Thomas J. Debus
Pierre E. Dupont

Department of Aerospace and Mechanical Engineering
Boston University
Boston, MA 02445, USA

Robert D. Howe

Division of Engineering and Applied Sciences
Harvard University
Cambridge, MA 02138, USA

Contact State Estimation using Multiple Model Estimation and Hidden Markov Models

Abstract

In this paper we present an approach to estimating the contact state between a robot and its environment during task execution. Contact states are modeled by constraint equations parametrized by time-dependent sensor data and time-independent object properties. At each sampling time, multiple model estimation is used to assess the most likely contact state. The assessment is performed by a hidden Markov model, which combines a measure of how well each set of constraint equations fits the sensor data with the probability of specific contact state transitions. The latter is embodied in a task-based contact state network. The approach is illustrated for a three-dimensional peg-in-hole insertion using a tabletop manipulator robot. Using only position sensing, the contact state sequence is successfully estimated without knowledge of nominal property values. Property estimates are obtained for the peg dimensions as well as the hole position and orientation.

KEY WORDS—contact estimation, contact modeling, hidden Markov model, machine perception, multiple model estimation, task network

1. Introduction

One aspect of machine perception is the automatic determination of contact states and the object properties associated with those states. Such information is useful in a broad range of manipulation problems. A first category involves those tasks for which the current contact state dictates the motion or control strategy to be applied. Force controlled assembly belongs to this class of problem for which knowledge of both contact states and property values is important. Significant progress for this application has been made by De Schutter et al. (1999) and Lefebvre, Bruyninckx, and De Schutter (2003).

A second category involves those tasks for which machine perception could enhance the overall task performance. In current teleoperation applications for example, the operator is responsible for interpreting sensor feedback from the remote manipulator. Subsequent manipulation strategies are based on this interpretation. In many cases, however, machine perception of the remote environment could enhance the operator's perception of the environment and thereby dramatically improve task performance (Debus et al. 2001). For example, in remediation of toxic waste dumps, quantitative measurements of the size and weight of the containers helps to infer their contents and to determine optimal handling strategies (Griebenow 1994). Additional application areas include undersea mining and salvage, interplanetary exploration, and the defusing of explosives.

In this paper, we investigate a formalism for a perceptual system based on contact state estimation. The approach assumes known (i) a task description, expressed as a network of contact states, and (ii) contact state descriptions, using constraint equations, based on the available set of sensors, and parametrized by unknown object properties. Nominal parameter values of the objects in contact are not assumed. Thus, the contact state of a cylinder sliding on a plane can be distinguished regardless of the cylinder's radius and an estimate of the unknown radius is also obtained. The approach is demonstrated in this paper using only kinematic measurements collected during object manipulation to estimate contact states and their associated local geometric properties (e.g., dimension, orientation, location).

The paper is arranged as follows. In the next section we review the framework employed to estimate contact states and object properties from sensor data. Each part is explained and related prior work is presented. In the following section we

The International Journal of Robotics Research
Vol. 23, No. 4–5, April–May 2004, pp. 399-413,
DOI: 10.1177/0278364904042195

describe how contact states of objects can be described using sets of parametrized position-based constraint equations. Next, property estimation is discussed and the concept of multi-pass state estimation is introduced to improve equation conditioning. In Section 5, multiple model estimation using contact-point error distances as inputs of a hidden Markov model (HMM) is presented as a means to estimate the active contact states. In the subsequent section we describe an experimental evaluation of this approach. A three-dimensional peg-in-hole insertion is carried out and the contact state sequence between the peg and the hole is estimated as well as the geometrical properties of the peg and hole. Conclusions are presented in the final section of the paper.

2. Approach

The work described in this paper builds on the framework for a robotic perceptual system first proposed in Dupont et al. (1999). The perceptual system has as its inputs a description of the manipulation task as a sequence of contact states, the set of object properties to identify and the available sensor signals (e.g., position, force, vision). Its outputs are the sequence of contact states and their associated properties. These can include object shape, mass distribution, compliance, surface friction and texture. In this paper, however, only kinematic sensors will be considered; and as a consequence only geometric properties (i.e., dimension and orientation) will be discussed.

The structure of the perceptual system is motivated by four fundamental observations involving contact states, constraint equations, and task descriptions.

(i) *Contact states describe how two objects are in contact.* In this paper, a geometrical description of contacts is used to characterize their state. As an example, Figure 1 illustrates a possible set of contact states associated with a peg-in-hole insertion task. As pictured, contact states can consist of no contact between the objects (C_1), a single contact (C_2 and C_3) or multiple contacts (C_4).

(ii) *Object properties are measurable only in certain contact states*. The mechanics of the contacts and the available sensing modalities determine when each property is accessible. For example, the weight of an object can be estimated from robot force measurements only when it is freely supported by the robot, and the friction coefficient between two surfaces can only be estimated when they are sliding against each other.

(iii) *Each contact state can be described by a set of constraint equations.* A mathematical description of the physics governing the contact state, written as constraint equations, is used to relate the sensor data and the properties of the objects in contact. Classical examples of this approach include the estimation of robot kinematic and dynamic parameters. In this paper we focus on kinematic constraints arising due to contact between the manipulating object and objects in its environment.

(iv) *Tasks are sequences of contact states*. Manipulation tasks are readily described as a succession of contact states. For example, a simple pick-and-place task might include a sequence of five states, composed from a set of three distinct states: no-contact, grasp-on-table, freely-support-object, grasp-on-table, and no-contact. Tasks may be represented as a network with states as nodes and allowed transitions between states as links (e.g., Hannaford and Lee 1991; Cao and Sanderson 1994; McCarragher and Asada 1995; Rosen et al. 2002).

Using these concepts, the perceptual system is based on a contact state estimation algorithm composed of three parts: contact state modeling, property estimation and a contact state decision test. In this paper, contact state estimation is implemented using multiple model estimation. This technique is illustrated in Figure 2. Given a sequence of sensor data points, all corresponding to a particular contact state, the property estimation problem uses the contact state models to estimate properties of the manipulated objects.

For the contact state decision test, two complementary approaches are employed here. First, estimation residuals are used to judge how well a portion of the data stream corresponds to a path satisfying the constraint equations for a particular contact state. Secondly, conditional probability theory is used to adjust these results to account for knowledge of prior and anticipated contact states as embodied in the task's contact state network.

The reader can anticipate that the success of a contact state estimator will depend on the set of contacts and available sensors as well as the richness of the sensor data set. In the work presented here, only kinematic data are used to estimate local geometric properties. The exclusion of force/torque sensing is motivated by practical considerations encountered by the authors; force/torque sensing is not always used on industrial systems, due to cost and reliability concerns (e.g., undersea oil platform maintenance).

2.1. Prior Work

State estimation has been investigated in the literature in several contexts. De Schutter et al. (1999) and Lefebvre, Bruyninckx, and De Schutter (2003) have employed state estimation in the context of compliant motion control. In this application, the control law depends on the contact state and accurate knowledge of the geometric parameters improves controller performance. Pook and Ballard (1993) employed state estimation in order to understand the qualitative control characteristics of an example task performed on a teleoperated system.

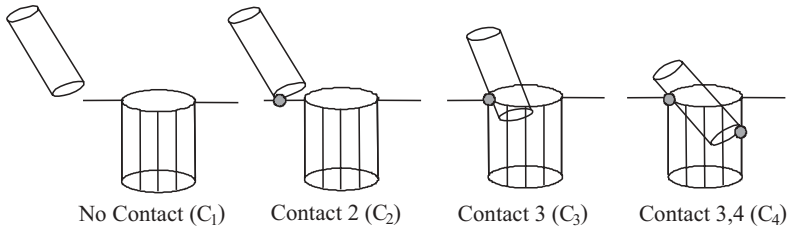

Fig. 1. Possible contact states during peg-in-hole insertion.

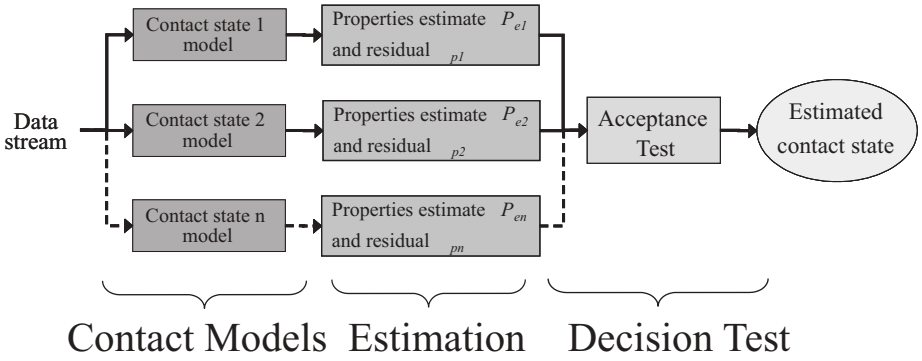

Fig. 2. Multiple model estimation.

