Robots for Australian Industries Conference, Melbourne, July 5-7, 1995.

State Transition Recognition in Robotic Assembly Using Hidden Markov Models

Geir E. Hovland* and Brenan J. McCarragher†
Department of Engineering
Faculty of Engineering and Information Technology
The Australian National University
Canberra, Australia
Fax: int + 61-(0)6-249 0506
E-mail: {brenan,geir}@faceng.anu.edu.au
* Sponsored by the Research Council of Norway.
† This research has been partly funded by a grant from the Australian Research Council.

Abstract – *A process monitor for robotic assembly based on Hidden Markov Models (HMMs) is presented. The measurements are the force/torque signals arising from interaction between the workpiece and the environment for a planar assembly task. The HMMs represent a stochastic knowledge-based system where the models are trained off-line with the Baum-Welch re-estimation algorithm. After the HMMs have been trained, we use them on-line in a robotic system to recognise events as they occur. Process monitoring with an accuracy of 98% was accomplished in 0.5-0.6s.*

1 Introduction

Process plants must deal with changing states, multiple faults, unexpected situations and unreliable measurements. To handle these problems real-time process monitoring is essential. Process monitoring is widely used as a component in many industrial processes. In robotic assembly, however, there is an increasing need for efficient process monitoring methods to account for existing uncertainties of workpieces and the environment.

One example of process monitoring in robotic assembly is presented by Donald [2], where a theory of planning multi-step error detection and recovery strategies for compliant motion assemblies is described. The theory provides a technology for constructing plans that might work, but fail in a "reasonable" way when they cannot. However, the direct applicability in robotics is limited.

Another example is found in the paper by McCarragher and Asada, [7]. They present a model-based approach which incorporates the dynamic nature of the process to highlight the discrete changes of state. This method uses qualitative force/torque measurements and hence is well suited for fast real-time monitoring in robotic assembly. However, by using qualitative measurements, a significant amount of information is ignored. It is therefore expected that more accurate, reliable and robust process monitoring in robotic assembly can be achieved by a more comprehensive treatment of the force/torque measurements.

Hidden Markov Models (HMMs) have been successfully applied to different areas within robotics by other authors. Hannaford and Lee [4] use a HMM as a process monitor for a peg-in-the-hole assembly, where the HMM describes the task structure and each state in the HMM describes a sub-task. The different states in the HMM are characterised by a different mean and covariance of the force/torque measurements. A peg-in-the-hole task with four sub-tasks is studied, ie. moving, tapping, inserting and extracting. The moving sub-task has a force/torque mean of zero, both the tapping and the inserting sub-tasks have a positive (but different) mean value and the extracting sub-task a negative mean value, and these values distinguish the different states (sub-tasks).

Zhu [11] describes how an HMM is used to describe and predict the motions of obstacles in a dynamic environment. Initially an image of the scene is acquired by a visual sensing device. This image is processed and any potential obstacles in the environment are detected. The HMM is then used to make predictions of the obstacle motions and a collision-free path is searched for and finally maneuver commands are generated. As the robot proceeds, this cycle repeats.

In the paper by Yang et.al. [10] the HMM approach is applied to skill learning in telerobotics. They use a discrete 5-state left-to-right model to learn different trajectories in both Cartesian and joint space for a telerobotic system. The HMM then represents the skill/command that generated the trajectories.

In this paper we present a method for recognition of discrete state transitions in robotic assembly by using force and torque measurements. Each transition in the assembly is described by a HMM. HMMs are suited to transition recognition in robotic assembly for two reasons. First, the state space is naturally discretised which is a basic requirement for any Markov modeling. Second, the sensory signals, particularly force sensing, have stochastic noise which is a requirement for HMMs.

The amount of force information available when there is a sudden change in contact state is large. This is the main reason for letting the HMMs represent the contact state transitions instead of the contact states. We allow for dynamic motions of the workpiece, friction and sensor noise. Also, we do not require exact knowledge of the positions of the workpiece and the environment. Moreover, since our method is both model-based and uses empirical data, we ensure the validity of the models. The method presented here is a significant improvement in the area of process monitoring of robotic assembly.

2 Assembly Process

We consider a planar peg-in-the-hole assembly process where we have only one type of contact, the *edge-surface* contact. The edge or the surface involved can be a part of either the workpiece or the environment. Examples of this contact type is shown in Figure 1. We do not model *edge-edge* contacts, since this type of contact is physically difficult to achieve.

