Automated Assembly for Mesoscale Parts

David J. Cappelleri, *Member, IEEE*, Peng Cheng, Jonathan Fink, *Student Member, IEEE*, Bogdan Gavrea, and Vijay Kumar, *Fellow, IEEE*

***Abstract*—This paper describes a test-bed for planar micro and mesoscale manipulation tasks and a framework for planning based on quasi-static models of mechanical systems with intermittent frictional contacts. We show how planar peg-in-the-hole assembly tasks can be designed using randomized motion planning techniques with Mason's models for quasi-static manipulation. Simulation and experimental results are presented in support of our methodology. We develop this further into a systematic approach to incorporating uncertainty into planning manipulation tasks with frictional contacts. We again consider the canonical problem of assembling a peg into a hole at the mesoscale using probes with minimal actuation but with visual feedback from an optical microscope. We consider three sources of uncertainty. First, because of errors in sensing position and orientation of the parts to be assembled, we must consider uncertainty in the sensed configuration of the system. Second, there is uncertainty because of errors in actuation. Third, there are geometric and physical parameters characterizing the environment that are unknown. We discuss the synthesis of robust planning primitives using a single degree-of-freedom probe and the automated generation of plans for mesoscale manipulation. We show simulation and experimental results of our work.**

***Note to Practitioners*—Micro and mesoscale systems technology is poised to be an extremely strong economic driver in this century. Market estimations predict large quantities of products involving this technology within the next decade and more specifically, meso and microscale assembly shows enormous potential in a vast range of industrial applications. As more microelectromechanical systems (MEMS), microfluidic, and optoelectronic devices come on the market, the complexity and cost of the required manufacturing equipment and/or skill level of humans to assemble such devices has also increased. Even though there have recently been substantial advances in the fabrication of microparts, the assembly and packaging of microsystems and products still account for roughly 80% of the cost of commercial products. Manual, labor-intensive manufacturing will no longer be an option in this next generation of products that will require assembly at the meso, micro, and nanoscales. Thus, the global trend of miniaturizing products has led to many new assembly challenges that need to be solved in order for companies to remain competitive. Therefore, driven by these increasingly competitive requirements for faster throughput, higher yield, and quicker "time to profit" of products, the need for automated robotic assembly of meso and microsystems is quite apparent. This paper is on deriving the fundamental concepts needed to make these types of systems a reality. System modeling, model fitting, open loop motion plans, robust motion primitives, and quasi-open loop motion plans are synthesized for the canonical peg-in-the-hole assembly task at the mesoscale. Insights on extending the methodology presented to smaller length scales are also provided.**

***Index Terms*—Mesoscale manipulation, microassembly, planning.**

Manuscript received November 01, 2010; accepted March 06, 2011. Date of publication April 21, 2011; date of current version July 07, 2011. This paper was recommended for publication by Associate Editor A. Frank van der Stappen and Editor K. Goldberg upon evaluation of the reviewers' comments. This work was supported in part by the National Science Foundation (NSF) under Grant DMS01-39747, Grant IIS02-22927, Grant IIS-0413138, and ARO Grant W911NF-04-1-0148. The work of B. Gavrea was supported in part by the Sectoral Operational Program Human Resources Development 2007–2013 of the Romanian Ministry of Labor, Family and Social Protection under the Financial Agreement POSDRU/89/1.5/S/62557.

D. Cappelleri is with the Department of Mechanical Engineering, Stevens Institute of Technology, Hoboken, NJ 07030 USA (e-mail: David.Cappelleri@stevens.edu).

P. Cheng is with The MathWorks Inc., Natick, MA 01760 USA (e-mail: Peng.Cheng@mathworks.com).

J. Fink is with the Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA 19104 USA (e-mail: jonfink@grasp.upenn.edu).

B. Gavrea is with the Department of Mathematics, Faculty of Automation and Computer Science, Technical University of Cluj-Napoca, Cluj-Napoca 400114, Romania (e-mail: Bogdan.Gavrea@math.utcluj.ro).

V. Kumar is with the Department of Mechanical Engineering and Applied Mechanics, University of Pennsylvania, Philadelphia, PA 19104 USA (e-mail: kumar@grasp.upenn.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASE.2011.2132128

I. Introduction

While mass production techniques derived from hard automation are routinely applied to microscale (several microns to tens of microns) and mesoscale (hundreds of microns to mm) parts, we are not able to achieve micro and mesoscale assembly in a semi-structured environment with uncertainties. Indeed, reliably manipulating parts at this scale remains challenging. There are many reasons for this. For example, there is a lack of good models of mechanics of contact interactions at these scales. It is also difficult to measure forces at the micronewton level (0–50 μN range) reliably using off-the-shelf force sensors and thus, force-feedback control schemes have proved to be very challenging. Finally, it is even more difficult to grasp and manipulate parts at the micro and meso level than it is at the macro level.

A natural question, one that has been asked before [1]–[3], is if simple open loop or quasi-open loop plans that do not require precise feedback in real-time, can be designed to accomplish such tasks. In this paper, we explore such open loop and quasi-open loop plans for a mesoscale assembly task. The manipulation task that we are interested in is complicated due to the fact that the manipulated object is subject to an unknown and hard-to-model distribution of contact forces between the support surface and the object as well as unknown frictional contact forces between the manipulation tool and the object and between the object and its environment. This is further complicated by the fact that contacts are intermittent. Clearly, analytical solutions to the forward dynamics problem are impossible except in the simplest of cases, so simulation-based solutions are the only viable option.

We aim to use simulation and motion planning tools to design manipulation plans that rely only on an estimate of initial position and orientation (open loop) or estimates of state information at a small number of discrete-time intervals (quasi-open loop). However, manipulation plans are not robust to errors resulting from the uncertainty in the system model. This uncertainty comes from three sources: (a) the force distribution or the pressure distribution across the contact between parts and the surface they rest on; (b) the coefficient of friction between the part and the contacting surface; and (c) the coefficient of friction between the probe and the manipulated part. While it is possible to measure these parameters in a laboratory setting, it is difficult to directly measure these parameters just before or while performing the task.

Using only open loop or quasi-open loop motion plans makes the assembly task complicated. Only plans that are robust to uncertainty in initial positioning and the dynamics can be utilized. Sensors that limit the initial positioning errors and models that lend themselves to the analysis of uncertainty of both task error (initial placement of parts) and model error (estimation of parameters) are desired. It is important not only to have accurate system parameter information but also important to have error models for the system that can be used to plan the manipulation task. Therefore, the goal here is to be able to come up with a manipulation plan, consisting of pushing motions, that guarantees successful assembly in the presence of uncertainty. This is a motion planning problem with frictional contacts at the micro and mesoscales. To study this problem, the automation of the canonical problem of the peg-in-the-hole assembly task at the mesoscale has been investigated and will be presented now. The technologies used to examine this problem can be directly applied to more complicated problems at this scale and slightly modified to apply to similar problems at the microscale.

II. Related Work

The derivation of the fundamental mechanics of pushing operations and sliding objects have been extensively studied by [4]–[6]. There is also extensive work addressing the analysis and simulation of mechanical systems with frictional contacts. In particular, semi-implicit and instantaneous-time models for predicting motion and contact forces for quasi-static multi-rigid-body systems have recently been developed [7], [8].

A good survey of motion planning under uncertainty is available in [9] and [10]. Pushing operations and the instantaneous motions of a sliding object during multiple contact pushing are examined and the manipulation primitive of stable rotational pushing is defined in [11]. The problem of planning pushing paths using stable pushes is discussed in [12]. In [13], the bounds of the possible motions of a pushed object are investigated. Reference [14] presents a comparison between the dynamic and quasi-static motions of a push object.

Open loop motion strategies, without the use of sensors, can be used to eliminate uncertainty and to orient polygonal parts [1], [2], [15]. Other motion planning techniques have been applied to planning pushing paths with the presence of obstacles [16] as well as for multiple manipulators and obstacles [17]. To remove the uncertainty associated with robot pushing tasks, [18] establishes stable orientation and positions by pushing objects with two-point fingers.

For microscale manipulation, sticking effects due to Van der Walls forces and static electricity make the manipulator motions and part release more complicated [19], [20]. Micromanipulators also have limited degrees of freedom when compared to manipulators at the macroscale. These problems are addressed in [3] with a parallel-jaw gripper and squeeze and roll primitives to orient a randomly oriented polygonal part up to 180° symmetry without the use of sensors. However, in the case of the mesoscale sized parts used here, these sticking effects are not an issue since the characteristic dimensions are on the order of 1000 μm (1 mm) [19], [21].

On the other hand, the literature addressing micromanipulation with real-time sensor feedback is more limited and quite challenging [22]–[27]. The primary reason for this is that obtaining accurate sensor data is a difficult problem at this scale. Sensors cannot easily be affixed to tiny precision instruments without compromising their functionality [20]. The use of high-resolution optical systems with controllable parameters for micro-assembly tasks are examined by [28]. Even with this sensor data, calibration and vision-based control at this scale can present technical difficulties. Without accurate sensor data, it is challenging to develop models, and therefore controllers, for micromanipulation.

Gripping and manipulation techniques for micro-assembly applications is an active area of research [29], [30]. Specifically, there is a body of work pertaining to pick-and-place micro-assembly tasks using microgripping techniques and strategies [31]–[37]. The focus here is rather on mesoscale (and eventually microscale) pushing operations, which are better suited for open loop or quasi-open loop manipulations. For manipulation and assembly tasks at small length scales, surface forces, such as stiction, friction, and electrostatic forces, dominate. They also increase with increased contact surface area. Thus, they can be reduced by pushing the parts with probe tips that have smaller surface contact areas than the jaws of typical microgrippers. Since most parts are planar, access to the parts from the top is possible. Grasping the part with a suction gripper can be performed, however, it can only be used to approximately position the parts due to the difficulties, in part, releases resulting from aforementioned dominant surface forces. Therefore, manipulation and assembly with a gripper is not a preferred technique and manipulations with point probes, with decreased surface contact area, are utilized here.

Modeling dry friction is a notoriously difficult problem area. Estimations of friction parameters for pushed objects to improve the control of pushing have been investigated previously on larger objects and with different strategies than the ones presented here. In [38], test pushes on different objects with known support surfaces are used to estimate support surfaces experimentally. It leaves the open question of how the hypothesized support points for an unknown object should be chosen. Similarly, in [39], a method for estimating the friction distribution of an object and the center of friction from pushing the object several times is presented. In both of these papers, a grid system of N possible support points is applied to the base of the object being pushed. The respective algorithms determine the distribution of the normal force of the object at these support locations. In our previous work, estimates of surface friction for mesoscale manipulation are experimentally determined [40]. Also, in some

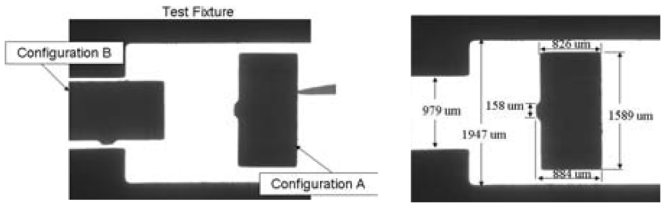

Fig. 1. Canonical peg-in-the-hole task: move part from configuration A to configuration B (left); typical part and fixture dimensions (right).

of our current experiments, the support surface is coated with a thin film of oil which circumvents the difficulties of modeling dry friction. At the micro and nanoscales, higher order nonlinearities and different asperity models can and have been employed for controlling a variety of stick and slip devices and end-effectors [41]–[43].

In particular, the problem of finding motion primitives that rely on pushing and are robust to errors has received significant attention [44], [45]. A pushing control system with visual feedback for open loop pushing is described in [46] as a way to mitigate the instability of pushing with point contacts. To remove the uncertainty associated with robot pushing tasks, [18] establishes stable orientation and positions by pushing objects with two-point fingers. The problem of planning pushing paths using stable pushes with line contact is discussed in [12], and conditions on the pushing directions are derived that ensure that line sticking contact will always be maintained. Sampling-based motion planning algorithms that attempt to find paths that minimize uncertainty have also been explored in the literature [47], [48].