Delson and West (1996) used human demonstration to program robots and in the process had to segment the data into subtasks that facilitated the generation of a robot program.

A variety of modeling approaches have been employed for contact state estimation. The most common methods include HMMs (Hannaford and Lee 1991; McCarragher and Hovland 1998; Rosen et al. 2002), generalized likelihood ratio tests (Eberman 1997), qualitative reasoning with thresholding (McCarragher 1994a), neural networks for offline segmentation (Fiorini et al. 1992) and Petri nets (McCarragher 1994b).

Methods based on Boolean combination of thresholds signals (Schulteis, Dupont, and Howe 1997) or techniques using probabilistic thresholds (De Schutter et al. 1999; Lefebvre, Bruyninckx, and De Schutter 2003) also appear. In the latter, for example, contacts between a manipulated object and the environment are modeled as virtual mechanisms whose geometry and nominal parameter values are known. For each contact state, an extended Kalman filter is designed. These filters are used to estimate geometric uncertainties using motion data, force data or both. The filters' innovations are then used with Bayesian rules to monitor contact state during manipulation tasks.

Threshold-based techniques usually do not employ a network of contact states and so assume that all contacts are equally likely. An exception is the work of Eberman (1997), in which a maximum likelihood estimator, combined with a task network, is used to distinguish between multiple models representing the number and the direction of constraints for a point moving in a maze.

The major contribution of this paper is a contact state estimation method combining multiple model estimation with a HMM representation of the robot's task (providing a network of allowable contact states and probabilities of transitions). This approach contrasts with prior applications of HMMs to contact state estimation in which the HMM was trained on raw sensor data (Hannaford and Lee 1991). Instead, the HMM developed here employs estimation residuals corresponding to contact point error distance for each contact state.

There are several advantages to this approach. First, only position data are needed. A second advantage is that, by eliminating time-varying unknowns from the estimation equations, nominal values need not be assumed for the estimated properties describing the objects in contact. Thus, the contact state of a cylinder sliding on a plane can be distinguished regardless of the cylinder's radius and an estimate of the unknown radius is also obtained.

3. Contact State Modeling

In this paper, contact states are modeled using their kinematic constraints. The associated properties are the dimensions and locations of objects, all of which can be estimated based on point contact locations. To model a contact state between a ma-

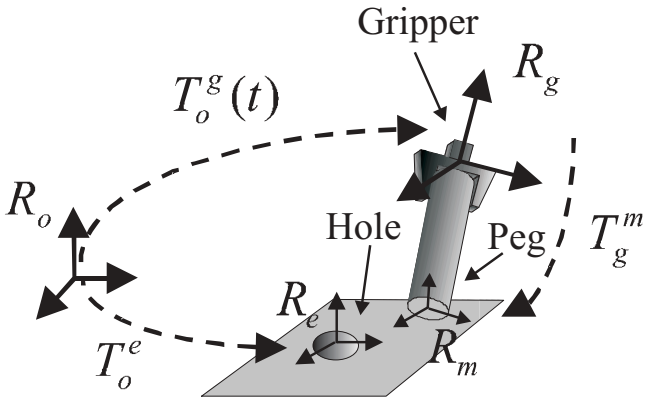

Fig. 3. Kinematic closure equation.

nipulated object and an environment object, constraint equations are first developed which describe the coincident contact points on each. The coordinates of these points are then related through a kinematic closure equation. To avoid the need for nominal values of geometric parameters, path constraints are imposed in a sufficient number to allow elimination of time-dependent contact coordinates. To measure a contact state's goodness of fit for a segment of sensor data, the error vector in the closure equation is projected on a line or plane orthogonal to the remaining time-varying contact coordinates. These steps are detailed in the subsections below.

3.1. Contact Point Constraints

Each contact state can be expressed using sets of parametrized constraint equations that describe the position of the contact state in different frames. For example, point contact between a manipulated object and an environment object can be expressed as

$$\begin{aligned} f_j\left(X_c^m(t), Y_c^m(t), Z_c^m(t)\right) &= 0 \\ g_j\left(X_c^e(t), Y_c^e(t), Z_c^e(t)\right) &= 0. \end{aligned} \tag{1}$$

Here, f_j and g_j are vector-valued functions of the contact's coordinates written with respect to body frames of the manipulated object, $\{X_c^m, Y_c^m, Z_c^m\}$, and the environment object, $\{X_c^e, Y_c^e, Z_c^e\}$. Examples are provided in Section 6.2.

3.2. Mapping Contact Point Coordinates to Sensor Measurements

In this paper, the only information assumed known regarding the manipulating robot is the gripper position and orientation. Therefore, a mapping between the sensor measurements and the contact point is defined to describe the motion of the contact point in the sensor frame, as shown in Figure 3.

R_o, R_g, R_m, and R_e represent coordinate frames for the manipulating robot, the gripper, the manipulated object and the environment, respectively. T_i^j represents a mapping from frame j to frame i.

In order to express the contact point in the sensor frame, the functions described in eq. (1) are related through the kinematic closure equation:

$$T_o^g(t)T_g^m(t)\begin{bmatrix} X_c(t) \\ Y_c(t) \\ Z_c(t) \\ 1 \end{bmatrix}^m = T_o^e(t)\begin{bmatrix} X_c(t) \\ Y_c(t) \\ Z_c(t) \\ 1 \end{bmatrix}^e. \tag{2}$$

Here, $T_o^g(t)$ is a homogeneous transform matrix which relates the gripper frame to the world frame based on the geometry of the remote manipulator. Similarly, $T_g^m(t)$ relates the manipulated object to the gripper frame, and $T_o^e(t)$ relates the environment object body frame to the sensor frame.

Substituting the contact constraints (1) into the kinematic closure equation (2), we can express the geometric constraints characterizing the contact states in terms of the (joint angle) sensor measurements and the properties associated with the objects in contact.

3.3. Contact Topology

In the kinematic closure equation (2), the left- and right-hand sides represent the fixed-frame coordinates of the contact point on the manipulated object and on the environment object, respectively. This equation can be modified to obtain a residual vector, $\vec{\varepsilon}^o$, corresponding to the error between the two contact points:

$$\begin{bmatrix} \varepsilon_x(t) \\ \varepsilon_y(t) \\ \varepsilon_z(t) \\ 0 \end{bmatrix}^o = T_o^g(t)T_g^m(t)\begin{bmatrix} X_c(t) \\ Y_c(t) \\ Z_c(t) \\ 1 \end{bmatrix}^m - T_o^e(t)\begin{bmatrix} X_c(t) \\ Y_c(t) \\ Z_c(t) \\ 1 \end{bmatrix}^e. \tag{3}$$

When a contact state is active, the residual vector computed from eq. (3) should be small when evaluated at the state's constraints of eq. (1).

Equations (1) and (3) are a set of nonlinear algebraic equations, which are parametrized by time-independent geometric parameters and by six time-dependent contact coordinates. In order to avoid the need for nominal values of the time-independent geometric parameters, the time-varying contact coordinates are to be eliminated.

By projection, up to two time-varying contact coordinates can be eliminated from eq. (3) and so four additional constraints must be provided by the contact state constraints of eq. (1). Polyhedral contacts provide the four needed constraints based on their geometry alone. For example, in a vertex-face contact state, constraining the contact point to a vertex eliminates three time-varying coordinates. Constraining the second contact point to lie on the face of the other object imposes a

fourth constraint. As a second example, in an edge–edge contact state, each edge imposes two constraints on the contact coordinates, again giving a total of four constraints.

3.3.1. Anticipated Path Constraints

A contact state composed of two curved surfaces is subject only to the two constraints limiting the contact points to the surfaces. To eliminate the time-varying coordinates in eq. (3), two more constraints are needed. Consider the example of contact 2 in Figure 1 in which the rim of the cylinder slides on a planar surface. Forcing the cylinder's contact point to lie on the rim imposes two constraints while forcing the second contact point to lie on the plane imposes a third constraint. In cases such as this, additional constraints can only be obtained by imposing limitations on the estimable paths within the contact state. When such constraints are imposed, the system can only detect those portions of the contact state corresponding to these special motions.

