulation analyzed in this paper refers to reorienting a component from any of its stable poses to its assemblable pose, which does not include the final assembly of components shown in Fig. 1.

Note that to simplify the analysis, we assume there is only one stable pose with a valid reachable grasp (pose 0 in Fig. 1). This assumption may not hold for the particular combination of objects and finger designs, meaning that our results can be improved just by choosing the most easily attained grasp pose.

IV. Design of the Robotic Work Cell

To study the acquisition task, we develop a robotic work cell (Fig. 4) that includes an industrial manipulator with the Dexterous DDHand mounted (Fig. 5) and a vision system for pose estimation of the parts. This section describes the end-effector, fingertip design, and supporting infrastructure.

A. End-effector Design

The task-level requirements of the end-effector (or hand; used interchangeably) decide its design direction:

1) Hand Payload: The maximum dimension for any parts in our case study is within 2 inches. The weight of the parts is on the order of tens of grams.
2) Arm payload: We assume a maximum payload of 5kg for the robot, which includes the weight of the end-effector and the part it is grasping.

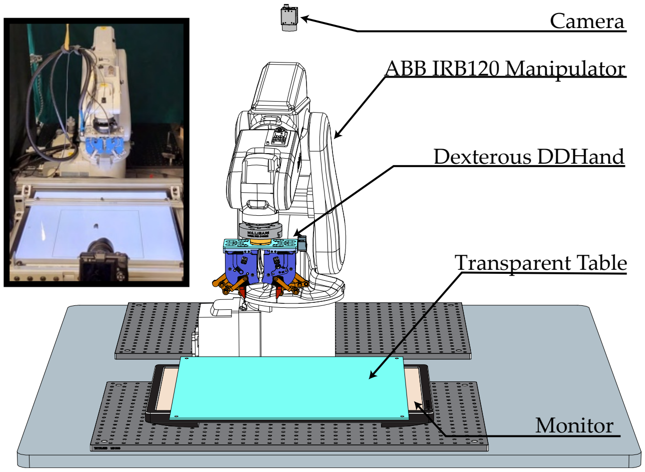


Fig. 4: A schematic of the work cell setup for the acquisition experiments. A picture of the actual setup is inset on the top left.

3) Dexterity: The hand needs a sufficient range of motion to execute the motion primitives. A hand needs to be dexterous enough to accomplish these regrasp actions.
4) Cycle Time: grasping and reorienting steps need to be quick. There are cycle time restrictions on the assembly operation.
5) Robustness: At a higher level, the manipulation actions should be repeatable with high success rates. At the hardware level, we seek low maintenance throughout the system's lifetime.

Low payload, faster cycle time and robustness requirements make direct-drive actuation a good candidate for this task. A direct-drive hand would enable high speed and transparency (transmission of force and motion between the end effector and the joints.), with the only caveat being reduced torque density. The Direct-Drive Hand (DDHand) [24] is one such end-effector system developed at the Manipulation Lab. The previous iteration of the design was made up of two finger modules, featuring a parallel 5-bar linkage connected to two brushless gimbal motors. The hand was designed to operate in a single plane and the linkage design parameters were emperically chosen. The overall weight of the DDHand falls under 2kg, well within the requirements of the task. The direct-drive actuation of the hand and the lack of gearboxes, compliant elements keeps the wear of the hand low over its lifetime.

In order to address the requirements of dexterity of the task, the DDHand design was updated. The schematic and linkage of the *Dexterous* DDHand are shown in (Fig. 5). A new 9-bar linkage was designed with three parallelogram closed-chains to mimic a parallel jaw gripper. The linkage behaves like a 2-bar serial linkage simplifying the closed-form solution for computing the forward and inverse kinematics and the Jacobian. Propagating a ground reference up the chain keeps the fingertips parallel to each other. This design choice was made to reduce complexity in the primitive design phase. The configuration, workspace and force application capability for the linkage is shown in Fig. 5. The rotor and stator bolt-circles were used as links to package the linkage in the tight space. The electronics for the hand were moved off-board close to the base of the robot and the motor conductors were routed through a 14-conductor wiring hardness and the encoder signals were differentially transmitted over two RJ45-terminated CAT5e cables. The rest of the system architecture is carried over from [24].

