A Probabilistic Planning Framework for Planar Grasping Under Uncertainty

Jiaji Zhou, Robert Paolini, Aaron M. Johnson, J. Andrew Bagnell and Matthew T. Mason
The Robotics Institute, Carnegie Mellon University
{jiajiz, rpaolini, dbagnell, matt.mason}@cs.cmu.edu, amj1@cmu.edu

***Abstract*— How can a robot design a sequence of grasping actions that will succeed despite the presence of bounded state uncertainty and an inherently stochastic system? In this paper, we propose a probabilistic algorithm that generates sequential actions to iteratively reduce uncertainty until object pose is uniquely known (subject to symmetry). The plans assume encoder feedback that gives a geometric partition of the post-grasp configuration space based on contact conditions. An offline planning tree is generated by interleaving computationally tractable open-loop action sequence search and feedback state estimation with particle filtering. To speed up planning, we use learned approximate forward motion models, sensor models, and collision detectors. We demonstrate the efficacy of our algorithm on robotic experiments with over 3000 grasp sequences using different object shapes, pressure distributions, and gripper materials where the uncertainty region is comparable to the size of the object in translation and with no information about orientation.**

I. Introduction

In order for robots to be successful in the real world, they must be aware of the ubiquitous uncertainty in the world. Much of the grasping literature focuses on planning grasping points to optimize a criterion, e.g., force closure [1], while assuming perfect knowledge of the world. However, traditional grasp planning is prone to failures. Consider the process of closing a parallel jaw gripper, the object will slide when the first finger engages contact and pushes the object before the other one touches the object. If the object does not end up slipping out, it can be jammed at an undesired position or grasped at an unexpected position. Pose uncertainty and the complete task mechanics of grasping need to be taken into consideration to achieve robust success.

This paper focuses on planar grasping under bounded uncertainty. We propose a tree-based probabilistic planning algorithm for a particle belief space with discretized action choices. Action selection for each node expansion uses a fixed depth open loop search subroutine. We use learned models from simulations based on the high fidelity mechanics model to significantly speed up this planning. We assume that the geometry of the robot gripper and the object is known. The mechanics model upon which the simulation is based assumes quasi-static rigid body mechanics with Coulomb friction (however the learned forward motion models could be adjusted to incorporate experimental data).

II. Problem Definition and Approach Overview

Problem: We are given an object of known geometry with bounded initial uncertainty $\delta_x, \delta_y, \delta_\theta$, a robot gripper with encoder feedback o, a set of grasping actions A. Design a deterministic strategy $a_{i+1} = \pi(Z_i)$ that maps sensor history $Z_i = \{z_1, \ldots, z_i\}$ to a collision-free action $a_{i+1} \in A$ such that the uncertainty size shrinks to an approximate singleton (below some tolerances $\varepsilon_x, \varepsilon_y, \varepsilon_\theta$) after a sequence of actions, as illustrated in the top row of Figure 1.

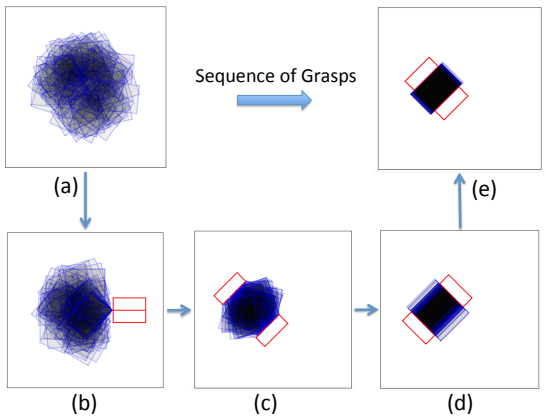


Fig. 1: A sequence of grasps with finger encoder feedback can shrink a bounded pose uncertainty to a singleton pose. Red boxes indicate the post-grasp gripper pose and filled rectangles with blue boundary correspond to object poses. The belief distribution of pose particles, whose density is proportional the darkness, is updated after observing the post-grasp finger distance. This sequence is only one of the many possible sequences (sensor histories) represented as paths from the root to leaf node in the tree in Figure 6.

The following observations are key to our approach:

- Contact conditions parametrized by the post-grasp finger distance observations partition the object configuration space into a few feasible subspaces.
- Given an initial uncertainty, the post-grasp sensor history is a sufficient representation of the current belief distribution over poses. Hence, a natural choice for the policy π is tree-structured that splits on new observations.

Note that reducing uncertainty to almost a singleton is often desired in most manufacturing settings. In many other applications reducing the poses to a smaller set is sufficient, which our planner is capable to deal with by changing the progress evaluation function (the SingletonRatio function in section VI) properly. Figure 2 shows a tree-based plan generated by our algorithm for a butterfly-shaped object where the centers are uniformly distributed in a circle of

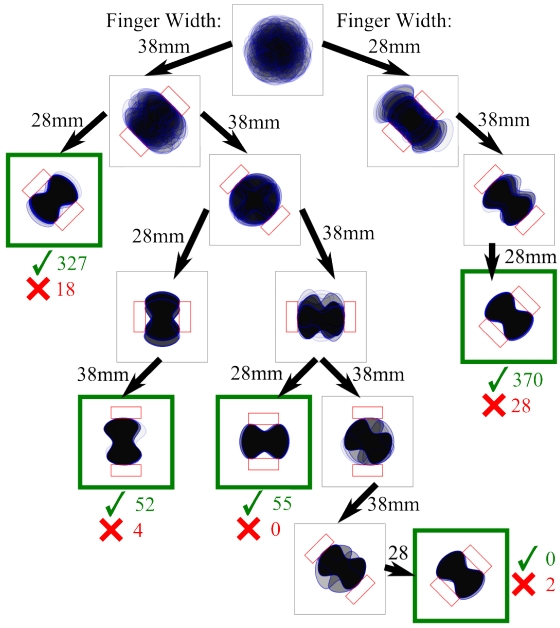


Fig. 2: A tree-based plan generated by our algorithm for a butterfly-shaped object.

