





In-Hand Manipulation With a Simple Belted Parallel-Jaw Gripper

Gregory Xie , *Member, IEEE*, Rachel Holladay , *Graduate Student Member, IEEE*, Lillian Chin , *Member, IEEE*, and Daniela Rus , *Fellow, IEEE*

Abstract—Many manipulation tasks require task-specific grasps that may initially be unavailable, requiring the robot to transition from grasp to grasp. Current approaches to in-hand manipulation often involve complex hardware or algorithms, which limits their use in real-world tasks. In this work, we present an integrated approach to in-hand manipulation through Belt Orienting Phalanges (BOP), maximizing gripper capability by considering both hardware and algorithm design. BOP is a parallel-jaw gripper where each finger has two sets of belts. Together, the two fingers' belts can impart forces and torques on to a grasped object to control its roll, pitch and translation. These movements form the basis of motion primitives that can be sequenced together for in-hand manipulation of objects as well as for complex motions such as syringe actuation and fingernail-style lifting. We characterize these motion primitives, develop a grasp-to-grasp motion planner, and demonstrate the potential of BOP through the real-world example of screwing a light bulb into a socket.

Index Terms—Dexterous manipulation, grippers and other end-effectors, in-hand manipulation.

I. INTRODUCTION

MANY real-world manipulation tasks require task-specific grasps, where the robot may be required to exert a force through a grasped object or may have geometric constraints on how the object is positioned [1], [2], [3]. For example, to screw a light bulb into a socket, the robot must grasp the bulb such that the bulb's threads are not occluded and such that the robot can exert the torque required to screw the bulb in.

However, in constrained environments, the robot may not be able to initially select a grasp that fulfills these requirements. In those cases, the robot must transition from an initially reachable grasp to a task-suitable grasp. Regrasping the object by placing it in an intermediate position is one viable strategy to achieve this transition, but it is usually significantly slower than performing in-hand object manipulation, especially for dexterous multi-step tasks [4], [5].

Simple and anthropomorphic grippers have demonstrated substantial in-hand object re-orientation capabilities [6], [7],

Manuscript received 20 August 2023; accepted 4 December 2023. Date of publication 25 December 2023; date of current version 29 December 2023. This letter was recommended for publication by Associate Editor Huixu Dong and Editor Hong Liu upon evaluation of the reviewers' comments. This work was supported by NSF EFRI under Grant 1830901 and in part by the Gwangju Institute of Science and Technology. (*Corresponding author: Gregory Xie.*)

The authors are with the Computer Science and Artificial Intelligence Lab, MIT, Cambridge, MA 02139 USA (e-mail: gregoryx@csail.mit.edu; rholladay@csail.mit.edu; ltchin@csail.mit.edu; rus@csail.mit.edu).

This letter has supplementary downloadable material available at <https://doi.org/10.1109/LRA.2023.3346750>, provided by the authors.

Digital Object Identifier 10.1109/LRA.2023.3346750

[8], [9], but few, if any have accomplished this with both simple control algorithms and hardware. Meanwhile, it is unclear if grippers specialized for in-hand manipulation are capable of providing the forces needed for dexterous multi-step tasks. We advocate for a systematic approach that strikes a balance between hardware complexity, algorithmic complexity, and gripper strength.

We present Belt Orienting Phalanges (BOP), a belted parallel-jaw gripper that provides a large in-hand grasp manipulation set and is capable of applying significant forces and torques on the grasped object. Taking inspiration from Ulrich [10], we maximize gripper capability by simultaneously considering both the hardware and the algorithms used to control it. BOP enables us to have straightforward control over the grasp pose while remaining mechanically simple and morphologically similar to simple grippers.

The gripper maintains control over the grasp through the use of four parallel belts oriented along the gripper fingers, two on each finger. As seen in Fig. 1, through combinations of belt movements, we are able to control the roll, pitch, and translation of the grasped object. We develop simple, proof-of-concept closed-loop controllers for each motion and use them as motion primitives in a geometric planner to generate grasp-to-grasp motion plans. Finally, we demonstrate our gripper and grasp-to-grasp planner in solving a real-world manipulation task by integrating them into a state-of-the-art task and motion planning (TAMP) framework [11].

In summary, we contribute the following:

- 1) Design and fabrication of Belt Orienting Phalanges (BOP), a belt-driven finger that when paired with a base gripper, allows for in-hand control of object translation, pitch and roll.
- 2) Characterization of BOP's force and torque limits, as well as reorientation performance.
- 3) Development of a grasp-to-grasp motion planner and integration into a task and motion planning (TAMP) framework.
- 4) Demonstration of BOP in accomplishing the real-world manipulation task of screwing a light bulb into a socket as well as executing alternate manipulation skills.

II. RELATED WORK

Previous in-hand manipulation systems can be viewed on a spectrum of mechanical complexity ranging from anthropomorphic hands to simple grippers. Anthropomorphic hands

