SWITCHING RECONFIGURABLE CONTROL BASED ON HIDDEN MARKOV MODELS

J. M. Lemos(‡), L. M. Rato(†) and J. S. Marques(+)

(‡)INESC/IST, R. Alves Redol 9, 1000 Lisboa – Portugal, email: jlml@inesc.pt
(†)INESC, R. Alves Redol 9, 1000 Lisboa – Portugal, email: lmr@inesc.pt
(+)ISR/IST, Torre Norte, Av. Rovisco Pais 1, 1000 Lisboa – Portugal, email: jsm@isr.ist.utl.pt

Keywords: Reconfigurable Control, Fault Tolerant Control, Supervisor, Switching Control, Hidden Markov Models, Nonlinear Filtering.

Abstract A supervisor algorithm for application in switching reconfigurable control of time varying plants is presented. The possible outcomes of the parameters configuring plant dynamics are described by a hidden Markov model whose state, assumed unavailable for direct observation, is estimated from plant i/o data using a nonlinear filtering algorithm. The use of a shared state representation allows a natural fitting of this supervisor to Predictive Reconfigurable Control.

1 Introduction.

There are several instances where a controller ought to be reconfigured in response to changes in plant dynamics. This may happen under normal operating conditions (*e. g.* a robot arm or a crane picking up a load or a chemical reactor undergoing temperature changes) or for mitigating the effect of a faulty condition. Assuming that a fault causes a change in plant dynamics, a possible remedy consists in reconfiguring the controller so that it matches the new situation. The use of reconfiguration allows extending the region of stability, while preventing performance degradation, when the controller plant is nonlinear and is to be operated over wide ranges. A reconfigurable controller structure (fig. 1) should thus include a supervisor block taking care of the task of selecting which controller, among the admissible ones $C_1, \ldots, C_N$, best fits the dynamics yielded by the new operating conditions. It is remarked that, although related, this structure should not be confused with gain schedulling [1]. In gain schedulling, the choice of the parametrization of the controller depends on a schedulling variable. Instead, in supervisory based reconfigurable control, the possible set of plant operating conditions is divided in subregions to which a linear model, hereafter referred as a *local model* is associated. For each

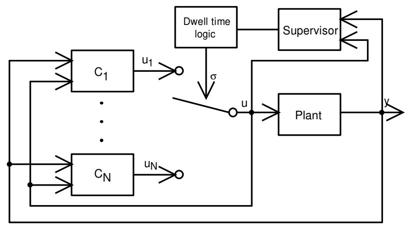

Figure 1: Multicontroller structure.

of the local models a linear controller (referred as *local controller*) is designed. The decision on which local controller output $u_1, \ldots, u_N$, is actually applied to the plant at a given moment is made by comparing the output of the local models with the plant output.

The supervisors considered so far in the literature [2, 3] assume that the plant parameters are unknown, but they don't explore any knowledge about the possible transitions between possible values of these parameters. There are however situations of interest in which changes in plant dynamics occur in sequence, with transitions to some points in plant parameter space being more probable than others. For instance, in a distributed collector solar field [4], when a cloud passes, the corresponding changes in radiation force the plant to browse a whole set of dynamic behaviours, which appear in sequence.

In this paper it is assumed that the set of parameters configuring plant dynamics is the outcome of a Markov chain with sparse transitions, *i.e.* operating in a time scale which is much slower than the fastest transient of the plant (fig. 2). The state of this Markov chain is not accessible for direct observation and can only be inferred from plant i/o data. This remits one to the framework of Hidden Markov Models (HMM) and to a problem which is intimatelly related to similar ones in Communication coding [5]. Problems related to the one at hand have been treated by authors dealing with detection and estimation for abruptly changing systems [6, 7, 8, 9, 10] but without recasting them in the framework of reconfigurable con-

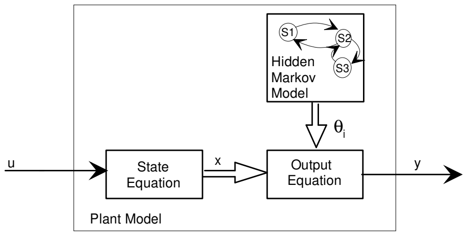

Figure 2: HMM based plant model.

trol. In the present paper this is done by exploring a shared state realization for the possible plant models [2]. According to this, the only equation in the plant model, affected by the change in dynamics is the output equation, the state equation remaining unchanged.