The direct-drive hand operates in one plane at a time. In order to manipulate objects in arbitrary dimensions, the operating plane needs to be repositioned by a supporting arm. Moving the operating plane comes with a cost as arm motions are slower than finger motions due to larger inertias and higher reduction ratios in the arm.

B. Fingertip Design

Fingertip concepts along two exploration directions were considered: type of finger contact patch and the angle of the finger with respect to the operating plane of the DDHand. Point, plane and line contact patches were explored at 0, 45 and 90 degrees with respect to the operating plane of the hand.

Two fingertips were chosen: In-plane (IP) fingers – a line contact patch parallel to the operating plane; Out-of-plane fingers – a line contact patch orthogonal to the operating plane. The fingertips were situated offset from the center of the hand to allow for manipulation when the operating plane was parallel to the ground without collision. The fingertips were fabricated using 0.05 inch Allen wrenches mounted on a 3D printed adapter for the DDHand fingertip mounting system (Fig. 5).

C. Infrastructure

The Dexterous DDHand is integrated into a system including vision, motion planning and control subsystems, and inter-process communication to link the subsystems together.

Fig. 4 shows a schematic of the system hardware. There are three main elements: The Dexterous DDHand; the vision system including a light source, a table and a camera; and the industrial manipulator for hand positioning. The hand is mounted on a force-torque sensor which connects to a manual tool changer. The effector is positioned in 6 DOF space using an ABB IRB 120 manipulator.

A PointGrey Grasshopper color camera is used in the vision system, pointed at a computer monitor which provides a high contrast white backlight against the black parts. The vision system estimates the current pose of a part with a combination of classical vision techniques (rectangle detection and SIFT-based feature detectors) to locate the part in the camera frame which is calibrated to the robot. The vision system is written in Python and runs on a dedicated 7th gen. Intel NUC i7 on a per part basis, that is, only templates for one part are compared at a time to minimize the processing overhead.

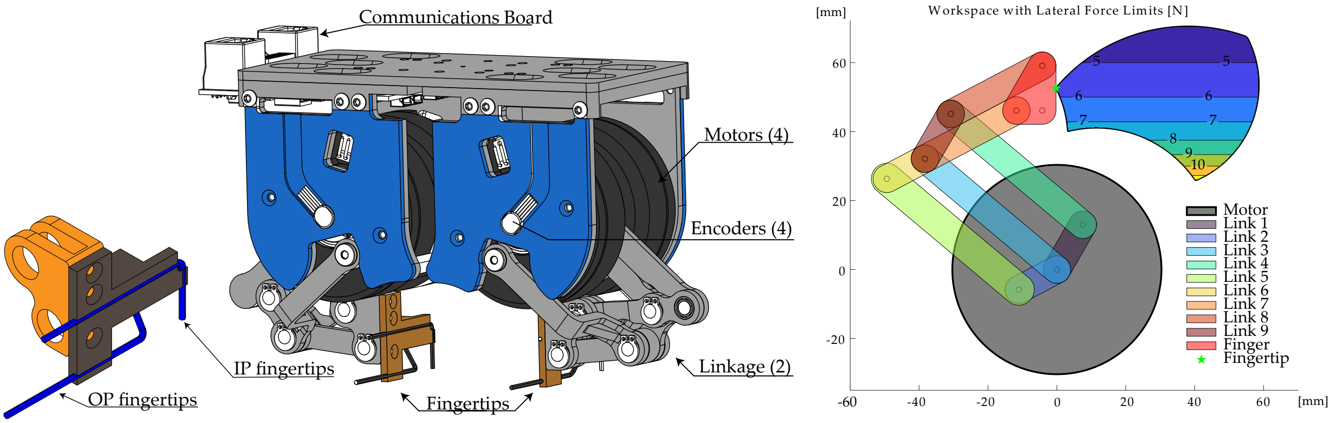


Fig. 5: Fingertip design (left), schematic (center) and workspace (right) of the Dexterous DDHand.

V. Dexterous Behaviors

The system is manually programmed with a hierarchy of motion primitives and behaviors. Once the initial pose of the part is identified, a behavior lookup identifies the appropriate sequence and specification of motion primitives to execute in order to progress to the final assembly pose of the object.

A. *Motion Primitives*

We define four motion primitives (Fig. 6) for composition into behaviors: pivot, topple, push, and squeeze grasp. These primitives are hand-designed sub-routines; they each take data in the form of 3-dimensional points and vectors, poses, or angle measures as parameters for the primitive exactly how to move with respect to the object being operated on.