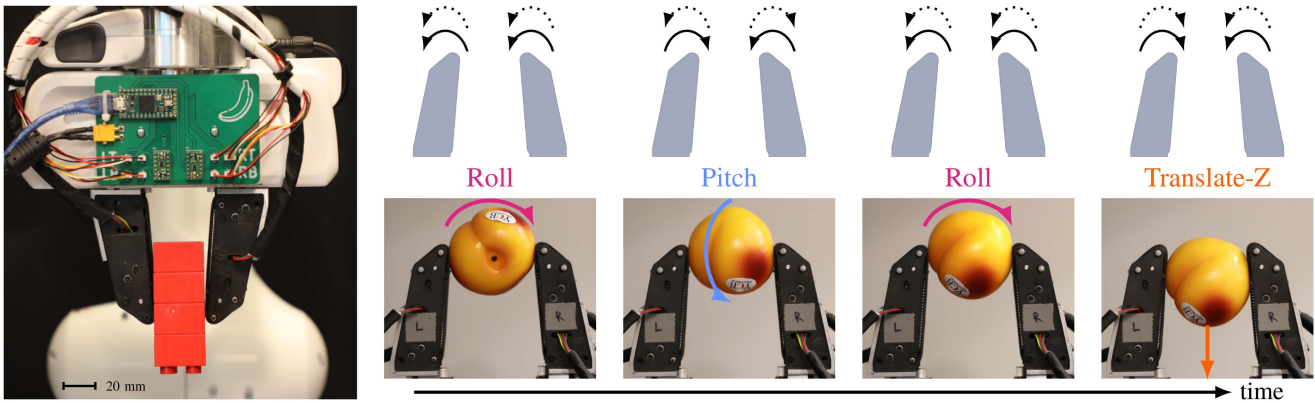


Fig. 1. Belt Orienting Phalanges (BOP), mounted on a Franka Emika Panda hand. Each finger has two belts that can move independently from one another. By controlling the movement of the four belts, we can translate the object within the hand and control its roll and pitch. BOP is able to sequence multiple motion primitives to achieve in-hand manipulation (e.g. roll, pitch, roll, Z -translation). The solid and dotted arrows describe the movement of the close and far belts, respectively.

with many degrees of freedom (DoFs) have demonstrated impressive in-hand manipulation capabilities, with learned controllers demonstrating finger-gaiting, finger-pivoting, and other dynamic behaviors [6], [7], [12]. However, these systems tend to be both mechanically and algorithmically complex as contact is repeatedly made and broken across 10-20 DoFs.

Similarly impressive in-hand manipulation results have been accomplished using simple parallel-jaw grippers by using complex algorithms to leverage forces external to the gripper, including gravitational, contact, or inertial forces [8], [9], [13], [14]. These systems are able to maintain hardware simplicity, but often at the cost of requiring accurate models of friction and dynamics.

A final class of grippers lie between these two extremes: they specialize their hardware towards certain types of in-hand manipulation to reduce algorithmic complexity. We will focus on this class, as BOP has a comparable number of DoF to and is most similar to these grippers.

One direction of hardware specialization is to simplify the high-DoF approach seen in anthropomorphic hands by finding the minimal number of minimally complex fingers that allows the hand to accomplish similar manipulation tasks. Early work on non-anthropomorphic, multi-fingered hands include the Stanford/JPL hand [15] and the UPenn hand [10].

In this class of specialized multifingered grippers, recent work has focused on using simulation to explore the design space of hands with specific finger arrangements [16] and developing in-hand manipulation capabilities for hands with underactuated fingers [17], [18], [19], [20]. Despite the decrease in hardware complexity compared to anthropomorphic hands, general in-hand manipulation with these grippers still requires the use of complex dynamic movements such as finger gaiting, requiring coordination between multiple fingers and reasoning over discontinuities introduced by contact.

Another orthogonal direction of specialization is the use of active surfaces. In these approaches, object movement is strongly coupled to the movement of the gripper surface, avoiding the need to make and break contact with the grasped object. Placing

belts parallel to the fingers allows for objects to be pulled further into a grasp or rolled around an axis perpendicular to the belts [21], [22], [23], [24], [25]. Gómez-de Gabriel and Wurdemann [26] and Rahman et al. [27] rotate the active surfaces by 90 degrees, creating a gripper where the active surface moves perpendicular to the fingers. Yuan et al. [28], [29] drive motion through coordinating the orientation of three cylinders or spheres.

BOP uses a pair of active surfaces in each finger, imparting pitch and roll control in addition to translational movement. This provides a larger reachable grasp set than other active surface grippers while still maintaining morphological similarity to traditional parallel-jaw grippers. The use of active surfaces also allows BOP to change grasps while maintaining contact with the object surface, avoiding the need to make and break contact. All together, these features enable simple control and planning for in-hand manipulation.

III. SYSTEM DESIGN

To present BOP, we first establish our design goals. We then describe the hardware design, a proof-of-concept closed-loop control strategy and a RRT-based planner that is used to plan grasp-to-grasp motions.

A. Design Goals

Our design goals for this gripper were to create an end effector that 1) could reach a large set of grasps through in-hand movements 2) had a simple mapping from control input to movement of the grasped object, 3) struck a balance between the amount of mechanical and algorithmic complexity, and 4) was morphologically similar to a traditional parallel-jaw gripper. The first two goals are critical for in-hand manipulation and the third relates to hardware and software robustness by reducing the number of possible failure points [30]. The last goal allows us to leverage off-the-shelf grippers to exert significant grasp forces and existing methods for grasp candidate generation.

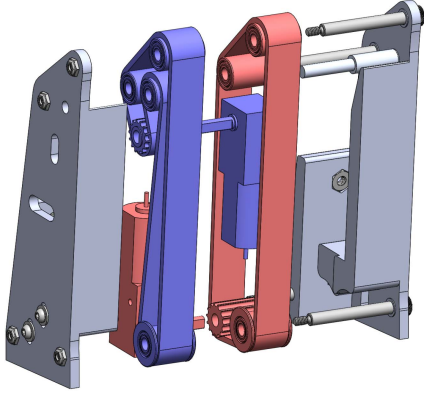


Fig. 2. Exploded model view of one finger. The blue and red colorations highlight each belt system with associated motor. The drive system for each belt is contained in the finger.

Briefly ignoring object geometry, we target the set of grasps that reorient the object in $SO(3)$. Achieving this requires a minimum of two rotations around non-redundant axes. We define a frame G which is rigidly attached to the gripper at the center of the line between the grasp contact points (Fig. 3). We assume the grasp contact points are always at the finger tips. We choose to target rotations around the X and Y axes of G , which allows reach of any $SO(3)$ rotation through an extrinsic X - Y - X or Y - X - Y rotation sequence, which follows two popular extrinsic Euler angle conventions. An example of this rotation sequence is shown in Fig. 1, where the gripper performs a roll-pitch-roll (X - Y - X) sequence to reorient a sphere to an arbitrary rotation.

