Robot Manipulation Planning Under Uncertainty Using Hybrid Dynamics

Ajinkya Jain
Department of Mechanical Engineering,
University of Texas at Austin,
Austin, TX 78712, USA
Email: ajinkya@utexas.edu

Scott Niekum
Department of Computer Science,
University of Texas at Austin,
Austin, TX 78712, USA
Email: sniekum@cs.utexas.edu

***Abstract*— The difficulty of many robot manipulation tasks stems from stochasticity and partial observability coupled with highly nonlinear dynamics. However, much of the nonlinearity in manipulation tasks can be explained by sudden changes in contact or articulated state. This fact allows the dynamics to be naturally factored into a discrete set of simpler models combined together as a *hybrid dynamics model*. To leverage these hybrid dynamics for efficient planning under uncertainty, we introduce a novel POMDP planner that computes and stabilizes around trajectories in a hybrid belief space. We evaluate the proposed planner in both a simulated navigation domain, as well as an assembly task with a real robotic manipulator.**

I. Introduction

One of the biggest challenges in robot manipulation is to develop feasible motion plans for systems having highly nonlinear dynamics in the presence of partial or noisy observations. However, much of the nonlinearity in manipulation tasks can be explained by sudden changes in contact or articulated state. Thus, the complex dynamics of such tasks can often be naturally decomposed into a discrete set of simpler linear contact and articulated motion models of which only one is active at any given time. For example, a furniture assembly task can be decomposed into four different dynamics models, as shown in Figure 1: the leg and the table can be (1) free bodies with respect to each other, (2) in planar contact, (3) restricted to helical motion while being screwed in, or (4) rigidly connected after the screw is fully tightened. Mathematically, such models can be expressed as hybrid models, which evolve over hybrid states comprising of the continuous states of the system along with a discrete state denoting the active linear dynamics model.

A hybrid dynamics model is not only a natural

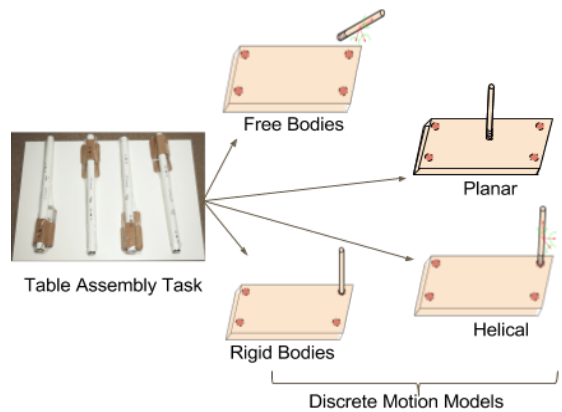

Fig. 1. Decomposition of dynamics model using linear articulated motion models for a furniture assembly task

way to model manipulation task dynamics but it also results in a modularized representation of dynamics. For example, the linear dynamics functions of the hybrid model can be learned over time individually from visual data [1], [2] and can be rearranged and recombined with different functions [2], [3] to develop dynamics models for new tasks and objects.

While representing nonlinear dynamics using hybrid dynamics models can help in developing feasible motion plans for manipulation tasks, having partial or noisy observations as feedback often lead to failure while executing these plans. Also, dynamics models may have modeling errors which can further add to uncertainty over the system states. Hence, reasoning about the uncertainty over states becomes crucial in order to develop robust motion plans.

Planning problems under uncertainty are often modeled as a *partially observable Markov decision process* (POMDP). POMDP problems have been shown in literature to be PSPACE complete [4]. Hence, manipulation planning under uncertainty with nonlinear interaction dynamics can be an intractable problem.

Some current robot manipulation planning under uncertainty approaches [5], [6] sidestep the problem by using model-free methods and find a motion plan by searching for an optimal sequence of parameterized manipulation actions or primitives. However, a lack of task dynamics model restricts these methods from generalizing to novel manipulation tasks and objects. More recent works [7], [8] use deep learning techniques to develop end-to-end control plans directly from vision; however, under sparse availability of training data (especially in robotics), the proposed approaches fail to develop generalized control policies for all system states.