radius 10mm and the frame angles are completely unknown. In each node, red boxes indicate the expected final gripper pose and filled rectangles with blue boundary correspond to particle-based belief distribution of the object poses where density is proportional the darkness. The actions choices are four different angled squeezes with the same hand center. A grasping action is applied at each non-terminal node which branches out child nodes depending on the observed post-grasp distances between fingers. At the terminal nodes (boxed in green), the robot predicts the average pose, with counts of successful and unsuccessful predictions for a total of 856 experimental grasp trials. Note that the tree is generated offline with stored action choice at each non-terminal node, and during execution time only the sensor history is recorded to localize the current node in the tree. No belief distribution needs to be stored (except the singleton average pose at the leaf node) nor any inference for action choice is required.

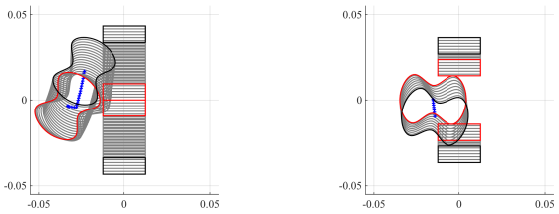
III. Related Work

Our work is related to the general framework of planning manipulation actions with funnels, i.e., the set of poses associated with a robot action that are guaranteed to reach a particular goal set. Sensorless uncertainty reduction techniques have proven to be successful in many applications, often using a possibilistic approach and assuming worst case motion error. Erdmann and Mason [2] demonstrated a parts feeding strategy using tray-tilting where mechanical motion alone can eliminate uncertainty. Lozano-Perez et al. [3] (Preimage backchaining) and Erdmann [4] (Backprojection) developed strategies to chain a sequence of operations to guarantee operation success despite uncertainty. With sensors, sequential composition of active feedback controllers with overlapping funnels can achieve the desired result, as studied in [5], [6]. The planner presented here builds a tree structure to reason about these relationships – prior work has built similar trees from contact conditions explicity as a stratifed set [7], simplicial complex [8], or general state transition graph [9].

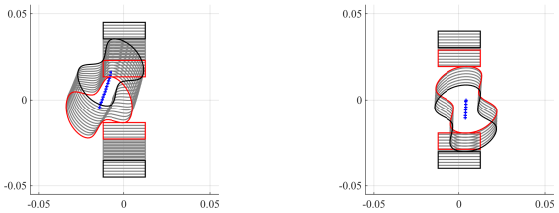
The planar manipulations studied here involve finite motions on a supporting surface with indeterminate pressure and friction distribution. Mason [10] first studied the mechanics of quasi-static pushing and came up with the voting theorem which dictates the sense of rotation given a push action and the center of pressure despite an uncertain pressure distribution. Close to our work is [11] where a 2 dimensional operational space based on the voting theorem is constructed for planning sensorless push-squeeze operations. This result includes worst case guarantee. However, many unrealistic assumptions are made in order to reduce the state space and create finite discrete transitions, including infinitely long fingers approaching the object from infinitely far away. Additionally, the execution length of each action is not addressed. Goldberg [12] proposed a Bayesian approach to plan a sequence of squeezing actions to reduce orientation uncertainty. The use of a diameter function along with the assumption of simultaneous contacts of frictionless infinitely long parallel jaw grippers enables a discrete transition analysis. These assumptions deviate from practical hardware and task settings. The open loop component of our algorithm eliminates these assumptions and reasons over all factors including hand and object geometry, finite friction, and potential collisions in between actions.

When the goal is only to generate a uncertainty-aware stable grasp, Dragiev et al. [13] used Gaussian process implicit surface to model shape and pose uncertainty with tactile feedback to tradeoff grasp exploration and exploitation while assuming objects remain static and the exact contact points on the hand can be detected. Laaksonen et al. [14] presented a probabilistic approach to maximize the posterior grasp stability through several grasp attempt using learned motion and sensor model. Our approach bears similar spirit but additionally leverages the geometrical structure of the configuration space for grasping to enable exact tree policy generation offline. In execution time, no belief distribution is stored nor any inference computation is required. Only the sensor history is maintained. Additionally, robustness is built into our planner since we can perform efficient and physically consistent sampling to capture the stochastic properties of the system mechanics.

Our work is also related to uncertainty reduction via tactile sensing. Koval et al. [15] proposed the manifold filtering algorithm which alleviates the particle starvation problem for pushing applications. Localization through performing a sequence of guarded moves is studied in [16], [17]. Paolini et al. [18] used a learned tactile model based on a simple hand with finger encoder feedback to localize the in-hand object pose. Our algorithm for generating sensored plans assume simple finger encoder readings for determining the post-grasp distance between fingers.



(a) Slipped to free space. (b) Jamming.



(c) Grasped with offset. (d) Grasped with offset.

Fig. 3: Simulation results based on [19] illustrating the process of a parallel jaw gripper squeezing along the y axis when the butterfly-shaped object is placed at different initial poses. The initial, final, and intermediate gripper configurations and object poses are in black, red, and grey, respectively. Blue plus signs trace out the center of mass trajectory of the object.