With a translational degree of freedom, any transform in $SE(3)$ can be reached through a rotation-translation-rotation sequence. The first rotation and translation reaches the desired translation, while the last rotation reaches the desired orientation without affecting translation. This analysis only motivates the controllable degrees of freedom of our gripper and the actual set of reachable grasps is dependent on the object geometry.

We assume access to a force-controlled base gripper with interchangeable fingers and choose to design active replacement fingers rather than designing an entire end effector assembly.

B. Hardware Design

Our finger design, as shown in Fig. 2, has two belts on each finger, allowing for three directions of object movement:

Z-Translation. By driving all belts in a common direction, we translate the grasped object along the Z -axis of G (Fig. 1-Translate- Z).

Roll. By driving the belts on one finger in the opposite direction of the belts on the other, we roll the grasped object around an axis parallel to the X -axis of G (Fig. 1-Roll).

Pitch. By driving the belts on each finger differentially, we pitch the object around the Y -axis of G (Fig. 1-Pitch).

We use off-the-shelf GT2 timing belts for overall ease of manufacturing and for their large tension carrying capability. To prevent excessive scrubbing of the object on the belt during

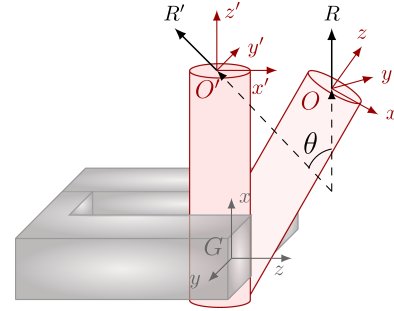


Fig. 3. Motion primitive controllers use an approximation for the change in object pitch θ . G is the gripper frame, while O is the object frame. As the object moves from start grasp to goal grasp, we estimate θ as the angle between R , the projected ray at the initial grasp onto the XZ plane of G and R' , a similarly projected ray from the current grasp onto G . The change in roll is estimated similarly, with projections going onto the XY plane of G . This approximation is only valid when the change in either pitch or roll is small.

rotations, the fingers are slightly tilted inward so that the tips of the fingers are closer together than the base. This allows us to assume that during normal operation, only the tips of the fingers contact the grasped object. Additionally, the tilt of the fingers allows us to achieve a three-point enveloping power grasp when the grasped object contacts the sides of the fingers and the palm.

The 3D printed sides (Markforged Onyx) also include support surfaces located behind the belts so that, when grasping using the middle of the fingers, the grasp force does not have to be completely supported by belt tension. Low friction PTFE tape is used to minimize friction between the belts and the support surface.

We size a gearbox and motor combination (1030:1, N20) so that, at a nominal contact normal force of 40 N, the belts slip on the object before the motor stalls. We design for a worst-case scenario, with an unrealistic friction coefficient of 1. The large gear ratio results in low motor speeds which limit the rate of manipulation. We use non-backdrivable worm gearboxes so that the belts do not have to be actively controlled when in-hand movements are not desired.

A PCB carries the motor drivers (TI DRV8835), a microcontroller (PJRC Teensy 4.0), and an interface to the host computer through a USB-serial connection. Encoders on the motors allow for closed loop belt velocity and belt position control. Including the PCB and wiring to the fingers, but excluding mounting adapters to the base gripper and other wiring, a system with two fingers weighs approximately 230 grams. Each finger is approximately $38 \times 86 \times 28$ mm.

Borrowing the terminology from multimodal motion planning [31], one important limitation of our hardware design is that we can only accomplish in-hand manipulations within a single mode. We define a mode as the set of object surface-gripper finger contacts, surfaces on an object are separated from each other along areas of high curvature. In order to switch modes, BOP would need to traverse an area of high curvature, which we do not allow because the fingers are unlikely to maintain contact with the object (e.g. rotating over the edges of a cube). The maximum curvature that the fingers can travel over will depend on the frictional properties between the object and finger. Another

Algorithm 1: $\text{Extend}(T, s_{near}, s_{target})$.

```

1: candidates = []
2: for  $p \in \text{primitives}$  do
3:   for  $t \in \text{steps}$  do
4:      $s_{candidate} = \text{SimulatePrimitive}(p, t, s_{near})$ 
5:     candidates.Add( $s_{candidate}$ )
6:  $s_{new} = \text{BestState}(s_{target}, \text{candidates})$ 
7:  $T.\text{AddVertex}(s_{new}); T.\text{AddEdge}(s_{near}, s_{new})$ 
8: return  $s_{new}$ 

```

limitation of the hardware is that some in-hand manipulations may not be possible if they cause a collision between the object and the gripper. Even with these limitations, the hardware design of BOP enables large amounts of in-hand manipulation.

C. Controller Design

We design proof-of-concept closed loop controllers for each of the motion primitives: Z -translation, pitch and roll. We define a frame O which is rigidly attached to the manipulated object. The feedback for the controllers comes from an external vision system providing an estimate of the object to grasp transform ${}^G\hat{X}^O$. We use a motion capture system (Optitrack) to provide this estimate, but any pose estimator could be used. We assume the object is rigid.

We use proportional controllers to control roll, pitch and Z -translation of the grasped object through a velocity control interface.

To track multiple rotations, we use a simple approximation to roll and pitch, visualized in Fig. 3. We find this approximation works well when the change in one of pitch or roll is small. In our case, this is true because we are interested in controlling motion along one primitive while maintaining zero movement along the others.

To estimate the change in Z -translation, we subtract the effects of roll and pitch before computing the difference between the current object frame and the initial object frame along the Z axis of G .