For the example of a cylinder sliding on a plane, there are two possible path constraints. One is to fix the contact point on the rim of the peg so that it slides without rolling on the plane. The other possibility is to force the peg to follow a particular curve on the surface of the plane. Of these two possible path constraints, the first is much more likely to occur as the peg is manipulated during the contact state. To maximize the opportunity to detect a contact state, the most likely path constraints are selected to define the constraint equations used for estimation. These constraints are referred to as "anticipated path constraints", i.e., the paths that the manipulator is likely to produce at some point during the contact state. Since the manipulator is not forced to follow an anticipated path constraint, successful estimation of the contact state depends on the manipulator following the path constraints for at least some short time interval during the associated contact state.

Note that it is also possible to use anticipated path constraints to impose more than four constraints on the contact coordinates. This can be done to further simplify the residual equations and so improve conditioning during estimation by reducing the number of time-independent unknowns to be estimated. Anticipated path constraints for each of the contacts appearing in Figure 1 are provided in Section 6.2.

In summary, the class of objects that can be included in the framework presented here is as follows.

- Polyhedral objects, where the possible contacts include contact between vertices, edges, and faces. All can be modeled as a combination of one-point contacts (e.g., a line contact can be modeled as two one-point contacts).
- Objects with curved surfaces, for which four constraints on the two contact points are imposed either by the geometry or by restrictions on the estimable paths within the contact state.

3.4. Formulation of Time-Independent Residual

For each contact state of a task, the residual vector defined by eq. (3) is reduced to a scalar which depends only on time-varying inputs (which are known) and time-independent unknown parameters. This process is formalized in the following five steps.

1. Body coordinate frames are defined for the manipulated and environment object involved in the contact.
2. For each object, geometric constraints on the contact point are defined with respect to the selected body frame in the form of eq. (1). These equations describe the sets of possible contact points on the two objects that correspond to the particular contact state.
3. Anticipated path constraints are defined with respect to the body frames so that the total number of constraints on the time-dependent contact coordinates due to both geometry and path is at least four.
4. The constraints from steps 2 and 3 are substituted into the residual vector defined in eq. (3).
5. The residual vector is made independent of the remaining time-varying contact coordinates by taking its projection along the line or plane orthogonal to the time-varying coordinates. The projected magnitude is used as the scalar residual in contact state estimation.

The projection procedure is comprised of two steps. First, a change of coordinates is used to isolate the time-varying contact components among $\{X_c^m, Y_c^m, Z_c^m; X_c^e, Y_c^e, Z_c^e\}$. Often, the two time-varying components are associated with one object and the desired coordinate change corresponds to the body frame of that object. For example, if the time-varying components are X_c^m and Z_c^m then eq. (3) can be transformed to the body frame of the manipulated object:

$$\begin{bmatrix} \varepsilon_x(t) \\ \varepsilon_y(t) \\ \varepsilon_z(t) \\ 0 \end{bmatrix}^m = \begin{bmatrix} X_c(t) \\ Y_c \\ Z_c(t) \\ 1 \end{bmatrix}^m - \left(T_o^g(t) T_g^m(t)\right)^{-1} T_o^e(t) \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix}^e . \tag{4}$$

While not shown here, when the two time-varying components are split between the two bodies (e.g., edge–edge contact), it is still possible to find a change of coordinates such that each time-varying component appears in only one component of the residual vector.

The final step of the projection procedure is to identify a projection vector orthogonal to the time-varying coordinates. In eq. (4), neglecting the fourth component, the body-frame projection vector is seen by inspection to be $\vec{n}^m = [0, 1, 0]^T$ which yields the projected magnitude

$$\varepsilon_p = \vec{n}^m \cdot \vec{\varepsilon}^m . \tag{5}$$

This scalar depends only on time-varying known inputs and time-independent unknown parameters.

The projection vector $\vec{n}^m$, which is orthogonal to the time-varying coordinates in the body frame, is recognized as the contact normal. In addition, the projected magnitude of the residual vector ε_p along this normal corresponds to the interpenetration distance, with its sign indicating either the interpenetration of the two objects or the distance between them.

A special case occurs when an extra path constraint is imposed which reduces the number of time-varying coordinates to be eliminated by projection to one. In this case, the residual vector (4) can be projected on the plane orthogonal to the time-varying coordinate or on any vector in that plane, including the normal vector. If planar projection is used, the scalar residual can be taken as the magnitude of the projected vector in frame i,

$$\varepsilon_p = \left\| \vec{\varepsilon}^i - \left(\vec{t}^i \cdot \vec{\varepsilon}^i\right) \vec{t}^i \right\|_2, \tag{6}$$

in which $\vec{t}^i$ is a unit vector in the direction of the time-varying coordinate. In this paper, both normal and planar projections are used.

4. Property Estimation

Using the approach described above, each contact state is expressed as a scalar residual equation, ε_p, which is parametrized by the properties of the two objects in contact and by the available sensing information:

$$\varepsilon_p = F(q_i, P_j), i = 1, \ldots, 12; j = 1, \ldots, \eta. \tag{7}$$

Here, q_i are the time-dependent elements of the matrix $T_o^g(t)$ corresponding to the kinematic data from the manipulating robot and P_j is the vector of properties associated with the objects in contact. Note that this equation might not be in its minimal form, resulting in non-identifiable parameters. Parameters are considered identifiable when they are multiplied by independent input signals. If multiple parameters are associated with the same input signal, then they are grouped together to avoid linear dependency of the Jacobian matrix (Hollerbach and Wampler, 1996). This concept is illustrated by the following equation:

$$\varepsilon_p = q_1(P_1 + P_2) + q_2 \cos P_3 + P_4. \tag{8}$$

By inspection, it is clear that P_1 and P_2 are not individually identifiable since they both multiply the same input. Their sum, however, forms a single identifiable parameter.

4.1. Estimation Algorithm: Nonlinear Least-Squares

Based on eq. (7), a nonlinear least-squares algorithm is used to estimate the properties P_j that minimize the residual ε_p given the kinematic data q_i of the manipulating robot. The Levenberg–Marquardt algorithm is used since it is robust to the choice of initial parameter values (Moré 1977). A wide range of initial conditions was successfully used experimentally. In the experiment presented in this paper, initial parameters were arbitrarily set to 1% of the parameters' actual values.

In the multiple model estimation shown in Figure 2, the parameters of all contact states are estimated simultaneously in a moving data window of fixed length ξ. The selection of the moving data window length ξ is determined by two considerations. Its lower bound is provided by η, the number of properties to be estimated in eq. (7). This is the minimum number of time steps needed to solve the overdetermined set of equations representing the contact states. Its upper bound corresponds to the minimum time interval a contact state is expected to be active. For a minimum time interval T_m and sampling frequency F_s, the window length is bounded by

$$\eta < \xi < F_s \cdot T_m. \tag{9}$$

In this paper, parameter estimates are first computed to perform contact state estimation and subsequently refined to obtain final values. In the first case, estimates are computed using the moving data window described above. Final values are computed by estimating over the entire data set for which contact states have been estimated as active.

4.2. Equation Conditioning and Multi-Pass Estimation

If a contact state is active and its associated contact equation is well conditioned (i.e., linear independency of the Jacobian matrix relating the parameters to the equation), then the constraint equations can be expected to fit the sensor data and to provide, through nonlinear least-squares, good estimates of the parameters and consequently small residuals.

Input trajectories are considered exciting when the condition number computed during the estimation step is below a user-defined threshold. The condition number provides an upper bound on the error magnitude of the estimates (Lawson and Hanson 1974), and it has been shown that a condition number of 100 or less is well suited for estimation (Schröer et al. 1992). As a consequence, estimation data windows with condition numbers exceeding this threshold are discarded.

The condition number can also be decreased by eliminating parameters multiplying low-excitation inputs. Such elimination can be carried out online by substituting parameter estimates from previously identified contact states into the estimation residuals to be used to estimate subsequent contact states. The assumption inherent in this approach is that the contact states occurring early in a task experience high excitation and consequently exhibit low condition numbers. When this assumption is not met, a robust alternative is multi-pass estimation.

In this approach, the system first estimates those contact states for which high input excitation is most likely. After these states are identified during task execution, their parameter

estimates are used to estimate contact states occurring both prior to and subsequent to these states. If the well-conditioned states occur early in the task then most of the estimation can be performed online. If these states occur near the end of the task, then multi-pass estimation corresponds to post-processing in its estimation of the early contact states.

