2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
Macau, China, November 4-8, 2019

A Data-Driven Framework for Learning Dexterous Manipulation of Unknown Objects

Andrew S. Morgan, *Student Member*, IEEE, Kaiyu Hang, *Member*, IEEE,
Walter G. Bircher, *Student Member*, IEEE, and Aaron M. Dollar, *Senior Member*, IEEE

***Abstract*— We address the problem of developing precision, quasi-static control strategies for fingertip manipulation in robot hands. In general, analytically specifying useful object transition maps, or hand-object Jacobians, for scenarios in which there is uncertainty in some key aspect of the hand-object system is difficult or impossible. This could be in scenarios with standard fully-actuated hands where, for instance, there is not an accurate model of the contact conditions, or in scenarios with fewer control inputs than mechanical degrees of freedom (such as underactuated hands or those that are controlled by synergies or impedance controlled frameworks), since the output space is of higher dimension than the input space. In this work, we develop a method for extracting object transition maps by tracking the state of the grasp frame. We begin by modeling a compliant, underactuated hand and its mechanical properties through an energy-based approach. From this energy model, we provide controlled actuation inputs to change the state of the grasp frame. We observe the response from these actions and develop a regression map of the action-reaction pairs, where the map is subject to our intent for grasp frame movement and the regional relationship between the contacts. Once the regression model is developed, we perform within-hand planning of the grasp frame with newly introduced objects. This approach is agnostic to the global geometry of the object and is able to adapt when undesirable contact conditions, such as sliding, occur. The learning-based methodology estimates the non-linearities representative in the properties of the system. We test our framework physically on an adapted Yale Openhand Model O. By transferring the learned model from simulation to the physical hand without adaptation, we show that this energy modeling approach is robust to inaccuracies in parameter estimation. We demonstrate its efficacy in a handwriting task.**

I. Introduction

Enabling robots to perform fine-grained manipulation tasks in unstructured, unmodeled environments is essential to successful implementations of future service robots. However, current planning and control for dexterous manipulation is faced with inherent challenges in these dynamic environments. Specifically, it is imperative that the robot estimates parameters for its model of the world [1]. Manipulation within such settings typically requires *a priori* knowledge of object, contact, and gripper models, or alternatively, expensive high-fidelity sensor suites for parameter estimation. Even so, these advanced sensing capabilities are not always enough for the task at hand. Tradeoffs are introduced in soft, compliant, and underactuated hands that passively adapt to the environment, which enables them to easily grasp objects. Though, encoders are typically not installed at each of the joints and tactile sensing is often unavailable. Moreover, their state is not fully controllable via the available actuation inputs, making the development of traditional object transition maps, which represent the function from input actuator velocities to output object velocity, a strenuous task.

Research supported by the U.S. National Science Foundation under grant IIS-1734190.

A. S. Morgan, K. Hang, W. G. Bircher, and A. M. Dollar are with the Dept. of Mech. Eng. and Mater. Science, Yale University, CT, USA, ({Andrew.Morgan; Kaiyu.Hang; Walter.Bircher; Aaron.Dollar}@yale.edu).

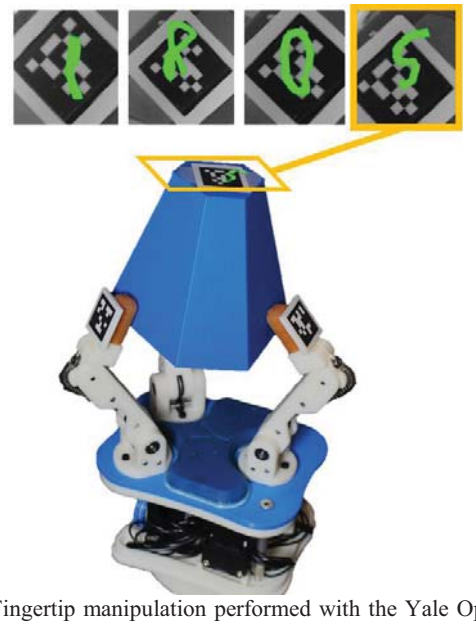


Fig. 1. Fingertip manipulation performed with the Yale Openhand Model O to write out the letters, 'I', 'R', 'O', 'S' with the tip of an object, after being trained from 29,000 path tracing experiences.

This quest for dexterous capabilities has introduced various levels of modeling in the field of grasping and manipulation, investigating topics ranging from contact and fingertip models [2] to stability measures of the whole hand-object system [3]. These mechanics-inspired models have aided in prediction of how contacts, fingertips, and objects react in given conditions [4]. This underlying understanding has led to additional works in grasp planning [5], [6], fingerpad geometry optimization [7], finger gaiting [8], [9], and object stability analyses [10]. Nevertheless, manipulation remains difficult for tractable systems in real-world applications.

Traditional approaches to grasping and manipulation have been through the use of high-dimensional, fully-actuated hands often augmented with tactile sensing capabilities. While these systems enable users to control each actuator individually, their high degree of dimensionality becomes difficult to control post-contact, since the closed kinematic chain risks overconstraint [11]. During overconstraint, linearly coupled actuation makes precise object manipulation difficult.

An advantageous approach to manipulation has been through the use of soft, compliant, and underactuated hands, which passively adapt to the geometries of the object [12], [13]. Numerous works have shown their efficacy in grasping tasks, even presenting their utility in the difficult pinch and flip manipulation [14]. Although their inherent compliance of such hands is beneficial for grasping, manipulation remains difficult since the closed mechanism has more degrees of freedom than degrees of actuation. In [15], it was shown that precision dexterous manipulation is possible with underactuated hands given quasi-static assumptions in elastic perturbations and when holonomic contact configurations can be guaranteed. Through this notion, our previous works have addressed manipulation with underactuated hands in the planar case by simplifying quasi-static control inputs through the formulation of Precision Manipulation Primitives (PMPs) represented by a signed binary Jacobian. Though, this simplified approach presented its shortcomings in object motion accuracy, and required additional control modeling to increase manipulation resolution [16]. Following work developed a state transition model for object movement based on the PMPs, and was shown to increase accuracy in the planar manipulation case [17]. Nonetheless, most previous work has addressed underactuated manipulation in 2D, where only one other work in the spatial domain has been completed to the best of the authors' knowledge [18].