The structure of the paper is as follows: In section 2 the plant model employed is described and the problem is formulated. In section 3 an algorithm solving the problem is derived using standard nonlinear filtering (Bayes law) techniques. Section 4 includes simulation results and section 5 summarizes conclusions.

2 Problem Formulation.

Consider a plant described in discrete time t by the following ARMAX model

$$A(q)y(t) = B(q)u(t) + C(q)\eta(t) \tag{1}$$

where A, B, C are polynomials in the forward shift operator q, such that A and B and A and C have no unstable common factors, C is Hurwitz, $\partial A = \partial C = n > \partial B = m$ with ∂X denoting the degree of polynomial X. The signals u and y are, respectively, the plant input and output, and η is a zero mean, white noise sequence of identically distributed gaussian random variables. It is assumed that C is a fixed, known polynomial (interpreted as an observer polynomial) and A and B are selected from a set of possible plant parameters, $\mathcal{P} = \{A_i, B_i, i = 1, ..., N\}$

2.1 Shared state realization

Divide eq. (1) by $C(q)$ and add and subtract $y(t)$ to get

$$y(t) = (C(q)-A(q))\frac{1}{C(q)}y(t)+B(q)\frac{1}{C(q)}u(t)+\eta(t) \tag{2}$$

As shown in the appendix, (2) admits the state-space representation

$$x(t) = \Phi_o x(t-1) + \Gamma_{oy} y(t) + \Gamma_{ou} u(t) \tag{3}$$

the matrices Φ_o, Γ_{oy} and Γ_{ou} being independent of the coefficients of polynomials A and B. The output equation corresponding to the state realization associated to $x(t)$ is

$$y(t) = \theta' x(t) + \eta(t) \tag{4}$$

where $\theta' = [c_1 - a_1 ... c_n - a_n \quad b_0 ... b_m]$ with a_i, b_j, c_i, $i = 1, ..., n$ $j = 1, \ldots, m$ the coefficients of polynomials A, B and C and prime denotes transpose. The state dynamic equation depends only on the coefficients of the polynomial C which are assumed to be known, and thus do not change in time. Polynomial C is interpreted as an observer polynomial. Thus, the above representation is called a *shared-state representation* [2] since the state is common to all the possible models in $\mathcal{P}$, and only θ in the output equation (4) changes When the operating condition change. The plant is thus modelled as in fig. 2.

2.2 HMM of plant dynamics

At each time t the value of θ, representing plant dynamics, is issued from the state of a finite state Markov chain with states $S_i, i = 1, ..., N$. When the Markov chain is in state S_i, it is assumed that the vector θ determining plant dynamics has the value θ_i. Let $P(t) \triangleq [P_1(t)...P_N(t)]'$ be the vector of *a priori* probabilities of the states of the Markov chain at time t, *i.e.* $P_i(t)$ represents the probability that the Markov chain is in state S_i at time t. Let $A_M^{'} = [a_{ij}]'$ be the (N×N) state transition matrix *i.e.*, such that

$$P(t) = A_M^{'} P(t-1)$$

In this paper it is assumed that A_M is known. The methods described here can be complemented with an algorithm for the on-line estimation of A_M [5], but this offcourse increases the overall algorithm complexity. Furthermore, it is assumed that the transitions of the Markov chain occur in a time scale which is slower than the one of (1). Since the state of the Markov chain is not accessible for direct observation, but must be inferred from the observations, it will be referred hereafter as an hidden Markov Model (HMM).