We run multiple controllers during a motion primitive to reduce extraneous movement. Experimentally, we observe that we need to run all three controllers during the execution of a Z -translation primitive, and only the Z -translation controller with either the roll (or pitch) controller during a roll (or pitch) primitive. We stop the controllers and end the motion primitive if the errors for each of the active controllers are less than user-defined thresholds or the motion takes longer than a user-defined timeout. We evaluate the accuracy of our motion primitives in Section IV.

D. In-Hand Motion Planning

To plan in-hand motions that move the manipulated object from a start grasp to a goal grasp, we use bidirectional RRT-Connect [32], [33]. We assume that the object is rigid and the object model is known. The use of a motion planner allows our system to generalize across objects. We are able to easily use RRT with few modifications because of the relationship between

control input and object movement enabled by our closed-loop motion primitives.

Our state is the grasp transform represented as a translation and quaternion: $s = (x, y, z, q_x, q_y, q_z, q_w)$. In the standard bidirectional RRT-Connect algorithm, we replace the `Sample()` and `Extend()` functions.

To generate random states, we rejection-sample grasps using a heuristic from Tedrake [34]. We sample a random point (p_1) on the object's mesh and ray-cast along its mesh normal (n_1) to find the point and its associated normal (p_2, n_2) on the mesh, opposite from the sampled point. We create a grasp transform with the y -axis of the frame aligned with n_1 at the average of p_1 and p_2 . We reject grasps if the grasp contact normals are not well aligned, if the grasp width is too large, or the grasp is in collision. Because the space of valid grasps is much smaller than the space of rigid body transforms, we choose to sacrifice probabilistic completeness and only sample grasps using the heuristic.

The `Extend()` (Algorithm 1) uses the motion primitives described in the previous section. For each primitive, we simulate a discrete number of motions in the positive and negative directions, incrementing the translation or angle by a fixed step size. To select the best primitive and step, we use the geodesic distance, with weight chosen so that a translation 10 cm is equivalent to a rotation of π rad. In the planner, we consider two states to be equal if the distance is below a user-specified threshold.

The grasp set reachable by this planner is limited by both the grasp validity criteria and by only sampling with the heuristic. The sole use of the heuristic can prevent the planner from sampling certain grasps (e.g. we will never sample graphs where both contact normals are misaligned) and the validity criteria can reject grasps that are actually valid (e.g. all grasps on a square pyramid, depending on the normal tolerance). This can prevent the planner from finding paths between two grasps even if the hardware is capable of traversing between them.

This could be resolved with a higher quality grasp validity criteria that can more accurately separate valid grasps from invalid grasps and the reintroduction of uniform sampling, with the tradeoff of increased planning time.

IV. PERFORMANCE CHARACTERIZATION

We evaluate BOP's performance with two experiments: 1) quantifying the force and torque that can be exerted by the motion primitives and 2) quantifying the accuracy and repeatability of the motion primitives under closed-loop control. These experiments help define the set of objects that can be manipulated in-hand and confirm that several primitives can be accurately sequenced together for grasp-to-grasp planning.

For both experiments, we mount the fingers to a back-drivable test platform driven by a torque-controlled BLDC (mjbots mj5208, mjbots moteus r4.5). We define the grasp force to be two times the contact force at one of the gripper fingers.

A. Force and Torque

To measure the forces and torques from BOP, a force/torque sensor (ATI gamma SI-32-2.5) is attached to a Franka Emika

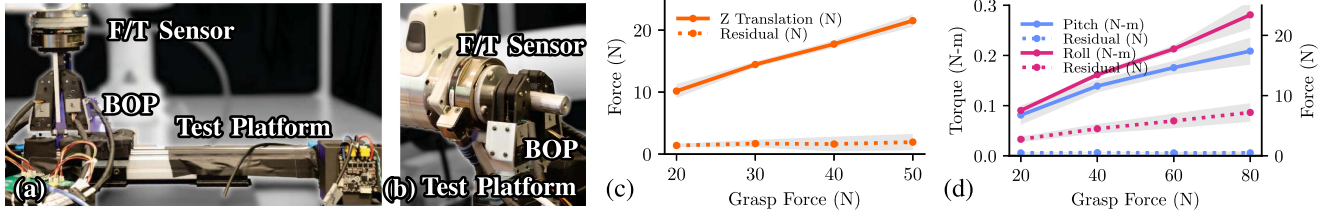


Fig. 4. Force and Torque Characterization for the three motion primitives. (a) The setup used to characterize the force exerted by the Z -translation primitive and the torque exerted by the pitch primitive. (b) The setup used to characterize the torque exerted by the roll primitive. (c) and (d) Both graphs show the result of 20 trials for each motion primitive swept over a range of grasp forces. We also report a grasp force residual, which is the difference in force that each finger is applying due to non-ideal robot stiffness control. The standard deviation is shown in grey.

Panda manipulator. We command zero stiffness in all of the Cartesian axes except for the axis corresponding to the motion primitive being evaluated. To test the force and torque exerted by the Z -translation and pitch primitives, we mount a 6.35 mm flat plate at the XZ plane of the force/torque sensor reference frame (Fig. 4(a)). To test the torque exerted by the roll primitive, we mount a 19.05 mm cylinder along the Z axis of the force/torque sensor reference frame (Fig. 4(b)).

For each trial, we close the test platform on the test plate (or cylinder) with the specified grasp force and command the finger belt velocity to the corresponding primitive. We report the resulting force or torque across 20 trials, 10 in the positive direction and 10 in the negative direction. Although the robot is commanded to exhibit zero stiffness, joint friction and other effects cause an unequal force to be exerted by each finger, which we report as a grasp force residual. Because of the grasp force residual, the forces and torques BOP is able to exert on objects during in-hand manipulation in free space will be higher than the ones measured in this experiment.