This work builds on the observation that precision dexterous capabilities with underactuated, compliant, or soft hands is possible but likely follows in the form of a non-linear map from actuation input velocities to object velocity. Through leveraging their inherent ability to easily acquire and maintain a stable grasp, even with parametric uncertainties or undesirable external perturbations, we model the system as a parallel mechanism in fingertip manipulation [19]. Its configuration is then obtained by solving for the minimum energy configuration of the hand-object system, while maintaining a stable grasp frame $\in SE(3)$.

Training data is acquired by observing the movement of the grasp frame after random actuation, and doing so over many regional contact relationships. The grasp frame is comprised of exactly three contacts. By varying the distances between the three contacts with respect to one another, while also honoring the kinematic constraints of the hand, we can represent a fluid representation of varied object geometries. Ultimately, this framework represents an approach that is agnostic to the global geometry of the object, allowing for object generalization. Additionally, it allows the system to recover when undesirable events occur at the contacts, e.g. slip or rolling, since the regression framework is trained with various relationships between the contacts.

We train a Random Forests Regressor (RFR) subject to object velocity intent (Cartesian velocity reference). This learned model is later used for predicting actuation inputs required to reach a desired object pose. In order to control the movement of newly introduced geometries, we define a point on the object we intend to control, *Point of Manipulation,* (POM), and compute the rigid body transformation from the grasp frame to create a plan. Once computing a Cartesian velocity reference, we control our grasp frame, consequently the POM, through a closed-loop approach by continuously querying our RFR model until completion (Fig. 1).

To the best of our knowledge, this is the first work that learns object transition maps in a data-driven framework for spatial, underactuated dexterous manipulation. The rest of this paper is organized as follows: Sec. II presents the energy model used to describe the hands, Sec. III presents the framework and the control algorithm, Sec. IV presents data collection, Sec. V. analyzes the experimentation results, and, Sec. VI concludes this paper with a future work discussion.

II. Object Motion Models

In this section, we will discuss the underlying problems in formulating the hand-object Jacobian for underactuated hands. We will follow by presenting the formulation of the grasp frame and the energy model used for this work to estimate motions of an underactuated system.

A. The Hand-Object Jacobian

In traditional, fully-actuated manipulators, there exist models that fully describe how an object should respond to actuation inputs during fingertip manipulation. These models typically come in the form of forward and inverse kinematic relationships that can be calculated analytically given *a priori* knowledge of the system. Depending on the system's input dimensionality, these models can become non-trivial to derive. Additionally, accuracy of the manipulation is subject to estimation in physical contact parameters. Unfortunately for underactuated hands, developing an analytic map from actuation inputs to object movements is still an unsolved problem, since reconfiguration of the hand is solved with respect to forces applied by the hand on the object. Estimating the reconfiguration of the finger at the point of contact requires knowledge about the contact force specifically, such as location and direction, which is not always known when the hand is not equipped with tactile sensing capabilities.

Without the use of force sensing, finding the analytic input-output relationship for an underactuated system is non-trivial since input dimensionality is less than that of the constrained degrees of freedom of the mechanical system. For a manipulator comprised of k serial-link fingers, each having j_i, joints per finger, the hand or joint configuration, $q \in \mathbb{R}^{\sum_{i=1}^{k} j_i}$, fully defines the state of the system. In traditional grasping and manipulation modeling, the manipulator Jacobian, or similarly noted as the hand Jacobian, $J_h(q)$, represents the map from actuator velocities to fingertip velocities. We will suppress the q dependence for the rest of this paper. In fact, $J_h = blkdiag(J_1, \dots, J_i)$, i.e. the block diagonal matrix of Jacobians for each finger in the hand. Similar to the hand configuration, a manipulator can be represented by its actuator positions, a. A system is fully actuated if $dim(a) = dim(q)$ and underactuated if $dim(a) < dim(q)$. For soft robotic systems, or in general for systems with fewer control inputs than degrees of freedom, the overall controllability of the system decreases as the difference between $dim(q)$ and $dim(a)$ increases. We can represent the spatial representation of the hand Jacobian as:

$$\dot{x} = J_h \dot{q} \tag{1}$$

where fingertip velocities $\dot{x} \in \mathbb{R}^{3k}$, $J_h \in \mathbb{R}^{3k \times q}$, and joint velocities $\dot{q} \in \mathbb{R}^{q}$. Similarly, the Grasp Matrix, G, in the velocity domain represents the map from external contact velocities to object frame velocity, $v \in se(3)$. This

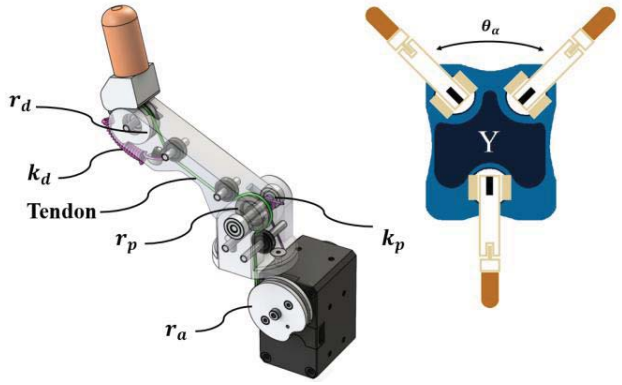


Fig. 2. A single motor controls the actuation of a single finger about its proximal and distal joints via a tendon. Abduction between fingers is kept constant in this work.