The problem to be solved in this paper consists in deciding at each time t, on the basis of the available observations of plant input and output, what is the most probable of the models in the set $\mathcal{P}$. The result of this decision is used to select which controller is to be applied to the plant, according to the structure of fig. 1. According to the dwell time logic, the index σ of the controller C_σ applied to the plant is such that a controller remains connected to the plant at least a minimum interval of time ("dwell time"). In this way, unstability due to chattering among controllers is prevented [2, 3].

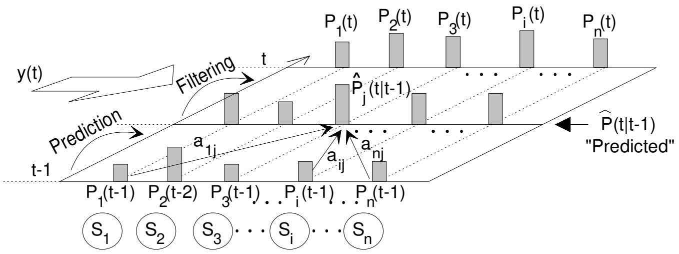

Figure 3: Interpretation of the hidden Markov chain state estimation algorithm.

3 Supervisor.

For the HMM based plant model a supervisor is now obtained by nonlinear filtering techniques. This closely resembles the Baum algorithm [5]. From this supervisor a suboptimal algorithm is then derived, which does not take into account the dynamics of the state transitions.

3.1 Nonlinear filtering supervisor

Let $P_i(t-1), i = 1, ..., N$ be the probability that the Markov chain is in state S_i at time $t-1$, given the observations up to time $t-1$, $\mathcal{O}_{t-1}$, and $P_j(t|t-1)$ the probability of state S_j at time t, given observations up to time $t-1$. Using Bayes law and standard nonlinear filtering techniques, these are recursively propagated in time by the operations of:

A)*Prediction*, defined by

$$P_j(t|t-1) = \sum_{i=1}^{N} a_{ij} P_i(t-1) \quad j = 1, ..., N \qquad (5)$$

B)*Filtering*, defined by

$$P_j(t) = k_t C_\eta \exp[-\frac{(y_t - x^{'}(t)\theta_j)^2}{\sigma_\eta^2}] P_j(t|t-1) \qquad (6)$$

Fig. 3 provides an interpretation of this algorithm. The predicted probability of S_j, for each $j = 1, ..., N$, receives contributions from all the states. The contribution of the generic state S_i is the transition probability from S_i to S_j (i.e., a_{ij}) weighted by the probability $P_i(t-1)$. This yields the convolution sum (5). The filtering operation corrects the predicted probabilities by using the last observation. It ammounts to a pointwise multiplication given by (6).

3.2 Suboptimal supervisor

Define

$$M(t) \triangleq k_t C_\eta diag(\exp[-\frac{(y(t) - x^{'}(t)\theta_i)^2}{\sigma_\eta^2}])$$

Eqs. (5) and (6) are written together in matricial form as

$$P(t) = MA^{'}P(t-1)$$

Since, in general, M and $A^{'}$ do not commute, it is not possible to group all the prediction operations and then all the filtering operations. There is, however, one interesting special case, for which $A^{'} = I$. This means that there can be no transitions among states, and each $\theta_i, i = 1, ..., N$ is characterized by an *a priori* probability P_{io}. The *a posteriori* probabilities for each state are then given by

$$P_i(t) = \Pi_{\tau=1}^{t}(k_\tau C_\eta \exp[-\frac{(y_(\tau) - x^{'}(t)\theta_i)^2}{\sigma_\eta^2}] P_{io}) \qquad (7)$$

Define the prediction errors of model i by

$$\varepsilon_i(t) \triangleq y(t) - x^{'}(t)\theta_i \qquad i = 1, ..., N$$

and take logarithm in (7) to get

$$\log P_i(t) = \sum_{\tau=1}^{t}[\log k_\tau + \log C_\eta - \frac{1}{\sigma_\eta^2}\varepsilon_i^2(\tau) + \log P_{io}]$$

Since $\log k_\tau$ and $\log C_\eta$ do not depend on i, choosing the model with the highest probability is equivalent to select the model corresponding to the smallest euclidean error norm, corrected by the *a priori* probabilities:

$$\overline{\pi}_i = ||\varepsilon_i(t)||^2 - \log P_{io}$$

This is the criterion used in [2], where the plant is assumed constant but unknown and all the possible models are equiprobable.