Definition 1 (Discrete State) *A discrete state is defined as a contact formation consisting of one or two edge-surface contacts between the workpiece and the environment. The set of discrete states is then the set of all possible contact formations.* □

Definition 2 (Event) *We now define an event in robotic assembly as the change of discrete state. The events will be discrete in time and describe the gain or a loss of a*

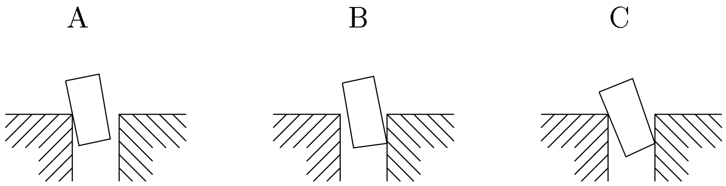

Figure 1: *Examples of different edge-surface contacts between the workpiece and the environment. Note that contact state C is a combination of the two edge-surface contacts A and B.*

edge-surface contact between the workpiece and the environment. □

We use the notation from Astuti and McCarragher [1] where γ_i is used for contact states, $M = \{\gamma_1, \gamma_2, \cdots, \gamma_m\}$ is a finite set of discrete states and $E \subset M \times M$ is a finite set of discrete events. Any discrete event e_k is then given by an ordered pair $e_k = (\gamma_i, \gamma_j) \in E$, describing a change from the discrete state γ_i to γ_j.

3 Hidden Markov Model

An HMM is defined as a doubly stochastic process with an underlying stochastic process that is *not* observable (it is hidden), but can only be observed through another set of stochastic processes that produce the sequence of observed symbols, [8]. This is illustrated in Figure 2. In our work the hidden models represent the discrete events defined in the

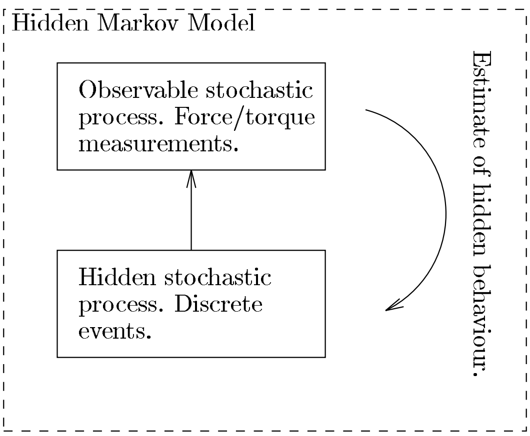

Figure 2: *Illustration of a HMM. The underlying stochastic process can only be observed through another stochastic process.*

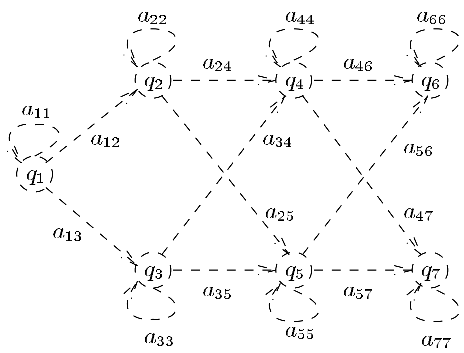

Figure 3: *HMM λ^k for each event. q_i are the states and a_{ij} are the hidden stochastic process state change probabilities. q_1 is the initial state while q_6 and q_7 are the absorbing states.*

previous section which are observed through the stochastic force/torque measurements.

The HMM is defined as a doubly stochastic process. This means that in addition to the force/torque measurements being stochastic, the underlying process must also be stochastic. In our case this underlying stochastic process corresponds to the internal behaviour of the events, since we are using one HMM for each event. This internal behaviour is influenced by several factors. In a peg-in-the-hole assembly the internal stochastic behaviour is for example the peg's position in the gripper, friction between the workpiece and the environment or tolerancing errors in the geometries of the workpiece or the environment. This is very important since the whole theory of Hidden Markov Models is based on the fact that both the underlying process and the observed process are stochastic. In the paper by Hannaford and Lee [4] the underlying process is not the internal behaviour of the events, but rather the sequence of discrete events. In that case the assumption has to be made that the sequence of discrete events is stochastic.

There are many possible HMM models for the events. One type of model which has been successfully applied to speech recognition and also works well for robotic assembly, is the absorbing parallel left-to-right model illustrated in Figure 3. For this model an event always starts in the state q_1 and ends in q_6 or q_7, but there are several possible paths through the states. The observed symbols proceed forward in time and this is the main reason for using a left-to-right model. The left part of the model is associated with the early measurements while the right part of the model is associated with measurements at some time later. Since these measurements might be very different, a left-to-right model is ideal for extracting this information. Left-to-right models are a good tool to describe how signals evolve in time. This is an important property of the HMMs. With this property we are able to describe dynamic motions with dynamic force/torque measurements of the workpiece, as opposed to quasi-static motions.