IV. Grasping Mechanics

This section describes the mechanics model (detailed in [19]) to resolve the instantaneous object motion (if any) given the intended motion of the hand. All physical vector quantities in this section are with respect to the local object frame. The simulation software uses Matlab ode45 and polygonal collision geometry where curves are approximated by dense points along the boundary. Figure 3 shows example grasping simulation results on a non-trivial butterfly-curved object.

When the hand moves the object on a frictional supporting surface quasi-statically (with negligible inertia force), the applied wrench should balance the reaction frictional wrench from the surface. We refer the readers to [10], [20] for a detailed treatment on planar sliding mechanics. The principle of maximum dissipation as a generalized Coulomb's Law for point friction is formalized in [21]. Goyal et al. [20] further extended to planar frictional wrench and proved that all the possible static and sliding frictional wrenches, regardless of the pressure distribution, form a convex set whose boundary is called as limit surface. Zhou et al. [22] proposed a convex polynomial level-set representation identifiable through sum-of-squares relaxation. Denote by $H(\mathbf{F})$ the convex polynomial function for the limit surface, the resultant body twist $\mathbf{V}$ for the applied body frictional wrench $\mathbf{F}$ is parallel to the gradient $\nabla H(\mathbf{F})$:

$$\mathbf{V} = s\nabla H(\mathbf{F}) \qquad s > 0, \tag{1}$$

where s is a positive scalar.

A. Multi-contacts

The force-motion model can be combined with a linear complementarity treatment of unilateral frictional contacts [23] to derive the quasi-static mechanics. Denote by $\mathbf{p}_i$ the ith contact point and the Jacobian matrix $J_{\mathbf{p}_i} = \begin{bmatrix} 1 & 0 & -\mathbf{p}_i^y \\ 0 & 1 & \mathbf{p}_i^x \end{bmatrix}$. The total applied wrench $\mathbf{F}$ is the sum of normal and frictional wrenches over all applied contacts:

$$\mathbf{F} = \sum_{i=1}^{m} J_{\mathbf{p}_i}^T (f_{\mathbf{n_i}} \mathbf{n}_{\mathbf{p}_i} + D_{\mathbf{p}_i} \mathbf{f}_{\mathbf{t_i}}). \tag{2}$$

where m is the total number of contacts, $f_{\mathbf{n_i}}$ is the normal force magnitude along normal $\mathbf{n_i}$, and $\mathbf{f}_{\mathbf{t_i}}$ is the vector of tangential friction force magnitudes along the column vector basis of $D_{\mathbf{p_i}} = [\mathbf{t}_{p_i}, -\mathbf{t}_{p_i}]$. The velocity at the contact point $\mathbf{p}_i$ on the object is given by $J_{\mathbf{p}_i}\mathbf{V}$. The applied velocity by the hand at $\mathbf{p}_i$ is denoted by $\mathbf{v}_{\mathbf{p}_i}$. The first order complementarity constraints on the normal force magnitude and the relative velocity are given by:

$$0 \le f_{n_i} \perp (\mathbf{n}_{\mathbf{p}_i}^T (J_{\mathbf{p}_i}\mathbf{V} - \mathbf{v}_{\mathbf{p}_i})) \ge 0, \tag{3}$$

where $\perp$ means the product between the left symbol and the right symbol equals zero. The complementarity constraints for Coulomb friction are given by:

$$0 \le \mathbf{f}_{t_i} \perp (D_{\mathbf{p}_i}^T (J_{\mathbf{p}_i}\mathbf{V} - \mathbf{v}_{\mathbf{p_i}}) + e\lambda_i) \ge 0, \tag{4}$$

$$0 \le \lambda_i \perp (\mu_i f_{n_i} - e^T \mathbf{f}_{t_i}) \ge 0, \tag{5}$$

where μ_i is the coefficient of friction at $\mathbf{p}_i$ and $e = [1;1]$. λ_i is an positive auxillary variable as part of enforcing sliding/sticking complementarity [23]. It can be shown [19] that if we use a convex quadratic form (ellipsoid limit surface), i.e., $H(\mathbf{F}) = \mathbf{F}^T A \mathbf{F}$, then the entire problem (equations 1 to 5) is a standard linear complementarity problem (LCP). For generic high order convex polynomial form, it's equivalent to solving a sequence of LCP problems. Importantly, the no-solution case corresponds to the object being jammed or grasped by the gripper in which case the simulation terminates.

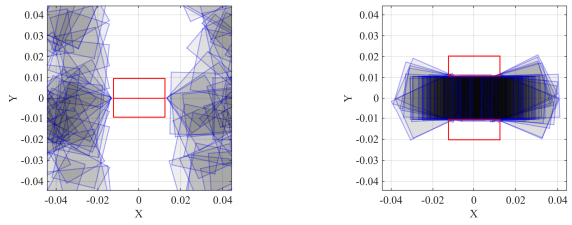
To model the stochastic nature of the sliding mechanics, we sample physically consistent polynomial parameters for $H(\mathbf{F})$ and coefficients of friction between the hand and object, as detailed in [19]. Throughout the paper, the degree of freedom[1] n_{df} in the Wishart distribution [24] for sampling positive semi-definite (PSD) matrices is set as 100.

V. Configuration Space Partition and Fast Approximate Models

A. Post-grasp Configuration Space Partition

Contact conditions parametrized by the post-grasp distance between the fingers naturally partition the configuration space to lower dimensional subspaces. Figure 4 shows the partitions of the object poses based on different grasping outcomes indicated by the post-grasp finger widths. The

[1]When we sample the PSD matrix A in $H(\mathbf{F}) = \mathbf{F}^T A \mathbf{F}$, the variance of A_{ij} equals $\frac{1}{n_{df}}(\hat{A}_{ij}^2 + \hat{A}_{ii}\hat{A}_{jj})$, where $\hat{A}$ is an initial estimate for a given pressure distribution.