4 Simulation examples

In this section, simulation examples are presented in which the above HMM based supervisor is used in a reconfigurable controller for a time varying plant. In the following, the accumulated loss is defined as $J_{acc}(t) \triangleq \sum_{i=1}^{t} y^2(i)$

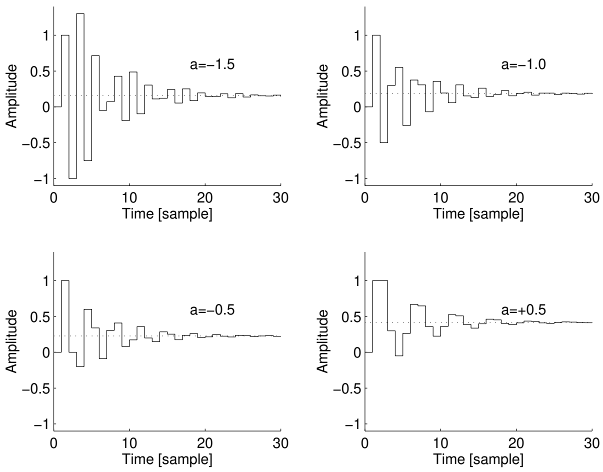

Figure 4: Example 1 - Step response of the local models.

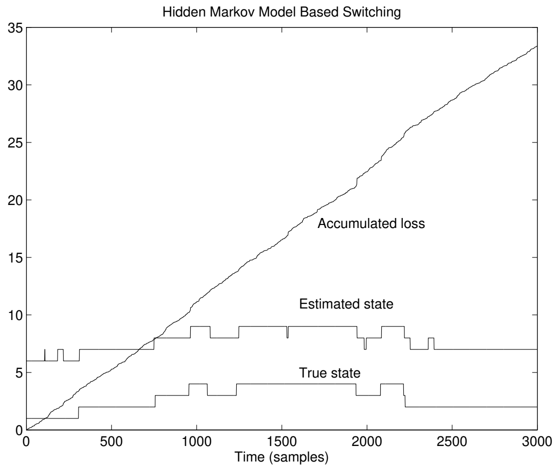

Figure 5: Example 1 - HMM switching.

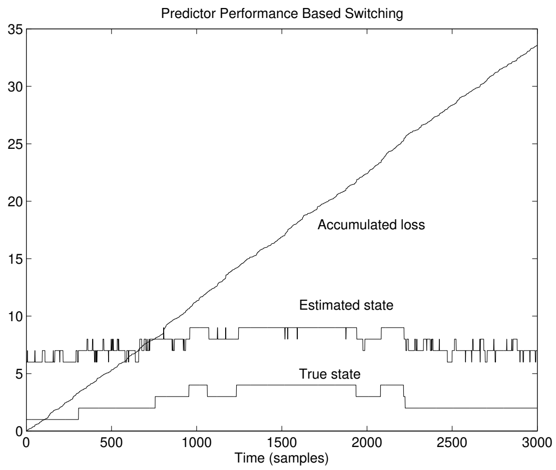

Figure 6: Example 1 - Predictor based switching.

