Joint Inference of Kinematic and Force Trajectories with Visuo-Tactile Sensing

Alexander (Sasha) Lambert$^{1,3}$, Mustafa Mukadam$^{1,3}$, Balakumar Sundaralingam$^{2,3}$, Nathan Ratliff$^3$, Byron Boots$^{1,3}$, and Dieter Fox$^{3,4}$

Abstract—To perform complex tasks, robots must be able to interact with and manipulate their surroundings. One of the key challenges in accomplishing this is robust state estimation during physical interactions, where the state involves not only the robot and the object being manipulated, but also the state of the contact itself. In this work, within the context of planar pushing, we extend previous inference-based approaches to state estimation in several ways. We estimate the robot, object, and the contact state on multiple manipulation platforms configured with a vision-based articulated model tracker, and either a biomimetic tactile sensor or a force-torque sensor. We show how to fuse raw measurements from the tracker and tactile sensors to jointly estimate the trajectory of the kinematic states and the forces in the system via probabilistic inference on factor graphs, in both batch and incremental settings. We perform several benchmarks with our framework and show how performance is affected by incorporating various geometric and physics based constraints, occluding vision sensors, or injecting noise in tactile sensors. We also compare with prior work on multiple datasets and demonstrate that our approach can effectively optimize over multi-modal sensor data and reduce uncertainty to find better state estimates.

I. INTRODUCTION & RELATED WORK

Manipulation is a difficult problem, complicated by the challenge of robustly estimating the state of the robot’s interaction with the environment. Parameters such as the contact point and the force vector applied at that point, can be very hard to robustly estimate. These parameters are generally partially observable and must be inferred from noisy information obtained via coarse visual or depth sensors and highly sensitive but difficult to interpret tactile sensors.

For instance, in the case of “in-hand” manipulation problems, where a held object is often partially occluded by an end-effector, tactile sensing offers an additional modality that can be exploited to estimate the pose of the object [1].

Vision and tactile sensors have been used to localize an object within a grasp using a gradient-based optimization approach [2]. This has been extended to incorporate constraints like signed-distance field penalties and kinematic priors [1]. However, the former is deterministic and the latter handles uncertainty only per time-step, which is insufficient since sensors can be highly noisy and sensitive. Particle filtering-based approaches have been proposed that can infer the latent belief state from bi-modal and noisy sensory data, to estimate the object pose for two-dimensional grasps [3] and online localization of a grasped object [4]. These approaches are often limited in scope. For example, [4] uses vision to only initialize the object pose and later relies purely on contact information and dynamics models. In general, particle filtering based methods also suffer from practical limitations like computational complexity, mode collapse, and particle depletions in tightly constrained state spaces.

Beyond manipulation, state estimation is a classic problem in robotics. For example, Simultaneous Localization and Mapping (SLAM) has been studied for many decades, and many efficient tools have been developed to address noisy multi-modal sensor fusion in these domains [5]–[7]. One of the more successful tools, the smoothing and mapping (SAM) framework [7], uses factor graphs to perform inference and exploits the underlying sparsity of the estimation problem to efficiently find locally optimal distributions of latent state variables over temporal sequences. This technique offers the desired combination of being computationally fast while accounting for uncertainty over time, and has been recently incorporated into motion planning [8], [9].

This framework has also been explored for estimation during manipulation [10]–[12]. In particular, Yu et al. [11] formulate a factor graph of planar pushing interaction (for non-prehensile and underactuated object manipulation) using a simplified dynamics model, with both visual object-pose and force-torque measurements and show improved pose
recovery over trajectory histories compared to single-step filtering techniques. However, the scope of [11] is limited to the use of a purpose-built system, equipped with small-diameter pushing-rods kept at a vertical orientation, allowing for high-fidelity contact-point estimation. A fiducial-based tracking system is also used. Such high precision measurements are impractical in a realistic setting.

In this work, we extend the capabilities of such factor graph inference frameworks in several ways to perform planar pushing tasks in real world settings. We extend the representation to incorporate various geometric and physics-based constraints alongside multi-modal information from vision and tactile sensors. We perform ablation benchmarks to show the benefits of including such constraints, and benchmarks where the vision is occluded or the tactile sensors are very noisy, using data from on our own generalized systems.