III. Canonical Problem

The canonical problem of the peg-in-the-hole task at the mesoscale is defined here as assembling a planar, rectangular part into a planar, rectangular slot (Fig. 1) using pushing operations.

In macro assembly, the peg-in-the-hole situation (or simply insertion) is relatively generic of most assemblies. In precision micro and mesoscale assembly, peg-in-the-hole tasks also occur but with more difficulties such as [21], [49], and [50]:

- Identifying the peg and the hole, their exact position, the axis of the hole.
- Gripping/manipulating the peg.
- Aligning the peg axis with the hole..
- Inserting the peg in the hole while limiting the constraints on the components.
- Position control.
- Uncertainties in sensing, actuation, manufacturing, and modeling.

This task is a good representation of the micro and meso-assembly environment for the following reasons: First, it's a two-dimensional problem, which is typical at both the micro and mesoscales. Second, the inertial forces in the rigid body dynamic model are not significant as is also typical in tasks at these scales. Therefore, frictional forces are very important here. Finally, the task involves multicontact interactions between the part and the environment. Therefore, the techniques applied in the study of this mesoscale problem can be generally applied to other examples at this and the microscale.

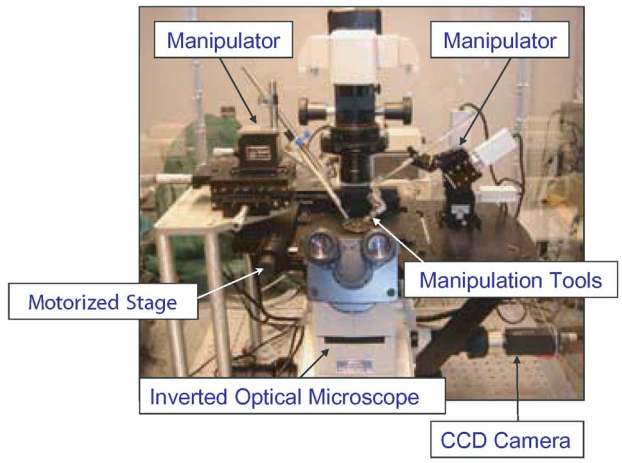

Fig. 2. Micro/mesoscale manipulation experimental test-bed.

IV. Experimental Setup

The manipulation system used in this study is shown in Fig. 2. It consists of an inverted optical microscope and charge-coupled device (CCD) camera (for sensing the configuration), a four-axis computer controlled micromanipulator, controller, a three-axis manual micromanipulator, 5 μm and 25 μm diameter tip tungsten probes, a motorized XY stage, and control computer. The 4X microscope objective used in this application produces a field-of-view (FOV) of 3.37 mm $\times$ 2.52 mm. The CCD camera records the images in the FOV and sends them to the control computer at 30 Hz (lower frequency with image processing). The micromanipulator with controller has a minimum incremental motion of 0.1 μm along four axes, with a maximum travel of 20 mm, with speeds ranging from 1.6 μm/s to 1.7 mm/s.

The fixtures and parts are made out of beryllium copper using a photochemical machining (PCM) process (sometimes referred to as chemical milling or chemical etching) (Fig. 1). This technique is useful for manufacturing high-precision flat metal parts [51]. The beryllium copper material is 1.5 mil ($0.0015''$ or 40μm) thick. The fixture has two channels of different thickness etched into it (1936 μm and 976 μm, respectively) but it also has a rectangular section etched out of it that is 1.63 mm $\times$ 1.15 mm $(0.641'' \times 0.454'')$. It is used as a playpen area where unconstrained motion tests can be performed for system characterization experiments. The hole or fixture is attached to a glass microscope slide which is coated with a thin layer of mineral oil for the parts to slide on. Various size parts were manufactured through the PCM process. Typical dimensions for one such part are shown in Fig. 1. Standard image processing techniques, such as template matching and sum-of-squared differences (SSD) tracking, are used to sense the state of the system in real-time. The state consists of the position (X, Y) and orientation (Θ) of the peg and the probe tip position(s) (X_{tip}, Y_{tip}).

V. Open Loop Assembly Problem Formulation

Due to the small size and mass of the part, the inertial forces do not dominate during these planar pushing operations [20]. For these manipulations, the velocity for the probe is set to one

of three discrete values: 140.0, 75.0, or 7.4 $\mu\mathrm{m/s}$. The inertial forces are then on the order of nano-newtons for the accelerations involved, while the frictional forces are on the order of micro-newtons. Therefore, the frictional forces dominate the inertial forces and one can adopt a quasi-static framework to solve for motions of the peg. This framework together with the frictional constraints and the rigid body constraints can be posed as a complementarity problem [52], [53] and subsequently solved to determine the overall motion of the system at every time-step. Simulation-based randomized planning algorithms can then be used to design manipulation plans for the peg insertion problem [10].

A. Quasi-Static System Modeling

A semi-implicit time-stepping scheme [8] that is specialized to a $2.5D$ problem, where all parts and contact interactions are essentially planar and surface friction in the plane is modeled with a simple force distribution, is used here. All contacts are assumed to behave according to Coulomb's friction model. The interaction between the part and the supporting surface is modeled by three frictional point contacts as in [4]. This problem of a planar polygonal part sliding with surface friction and quasi-static constraints can be formulated as a mixed linear complementarity problem (MLCP) [8]. For a multibody system with n_q degrees of freedom, the quasi-static equations of motion, and time-stepping are

$$0 = W_n \lambda_n^{l+1} + W_f \lambda_f^{l+1} + F_{ext} \tag{1}$$

$$q^{l+1} - q^l = hG(q)\nu^{l+1} \tag{2}$$

where $q^l \in \mathcal{R}^{n_q}$ is the generalized configuration vector, $\nu^{l+1} \in \mathcal{R}^{n_\nu}$ is the generalized velocity vector, $G(q) \in \mathcal{R}^{n_q \times n_\nu}$ is the Jacobian matrix, $W_n \in \mathcal{R}^{n_\nu \times (n_c+n_s)}$ contains the normal wrenches for each of the $n_c + n_s$ pushing (n_c), and support (n_s) contacts with normal constraint forces $\lambda_n \in \mathcal{R}^{(n_c+n_s)}$. $W_f \in \mathcal{R}^{n_\nu \times (2n_c+n_d n_s)}$ contains the frictional wrenches (with the friction cone linearized into n_d directions for each contact) with frictional forces $\lambda_f \in \mathcal{R}^{2n_c+n_d n_s}$, and F_{ext} represents the external forces. The rigid body nonpenetration constraint and linearized Coulomb friction law result in a set of complementarity conditions.

The uncertainty in support pressure distribution and continuous nature of the surface contact makes it difficult to model the system. However, if the support is modeled by three frictional point contacts, the normal force distribution can be found based on a out-of-plane force balance, and the maximum allowable frictional forces resulting from these normal forces can be used in the MLCP formulation. However, uncertainties in the exact locations of the correct three support points are now introduced into the model. For the purposes of simulation and modeling, the manipulator can be considered an arbitrary convex polygon in the plane with position p_m. Because of the choice to use a three-point support approximating pressure distribution, the part can, in fact, be any planar polygonal shape. The probe is modeled as a polygon with a 25 $\mu\mathrm{m}$ side. During the experiments, contact between the probe and part occurs at one of the vertices on this side (point contact) or with the side (line contact).

By partitioning the MLCP formulation in [8] and [54], one can take advantage of these known normal forces for the support points and the assumption that the bodies are constrained to the plane to reduce the dimension of the MLCP that must be solved.

B. Planning Problem Formulation

The goal is to determine a sequence of manipulator steps that will successfully accomplish the peg-in-hole task. To model this system, suppose the rectangular peg has three support points at positions r_1, r_2, r_3 from its center-of-mass. The peg is oriented at θ degrees from horizontal and the manipulator (probe) is pushing at a point p.

For a given configuration of the peg, the control inputs explored by the motion planner are determined by calculating and discretizing the peg edges that can be pushed by the probe. Further, it is required that one of the two components of the motion of the probe to be zero. In other words, for simplicity, motions are restricted to be along either the X or Y axes.

When a peg state near the specified goal is reached, a motion planning algorithm will compute the series of manipulator movements to apply each necessary control input. The manipulation plan is completely represented as a vector $\underline{u}$, where each element corresponds to a probe movement command. A probe movement command will be $\Delta u_i = (\Delta x, 0)$ or $(0, \Delta y)$ corresponding to a probe movement of Δx or Δy along the X axis or Y axis, respectively. Each element Δu_i is related to the quasi-static time-stepping model by calculating the duration for the push according to the specified probe velocity.

VI. System Identification

A software tool has been specifically designed to provide a simulation environment well suited for design and planning tasks that require accuracy and flexibility in [55]. It employs a plug-in architecture so that modules can be substituted for a variety of motion models (first-order, quasi-static, dynamic) and time-stepping methods. Most importantly, it provides for rapid development of design optimization or motion planning algorithms while providing the ability to easily choose the appropriate dynamic modeling for a given application. It has been used here with the quasi-static system model.

There are several parameters present in the simulation model that affect the mechanics of the manipulation task and are unknown. These are the following.

- μ_s: Coefficient of surface friction.
- μ_t: Coefficient of manipulator-part friction.
- $\underline{r_s}$: A 3 × 2 matrix specifying the x and y positions of the support-point locations.

The coefficients of friction μ_s and μ_t are constrained to the range [0.0, 1.0]. Support-point locations must obviously lie within the dimensions of the part being modeled. Additionally, their convex hull must include the part's center-of-mass so that the appropriate normal force at each support point can be calculated. The set of parameters is an eight-dimensional parameter space. The goal is to find the point in the parameter space and the neighborhood of the point that most closely characterizes the uncertainty in the system.

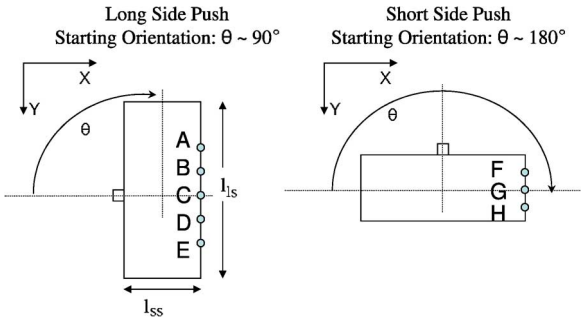

Fig. 3. Nominal initial conditions of part for manipulation tests.

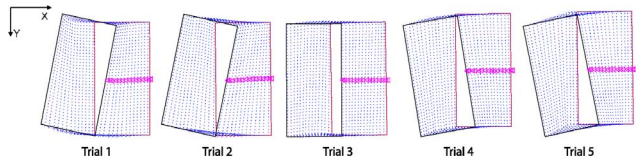

Fig. 4. Experimental trajectories for manipulation Test 1.

In order to design open loop or quasi-open loop manipulation plans robust to the system uncertainties, nominal values for the critical system parameters must be identified. Many sets of manipulation tests were used to identify and characterize the parameters in the eight-dimensional parameter space. Manipulation tests consisted of horizontal moves with contact between the probe and the part over distances of approximately 700 μm, executed on both the long and short side of the part. Fig. 3 shows schematics of the part in its nominal initial conditions for eight sets of tests. The coordinate system is chosen to align it with that of the images obtained from the vision system. Pushes were made on the long side of the part at five nominal positions—at the midpoint of the side (Fig. 3, pt. C), midpoint $+l_{ls}/4$ (pt. B), midpoint $-l_{ls}/4$ (pt. D), midpoint $+l_{ls}/2$ (pt. A), and midpoint $-l_{ls}/2$ (pt. E), where l_{ls} corresponds to the length of the long side of the part. For these pushes, the part was nominally placed at angle of 90°. The short side pushes were located at three nominal positions—at the midpoint (pt. G), midpoint $+l_{ss}/2$ (pt. F), and midpoint $-l_{ss}/2$ (pt. H), where l_{ss} corresponds to the length of the short side of the part. These pushes start with the part nominally placed at angle of 180°. A minimum of three trials for each manipulation test were executed.