(a) Free space: the object slips out or is not touched.

(b) The first constrained space: the object is grasped at the long edges.

(c) The second constrained space: the object is jammed between the diagonal vertices.

(d) The third constrained space: the object is grasped at the short edges.

Fig. 4: Partition of the configuration space of a rectangle (30×21 mm) based on the post-grasp distance between the fingers. Red boxes correspond to the average gripper pose.

results are generated by first simulating trajectory roll-outs of a squeezing action with different initial poses using the grasping mechanics model described in section IV, and then performing k-means clustering of the final finger distances. In this case, k equals 4. The choice of k depends on the minimum number of characteristic equilibria. Note that the gripper widths within the same cluster has nonzero variance which matches the non-negligible real world sensor noise.

B. Fast approximate models

Our planning algorithm requires three components:

1) A forward motion model that maps an initial object pose to a final object pose, both in the hand frame. Note that all the squeezing actions with the same initial finger opening width (in general the same initial finger configuration) can share the model since the post-action object poses are always transformed to the hand frame for the subsequent action.
2) A sensor model that maps a final object pose to the post-grasp observation (in this case, expected finger width).
3) A collision detector that checks whether a given object pose is in collision with the initial placement of the hand.

Although they are all available through trajectory roll-out simulations, planning with particle based belief representation needs to be orders of magnitude faster. We need to compute a forward model of planar grasping. A small deviation in initial pose can be the difference between a successful grasp and a missed one. By probabilistically modeling a planar grasp, we can capture this behavior. We use kernel conditional density estimation (KCDE) [25] to capture the inherent non-linear, multimodal nature of the distribution. Denote by $\mathbf{q} = [x, y, \theta]^T$ the object pose in the hand frame and let $\hat{\mathbf{q}} = [x, y, \rho\theta]^T$ be the normalized pose [26] with consistent unit per dimension, where ρ is the radius of gyration (or any meaningful characteristic length) of the object. Denote the RBF kernel by

$$K_h(\hat{\mathbf{q}}_1, \hat{\mathbf{q}}_2) = \frac{1}{\eta} \exp\left(-\|\hat{\mathbf{q}}_1 - \hat{\mathbf{q}}_2\|^2/(2h^2)\right),$$

where η is a normalization term and h is the bandwidth parameter. Suppose we have collected a data set $\{\hat{\mathbf{q}}^i, \hat{\mathbf{q}}^f\}$ of N initial and final object poses. The conditional probability of a final normalized pose $\hat{\mathbf{q}}^{f*}$ to occur given an initial normalized pose $\hat{\mathbf{q}}^{i*}$ is given by:

$$P(\hat{\mathbf{q}}^{f*}|\hat{\mathbf{q}}^{i*}) = \frac{\sum_j^N K_{h_1}(\hat{\mathbf{q}}^{i*}, \hat{\mathbf{q}}^i_j) K_{h_2}(\hat{\mathbf{q}}^{f*}, \hat{\mathbf{q}}^f_j)}{\sum_j^N K_{h_1}(\hat{\mathbf{q}}^{i*}, \hat{\mathbf{q}}^i_j)}. \quad (6)$$

The bandwidths h_1 and h_2 throughout the experiments are set as 0.5. To sample from this conditional distribution, note that given an initial pose $\hat{\mathbf{q}}^{i*}$, Equation 6 becomes a mixture of Gaussians:

$$P(\hat{\mathbf{q}}^{f*}|\hat{\mathbf{q}}^{i*}) = \sum_j^N w_j K_{h_2}(\hat{\mathbf{q}}^{f*}, \hat{\mathbf{q}}^f_j), \quad (7)$$

where

$$w_j = \frac{K_{h_1}(\hat{\mathbf{q}}^{i*}, \hat{\mathbf{q}}^i_j)}{\sum_j^N K_{h_1}(\hat{\mathbf{q}}^{i*}, \hat{\mathbf{q}}^i_j)} \quad (8)$$

To sample $\hat{\mathbf{q}}^{f*}$, we can first sample a center $\hat{\mathbf{q}}^f_j$ according to the weights w_j, and then sample from that corresponding Gaussian. Further speed-up is achieved by only using k nearest neighbors for sampling. On a 2.4GHz i5 core, it takes about 1 second to roll out 5000 particles for a KCDE model trained with 10000 data points and using 400 nearest neighbors (with kd-tree representation) for prediction, compared with 0.5 to 3 seconds for rolling out 1 particle in simulation.

We use a regression tree to predict the post-grasp finger distance given a final pose. The collision detector is trained using a binary classification tree.

C. Belief Update With Subspace Projection

The predictions from the KDCE model often have non-zero variance in the full SE(2) space. However, if we know that a particle belongs to a constrained subspace as indicated by the observation, then the residuals orthogonal to the subspaces (manifolds) should be eliminated. Koval et al. [15] used this idea to improve filtering algorithm that samples particles from the constrained manifold with weights proportional to the motion model probability (dual sampling), when the tactile sensor indicates that the contacts are active. In our experiments, we have found that a projection onto the constrained subspace is sufficient. The free space and constrained subspaces are represented by storing

final poses associated with the observation clusters through many trajectory roll-outs. These roll-outs start with randomly sampled initial poses from a large enough uncertainty area which covers all the possible poses during the grasping sequence.