4.1 Example 1

The plant is assumed to be second order, modelled by the difference equation with time varying coefficients:

$$y(t+1)-a(\lambda(t))y(t)+0.7y(t-1) = u(t)-0.5u(t-1)+\eta(t+1)$$

The coefficient a is a function of the state $\lambda(t)$ of a Markov chain with $N = 4$ states, taking the values: $a(\lambda_1) = -1.5$, $a(\lambda_2) = -1.0$, $a(\lambda_3) = -0.5$, $a(\lambda_4) = +0.5$. The step responses of each of these local models are shown in fig. 4. For the ARMAX plants resulting from a taking a value in each of the possible Markov chain states, a minimum variance controller (based on the full complexity active model) has been designed and is applied to the plant according to a dwell time strategy. The transition between states of the Markov chain occurs only every 10 samples, the state transition matrix being

$$A_M = \begin{bmatrix} 0.98 & 0.02 & 0.00 & 0.00 \\ 0.01 & 0.98 & 0.01 & 0.00 \\ 0.00 & 0.01 & 0.94 & 0.05 \\ 0.00 & 0.00 & 0.02 & 0.98 \end{bmatrix}$$

Fig. 5 shows the results obtained in a simulation run with the HMM based supervisor and fig. 6 with the predictor based supervisor (the estimated state index is shown added to 5 for clarity of presentation). Since this last strategy does not take into consideration the HMM model of state transition, the number of state estimation errors is bigger than in the first case, as expected. This is clearly seen in fig. 7, where the sequences of accumulated errors are shown for both supervisors.

4.2 Example 2

In example 2 the acquisition phase of the optimal supervisor is illustrated. The plant of example 1 is considered, but now a may assume one of the $N = 15$ values given by

$$a(\lambda_i) = -(0.5 + 0.1(i-1)) \quad i = 1, \ldots, 15$$

and the state transition matrix is given by $A_M = 0.9I + \overline{A}$ where I is the identity of order 15 and

$$\overline{A}(i, i-1) = \overline{A}(i, i+1) = 0.05 \quad 1 < i < 15$$

$$\overline{A}(1, 2) = \overline{A}(14, 15) = 0.1 \quad i = 1, 15$$

The (HMM) state estimate is initialized at state 1 and the true state is initialized at 10. Fig. 8 shows the time evolution of the HMM state probabilities, together with the true (dotted) and estimated state trajectories in the first 50 time steps.

Acknowledgement

The work of the first and second authors reported in this paper has been performed under contract PRAXIS XXI, 3/3.1/CEG/2605/95.

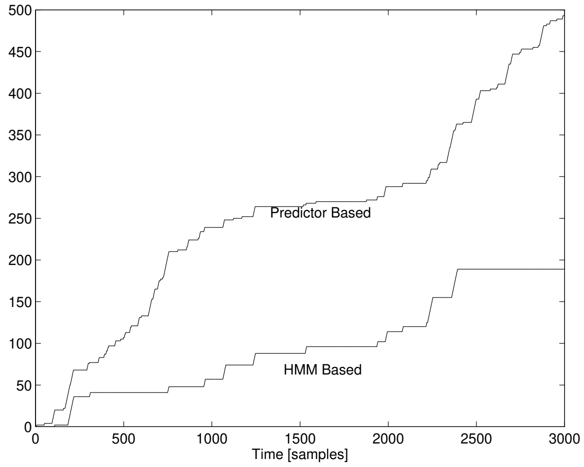

Figure 7: Example 1 - Accumulated errors.

5 Conclusion

A HMM based supervisor algorithm for application in switching reconfigurable control of time varying plants is presented and compared with a suboptimal strategy based on a bank of predictors. The use of a shared state representation provides a natural fitting of the HMM supervisor to reconfigurable control. In the special case in which the plant is unknown but time constant, the HMM supervisopr is equivalent to the minimum prediction error criterion. If however the plant changes in time, this last criterion is only suboptimal.