As shown in Fig. 4(c) and (d), all primitives have a roughly linear relationship with grasp force. We see that the Z -translation primitive can apply a force of approximately 18 N at a grasp force of 40 N. Assuming a conservative friction coefficient (0.23, about half of what was observed in this experiment) and that the object geometry is such that a valid grasp exists, only four objects from the Yale-CMU-Berkeley (YCB) dataset are too heavy to manipulate using the Z -translation primitive without slipping [35].

At the same grasp force, the pitch primitive can apply a maximum torque of approximately 0.14 Nm. Assuming the belts contact the object at a point, we estimate an effective contact-to-contact distance of 15 mm. Assuming the center of mass of the grasped object is 5 cm from the finger tips, the same conservative friction coefficient as above, and a grasp force of 40 N, balancing torques show that the gripper should be able to use its pitch primitive to manipulate objects that weigh less than 140 grams in any orientation.

Under the above assumptions, there is a large difference in the maximum object weight for the Z -translation (0.9 kg) and pitch (0.14 kg) primitives at the same grasp force. This is due to the small contact patch of the gripper.

At a grasp force of 40 N, the roll primitive is able to exert a torque of 0.16 Nm on the test cylinder. In this experimental setup, the roll torque is related to the cylinder diameter (19.05 mm).

We do not characterize this relationship. Overall, this experiment shows that BOP is able to exert significant forces and torques on the grasped object.

B. Motion Primitives

We first test the accuracy and consistency of a single closed-loop motion primitive on 3D printed objects: a cube ($50 \times 50 \times 50$ mm), a cylinder (50×50 mm), and a sphere (50 mm). We test our Z -translation and pitch primitives on the cube, all three primitives on the cylinder, and both rotation primitives on the sphere. We use a grasp force of 20 N. We use a plastic jig to place the object in the gripper at a nominal position and then command a closed-loop motion primitive, shown in Fig. 5(a). The rotation primitives are commanded to a maximum of π radians and the Z -translation primitive is commanded to a maximum of 20 mm. In Fig. 5(b), (c), and (d) we report the distance between the expected pose and the measured pose for five trials per motion primitive and object, using the distance metric described at the end of Section III-D. For scale, an error of 0.001 is equivalent to a rotation error of 0.03 radians or a translation error of 1 mm.

Overall, the motion primitives are both accurate and consistent. Despite being closed loop, the error for all the motion primitives tends to increase as the object is commanded to larger displacements. Our closed loop controllers have no knowledge that the system is non-holonomic and thus cannot reduce the pose error in directions that they do not directly control. Pose error in these directions may be accumulated due to errors in approximating roll and pitch or due to noisy object pose estimates. At the largest displacements, the error remains under 10% of the distance between start and desired end poses.

We notice that the pitch primitive is more variable when applied to the cylinder, as the surface curvature changes throughout the motion. The roll primitive performs worse on the sphere, further suggesting that the surface curvature of the object impacts the performance of the rotation primitives.

Next, we test the accuracy and consistency of a sequence of motion primitives. We command a sequence of: -1 cm Z -translation, $\pi/2$ pitch, $\pi/2$ roll, and $\pi/2$ pitch on the 50 mm cylinder. We run 10 trials and report the mean and standard deviation of the error after each step in Fig. 6.

As expected, each subsequent motion primitive tends to add error and increase the variance in the object pose. Interestingly,

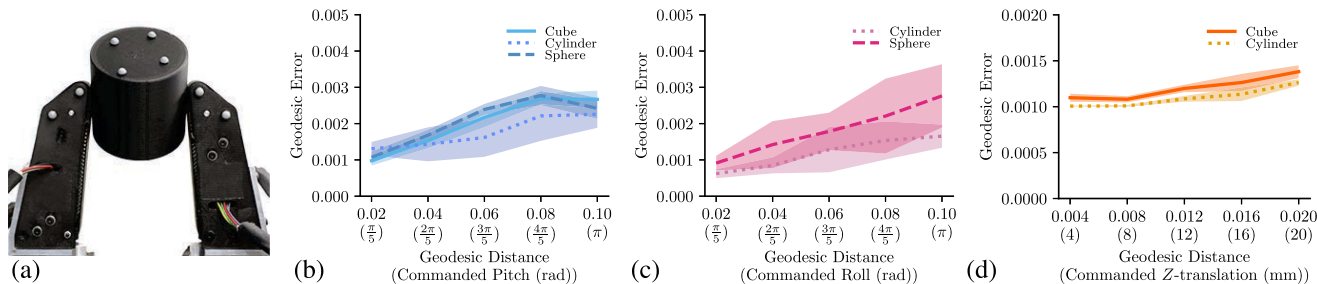


Fig. 5. (a) Accuracy of single motion primitives. For each experiment, a test object is placed in the gripper and a motion primitive is executed. (b)–(d) We report the error between the expected and end pose of the object across five trials for each motion primitive command and object using the distance metric described in Section III-D. The standard deviation is shown by the shaded region. An error of 0.001 is equivalent to a rotation error of 0.03 radians or a translation error of 1mm.

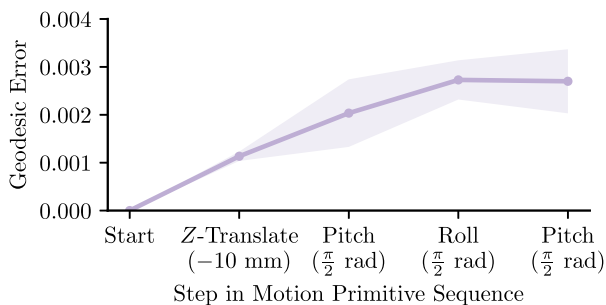


Fig. 6. Accuracy of a sequence of motion primitives. In this experiment, a test cylinder was placed in the gripper and a sequence of four motion primitives was performed. We report the error between the expected and end pose of the object after each motion primitive across 10 trials. The standard deviation is shown by the shaded region.

we see that with this object and motion primitive sequence, the roll primitive reduces the variance in the object pose by aligning the cylinder axis with the X axis of the gripper frame. This forces the pitch change to be 90 degrees. At the end of our 4-step sequence, we see that the average error is small, remaining under 0.003.