To perform belief updates after rolling out the KCDE forward motion, we use the sensor model to predict the post-grasp gripper distance for each particle. For particles that belong to the same subspace based on the predicted observation, we perform projection onto the subspace using a nearest neighbor association.

VI. Planning

The offline planning tree generation algorithm is summarized in Algorithm 1. We need to first perform three pre-planning steps:

1) Sample initial poses from a large enough uncertainty (expect to be larger than the specified perception uncertainty) and do trajectory roll out simulations for a hand squeezing action.
2) Perform K-means clustering on the post-grasp finger distances and construct subspace partitions. Train approximate motion models, sensor models and collision detectors.
3) Sample initial poses from a query initial uncertainty and construct the root node with associated initial belief particles before calling Algorithm 1 to construct an offline sensored tree plan.

Note that different query initial uncertainty for the same object can share the approximate models so the first two time-consuming simulation steps only need to be done once. Our goal is to increase the singleton ratio using the shortest number of actions. We list the following key functions as below:

- *SingletonRatio*(X, ε): Computes the ratio of a given set of poses X within tolerance values $\varepsilon = [\varepsilon_x, \varepsilon_y, \varepsilon_\theta]$ with respect to the average pose. To compute the average pose of a given set of poses: we mapped the poses $[x, y, \theta]$ to the augmented space $[x, y, \cos(2\theta), \sin(2\theta)]$ (for rotational symmetry) to take the average and then map back to the original space.
- *MarkTerminalNode*(C): Marks the node C as a terminal (leaf) node when the singleton ratio of its pose particles is beyond a threshold r_s. The average pose is treated as the final prediction and no more action is needed. We use $r_s = 0.95$ throughout the experiments.
- *CreateNode*(X, Z, r, d): Creates a new node C where $C.X$ is the pose particles, $C.Z$ is the observations (optional), $C.r$ is the singleton ratio and $C.d$ is the depth in the tree.
- *ForwardMotionModel-KCDE*(X, a): Maps the initial set of poses X to a final set of poses for a given action a. Inside the function, the initial poses are first transformed to hand frame, then the final poses are sampled from the learned KCDE model, before finally transformed back to the world frame.
- *CollisionFree-Decision*(X, a): Checks whether given object poses X are in collision with the initial placement of the hand parameterized by the action a.
- *SensorModel-RegressionTree*(X): Given a set of final object poses X , use the learned regression tree to predict the observations Z, i.e., expected post-grasp gripper widths.
- *BeliefUpdateProjection*(X, Z): Project the particle poses X onto the corresponding constrained manifold indicated by the observations Z.
- *AddToTree*(C, N, Z): Add N as a child node to C and label the edge with observation Z.
- *UpdateBestAction*(a^*, H): When an queue element's action sequence results in a larger singleton ratio, update the best first action to apply.

Input: current node **C**, singleton tolerance values ε and threshold ratio r_s, max subtree search depth d
Output: Planning Tree
if $C.r > r_s$ **then**
    MarkTerminalNode(C);
    Return;
else
    a, X, Z = OpenLoopSearch(C, ε, d) ;
    C.action = a;
    foreach *partition* (X_i, Z_i) *based on observations* **do**
        r_i = SingletonRatio(X_i, ε);
        N = CreateNode(X_i, Z_i, r_i, $E.d+1$);
        AddToTree(C, N, Z_i) ;
        ExpandTree(**N**, ε, r_s, d);
    end
end

Algorithm 1: ExpandTree

condition **Input**: Node element E to search for action, singleton tolerance values ε, search depth d
Output: Action a^*, Particle pose-observation value pairs (X, Z)
Q = EmptyQueue();
AddQueueElement(Q, E);
while *subtree depth* $< d$ **do**
    H = PopQueueElement(Q);
    UpdateBestAction(a^*, H) ;
    OpenLoopSearchBFSExpand(H, ε, Q)
end
X = ForwardMotionModel-KCDE($E.X$, a^*);
Z = SensorModel-RegressionTree(X);
X = BeliefUpdateProjection(X, Z);

Algorithm 2: OpenLoopSearch

The sensorless subroutine *OpenLoopSearch* described in Algorithm 2 uses a breadth-first search to find the optimal length-d sequence of actions that maximizes the ratio of singletons. Algorithm 3 describes the single-level expansion procedure inside *OpenLoopSearch*. We use a pruning criteria

Input: Node element E, singleton tolerance values ε, queue Q
foreach *action* a **do**
  if *CollisionFree-DecisionTree*$(E.X, a)$ **then**
    X_p = ForwardMotionModel-KCDE(X_t, a);
    r = SingletonRatio(X_f, δ);
    if $r > \alpha E.r$ **then**
      N = CreateNode(X_f, {}, r, $E.d+1$);
      AddQueueElement(Q, N);
    end
  end
end

Algorithm 3: OpenLoopSearchBFSExpand

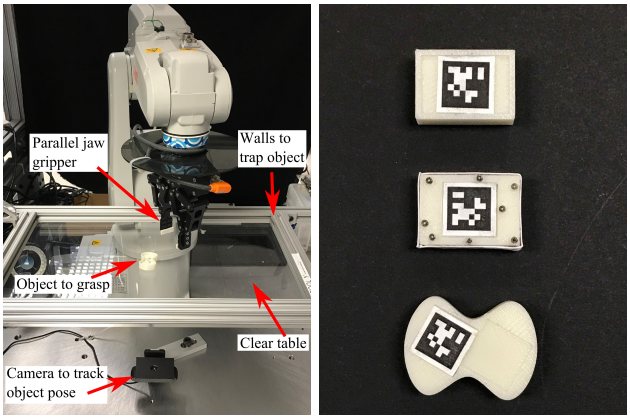
that avoids expansion if the singleton ratio decreases by (1 - α) compared with its parent. α is set as 0.75 throughout the experiments. The first action in the sequence returned by the open loop search will be recorded as the action choice for the current node, followed by expansions of children nodes which split on the possible sensor observations. Note that we can also execute the entire open loop action sequence and perform expansions afterwards. At the extreme is complete sensorless planning where the open loop planner keeps searching until the best singleton ratio exceeds the threshold r_s given time budget[2].