The method employed for the system identification seeks to answer the following question: Given experimental data consisting of trajectories for each of the manipulation test experiments, what is the parameter vector $\mathbf{p}$, consisting of μ_s, μ_t, and r_s, that best explains the experimental data? This estimation algorithm estimates parameters such that the simulated manipulation test results match the experimental results from the execution of real manipulation tests as well as possible. Because the manipulated part's motion is a nonconvex and nonsmooth function of the system parameters, one cannot use gradient methods for optimization. This limits the choice of optimization algorithms to ones that rely only on the evaluation of an objective function at different points. The objective function $(F(\mathbf{p}))$ to be minimizing here is related to the fitting of simulation to experiments over several trial test manipulation runs. For each trial, $\mathbf{x}_i$ and a given parameter vector $\mathbf{p}$, the simulated motion $\mathbf{s}_i$ is computed, and the root-mean-squared error along each axes x, y, θ (with θ scaled by the characteristic length of the part to normalize) determined. A quality measure for the single trial simulation fit is computed via the L_∞-norm (3). When fitting across several experimental trials, the total objective becomes the average of quality measures across all trials (4)

$$f(\mathbf{x}_i, \mathbf{p}) = \left\| \sqrt{\frac{1}{T}\sum_{t=1}^{T} \left(\mathbf{x}_i(t) - \mathbf{s}_i(\mathbf{p}, t)\right)^2} \right\|_\infty \tag{3}$$

$$F(\mathbf{p}) = \frac{1}{N}\sum_{i=1}^{N} f(\mathbf{x}_i, \mathbf{p}). \tag{4}$$

It should be noted that a single evaluation of $F(\mathbf{p})$ requires N individual simulations.

The Nelder–Mead algorithm [56] (also known as the downhill simplex method), a common method used for solving nonlinear optimization problems when the objective function is nondifferentiable, is used here. The algorithm uses a $dim(\mathbf{p})+1$ simplex to find locally optimal solutions through successive evaluations of an objective function and transformations of points on the simplex. The algorithm is initialized with a value of $\mathbf{p}$ based on random valid support point locations and coefficients of friction. The objective function $F(\mathbf{p})$ is customized to return arbitrarily large values when given a parameter choice $\mathbf{p}$ violates the constraints. The algorithm is stopped when the simplex size (measured as the average distance from simplex center to each point) goes below a threshold. This is an adequate stopping criteria as the algorithm contracts the simplex when it finds a minimum.

Since the Nelder–Mead method is a downhill search technique, it can be stuck in local minima and fail to find the globally optimal solution. In order to reduce this possibility, the algorithm is initialized with several different random points in the parameter space. Since it is believed that the parameter space may have multiple points that closely match the experiments, this practice has the added benefit of finding several good parameter selections that can be used to help characterize the model uncertainties that a manipulation planning algorithm will have to consider.

1) Experimental Errors: At least three trials for each manipulation test were performed. The trajectories for each of the trials were not all the same. They were consistent for the most part, but for certain starting configurations small variations from the nominal starting configuration of the part and probe can produce substantial changes in the resulting trajectory. For each manipulation test performed, the X trajectory of the parts all showed the best correlation between the trials, while the Y and θ trajectories appear coupled and comprise most of the trajectory errors. Test 1 is an example of a manipulation test that is

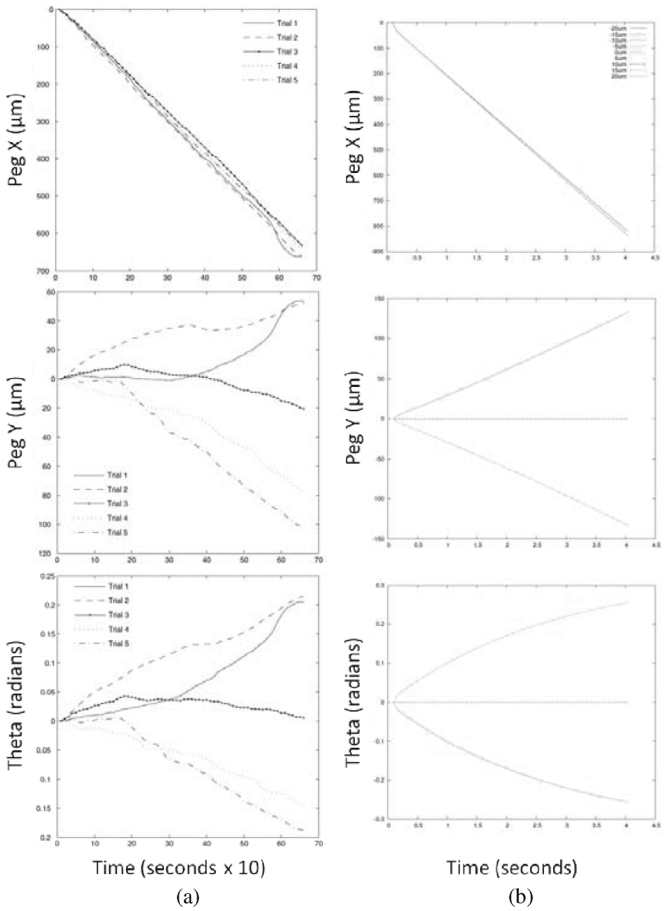

Fig. 5. Manipulation test showing large sensitivity to small errors in initial configuration. (a) Experimental. (b) Simulation.

sensitive to initial conditions, as shown in Fig. 4. This test involves pushing at the midpoint of the long side of the rectangular part. A horizontal push directly at the center-of-mass of the part should intuitively result in a pure translation, as seen in the figure for Trial 3. In general, small perturbations from this starting configuration will yield either clockwise or counterclockwise rotations for the same nominal test as seen in Trials 1, 2, 4, and 5. Most of this can be attributed to errors from the nominal starting position at the beginning of the tests. The accuracy of the vision system is conservatively estimated to be ± 1 pixel, which corresponds to position measurement errors of the probe and the part of roughly $\pm 5\ \mu\mathrm{m}$ and angular measurements of roughly $\pm 0.005^\circ$ error.

Fig. 5(a) demonstrates the experimental trajectories from Test 1 that should result in a pure translation but actually produce a wide range of rotations. However, as we can see from Fig. 5(b), our model and therefore the simulation predict this sensitivity. When the simulator is presented with initial conditions for the part that lie in a neighborhood of within 5 $\mu\mathrm{m}$ and 0.005° of the nominal position, the resulting trajectories exhibit a variation that is similar to that observed experimentally. Because the outcome is more sensitive to small errors in initial configuration, it is clear such pathological initial conditions should be avoided for manipulation planning.

2) Parameter Estimation Results: Only experimental results from manipulation tests that were generally repeatable were used—avoiding "pathological" configurations. The Nelder–Mead-based estimation algorithm is able to match simulated trajectories to experimental trials with average root-mean-squared error of 20–40 $\mu\mathrm{m}$ in position and of 1°–3° in orientation over a 600 $\mu\mathrm{m}$ motion of the probe/part. Smaller motions result in smaller errors. The failure to obtain more accurate matches across large sets of trials could be partially attributed to measurement errors, not only the 5 $\mu\mathrm{m}$, 0.005° error in measuring the trajectories, but also the effect of these errors in estimating the initial configuration. However, it is also likely that failures are the result of inconsistencies in the supporting surface.

TABLE I
ESTIMATED PARAMETER VALUES FOR EXAMPLE

	$\mathbf{p}_1^*$	$\mathbf{p}_2^*$
$\frac{r_s}{(\mu\mathrm{m})}$	$(411.3, -563.3)$ $(-409.1, 605.9)$ $(-273.2, -102.1)$	$(-412.4, 794.0)$ $(-217.8, 774.5)$ $(278.8, -538.8)$
μ_t	0.666	0.536
μ_s	0.757	0.013

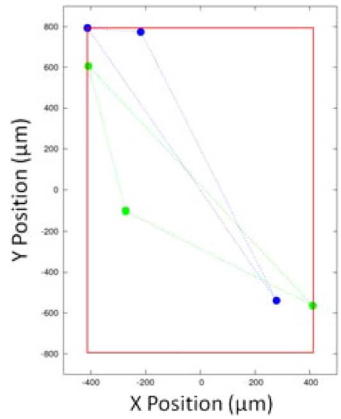

Fig. 6. Support point locations for $\mathbf{p}_1^*$ and $\mathbf{p}_2^*$.

By initializing the estimation algorithm with several random parameter choices, the algorithm often discovers more than one local minima in the parameter space. One choice of parameters can closely match some experimental trials while another choice can match other trials well. For a concrete illustration of the result from parameter estimation, a simple parameter estimation is performed using two experimental tests as input. The algorithm is initialized with two random sets of parameters $\mathbf{p}$ and found two possible solutions $\mathbf{p}_1^*$ and $\mathbf{p}_2^*$ (Table I, Fig. 6). As an additional test of the quality of $\mathbf{p}_1^*$ and $\mathbf{p}_2^*$, results of simulating another trial that was *not* used for parameter estimation were evaluated. The uncertainty of the locally optimal parameter solutions is approximated by the size of the Nelder–Mead simplex which is also the stopping criteria of the implementation and set to 10 $\mu\mathrm{m}$ in this case. Specific values for $\mathbf{p}_1^*$ and $\mathbf{p}_2^*$ are shown in Table I and Fig. 6. Although the parameter sets $\mathbf{p}_1^*$ and $\mathbf{p}_2^*$ are distinct, they are qualitatively similar since they produce trajectories with comparable errors from experimental trajectories. Note: the solution for the support point locations are not unique, i.e., the same trajectory can be solved for using different sets of support points [4].

The solutions $\mathbf{p_1^*}$ and $\mathbf{p_2^*}$ are similar but have a large discrepancy in the value of μ_s. In fact, it was observed that in simulation the value of μ_s has minimal affect on the trajectory of the part. This is due to the fact that it is the distribution of support force and not the coefficient of surface friction that will affect the motion. Since the test-bed cannot sense the force being applied by the probe, one cannot observe the actual frictional forces being exerted.

VII. Randomized Planning Algorithm and Experimental Results

A sampling-based motion planning algorithm inspired by the RRT algorithm [57] with modifications for dynamic systems described in [58] was developed here to produce relatively short manipulation paths. Instead of searching for continuous input trajectories $u(q,t)$, the input is parameterized by a r-dimensional vector $\underline{u}$ using piecewise-constant functions with compact support. When using an RRT algorithm for dynamic systems such as this, there is always a tradeoff between the coarseness of the discretization and the number of iterations necessary to find a goal. For this problem, discretizing the reachable peg surface into 100 pushing locations led to solutions in as few as 12 iterations of the RRT algorithm. However, since the number of control inputs to test at each iteration is large and the LCP-based method of simulation can be computationally expensive, the algorithm took on the order of 10–20 min (on a 1.8 GHz PC) to find a solution. Typical solutions include about 7–10 pushing operations which is comparable to an initial, intuitive, solution to the problem. It should be noted that based on the assumptions used here, one cannot smooth or shorten the manipulation paths produced. Since we are searching in the space of control inputs on a system that is inherently nonsmooth, it is not possible to go back and attempt to "smooth" the part trajectory. Furthermore, smoothing the control input trajectory could result in a drastically different part trajectory due to the nature of the problem. It was found that while RRT produced open loop manipulation plans that were feasible when used in experimentation and sometimes succeeded, the plans are not robust to error in the support friction modeling or the initial positioning of the peg. Thus, the overall success rates for the RRT-based plans were approximately 60%–70%. Fig. 7 shows snapshots of the experimental execution of an automatically generated RRT-based motion plan. Because the clearance between the rectangular part and the slot is 137 μm, it is possible to use the approximate model for successful assembly. However, it is clear that tighter tolerance tasks will require better models.