Every one of the following primitives takes as input the object position and orientation as a 3-dimensional pose in the robot's frame. In the scope of our experiments, this input is provided by a 2D vision system, but other systems can be used to provide the initial pose, e.g. the pose can be computed by forward propagation of the previous action.

Each primitive takes a set of geometric quantities as parameters and executes a trajectory in the task space of the DDHand.

1) Pivot (P) and Topple (T): The pivot primitive rotates a part by some angle, about an edge in contact with the ground plane. A topple has the same effect as a pivot. The fingertip moves horizontally, making contact near the top of the object and continuing until the rotation of the object is complete. To make this action a little more robust, the fingertip motion is offset toward the center of rotation by a small amount to increase the normal force at the pivot point. With this empirical adjustment, the point of contact on the ground does not slip nominally, and the motion ends with the fingertip pinning the object to the ground.

Both pivot and topple use the initial contact point of the toppling finger as a 3D point in the object frame shown as p_1 in Fig. 6. The pivot point p_2 marks the center of rotation of the finger. This input is tuned to match the point in space at which the object will remain in contact with the ground during pivot or topple. We give a vector $\vec{a}$ to the pivot and topple primitives, where the magnitude of the vector indicates the angle θ in radians of the arc of the trajectory made by these primitives, and the direction $\hat{a}$ indicates the axis of rotation.

We derive a primitive that combines the pivot and the topple into another primitive called Pivot-Topple. This primitive results in an improvement in stability of the rotation.

The pivot-topple primitive is hand-designed for this task. The control problem for pivoting has been extensively researched ([18], [25], [26]) and a suitably robust solution may be switched out for the one implemented in this work.

2) Push (P): One finger moves horizontally through the object's initial position. A push can be used to eliminate uncertainty, aligning the part with the fingertip. The push uses the initial contact point of the pushing finger as a 3D pose relative to the object pose to mark where the pushing finger should make contact with the object at the start of the action, shown in Fig. 6 as p_2. The push primitive also uses a vector $\vec{d}$ as input, where the magnitude of the vector $|\vec{d}|$ indicates the length of the trajectory, and the normalized vector $\hat{d}$ indicates the direction in which to push.

3) Squeeze Grasp (S): Two fingertips approach the object from opposite directions and apply a squeezing force. This primitive can eliminate some uncertainty, centering and/or aligning the object. The Squeeze Grasp uses two 3D poses p_1 and p_2 to indicate the initial contact point of finger 1 and finger 2, respectively. These points indicate where the fingers should begin the squeeze. A second input is given as a binary. This input indicates to the primitive whether the squeeze should be made going "in" or "out". For example, when grabbing an object with an inside squeeze, we use "out" with this primitive. Lastly, the length of the distance that each finger should move is given as a magnitude $|\overrightarrow{d}|$. The squeezing force is determined by the distance the endpoint of the squeeze penetrates into the object and the proportional gain of the DDHand joint controller mapped to the fingertip.

VI. Experiments

The system was tested on the small, medium, and long bumpers. We manually designed behaviors to move the