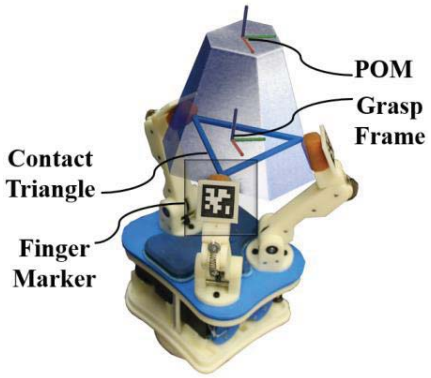


Fig. 3. The grasp frame is comprised of exactly three points between the hand and the object, represented by the frame in the center of the blue contact triangle constraint. The *Point of* Manipulation (POM) is then a rigid body transformation from the grasp frame as to allow for object manipulation planning.

representation, assuming a single point contact on the tip of each finger, follows the form:

$$\dot{x} = G^T v \tag{2}$$

In the spatial case, $G \in \mathbb{R}^{6\times b}$, where $b = \sum_i rank(B_i)$. For this notation, B_i is the basis contact model for each contact on the i^{th} finger. According to the fundamental grasping constraint [20], in order to maintain a stable grasp on the object without sliding, contact velocities of the manipulator must be equal to contact velocities on the object. This further eludes to the hand-object Jacobian, H, which is a direct map from actuator velocities to object velocity. This formulation assumes a point contact with friction model, enabling the finger to apply a force in x, y, and z^+ directions. This requires $B_i \in \mathbb{R}^{6\times 3}, \forall i \in \{1, \dots, k\}$ and $(G^T)^+$ to be solvable as the pseudo-inverse of the Grasp Matrix transposed:

$$v = (G^T)^+ J_h \dot{q} = H\dot{q} \tag{3}$$

Concretely, in order to solve for the hand-object Jacobian for underactuation, we want to find a transformation matrix, T, which maps H and input actuation velocities, $\dot{a}$, to object frame velocity, v:

$$v = HT\dot{a} \tag{4}$$

where, $T\dot{a} = \dot{q}$ and $T \in \mathbb{R}^{q\times a}$. If T can be represented symbolically and H can be formed analytically, the object can be manipulated as desired by solving for the representative actuator velocities to control the system.

In serial-link, tendon-driven underactuated mechanisms, a tendon constraint dictates the relationship between subsequent joints controlled by the same actuator. Evaluating a single i^{th} finger in the two-link case and by assuming the routed tendon is inextensible:

$$r_a \dot{a} = r_p \dot{q}_p + r_d \dot{q}_d \tag{5}$$

where r_a, r_p. and r_d are the radii of the actuator, proximal, and distal pulleys and $\dot{a}, \dot{q}_p$, and $\dot{q}_d$ are the velocities about the same joints, respectively (Fig. 2). Similarly, previous texts have represented this constraint in the form of an actuator Jacobian, J_a [15]:

$$J_a = \begin{bmatrix} \frac{r_p}{r_a} & \frac{r_d}{r_a} \end{bmatrix}, \quad \dot{a} = J_a \begin{bmatrix} \dot{q}_p \\ \dot{q}_d \end{bmatrix} \tag{6}$$

Though, through this single velocity constraint and due to the coupling of the joints, $\dot{q}_p$ and $\dot{q}_d$ are linearly dependent, presenting our inability to solve for matrix T symbolically. Therefore, without any additional constraints added to the system, in this work, we want to learn a representation of HT so that we can map $\dot{a}$ to v.

B. *The Grasp Frame*

Given a stable grasp, any point on an object can be represented by a transformation from the object frame to any desired point on the rigid body. In this work, we denote the object frame as the grasp frame, which is adapted from [21]. The grasp frame formulates a standardized representation for object movement constructed by three contacts. While assuming a stable, point contact grasp, the grasp frame uniquely describes the motion of any point on the object during manipulation. Let us define contact points between the object and finger tip $P = p_1, \dots, p_i$ where $p_i \in \mathbb{R}^3, \forall i \in \{1, \dots, k\}$ and where any three points in P can define the grasp frame. Additional contacts can be added, but the relative positions between three uniquely defines a grasp constraint, i.e. additional contacts $\{p_4, \dots, p_N\} \subset P$ are redundant if all remain fixed to the object. We will define $\mathcal{X}$ to be the pose of the grasp frame, defined by:

$$\begin{aligned} \mathcal{X} &= [g_x, g_y, g_z \,|g_o] \in SE(3) \\ g_o &= \frac{1}{3}(p_1 + p_2 + p_3) \\ g_x &= \frac{p_2 - p_1}{\|p_2 - p_1\|} \\ g_z &= \frac{(p_3 - p_2) \times g_x}{\|(p_3 - p_2) \times g_x\|} \\ g_y &= g_z \times g_x \end{aligned} \tag{7}$$

In this notation, g_x, g_y, and g_z are the directional vectors of the x, y, and z axes, respectively, about an origin g_o, all with respect to the world frame. Given $\mathcal{X}$, any desirable *Point of Manipulation* (POM) which represents a frame, $M \in SE(3)$, can be a point represented in space, $p_m \in \mathbb{R}^3$, by a transformation from $\mathcal{X}$. This transformation assumes the

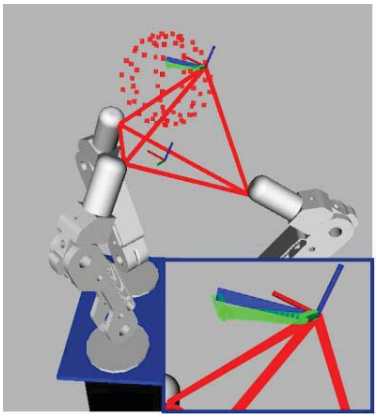


Fig. 4. Energy model simulation for manipulating the grasp frame. For data collection, the grasp frame was randomly manipulated to acquire the object transition map, while varying contact relationships between the fingers. Final evaluation of the map was conducted by querying random points within the POM's potential workspace (red sphere) and executing the path. (Bottom) We note that the POM path (green) executed deviates slightly from the projected path (blue) at the beginning but converges to the end location.

object is a rigid body, in addition to the contact triangle constraint $const(\mathcal{T})$ where $\mathcal{T} \in \mathbb{R}^3$. This constraint signifies that the distances between contacts formulating the grasp frame remain constant during manipulation, i.e. a point contact is maintained and no sliding or rolling is observed on the object. The relationship between contacts is, $\mathcal{T} = \{dist(p_1, p_2), dist(p_2, p_3), dist(p_3, p_1)\}$ where $dist(\cdot)$ is Euclidean distance. This contact triangle constraint is used later in the regression framework (Sec. III).