A hybrid dynamics model with linear dynamics functions also helps in simplifying the problem of robot manipulation planning under uncertainty. With a model based on linear dynamics functions at hand, linear filters can be designed to reduce the uncertainty over states even under noisy observations in very few iterations. Also for executing motion plans, belief space regulators (*B-LQR*) [4] can be designed on individual linear dynamics functions to stabilize the trajectory around the planned path.

In this work, we address the POMDP problem of manipulation planning with hybrid dynamics by proposing a Bayesian inference-based hybrid state evolution model and combine it with a POMDP solving algorithm for developing feasible motion plans under partial observability. We develop motion plans in belief space using direct trajectory optimization with an assumption of maximum likelihood observations as true observations and stabilize planned paths by designing a belief space LQR controller. This assumption has been shown to be a fair assumption to make for developing belief space motion plans by Platt et al. [4]. A hybrid Bayesian filter is also introduced to reduce uncertainty over states using continuous state observations. The proposed motion planning algorithm is tested on a benchmark problem of autonomous robot navigation under uncertainty and a manipulation task of partially assembling a toy airplane from the YCB dataset [9].

II. Related Works

Modeling complex nonlinear dynamics of systems using hybrid dynamics models (or switching-mode models) have been popular in various disciplines including the systems and controls community [10], [11], [12], [13], [14]. As hybrid dynamics models can easily encapsulate event-based transition in the system dynamics, they have been extensively used to model systems undergoing significant changes in their dynamics [15], [16]; making them suitable for modeling dynamics involving contacts. Johnson et al. [14] focused on the contact dynamics between objects and derived a hybrid dynamical model of system dynamics based on Lagrangian Dynamics, Newtonian impact laws and complementarity contact conditions. The authors indexed the discrete modes of the hybrid system using an active constraint set for the contacts.

In a closely related work to ours, Hauser et al [11] developed the randomized multi-modal motion planner (Random-MMP) for robot manipulation planning to reason in a motion space having multi-modal structure due to contacts. In another similar work, Jentzcsh et al. [12] defined the manipulation primitives for serial kinematics as different discrete modes and proposed a multi-modal motion planning algorithm (MOPL) based on bi-directional rapidly-exploring random tree (RRT) to develop motion plans. Our work differs from the previous multi-modal motion planning methods [11], [12] in the aspect that it also reasons over state uncertainty while developing motion plans and therefore can be used with noisy observations.

Brunskill et al. [10] proposed a point-based POMDP planning algorithm for solving continuous-state POMDPs based on the hybrid system dynamics. They approximate the complex nonlinear system dynamics using a hybrid multi-modal dynamics model with continuous state-dependent discrete mode switching conditions. However, they place artificial problem-specific constraints on the choice of possible dynamics models for hybrid dynamics model, limiting the scope of generalizability of the approach.

Some other major approaches for manipulation planning are: developing manipulation plans using task and/or motion constraints [17], [18], using *dynamic movement primitives (DMPs)* to represent primitives for different phases (modes) of a multi-phase manipulation task, learning a library of such manipulation skills and sequencing them to perform a novel manipulation task [19], and directly learning a generalized control policy by training a neural network using multiple local time-varying linear quadratic Gaussian (LQG) controllers using policy gradient methods ([7]).

III. Background

A hybrid system is a dynamical system whose states evolve with time over both continuous space $x \in X = \mathbb{R}^N$ and a finite set of discrete states $q \in Q$. Each discrete state of the system corresponds to a separate set of dynamics that governs the evolution of continuous states. Discrete state transitions of the system can be represented as a directed graph with each possible discrete state q corresponding to a node and edges ($e \in E \subseteq Q \times Q$) marking possible transitions between the nodes. In hybrid systems, these transitions are conditioned on the continuous states. A transition from the discrete state q to another state q' happens if the continuous states $\mathbf{x}$ are in the *guard set* $G(q, q')$ of the edge $e_q^{q'}$ where $e_q^{q'} = \{q, q'\}$, $G(\cdot) : E \to P(X)$ and $P(X)$ is the power set of X. Thus, a hybrid system H can be defined as

$$\begin{aligned} x_{t+1} &= f_q(x_t, u_t) \\ z_t &= h_q(x_t) \end{aligned} \tag{1}$$

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$ and $z \in \mathbb{R}^l$ are the continuous state, control input and observation variables, respectively and $q \in Q$ represents the active discrete state of the system. Evolution of the discrete state of the system can be modeled by a finite state Markov chain as

$$\mu_{t+1} = \Pi \mu_t \tag{2}$$

where $\Pi = \{\pi_{ij}\}$ is the discrete state transition matrix and $\mu \in \mathbb{R}^Q$ is the probability distribution over discrete states at time t.