The success rates for the RRT manipulation plans lead us to wonder if the cause for failures is due to problems with the robustness of the plans themselves or if perhaps inaccuracies in the simulation of the pushing dynamics are to blame. Therefore, we can parameterize the structure of an intuitively derived manipulation plan to investigate this cause. Table II shows a plan designed with this methodology. Note, the origin for the coordinate frame is the top left corner of the fixture in the image. Intuitively, we know that the peg has to be rotated first before it can be pushed into the hole. Thus, the first moves in the intuitive plan is to move the probe tip from it's initial position at the midpoint of the long side of the peg towards the top end of the peg (Δu_1) and then translate it along the X axis (Δu_2) in order to rotate it. Since the part is not pinned at its center of mass, it will not rotate perfectly 90° so in the next step, the probe tip is moved down (Δu_3) to get the total rotation of the peg as close to 90° as possible. The inputs $\Delta u_4 - \Delta u_6$ are moves to reposition the probe tip to the right, middle region of the short end of the peg so that in the final move, an X axis move (Δu_7) can be used to translate the part into the hole. Therefore, the output of our intuition is a predefined seven step sequence with unknown probe movement amounts. The parameters for this plan are found by trial-and-error with our simulator. Note: this procedure can be automated in the future but is currently a manual process.

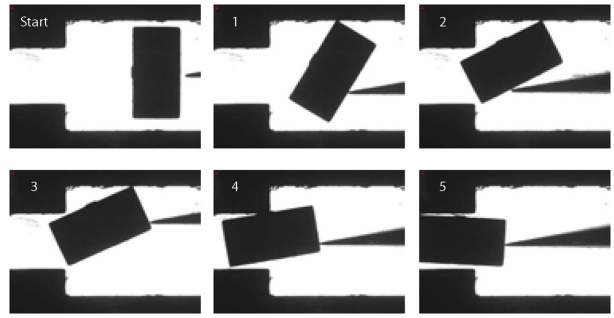

Fig. 7. Manipulation task: experimental results obtained from a plan generated by the model in Table I.

TABLE II
Manipulation Plan Parameters From Simulator

Peg Start Pos. (X,Y,θ)	(2.33 mm, 1.21 mm, 89°)
Tip Start Pos. (X,Y)	(3.05 mm, 1.24 mm)
Δu_1	$(0, -600)\mu$m
Δu_2	$(-1500, 0)\mu$m
Δu_3	$(0, 50)\mu$m
Δu_4	$(0, -50)\mu$m
Δu_5	$(1550, 0)\mu$m
Δu_6	$(0, 775)\mu$m
Δu_7	$(-2550, 0)\mu$m

Several trials of the intuitive plan with the parameters determined from the simulations were executed with the micromanipulation test-bed. In the trials, the starting position for the peg varied from the nominal starting position by at most 26 μm, 74 μm, and 3° in the X, Y, and θ coordinates. The maximum differences between the probe starting and nominal positions were 11 μm and 21 μm in X and Y, respectively. All of the trials resulted in successful placement of the peg in the hole. Fig. 8 shows a detailed comparison between simulation and experimental results for an intuitively designed plan. The discrepancy at time $t = 0$ can easily be explained by the initial peg position error. Subsequent errors are both a result of this initial error as well as inaccuracies of the time-stepping model and its parameters. Note that even with the error in the initial peg position and model inaccuracies, the plan is successful in experimentation and closely matches the simulated data.

The results described here are promising and illustrate the potential for the use of quasi-static mechanics with models of frictional contact. However, there are some shortcomings. The RRT algorithm that was used did produce feasible open loop manipulation plans. While these plans were sometimes successful

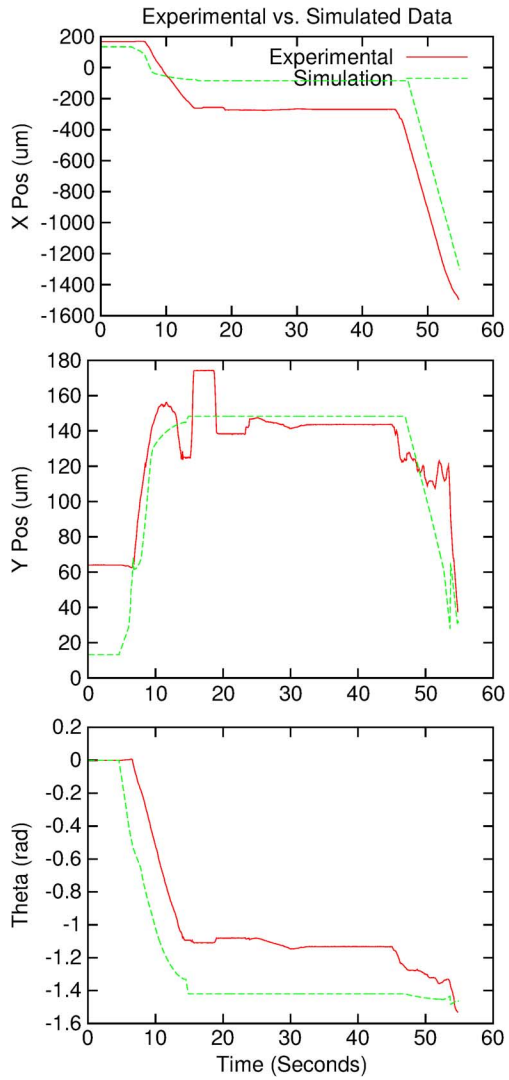

Fig. 8. Plot comparison of insertion plan in simulation and in experimental trial.

when carried out experimentally, they were not robust to error associated with the uncertainty in the support friction models or to those associated with the initial positioning of the peg. Thus, an intuitive motion plan was generated to increase robustness and executed experimentally to successfully accomplish the task. The higher success rate for this plan leads us to conclude that the plans generated automatically via the RRT algorithm are missing the "qualities" of a plan that is robust to modeling error. Therefore, in the following sections, an investigation to determine the exact properties that constitute a robust manipulation plan is presented. The improved framework to automatically design quasi-open loop plans, robust to the system uncertainties, will now be described as an enhanced way to solve this mesoscale assembly problem.

VIII. Robust Motion Framework for Quasi-Open Loop Assembly

We will now investigate the quasi-open loop assembly problem, in which an open loop control is calculated based on visual feedback at discrete moments. The manipulation problem is studied in the framework of motion planning for systems that are subject to both differential equations and uncertainties. In this section, the framework for the general problem and a general planning methodology based on robust motions will be described. The application of the general method in the manipulation problem is described in Section IX.

A. Problem Description

Assume that the motion of the control system in the given environment is characterized by $\dot{x} = f(x, u, p)$, in which $x \in X \subset \Re^n$ is the state, $u \in U \subset \Re^m$ is the input, and $p \in P \subset \Re^l$ is the parameters for the system and environment. Given a control $\tilde{u} : [0, t_{\tilde{u}}] \to U$, a parameter history $\tilde{p} : [0, t_{\tilde{p}}] \to P$, and a state $x_0 \in X$ for some $t_{\tilde{u}} > 0$ (varies with $\tilde{u}$), the trajectory (a.k.a. motion) under $\tilde{u}$ and $\tilde{p}$ from x_0 is $\tilde{x}(\tilde{u}, \tilde{p}, x_0, t) = x_0 + \int_0^t f(\tilde{x}(\eta), \tilde{u}(\eta), \tilde{p}(\eta))d\eta$.

Consider three bounded uncertainty sets stemming from sensing $(B_{s_x^u})$, control/actuation $(B_{c_{\tilde{u}}^u})$, and the environment $(B_{e_{\tilde{p}}^u})$.

1) *Sensing uncertainty:* It is assumed that sensors can estimate the global state of the system with bounded error s_x^u. Let x and x^s, respectively, represent the actual and sensed states of the system. There is then $x \in B_{s_x^u}(x^s)$, in which $B_r(x') = \{x | \, \|x, x'\| \leq r\}$ is the r-neighborhood of state x with respect to a metric $\|\cdot, \cdot\|$ on X. Note: $x' \in X$ represents any state in X except the initial state x_0.
2) *Control uncertainty:* It is assumed that actuators will realize the commanded control with a bounded error $c_{\tilde{u}}^u$. Let $\tilde{u}$ and $\tilde{u}^i$, respectively, represent the actual and intended controls for the system. Then, there is $\tilde{u} \in B_{c_{\tilde{u}}^u}(\tilde{u}^i)$.
3) *Modeling uncertainty*: It is assumed that the geometry and the physics of the underlying model are parameterized by $\tilde{p}$ with bounded error $e_{\tilde{p}}^u$. Let $\tilde{p}$ and $\tilde{p}^n$, respectively, represent the actual and nominal parameter history. Thus, there is $\tilde{p} \in B_{e_{\tilde{p}}^u}(\tilde{p}^n)$.

Given a sensed initial state x_{init} and a goal set $X_{\text{goal}} = B_\tau(x_{\text{goal}})$ for a specified τ and x_{goal}, the objective is to compute a control $\tilde{u}$ (that may depend on feedback information) which will drive the system from x_{init} to X_{goal} under uncertainties.

B. Planning With Robust Motion Primitives

To solve the above problem is quite difficult. Because complete algorithms are difficult to find except for the simplest of problems, the synthesis of plans that are obtained by composing *robust motion primitives* are pursued. Robust motion primitives are used to define controls whose resulting trajectories will preserve a specified property of interest in the presence of uncertainties. A *property of interest* is modeled by a characteristic function, κ, which maps a trajectory into 0 or 1. If $\kappa(\tilde{x}) = 1$, then it can be said that the trajectory $\tilde{x}$ satisfies the given property and is called a *κ-motion*. The *admissible set* for a property κ is $\mathcal{A}_\kappa = \{\tilde{x} | \, \kappa(\tilde{x}) = 1\}$. If the system has uncertainty bound $e^u = (s_x^u, c_{\tilde{u}}^u, e_{\tilde{p}}^u)$, the *uncertainty neighborhood* of trajectory $\tilde{x} = (x_0, \tilde{u}, \tilde{p})$ is $\{\tilde{x}' | \, \|x_0', x_0\| \leq s_x^u, \|\tilde{u}', \tilde{u}\| \leq c_{\tilde{u}}^u, \|\tilde{p}', \tilde{p}\| \leq e_{\tilde{p}}^u\}$. A κ-motion is a robust motion primitive only if its uncertainty neighborhood is contained within the admissible set.

The *composition of robust motion primitives* is now considered. Let κ_1 and κ_2 be two properties. If there exists a robust

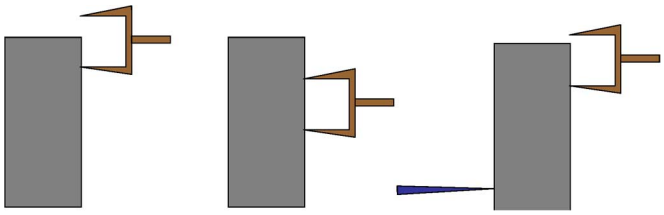

Fig. 9. Planar manipulation with a single degree-of-freedom, dual-tip probe and a passive single-tip probe. There are three sets of operations that can be performed.

κ_1-motion and a robust κ_2-motion such that the κ_1-motion can be reliably appended to the κ_2-motion under uncertainties, then it is possible to sequentially compose the motion primitives.

Thus, this approach to planning will involve the construction of a set of robust motion primitives followed by their sequential composition. At this point, a graph search based motion planning algorithm in [10] can be used to synthesize the motion plan. It is worth mentioning that such algorithms are generally not complete because they restrict the search space from the original control space to a smaller one consisting only of robust motion primitives.