If we view the planner as approximately solving a POMDP [27], the *OpenLoopSearch* can be treated as a computational tractable heuristic in choosing the action rather than subtree search that also splits the nodes using predicted observations, hence reducing the branching factor from $k|A|$ to $|A|$.

VII. Experiments

The algorithm is implemented in Matlab. Example trees in Figure 6 (large uncertainty) and Figure 2 (small uncertainty) take less than 2 hours and 5 minutes to generate offline, respectively. For all trees in the experiments, we first generate simulation data through 9000 trajectory roll-outs with initial poses sampled from a region 3 times the size of uncertainty. Convex quadratic forms of $H(\mathbf{F})$ are fitted [22] assuming the same kind of pressure distributions around the boundary of the objects, despite using various different pressure distributions for experiments, demonstrating robustness of plans with respect to model parameters. The uncertain coefficient of friction between the object and the robot hand is uniformly sampled from 0.4 to 0.6. The gripper fingers are modeled as 27.5mm $\times$ 9.5mm rectangles. We then use the simulation data to train KCDE-based forward motion models, sensor models and collision detectors. To generate the tree plan, we uniformly sample 6000 particles from the initial uncertainty region. The tolerance values ε_x and ε_y both equal 2.5mm, and ε_θ equals 5 degrees. Additionally we construct a kd tree and use 400 nearest neighbors for KCDE-based forward motion model prediction.

[2]We have found that feasible and better plans are much easier to generate with feedback whereas completely open loop plans. For example, the best 4 step open loop plans with the same action choices (increasing search depth does not improve performance) for the high friction low uncertainty experimental settings (columns 1 to 3 of Table I) do not exceed 70 percent success due to unrecognizable repeated jamming.



(a) Grasping platform showing ABB arm, Robotiq gripper, object to grasp, transparent table, and underneath camera for verification.

(b) Test objects: (top) rectangular with boundary pressure distribution, (middle) rectangular (glued with hard paper) with discrete point pressure distribution, (bottom) butterfly with boundary pressure distribution.

Fig. 5: Experimental Setup

A. Setup

The experimental setup for planar grasping is shown in Figure 5a. We use a 6 degree-of-freedom ABB-120 industrial robot. We attach to the arm a Robotiq C-85 2-fingered parallel jaw gripper [28]. The two fingers open and close simultaneously. The robot wrist is synchronized with the finger motion to maintain the tip at fixed height during grasping. The vision system consists of a Logitech c930e webcam looking through a transparent acrylic table. We attach an an AprilTag [29] to the bottom of the object for ground truth verification with the predicted singleton pose. We obtain a 0.3mm object pose estimation accuracy near the center of the table, and a 1.5mm accuracy near the edge of the table. If the object or robot during a grasp would be too close to the edge of the table, the robot drags the object back to the center before starting its next grasp.

B. Results Analysis

During execution, we read the finger distance from the Robotiq hand encoders and descend to the child node with the closest expected observation. The robot stops execution when terminal node is reached and reads the ground truth object pose value from the vision system. Define the the combined distance metric between the predicted pose $q_1 = [x_1, y_1, \theta_1]$ at the terminal node and the ground truth pose $q_2 = [x_2, y_2, \theta_2]$ from the camera as follows:

$$d(q_1, q_2) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} + \rho d(\theta_1, \theta_2) \qquad (9)$$
$$d(\theta_1, \theta_2) = \min(|\theta_1 - \theta_2|, 2\pi - |\theta_1 - \theta_2|).$$

experiment	1	2	3	4	5	6	7	8*
shape	butterfly	butterfly	rectangle	rectangle	rectangle	rectangle	rectangle	rectangle
pressure	boundary	boundary	boundary	points	points	boundary	boundary	boundary
uncertainty size	10mm	15mm	12.75mm	12.75mm	25mm	25mm	25mm	25mm
gripper-object materials	felt-abs	felt-abs	foam-abs	steel-abs	felt-paper	foam-abs	steel-abs	steel-abs
grasp trials	856	327	682	968	193	202	95	100
success rate	0.939	0.835	0.972	0.872	0.917	0.896	0.768	0.851

TABLE I: Experimental results of grasping with different combinations of object shapes, frictional materials, supporting pressure distributions and uncertainty sizes. The orientations are completely unknown. Note that the planner is unaware of the exact pressure distributions and coefficients of friction between the object and the gripper. The motion and sensor models trained from stochastic simulations capture the uncertainty in system parameters. Experiment 8* is an improved result over experiment 7 after regulating grasping force via motor current monitoring.