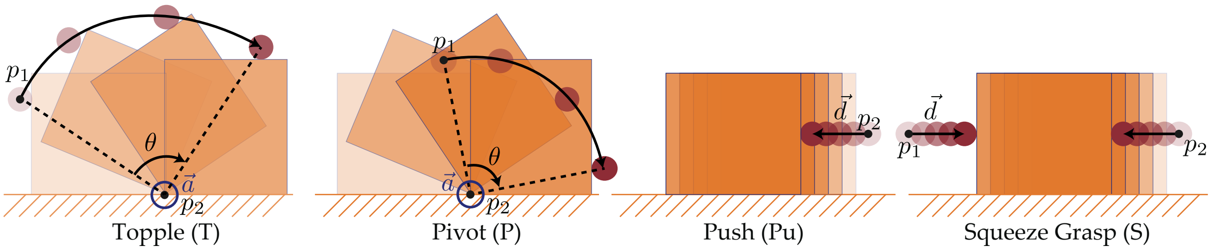
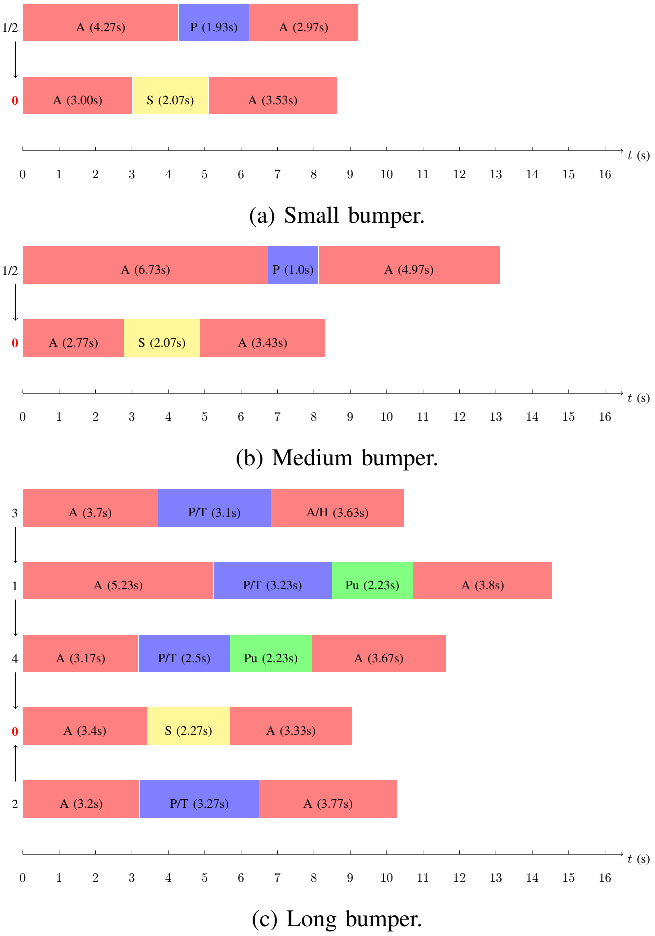


Fig. 6: Motion primitives that make up the overall behavior of the system. Pivot and topple are variations of the same motion primitive and differ only in the location of the fingers.



(a) Small bumper.

(b) Medium bumper.

(c) Long bumper.

Fig. 7: Time spent in the transition between stable poses. For the assemblable pose (0) the grasp time is included in the total time. Here A denotes the time spent in reorienting the gripper and includes the time the arm takes to clear the vision system for measurement.

Part	Pose	Success	Success Rate	Primitive Time
Small Bumper	0	55/60	91.66%	2.07
	1	30/30	100%	4.0
	2	25/30	83.33%	4.0
Medium Bumper	0	30/30	100%	2.07
	1	30/30	100%	3.07
	2	30/30	100%	3.07
Long Bumper	0	82/90	91.11%	2.27
	1	30/30	100%	12.46
	2	27/30	90%	5.54 or 15.73
	3	25/30	83.33%	5.37 or 15.56
	4	30/30	100%	7.0

TABLE I: Overview of the success rates and time taken to execute each manipulation. Note three points: 1) This is the aggregate time to execute all primitives required to reposition the part into its assemblable pose from the initial pose; 2) pose 2 and pose 3 of the long bumper each has two values for primitive time due to the uncertainty in the long bumper's orientation, which leads to different sequences of primitives used. 3) The vision system is called at each step and the vision processing is not included in this time – these times can be further reduced by removing the vision step.)

objects from one stable pose to another, as shown in Fig. 7. Fig. 3 shows a sequence of motions for pivoting the long bumper from pose 1 to 0. For each stable pose, we conducted at least 30 trials. Success was declared when the part was grasped in the hand after the final squeeze-grasp primitive in pose 0. Failure was declared if any primitives failed or the hand did not not successfully acquire the part. Failures from the vision system are not reported. At least 30 trials of each stable pose were executed with the part starting in a random location on a 8x11 rectangular light table. Table I shows the results of these experiments.

All three objects have a possibility of falling off from the squeeze grasp executed by the IP fingers, due to small pose estimation errors from the vision system. This is the only mode of failure in pose 0 for all objects.

The second failure mode is seen when a starting poses appears as a degenerate rectangle in the top view. For the small and medium bumpers, pose 1 and 2 look very similar in the top view, thus the vision system cannot distinguish between them. To handle this, the same behavior is implemented for both pose 1 and 2. This results in different pivoting forces for both poses resulting in a lower success probability for one of them. The medium bumper doesn't show this behavior as it has a length to width ratio that is close to unity.