Overall, experimental results support that the motion primitives are consistently accurate and that we can reliably perform a sequence of motion primitives. This allows BOP to execute the multi-step plans generated by our grasp-to-grasp planner.

V. DEMONSTRATIONS

Finally, we show how a robot arm with BOP can complete a variety of dexterous manipulation tasks. We first use an existing task and motion planning (TAMP) framework to leverage BOP’s strengths of in-hand manipulation and force exertion to complete a real-world, multi-step manipulation task: screwing a light bulb into a socket. We then demonstrate how the active surfaces of BOP allow for various other manipulation capabilities beyond our three motion primitives. Please see the supplemental material for a video of the light bulb demo and the additional grasping capabilities.

A. Demonstration in a Multi-Step Manipulation Task

We demonstrate this gripper in context of a manipulation task: screwing a light bulb into a socket. As shown in Fig. 7, BOP is

mounted on a Franka Emika Panda and on a table there is a light bulb in a box and a light bulb socket. Since the light bulb is in the box with the threads facing upward, the only reachable grasps occlude the threads, preventing the light bulb from being screwed into the socket. Thus the robot will need to change its grasp on the bulb before inserting the bulb. Additionally, screwing in the light bulb requires many turns, which BOP can do by applying a continuous rotation.

To plan the sequence of actions needed to install the light bulb, we cast this as a TAMP problem. Solving a TAMP problem corresponds to selecting the sequence of actions for the robot to execute and selecting the discrete and continuous values, such as grasps, paths, objects, etc., that parameterize those actions [36]. We use the TAMP framework PDDLStream [11] and assume that we are given geometric models of the robot, the objects and the environment along with the poses of each object. We use a motion capture system to track the pose of the manipulated object, the light bulb, for the closed-loop control of BOP’s motion primitives.

Aside from a set of generic manipulator actions (`move`, `move_holding`, `pick`, `place`) we define two custom actions: `inhand_move` and `screw_in`. The `inhand_move` action uses the motion planner defined in Section III-D to plan in-hand grasp-to-grasp motion for a held object. The `screw_in` action first uses a guarded move to insert the bulb into the socket and then rotates the bulb by a fixed amount using the roll primitive.

Fig. 7 shows a plan of: `move`, `pick`, `inhand_move`, `move_holding`, `screw_in`, where the `inhand_move` and `screw_in` actions are emphasized. In this plan, the in-hand manipulation involves the Z -translation primitive followed by the pitch primitive. We believe this task demonstrates BOP’s ability to perform in-hand manipulations while simultaneously applying significant forces on the object.

B. Additional Manipulation Capabilities

Inspired by the manipulation task taxonomy from Bullock et al. [37], we demonstrate nail lifting, nonprehensile surface translation, and in-hand syringe actuation.

Nail lifting, shown in Fig. 8(a), uses a hand-scripted strategy to grasp a thin object. In this manipulation, the robot first grasps the object with a weak pinch grasp and then runs the belts on

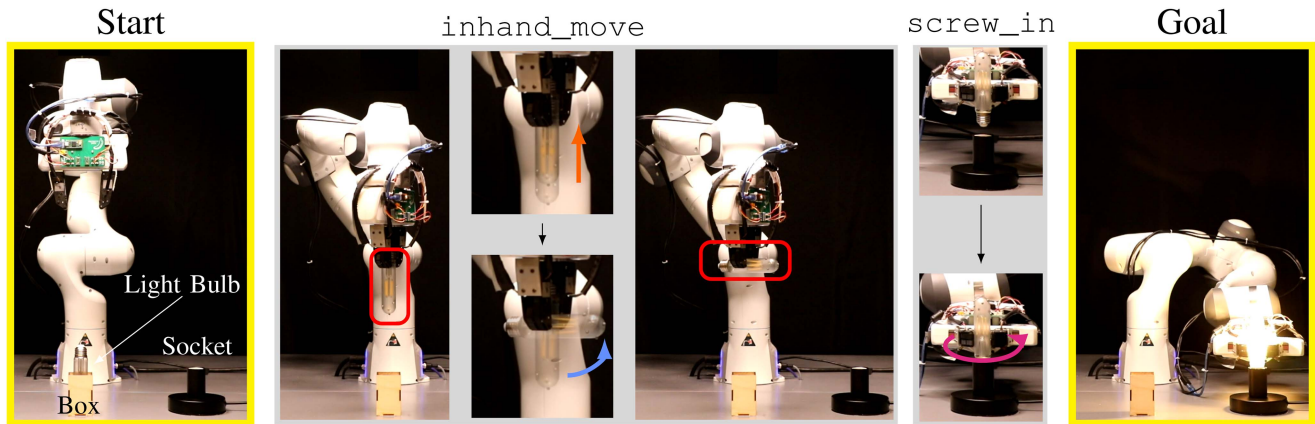


Fig. 7. We use a TAMP framework, augmented with actions enabled by BOP, to screw a light bulb into a socket. The box the light bulb starts in constrains the set of available grasps, forcing the robot to manipulate the light bulb in-hand. To complete the task, the robot executes a plan of `move`, `pick`, `inhand_move` (specifically, a z -translation, then a pitch rotation), `move_holding`, `screw_in`. Here we highlight the `inhand_move` and `screw_in` actions.