We conduct our tests on two systems, a dual-arm ABB Yumi manipulator equipped a gel-based Syntouch Biotac tactile sensor [13] and a Barrett WAM arm equipped with a pushing probe end effector mounted with a force torque sensor (see Fig[1]). Both of these systems are also set up with a vision-based articulated tracking system that leverages a depth camera, joint encoders, and contact-point estimates [1].

Through inference, we jointly estimate the history of not only object poses, and end-effector poses, but also, contact points, and applied force vectors. Estimating contact points and applied force vectors can be very useful in tractable dynamics models to predict future states and can be beneficial to contact-rich planning and control for manipulation [14].

With our experiments, we show that we can contend with a range of multi-modal noisy sensor data and perform efficient inference in batch and incremental settings to provide high-fidelity and consistent state estimates.

II. DYNAMICS OF PLANAR PUSHING

In this section, we review the dynamics model for pushing on planar surfaces. The quasi-static approximation of this model is used in the next section to describe the motion model of the pushed object within the factor graph for estimation.

Given an object of mass $m$ being pushed with an applied force $f$, we can describe the planar dynamics of the rigid body through the primary equations of motion

$$ f + f_{\mu} = m\ddot{x}_{CM}, \quad \tau + \tau_{\mu} = I_{CM}\dot{\omega} \quad (1) $$

where $x_{CM}$ is the object position measured at the center-of-mass (CM), $\omega$ the angular velocity of the object frame, $I_{CM}$ the moment of inertia, and $f_{\mu}$ the linear frictional force. The applied and frictional moments are defined as $\tau = \dot{x}_{CM} \times f$ and $\tau_{\mu} = x_{CM} \times f_{\mu}$ respectively.

We can estimate the frictional loads on the object by considering the contribution of each point on the support area $A$ of the object [10]. The friction force $f_{\mu}$ and corresponding moment $\tau_{\mu}$ is found by integrating Coulomb’s law across the contact region of the object with the surface

$$ f_{\mu} = -\mu_{s}\int_{A} \frac{v(r)}{|v(r)|} P(r) dA, \quad \tau_{\mu} = -\mu_{s}\int_{A} r \times \frac{v(r)}{|v(r)|} P(r) dA \quad (2) $$

where $v(r)$ denotes the linear velocity at a point $r$ in area $A$, and $P(r)$ the pressure distribution. The coefficient of friction is assumed to be uniform across the support area.

For pusher trajectories that are executed at near-constant speeds, inertial forces can be considered negligible. The push is then said to be quasi-static, where the applied force is large enough to overcome friction and maintain a velocity, but is insufficient to impart an acceleration [15]. Then, the applied force $f$ must lie on the limit surface. This surface is defined in $(f_x, f_y, \tau)$ space and encloses all loads under which the object would remain stationary [16]. It can be approximated as an ellipsoid with principal semi-axes $f_{max}$ and $\tau_{max}$ [17]

$$ \left(\frac{f_x}{f_{max}}\right)^2 + \left(\frac{f_y}{f_{max}}\right)^2 + \left(\frac{\tau}{\tau_{max}}\right)^2 = 1 \quad (3) $$

where $f_{max} = \mu_{s}f_n$, and $f_n$ is the normal force. In order to calculate $\tau_{max}$, we assume a uniform pressure distribution and define $r$ with respect to the center of mass $(r = r_{CM})$:

$$ \tau_{max} = -\mu_{s}\mu_{F} \int_{A} r_{CM} |dA| $$

For quasi-static pushing, the velocity is aligned with the frictional load, and therefore must be parallel to the normal of the limit surface. This results in the following constraints on the object motion

$$ \frac{v_x}{\omega} = c^2 \frac{f_x}{\tau}, \quad \frac{v_y}{\omega} = c^2 \frac{f_y}{\tau}, \quad c = \frac{\tau_{max}}{f_{max}} \quad (4) $$

used within our estimation factor graph in the next section.