The HMM force-based recognition method is described as follows. Assume we have a set of E events as defined in the previous section to be recognised. For each event $e_k = (\gamma_i, \gamma_j) \in E$ we have a training set $\mathbf{L}^k = \{\mathbf{L}_1^k, \mathbf{L}_2^k, \cdots, \mathbf{L}_n^k\}^T$ where the individual $\mathbf{L}^k$'s are matrices containing the force/moment time series describing event e_k. In the planar case we have that $\mathbf{L}_l^k = \{\mathbf{l}_{Fxl}^k, \mathbf{l}_{Fyl}^k, \mathbf{l}_{Mzl}^k\}^T$ where $\mathbf{l}_{Fxl}^k, \mathbf{l}_{Fyl}^k$ and $\mathbf{l}_{Mzl}^k$ are vectors containing the force measurements in the x- and y-direction and the moment measurements in the z-direction, respectively, describing event e_k.

The event recognition then follows the following steps:

1-Training. First we build an HMM for each event in the event set. We use the observations from the training set $\mathbf{L}$ to estimate the optimum parameters for each event e_k, represented by λ^k, for the k^{th} event in the event set, $e_k \in E$. We use well known estimation techniques like the Baum-Welch re-estimation formulas.

2-Operation. From on-line measurements we have a single test sequence, $\mathbf{Y} = \{\mathbf{y}_{Fx}, \mathbf{y}_{Fy}, \mathbf{y}_{Mz}\}^T$ containing the 3-DOF force/torque measurement vectors in the x-, y- and z-direction, respectively. For each event model in the set $e^k \in E$, we calculate $P_k = Pr(\mathbf{Y}|\lambda^k)$, the probability of the measurements matching the event model λ_k.

3-Evaluation. We choose the event whose model probability is highest, ie.

$$k^* = \text{argmax}_{e_k \in E}[P_k]$$

By comparing the scores for the different events, we are also able to calculate a confidence level of the decision. One possibility is to define the confidence level of the decision as the probability of the chosen model divided by the sum of all model probabilities, ie.

$$C = \frac{P_{k^*}}{\sum_k P_k} \in (0, 1] \tag{1}$$

In section 3.1 we describe the observation symbols generated by the observable stochastic process. In section 3.2–3.4 we describe the training, operation and evaluation steps in detail.

3.1 Observation Symbols

One key element in successful process monitoring is the observation symbols. In this paper we use the frequency components of the force/torque signals as observation symbols. The training will then estimate the mean and the covariance of the energy in different frequency components of the force/torque measurements. Hannaford and Lee [4] use the mean and covariance of the measurements directly in a robotic system where an HMM is used as a process monitor. There are several reasons for using frequency components. First, when an event occurs in robotic assembly, as well as in any dynamic system where there is a sudden change in the constraint equations, the frequency band is broad and occurs within a short time scale of the event, see Eberman and Salisbury [3]. Therefore, a lot of information can be extracted by using frequency components as observation symbols. Second, by normalising the total energy in the measurements, we make the observation symbols independent of the signal magnitudes, ie. a gentle contact (event) will have similar observation symbols as a brutal contact. In this paper we use continuous observation symbols represented by the continuous vector $\mathbf{x}_t$. The observation probability $b_j(\mathbf{x}_t)$ is then given by

$$b_j(\mathbf{x}_t) = \mathcal{N}[\mathbf{x}_t, \mu_j, \mathbf{\Sigma}_j] \tag{2}$$

where $\mathcal{N}$ is the normal density, μ_j and $\mathbf{\Sigma}_j$ are the mean vector and covariance matrix associated with the internal state j in the HMM. It is possible to use discrete observation symbols, but better performance has been reported for continuous HMMs. This is mainly because of the vector quantisation error in discrete observation symbols.

We now define the transform

$$\mathcal{C}_s(\omega_i) = \left| \log \frac{\mathcal{F}(\omega_i)_s}{\mathcal{F}(0)_s} \right| \tag{3}$$

where $s \in \{F_x, F_y, M_z\}$ are the three degree of freedom force/torque measurements for a planar assembly task from time t to $t + W$ where W is a design parameter, representing the length of the observation window. $\mathcal{F}$ is the Fourier transform and the ω_i's are the observation frequencies. As mentioned above, we divide each frequency component by $\mathcal{F}(0)_s$ to make them independent of signal magnitudes. We use the logarithm function to make the elements of the covariance matrices $\mathbf{\Sigma}_j$ smaller. Small covariances will make the observation probabilities in equation 2 larger relative to the hidden process state probabilities. This is desirable in the training and the operation of the HMMs. Small observation probabilities will give us only limited amount of information about the hidden stochastic process.