The issue of inconsistencies in the reporting of stable poses by the vision system also shows up in the reorientation of the long bumper. Consider the long bumper in pose 2 or 3. Depending on the orientation of the rectangle, a pivot-topple can result in either pose 0 or pose 1. For this reason, we report two times for the execution of the long bumper.

The longer times are the worst case scenario when the manipulation of pose 2 or 3 results in pose 1 and the shorter times are the best case scenarios when the result is pose 0. For ease of reporting, we show the best case execution for pose 2 and the worst case execution for pose 3 in Fig. 7.

In the worst case, the long bumper has to go through is pose 2/3 → pose 1 → pose 4 → pose 0. This long chain of stable poses increases the probability of failure when the initial stable pose is pose 2 or 3.

VII. Discussion

This paper shows the utility of extrinsic dexterity in a real-world application. By pre-grasping with a dexterous and agile end-effector, we show an alternative to place-and-pick regrasping actions.

In addition, our prototype system solves an industrial-relevant problem that is extendible through state-of-the-art approaches in robotic manipulation. We discuss three such extensions in this section.

A. Ease of Redeployment

Making a case for automation is challenging in manufacturing. Two issues govern the economics of deploying automation: high throughput requirements in favor of and constant refreshes against. The popular teach-and-repeat method requires laborious reprogramming, making automation systems less economical. The method described in the paper takes a different approach. Instead of programming teach points in the task space of the end-effector, the method provides generalized primitives parameterized by geometric quantities. Reconfiguring the system for a new part requires only geometric knowledge of the part to inform the parameterization of the primitives and their order of execution.

This method can also benefit from manipulation research in the automatic generation of primitives given a task description, including trajectory optimization [27], sampling [11], [28] and learning-based [29] methods.

B. Robust execution of Motion Primitives

At the primitive level, the work implements hand-designed trajectories parameterized by geometric quantities. These are executed on the system with an impedance control scheme using a proportional-derivative controller to simulate a spring-damper system. This open-loop execution with compliance does not result in primitives robust to initial, sensor, or process noise. This results in some of the failure modes for the reported experiments. The primitives can be further augmented with control policies that have been recently shown robust to noise in geometry and initial pose [30].

C. Uncertainty Reduction Motions

Ideally, executing a primitive on a component would reliably result in the target stable pose, but the exact object pose is often subject to uncertainty. Moreover, since multiple primitives must be sequenced to get the object into its final assemblable stable pose, the errors could propagate and accumulate. Fig 8a shows an example of this uncertainty stack-up of the long bumper after two pivot primitives to get from pose 2 to pose 0. Errors compound with each action, eventually preventing additional primitives from being reliably executed without localizing the object.

In the current implementation, the vision system gives accurate object pose estimation after every primitive. Our vision system has a high accuracy of estimating the exact pose of the object, but calling this vision system increases the time as the robot needs to clear the workspace to avoid occlusions. It is beneficial to minimize the calls to the vision system and query object stable pose only at the beginning and end of the reorientation operation.

Our preliminary experiments show that certain actions can help eliminate pose uncertainties to a certain extent. Actions like pivot and topple increase the uncertainty of the final pose of the part, while actions like push and squeeze grasp reduce it [31]. Fig. 8b shows the propagation of uncertainty in the manipulation from pose 1 to 4 of the long bumper with a single push. The uncertainty collapses in one direction. Fig 8c shows the effect of uncertainty from pose 4 to 1 with two pushes. The uncertainty is now completely collapsed.

VIII. Acknowledgement

This work was supported by the NSF under grants IIS-1813920 and IIS-1909021 and by an anonymous industrial sponsor.

References

[1] R. Paul, “Modelling, Trajectory Calculation and Servoing of a Computer Controlled Arm:,” Defense Technical Information Center, Fort Belvoir, VA, Tech. Rep., Nov. 1972.

[2] P. Tournassoud, T. Lozano-Perez, and E. Mazer, “Regrasping,” in *1987 IEEE International Conference on Robotics and Automation Proceedings*, vol. 4, Mar. 1987, pp. 1924–1928.

[3] T. Lozano-Perez, J. Jones, E. Mazer, P. O’Donnell, W. Grimson, P. Tournassoud, and A. Lanusse, “Handey: A robot system that recognizes, plans, and manipulates,” in *1987 IEEE International Conference on Robotics and Automation Proceedings*, vol. 4, Mar. 1987, pp. 843–849.