III. STATE ESTIMATION WITH FACTOR GRAPHS

To solve state estimation during manipulation we formulate a factor graph of belief distributions over any state and force vector trajectory and perform inference over the trajectory given noisy sensor measurements. The graph construction and inference is performed with GTSAM [7], [18] via sparsity exploiting nonlinear least squares optimization to find the maximum a posteriori (MAP) trajectory that satisfies all the constraints and measurements. In the batch setting we use a Gauss-Newton optimizer and in an incremental setting we use iSAM2 that performs incremental inference via Bayes trees [19]. All random variables and measurements are assumed to have a Gaussian distribution. In the remainder of this section, we describe the construction of the relevant factor graphs depicted in Fig[2].

A. Model Design

We construct three different factor graphs for state estimation in our pushing task: CP, SDF, and QS (see Fig[2]). All three models include the latent state variables for a given time $t$: the planar object pose $x_t \in SE(2)$, the projected end-effector pose $e_t \in SE(2)$, and the contact point $p_t \in \mathbb{R}^2$.

**Measurements:** Each of the latent state variable is accompanied by an associated measurement factor $M$ which projects corresponding measurements from $SE(3)$ into the pushing plane. The object poses are estimated by the visual tracking system with measurements $y_t \in SE(3)$. Likewise, the end-effector pose measurements $e_t \in SE(3)$ may be provided from robot forward kinematics, or from the tracking system (DART includes a prior on joint measurements). The contact-point measurements $w_t \in SE(3)$ are provided by a tactile
sensor model. In the QS graph (Fig. 2c), we include a new state variable for the applied planar contact force \( \mathbf{f}_c \in \mathbb{R}^2 \) with corresponding measurements \( \alpha_t \in \mathbb{R}^3 \). For simplicity of graphical representation, we combine the contact point and force variables:

\[
\begin{align*}
\hat{\mathbf{p}}_t &= \begin{bmatrix} \mathbf{p}_t \end{bmatrix}, \\
\hat{\mathbf{w}}_t &= \begin{bmatrix} \mathbf{w}_t \end{bmatrix}, \\
\end{align*}
\]

Geometric Constraints: We assume constant point-contact between the end-effector and the object. We include the factor \( \mathbf{C} \) which incurs a cost on the difference between the contact point \( \mathbf{p}_t \) and the closest point to a surface (\( \xi \)):

\[
\mathbf{C}(\xi, \mathbf{p}_t) = \mathbf{G}(\xi, \mathbf{p}_t) - \mathbf{p}_t
\]

where \( \mathbf{G}(\xi, \mathbf{p}_t) \) is the projection of \( \mathbf{p}_t \) onto \( \xi \), and \( \xi = \xi(\mathbf{x}) \) returns the surface geometry of a body with a given pose: \( \xi = \xi(\mathbf{x}_t) \) for the object, and \( \xi = \xi(\mathbf{e}_t) \) for the end-effector. Additionally, the object and the end-effector must be prevented from occupying the same region in space. Such a constraint is necessary in practice where contact-point estimation is often noisy. Therefore, we introduce a factor \( \mathbf{S} \) to penalize intersecting geometries with a signed distance field. Let the point on the end-effector furthest into the object be denoted by \( \delta \in \mathbb{R}^2 \), where \( \delta = \delta(\mathbf{x}, \xi(\mathbf{e})) \). The projection of \( \delta \) onto \( \xi(\mathbf{x}_t) \) (the surface of the object) is then defined by \( G_\delta = G(\xi, \mathbf{x}_t) \), and we can apply a penalty

\[
\mathbf{S}(\mathbf{x}, \mathbf{e}) = \begin{cases} 
G_\delta - \delta, & \text{if intersecting} \\
0, & \text{otherwise}
\end{cases}
\]

Dynamics: We add a constant velocity prior \( V \) to impose smoothness on state transitions. For example, for finite-difference velocities of object poses we have

\[
V(\mathbf{x}_{t-1}, \mathbf{x}_t, \mathbf{x}_{t+1}) = \frac{\mathbf{x}_t - \mathbf{x}_{t-1}}{\Delta t} - \frac{\mathbf{x}_{t+1} - \mathbf{x}_t}{\Delta t + 1}
\]

where \( \Delta t \) and \( \Delta t + 1 \) denote the timestep sizes at \( t \) and \( t + 1 \).