In this paper, multi-pass estimation is implemented offline to allow for arbitrary ordering of the contact states during task execution.

5. Acceptance Test by Hidden Markov Model

As depicted in Figure 2, the residuals of well-conditioned data windows become the inputs to an acceptance test that provides an estimate of the contact state. The approach employed here uses a HMM to combine the residuals, representing how well the constraint equations fit the sensor data, with the probability of specific contact state transitions, embodied in a task contact state network.

A HMM can be described as a probabilistic observer by which a stochastic hidden process can be observed using the probabilistic structure of the task state network and a probabilistic relationship between the states and one or several observable stochastic signals. The contact state network of the HMM is described by n, the number of states, ρ, the n-vector of initial state probabilities, and A, the $n \times n$ state transition probability matrix (Rabiner 1989) as shown in eq. (10):

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \quad \rho = \begin{bmatrix} \rho_1 & \rho_2 & \cdots & \rho_n \end{bmatrix}^T \text{ with } \sum_{i=1}^{n} \rho_i = 1 \quad \text{with } \sum_{j=1}^{n} a_{ij} = 1 \quad . \tag{10}$$

The probabilistic relationship between the observable signal and the different states that comprise the task network is given by the observation sequence, O, of length T. For a continuous signal this relationship can be described using a probability density function, $B(O)$. Gaussians are used here for practicality:

$$B_i(O_t) = \frac{1}{(2p)^{1/k}|U_i|^{1/2}} \exp\left\{\frac{1}{2}(O_t - m_i)^T U_i^{-1}(O_t - m_i)\right\} \quad \text{with } 1 \le i \le n;\ 1 \le t \le T. \tag{11}$$

Here, k represents the maximum number of components of the signal O. Also, m_i and U_i are a $k \times 1$ vector of means and a $k \times k$ covariance matrix, respectively.

In summary, the inputs and output of the HMM are as follows.

Inputs. Number of states, n; initial state probability, ρ; state transition probability, A; observable signal, O; and probability density function, $B(O)$, defined by m_i and U_i.

Output. Estimated sequence of states associated with the observation sequence, O.

Computing the estimated sequence of states, $Q = q_1 q_2 q_3 \ldots q_T$, associated with observation signal, O, consists of finding the Q that minimizes $P(O, Q/\lambda)$, where $\lambda = (n, \rho, A, B)$. The conditional probability $P(O, Q/\lambda)$ can be expressed as in Rabiner (1989):

$$P(O, Q/\lambda) = \rho_{q_1} B_{q_1}(O_1) a_{q_1 q_2} B_{q_2}(O_2) a_{q_2 q_3} \cdots a_{q_{T-1} q_T} B_{q_T}(O_T). \tag{12}$$

To solve the minimization problem efficiently, a dynamic programming technique known as the Viterbi algorithm is employed (Viterbi 1967). As a result, each data point is classified by contact state.

6. Experiment: Three-Dimensional Peg-In-Hole Insertion

In this section, the implementations of the proposed contact and property estimation techniques are described for a tabletop manipulator. As an example of a common assembly task, peg-in-hole insertion is considered. The goal of the experiment is to estimate the sequence of contact states composing the task and to extract associated task properties. These properties, summarized in Table 1, include the radius and length of the peg as well as the location and orientation of the hole.

6.1. System Configuration

A PHANToM® haptic device, as shown in Figure 4, is used as the manipulating robot. The positions of the system's six joints are measured using high-resolution optical encoders. The kinematics of the robot is known and a closed-loop calibration technique (Hollerbach and Wampler 1996) is used to improve the absolute accuracy of the system.

A cylindrical peg is attached directly to the tip of the manipulator robot. Holes of different clearances are drilled perpendicularly to the surface of a rectangular aluminum block that is mounted on a three-degrees-of-freedom vise (roll–tilt–pan). The insertion is performed manually, using the manipulating robot as a way of recording the kinematic data.

Table 1. Estimated Parameters

R_p	Peg radius
L_p	Peg length
R_h	Hole radius
H_x, H_y, H_z	Hole's center coordinates
β_1, β_2	Orientation (pitch and yaw) of the plane
α_i	Angular coordinate locating the contact point on the rim of the peg for contact i
δ_i	Angular coordinate locating the contact point on the rim of the hole for contact i

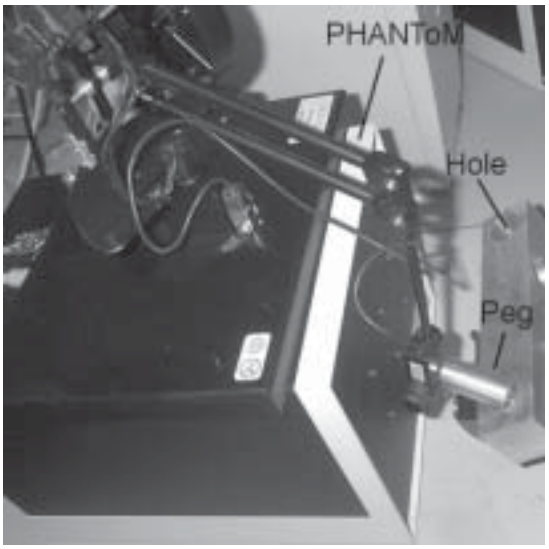

Fig. 4. Experimental apparatus: PHANToM 1.5, cylindrical peg and orientable hole.

6.2. Contact Constraints

In the spatial peg-in-hole insertion described in Figure 1, four primary contacts can be defined as depicted in Figure 5. In contact 2, for example, the contact point belongs to the bottom rim of the peg and to a plane in the environment frame.

Additional surface, line, and combined one-point contacts can occur; however, parametrizations of the primary contacts can be selected to include all of the desired geometric properties. Formulation of constraint equations for the primary contacts proceeds using the five steps developed in Section 3.4.

Step 1. Frames are assigned to the peg and hole respectively, such that their location and orientation correspond with the center and axis orientation of both objects.

Step 2. Using eq. (1), we can describe the position based constraint equations associated with each contact state as follows.

Contact 2

$$\begin{aligned} X_c^m(t) &= R_p \cos\alpha_2(t) \\ Y_c^m(t) &= R_p \sin\alpha_2(t) \\ Z_c^m(t) &= 0 \\ X_c^e(t) &= X_c^e(t) \\ Y_c^e(t) &= Y_c^e(t) \\ Z_c^e(t) &= 0 \end{aligned} \tag{13}$$

Contact 3

$$\begin{aligned} X_c^m(t) &= R_p \cos\alpha_3(t) \\ Y_c^m(t) &= R_p \sin\alpha_3(t) \\ Z_c^m(t) &= Z_c^m(t) \\ X_c^e(t) &= R_h \cos\delta_3(t) \\ Y_c^e(t) &= R_h \sin\delta_3(t) \\ Z_c^e(t) &= 0 \end{aligned} \tag{14}$$

Contact 4

$$\begin{aligned} X_c^m(t) &= R_p \cos\alpha_4(t) \\ Y_c^m(t) &= R_p \sin\alpha_4(t) \\ Z_c^m(t) &= 0 \\ X_c^e(t) &= R_h \cos\delta_4(t) \\ Y_c^e(t) &= R_h \sin\delta_4(t) \\ Z_c^e(t) &= Z_c^e(t). \end{aligned} \tag{15}$$

As summarized in Table 1, R_p and R_h represent the peg and hole radius, and α_i and δ_i are the time-dependent angular coordinates locating the contact point on the rim of the peg and on the rim of the hole, respectively.

Step 3. Only three geometric constraints were provided for each contact state in step 2. Thus, at least one anticipated path constraint is needed for each contact state. These constraints are based on paths the operator is likely to employ during the assembly task. While not unique, those described below have been found to be well suited to the task. Note that the minimum number of anticipated path constraint is added for contact 2, while two anticipated path constraints are added for contacts 3 and 4.

Contact 2. The peg slides on the plane without rolling. Thus, the contact point on the peg is fixed. As a consequence, $\alpha_2(t)$ becomes time-independent in eq. (13).

Contact 3. The peg slides, without rotating about its axis, across a fixed point on the rim of the hole. Thus, the contact point is fixed on the hole's rim, while it describes a line on the side of the peg, introducing two more constraints. As a consequence, $\alpha_3(t)$ and $\delta_3(t)$ become time-independent in eq. (14).

Contact 4. The peg slides without rolling along the cylindrical interior of the hole, parallel to the hole's axis. The contact point is fixed on the peg and describes a line on the interior surface of the hole, introducing two constraint equations. As a consequence, $\alpha_4(t)$ and $\delta_4(t)$ become time-independent in eq. (15).