C. *The Manipulation Energy Model*

Underactuated mechanisms that leverage springs for passive adaptability can be modeled in accordance to energy. Each finger is modeled as a serial-link chain of rigid bodies and springs. The tendon force supplied by the actuator is counteracted by the force of the contacts and the return force of the springs, and is typically dampened by friction in the tendon's transmission. Leveraging the simplicity of acquiring and maintaining a stable grasp with underactuated hands, we can form a simple parallel mechanism with springs model of the system. While this formulation appears to neglect contact force at the fingertips, this requirement is maintained from the triangle constraint in Sec. IIB. We also assume that friction in the tendon's transmission is negligible.

As a parallel mechanism, the actuation of a single link, or finger, corresponds to a change in pose for $\mathcal{X}$ and can be solved through energy balancing. We expect the hand-object system to maintain energy equilibrium between the contacts $\{p_1, p_2, p_3\}$ and the grasp frame after movement. In fact, while neglecting friction about the joints, the system will equilibrate to the minimum energy configuration subject to Eq. (5) after an actuation input or external disturbance. The energy, E^i, within each finger is, of course, configuration dependent. For the two-link case, where $\theta^i = \{\theta_{pi}, \theta_{di}\}$, or the joint configuration of the proximal and distal joint, respectively, the energy of the finger can be represented as:

$$E^i(\theta^i) = \frac{1}{2}\left(k_p\theta_{pi}^2 + k_d\theta_{di}^2\right) \tag{8}$$

where k_p and k_d are the spring constants for the proximal joint and the distal joint, respectively (Fig. 2). Through this constraint, we can solve for the joint configuration of the hand by minimizing the elastic energy between all of the fingers. Assuming our grasp constraint is still valid, and that a single point contact at the fingertips is maintained, the integrity of the grasp frame should still hold. Therefore, the equilibrated configuration of the hand, q^*, can be found through:

$$q^* = \arg\min_q \sum_i E^i(\theta^i) \quad s.t.\ Eq.(5) \tag{9}$$

III. Learning the Object Transition Map

The proposed methodology in this work fundamentally estimates the pseudo-inverse of the product of the hand-object Jacobian, H, and the transformation matrix, T, namely $(HT)^+$, creating a map from desired object movement (Cartesian velocity reference, $\dot{\mathcal{X}}$) to actuation velocities, $\dot{a}$. In the traditional system estimation problem, we observe $\dot{\mathcal{X}}$ given $\dot{a}$ and formulate a forward map of how a system responds to an input. Conversely, in the control problem, we want to control $\dot{\mathcal{X}}$ through $\dot{a}$, which clarifies the formulation of the inverse function. In the simplified understanding of this problem, we estimate what actuation inputs result in desirable movement of the contacts in order to manipulate the grasp frame.

As in Sec. IIA, analytically deriving this transition map for soft or underactuated mechanisms is infeasible if solely using the velocity constraints previously described. It may be possible that additional constraints are justified for specific mechanisms, but this is not the case in general. By creating a representative model of the system, we are able to estimate the input-output relationship, or the action-reaction pairs, by performing an action and observing the reaction. This reaction is likely a non-linear relationship between contact movement and actuation inputs. We find that the proposed energy model is tolerant to uncertainties in physical parametric estimation, allowing us to easily transfer the regression model learned from simulation to the physical hand. While this model serves our purpose well, other input-output models would suffice depending on the mechanism used and the task.

The desired transition map is learned through the presented energy minimization approach. Holistically, actuation inputs to the hand, $\dot{a}$ result in a Cartesian velocity movement of the grasp frame, $\dot{\mathcal{X}}$, displacing the grasp frame to a new pose $\mathcal{X}$. While maintaining a constant relationship for the triangle constraint $\mathcal{T}$ which ensures a point contact, the model is actuated randomly throughout the entire workspace to evaluate the relationship between $\mathcal{X}$, $\dot{\mathcal{X}}$, and $\dot{a}$. This presents the forward system estimation problem, where $(\mathcal{X}, \dot{a}, \mathcal{T}) \rightarrow \dot{\mathcal{X}}$. By collecting this reaction, $\dot{\mathcal{X}}$, from an action, $\dot{a}$, we can form the inverse of this function for the control problem.

Once a single relationship in $\mathcal{T}$ is exhausted, i.e. $\mathcal{X}$ has reached all points in the workspace for the given triangle constraint, we can randomly re-initialize q. By doing so, this specifies a new relationship in $\mathcal{T}$ for the next iteration, that fundamentally represents a new hand-object configuration with a dissimilar object geometry. The configuration is

validated for force closure and the object is then manipulated randomly once again according to the energy model. Through this initialization approach, the learned map observes a fluid representation of relational object geometries, increasing object generality. Additionally, as aforementioned, this formulation is beneficial in that it neglects global object geometry. This not only allows the learned map to generalize over different objects, but also to adapt when undesirable contact scenarios occur. Specifically, we seek to learn the function, $h(\cdot)$, with our non-linear regressor:

$$h: \left(\mathcal{X}, \dot{\mathcal{X}}, \mathcal{T}\right) \rightarrow \dot{a}$$

In the system estimation problem, the Cartesian velocity reference, $\dot{\mathcal{X}}$, represents a reaction given actuation. In the control problem, it now serves as an abstraction for intended object movement. Given $\mathcal{X}$ and the intended goal of the grasp frame, $\mathcal{X}_g$, we can calculate the vector representing the linear transition between the two, and scale it according to our desired object transition velocity, calculating $\dot{\mathcal{X}}$. Naturally, there is a trade-off between speed and stability during manipulation, due to our quasi-static assumptions, influencing the length of the vector $\dot{\mathcal{X}}$ used in execution. Now, the learned function $h(\cdot)$ estimates the actuation velocity, $\dot{a}$, required to receive the desired system reaction.