Similar to [11], we introduce an additional factor \( \mathbf{D} \) to condition object state transitions on quasi-static pushing. The corresponding graphical model is denoted by QS and is shown in Fig. 2c. From Eq. 4 we get

\[
\mathbf{D}(\mathbf{x}_{t-1}, \mathbf{x}_t, \hat{\mathbf{p}}_t) = \begin{bmatrix} \mathbf{v}_t \\
- \mathbf{c}^2 \mathbf{f}_c / \tau_c
\end{bmatrix}
\]

where \( \mathbf{v}_t = (x_{\text{trans},t} - x_{\text{trans},t-1}) / \Delta t \) and \( \mathbf{c} = (x_{\text{rot},t} - x_{\text{rot},t-1}) / \Delta \) are the finite-difference linear and angular velocity, respectively. The final cost function is optimized with respect to the set of variables \( \Phi = \{(\mathbf{x}, \mathbf{e}, \hat{\mathbf{p}})\}_{t=1}^{T} \) over a trajectory of length \( T \):

\[
\Phi^* = \arg \min_{\Phi} \sum_{t=1}^{T} \left\{ \|D(\mathbf{x}_{t-1}, \mathbf{x}_t, \hat{\mathbf{p}}_t)\|_{\mathbf{S}^2} + \|V(\mathbf{x}_{t-1}, \mathbf{x}_t, \mathbf{x}_{t+1})\|_{\mathbf{S}^2} \right. \\
+ \|\mathbf{C}(\mathbf{p}_t, \mathbf{e}_t)\|_{\mathbf{S}^2} + \|\mathbf{C}(\mathbf{p}_t, \mathbf{e}_t)\|_{\mathbf{S}^2} + \|\mathbf{C}(\mathbf{p}_t, \mathbf{e}_t)\|_{\mathbf{S}^2} \left. \right\} \\
+ \|\mathbf{M}(\mathbf{e}_t, \mathbf{z}_t)\|_{\mathbf{S}^2} + \|\mathbf{M}(\mathbf{e}_t, \mathbf{z}_t)\|_{\mathbf{S}^2}
\]

The above equation provides the locally optimal i.e. MAP solution of the estimation problem.

IV. BASELINE COMPARISON

In order to first ascertain the general performance of our approach, we evaluate the QS-graph on the MIT planar pushing dataset [20] using batch optimization. This data contains a variety of pushing trajectories for a single-point robotic pushing system. The object poses were tracked with a motion capture system, and contact forces were measured with a pushing probe mounted on a force-torque sensor. We use this data as ground truth, since it is considered to be sufficiently reliable. We restrict our experiments to a subset of this data, using trajectories with zero pushing acceleration and velocities under 10 cm/s in order to maintain approximately quasi-static conditions. Additionally, we only consider trajectories on the ABS surface, but examine different object types (ellip1, rect1, rect3) with approximately 100 trajectories per object and measurements provided at 100Hz. Gaussian noise is artificially added to the measurements prior to inference, with the following sigma values: \( \sigma_{\text{trans}} = 0.5 \text{cm}, \sigma_{\text{rot}} = 0.5 \text{rad}, \sigma_{\text{trans}} = 0.5 \text{cm}, \sigma_{\text{rot}} = 0.5 \text{rad}, \sigma_{\text{p}} = 0.5 \text{cm}, \sigma_{\text{r}} = 0.5 \text{N} \).

The resulting RMS and covariance values post-optimization are shown in Table 1. The optimized values exhibit marked reductions in error compared to the sigma values of the initial measurements. Note that, for object poses we only include values in which the object is in motion, in order to exclude trivial stationary estimates. All position-related values are in cm, with angular values in radians, and forces in Newtons. An example of an optimized trajectory is shown in Fig. 3. Although the observation noise is artificial, these results indicate that latent state estimates may still be successfully recovered with the
TABLE I: RMS and Covariance values on the MIT Dataset.