With n observation frequencies, the continuous vector $\mathbf{x}_t$ is then given by

$$\mathbf{x}_t = \left[\mathcal{C}_{F_x}(\omega_1), \cdots, \mathcal{C}_{F_x}(\omega_n), \mathcal{C}_{F_y}(\omega_1), \cdots, \mathcal{C}_{F_y}(\omega_n), \mathcal{C}_{M_z}(\omega_1), \cdots, \mathcal{C}_{M_z}(\omega_n)\right]^T \tag{4}$$

3.2 Training

The first step of the HMM recognition method is to build an HMM for each event $e^k \in E$. The most difficult problem in HMMs is how to adjust the model parameters to maximise the probability of the observed data matching the model. There is no known way to solve this analytically for a maximum likelihood model [5], so using a numerical method is desirable. The method we use is a standard estimation method called the Baum-Welch re-estimation formula, see for example [8].

We use the following notation for an HMM based on continuous observations:

$$\begin{array}{rclrcl} T & = & \text{length of observation sequence} & \pi & = & \text{initial state probability} \\ N & = & \text{number of hidden process states} & \delta & = & \text{absorbing state probability} \\ Q & = & \{q_1, q_2, \cdots, q_N\} \text{ hidden process states} & \mathbf{x}_t & = & \text{observation vector at time } t \\ A & = & \{a_{ij}\} \text{ probability of transition } q_i \to q_j & B & = & \{b_j(\mathbf{x}_t)\} \text{ observation probability} \end{array}$$

We use the compact notation $\lambda = (A, B, \pi)$ to represent an HMM. For the model in Figure 3 we have

$$\begin{array}{rclrcl} N & = & 7 & \pi & = & [\,1\,0\,0\,0\,0\,0\,0\,]^T \\ Q & = & \{q_1, q_2, q_3, q_4, q_5, q_6, q_7\} & \delta & = & [\,0\,0\,0\,0\,0\ \ \delta_1\ 1-\delta_1]^T \end{array}$$

where $\delta_1 \in [0, 1]$ will be changed during the training. δ_1 is the probability of ending in hidden state q_6, while $1 - \delta_1$ is the probability of ending in hidden state q_7.

The Baum-Welch method starts with initial values of all the HMM parameters, and after applying the method the probability of the observed data matching the HMM is increased. Applying the Baum-Welch formula once is called an iteration. For the probability to reach a (local) maximum, we have to run several training iterations.

The Baum-Welch re-estimation formula is based heavily on two variables, the *forward* and *backward* variables. For a description of how to calculate these variables, see [8]. The variables are defined as follows:

Definition 3 (Forward variable α)

$$\alpha_t(j) = Pr(\mathbf{x}_0, \mathbf{x}_2, \cdots, \mathbf{x}_t, j = q_j \mid \lambda^k) \tag{5}$$

where $\alpha_t(j)$ is the probability of the partial observation sequence (until time t) and hidden process state q_j at time t, matching the model λ^k. □

Definition 4 (Backward variable β)

$$\beta_t(j) = Pr(\mathbf{x}_{t+1}, \mathbf{x}_{t+2}, \cdots, \mathbf{x}_T \mid j = q_j, \lambda^k) \tag{6}$$

where $\beta_t(j)$ is the probability of the partial observation sequence from $t + 1$ to the end, given hidden process state q_j at time t, matching the model λ^k. □

Definition 5 (HMM probability) *The HMM probability is defined as the probability of the entire observation sequence $\mathbf{x}_0, \cdots, \mathbf{x}_T$ matching the model λ^k. For example if the model λ^k represents a state k in the hidden stochastic process, the HMM probability is then the probability of the observed stochastic process being generated from this state in the hidden stochastic process. Finding this probability for a model λ^k is called* scoring *the model. This is done by using the forward variable α as follows.*

$$Pr(\mathbf{x}_0, \cdots, \mathbf{x}_T | \lambda^k) = \sum_{j \in N} \alpha_T(j) \tag{7}$$

□

In robotic assembly the HMM probability for λ^k is the probability of the observed force/torque measurements matching the model λ^k for event $e^k \in E$.

From the forward and backward variables we now define two new variables describing the HMM state probabilities. The difference between HMM state probabilities and hidden process state probabilities, is that the HMM state probabilities take into account both the hidden process and the observable process. These new variables will be used in the final form of the Baum-Welch re-estimation formula.