Algorithm 1 POM Control

Input: $M_g, d, iter$
Output: ϵ ▷ waypoint offset

$\mathcal{T} \leftarrow Camera.calcT()$
$M \leftarrow Camera.readPOM()$
$t \leftarrow calcTrans(\mathcal{T}, M)$ ▷ grasp frame to POM
$\mathcal{X}_g \leftarrow invTrans(M_g, t)$ ▷ POM to $\mathcal{X}$ coord
for $i = 1$ to $iter$ **do**
  $\mathcal{X} \leftarrow invTrans(M, t)$
  $\dot{\mathcal{X}} \leftarrow carVel(\mathcal{X}, \mathcal{X}_g)$ ▷ compute cart. vel. ref.
  $\dot{a} \leftarrow h(\mathcal{X}, \dot{\mathcal{X}}, \mathcal{T})$ ▷ find actuator velocity
  $Hand.execute(\dot{a})$ ▷ move hand
  $\mathcal{T} \leftarrow Camera.calcT()$
  $M \leftarrow Camera.readPOM()$
  $t \leftarrow calcTrans(\mathcal{T}, M)$
  $\epsilon \leftarrow dist(M, M_g)$
  if $\epsilon < d$ **then** ▷ solution found
    break
return ϵ

Specifically, for the handwriting task, we seek to control a given point on the object that is offset from the grasp frame. Once the object transition map is created, controlling the POM pose, M, is achieved by controlling the pose of the grasp frame. The algorithm for control is summarized in Alg. 1. Inputs include the goal position of the POM, M_g, minimum distance threshold from M to M_g signifying completion, d, and the number of iterations allotted to complete the task, $iter$. The algorithm begins by computing the transformation, t, from the grasp frame into the POM. This requires knowing $\mathcal{T}$, which is calculated in $calcT(\cdot)$ by either extracting location from markers on the hand or from the simulation environment. The $invTrans(\cdot)$ method then converts coordinates from the POM into $\mathcal{X}$ given t to solve for the final grasp frame configuration of the hand, $\mathcal{X}_g$. The Cartesian velocity reference is then computed in $carVel(\cdot)$ by taking the difference between $\mathcal{X}$ and $\mathcal{X}_g$ then scaling appropriately for the task. Actuation velocities are then estimated in the learned function, $h(\cdot)$, and executed on the hand. The loop breaks once M has reached a distance within the threshold, d, to its desired goal position or exceeds the number of loop iterations. Waypoints are completed iteratively, by changing the goal position of the POM to a new waypoint upon completion.

IV. Data Collection

A. The Model O

An adapted Model O from the Yale Openhand Project [22] was used for this work. Physical modifications include a soft, rounded fingertip (durometer 30), pulleys throughout the finger to reduce friction in the tendon's transmission, and bearings within each of the joints. As a tendon-driven mechanism, each finger is powered by a single actuator (Dynamixel XM430-W350-R) in position control mode. The hand is comprised of three fingers, each of two links. An additional motor is attached to enable ab/adduction for the fingers, which is set to a fixed position of $\theta_a = 90°$ and is not used in this work (Fig. 2,3,4). Return forces for each finger are supplied by a torsional spring about the proximal joint, and an extension spring about the distal joint. The Model O is not symmetric about the z axis of the gripper, i.e. the locations of the proximal joints do not form an equilateral triangle, further adding geometric complexities to the manipulation problem.

B. Collection and Training

Data collection was achieved through the use of the aforementioned energy minimization approach (Sequential Least Squares Programming Optimization). We began by constructing a representative Model O in simulation with estimated pulley radii, spring constants, and link lengths. Due to the energy model's robustness to parametric uncertainties, general estimation of these values did suffice. We then solved for the initial configuration of the hand given three contact points in space. This defined the values static in $\mathcal{T}$. Once a valid configuration was found, we applied a random actuation set to the fingers and recorded the movement (Fig. 4).

We observed the original location of the grasp frame as $\mathcal{X}$. Given this random actuation, $\dot{a}$, we then observed the change, $\dot{\mathcal{X}}$ for a single actuation step. Evaluating these action-reaction pairs, we formed a feature set, $\mathcal{S}$, and regression set, $\mathcal{R}$, describing the state of the system and its motion, while maintaining integrity on $\mathcal{T}$. Denoted by $s_n = \left(\mathcal{X}_n, \dot{\mathcal{X}}_n, \mathcal{T}_n\right)$ an input feature, and by $\dot{a}_n$, an output feature, the training dataset is defined as:

$$\mathcal{S} = \{s_n\}_{n=1:N}, \quad \mathcal{R} = \{\dot{a}_n\}_{n=1:N}$$

A total of 29,000 action-reaction pairs were collected in simulation over the course of 50mins for $\mathcal{S}$ and $\mathcal{R}$, representing 500 different relationships in $\mathcal{T}$. For each representation in $\mathcal{T}$, a virtual object radius, i.e. a circle fit to intersect all three contact points, was computed and was restricted to be within 3.5-5.5cm. To maintain a tractable amount of data, every possible $\mathcal{X}$ with respect to $\mathcal{T}$ that is possible according to the gripper's workspace was not computed. This would merely result in a very large dataset and would likely be redundant for similar relationships in $\mathcal{T}$, e.g. when a single contact segment slightly changes. It was assured that for similar relationships in $\mathcal{T}$, a