IV. Hybrid Dynamics under Uncertainty

We address the POMDP problem with hybrid systems by defining a belief over the hybrid states of the system and developing a belief evolution model using extended discrete state transition dynamics and a Bayesian state estimator. Two parts of the belief evolution model, prediction and observation based update, are discussed in greater detail in the following sections.

A. Extended Hybrid Belief Dynamics

A belief state over the states of the hybrid system can be defined as $B = \{b^x, b^q\}$, where b^x and b^q correspond to belief distributions over the continuous states x and discrete state q respectively. Transition dynamics for the belief over continuous states can be defined as

$$\begin{aligned} b^x_{t+1} &= p(x_{t+1}|x_t, u_t) \\ &= \sum_{q' \in Q} p(x_{t+1}|x_t, u_t, q_t = q')\ p(q_t = q'|x_t) \end{aligned} \tag{3}$$

where x_{t+1} and x_t represent the continuous states at time $t+1$ and t respectively, while, q_t and u_t represent the discrete state and continuous control input to the system at time t.

Under stochastic continuous state dynamics, the definition of discrete state transition matrix as given in Equation 2 needs to be extended. Assuming the transitions of discrete states are given by a directed graph with self-loops, we can define the extended discrete state transition matrix Π at time t as $\Pi_t = \{\pi_t(i, j) = p(q^j_{t+1}|q^i_t, b^x_{t+1}) \ \ \forall q^i, q^j \in Q\}$ as

$$\begin{aligned} \pi_t(i, j) &= \eta \int_{\mathbb{R}^N} \mathbf{1}^{q^j}_{q^i}(\mathbf{x}) b^x_{t+1}(\mathbf{s}) \mathbf{ds}, \quad if\ \exists\ e^{q^i}_{q^j}, \\ &= \ \epsilon, \qquad\qquad otherwise \end{aligned} \tag{4}$$

where $\mathbf{1}^{q^j}_{q^i}(\mathbf{x})$ is an indicator function defined as

$$\mathbf{1}^{q^j}_{q^i}(\mathbf{x}) = \begin{cases} 1, & \text{if } \mathbf{x} \in G(q^i, q^j) \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

η is a normalization constant, given as $\eta = \sum_{k=1}^{|Q|} \pi(i, k)$ and ϵ is a small probability to handle scenarios in which the received observations do not correspond to any legal discrete transition based on the current belief.

Calculating the extended discrete state transition matrix Π_t at each time step using Eq. 4 can be computationally expensive. An approximation of Π_t can be obtained by sampling n random points from the belief over continuous states b^x_{t+1} and calculating ratio of points lying in the guard set $G(q^i, q^j)$ to the total number of sampled points for each discrete state q^j.

B. Belief Evolution Model: Prediction

At each time step t, a belief over the continuous states is first obtained using Equation 3. If we assume that the belief over the continuous states b^x is a normal distribution, we can propagate the current belief b^x_t through the system dynamics of each discrete state, $\mathcal{F}^{q'}(x_t, u_t)$, individually. A weighted sum of the propagated belief set can then be taken to obtain a prior of the belief at next time step $\hat{b}^x_{t+1}$

$$\hat{b}^x_{t+1} = \sum_{q'} \mathcal{F}^{q'}(b^x_t, u_t)\ b^{q_{q'}}_t \tag{6}$$

where $b^{q_{q'}}_t = p(q_t = q'|x_t)$ is q'-th component of b^q_t. A normal distribution is fitted on the obtained mixture of Gaussians to maintain the normal distribution representation of belief b^x .