Definition 6 (HMM state probability $\sigma(j)$)

$$\sigma_t(j) = \frac{\alpha_t(j)\beta_t(j)}{Pr(\mathbf{x}_0, \cdots, \mathbf{x}_T \mid \lambda^k)} \tag{8}$$

where $\sigma_t(j)$ is the probability of being in the hidden process state q_j at time t, given the observation sequence $\mathbf{x}_0, \cdots, \mathbf{x}_T$ and the model λ^k. □

Definition 7 (HMM state transition probability $\sigma(i,j)$)

$$\sigma_t(i,j) = \frac{\alpha_t(i)a_{ij}b_j(\mathbf{x}_{t+1})\beta_{t+1}(j)}{Pr(\mathbf{x}_0, \cdots, \mathbf{x}_T \mid \lambda^k)} \tag{9}$$

where $\sigma_t(i,j)$ is the probability of a path being in state q_i at time t and making a transition to state q_j at time $t+1$, given the observation sequence and the model. □

We now use the definitions for $\sigma(j)$ and $\sigma(i,j)$ to express the Baum-Welch re-estimation formula. For each event $e_k = (\gamma_i, \gamma_j) \in E$ we have a training set $\mathbf{L}^k = \{\mathbf{L}_1^k, \mathbf{L}_2^k, \cdots, \mathbf{L}_n^k\}$ where the individual $\mathbf{L}^k$'s are matrices containing force/moment samples describing event e_k. In the planar case we have that $\mathbf{L}_l^k = \{\mathbf{l}_{Fxl}^k, \mathbf{l}_{Fyl}^k, \mathbf{l}_{Mzl}^k\}^T$ where $\mathbf{l}_{Fxl}^k, \mathbf{l}_{Fyl}^k$ and $\mathbf{l}_{Mzl}^k$ are vectors containing the force measurements in the x- and y-direction and the moment measurements in the z-direction, respectively, describing event e_k. These measurements are transformed into observation symbols $\mathbf{x}_t$ as described in the previous section. Given the model λ^k with initial parameters, the re-estimation formula for the internal state transition parameters a_{ij}^k is then given by

$$\bar{a}_{ij}^k = \frac{\sum_{\mathbf{L}_l^k \in \mathbf{L}} \sum_{t=1}^{T-1} \sigma_t^{\mathbf{L}_l^k}(i,j)}{\sum_{\mathbf{L}_l^k \in \mathbf{L}} \sum_{t=1}^{T-1} \sum_{j \in N} \sigma_t^{\mathbf{L}_l^k}(i,j)} \tag{10}$$

We see that the estimate $\bar{a}_{ij}^k$ is the sum of all transitions from the state q_i to the state q_j divided by the probability of being in state q_i for model λ^k. The new estimates $\bar{a}_{ij}^k$ are based on the HMM state transition probabilities $\sigma(i,j)$, so we see now how the hidden process is estimated through the observable process. Since the $\mathbf{x}_t = \mathbf{x}_t^{\mathbf{L}_i^k}$'s are different for each training sequence, we also have that the σ_t's are different for each training sequence $\mathbf{L}_i^k$ and hence are written $\sigma_t^{\mathbf{L}_i^k}$.

By using multivariate Gaussian distributions, the estimates for μ_j^k and $\mathbf{\Sigma}_j^k$ for the model λ^k in equation 2 are given by

$$\bar{\mu}_j^k = \frac{\sum_{\mathbf{L}_i^k \in \mathbf{L}} \sum_{t=1}^{T} \sigma_t^{\mathbf{L}_i^k}(j)\mathbf{x}_t}{\sum_{\mathbf{L}_i^k \in \mathbf{L}} \sum_{t=1}^{T} \sigma_t^{\mathbf{L}_i^k}(j)} \tag{11}$$

$$\bar{\mathbf{\Sigma}}_j^k = \frac{\sum_{\mathbf{L}_i^k \in \mathbf{L}} \sum_{t=1}^{T} \sigma_t^{\mathbf{L}_i^k}(j)(\mathbf{x}_t - \bar{\mu}_j^k)(\mathbf{x}_t - \bar{\mu}_j^k)^t}{\sum_{\mathbf{L}_i^k \in \mathbf{L}} \sum_{t=1}^{T} \sigma_t^{\mathbf{L}_i^k}(j)} \tag{12}$$

Theorem 1 (Maximum likelihood estimates) *The Baum-Welch re-estimation formula given in the equations 10 − 12 is an iterative solution for a (local) maximum likelihood HMM.*

Proof: See [5]. ∎

Define the scalar $S^k = \sum_{\mathbf{L}^k \in \mathbf{L}} Pr(\mathbf{x}_0^k, \cdots, \mathbf{x}_T^k \mid \lambda^k)/n_l$ to be the model score for λ^k based on the entire training set, where n_l is the size of the training sets $\mathbf{L}^k$. The training procedure is then described as follows.