Letter	*A*	*B*	*C*	*D*	*E*
Waypoints	7	7	4	5	7
Time (s)	24.2	21.5	9.3	28.9	20.6
PTP Distance	3.92cm	4.42cm	1.91cm	3.68cm	2.85cm
Dist. Traveled	4.60cm	5.31cm	2.13cm	4.72cm	3.10cm
Waypoint Err.	1.01±0.33mm	0.85±0.24mm	0.85±0.32mm	1.6±0.38mm	1.5±0.2mm

Fig. 5. Physical experiment tracking the point of manipulation of the object in Figure 1. Letters A-E are written according to their respective waypoints in a top-down view (start: square, connecting: circle, finish: star). The progression of waypoints, depicted in letter B, represents the intended linear manipulation path for the object. The number of waypoints, time, expected point-to-point (PTP) distance, actual distance traveled, and waypoint error is recorded. Letters with abrupt transitions in Cartesian velocity references deviate slightly off the path.

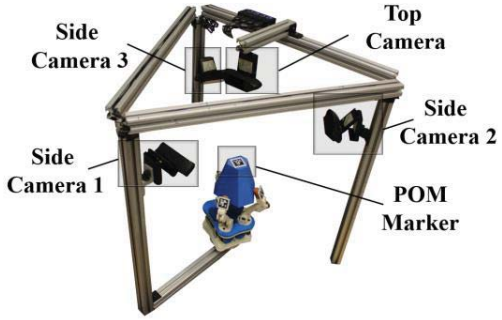


Fig. 6. Four-camera setup for tracking state of the contacts and the grasp frame.

representative spread of the workspace was realized through a nearest-neighbors approach. A Random Forests Regressor (RFR) [23] consisting of 200 trees at a depth of 30 was trained using 5-fold cross validation, with an average R^2 score of 0.965 for all three regressed outputs. This regressed model, composed from data solely collected in simulation, will serve as the basis for our execution in the following section.

V. Experiments

To show the efficacy of the proposed RFR model, we evaluated it both in simulation and on a physical hand using Alg. 1. For the physical evaluations, a state detection system (Fig. 6) was created to track the location of the contacts and pose of the POM via 4 overhead cameras during manipulation. AprilTags provided a pose $\in SE(3)$ for each finger and for the POM. Contact positions to construct $\mathcal{T}$ were extracted by calculating a transformation from the AprilTag on the back of the finger (see Finger Marker, Fig. 3) to the point of contact.

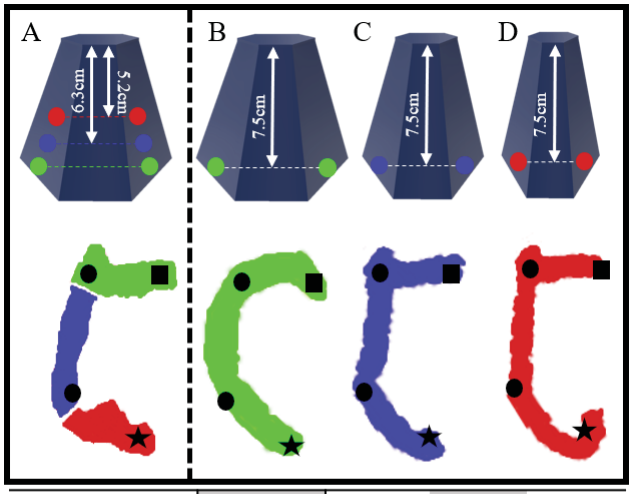
A. Waypoint Evaluation

The hand was initialized to five different starting configurations in the energy-based simulation, representing five different object geometries. The virtual object radius was ensured to be within that of the training data. The POM was translated to 5cm above the grasp frame and with the same rotation as $\mathcal{X}$. Once initialized for each object, the hand was commanded to move between 500 randomly selected waypoints within a sphere of radius 2.5cm (Fig. 4). During random waypoint selection, it was ensured that each waypoint was unique, but it was not guaranteed that the waypoint was within the reachable workspace of the learned model or that of the hand-object system. Once executed, if the previous waypoint was completed, the system started from that location. Conversely, if the POM did not reach its intended waypoint, the system was reset before continuation to the next waypoint.

The trajectory of the POM is tracked through the manipulation. We note that in the example, Fig. 4, the path of the POM deviates slightly at the beginning of the movement before finally converging within 1mm of the intended waypoint. This response appeared often for the completed trajectories. Waypoints that were tagged as incomplete reached within 5mm of the desired goal location 73% of the time, and could not move closer without violating the contact constraint. While we seek to have a fully connected, fluid workspace for all object geometries, this is not always feasible given physical system constraints. Increased waypoint completion is likely to be realized if all waypoint goals were suited within the workspace of the hand-object system, which we did not guarantee. Results are summarized in Table 1.

TABLE I. Evaluation of 500 Waypoints

Object	*Seg1 (cm)*	*Seg2 (cm)*	*Seg3 (cm)*	*Radius (cm)*	*Completed/ Traveled(cm) (100 iter.)*	*Completed/ Traveled(cm) (300 iter.)*
1	9.1	6.1	9.1	4.87	211/1.7	334/2.0
2	6.7	4.2	6.7	3.53	318/1.8	441/2.0
3	7.4	4.8	7.8	4.03	243/1.8	397/2.1
4	7.3	5.2	7.3	3.93	358/1.9	499/2.4
5	7.7	4.7	6.7	3.90	297/1.9	417/2.2



	A	*B*	*C*	*D*
Obj	Sm/Med/Lg	Lg	Med	Sm
Radius (cm)	4.51/4.05/3.52	4.51	4.05	3.52
Dist. Traveled (cm)	2.62	2.13	2.42	2.53
Waypoint Err. (mm)	0.92±0.38	0.85±0.32	1.1±0.42	1.3±0.33

Fig. 7. Evaluation of writing the letter 'C' when changing contact location on a single object (A) and varying radii on different objects (B-D). Dots on the object signify contact points. The two objects are designed such that they have the same radii as observed during the contact changes in (A) (4.05cm and 3.52cm, respectively). When executing (A), the system must adjust to the rigid body transformation change from the POM to reach the desired waypoints in a top-down view.