6 Appendix

A) Shared state realization

Let $C^*(q^{-1}) = q^{-\partial C}C(q) = 1 + \sum_{i=1}^{n} c_i q^{-i}$ and define

$$\zeta(t) \triangleq \frac{1}{C^*(q^{-1})}y(t) \quad \nu(t) \triangleq \frac{1}{C^*(q^{-1})}u(t) \tag{8}$$

Defining the state

$$x_\zeta(t) \triangleq \begin{bmatrix} \zeta(t) \\ \zeta(t-1) \\ \vdots \\ \zeta(t-n+1) \end{bmatrix} \tag{9}$$

the state-space realization of (8) is seen to be

$$x_\zeta(t) = \overline{\Phi}_o x_\zeta(t-1) + \overline{\Gamma}_o y(t) \tag{10}$$

where

$$\overline{\Phi}_o \triangleq \begin{bmatrix} c_1 & c_2 & \dots & & c_{n-1} & & c_n \\ -- & -- & -- & -- & -- & -- & -- \\ 1 & 0 & \dots & | & 0 & | & 0 \\ 0 & 1 & \dots & | & 0 & | & 0 \\ \vdots & \vdots & \dots & | & \vdots & | & \vdots \\ 0 & 0 & \dots & | & 0 & | & 0 \end{bmatrix} \tag{11}$$

$$\overline{\Gamma}_o = [1\ 0\ \dots\ 0]' \tag{12}$$

Similarly, for (8), define the state

$$x_\nu(t) \triangleq \begin{bmatrix} \nu(t) \\ \nu(t-1) \\ \vdots \\ \nu(t-n+1) \end{bmatrix} \tag{13}$$

The state-space realization of (8) is seen to be

$$x_\nu(t) = \overline{\Phi}_o x_\nu(t-1) + \overline{\Gamma}_o y(t) \tag{14}$$

Define the state

$$x \triangleq \begin{bmatrix} x_\zeta \\ --- \\ x_\nu \end{bmatrix} \tag{15}$$

The state x verifies eq. (3), in which

$$\Phi_o \triangleq \begin{bmatrix} \overline{\Phi}_o & | & \underline{O} \\ -- & - & -- \\ \underline{O} & | & \overline{\Phi}_o \end{bmatrix} \ \Gamma_{oy} \triangleq \begin{bmatrix} \overline{\Gamma}_o \\ -- \\ \underline{O} \end{bmatrix} \ \Gamma_{ou} \triangleq \begin{bmatrix} \underline{O} \\ -- \\ \overline{\Gamma}_o \end{bmatrix} \tag{16}$$

B)Proof of eqs.(5,6)

Let $p(\lambda_t|\mathcal{O}_t)$ be the probability density function (p.d.f.) of the state of the Markov chain at time t, λ_t, given the observations $\mathcal{O}_t$ up to time t. As shown in the appendix using Bayes law, $p(\lambda_t|\mathcal{O}_t)$ is recursively propagated in time using the nonlinear filtering operations:

1. *Prediction*

$$p(\lambda_t|\mathcal{O}_{t-1}) = \int_{-\infty}^{\infty} p(\lambda_t|\lambda_{t-1})p(\lambda_{t-1}|\mathcal{O}_{t-1})d\lambda_{t-1} \tag{17}$$

2. *Filtering*

$$p(\lambda_t|\mathcal{O}_t) = k_t p(y_t|\lambda_t)p(\lambda_t|\mathcal{O}_{t-1}) \tag{18}$$

Here, k_t is a normalizing constant such that

$$\int_{-\infty}^{\infty} p(\lambda_t|\mathcal{O}_t)d\lambda_t = 1$$

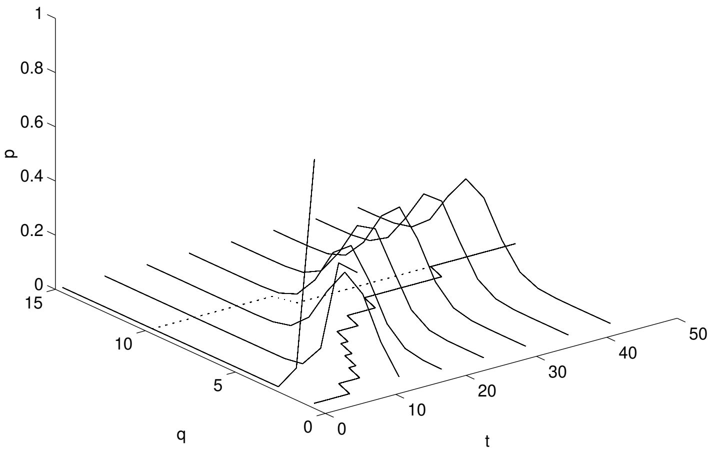

Figure 8: Example 2 - HMM state probabilities and estimated and true state.