1. Choose an initial model λ^k for each event $e^k \in E$.
2. Get the new model parameters from equation 10 to 12. These new model parameters are based on the entire training set $\mathbf{L}^k$ for that event. This is called *supervised* training since we know that the training set $\mathbf{L}^k$ belongs to the event e^k.
3. If the change in the model score S^k is below a chosen threshold value, the training is completed. Otherwise go to step 2.

3.3 Event Detection

As described earlier in this paper, the amount of information available when there is a change in the constraint equations is large. This is illustrated in Figure 4 which is a similar result to Figure 1 in [3]. This large amount of information available when an event occurs is the main reason for letting the HMMs represent the events rather than the contact states. However, to be able to use this information we need an event detector. A change in the constraint equations is characterised by a sudden change in at least one of the force measurements. The event detector used here is described by the following threshold test.

$$[F_x(t) - F_x(t - \Delta T)]^2 + [F_y(t) - F_y(t - \Delta T)]^2 \geq \Delta F \tag{13}$$

where ΔT and ΔF are design parameters. If the threshold test is able to detect gentle contacts, it will also detect brutal contacts. Hence, the design parameters are found by looking at the force measurements for gentle contacts.

3.4 Event Recognition

Once all the models λ^k, $e^k \in E$, are trained, the recognition is straightforward.

1. For each model in the event set, $e^k \in E$, score the model by finding the model probability, ie. find $Pr(\mathbf{x}_0, \cdots, \mathbf{x}_T \mid \lambda^k) = \sum_{j \in N} \alpha_T(j)$. We see that the backward variable is not used for event recognition, it is only used in the training procedure.

2. The recognised event is then the event whose model has the highest score, ie.

$$k^* = \mathrm{argmax}_{e^k \in E}[Pr(\mathbf{x}_0, \cdots, \mathbf{x}_T \mid \lambda^k)] \tag{14}$$

4 Experiments

The HMM transition recognition method was tested with a 5-degree of freedom Eshed Scorbot using a JR3 force/torque sensor, see Figure 5. The measurements were sampled

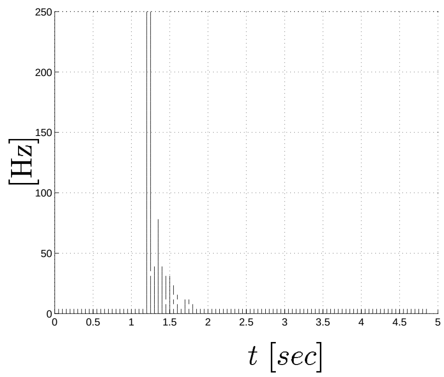

Figure 4: *Spectogram of one of the force measurements for a gain of contact. The event starts at t=1.2sec. The measured force was sampled at 500 Hz. The FFT was computed for every 128 data points. Notice that the frequency band is broad and occurs within a short time scale of the event.*

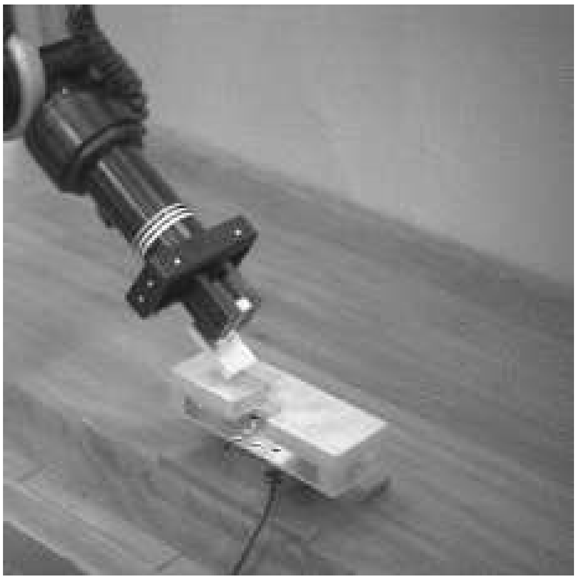

Figure 5: *Setup for a peg-in-the-hole assembly using a 5-degree-of-freedom Eshed Scorbot and a JR3 force/torque sensor.*

at 500Hz with a lowpass filter with cutoff at 31.25Hz for a planar peg-in-the-hole assembly with 12 possible discrete events, as shown in Figure 6. Ideally we would have used no filtering because the high frequency components are an important information channel. However, for the threshold test in equation 13 we need to use a lowpass filter. Otherwise, high-frequency noise will trigger the threshold test. ΔT was chosen 20 samples (40ms) and ΔF was chosen 1.0.