Five different objects were tested with virtual radii ranging from 3.5cm to 4.9cm. The segment lengths constructing the relationship in $\mathcal{T} = \{Seg1, Seg2, Seg3\}$ were selected to represent varying object geometries. For each object, we record the speed in which the object transitioned to the desired goal configuration and if the waypoint was completed. As provided in Table 1, time was captured by evaluating the number of iterations required to reach each waypoint, 100 iterations (3.3 seconds) and 300 iterations (10 seconds). As the number of iterations increases, the average distance traveled and the number of waypoints completed also increases. We found that the best virtual object radius for waypoint completion was around 4 cm with equal segment lengths for $Seg1$ and $Seg3$ of the contact triangle. We observed during experimentation that as the object radius increased, the workspace of the POM tended to decrease. Intuitively, this corresponds to the limited workspace of the finger post-contact with a growing object radius. The observed phenomena did not affect the average length of completed waypoints.

B. *Handwriting Task*

The learned model was evaluated physically by analyzing its performance in a modified handwriting task. For this execution, the learned model was transferred directly from simulation to the physical environment. This experiment was performed with an irregular-shaped, tapered object as to allow slippage and recovery (Fig. 1, 5) of virtual radius 4.51cm ($\mathcal{T}$ ={7.5cm, 8.0cm, 7.8cm}, 41g). The task was to write the first five letters of the English alphabet in capital form. Waypoints were created according to the rigid body transformation from $\mathcal{X}$ to the POM, which was tracked according to the AprilTag attached to the top of the object (7.5cm above the grasp frame). Once transformed, we queried the RFR model in $\mathcal{X}$ coordinates to find the actuation inputs required for movement, as in Alg. 1. The center of the object AprilTag was tracked during the manipulation and performed the 'writing' of the letters, i.e. the traced letters of the alphabet were written according to the center of the tag. Once a single waypoint was reached within a threshold of 2mm, or the maximum number of 300 iterations (10 seconds) was exceeded, the Cartesian velocity reference was updated to direct the system to the next waypoint. In physical experimentation, the waypoints were always reached within the allotted iteration limit.

As depicted in Fig. 5, the model's execution resulted in discernible English letters in capital form (letters A-E are evaluated). It is noted that in some cases the POM did deviate slightly from the expected point-to-point path signified by the waypoints (Fig. 5, Letter B). We believe this response to be an artifact of the current configuration of q , where, as learned from the simulated energy model, there are instances where the hand has difficulty changing directions abruptly while maintaining integrity on $\mathcal{T}$. This phenomenon tends to occur when the system realizes that the force application direction of the finger would be insufficient to maintain force closure, allowing the contacts to slip. Thus, the grasp frame must move slightly away from its desired linear motion in order to set up the approach for the next waypoint. This phenomenon is most clearly depicted in the writing of letters, 'B' and 'D'. In these executions, the POM traveled approximately an additional centimeter during writing.

During this experiment, we also noted the waypoint error realized during execution. Depending on the grasp frame location within the workspace, POM movement generally resulted in Cartesian steps from 0.5 to 1.4mm as observed by the overhead camera. As in Alg. 1, the object was manipulated until the POM reached within $d = 2$mm of the intended waypoint. The waypoint error was calculated individually for each letter. As noted in Figure 5, we did not see a direct correlation between the waypoint error and the additional distance traveled between waypoints. We again attribute the increase in error to the current configuration of q during manipulation. Given these results, we believe it is possible to further increase waypoint resolution by appropriately scaling the Cartesian velocity reference as the POM approaches the waypoint. In all cases, the POM was able to reach the waypoint within the set threshold and within the allotted time.

C. *Adaptability to Contact Changes and Varied Radii*

A final evaluation was completed with the same physical setup as in the handwriting experiment. In this execution, we desired to write the letter 'C' with the POM of an object. Using the same object as previously, which we will denote as the large object, we evaluated the robustness of waypoint completion and system updating once undesirable contact conditions occur. We simulate this by first moving the POM to the desired waypoint and then pushing the object down in the grasp, shrinking the virtual object radius by determining a new relationship in $\mathcal{T}$ and changing the transformation between $\mathcal{X}$ and M. Results are depicted in Fig. 7A.

Execution began by tracing the top portion of the letter 'C' with the same contact relationship in Sec. VB. By pushing the object down in the grasp, we formed a new contact relationship

$\mathcal{T} = \{6.9cm, 6.7cm, 7.4cm\}$ with a virtual radius of 4.05cm and a POM distance of 6.3cm. Once the long edge of the letter was completed, the object was again moved down towards the palm creating a relationship $\mathcal{T} = \{5.9cm, 5.9cm, 6.3cm\}$ with a virtual radius of 3.52cm and a POM distance of 5.2cm. The final section of the letter 'C' was then completed.

During this evaluation, we note that the waypoint error and the overall geometry of the traced letter is similar to that of the execution in Sec. VB, showing robustness to Alg. 1 and the system. To further illustrate this, two additional objects, noted medium and small, were created to be representative of the virtual radii noted after object movement during the experiment. The distance to POM was set to the same as in Sec. VB, equal to 7.5cm. We again performed the writing of the letter 'C' to evaluate the waypoint error and completion. We find that this execution deviates slightly compared the original execution (Fig. 7C-D).

VI. Conclusions and Future Work

In this work we described a framework for extracting object transition maps for systems with fewer actuation inputs than mechanical degrees of freedom. This approach attempts to directly estimate the inherent non-linearities evident in controlling compliant or underactuated systems by evaluating an energy model and training a non-linear regressor. For our instantiation of this framework on the Yale OpenHand Model O, we formulated an energy model for passively elastic hands and it serves as the basis for data collection.