Step 4. To define the residual vector (3), the following assumptions are made.

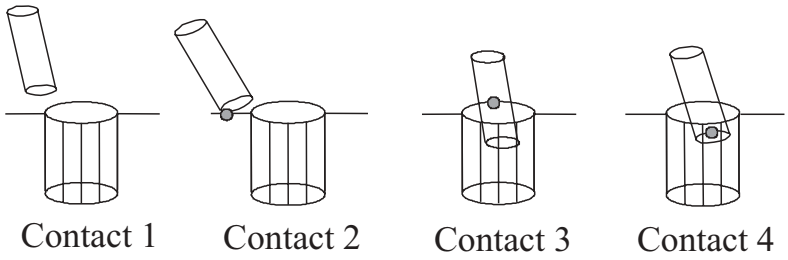

Fig. 5. Primary one-point contacts for peg-in-hole insertion.

1. The position of the gripper in the base frame is known ($T_o^g(t)$ is known).
2. The peg does not slip in the gripper (T_g^m is time-independent).
3. The shape of the manipulated object is known (cylindrical peg).
4. The grasping configuration of the peg is known.
5. Objects are assumed to be rigid.
6. The hole stays fixed (T_o^e is time-independent).
7. The orientation of the hole's axis, while unknown, is orthogonal to the surface in which it is drilled.

Under these assumptions, eq. (3) can be reduced to the following set of equations:

$$\begin{bmatrix} \varepsilon_x(t) \\ \varepsilon_y(t) \\ \varepsilon_z(t) \\ 0 \end{bmatrix}^o = \begin{bmatrix} q_{11} & q_{12} & q_{13} & q_{14} \\ q_{21} & q_{22} & q_{23} & q_{24} \\ q_{31} & q_{32} & q_{33} & q_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & L_p \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} X_c(t) \\ Y_c(t) \\ Z_c(t) \\ 1 \end{bmatrix}^m - \begin{bmatrix} \cos\beta_2 & 0 & \sin\beta_2 & H_x \\ \sin\beta_1 \sin\beta_2 & \cos\beta_1 & -\sin\beta_1 \cos\beta_2 & H_y \\ -\cos\beta_1 \sin\beta_2 & \sin\beta_1 & \cos\beta_1 \cos\beta_2 & H_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} X_c(t) \\ Y_c(t) \\ Z_c(t) \\ 1 \end{bmatrix}^e . \tag{16}$$

As shown in Table 1, β_1 and β_2 describe the orientation of the plane, L_p is the length of the peg, and H_x, H_y, H_z describe the hole's center coordinates. q_{ij} are known functions of the robot's kinematic parameters and joint angles. The vectors $[X_c^m(t), Y_c^m(t), Z_c^m(t)]^T$ and $[X_c^e(t), Y_c^e(t), Z_c^e(t)]^T$ represent the substituted constraints from steps 2 and 3.

Step 5. Equation (16) is made time-independent of the remaining time-varying coordinates using a two-step process. First, a change of coordinates is used to isolate the time-varying unknowns. Then, the components of the resulting vectors are projected onto a vector orthogonal to these time-varying coordinates, yielding the projected magnitude ε_p.

Contact 2. $X_c^e(t)$ and $Y_c^e(t)$ are the only time-dependent unknowns remaining in eq. (16). As a consequence, eq. (16) is transformed to the body frame of the environment object. Then, the resulting vector is projected along the contact normal, which is parallel to the z-axis of the frame associated with the hole, yielding the scalar ε_{p2}.

$$\begin{cases} \varepsilon_{p2} = \varepsilon_{ez} \\ \varepsilon_{ez} = \boldsymbol{K_2} + q_{14}\sin\boldsymbol{\beta_2} + q_{24}\cos\boldsymbol{\beta_2}\sin\boldsymbol{\beta_1} - q_{34}\cos\boldsymbol{\beta_1}\cos\boldsymbol{\beta_2} \\ \quad + \boldsymbol{L_p}[-q_{33}\cos\boldsymbol{\beta_1}\cos\boldsymbol{\beta_2} + q_{23}\sin\boldsymbol{\beta_1}\cos\boldsymbol{\beta_2} + q_{13}\sin\boldsymbol{\beta_2}] \\ \quad + \boldsymbol{R_p}\cos\boldsymbol{\alpha_2}[-q_{31}\cos\boldsymbol{\beta_1}\cos\boldsymbol{\beta_2} + q_{21}\sin\boldsymbol{\beta_1}\cos\boldsymbol{\beta_2} \\ \quad - q_{11}\sin\boldsymbol{\beta_2}] \\ \quad + \boldsymbol{R_p}\sin\boldsymbol{\alpha_2}[-q_{32}\cos\boldsymbol{\beta_1}\cos\boldsymbol{\beta_2} + q_{22}\sin\boldsymbol{\beta_1}\cos\boldsymbol{\beta_2} \\ \quad - q_{12}\sin\boldsymbol{\beta_2}] \\ \text{where } K_2 = H_z\cos\beta_1\cos\beta_2 - H_x\sin\beta_2 \\ \qquad - H_y\cos\beta_2\sin\beta_1. \end{cases} \tag{17}$$

The bolded variables in this equation, and those that follow, are the identifiable unknown parameters. Note that the terms forming K_2 are not individually identifiable since they do not multiply any input. In this case, they depend on the hole's center coordinates, which clearly cannot be identified in contact 2.

Contact 3. $Z_c^m(t)$ is the only time-dependent unknown remaining in eq. (16). As a consequence, eq. (16) is transformed to the body frame of the manipulated object. Then, the resulting vector is projected on the plane orthogonal to $Z_c^m(t)$, yielding the projected magnitude ε_{p3}.

$$\begin{cases} \varepsilon_{p3} = \sqrt{\varepsilon_{mx}^2 + \varepsilon_{my}^2} \\ \varepsilon_{mx} = q_{11}\boldsymbol{K_{3a}} + q_{21}\boldsymbol{K_{3b}} + q_{31}\boldsymbol{K_{3c}} - \boldsymbol{R_p}\cos\boldsymbol{\alpha_3} \\ \quad -q_{11}q_{14} - q_{21}q_{24} - q_{31}q_{34} \\ \varepsilon_{my} = q_{12}\boldsymbol{K_{3a}} + q_{22}\boldsymbol{K_{3b}} + q_{32}\boldsymbol{K_{3c}} - \boldsymbol{R_p}\sin\boldsymbol{\alpha_3} \\ \quad -q_{12}q_{14} - q_{22}q_{24} - q_{32}q_{34} \\ \text{where} \begin{cases} K_{3a} = H_x + R_h\cos\delta_3\cos\beta_2 \\ K_{3b} = H_y + R_h\cos\delta_3\sin\beta_1\sin\beta_2 \\ +R_h\sin\delta_3\cos\beta_1 \\ K_{3c} = H_z - R_h\cos\delta_3\cos\beta_1\sin\beta_2 \\ +R_h\sin\delta_3\sin\beta_1. \end{cases} \end{cases} \tag{18}$$

Contact 4. $Z_c^e(t)$ is the only time-dependent unknown remaining in eq. (16). As a consequence, eq. (16) is transformed to the body frame of the environment object. The resulting vector is projected on the plane orthogonal to $Z_c^e(t)$, yielding the projected magnitude ε_{p4}.