Let

$$p(\lambda_{t-1}|\mathcal{O}_{t-1}) = \sum_{i=1}^{N} P_i(t-1)\delta(\lambda_{t-1} - S_i) \quad (19)$$

where $\delta(.)$ is the Dirac delta function. Replace (19) in (17) and use the properties of the Dirac delta function to get

$$p(\lambda_t|\mathcal{O}_{t-1}) = \sum_{i=1}^{N} P_i(t-1)p(\lambda_t|\lambda_{t-1} = S_i) \quad (20)$$

Write the equation defining the state transitions of the Markov chain as

$$p(\lambda_t|\lambda_{t-1} = S_i) = \sum_{j=1}^{N} a_{ij}\delta(\lambda_t - S_j) \quad (21)$$

and replace it in (20) to get, upon exchange of summations

$$p(\lambda_t|\mathcal{O}_{t-1}) = \sum_{j=1}^{N}(\sum_{i=1}^{N} a_{ij}P_i(t-1))\delta(\lambda_t - S_j) \quad (22)$$

which means that the probability $P_j(t|t-1)$ of state S_j at time t, given observations up to time $t-1$ is given by eq. (5).

Consider now the filtering operation (18). When the Markov chain is in state S_i, the observations are made according to eq. (4). Let the p.d.f. of $\eta(t)$ be gaussian

$$p_\eta(\alpha) = C_\eta \exp(-\frac{\alpha^2}{\sigma_\eta^2}) \quad (23)$$

with C_η and σ_η constants. Then

$$p(y_t|\lambda_t = S_i) = C_\eta \exp[-\frac{1}{\sigma_\eta^2}(y - x^T(t)\theta_i)^2]|_{y=y(t)} \quad (24)$$

By combining (18) and (24), for $j = 1, ..., N$, eq. (6) is yielded.

References

[1] Murray-Smith, R. and T. A. Johansen (1997). "*Multiple Model Approaches to Modelling and Control*", Taylor & Francis.

[2] Morse, A. S. (1996). "Supervisory control of families of linear set-point controllers - Part 1: Exact matching". *IEEE Trans. Aut. Control*, **41**, 1413-1431.

[3] Narendra, K. S. and J. Balakrishnan (1997). "Adaptive control using multiple models". *IEEE Trans. Aut. Control*, **42**, *2*, 171-187.

[4] Coito, F., J. M. Lemos, R. N. Silva and E. Mosca (1997). "Adaptive control of a solar energy plant: Exploiting accessible disturbances". *Int. J. Adaptive Control and Signal Proc.*, **11**, 327-342.

[5] Rabiner, L. R. (1989). "A tutorial on hidden Mrakov models and selected applications in speech recognition". *Proc. IEEE*, **77**, *2*, 257-286.

[6] Tugnait, J. K. (1982). "Detection and estimation for abruptly changing systems". *Automatica*, **18**, *5*, 607-615.

[7] H. Blom, Y. Bar-Shalom (1988). "The interacting multiple model algorithm for systems with Markovian switching coefficients". *IEEE Trans. Aut. Control*, **33**, *8*, 780-783.

[8] Isard, M. and A. Blake (1998). "A mixed-state condensation tracker with automatic model-switching", *Proc. Int. Conf. Computer Vision*, 107-112.

[9] C. Tsai and A. Haddad (1997). "Averaging, aggregation and optimal control of singularly perturbed stochastic hybrid systems", *Int. J. Control*, **68**, *1*, 31-50.

[10] Petridis, V. and A. Kehagias (1998). "A multi-model algorithm for parameter estimation of time-varying nonlinear systems", *Automatica*, **34**, *4*, 469-475.