This approach benefits from its simplicity to implement, its robustness to parameter estimations, and its generalization to object geometries. The object was represented as a standardized grasp frame between three contacts, which in turn neglects the requirement to know global object geometry. By changing the relationship between contacts during data collection, we are able to fundamentally simulate an array of objects. We find that we can transfer the learned regression model from simulation with ease to the physical hand, through only roughly estimating spring constants.

By evaluating the action-reaction pairs of the system, we formed a Random Forests regression map that represents the actuation input required to transition the object pose towards the desired state. From training over varied contact relationships, the learned regression map is able to adapt to newly introduced objects and even continue manipulating when undesirable contact conditions, such as rolling, occur. We show the efficacy of the developed framework by deliberately moving contacts during manipulation, which changes the local object geometry, to complete a writing task.

As future work, we are interested in extending this framework for additional contacts, and evaluating how finger-gaiting can play a role in extending object manipulation. We plan to evaluate how models differ between those developed in simulation and that of data collected on a physical hand. We believe advanced within-hand manipulation planning can play a vital role in precision of the POM movement. Additionally, we plan to define this framework by further investigating the analytic structure of its estimated models.

References

[1] A. M. Okamura, N. Smaby, and M. R. Cutkosky, "An overview of dexterous manipulation," in *Proceedings 2000 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 255–262.

[2] A. Bicchi and V. Kumar, "Robotic grasping and contact: a review," in *Proceedings 2000 IEEE International Conference on Robotics and Automation (ICRA)*, vol. 1, pp. 348–353.

[3] M. A. Roa and R. Suárez, "Grasp quality measures: review and performance," *Auton. Robots*, vol. 38, no. 1, pp. 65–88, Jan. 2014.

[4] K. T. Yu, M. Bauza, N. Fazeli, and A. Rodriguez, "More than a million ways to be pushed. A high-fidelity experimental dataset of planar pushing," in *Proceedings 2016 IEEE International Conference on Intelligent Robots and Systems (IROS)*, pp. 30–37.

[5] A. Kimmel, R. Shome, Z. Littlefield, and K. Bekris, "Fast, Anytime Motion Planning for Prehensile Manipulation in Clutter," *arXiv:1806.07465*, Jun. 2018.

[6] K. Hang, J. A. Stork, N. S. Pollard, and D. Kragic, "A Framework for Optimal Grasp Contact Planning," *IEEE Robot. Autom. Lett.*, vol. 2, no. 2, pp. 704–711, Apr. 2017.

[7] H. Song, M. Y. Wang, and K. Hang, "Fingertip Surface Optimization for Robust Grasping on Contact Primitives," *IEEE Robot. Autom. Lett.*, vol. 3, no. 2, pp. 742–749, Apr. 2018.

[8] R. Platt, A. H. Fagg, and R. A. Grupen, "Manipulation gaits: sequences of grasp control tasks," in *Proceedings 2004 IEEE International Conference on Robotics and Automation (ICRA),* p. 801–806 Vol.1.

[9] L. Han and J. C. Trinkle, "Dextrous manipulation by rolling and finger gaiting," in *Proceedings 1998 IEEE International Conference on Robotics and Automation (ICRA)*, vol. 1, pp. 730–735.

[10] D. J. Montana, "The condition for contact grasp stability," in *Proceedings 1991 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 412–417.

[11] D. Prattichizzo and J. C. Trinkle, "Grasping," in *Springer Handbook of Robotics*, 2016, pp. 955–988.

[12] A. M. Dollar and R. D. Howe, "The highly adaptive SDM hand: Design and performance evaluation," in *International Journal of Robotics Research*, 2010, vol. 29, no. 5, pp. 585–597.

[13] R. Deimel and O. Brock, "A novel type of compliant and underactuated robotic hand for dexterous grasping," *Int. J. Rob. Res.*, vol. 35, no. 1–3, pp. 161–185, Jan. 2016.

[14] R. R. Ma, L. U. Odhner, and A. M. Dollar, "Dexterous manipulation with underactuated fingers: Flip-and-pinch task," in *Proceedings 2012 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3551–3552.

[15] L. U. Odhner and A. M. Dollar, "Dexterous manipulation with underactuated elastic hands," in *Proceedings 2011 IEEE International Conference on Robotics and Automation* (ICRA), pp. 5254–5260.

[16] B. Calli and A. M. Dollar, "Vision-based model predictive control for within-hand precision manipulation with underactuated grippers," in *Proceedings 2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2839–2845.

[17] A. Sintov, A. S. Morgan, A. Kimmel, A. M. Dollar, K. E. Bekris, and A. Boularias, "Learning a State Transition Model of an Underactuated Adaptive Hand," *IEEE Robot. Autom. Lett.*, vol. 4, no. 2, pp. 1287–1294, Apr. 2019.

[18] R. R. Ma and A. M. Dollar, "An underactuated hand for efficient finger-gaiting-based dexterous manipulation," in *2014 Proceedings IEEE International Conference on Robotics and Biomimetics (ROBIO),* 2014, pp. 2214–2219.

[19] J. Borras and A. M. Dollar, "A parallel robots framework to study precision grasping and dexterous manipulation," in *2013 Proceedings IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1595–1601.

[20] R. M. Murray, Z. Li, and S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation*, vol. 29. CRC Press, 1994.

[21] K. Tahara, S. Arimoto, and M. Yoshida, "Dynamic object manipulation using a virtual frame by a triple soft-fingered robotic hand," in *Proceedings 2010 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4322–4327.

[22] R. Ma and A. Dollar, "Yale OpenHand Project: Optimizing Open-Source Hand Designs for Ease of Fabrication and Adoption," *IEEE Robot. Autom. Mag.*, vol. 24, no. 1, pp. 32–40, Mar. 2017.

[23] A. Liaw and M. Wiener, "Classification and Regression by randomForest," *R news*, vol. 2, no. December, pp. 18–22, 2002.