$$\begin{cases} \varepsilon_{p4} = \sqrt{\varepsilon_{ex}^2 + \varepsilon_{ey}^2} \\ \varepsilon_{ex} = \boldsymbol{K_{4a}} + q_{14}\cos\boldsymbol{\beta_2} + q_{24}\sin\boldsymbol{\beta_1}\sin\boldsymbol{\beta_2} - q_{34}\cos\boldsymbol{\beta_1}\sin\boldsymbol{\beta_2} \\ +\boldsymbol{L_p}[q_{13}\cos\boldsymbol{\beta_2} + q_{23}\sin\boldsymbol{\beta_1}\sin\boldsymbol{\beta_2} - q_{33}\cos\boldsymbol{\beta_1}\sin\boldsymbol{\beta_2}] \\ +\boldsymbol{R_p}\cos\boldsymbol{\alpha_4}[q_{11}\cos\boldsymbol{\beta_2} + q_{21}\sin\boldsymbol{\beta_1}\sin\boldsymbol{\beta_2} - q_{31}\cos\boldsymbol{\beta_1}\sin\boldsymbol{\beta_2}] \\ +\boldsymbol{R_p}\sin\boldsymbol{\alpha_4}[q_{12}\cos\boldsymbol{\beta_2} + q_{22}\sin\boldsymbol{\beta_1}\sin\boldsymbol{\beta_2} - q_{32}\cos\boldsymbol{\beta_1}\sin\boldsymbol{\beta_2}] \\ \varepsilon_{ey} = \boldsymbol{K_{4b}} + q_{24}\cos\boldsymbol{\beta_1} + q_{34}\sin\boldsymbol{\beta_1} + \boldsymbol{L_p}[q_{23}\cos\boldsymbol{\beta_1} \\ +q_{33}\sin\boldsymbol{\beta_1}] \\ +\boldsymbol{R_p}\cos\boldsymbol{\alpha_4}[q_{21}\cos\boldsymbol{\beta_1} + q_{31}\sin\boldsymbol{\beta_1}] + \boldsymbol{R_p}\sin\boldsymbol{\alpha_4}[q_{22}\cos\boldsymbol{\beta_1} \\ +q_{32}\sin\boldsymbol{\beta_1}] \\ \text{where} \begin{cases} K_{4a} = H_z\cos\beta_1\sin\beta_2 - H_x\cos\beta_2 \\ \quad -H_y\sin\beta_1\sin\beta_2 - R_h\cos\delta_4 \\ K_{4b} = -H_y\cos\beta_1 - H_z\sin\beta_1 - R_h\sin\delta_4. \end{cases} \end{cases} \tag{19}$$

Equations (17)–(19) allow the estimation of β_1, β_2, R_p, L_p and K_i. The remaining parameters of Table 1 can be estimated indirectly by solving the algebraic equations provided by the identified K_i.

6.3. State Network Modeling

The primary contacts of Figure 5 can be combined and ordered according to the expected sequence during the insertion task. The resulting sequence of three contact states is shown in Fig. 1. First, the peg is slid towards the hole on the planar surface (contact 2). As the peg enters the hole, it pivots on the rim of the hole (contact 3) and typically maintains this contact until the other side of the hole is reached. It then remains in double contact with the rim and the inside of the hole (contacts 3 and 4) until the peg is inserted far enough that the task can be easily completed. While not unique, this sequence is sufficient to estimate the properties listed in Table 1.

As shown in Figure 6(a), a state network can be used to model this sequence of contact states $C_1 - C_4$. Connections between states are labeled with their state transition probabilities a_{ij} as defined in eq. (10). Note that C_1, labeled "no contact", represents not only no contact, but also all other possible contacts besides $C_2 - C_4$. As a second example, an alternative two-state network appears in Figure 6(b). It is comprised of contact state C_2 and contact state C_0, representing all other possible contact states.

6.4. Residual Conditioning

To investigate input excitation and residual conditioning, the residuals for experimental data corresponding to the expected state sequence $C_1 - C_4$ are plotted in Figure 7(a). While not depicted, the magnitude of the condition numbers corresponds with that of the residual values. C_2 experiences a small residual when active and possesses the lowest condition number. In contrast, the residual for C_3 is large and poorly conditioned regardless of whether or not the contact state is active. The reason for this is that C_2 experiences greater input excitation than C_3 since its pitch and yaw angles are not constrained by the hole.

This is a case when the propagation of parameter estimates can improve the conditioning and estimation of a contact state. Figure 7(b) depicts the case when the parameter estimates from the easily estimated C_2 (i.e., β_1, β_2, R_p and L_p) are substituted into the residual equations of C_3 and C_4. While the residual of C_4 is largely unaffected, the residual of C_3 decreases substantially allowing contact state 3 to be easily estimated. If C_2 always occurred before C_3 then the contact states and their parameters could be estimated online in a single pass.

To allow for arbitrary ordering of states, however, a multi-pass approach, as described in Section 4.2, is employed. In the first pass, the state network of Figure 6(b) is used to estimate contact state C_2. Then, in a second pass, the four- state network of Figure 6(a) is employed and parameter estimates from C_2 are used to re-estimate the residuals for contact states occurring both before and after C_2. A description of the two HMMs corresponding to the two state networks follows.

6.5. Contact State Estimation by HMM

The multiple model estimation approach illustrated in Figure 2 is used to estimate the contact state sequence. Orientation and position of the robot's tip are recorded at a rate of 25 Hz, and contact state residuals (17)–(19) are computed at each time step using a 20-point moving data window. The residuals constitute the observation signal used as inputs to the HMMs. The two-state fully connected model of Figure 6(b) is first utilized to estimate contact state 2 and its parameters. Then, the four-state, fully connected model of Figure 6(a) is used, together with the parameter estimates from C_2, to estimate the remaining states and parameters. The design details of the four-state HMM are described below.

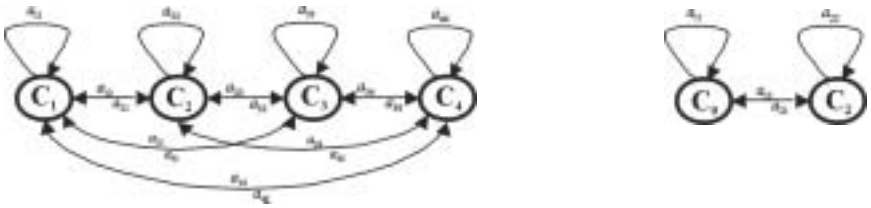

Fig. 6. Contact state networks: (a) four-state network based on the contact states of Figure 1; (b) two-state network for distinguishing contact state from all other possible contact states.

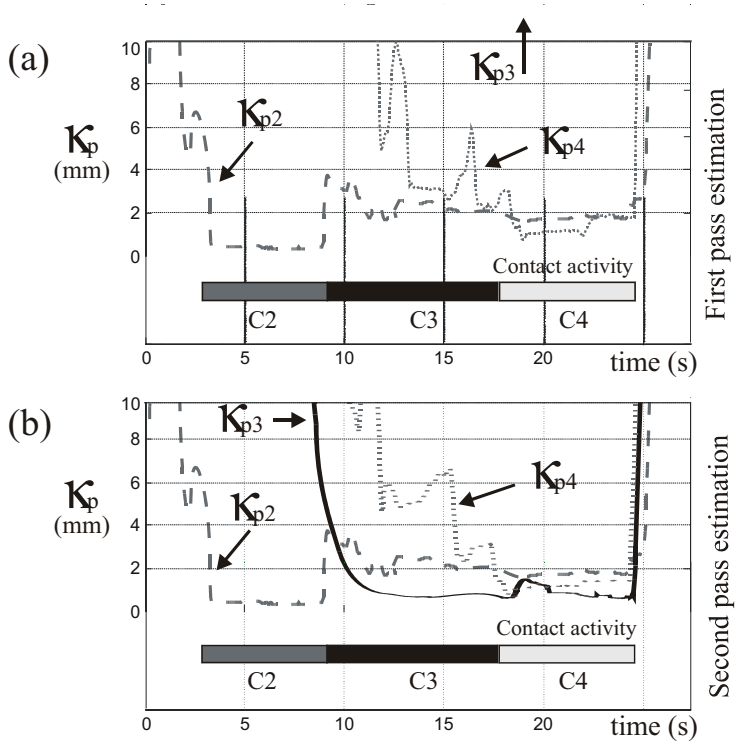

Fig. 7. Multi-pass estimation: (a) initial estimation residuals; (b) estimation residuals computed using parameter estimates propagated from C_2.

To obtain values for the initial probability vector p, the probability transition matrix A, and the probability density function B, training techniques such as the Baum–Welch algorithm can be used (Baum and Petrie 1966). Such techniques are well suited for applications with large numbers of states, e.g., speech recognition. Here, the task is comprised of a maximum of only four states and so the HMM parameters can be assigned manually.

The Gaussian probability density function B describes the relationship between the observation signal and the contact states. For each state, this relationship is defined by a mean vector m and a covariance matrix U. The observation signals are the residuals associated with the contact states. When a contact state is active, the associated residual is expected to be the smallest, given that the contact equations are well conditioned. This pattern, illustrated in Figure 7(b), is encoded in m as shown in eq. (20). The rows represent the mean of the residuals $\varepsilon_{p2} - \varepsilon_{p4}$, while the columns correspond to the states $C_1 - C_4$. For example, $m_{13} = 2$ implies that the expected mean of ε_{p2} is 2 when contact C_3 is active. The values shown were determined empirically using a two-step process. First, several contact sequences were manually segmented, and the m and U values corresponding to each state of the HMM were extracted. Then, final values were obtained by averaging these

experimental values. A similar technique was applied to determine the values of the covariance matrix U. Note that, since C_4 is composed of contacts 3 and 4, it is expected that both ε_{p3} and ε_{p4} are low when the state is active:

$$m = \begin{bmatrix} 10 & 0.5 & 2 & 2 \\ 90 & 40 & 1 & 1 \\ 80 & 90 & 5 & 1 \end{bmatrix}. \tag{20}$$

To define A, the probability transition matrix, and ρ, the initial probability vector, the following task knowledge is used.