The 12 possible events are as follows:

$$\begin{aligned}
e_1 &= p_0 \rightarrow p_1 \quad e_2 = p_1 \rightarrow p_0 \quad e_3 = p_0 \rightarrow p_2 \quad e_4 = p_2 \rightarrow p_0 \\
e_5 &= p_0 \rightarrow p_4 \quad e_6 = p_4 \rightarrow p_0 \quad e_7 = p_1 \rightarrow p_2 \quad e_8 = p_2 \rightarrow p_1 \\
e_9 &= p_2 \rightarrow p_3 \quad e_{10} = p_3 \rightarrow p_2 \quad e_{11} = p_3 \rightarrow p_4 \quad e_{12} = p_4 \rightarrow p_3
\end{aligned}$$

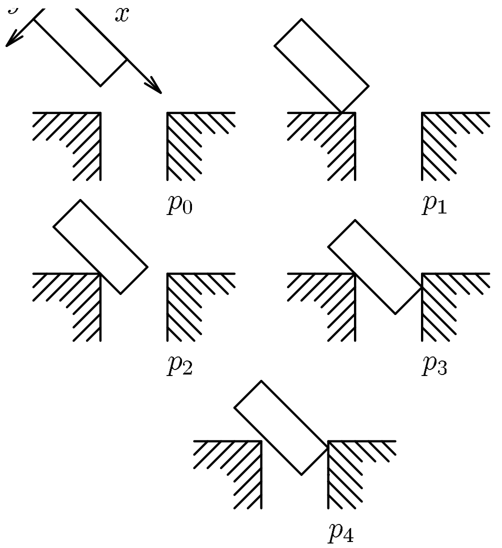

Figure 6: *Planar peg-in-the-hole assembly with 5 contact states and 12 possible discrete events.*

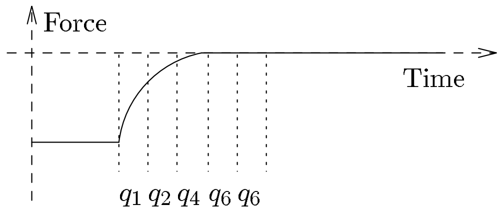

Figure 7: *The initial model parameters for the state sequence $q_1 \to q_2 \to q_4 \to q_6$ are calculated from the windows shown in the figure. The parameters for q_3, q_5 and q_7 are initialised in a similar manner but with a different position of the peg in the gripper.*

Good initial model parameters are essential for the maximum likelihood estimation algorithm. In the experiments presented here, we have used initial estimates as described in Figure 7 for the model in Figure 3. Only one of the force measurements for a typical transition is shown, to simplify the illustration. For each event model the training set contains samples with different positions of the peg in the gripper, to account for the stochastic behaviour of the discrete events. The measurements for two of these positions are used for calculating the initial parameters. For the first position of the peg we initialise $q_1 \to q_2 \to q_4 \to q_6$ and for the other one $q_1 \to q_3 \to q_5 \to q_7$. The parameters for q_1 were taken as means of the two cases for each model. The initial parameters for the a_{ij}'s were chosen as follows.

$$\mathbf{A} = \begin{bmatrix} 0.04 & 0.48 & 0.48 & 0 & 0 & 0 & 0 \\ 0 & 0.04 & 0 & 0.8 & 0.16 & 0 & 0 \\ 0 & 0 & 0.04 & 0.16 & 0.8 & 0 & 0 \\ 0 & 0 & 0 & 0.04 & 0 & 0.8 & 0.16 \\ 0 & 0 & 0 & 0 & 0.04 & 0.16 & 0.8 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

The state transition recogniser was tested with different sizes of the training set. We used 10 observation frequencies for the force measurements in the x- and y-direction, hence $dim(\mathbf{x}_t) = 20$. Diagonal covariance matrices Σ_j for each state and each model λ_e were used. The length of the observation window was $W = 256$ samples (0.51s) and this window was divided into four sub-windows of length 64 samples, so the observation symbol vectors were calculated every 0.128 seconds. The results were as follows.

Size of Training Set	Successful Recognition Rate
20	87.5 %
30	96.5 %
40	98.0 %

5 Evaluation

The HMM force/torque based event recogniser has several advantages over existing methods, but the method also has its limitations.

5.1 Advantages

Dynamic motions – We allow for dynamic motions of the workpiece relative to the environment. There are no restrictions on the robot accelerations as in quasi-static motions. By allowing dynamic motions, the workpiece is no longer restricted to a certain kind of trajectory. We are now able to design continuous controllers with no restrictions on the accelerations.