IX. Motion Planning With Uncertainty

A. Experimental System Configuration

The micro/mesoscale manipulation test-bed is now configured with two types of probes: a passive Single-Tip Probe (STP) and an active Dual-Tip Probe (DTP). The STP is passive and although it can be positioned, its motion is not controlled during manipulation since it is mounted to the manual manipulator. The DTP is mounted to the computer-controlled micromanipulator and is actuated along one direction (the X axis) and can be used either for single- or two-point contact (see Fig. 9).

The control of the DTP is fully characterized by $u = (d_2, v_p, p^t)$ (see Fig. 10), denoting a push in X-direction with relative distance d_2 with duration p^t and constant speed v_p. The other two inputs are continuous.

As mentioned before, there are three sources of uncertainty. The sensing uncertainty arises because of the limitation on the magnification and resolution of the camera. Because of the 4X objective being used, each pixel subtends only 5.26 μm, the errors in positions are approximately ± 5 μm and the error in estimating the orientation of a 1616 $\mu\mathrm{m} \times 837$ μm part is $\pm 0.3°$. The control uncertainty exists only in the probe position. The errors in probe position relative to the part are also of the order of ± 5 μm. Errors in geometric parameters stem from manufacturing imperfections. The part is not a perfect rectangle, as shown in Fig. 1. The tips of the DTP are of different lengths, in which one tip is longer than the other, reflected in the angle β in Fig. 10 (bottom). However, it is assumed the exact dimensions are known. The principal source of modeling error stems from surface friction and the coefficient of friction between the probe(s) and the part.

B. System Dynamics

Again, a quasi-static model for the system is used. It is assumed that the support plane is uniform and all pushing motions

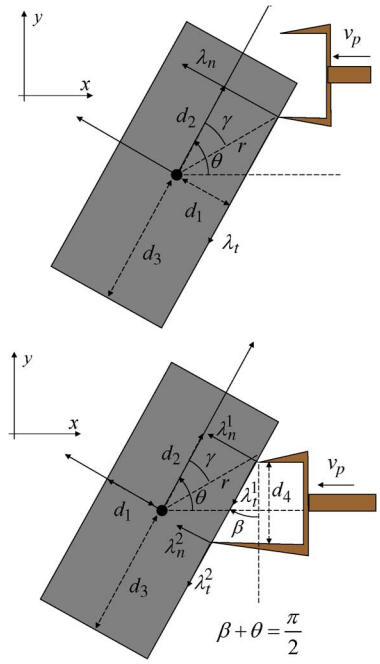

Fig. 10. Pushing with one-point (top) and two-point contact (bottom). In the bottom picture, the DTP is shown with the exaggerated misalignment between its two tips for better visualization.

of the probes are parallel to this plane. The most important assumption is about the support friction. Rather than using a three-point support model, a viscous damping model is used. Because the support surface is coated with oil (Extra Heavy Mineral Oil, LSA, Inc.), it is reasonable to assume viscous damping at the interface. Based on experimental data, the model $f = Ev$ is chosen in which $v = [v_x, v_y, v_\theta]^T$ is the velocity of the part (peg) with configuration x, y, θ; f is the corresponding vector of forces and moments; and E is the damping diagonal matrix with diagonal elements e_x, $e_y = e_x$, and e_θ. The coefficient of friction between the probe and the part is μ. These parameters are computed by parameter fitting with experimental results (see Section X-A). Finally, it is assumed that the only contacts that occur are between the probe and the part. Although the assembly task is considered as the goal, only the problem of guiding the part into the designated slot without any collisions with the environment is being considered at this time.

From quasi-static analysis, one obtains $Ev = \sum_i (w_n^i \lambda_n^i + w_t^i \lambda_t^i)$, where w denotes the wrench vector and λ the magnitude of the contact force, with subscripts n and t indicating normal and tangential directions, and the superscript i denoting the i^{th} contact. Again, complementarity constraints for sticking, sliding and separation are included in the model [8], [54].

C. Properties of Motions and Admissible Sets

There are many properties of interest for pushing primitives for the mesoscale manipulation task at hand, e.g., inputs that guarantee particular motions on of a part. Three such properties are used here from which robust motions can be systematically constructed. The first property is to maintain the one-point sticking contact with counter clockwise (or clockwise) rotation.

The second property is to maintain the two-point sticking contact for the DTP. The third property is that the orientation of the final state of the motion is close to 0 or π radians (because the slot is horizontally oriented). Sufficient conditions for motion primitives that guarantee each of these properties are presented next.

1) One-Point Sticking Contact With Counter Clockwise Rotation: Only the case in which $\theta \in (0, \pi)$ and the probe pushes on the long edge of the part (see Fig. 10 top) is considered here. However, other cases, such as pushing on the short edge or the left side of the part, can be analyzed similarly.

The following provides the conditions for a static point:

$$\left|\frac{\lambda_n}{\lambda_t}\right| = \left|\frac{d_1 d_2 e_x \cos\theta + \left(e_\theta + d_1^2 e_x\right)\sin\theta}{\left(e_\theta + d_2^2 e_x\right)\cos\theta + d_1 d_2 e_x \sin\theta}\right| > \frac{1}{\mu}$$

$$\lambda_n = -\frac{e_x v_p \left(d_1 d_2 e_x \cos\theta + \left(e_\theta + d_1^2 e_x\right)\sin\theta\right)}{f_d(x,u,E)} > 0$$

$$v_\theta = \frac{e_x v_p (d_1 \cos\theta - d_2 \sin\theta)}{f_d(x,u,E)} > 0 \tag{5}$$

in which $f_d(x,u,E) = e_\theta + (d_2^2 + d_1^2)e_x > 0$, $v_p < 0$, λ_n is the perpendicular contact force, and λ_t is the tangential contact force. In order to form a sticking contact, the ratio of λ_n/λ_t must be less than $1/\mu$, in which μ is the friction coefficient.

From (5), one can infer the property of the whole motion just from its initial point, which is stated in the following lemma.

Lemma 1: If the part starts a counter clockwise rotation with sticking contact at the initial point with orientation $\theta \in (0, \pi)$ [satisfying (5)], as shown in Fig. 10 (top), then the part will keep counter clockwise rotation with sticking contact until its orientation θ reaches

$$\pi - \max\left\{\tan^{-1}\frac{d_1}{d_2}, \tan^{-1}\frac{d_1 d_2 e_x}{e_\theta + d_1^2 e_x}\right\}. \tag{6}$$

Therefore, we need to: (1) choose a set of pushing contact points using uniform sampling with dispersion along the edge being pushed; (2) for each pushing contact point: (2a) check whether this push is sticking under the nominal condition; (2b) check whether the push is still sticking when all the combination of maximal uncertainties are involved; and (2c) if the push is always sticking, then this nominal push is robust and reported; and (3) if all the pushing points in the current sample set are not robust, decrease the sampling dispersion to get a denser set of pushing points and repeat steps (2a–2c). The iteration will stop when the sampling dispersion is no larger than the maximal uncertainty.

Proof: The derivatives of λ_n/λ_t and v_θ with respect to θ are as follows:

$$\frac{\partial(\lambda_n/\lambda_t)}{\partial\theta} = \frac{e_\theta\left(e_\theta + \left(d_1^2 + d_2^2\right)e_x\right)}{\left(\left(e_\theta + d_2^2 e_x\right)\cos\theta + d_1 d_2 e_x \sin\theta\right)^2} \tag{7}$$

$$\frac{\partial v_\theta}{\partial\theta} = -\frac{e_x v_p (d_2\cos\theta + d_1\sin\theta)}{e_\theta + \left(d_1^2 + d_2^2\right)e_x}. \tag{8}$$

It can be observed that both derivatives are strictly positive before θ reaches (6). Therefore, if the part rotates counterclockwise ($v_\theta > 0$) in the sticking mode ($|\lambda_n/\lambda_t| > (1/\mu)$) at the initial point, then the part will keep staying in the sticking mode because λ_n/λ_t will keep increasing and v_θ will keep strictly positive as θ increases. ■

2) The Two-Point Sticking Contact: Only the case in which $\theta \in (0, \pi)$ and the DTP pushes on the long edge of the part and the contact is sticking is described here (see Fig. 10 bottom).

The following equations ensure that the two point contact will be sticking at a static point:

$$\left|\frac{\lambda_n}{\lambda_t}\right| = |\tan\theta| = |1/\tan\beta| > \frac{1}{\mu}$$

$$\lambda_n^1 = \frac{e_x v_p \cos\beta(d_1\sin\beta + d_2\cos\beta - d_4)}{d_4} > 0$$

$$\lambda_n^2 = -\frac{e_x v_p \cos\beta(d_1\sin\beta + d_2\cos\beta)}{d_4} > 0. \tag{9}$$

The following lemma shows whether the whole motion has a two-point sticking contact can be determined from the initial point.

Lemma 2: If the part starts with two-point sticking contact, as shown in Fig. 10 (bottom), then the pushing will stay in the two-point sticking contact mode.

Proof: It is because (9) depends on the orientation and the orientation is invariant when the initial point has the two-point sticking contact. ■

3) The Orientation of the Final State is Close to 0 or π Radians: This property will be achieved in a motion by pushing the part with the active DTP with a separation

$$d_v \geq d_w + 2s_x^u + 2c_u^u \tag{10}$$

to the passive STP to guarantee the intended rotation under sensing and control uncertainties (see Fig. 11 left). Such pushing will ensure that the final orientation will be in θ_t-neighborhood of π, in which

$$\theta_t = \sin^{-1}\frac{d_w + 2s_x^u + 2c_u^u d}{\sqrt{d_w^2 + d_l^2}} - \alpha. \tag{11}$$

Remark: In order to guarantee existence, the pointed cone assumption requires $d_v \geq d_w \tan\sigma$, where $\sigma = tan^{-1}\mu$ is the angle of the friction cone. This is clearly satisfied by (10). However, for this motion, one cannot guarantee uniqueness of the resulting trajectory. In this case, the property of interest (the desired change in orientation) does not depend on the specifics of the trajectory and thus the lack of a guarantee on uniqueness is not a problem.

D. Computing Robust Motions From the Admissible Sets

Generally, a simple sampling-based algorithm is used to find a robust motion with respect to a given property at a given state. The sampling dispersion is incrementally decreased along each dimension of the input space until the dispersion reaches the respective control uncertainty bounds. Initially, the sampling dispersion in each dimension of the input space is chosen to be the half of the maximal distance. In each iteration, the sample points in each dimension with respect to its current sampling dispersion are combined to generate all possible inputs. Each input is tested for membership in the admissible set under the nominal state and parameters. If no input is in the admissible set, then the sampling dispersion is decreased by half and the algorithm

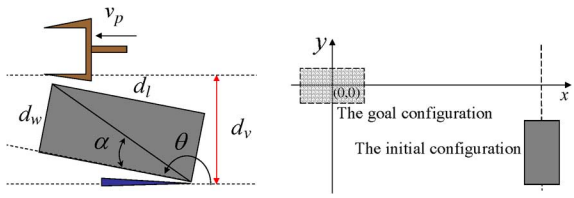

Fig. 11. Robust rotational motion (left) and planning problem setup (right).

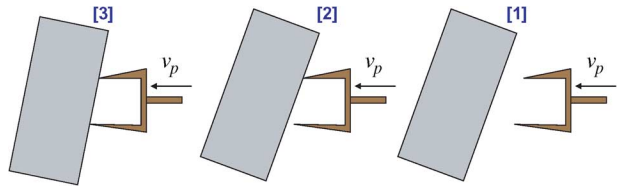

Fig. 12. The t^2 motion starting from the right figure to the left.