- State C_1 (no contact) occurs first: $\rho_1 > \rho_j$, $j = 2, 3, 4$.
- State transitions are short: $a_{ii} > a_{ij}$, $1 \leq i, j \leq 4$.
- Transitions between C_2 and C_4 are impossible: $a_{24} = a_{42} = 0$

For the two-state model of Figure 6(b), the selection of the probability transition matrix is straightforward. The values were chosen such that the probability of remaining in the current state is much higher than the probability of leaving the state:

$$A = \begin{bmatrix} 0.99 & 0.01 \\ 0.01 & 0.99 \end{bmatrix}. \tag{21}$$

Because there are only two states, this model does not encode any information about the likely sequence of contact states. In contrast, the four-state model of Figure 6(a) does permit the inclusion of such information, and the selection of its A impacts what contact state sequences can be successfully estimated. This topic is explored in the following subsection.

6.6. Sensitivity of Contact State Estimation to Robot Path

While the state transition probability matrix encodes the probability of each contact state transition, the particular robot path employed in task execution may or may not correspond to the most likely sequence of transitions described by A. To be robust to variations in robot path, the matrix A must accommodate variations in contact state sequence. Such a matrix, A_f (f = flexible transition matrix), which was obtained empirically, is compared here with one which permits only the most likely state transitions, A_r (r = rigid transition matrix).

$$A_f = \begin{bmatrix} .7 & .1 & .1 & .1 \\ .15 & .7 & .15 & 0 \\ .1 & .1 & .7 & .1 \\ .15 & 0 & .15 & .7 \end{bmatrix} \tag{22}$$

$$A_r = \begin{bmatrix} .99 & .01 & 0 & 0 \\ 0 & .99 & .01 & 0 \\ 0 & 0 & .99 & .01 \\ .01 & 0 & 0 & .99 \end{bmatrix}. \tag{23}$$

These transition matrices are used to define two models, HMM_{2f} and HMM_{2r}, respectively. Employing two-pass estimation, a two-state model, HMM_1 first estimates contact state 2 and provides estimates of its parameters. In a second pass, HMM_{2f} and HMM_{2r} estimate their four contact states using residuals computed with the parameters estimated during the first pass. For comparison, manual segmentation was performed by the operator who pressed a switch at each perceived state transition.

Figure 8 depicts the results for the most likely sequence of contact states in peg insertion, $\{C_1, C_2, C_3, C_4, C_1\}$. In contrast, Figure 9 shows the results for a state sequence including some unexpected transitions, $\{C_1, C_3, C_2, C_3, C_4, C_1\}$. In both figures, the two-state HMM_1 demonstrates agreement with manual segmentation. For the most likely sequence of states in Figure 8, the additional transition flexibility of HMM_{2f} produces two short time segments in which the state is falsely identified. None the less, both models successfully match manual segmentation for all contact states.

For the unexpected sequence of contact states in Figure 9, the rigid state transition matrix of HMM_{2r} introduces many false transitions to satisfy the state transition matrix. For example, the first actual state change is from C_1 to C_3. The rigid model must pass through C_2 in order to make this transition. The next actual state change is from C_3 to C_2. The rigid model can only make this transition by the three state changes, $C_3 \rightarrow C_4 \rightarrow C_1 \rightarrow C_2$. The flexible state transition matrix of HMM_{2f} avoids all of these false transitions and successfully identifies large portions of the active states. It was found that this model was successful for a variety of state sequences. In addition, state estimation was robust to variations in the flexible state transition matrix, eq. (22). These results show that a flexible state transition matrix can successfully accommodate a broad range of robot paths during task execution.

6.7. Parameter Estimation

While initial parameter estimates are obtained using moving data windows during state estimation, final parameter estimates are calculated by time averaging the estimates associated with every detected time segment of contact states C_2, C_3 and C_4. Falsely identified time segments can be identified using statistical tests and discarded prior to averaging. As shown in Table 2, parameter estimates for a typical trial were found to be within 5% of the measured properties. Note that a relatively large ratio of peg-to-hole diameters (0.98) was employed to facilitate manual segmentation during algorithm development. Automatic segmentation of smaller ratios has also been performed successfully. For these cases, a triple contact state, illustrated in Figure 10, was also estimated.

7. Conclusions

In this paper, a perceptual system based on contact state estimation was presented. The approach employs multiple model

Table 2. Comparison of Estimated and Directly Measured Parameter Values

Properties	Direct Measurement	Estimated Value
R_p (peg radius)	12.5 ± 0.5 mm	11.8 ± 1 mm
L_p (peg length)	62.7 ± 0.5 mm	62.1 ± 1 mm
H_z (hole center, z)	–91.0 ± 0.5 mm	–90.8 ± 1 mm
H_x (hole center, x)	21.0 ± 0.5 mm	20.6 ± 1 mm
H_y (hole center, y)	–55.0 ± 0.5 mm	–54.8 ± 1 mm
β_1 (hole pitch)	18.0 ± 0.5 deg	18.5 ± 1 deg
β_2 (hole yaw)	20.0 ± 0.5 deg	19.2 ± 1 deg

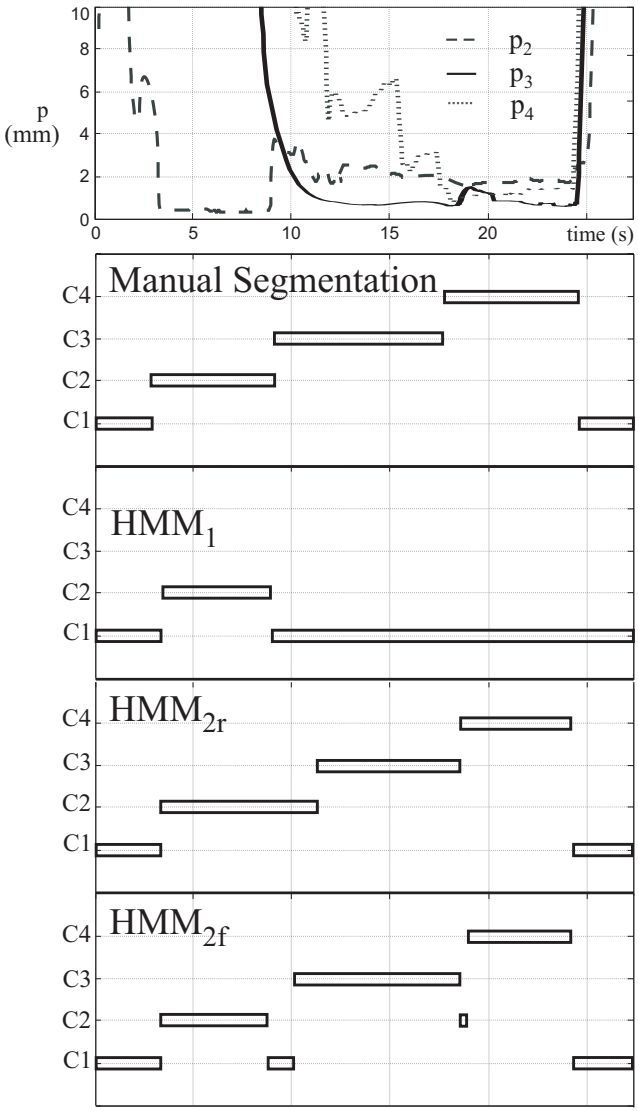

Fig. 8. State estimation of the most likely insertion state sequence.