| Object | RMS($x_{trans}$) | RMS($x_{rot}$) | Σ(x$_{trans}$) | Σ(x$_{rot}$) |
|--------|------------------|----------------|----------------|----------------|
| ellip1 | 0.0262           | 0.283          | 2.723e-4       | 4.171e-10     |
| rect1  | 0.0253           | 3.471e-5       | 2.931e-4       | 4.19e-10      |
| rect3  | 0.0182           | 1.672e-5       | 2.563e-4       | 4.18e-10      |

| Object | RMS($y_{trans}$) | RMS($y_{rot}$) | Σ(y$_{trans}$) | Σ(y$_{rot}$) |
|--------|------------------|----------------|----------------|----------------|
| ellip1 | 7.75e-2          | 9.47e-2        | 4.74e-3        | 7.11e-3        |
| rect1  | 8.59e-2          | 9.18e-2        | 5.89e-3        | 6.01e-3        |
| rect3  | 0.372            | 0.376          | 0.148          | 0.154          |

| Object | RMS($f_{trans}$) | RMS($f_{rot}$) | Σ(f$_{trans}$) | Σ(f$_{rot}$) |
|--------|------------------|----------------|----------------|----------------|
| ellip1 | 1.18             | 9.54e-2        | 9.827e-3       | 1.635e-4       |
| rect1  | 0.145            | 9.683e-2       | 9.862e-3       | 1.823e-4       |
| rect3  | 0.113            | 9.754e-2       | 9.145e-3       | 1.856e-4       |

| Object | RMS($x_{trans}$) | RMS($x_{rot}$) | Σ(x$_{trans}$) | Σ(x$_{rot}$) |
|--------|------------------|----------------|----------------|----------------|
| ellip1 | 3.42e-2          | —              | 2.54e-3        | —              |
| rect1  | 4.52e-2          | —              | 6.21e-3        | —              |
| rect3  | 3.26e-2          | —              | 3.41e-3        | —              |

Fig. 3: Example of performing the inference on a trajectory from the MIT pushing dataset, using the QS graph. Noise is artificially added to measurements prior to smoothing. Two-sigma contours and force vectors are displayed at every 15th time-step for visual clarity.

addition of geometric and physics-based priors, and without over-constraining the optimization.

V. STATE ESTIMATION IN OPEN AND CLUTTERED SCENES

We first perform pushing experiments with the Barrett-WAM manipulator acting on a laminated box as shown in Fig. 4. The system is observed by a stationary PrimeSense depth camera located 2.0m away from the starting push position of the end-effector. Vision-based tracking measurements of the object pose are provided by DART, configured with contact-based priors and joint estimates [1]. The robot is equipped with a Force-Torque sensor and a rigid end-effector mounted with a spherical hard-plastic pushing probe. The contact forces are measured by the F/T sensor, with contact point measurements provided through optimization in DART. Ground-truth poses are provided via a motion-capture system. The table is mounted with a smooth delrin sheet to provide approximately uniform friction across the pushing area.

We performed 100 pushing trials with varying initial end-effector and object poses. The end-effector trajectories were varied in curvature and maintained a translational speed close to 6 cm/s to approximate quasi-static conditions. Object pose-tracking measurements were provided at roughly 25Hz, with end-effector poses and force/contact measurements published at 250Hz. Incremental inference of the factor graph is performed after 5 object pose measurements.

In order to provide real-time performance, DART maintains a belief distribution over state at a single timestep. However, this can make tracking susceptible to unreliable

Fig. 4: Left: Setup for pushing experiments with occlusion using Barrett-WAM manipulator. The white box is the pushed object, with general pushing direction indicated by the blue arrow. The system is observed by a depth camera to the left (out of frame). Right: visualization of the tracked system in DART [21], with the observed pointcloud marked in dark grey.