[4] J. K. Salisbury Jr., “Kinematic and Force Analysis of Articulated Hands,” PhD Dissertation, Stanford University, 1982.

[5] A. Okamura, N. Smaby, and M. Cutkosky, “An overview of dexterous manipulation,” in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065)*, vol. 1, Apr. 2000, pp. 255–262 vol.1.

[6] M. T. Mason, “Manipulator grasping and pushing operations,” Ph.D. dissertation, Massachusetts Institute of Technology, 1982.

[7] K. M. Lynch and M. T. Mason, “Stable pushing: Mechanics, controllability, and planning,” *The international journal of robotics research*, vol. 15, no. 6, pp. 533–556, 1996.

[8] E. Huang, A. Bhatia, B. Boots, and M. Mason, “Exact bounds on the contact driven motion of a sliding object, with applications to robotic pulling,” in *Proceedings of Robotics: Science and Systems*, 2017.

[9] A. Rao, D. J. Kriegman, and K. Y. Goldberg, “Complete algorithms for feeding polyhedral parts using pivot grasps,” *IEEE Transactions on Robotics and Automation*, vol. 12, no. 2, pp. 331–342, 1996.

[10] Y. Hou, Z. Jia, A. Johnson, and M. Mason, “Robust planar dynamic pivoting by regulating inertial and grip forces,” in *Algorithmic Foundations of Robotics XII. Springer Proceedings in Advanced Robotics*, 2020.

[11] X. Cheng, E. Huang, Y. Hou, and M. T. Mason, “Contact mode guided sampling-based planning for quasistatic dexterous manipulation in 2d,” in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 6520–6526.

[12] K. Y. Goldberg, “Orienting polygonal parts without sensors,” *Algorithmica*, vol. 10, no. 2, pp. 201–225, 1993.

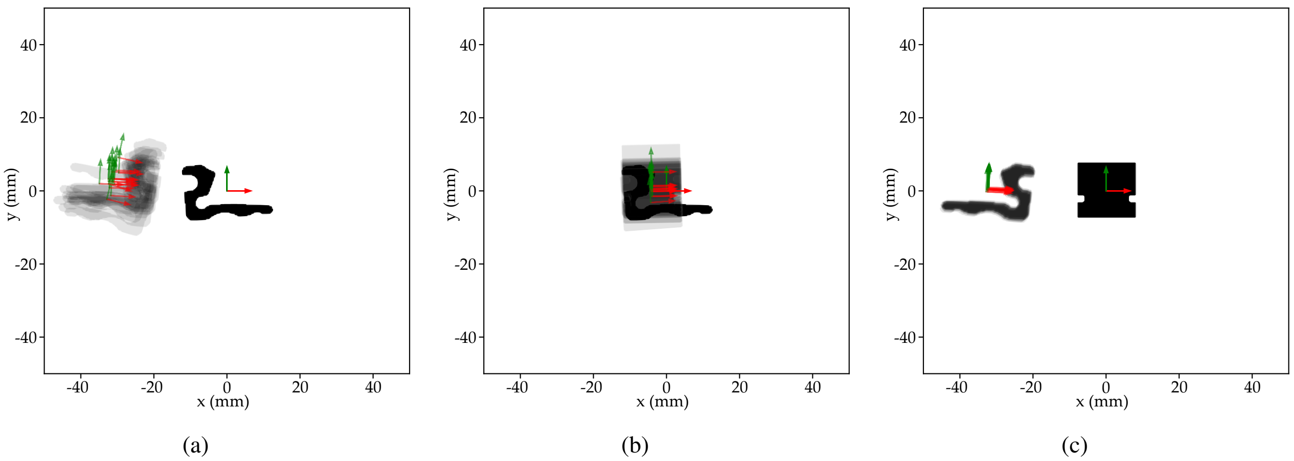


Fig. 8: 30 Trials of the long bumper executing (a) pose 0 → 1, (b) pose 0 → 4 with a push along the x-axis; (c) pose 4 → 1 with a push along the x-axis and y-axis. In each case the initial pose is in black, centered, and 30 resulting poses are overlapping in gray, to the left.