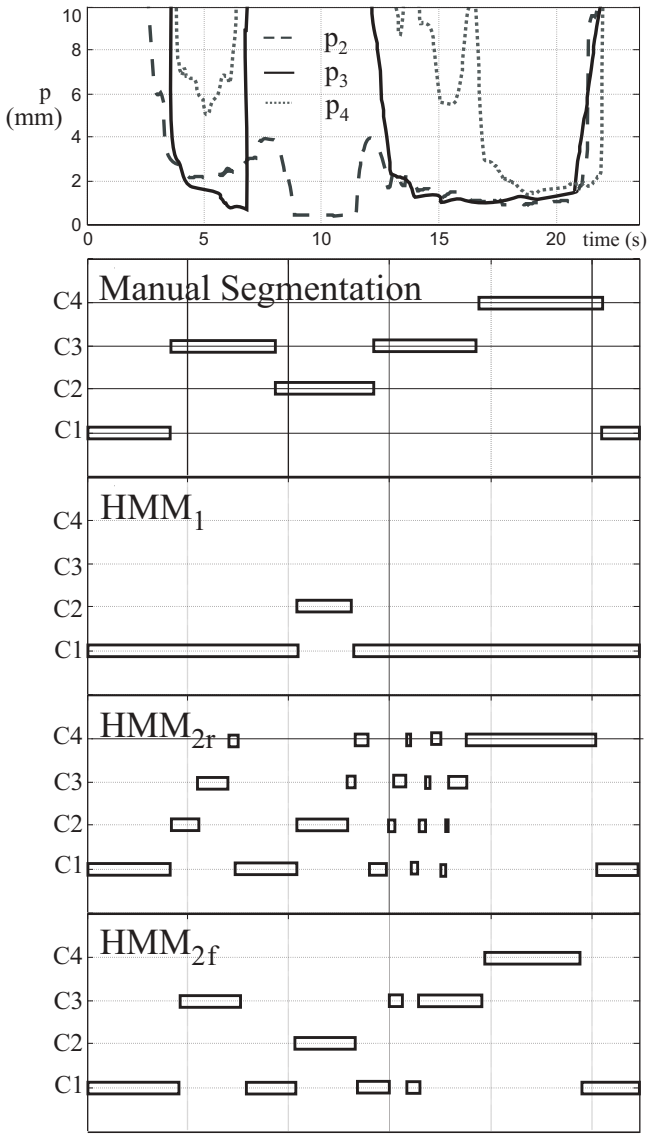

Fig. 9. Segmentation of an atypical state sequence.

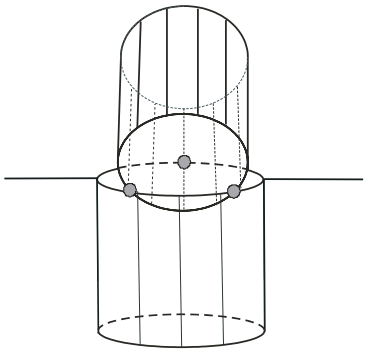

Fig. 10. Contact used for small clearance hole.

estimation and uses a HMM as a decision test. Since contact state estimation is based on property estimation, this approach provides a unified solution to the two estimation problems.

The experimental implementation demonstrates that the algorithm can successfully estimate arbitrary contact state sequences composing a task. Object properties are estimated to a level of accuracy far exceeding what could be achieved by an operator. The only inputs needed by the algorithm are the forward kinematics of the robot, constraint equation descriptions of the contact states, and a HMM description of the task to be performed. The flexibility of the proposed approach makes it straightforward to augment the perceptual capabilities of an existing system. Furthermore, the technique can be easily extended to consider additional sensors (e.g., force) and the estimation of other properties (e.g., inertia, friction).

For the position-based constraint equations employed here, the residuals were made independent of time-varying unknowns by the introduction of anticipated path constraints and by projection. The resulting estimation problem involves solving nonlinear algebraic equations over a moving data window and, owing to the elimination of time-varying unknowns, nominal parameter values are not needed. In the experimental example, this means that estimation success is independent of peg length and radius as well as hole location and diameter.

References

Baum, L. E., and Petrie, T. 1966. Statistical inference for probabilistic functions of finite state Markov chains. *Annals of Mathematical Statistics* 37:1554–1563.

Cao, T. H., and Sanderson, A. C. 1994. Task decomposition and analysis of robotic assembly task plans using Petri nets. *IEEE Transactions on Industrial Electronics* 41(6):567–583.

De Schutter, J., Bruyninckx, H., Dutre, S., De Geeter, J., Katupitiya, J., Demey, S., and Lefebvre, T. 1999. Estimating first-order geometric parameters and monitoring contact transitions during force controlled compliant motion. *International Journal of Robotics Research* 18(12):1161–1184.

Debus, T., Stoll, J., Howe, R., and Dupont, P. 2001. Combined human and machine perception in teleoperated assembly. *Experimental Robotics VII*, D. Rus and S. Singh, editors, Springer-Verlag, New York, pp. 51– 60.

Delson, N., and West, H. 1996. Segmentation of task into subtasks for robot programming by human demonstration. *ASME Japan/USA Symposium on Flexible Automation*, Boston, MA, pp. 41– 47.

Dupont, P., Schulteis, T., Millman, P., and Howe, R. 1999. Automatic identification of environment haptic properties. *PRESENCE: Teleoperators and Virtual Environments* 8(4):392– 409.

Eberman, B. 1997. A model-based approach to Cartesian manipulation contact sensing. *International Journal of Robotics Research* 16(4):508–528.

Fiorini, P., Losito, S., Giancaspro, A., and Pasquariello, G. 1992. Neural networks for offline segmentation of teleoperation tasks. *Proceedings of the IEEE International Symposium on Intelligent Control*, Glasgow, UK, Vol. 1, pp. 17–22.

Griebenow, B. E. 1994. Buried-waste integrated demonstration retrieval projects. *Transactions of the American Nuclear Society* 70:402.

Hannaford, B., and Lee, P. 1991. Hidden Markov modal analysis of force/torque information in telemanipulation. *International Journal of Robotics Research* 10(5):528–539.

Hollerbach, M. J., and Wampler, C. M. 1996. The calibration index and taxonomy for robot kinematic calibration methods. *International Journal of Robotics Research* 15(6):573–591.

Lawson, C. L. and Hanson, R. J. 1974. *Solving Least-Squares Problems, Series in Automatic Computation.* Prentice-Hall, Englewood Cliffs, NJ.

Lefebvre, T., Bruyninckx, H., and De Schutter, J. 2003. Polyhedral contact formation modeling and identification for autonomous compliant motion. *IEEE Transactions on Robotics and Automation* 19(1):26– 41.

McCarragher, B. J. 1994a. Force sensing from human demonstration using a hybrid dynamical model and qualitative reasoning. *Proceedings of the International Conference on Robotics and Automation*, San Diego, CA, Vol. 1, pp. 557–563.

McCarragher, B. J. 1994b. Petri net modeling for robotic assembly and trajectory planning. *IEEE Transactions on Industrial Electronics* 41(6):631– 640.

McCarragher, B. J., and Asada, H. 1995. The discrete events modeling and trajectory planning of robotics assembly tasks. *ASME Journal of Dynamic Systems, Measurement, and Control* 117:394– 400.

McCarragher, B. J., and Hovland, G. E. 1998. Hidden Markov models as a process monitor in robotics assembly. *International Journal of Robotics Research* 17(2):153–158.

Moré, J. J. 1977. The Levenberg–Marquardt algorithm: implementation and theory. *Numerical Analysis, Lecture Notes in Mathematics*, G.A. Watson, editor, Springer- Verlag, Berlin, pp. 105–116.

Pook, P. K., and Ballard, D. H. 1993. Recognizing teleoperated manipulations. *Proceedings of the IEEE International Conference on Robotics and Automation*, Atlanta, GA, Vol. 2, pp. 578–585.

Rabiner, L. R. 1989. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE* 77(2):257–286.

Rosen, J., Brown, J., Chang, L., Barreca, M., Sinanan, M., and Hannaford, B. 2002. The BlueDRAGON—a system for measuring the kinematics and the dynamics of minimally invasive surgical tools in vivo. *Proceedings of the IEEE International Conference on Robotics and Automation*, Washington DC, pp. 11–15.

Schröer, K., Uhl, L., Albright, S., and Huttenhofer, M. 1992. Ensuring solvability and analyzing results of the nonlinear robot calibration problem. *Proceedings of the 2nd International Symposium on Measurement and Control in Robotics*, Tsukuba, Japan, pp. 851– 858.

Schulteis, T., Dupont, P., and Howe, R. 1997. Experimental identification of kinematic constraints. *Proceedings of the IEEE International Conference on Robotics and Automation*, Albuquerque, NM, pp. 2677–2682.

Viterbi, A. J. 1967. Errors bounds for convolutional codes and an asymptotically optimal decoding algorithm. *IEEE Transactions on Information Theory* 13:260–269.