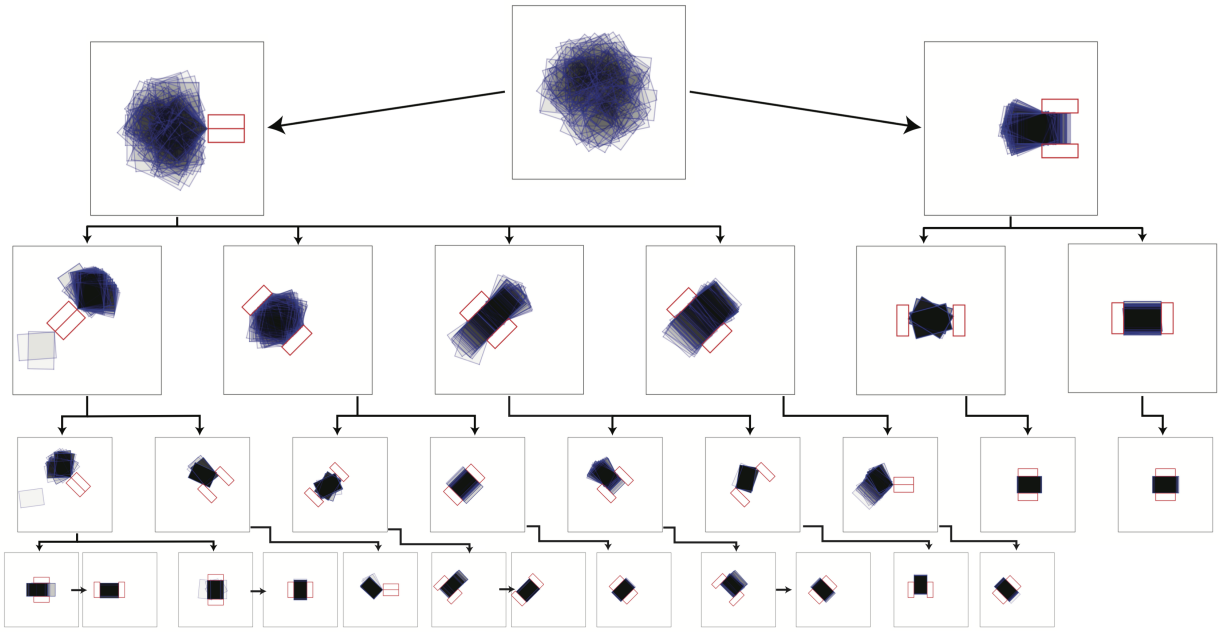


Fig. 6: A generated tree-based plan for a rectangular object (30mm * 21mm) where the centers are uniformly distributed in a circle of radius 25mm and the frame angles are completely unknown. A total of 20 action choices are used.

A grasp sequence is considered successful if the combined metric is smaller than 3mm. Table I shows the experimental results of over 3000 experiments on the robot. The orientations are completely unknown. The centers of the objects are uniformly distributed in a circle of radius specified in the third row. For low uncertainty settings (experiments 1 to 4), the action space consists of 4 different angles with one fixed hand frame center. For high uncertainty settings (experiments 5 to 8), the action space consists of 4 different angles with 5 different hand frame centers along a line, hence a total of 20 actions. The most frequent failures are due to unexpected dynamic behavior when objects are jammed in an unstable equilibrium: the large force applied by the stiff gripper on the light objects cause "fly away" or slipping phenomenon not captured by the quasi-static simulation, particularly for the case of low friction steel gripper material for large uncertainty in experiment 7 of Table I. Other failure patterns include missing multi-modal patterns during node expansion and cascading small amount of objects movement when the gripper looses each grasp, both of which can cause unexpected collision.

VIII. Conclusion and Future Work

This paper presents a tree-based probabilistic planning algorithm for planar grasping under uncertainty capable of generating action sequences with or without sensor feedback. To improve planning speed, the forward motion model is approximated by kernel conditional density estimation. Regression and classification trees are trained to approximate the sensor model and the collision detector.

Learning from experimental data is crucial to capture high variance input-output mappings that deviate from the underlying physics model assumptions, e.g., initial object poses that can "fly away" later during the grasp action. To avoid these issues in the future, the planner could prioritize more

stable actions while making uncertainty reduction progress. We have not addressed the optimality of the plans. The trees generated are not optimal: the sequences can be longer than necessary and some terminal nodes are not visited. The system would also benefit from automatic tuning of the bandwidth for the KCDE-based forward motion model. The plan suffers from slow convergence speed and generates redundant actions if the bandwidth is too large, whereas low probability outcomes are missed during node expansion if the bandwidth is too small. Using a larger set of manipulation actions including pure pushing and push-grasps can significantly increase the planner's capability to deal with more complicated object geometry and larger uncertainty region. We plan to use Monte Carlo Tree Search techniques [30] to deal with the computational challenges brought by a larger branching factor.

The learned forward motion and sensor model is task-specific to inputs of object geometry and gripper action type. It's possible to encode these inputs as features of the learned models for generalization purpose. Extending the framework to the three-dimensional setting remains as a challenging open problem due to increased state space and the necessity of intelligent design of three-dimensional actions. Although our planner is capable of generating open-loop plans with fewer assumptions compared with existing literature, we note that planning is easier, faster, and more robust with even the simplest feedback – finger encoder readings (available for most off-the-shelf robot hands). This sets up a promising framework of integrating basic proprioceptive feedback on an industrial robot arm and gripper that produces more robust plans without the need for high cost external sensors [31].

ACKNOWLEDGMENTS

The authors are inspired by the block sorting demonstration in the CMU Personal Robotics Lab and would like to thank Michael Koval, Siddhartha Srinivasa and Nancy Pollard for thoughtful discussions.

REFERENCES

[1] C. Ferrari and J. Canny, "Planning optimal grasps," in *Robotics and Automation, 1992. Proceedings., 1992 IEEE International Conference on*, pp. 2290–2295, IEEE, 1992.

[2] M. A. Erdmann and M. T. Mason, "An exploration of sensorless manipulation," *IEEE Journal on Robotics and Automation*, vol. 4, no. 4, pp. 369–379, 1988.