[13] N. Chavan-Dafle and A. Rodriguez, "Prehensile pushing: In-hand manipulation with push-primitives," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 6215–6222.

[14] V. Babin, D. St-Onge, and C. Gosselin, "Stable and repeatable grasping of flat objects on hard surfaces using passive and epicyclic mechanisms," *Robotics and Computer-Integrated Manufacturing*, vol. 55, pp. 1–10, 2019.

[15] M. T. Mason, S. S. Srinivasa, and A. S. Vazquez, "Generality and simple hands," in *Robotics Research*. Springer, 2011, pp. 345–361.

[16] A. M. Dollar and R. D. Howe, "A robust compliant grasper via shape deposition manufacturing," *IEEE/ASME transactions on mechatronics*, vol. 11, no. 2, pp. 154–161, 2006.

[17] N. Chavan-Dafle, A. Rodriguez, R. Paolini, B. Tang, S. Srinivasa, M. Erdmann, M. Mason, I. Lundberg, H. Staab, and T. Fuhlbrigge, "Extrinsic dexterity: In-hand manipulation with external forces," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, 2014, pp. 1578–1585.

[18] Y. Hou, Z. Jia, and M. Mason, "Manipulation with shared grasping," in *Proceedings of Robotics: Science and Systems*, Corvalis, Oregon, USA, Jul. 2020.

[19] A. Bhatia, A. M. Johnson, and M. T. Mason, "Direct drive hands: Force-motion transparency in gripper design," in *Robotics: Science and Systems*, 2019.

[20] H. Asada, T. Kanade, and I. Takeyama, "Control of a direct-drive arm," *The ASME Journal of Dynamic Systems Measurement and Control-transactions*, vol. 105, no. 3, pp. 136–142, 1983.

[21] N. C. Dafle, A. Rodriguez, R. Paolini, B. Tang, S. S. Srinivasa, M. Erdmann, M. T. Mason, I. Lundberg, H. Staab, and T. Fuhlbrigge, "Extrinsic dexterity: In-hand manipulation with external forces," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, May 2014, pp. 1578–1585.

[22] M. Pivtoraiko and A. Kelly, "Kinodynamic motion planning with state lattice motion primitives," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sep. 2011, pp. 2172–2179.

[23] P. Pastor, M. Kalakrishnan, S. Chitta, E. Theodorou, and S. Schaal, "Skill learning and task outcome prediction for manipulation," in *2011 IEEE International Conference on Robotics and Automation*. Shanghai, China: IEEE, May 2011, pp. 3828–3834.

[24] A. Bhatia, A. M. Johnson, and M. T. Mason, "Direct drive hands: Force-motion transparency in gripper design," in *Robotics: Science and Systems*, Jun. 2019.

[25] Z. J. Yifan Hou and M. T. Mason, "Fast planning for 3d any-pose-reorienting using pivoting," in *Proceedings of (ICRA) International Conference on Robotics and Automation*. IEEE Robotics and Automation Society (RAS), May 2018, pp. 1631 – 1638.

[26] A. Holladay, R. Paolini, and M. T. Mason, "A General Framework for Open-Loop Pivoting," in *IEEE Int. Conf. on Robotics and Automation.*, 2015.

[27] J.-P. Sleiman, J. Carius, R. Grandia, M. Wermelinger, and M. Hutter, "Contact-Implicit Trajectory Optimization for Dynamic Object Manipulation," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Nov. 2019, pp. 6814–6821.

[28] X. Cheng, E. Huang, Y. Hou, and M. T. Mason, "Contact mode guided motion planning for quasidynamic dexterous manipulation in 3d," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 2730–2736.

[29] O. Kroemer, C. Daniel, G. Neumann, H. van Hoof, and J. Peters, "Towards learning hierarchical skills for multi-phase manipulation tasks," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, May 2015, pp. 1503–1510.

[30] Y. Hou and M. T. Mason, "Robust Execution of Contact-Rich Motion Plans by Hybrid Force-Velocity Control," *2019 International Conference on Robotics and Automation (ICRA)*, pp. 1933–1939, May 2019.

[31] J. Zhou, R. Paolini, A. M. Johnson, J. A. Bagnell, and M. T. Mason, "A Probabilistic Planning Framework for Planar Grasping Under Uncertainty," *IEEE Robotics and Automation Letters*, vol. 2, no. 4, pp. 2111–2118, 2017.