Since the belief over discrete state $\hat{b}^q_{t+1}$ is given by a discrete distribution, a prior state estimate is calculated using Equation 4 where the mode transition matrix is updated using the continuous state prediction obtained in the last step, $\hat{b}^x_{t+1}$.

C. Belief Evolution Model: Update

We use a hybrid estimation algorithm based on Bayesian filters to reduce the uncertainty over states using noisy continuous state observations. The proposed algorithm consists of two layers of filters: first to estimate the continuous states of the system and second to estimate the discrete states of the system. Upon receiving observation z_{t+1}, the continuous state prior is updated by taking a weighted sum of a bank of extended Kalman filters running independently, with each discrete mode having an individual filter. The weights for the sum is determined using the prior for the discrete mode $\hat{b}^q_{t+1}$. The complete update step for continuous states can be written as

$$b^x_{t+1} = \hat{b}^x_{t+1} + \sum_{q'} \Big(\mathbf{K}^{q'}_{t+1}(z_{t+1} - \mathcal{H}^{q'}_{t+1}(\hat{b}^x_{t+1})) \Big) \hat{b}^{q_{q'}}_{t+1} \tag{7}$$

where $\mathbf{K}^{q'}_{t+1}$ is the Kalman Gain for discrete state q' at time $t+1$ and $\hat{b}^{q_{q'}}_{t+1}$ is q'-th component of $\hat{b}^q_{t+1}$. Update for the discrete state can be obtained by using a Bayesian filter update given as

$$b^q_{t+1} = \gamma \mathbf{M}_{t+1} \circ \hat{b}^q_{t+1} \tag{8}$$

where $\mathbf{M}_{t+1} = [P(z_{t+1}|q_{t+1} = q')]^T \quad \forall q' \in Q$, $\circ$ is the element-wise multiplication operator, γ is a normalization constant and

$$P(z_{t+1}|q_{t+1} = q') = z_{t+1} \sim \mathcal{H}^{q'}_{t+1}(b^x_{t+1})$$

where $\mathcal{H}^{q'}_{t+1}(.)$ is the observation function for state q'.

V. Motion Planning in Belief Space

The proposed belief evolution model for hybrid dynamics can be used to develop motion plans in belief space. Belief space motion plans are developed using a two stage algorithm: construct an initial plan using direct trajectory optimization and stabilize around it by using a belief space linear quadratic Gaussian (LQG) controller [4]. We use direct transcription as the direct trajectory optimization method with sequential least squares programming (SLSQP) as the optimization algorithm. In the planning phase, we assume *maximum likely observations (MLO)* obtained by propagating the current belief over continuous states through the system dynamics (6) as true observations. However, while stabilizing the trajectory the most likely discrete state is taken as the governing dynamics for the system.

Under high uncertainty over continuous states, a probabilistic distribution over discrete states accounts for the ambiguity in the active dynamics model for continuous states at time t. However, as the uncertainty over continuous states goes down, a parameter σ_0 can be chosen, below which this distribution can be replaced by a deterministic discrete state selection approach. Under the deterministic approach, belief over discrete states b^q is defined with current active dynamics model having full probability mass. The complete motion planning algorithm is shown in Algorithm 1.

```
Input  : Initial belief state, B = {b^x, b^q, goal, Θ}
Output: vector of control actions u_{1:T}
while b^x not at goal do
    (ū_{1:T}, b̄^x_{1:T}, b̄^q_{1:T}) = create_plan(b^x, b^q)
    for t ← 1 to T − 1 do
        u_t = LQR_stabilize(b^x_t, b^q_t, ū_t, b̄^x_t, b̄^q_t);
        (b^x_{t+1}, b^q_{t+1}) = propagate_belief(b^x_t, b^q_t, u_t,
          z);
        if (b^x_{t+1} − b̂^x_{t+1}) > θ then
            break;
        end
    end
    b^x = b^x_{t+1};
    b^q = b^q_{t+1};
end
```