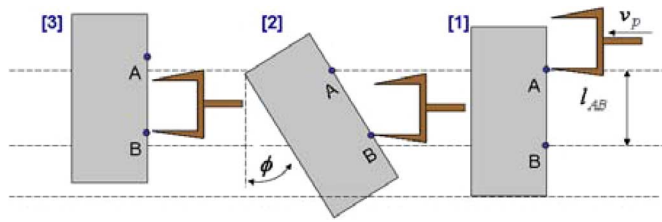

Fig. 13. Vertical translational motion starting from the right figure to the left.

goes into the next iteration. If an input is in the admissible set under the nominal state and parameters, then this input is tested to see whether it is still in the admissible set under the maximal uncertainties in sensing, control, and parameters. If yes, then the motion from this input is returned as a robust motion. If no robust motion is found when the algorithm stops, then there exists no robust motion with respect to the given property under such uncertainties. This method could be expensive computationally if the uncertainty space is high dimensional. In that case, human intuition could provide some heuristics for searching robust motions.

E. Planning With Robust Motion Primitives

The assembly task considered here has the initial configuration with orientation $\pi/2$, a goal position of (0, 0), and a goal orientation of π, with position tolerance of $\epsilon_p = 76\ \mu$m and orientation tolerance $\epsilon_\theta = 5^\circ$ (see Fig. 11 right). Ideally, there will be two DTP in the system, one on each side of the part, and thus no requirement on the x, y position of the part in the initial configuration would be needed and only the initial orientation requirement necessary for defining the problem. However, due to the limitations in our experimental platform, the peg can only be pushed with a single DTP in one direction along the X axis. Thus, in the experiments, we choose the initial x, y position of the peg to ensure that a single DTP is able to complete the task. Note: obstacles in the environment are currently being ignored. For such a task, the planning algorithm relies on composing the simple robust motions defined above. The following three higher level robust motion primitives are first constructed using these simple ones.

1) Robust Translation in the X-Direction: This robust translation is achieved by using the DTP to push the part in the X-direction, while maintaining two-point sticking contact. However, because the two tips of a DTP may not be aligned (see β in Fig. 10 bottom) or sensing errors exist, two-point contact might not be established, or can only be established after the one-point contact is established first at either the top or the bottom tip. To increase robustness, a *to-two-contact* property is defined, denoted as t^2, by a sequential composition of a one-point sticking contact motion with a counter clockwise rotation followed by a two-point sticking contact motion (see Fig. 12). Lemmas 1 and 2, respectively, provide conditions for one-point and two-point sticking contact motions. The following lemma will ensure that two sticking motions can be combined to ensure a t^2 motion.

Lemma 3: Assume that the top tip first establishes the contact. When the misalignment parameter, β, of the DTP satisfies

$$|\tan\beta| < \min\left\{\mu, \frac{d_2}{d_1}, \frac{d_1 d_2 e_x}{e_\theta + d_2^2 e_x}\right\}; \quad \beta + \theta < \frac{\pi}{2} \qquad (12)$$

the counterclockwise rotation with one-point sticking contact can be followed by a two-point sticking motion. Note: definitions for the geometrical parameters in (12) are found in Fig. 10 (bottom); μ = coefficient of friction between the probe and part, while e_x and e_θ are components of the damping matrix, as described in Section IX-B.

Proof: The first inequality in (12) ensures that two-point sticking contact is admissible and can be established before the one-point sticking contact motion stops. The second inequality ensures that a counter clockwise rotation with one-point sticking contact will precede the two-point sticking contact motion. ■

2) Robust Translation in the Y-Direction: This translation is achieved by composing a robust motion with one point sticking contact and intended rotation followed by a robust t^2 motion (see Fig. 13). The amount of the net vertical translation is $l_{AB}(1-\cos\phi)$ under nominal conditions (no uncertainty).

3) Robust Rotation: This motion is achieved with the pushing described in Section IX-C3.

Planning algorithm

Utilizing these three higher level robust motion primitives, the planning algorithm consists of the following steps:

Step 1) *Move in the Y direction by pushing along the long edge of the part such that $Y \in [-(\epsilon_p/2), \epsilon_p/2]$.* A sequence of Y-direction motions is used, guaranteeing that the net Y translation of $l_{AB}(1-\cos\phi)$ will result (Fig. 13). It will have the following error bound $d_v^u = \max\{d_1^e, d_2^e\}$, in which $d_1^e = |l_{AB}(1-\cos\phi) - (l_{AB} - 2s_p^u - 2c_p^u)(1 - \cos(\phi - 2s_\theta^u))|$, $d_2^e = |l_{AB}(1-\cos\phi) - (l_{AB} + 2s_p^u + 2c_p^u)(1-\cos(\phi + 2s_\theta^u))|$, s_p^u, and c_p^u are, respectively, the sensing and control error bounds in the position, and s_θ^u is the sensing error bound in the orientation. To ensure that $Y \in [-(\epsilon_p/2), \epsilon_p/2]$ can be achieved using the vertical primitive under sensing and control uncertainties, the following conditions on the uncertainty bounds must be satisfied: $s_p^u + d_v^u \le (\epsilon_p/2)$, $\phi > 2s_\theta^u$, $l_{AB} > 2s_p^u + 2c_p^u$.

Step 2) *Rotate to* $\theta = \pi$. As shown in (11) and Fig. 11 (left), the distance of the orientation of the part to the horizontal line will be bounded. To ensure that the final t^2 pushing can be robustly applied, it is required that uncertainty bounds satisfy: $\theta_t = \sin^{-1}((d_w + 2s_p^u + 2c_p^u)/\sqrt{d_w^2 + d_l^2}) - \alpha < \theta_{t^2}^{\max}$, in which $\theta_{t^2}^{\max}$ is the maximal orientation of the part allowing a robust t^2 pushing.

Step 3) *If necessary, move in the* Y *direction by pushing along the short edge of the part such that* $Y \in [-(\epsilon_p/2), \epsilon_p/2]$.

Step 4) *Translate the part in* X *direction to the goal (0, 0)*. With the robust t^2 motion primitives, the final configuration of part will be $X \in [p^x + r\cos(\gamma+\beta) - c_p^u, p^x + r\cos(\gamma+\beta) + c_p^u]$, $Y \in [p_y - r\sin(\gamma+\beta) - c_p^u, p_y - r\sin(\gamma+\beta) + c_p^u]$, and $\theta = \beta$ in which p^x, p^y is the position of the top tip of the DTP, d_2, r and γ are as shown in Fig. 10 (bottom). These equations also impose restrictions on the uncertainty bounds to ensure the intended tolerance: $r_{\max}(\cos(\gamma_{\max} - \beta_{\max}) - \cos\gamma_{\max}) + 2c_p^u < \epsilon_p$ and $r_{\max}\sin\beta_{\max} + 2c_p^u < \epsilon_p, \beta_{\max} < \epsilon_\theta$, in which $\gamma_{max} = \tan^{-1}(d_1/d_3)$, $r_{\max} = \sqrt{d_1^2 + d_3^2}$, and $\beta_{\max}$ is the maximal magnitude for β.

X. Simulation and Experimental Results

A. Estimating of System Parameters

The parameter fitting was done with the experimental data obtained using the STP in a similar manner as before. The diagonal elements of damping matrix E are estimated to be $e_x = e_y = 160.89N \cdot \text{sec.}/\text{m}$ and $e_\theta = 60.64N \cdot m \cdot \text{sec}$. The coefficient of friction between the part and the probe is estimated to be $\mu = 0.30 \sim 0.36$. Results show $30 \sim 40$ μm position errors across a translation of about 600 μm and about 3° orientation errors for a 45° rotation.

It should be noted that the model is fit once for each part/substrate set. With our approach, the same model can be used with parts that have the same nominal geometry on the same substrate. If the nominal part geometry or substrate changes, then a new model should be fit. Although not in the scope of this work, this model fitting procedure can be automated to limit the overhead associated with this. For example, once a new part is identified (or new substrate installed) a series of free rotation pushes on the part can be automatically administered, trajectories recorded, and model parameters fit for this particular scenario.

B. Comparison Between Robust and Nonrobust Motions

Trajectories from robust motion primitives show less variation (and are therefore more predictable) than trajectories from other motion primitives. Fig. 14 shows the experiment setup (top) and experimental trajectory plots for comparison of the robust and nonrobust motions using the DTP and STP. Tests 1 and 2 are for robust and nonrobust t^2 motions with the DTP. Test 1 was verified to satisfy the robust t^2 motion conditions for *One-point sticking contact with CCW rotation* and *Two-point sticking contact*. The experiments showed that the two-point contact is well maintained because the orientation θ (bottom plot) is almost constant after the two point contact is established. Test 2 did not satisfy the two-point sticking contact conditions, and therefore the two-point contact was broken once it was established (nonconstant, decreasing θ). It was also observed that Test 1 has maximal trajectory differences of 20 μm in X, 15 μm in Y, and 0.023 radians in θ, which are smaller than the corresponding numbers for Test 2 (maximal trajectory differences at 15 μm in X, 25 μm in Y, and 0.1 radians in θ). Test 3 shows results for the robust CCW rotation with one-point sticking contact primitive using the STP.

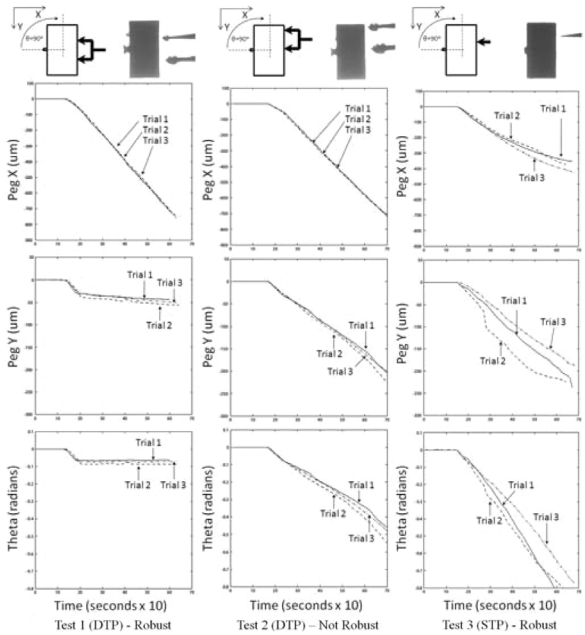

Fig. 14. Experimental results for robust and nonrobust motions with the DTP and the STP.

TABLE III
Y-Translation Primitive: Net Displacement of the Part

Test No.	X (μm)	Y (μm)	θ
1	1996	19	0.6°
2	1975	18	0.5°
3	1559	20	1.1°
Average	1843	19	0.7°
Simulation	1443	11	0.0°
Theory	*NA*	11	0.0°

C. Planning in Both the Simulation and Experiment

Table III shows the comparison between theoretical, simulated and experimental results for robust translation in the Y-direction primitive. Tables IV and V compare the experimental and simulated results for executing robust rotational motion and robust translation in the X-direction motion, respectively. At least three experimental tests were done for each motion type and the average values of the tests are shown in the tables.

For the robust Y translation tests, the initial robust one-point sticking contact is maintained until a desired part rotation angle, ϕ, is achieved. This is then followed by a robust t^2 push to restore the part to the upright position. Simulation and theoretical

TABLE IV
ROTATIONAL MOTION: NET DISPLACEMENT OF THE PART

Test No.	$X\,(\mu\mathrm{m})$	$Y\,(\mu\mathrm{m})$	θ
1	381	34	88°
2	434	32	90°
3	370	14	90°
Average	382	27	89°
Simulation	295	0.05	90°

TABLE V
X-TRANSLATION PRIMITIVE: NET DISPLACEMENT OF THE PART

Test No.	$X\,(\mu\mathrm{m})$	$Y\,(\mu\mathrm{m})$	θ
1	954	5	0.1°
2	944	11	0.7°
3	965	11	0.7°
4	959	5	0.5°
5	954	0	0.0°
Average	955	6	0.4°
Simulation	949	0.2	0.0°

results match very well for the ϕ values tested (Table III). Experiments show a somewhat higher (7–9 $\mu\mathrm{m}$) net displacement than the predicted Y translation, but it is likely due to measurement errors—errors in estimating position are $\pm 5\ \mu\mathrm{m}$. Sliding was not observed in the pushing from the image analysis. This Y-motion is obviously coupled with X-direction motion of the part. There are no theoretical predictions for this motion since our primitives only guarantee that one or two-point sticking contacts will occur and not precisely when they will occur. However, the average values for this X-direction motion from the experiments are comparable to those predicted in simulation.