Fig. 5: Mean error and standard deviations of object pose estimates (after the last iSAM2 step has been performed). CP, SDF, and QS model results are compared raw measured values, and to those produced by the graph described in Yu et al. [11]. Tracking performance is greatly improved with the inclusion of geometric and physics-based priors. The comparison with [11], which does not use SDF priors, indicates the importance of enforcing these constraints in practice.
TABLE II: Error Results for Force and Contact Recovery

| Component                      | RMSE | MAE  | σ    |
|--------------------------------|------|------|------|
| Force magnitude (N)            | 0.352| 0.195| 0.043|
| Force direction (deg.)         | 3.15 | 2.54 | 0.78 |
| Contact location (cm)          | 0.32 | 0.14 | 0.18 |
Fig. 8: Examples of pushing trajectories performed on the YUMI system. Initial object and finger pose estimates are provided by the DART tracking system. Contact points and force measurements are estimated by the analytic tactile sensor model [13]. Each trajectory is optimized using the QS graph depicted in Fig. 2c. Two-sigma values and force vectors shown at every 10th timestep for visual clarity. Joint inference over kinematic and force trajectories decreases uncertainty in poses as well as contact points and forces, and smoothens noisy tactile data to agree with physics-based constraints.

Fig. 9: Visualizations of measurements for corresponding trajectories in Fig. 8. Measured positions, contact points and force-vector outputs from the learned sensor model are shown on the left-hand side. Optimized values are shown on the right, indicating consistency of finger-object surface contact. Our approach produces force trajectories which more closely adhere to quasi-static mechanics. Joint inference allows kinematic trajectories to inform the force estimates, aligning forces to the object center of mass during linear motion, and correcting applied moments when motion is non-linear.

VII. CONCLUSION

We proposed a factor graph-based inference framework to solve estimation problems for robotic manipulation in batch and incremental settings. Our approach can leverage geometric and physics-based constraints along with vision and tactile based multi-modal sensor information to jointly estimate the history of robot and objects poses along with contact locations and force vectors. We perform several benchmarks on various datasets with multiple manipulators in real environments and show that our framework can contend with sensitive, noisy sensor data and occlusions in vision to efficiently solve for locally optimal state estimates that closely match ground truth. Future work will include incorporating the approach within a motion planning context [9], combining vision and tactile modalities in learning predictive sensor models [22], [23], and the possibility of integration into a hierarchical task-planning framework.
REFERENCES

[1] T. Schmidt, K. Hertkorn, R. Newcombe, Z. Marton, M. Suppa, and D. Fox, “Depth-based tracking with physical constraints for robot manipulation,” in Robotics and Automation (ICRA), 2015 IEEE International Conference on. IEEE, 2015, pp. 119–126.

[2] J. Bimbo, L. D. Seneviratne, K. Althoefer, and H. Liu, “Combining touch and vision for the estimation of an object’s pose during manipulation,” in Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on. IEEE, 2013, pp. 4021–4026.

[3] L. Zhang and J. C. Trinkle, “The application of particle filtering to grasping acquisition with visual occlusion and tactile sensing,” in Robotics and Automation (ICRA), 2012 IEEE International Conference on. IEEE, 2012, pp. 3805–3812.

[4] M. Chalon, J. Reinecke, and M. Pfanne, “Online in-hand object localization,” in Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on. IEEE, 2013, pp. 2977–2984.

[5] M. Montemerlo, S. Thrun, D. Koller, B. Wegbreit, et al., “Fastslam: A factored solution to the simultaneous localization and mapping problem,” Artificial Intelligence, vol. 156, no. 1–2, pp. 91–114, 2004.

[6] M. Montemerlo, J. Dong, F. Dellaert, and B. Boots, “Continuous-time Gaussian process motion planning via probabilistic inference,” The International Journal of Robotics Research (IJRR), 2018.

[7] M. Mukadam, J. Dong, Y. Xu, F. Dellaert, and B. Boots, “Simultaneous trajectory estimation and planning via probabilistic inference,” The International Journal of Robotics Research (IJRR), 2018.

[8] M. Mukadam, J. Dong, F. Dellaert, and B. Boots, “Simultaneous trajectory estimation and planning via probabilistic inference,” in Proceedings of Robotics: Science and Systems (RSS), 2017.

[9] K.-T. Yu, J. Leonard, and A. Rodriguez, “Shape and pose recovery from planar pushing,” in Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on. IEEE, 2015, pp. 1208–1215.

[10] K.-T. Yu and A. Rodriguez, “Realtime state estimation with tactile and visual sensing. application to planar manipulation,” in 2018 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2018, pp. 7778–7785.