Algorithm 1: Motion Planning for Hybrid Belief

```
Function create_plan(b^x, b^q)
    optimize_trajectory(u_{1:T}, b^x_{1:T}, b^q_{1:T})
    subject to:
    Hybrid belief dynamics, given as:
    for t ← 1 to T − 1 do
        if cov(b^x_t) > σ_0 then
            b^x_{t+1}, b^q_{t+1} =
              stochastic_belief_propagation(b^x_t,
              b^q_t, u_t, z_{t+1});
            Eqns: [4, 6, 7, 8]
        else
            b^x_{t+1}, b^q_{t+1} =
              deterministic_belief_propagation(b^x_t,
              b^q_t, u_t, z_{t+1});
            Eqns: [4, 6, 7] with deterministic b^q
        end
        where, z = Maximum_Likely_Observation;
    end
    return u_{1:T}, b^x_{1:T}, b^q_{1:T};
```

Algorithm 2: *create_plan* function

VI. Experiments

Experiments were conducted to test the proposed belief evolution model and motion planner on a benchmark problem of autonomous navigation under uncertainty and a toy airplane assembly manipulation task with noisy observations.

A. Benchmark-I: Reach-the-goal Task

The reach-the-goal task aims at navigating a mobile robot in a light-dark domain [4] where the quality of observations improves proportionally with the amount of light. The task objective is to reach the goal with minimum uncertainty over its states. To check the performance of our motion planning algorithm under state-dependent dynamics, we extend the domain to have spatially varying dynamics; this is analogous to a real world scenario of off-road terrain with different surfaces having different ground friction. The 2-D domain ($\{x, y\} \in [-10, 10]$) was considered to contain three different linear dynamics functions, given as

$$
\begin{aligned}
f(\mathbf{x}_t, \mathbf{u}) &= \mathbf{x}_t + 0.5\mathbf{u}, \quad && if \;\; x < -1 \\
f(\mathbf{x}_t, \mathbf{u}) &= \mathbf{x}_t + \mathbf{u}, \quad && if \;\; x \in [-1, 4] \\
f(\mathbf{x}_t, \mathbf{u}) &= \mathbf{x}_t + [2\mathbf{u_1},\ \mathbf{u_2}]^T, \quad && if \;\; x > 4
\end{aligned}
$$

where $\mathbf{x}_t = \{x_t, y_t\}^T$. Belief over continuous states was considered to be a Gaussian distribution $b^x = \{\mu, \sigma\}$. Observation function was taken as $h(x_t) = x_t + w$ with zero-mean Gaussian observation noise $w \sim \mathcal{N}(\cdot|0, W(x))$ where

$$W(x) = \frac{1}{2}(5 - x_x)^2 + const$$

Matrices defining the cost function over error in states, control input, additional cost for final state error and covariance were taken as $Q = diag(0.5, 0.5)$, $R = diag(0.5, 0.5)$, $Q_{large} = 30$ and $\Lambda = 400$ respectively. We compare the performance of our algorithm in two modes: (1) a totally deterministic approach that assumes most-likely discrete state to be the governing dynamics mode and (2) the proposed hybrid approach. It can be seen from Figure 2 that if the uncertainty over continuous states is high, having a completely deterministic approach over active discrete states while planning can cause the belief to diverge completely from the actual robot state. Hence, it is critical to start the planning problem with a stochastic approach over active discrete states to develop meaningful belief space plans.

B. Toy Airplane Assembly Task

The objective of the toy airplane assembly task is to partially assemble the toy airplane from the YCB dataset [9] (shown in Figure 3). We consider the

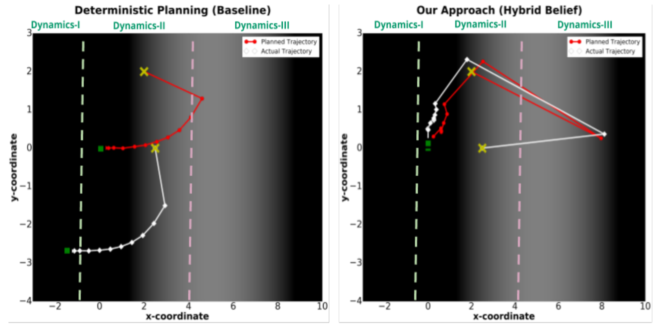

Fig. 2. Plots showing planned and actual robot trajectories under deterministic and hybrid planning modes. Initial belief mean $\mu = \{2, 2\}$, $cov = diag(5.0, 5.0)$ (yellow cross on red trajectory), True start position:={2.5,0, 0} (yellow cross on white trajectory), goal position:={0,0} (shown by green square). Brightness reflects quality of observation. Dashed lines separate different dynamics modes

first step of inserting the landing gear into one of the wings as a test case for our planner. A hybrid dynamics model with three continuous states $\mathbf{x}^{gear} = \{x^{gear}, y^{gear}, z^{gear}\}$, can be given as:

- *Planar Motion*: $f(\mathbf{x}_t^{gear}, \mathbf{u_t}) = \mathbf{x}_t^{gear} + [u_x, u_y, 0]^T$, if $z^{gear} \in$ surface of the wing
- *Inserting*: $f(\mathbf{x}_t^{gear}, \mathbf{u_t}) = \mathbf{x}_t^{gear} + [0, 0, u_z]^T$, if $\{x^{gear}, y^{gear}\} = \{x^{hole}, y^{hole}\}$
- *Free Body Motion*: $f(\mathbf{x}_t^{gear}, \mathbf{u_t}) = \mathbf{x_t^{gear}} + \mathbf{u}$, otherwise

Experiments were conducted using a bi-manual manipulator robot Gemini with two Kinova $Jaco^2$ 7-dof arms. The airplane wing was held fixed at a spot using the left arm of the robot, while the planner planned the workspace trajectory for the right arm holding the landing gear. Knowledge of hybrid dynamics while planning becomes crucial in such a scenario involving closed chain interactions; as devoid of this knowledge the planner may plan a path that involves applying very high forces from the tip of the landing gear on the wing so as to reduce its position uncertainty, which can lead to mechanical failures in either of the arms.

The goal for the planner was defined at a point with an offset of $24.6cm$ from the left arm end-effector

Fig. 3. Toy airplane: assembled (*on left*) and its parts

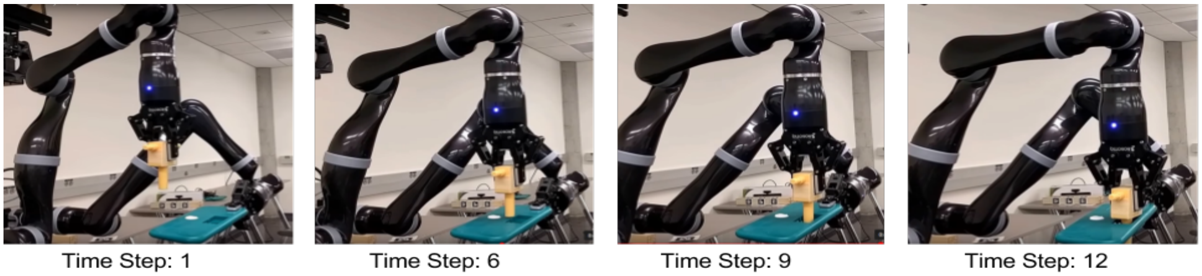

Fig. 4. Snapshots of Gemini assembling the toy airplane

along an axis pointing outwards from it. Observations were obtained by offsetting the joint feedback by the same amount. In the planner, the current state of the landing gear was defined relative to the goal location. As the end-effector gripping point on the plane wing need not to be exactly the same during each experiment, the defined goal did not necessarily correspond to the physical center of the hole in the wing, which resulted in the observation noise.

For the planner, the observation function was modeled as $h(x_t) = x_t + w$ with zero-mean Gaussian observation noise with a covariance of 5 cm. Belief over continuous states was considered to be a Gaussian distribution with an initial uncertainty of 2 $meter$ over states. Matrices defining the cost function over error in states, control input and additional cost for final state error were taken as $Q = diag(0.5, 0.5)$, $R = diag(0.1, 0.1)$, $Q_{large} = 5000$ respectively. The obstacle costmap was defined by calculating the Euclidean distance transform (EDT) on an occupancy grid of the workspace[20]. A smooth obstacle cost function was defined as:

$$c(x) = \begin{cases} \Lambda \ d(x), & \text{if } d(x) \in [\epsilon_{max}, 1] \\ \Lambda \ \epsilon_{max} \frac{(d(x)-\epsilon_{thres})^2}{(\epsilon_{max}-\epsilon_{thres})^2}, & \text{if } d(x) \in [\epsilon_{thres}, \epsilon_{max}) \\ 0.0, & \text{otherwise} \end{cases} \tag{9}$$

where $d(x)$ is the EDT at x [20]. We set $\epsilon_{max} = 0.95$, $\epsilon_{thres} = 0.7$ and $\Lambda = 10^{10}$. At each time step, the planned target for gear x^{gear}_{t+1} is treated as a workspace target for the arm and converted to corresponding joint space target using inverse kinematics.

To make sure that the planner searched preferentially in the feasible joint space of arm, an additional penalty term for workspace target with infeasible arm poses, was also added in the planning objective cost. For experimental considerations, the plane wing was approximated using a cuboid with dimensions $\{x, y, z\} = \{15cm, 30cm, 10cm\}$ while planning.

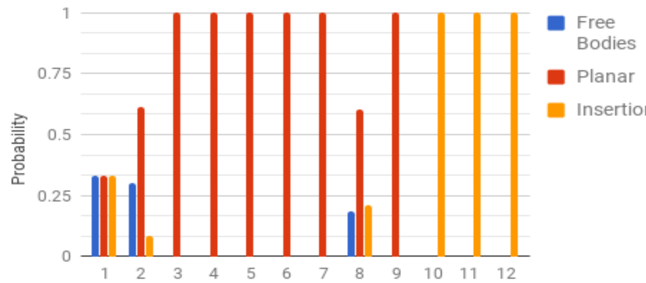

Fig. 5. Propagation of belief over discrete states b^q with time

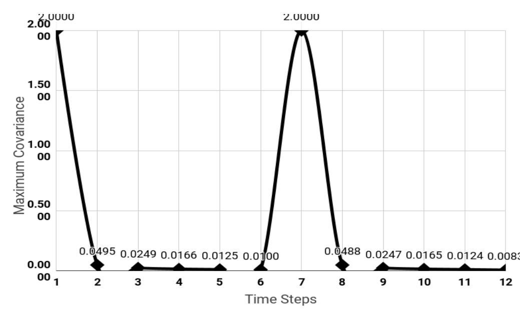

Fig. 6. Propagation of covariance over continuous states with time

Snapshots of the robot performing assembly task are shown in Figure 4. Figure 5 shows the propagation of planner's belief over discrete states. The parameter for stochastic to deterministic discrete belief transition σ_0, was chosen to be 0.040. This corresponds to a maximum uncertainty of our planner over continuous states to be 4 cm. This results in a sudden change in weights over discrete states between time steps 9 and 10 in Figure 5. Figure 6 shows the propagation of our planner's uncertainty over continuous states. It can be seen from the plot that the uncertainty goes down very rapidly within a couple of steps. To test robustness of the planner further, the planner's uncertainty over continuous states was artificial increased to its initial value of 2 $meter$ at time step 7. A rapid convergence in uncertainty to a value below 5 cm is observed in the subsequent steps, which shows the effectiveness of the proposed hybrid filter.

VII. CONCLUSION

This work aims to address two major difficulties in robot manipulation tasks: highly nonlinear task dynamics and partial observability of states. Tasks with nonlinear dynamics, especially involving contacts, can be naturally decomposed into simpler linear dynamics models, and such models can be combined to give a complete dynamics model using hybrid dynamics. With a hybrid dynamics model, the robot manipulation planning problem under uncertainty is addressed as a POMDP problem and a novel POMDP planner is proposed that can handle hybrid states. A Bayesian-inference based hybrid state estimator is also introduced to reduce the uncertainty over states using the received observations.

In future, the proposed POMDP planner will be used to develop an online method for learning dynamics model for manipulation tasks [2], [21] and developing feasible motion plans based on the learned dynamics model. Also, by rearranging and combining the modular linear dynamics models, approximate dynamics models for novel manipulation tasks will be learned. These approximate models can then be used to plan for these new tasks under partial observability.