In the robust rotational motion experiments, the separation distance is determined from (10) and the STP and DTP are centered about the center of the part that has orientation $\pi/2$. The STP probe is to the left of the part and is held stationary. The DTP, on the right side, pushes the part with its bottom probe tip for a distance of about 1100 $\mu\mathrm{m}$. From these experiments, one can see that the orientation of the peg is robustly rotated close to π even though uncertainties can cause significant mismatches in X and Y displacements (Table IV).

A push of approximately 950 $\mu\mathrm{m}$ was used for the robust X translation experiments. The predicted results are within the error margins of the experimental observations (Table V).

Combining these three types of robust motions together allows for the execution of the planned algorithm described in Section IX-E for the task shown in Fig. 11 (right). Because of the limited controllability of the X position of the peg in the current experimental platform and planning algorithm, the initial configuration is set to be $(x_{\mathrm{init}} = 2060.4\ \mu\mathrm{m}, y_{\mathrm{init}} = -9.2\ \mu\mathrm{m}, \theta_{\mathrm{init}} = \pi/2)$ such that one robust Y motion followed by one robust rotation followed by one robust X motion is able to push the peg into the hole. If the part Y-coordinate is not in the neighborhood of the goal Y-coordinate after the robust rotation primitive, the algorithm will call for and execute more robust Y-moves until the part is in this neighborhood. Then, the robust X-motion will be performed to move the part into the hole.

A simulation of the planned motion is shown in Fig. 15 (top). In the experiments, the peg is successfully pushed into the hole

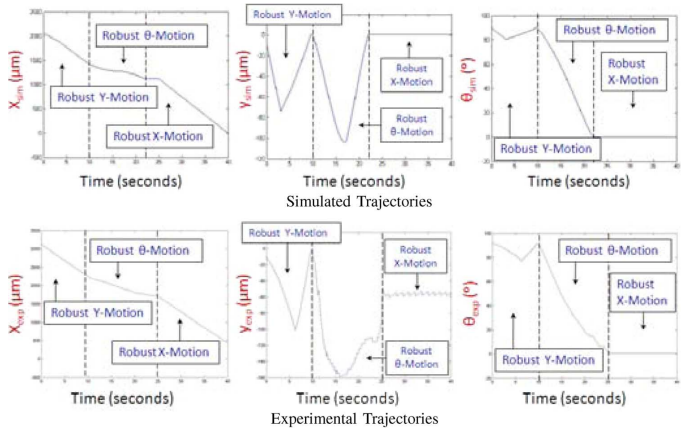

Fig. 15. Simulation (top) and experimental (bottom) results for a planning task.

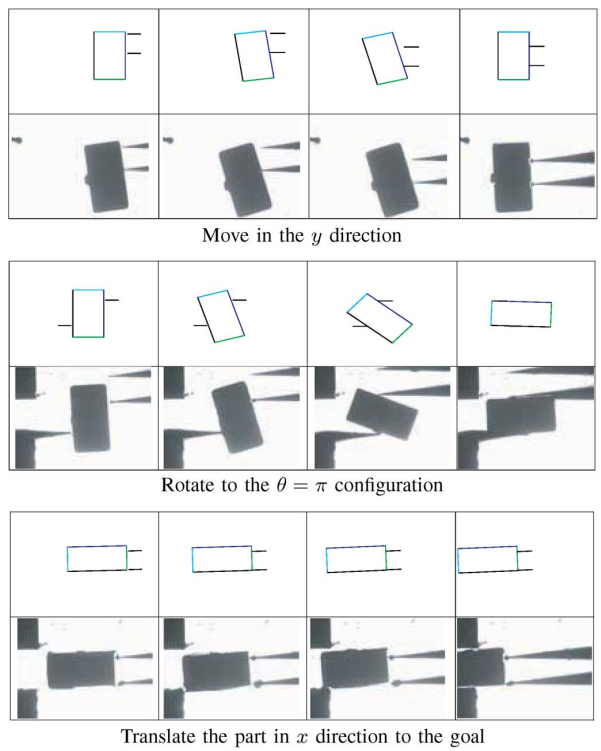

Fig. 16. Snapshots illustrating the assembly of the part into the slot.

three times over three trials. The experimental data from one trial is shown in Fig. 15 (bottom). The regions where the different robust motion primitives are being applied are labeled in the figure. Finally, the snapshots of the experiment are shown in Fig. 16 with the associated robust controls.

XI. FUTURE WORK

In our future work, we plan to extend the analysis presented here in a more general setting. Robust controls can be derived by analyzing the structure of the underlying LCP. In this sense, one can define a set of controls to be robust if the LCP solutions corresponding to these controls use the same complementarity basis and therefore no transition (such as stick-slip, take off) occurs during the time period for which the controls are applied.

Possible changes of complementarity basis can be easily verified by simulation under some standard nondegeneracy assumptions. We also plan on addressing more complicated problems associated with multipoint contacts, which characterize many assembly tasks. Contacts between the fixed obstacles can be treated the same as contacts with the stationary (passive) probe. Primitives and/or fixtures can be developed to take advantage of these contacts (i.e., self guiding fixtures) as was the case with the RRT-based planner. For example, see Fig. 7 where the top of the fixture is used to pivot the part around to change its orientation and the slot aligns the peg as it is inserted into the hole.

Finally, we intend to extend this work to the microscale. Our current model and motion primitives are derived at the mesoscale and do not consider adhesion forces between the support and part or manipulator tip and part that will be present with microscale parts 100s of microns and below in size. In order to extend this methodology to this scale, the model and primitives must be modified to incorporate these effects. Alternatively, or in conjunction with these changes, controlled environmental conditions (low friction surfaces, clean-room, humidity controlled environments, etc.) will be explored for the experimental test-bed to mitigate against these microscale sticking effects. Also, manipulations conducted with multiple active DTP's will be investigated to provide increased controllability to both the mesoscale and microscale parts. Using the point contacts to manipulate the microscale parts will decrease the amount of surface contacts on the part when compared to microgrippers and thus reduce the chances of the encountering stiction between the parts and probe tips.

XII. Summary

In the first part of this paper, we addressed the modeling, simulation, and planning of a simple assembly task in which a peg with a characteristic length of 1 mm is reoriented by pushing with a probe and inserted into a hole. We relied on vision to estimate the initial configuration of the system rather than for feedback control. An RRT algorithm was used to produce feasible open loop manipulation plans. While these plans were sometimes successful when carried out experimentally, they were not robust to error associated with the uncertainty in the support friction models or to those associated with the initial positioning of the peg. An intuitive open loop motion plan was generated to increase robustness and executed experimentally to successfully accomplish the task.

In the second half of this paper, a framework for motion planning under differential constraints and uncertainties in sensing, control (actuation), and geometric/dynamic parameters has been established. A way to characterize robust motion primitives with applications to quasi-static manipulation and assembly tasks is shown and measures to quantify robustness of motion primitives is proposed. Further, an algorithm to automatically synthesize quasi-open loop motion plans which sequentially compose robust motion primitives to move parts to goal positions with minimal actuation is described.

The main contribution of this new framework is the quantitative treatment of uncertainty and the incorporation of models of uncertainty into the synthesis of motion primitives and the motion plans. It does not address the problems associated with multipoint contact which characterize assembly tasks. The modeling of the contact friction has been simplified by considering lubricated surfaces which appear to be well modeled by viscous damping. Nevertheless, the ability to plan and reliably execute the plan for positioning and orienting parts using visual feedback with only a single degree-of-freedom actuator represents a significant accomplishment over previous studies on quasi-static manipulation. These methods can be generally applied to more complicated meso-manipulation and assembly problems and be extended to the microscale with the addition of adhesion force modeling.

References

[1] K. Goldberg, "Orientating polygonal parts without sensing," *Algorithmica*, vol. 10, no. 2/3/4, pp. 210–225, Aug.–Oct. 1993.

[2] M. Erdmann and M. Mason, "An exploration of sensorless manipulation," *IEEE J. Robot. Autom.*, vol. 4, no. 4, Aug. 1998.

[3] M. Moll, K. Goldberg, M. Erdmann, and R. Fearing, "Orienting micro-scale parts with squeeze and roll primitives," in *Proc. IEEE Int. Conf. Robot. Autom.*, Washington, DC, May 11–15, 2002, vol. 2, pp. 1931–1936.

[4] M. Mason, "Manipulator grasping and pushing operations," Ph.D. dissertation, Massachusetts Inst. Technol., Cambridge, MA, 1982.

[5] M. Mason, "Mechanics and planning of manipulator pushing operations," *Int. J. Robot. Res.*, vol. 5, no. 3, pp. 53–71, 1986.

[6] M. Peshkin and A. Sanderson, The motion of a pushed, sliding object, Part 1: Sliding friction Robotics Inst., Carnegie Mellon Univ., Pittsburgh, PA, Tech. Rep. CMU-RI-TR-85-18, Sep. 1985.

[7] P. Song, J. Pang, and V. Kumar, "A semi-implicit time-stepping model for frictional compliant contact problems," *Int. J. Numerical Methods Eng.*, vol. 60, no. 13, pp. 2231–2261, Aug. 2004, accepted for publication.

[8] J. Trinkle, S. Berard, and J. Pang, "A time-stepping scheme for quasistatic multibody systems," in *Proc. Int. Symp. Assembly Task Planning*, Jul. 2005, pp. 174–181.

[9] H. Choset, K. M. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. E. Kavraki, and S. Thrun, *Principles of robot motion: Theory, algorithms, and implementations*. Cambridge, MA: MIT Press, 2005.

[10] S. M. LaValle, *Planning Algorithms*. Cambridge, U.K.: Cambridge Univ. Press, 2006. [Online]. Available: http://planning.cs.uiuc.edu/

[11] K. Lynch, "The mechanics of fine manipulation by pushing," in *Proc. IEEE Int. Conf. Robot. Autom.*, Nice, France, May 1992, pp. 2269–2276.

[12] K. Lynch and M. Mason, "Stable pushing: Mechanics, controllability, and planning," *Int. J. Robot. Res.*, vol. 15, no. 6, pp. 553–556, Dec. 1996.

[13] J. Alexander and J. Maddocks, "Bounds on the friction-dominated motion of a pushed object," *Int. J. Robot. Res.*, vol. 12, no. 3, pp. 231–248, June 1993.

[14] D. Pham, K. Cheung, and S. Yeo, "Initial motion of a rectangular object being pushed or pulled," in *Proc. IEEE Int. Conf. Robot. Autom.*, 1990, vol. 2, pp. 1046–1050.

[15] S. Akella and M. T. Mason, "Posing polygonal objects in the plane by pushing," *Int. J. Robot. Res.*, vol. 17, no. 1, pp. 70–88, Jan. 1998.

[16] P. Agarwal, J. Latombe, R. Motwani, and P. Raghavan, "Nonholonomic path planning for pushing a disk among obstacles," in *Proc. IEEE Int. Conf. Robot. Autom.*, 1997, vol. 4, pp. 3124–3129.

[17] O. Ben-Shahar and E. Rivlin, "To push or not to push: On the arrangement of movable objects by a mobile robot," *IEEE Trans. Syst., Man, Cybern.-Part B: Cybern.*, vol. 28, no. 5, pp. 667–679, Oct. 1998.

[18] Z. Balorda, "Reducing uncertainty of objects by robot pushing," in *Proc. IEEE Int. Conf. Robot. Autom.*, 1990, pp. 1051–1056.

[19] R. Fearing, "Survey of sticking effects for micro parts handling," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, Pittsburgh, PA, Aug. 5–9, 1995, vol. 2, pp. 212–217.

[20] K. Boehringer, R. Fearing, and K. Goldberg, "Microassembly," in *Handbook of Industrial Robotics*, 2nd ed. New York: Wiley, 1999, pp. 1045–1066.