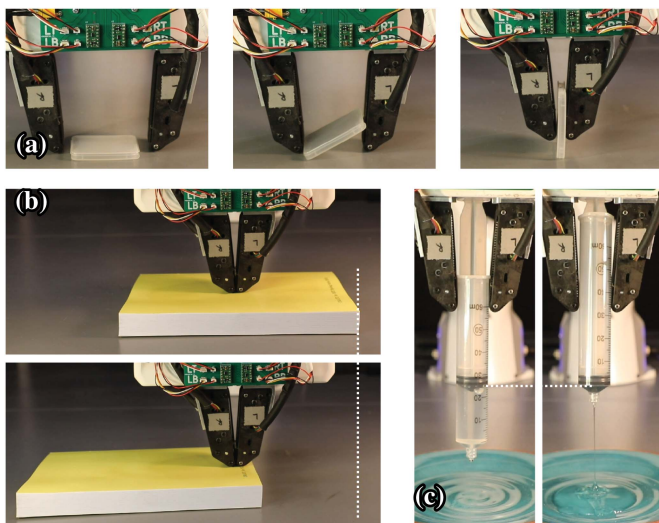


Fig. 8. BOP's design allow it to achieve unique motions such as (a) fingernail-style picking of a thin object off of a table, (b) scooting a book along a table, and (c) dispensing liquid from a syringe. The dotted grey line is used to emphasize the object movement.

the left finger in the direction of the palm, lifting that side of the object up. After a set time, the robot moves upward while continuing the close the gripper, allowing for a secure grasp on the object.

We test the effectiveness of this strategy on 3D printed squares with side lengths of 30 mm and thicknesses ranging from 0.5 mm to 4.0 mm in 0.5 mm increments, running 5 trials per thickness. A trial is successful if the robot is able to grasp the square on the larger sides and lift it from the table. The robot is able to successfully pick up all squares with a thickness greater than 2 mm. Failures on the thinner objects were caused by the fingers sliding over the top of the object or the object being jammed below the fingertips.

Nonprehensile surface translation, shown in Fig. 8(b), begins with a closed gripper and the robot applying a downward force on the surface of the object. Once the object is pinned, moving

the left finger belts towards the palm and the right finger belts away from the palm translates the book to the right. This strategy could be used to manipulate objects that are too thin or large to be grasped. The position of the end effector does not have to move to impart changes in the object's position. The success of this strategy depends on the frictional properties between the object and the belts, as well as between the object and the surface.

Syringe actuation, shown in Fig. 8(c), begins with the syringe grasped in-hand, with the plunger of the syringe in contact with the palm. Executing the Z -translation primitive moves the body of the syringe towards the palm while the palm prevents the plunger from moving, thus ejecting the material. This capability differs from the Z -translation primitive since there is palm-object contact, and shows how BOP can be used to manipulate articulated objects. Planning for this type of manipulation would require reasoning about how the object interacts with the entire gripper.

These examples serve as additional capabilities that could be sequenced in a multi-step manipulation task, in addition to the in-hand manipulation described in previous sections. Many more manipulation capabilities can be imagined, including the use of the Z -translation primitive to bring objects from a pinch grasp to the three-point enveloping power grasp described in Section III-B or the use of the roll primitive to impart a screwing motion on objects.

VI. DISCUSSION

In this work, we develop a gripper that is capable of a large range of in-hand motions while being able to exert significant forces on manipulated objects. By using belts to create multiple active surfaces within a single finger, BOP is able to avoid excessive hardware complexity while maintaining direct control over three of the grasped object's DoFs. This allows for the use of simple closed-loop controllers and planning algorithms to accomplish real-world tasks.

One notable limitation of BOP is that our proof-of-concept closed loop motion primitive controllers do not incorporate

geometric knowledge from the object model and fail when belts lose contact with the object. Future work could develop more robust controllers which reason about belt contact or use more compliant belt materials to help maintain contact with the object.

Overall, we believe that BOP is a significant advance towards reliably accomplishing multi-step, dexterous manipulation tasks in unstructured environments.

ACKNOWLEDGMENT

The authors would also like to thank Josh Pieper.