[3] T. Lozano-Perez, M. T. Mason, and R. H. Taylor, "Automatic synthesis of fine-motion strategies for robots," *International Journal of Robotics Research*, vol. 3, no. 1, 1984.

[4] M. Erdmann, "Using backprojections for fine motion planning with uncertainty," *The International Journal of Robotics Research*, vol. 5, no. 1, p. 19, 1986.

[5] R. R. Burridge, A. A. Rizzi, and D. E. Koditschek, "Sequential composition of dynamically dexterous robot behaviors," *The International Journal of Robotics Research*, vol. 18, no. 6, pp. 534–555, 1999.

[6] R. Tedrake, I. R. Manchester, M. Tobenkin, and J. W. Roberts, "Lqr-trees: Feedback motion planning via sums-of-squares verification," *The International Journal of Robotics Research*, vol. 29, no. 8, pp. 1038–1052, 2010.

[7] B. Goodwine and J. Burdick, "Motion planning for kinematic stratified systems with application to quasi-static legged locomotion and finger gaiting," *IEEE Transactions on Robotics and Automation*, vol. 18, pp. 209–222, Apr 2002.

[8] A. M. Johnson and D. E. Koditschek, "Toward a vocabulary of legged leaping," in *Proceedings of the 2013 IEEE Intl. Conference on Robotics and Automation*, pp. 2553–2560, May 2013.

[9] M. Sobotka and M. Buss, "Locomotion studies for a 5dof gymnastic robot," in *Intelligent Robots and Systems, Proceedings of the IEEE/RSJ International Conference on*, pp. 3275–3280, IEEE, 2005.

[10] M. T. Mason, "Mechanics and planning of manipulator pushing operations," *IJRR*, vol. 5, pp. 53–71, Fall 1986.

[11] R. C. Brost, "Automatic grasp planning in the presence of uncertainty," *The International Journal of Robotics Research*, vol. 7, no. 1, pp. 3–17, 1988.

[12] K. Goldberg and M. T. Mason, "Bayesian grasping," *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1264–1269, 1990.

[13] S. Dragiev, M. Toussaint, and M. Gienger, "Uncertainty aware grasping and tactile exploration," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pp. 113–119, IEEE, 2013.

[14] J. Laaksonen, E. Nikandrova, and V. Kyrki, "Probabilistic sensor-based grasping," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pp. 2019–2026, IEEE, 2012.

[15] M. C. Koval, N. S. Pollard, and S. S. Srinivasa, "Pose estimation for planar contact manipulation with manifold particle filters," *The International Journal of Robotics Research*, vol. 34, no. 7, pp. 922–945, 2015.

[16] S. Javdani, M. Klingensmith, J. A. Bagnell, N. S. Pollard, and S. S. Srinivasa, "Efficient touch based localization through submodularity," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pp. 1828–1835, IEEE, 2013.

[17] K. Hsiao, L. P. Kaelbling, and T. Lozano-Perez, "Grasping pomdps," in *Robotics and Automation, 2007 IEEE International Conference on*, pp. 4685–4692, IEEE, 2007.

[18] R. Paolini, A. Rodriguez, S. S. Srinivasa, and M. T. Mason, "A data-driven statistical framework for post-grasp manipulation," *The International Journal of Robotics Research*, vol. 33, no. 4, pp. 600–615, 2014.

[19] J. Zhou, J. A. Bagnell, and M. T. Mason, "A fast stochastic contact model for planar pushing and grasping: Theory and experimental validation," *arXiv preprint arXiv:1705.10664*, 2017.

[20] S. Goyal, A. Ruina, and J. Papadopoulos, "Planar sliding with dry friction. Part 1. Limit surface and moment function," *Wear*, vol. 143, pp. 307–330, 1991.

[21] J. J. Moreau, "Unilateral contact and dry friction in finite freedom dynamics," in *Nonsmooth Mechanics and Applications*, pp. 1–82, Springer, 1988.

[22] J. Zhou, R. Paolini, J. A. Bagnell, and M. T. Mason, "A convex polynomial force-motion model for planar sliding: identification and application," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 372–377, May 2016.

[23] D. E. Stewart and J. C. Trinkle, "An implicit time-stepping scheme for rigid body dynamics with inelastic collisions and coulomb friction," *International Journal for Numerical Methods in Engineering*, vol. 39, no. 15, pp. 2673–2691, 1996.

[24] J. Wishart, "The generalised product moment distribution in samples from a normal multivariate population," *Biometrika*, pp. 32–52, 1928.

[25] M. Rosenblatt, "Conditional probability density and regression estimators," *Multivariate analysis II*, vol. 25, p. 31, 1969.

[26] M. Erdmann, "On a representation of friction in configuration space," *IJRR*, vol. 13, no. 3, pp. 240–271, 1994.

[27] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*, vol. 1. MIT press Cambridge, 1998.

[28] ROBOTIQ, "Two-finger adaptive robot gripper." `http://robotiq.com/products/adaptive-robot-gripper/`, 2017. [Online; accessed 15-Feb-2017].

[29] E. Olson, "Apriltag: A robust and flexible visual fiducial system," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pp. 3400–3407, IEEE, 2011.

[30] L. Kocsis and C. Szepesvári, "Bandit based monte-carlo planning," in *European conference on machine learning*, pp. 282–293, Springer, 2006.

[31] J. F. Canny and K. Y. Goldberg, "" risc" industrial robotics: recent results and open problems," in *Robotics and Automation, 1994. Proceedings., 1994 IEEE International Conference on*, pp. 1951–1958, IEEE, 1994.