[21] A. Menciassi, A. Eisinberg, I. Izzo, and P. Dario, "From "macro" to "micro" manipulation: Models and experiments," *IEEE/ASME Trans. Mechatronics*, vol. 9, no. 2, pp. 311–320, Jun. 2004.

[22] F. Arai, Y. Nonoda, T. Fukuda, and T. Oota, "New force measurement and micro grasping method using laser raman spectrophotometer," in *Proc. IEEE Int. Conf. Robot. Autom.*, April 1996.
[23] W. Zesch and R. Fearing, "Alignment of microparts using force controlled pushing," in *Proc. SPIE Conf. Microrobotics Micromanipulation*, 1998, pp. 148–156.
[24] J. Thompson and R. Fearing, "Automating microassembly with orthotweezers and force sensing," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, Maui, HI, USA, Oct.–Nov. 2001, vol. 3, pp. 1327–1334.
[25] A. Ferreira, C. Cassier, and S. Hirai, "Automated microassembly system assisted by vision servoing and virtual reality," *IEEE Trans. Mechatronics*, vol. 9, no. 2, pp. 321–333, June 2004.
[26] T. Sievers and S. Fatikow, "Real-time object tracking for the robot-based nanohandling in a scanning electron microscope," *J. Micromechatronics—Special Issue on Micro/Nanohandling*, vol. 18, pp. 267–284, 2006.
[27] T. Wortmann, C. Dahmen, R. Tunnell, and S. Fatikow, "Image processing architecture for real-time micro- and nanohandling applications," in *Proc. 11th IAPR Conf. Machine Vision Applications*, 2009, pp. 418–421.
[28] B. Vikramaditya and B. Nelson, "Visually guided microassembly using optical microscopes and active vision techniques," in *Proc. IEEE Int. Conf. Robot. Autom.*, Albuquerque, NM, 1997, vol. 4, pp. 3172–3177.
[29] J. Cecil, D. Vasquez, and D. Powell, "A review of gripping and manipulation techniques for micro-assembly applications," *Int. J. Prod. Res.*, vol. 43, no. 4, pp. 819–828, February 2005.
[30] D. Cecil, D. Powell, and D. Vasquez, "Assembly and manipulation of micro devices-a state of the art survey," *Robot. Comput.-Integr. Manuf.*, vol. 23, no. 5, pp. 580–588, 2007.
[31] J. Feddema, P. Xavier, and R. Brown, "Micro-assembly planning with van der Waals force," in *Proc. IEEE Int. Symp. Assembly and Task Planning (ISATP '99)*, Jul. 1999, pp. 32–38.
[32] Y. Zhou and B. Nelson, "The effect of material properties and gripping force on micrograsping," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, San Francisco, CA, Apr. 2000, vol. 2, pp. 1115–1120.
[33] C. Keller, "Microgrippers with integrated actuator and force sensors," in *Proc. World Autom. Congr.*, 1998, pp. 217–222.
[34] J. Cecil and N. Gobinath, "Development of a virtual and physical cell to assemble micro devices," *J. Robot. CIM, Special Issue*, vol. 21, no. 4–5, pp. 431–441, Aug.–Oct. 2005.
[35] J. Alex, B. Vikramaditya, and B. Nelson, "A virtual reality teleoperator interface for assembly of hybrid MEMS prototypes," in *Proc. DETC'98 1998 ASME Eng. Tech. Conf.*, Atlanta, GA, Sep. 13–16, 1998, DETC98/MECH-5836.
[36] T. Kasaya, H. Miyazaki, S. Saito, and T. Sato, "Micro object handling under SEM by vision-based automatic control," in *Proc. IEEE Int. Conf. Robot. Autom.*, Detroit, MI, 1999, pp. 2189–2196.
[37] A. Das, P. Zhang, W. Lee, D. Popa, and H. Stephanou, "μ^3: Multiscale, deterministic micro-nano assembly system for construction of on-wafer microrobots," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Rome, Italy, Apr. 2007, pp. 461–466.
[38] K. Lynch, "Estimating the friction parameters of pushed objects," in *Proc. 1993 IEEE/RSJ Int. Conf.*, Jul. 1993.
[39] T. Yoshikawa and M. Kurisu, "Identification of the center of friction from pushing an object by a mobile robot," in *Proc. IEEE/RSJ Int. Workshop Intell. Robot. Syst. IROS*, November 1991, vol. 1, pp. 186–193.
[40] D. Cappelleri, J. Fink, and V. Kumar, "Modeling uncertainty for planar meso-scale manipulation and assembly," in *Proc. ASME 2006 Int. Design En. Tech. Conf. (IDETC), Philadelphia, PA*, 2006.
[41] M. Rakotondrabe, Y. Haddab, and P. Lutz, "Development, modeling, and control of a micro-/nanopositioning 2-dof stick-slip device," *IEEE/ASME Trans. Mechatronics*, vol. 14, no. 6, pp. 733–745, Dec. 2009.
[42] B. Armstrong-Helouvry, P. Dupont, and C. C.-D. Wit, "A survey of models, analysis tools and compensation methods for the control of machines with friction," *Automatica*, vol. 30, no. 7, pp. 1083–1138, 1994.
[43] P. Dupont, V. Hayward, B. Armstrong, and F. Altpeter, "Single state elastoplastic models," *IEEE Trans. Autom. Control*, vol. 47, no. 5, pp. 787–792, May 2002.
[44] T. Lozano-Pérez, M. Mason, and R. Taylor, "Automatic synthesis of fine-motion strategies for robots," *Int. J. Robot. Res.*, vol. 3, no. 1, pp. 3–24, 1984.
[45] M. Erdmann, "Using backprojections for fine motion planning with uncertainty," *Int. J. Robot. Res.*, vol. 5, no. 1, pp. 19–45, 1986.
[46] M. Salganicoff, G. Metta, A. Oddera, and G. Sandini, "A vision-based learning method for pushing manipulation," in *Proc. AAAI Fall Symp. Machine Learning in Computer Vision.*, 1993b, pp. 90–94.
[47] N. Melchior and R. Simmons, "Particle RRT for path planning with uncertainty," in *Proc. IEEE Intl. Conf. Robot. Autom.*, 2007, pp. 1617–1624.
[48] R. Alterovitz, T. Simon, and K. Goldberg, "The stochastic motion roadmap: A sampling framework for planning with Markov motion uncertainty," in *Proc. Robot. Sci. Syst.*, 2007.
[49] S. Koelemeijer, F. Bourgeois, and J. Jacot, "What is the best way to increase efficiency in precision assembly?," in *Proc. IFIP TC5 WG5.5 3rd Int. Precision Assembly Seminar (IPAS)*, Bad Hofgastein, Austria, Feb. 19–21, 2006.
[50] P. Cheng, D. Cappelleri, B. Gavrea, and V. Kumar, "Planning and control of meso-scale manipulation tasks with uncertainties," in *Proc. Robot. Sci. Syst.*, Atlanta, GA, Jun. 2007.
[51] [Online]. Available: www.fotofab.com.
[52] M. Anitescu and F. Potra, "Formulating dynamic multi-rigid-body contact problems with friction as solvable linear complementarity problems," *Nonlinear Dynamics*, vol. 14, no. 3, pp. 231–247, 1997.
[53] D. Stewart and J. Trinkle, "An implicit time-stepping scheme for rigid body dynamics with inelastic collisions and coulomb friction," *Int. J. Numerical Methods in Engineering*, vol. 39, pp. 2673–2691, 1996.
[54] J. Pang and J. Trinkle, "Complementarity formulations and existence of solutions of dynamic multi-rigid-body contact problems with coulomb friction," *Math. Programming*, vol. 73, pp. 199–226, 1996.
[55] S. Berard, B. Nguyen, B. Roghani, J. Trinkle, J. Fink, and V. Kumar, "Da Vinci code: A multi-model simulation and analysis tool for multibody systems," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2007. [Online]. Available: http://www.robotics.cs.rpi.edu/dvc/
[56] J. Nelder and R. Mead, "A simplex method for function minimization," *Comput. J.*, vol. 7, pp. 308–315, 1965.
[57] S. LaValle, *Rapidly-Exploring Random Trees: A New Tool for Path Planning,* 1998. [Online]. Available: citeseer.ist.psu.edu/lavalle98rapidlyexploring.html
[58] J. Kim, J. M. Esposito, and V. Kumar, "An RRT-based algorithm for testing and validating multi-robot controllers," *Robot. Sci. Syst.*, Jun. 2005.

David J. Cappelleri (M'09) received the B.S. and M.S. degrees in mechanical engineering from Villanova University, Villanova, PA, and from The Pennsylvania State University, University Park, and the Ph.D. degree from the University of Pennsylvania, Philadelphia, in mechanical engineering and applied mechanics in 2008. He worked in the medical device industry for three years before returning to school to earn the Ph.D. degree.

He is currently an Assistant Professor in the Department of Mechanical Engineering at Stevens Institute of Technology, Hoboken, NJ.

Prof. Cappelleri is a recipient of the Harvey N. Davis Distinguished Assistant Professor Teaching Award in 2010 and the Association for Lab Automation (ALA) Young Scientist Award for his paper at IEEE CASE 2009. He is an elected member of the IEEE Robotics and Automation Society Technical Committee on Micro/Nano Robotics and Automation, American Society of Mechanical Engineers (ASME) Design Engineering Division Mechanisms and Robotics Committee, and the ASME Design Engineering Division Micro and Nanosystems Technical Committee.

Peng Cheng received the B.S. and M.S. degrees in electrical engineering from Tsinghua University, Beijing, China, in 1996 and 1999, respectively, and the Ph.D. degree in computer science from the University of Illinois at Urbana–Champaign, Urbana, in 2005, and the M.S. degree in computer science from Iowa State University in 2001.

Currently, he works at MathWorks, Inc. in the area of physical modeling. From 2005 to 2008, he worked as a Postdoctoral Fellow and Research Scientist in the GRASP Laboratory, University of Pennsylvania.

Jonathan Fink (S'06) received the B.S. degree in electrical and computer systems engineering from Rensselaer Polytechnic Institute, Troy, NY. Currently, he is working towards the Ph.D. degree at the Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia.

His research interests include collaboration and planning for multi-robot systems.

Bogdan Gavrea He received the Ph.D. degree in applied mathematics from the University of Maryland, Baltimore, in 2006.

Currently, he is an Assistant Professor in the Department of Mathematics, Faculty of Automation and Computers Science, Technical University of Cluj-Napoca. His research interests are in the area of numerical optimization, stochastic programming, numerical analysis and mathematical modeling and simulation in robotics.

Vijay Kumar (M'86–F'05) received the Ph.D. degree in mechanical engineering from The Ohio State University, Columbus, in 1987.

He is the UPS Foundation Professor and the Deputy Dean for Education in the School of Engineering and Applied Science, University of Pennsylvania. He has been on the Faculty in the Department of Mechanical Engineering and Applied Mechanics with a secondary appointment in the Department of Computer and Information Science, University of Pennsylvania since 1987.

Dr. Kumar is a Fellow of the American Society of Mechanical Engineers (ASME) and the Institution of Electrical and Electronic Engineers (IEEE). He has served on the editorial boards of the IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION, the *Journal of Franklin Institute*, the IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING, *ASME Journal of Mechanical Design*, the *ASME Journal of Mechanisms and Robotics*, and the Springer *Tracts in Advanced Robotics* (STAR). He is the recipient of the 1991 National Science Foundation Presidential Young Investigator Award, the Lindback Award for Distinguished Teaching, and the 1997 Freudenstein Award for significant accomplishments in mechanisms and robotics. He has won best paper awards at DARS 2002, ICRA 2004, ICRA 2008 RSS 2009, and DARS 2010. He is a Distinguished Lecturer in the IEEE Robotics and Automation Society and an elected member of the Robotics and Automation Society Administrative Committee.