REFERENCES

- [1] Z. Li and S. S. Sastry, "Task-oriented optimal grasping by multifingered robot hands," *IEEE J. Robot. Automat.*, vol. 4, no. 1, pp. 32–44, Feb. 1988.
- [2] R. Haschke, J. J. Steil, I. Steuwer, and H. Ritter, "Task-oriented quality measures for dextrous grasping," in *Proc. IEEE Int. Symp. Comput. Intell. Robot. Automat.*, 2005, pp. 689–694.
- [3] Y. Lin and Y. Sun, "Grasp planning to maximize task coverage," *Int. J. Robot. Res.*, vol. 34, no. 9, pp. 1195–1210, 2015.
- [4] P. Tournassoud, T. Lozano-Pérez, and E. Mazer, "Regrasping," in *Proc. IEEE Int. Conf. Robot. Automat.*, 1987, pp. 1924–1928.
- [5] M. Mason, "Toward robotic manipulation," *Annu. Rev. Control Robot. Auton. Syst.*, vol. 1, no. 1, pp. 1–28, 2018.
- [6] M. Andrychowicz et al., "Learning dexterous in-hand manipulation," *Int. J. Robot. Res.*, vol. 39, no. 1, pp. 3–20, 2020.
- [7] T. Chen, J. Xu, and P. Agrawal, "A system for general in-hand object re-orientation," in *Proc. Conf. Robot Learn.*, 2022, pp. 297–307.
- [8] N. Chavan-Dafle et al., "Extrinsic dexterity: In-hand manipulation with external forces," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2014, pp. 1578–1585.
- [9] Y. Karayiannidis et al., "In-hand manipulation using gravity and controlled slip," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2015, pp. 5636–5641.
- [10] N. Ulrich, "Grasping with mechanical intelligence," Ph.D. thesis, Univ. Pennsylvania, Philadelphia, PA, USA, 1989.
- [11] C. R. Garrett, T. Lozano-Pérez, and L. P. Kaelbling, "PDDLStream: Integrating symbolic planners and blackbox samplers via optimistic adaptive planning," in *Proc. Int. Conf. Automat. Plan. Scheduling*, 2020, pp. 440–448.
- [12] A. Handa et al., "DeXtreme: Transfer of agile in-hand manipulation from simulation to reality," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2023, pp. 5977–5984.
- [13] A. Holladay, R. Paolini, and M. Mason, "A general framework for open-loop pivoting," in *Proc. Int. Conf. Robot. Automat.*, 2015, pp. 3675–3681.
- [14] N. Chavan-Dafle, R. Holladay, and A. Rodriguez, "Planar in-hand manipulation via motion cones," *Int. J. Robot. Res.*, vol. 39, no. 2/3, pp. 163–182, 2020.
- [15] K. Salisbury and B. Roth, "Kinematic and force analysis of articulated mechanical hands," *J. Mechanisms, Transmiss., Automat. Des.*, vol. 105, no. 1, pp. 35–41, 1983.
- [16] W. Bircher, A. Morgan, and A. Dollar, "Complex manipulation with a simple robotic hand through contact breaking and caging," *Sci. Robot.*, vol. 6, no. 54, 2021, Art. no. eabd2666.
- [17] R. R. Ma and A. M. Dollar, "An underactuated hand for efficient finger-gaiting-based dexterous manipulation," in *Proc. IEEE Int. Conf. Robot. Biomimetics*, 2014, pp. 2214–2219.
- [18] A. S. Morgan, K. Hang, B. Wen, K. Bekris, and A. M. Dollar, "Complex in-hand manipulation via compliance-enabled finger gaiting and multi-modal planning," *IEEE Robot. Automat. Lett.*, vol. 7, no. 2, pp. 4821–4828, Apr. 2022.
- [19] Q. Lu, N. Baron, A. Clark, and N. Rojas, "The RUTH gripper: Systematic object-invariant prehensile in-hand manipulation via reconfigurable underactuation," in *Proc. Robot. Sci. Syst.*, 2020, doi: [10.15607/RSS.2020.XVI.093](https://doi.org/10.15607/RSS.2020.XVI.093).
- [20] S. Abondance, C. B. Teeple, and R. J. Wood, "A dexterous soft robotic hand for delicate in-hand manipulation," *IEEE Robot. Automat. Lett.*, vol. 5, no. 4, pp. 5502–5509, Oct. 2020.
- [21] V. Tincani et al., "Velvet fingers: A dexterous gripper with active surfaces," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2012, pp. 1257–1263.
- [22] A. Kakogawa, H. Nishimura, and S. Ma, "Underactuated modular finger with pull-in mechanism for a robotic gripper," in *Proc. IEEE Int. Conf. Robot. Biomimetics*, 2016, pp. 556–561.
- [23] R. Ma and A. Dollar, "In-hand manipulation primitives for a minimal, underactuated gripper with active surfaces," in *Proc. Int. Des. Eng. Tech. Conf. Comput. Inf. Eng. Conf.*, 2016, Paper DETC2016-60354.
- [24] S. Zuo, J. Li, and M. Dong, "Design, modeling, and manipulability evaluation of a novel four-DOF parallel gripper for dexterous in-hand manipulation," *J. Mech. Sci. Technol.*, vol. 35, pp. 3145–3160, 2021.
- [25] Y. Cai and S. Yuan, "In-hand manipulation in power grasp: Design of an adaptive robot hand with active surfaces," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2023, pp. 10296–10302.
- [26] J. M. Gómez-de Gabriel and H. A. Wurdemann, "Adaptive underactuated finger with active rolling surface," *IEEE Robot. Automat. Lett.*, vol. 6, no. 4, pp. 8253–8260, Oct. 2021.
- [27] N. Rahman, L. Carbonari, M. DiMperio, C. Canali, D. G. Caldwell, and F. Cannella, "A dexterous gripper for in-hand manipulation," in *Proc. IEEE Int. Conf. Adv. Intell. Mechatronics*, 2016, pp. 377–382.
- [28] S. Yuan, L. Shao, C. L. Yako, A. Gruebele, and J. K. Salisbury, "Design and control of roller grasper V2 for in-hand manipulation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 9151–9158.
- [29] S. Yuan, A. D. Epps, J. B. Nowak, and K. K. Salisbury, "Design of a roller-based dexterous hand for object grasping and within-hand manipulation," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 8870–8876.
- [30] A. Bhatia, A. Johnson, and M. Mason, "Direct drive hands: Force-motion transparency in gripper design," in *Proc. Robot. Sci. Syst.*, 2019, doi: [10.15607/RSS.2019.XV.053](https://doi.org/10.15607/RSS.2019.XV.053).
- [31] K. Hauser and V. Ng-Thow-Hing, "Randomized multi-modal motion planning for a humanoid robot manipulation task," *Int. J. Robot. Res.*, vol. 30, no. 6, pp. 678–698, 2011.
- [32] S. LaValle, *Planning Algorithms*. Cambridge, U.K.: Cambridge Univ. Press, 2006.
- [33] J. J. Kuffner and S. M. LaValle, "RRT-connect: An efficient approach to single-query path planning," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2000, pp. 995–1001.
- [34] R. Tedrake, "Robotic manipulation," 2022. [Online]. Available: <http://manipulation.mit.edu>
- [35] B. Calli, A. Singh, A. Walsman, S. Srinivasa, P. Abbeel, and A. M. Dollar, "The YCB object and model set: Towards common benchmarks for manipulation research," in *Proc. IEEE Int. Conf. Adv. Robot.*, 2015, pp. 510–517.
- [36] C. R. Garrett et al., "Integrated task and motion planning," *Annu. Rev. Control Robot. Auton. Syst.*, vol. 4, no. 1, pp. 265–293, 2021.
- [37] I. M. Bullock, R. R. Ma, and A. M. Dollar, "A hand-centric classification of human and robot dexterous manipulation," *IEEE Trans. Haptics*, vol. 6, no. 2, pp. 129–144, Apr./Jun. 2013.