REFERENCES

[1] J. Sturm, C. Stachniss, and W. Burgard, "A probabilistic framework for learning kinematic models of articulated objects," *Journal of Artificial Intelligence Research*, vol. 41, pp. 477–526, 2011.

[2] S. Niekum, S. Osentoski, C. G. Atkeson, and A. G. Barto, "Champ: Changepoint detection using approximate model parameters," DTIC Document, Tech. Rep., 2014.

[3] K. Hausman, S. Niekum, S. Osentoski, and G. Sukhatme, "Active articulation model estimation through interactive perception," in *IEEE International Conference on Robotics and Automation*, 2015.

[4] R. P. Jr, R. Tedrake, L. Kaelbling, and T. Lozano-Perez, "Belief space planning assuming maximum likelihood observations," *Robotics: Science and Systems*, 2010.

[5] M. R. Dogar and S. S. Srinivasa, "A planning framework for non-prehensile manipulation under clutter and uncertainty," *Autonomous Robots*, vol. 33, no. 3, pp. 217–236, 2012.

[6] T. Lozano-Pérez and L. P. Kaelbling, "A constraint-based method for solving sequential manipulation planning problems," in *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*. IEEE, 2014, pp. 3684–3691.

[7] S. Levine, N. Wagener, and P. Abbeel, "Learning contact-rich manipulation skills with guided policy search," in *2015 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2015, pp. 156–163.

[8] J. Fu, S. Levine, and P. Abbeel, "One-shot learning of manipulation skills with online dynamics adaptation and neural network priors," in *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*. IEEE, 2016, pp. 4019–4026.

[9] B. Calli, A. Singh, J. Bruce, A. Walsman, K. Konolige, S. Srinivasa, P. Abbeel, and A. M. Dollar, "Yale-cmu-berkeley dataset for robotic manipulation research," *The International Journal of Robotics Research*, vol. 36, no. 3, pp. 261–268, 2017.

[10] E. Brunskill, L. Kaelbling, T. Lozano-Perez, and N. Roy, "Continuous-State POMDPs with Hybrid Dynamics," *Symposium on Artificial Intelligence and Mathematics*, pp. 13–18, 2008.

[11] K. Hauser and V. Ng-Thow-Hing, "Randomized multi-modal motion planning for a humanoid robot manipulation task," *The International Journal of Robotics Research*, vol. 30, no. 6, pp. 678–698, 2011.

[12] S. Jentzsch, A. Gaschler, O. Khatib, and A. Knoll, "Mopl: A multi-modal path planner for generic manipulation tasks," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*. IEEE, 2015, pp. 6208–6214.

[13] A. Garulli, S. Paoletti, and A. Vicino, "A survey on switched and piecewise affine system identification," *IFAC Proceedings Volumes*, vol. 45, no. 16, pp. 344–355, 2012.

[14] A. M. Johnson, S. A. Burden, and D. E. Koditschek, "A hybrid systems model for simple manipulation and self-manipulation systems," *The International Journal of Robotics Research*, p. 0278364916639380, 2016.

[15] E. Frazzoli, "Robust hybrid control for autonomous vehicle motion planning," Ph.D. dissertation, Citeseer, 2001.

[16] E. R. Westervelt, J. W. Grizzle, and D. E. Koditschek, "Hybrid zero dynamics of planar biped walkers," *IEEE transactions on automatic control*, vol. 48, no. 1, pp. 42–56, 2003.

[17] D. Berenson, S. S. Srinivasa, D. Ferguson, and J. J. Kuffner, "Manipulation planning on constraint manifolds," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*. IEEE, 2009, pp. 625–632.

[18] C. R. Garrett, T. Lozano-Pérez, and L. P. Kaelbling, "Backward-forward search for manipulation planning," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*. IEEE, 2015, pp. 6366–6373.

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

[20] N. Ratliff, M. Zucker, J. A. Bagnell, and S. Srinivasa, "Chomp: Gradient optimization techniques for efficient motion planning," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*. IEEE, 2009, pp. 489–494.

[21] S. Niekum, S. Osentoski, C. G. Atkeson, and A. G. Barto, "Online bayesian changepoint detection for articulated motion models," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 1468–1475.