Model Based on Empirical Data – Our method is model based where the models are trained on real measurements from the process plant. Hence, no assumptions on the measured forces/torques are made. Training the models on empirical data ensures the validity of the models. After the training the model parameters can be used to extract meaningful information about the measured data.

Friction and Measurement Noise – Since our method is trained on empirical data, the effects from friction between the workpiece and the environment and measurement noise are accounted for. In fact, any linear and non-linear phenomena in the dynamic measurements are accounted for as long as the training data contains these phenomena.

Frequency Information – As mentioned earlier in this paper, the amount of information available when there is a change in contact formation is large. By using the frequency components of the measured data, we are able to extract this information. Frequency information also eliminates the difference between hard and gentle contacts.

5.2 Limitations

Training Data – One limitation of the HMM approach is the extensive amount of training data required. In [9] they found that training sets of size 500 to 1000 are adequate for many typical applications in speech processing. Once the HMMs are trained they are capable of high-performance transition recognition. However, we expect smaller training sets than in speech processing. This is mainly because we are using only one manipulator and only one force/torque sensor in all our experiments. In speech processing, the HMM recogniser has to recognise speech from several independent speakers.

Data Log Time – Another limitation of the HMM approach is the amount of on-line data log time. Once a transition is detected, characterised by a sudden change in at least one of the force measurements, the force measurements have to be logged for some time before the actual computation starts. In the experiments in the previous section, the data log time was 0.51 seconds.

Initial Parameters – Unless the initial model parameters are good, the estimated model parameters may be a local maximum. Although there exist other estimation algorithms, like the gradient projection method [6], they all end up in a local maximum if starting with poor initial parameters. In [9] several "toy" models were simulated where the estimated parameters were compared to the true values and the conclusion is: *It is absolutely mandatory to have a good initial guess of the means of the density functions to obtain good HMMs.* As described in the experiment section, we are able to find a good initial estimate for the models for use in robotic assembly.

6 Conclusion

We see that for the setup with 12 possible events, successful event recognition was accomplished with an accuracy of 98% with a relatively small training set. For a full assembly task it is expected that a larger training set is required. The method presented here allows for dynamic motions of the workpiece, it accounts for friction and measurement noise, it is model based where the models are trained on empirical data and it exploits the fact that there is a large amount of force information when there is a sudden change of contact state. All these properties make the method presented here a significant improvement to the process monitoring of robotic assembly.

References

[1] P. Astuti and B.J. McCarragher, *The Stability of a Class of Discrete Event Systems Using Markov Chains.* The International Journal of Control, to be published.

[2] B.R. Donald, *Planning Multi-Step Error Detection and Recovery Strategies.* The International Journal of Robotics Research, Feb. 1990, Vol. 9, No. 1, p.3-60.

[3] B. Eberman and J.K. Salisbury, *Application of Change Detection to Dynamic Contact Sensing.* The International Journal of Robotics Research, Oct. 1994, Vol. 13, No. 5, p.369-394.

[4] B. Hannaford and P. Lee, *Hidden Markov Model Analysis of Force/Torque Information in Telemanipulation.* The International Journal of Robotics Research, Oct. 1991, Vol. 10, No. 5, p.528-539.

[5] X.D. Huang, Y. Ariki, M.A. Jack, *Hidden Markov Models for Speech Recognition.* Edinburgh University Press, 1990.

[6] Q. Huo, C. Chan, *The Gradient Projection Method for the Training of Hidden Markov Models.* Speech Communication, 1993, Vol. 13, p.307-313.

[7] B.J. McCarragher and H. Asada, *Qualitative Template Matching Using Dynamic Process Models for State Transition Recognition of Robotic Assembly.* Journal of Dynamic Systems, Measurements and Control, June 1993, Vol. 115, no. 2A, p.261-269.

[8] L.R. Rabiner and B.H. Juang, *An Introduction to Hidden Markov Models.* IEEE ASSP Magazine, January 1986, p.4-16.

[9] L.R. Rabiner, B.H. Juang, S.E. Levinson, M.M. Sondhi, *Some Properties of Continuous Hidden Markov Model Representations.* AT&T Technical Journal, 1985, Vol. 64, No. 6, p.1251-1269.

[10] J. Yang, Y. Xu, C.S. Chen, *Hidden Markov Model Approach to Skill Learning and Its Application to Telerobotics.* IEEE Transactions on Robotics and Automation, Oct. 1994, Vol. 10, No. 5, p.621-631.

[11] Q. Zhu, *Hidden Markov Model for Dynamic Obstacle Avoidance of Mobile Robot Navigation.* IEEE Transactions on Robotics and Automation, June 1991, Vol. 7, No. 3, p.390-397.