where

$$\alpha(t)=a^{-1}\left[a\cos^2\frac{\pi t}{2}+\sin^2\frac{\pi t}{2}\right]$$
$$=a^{-1}\left[\frac{a+1}{2}+\frac{a-1}{2}\cos \pi t\right].$$

Therefore, from (4.3c)

$$\varphi(t-1)=b^{-1}\frac{a^2}{1-a^2}Q(t-1)+b^{-1}$$
$$=-b^{-1}\frac{a^{-t}\alpha(t)}{1-a^{-t}\alpha(t)}.$$

Substituting in (4.2) and recalling that $a > 1$, $0 < \gamma < 1$ gives immediately

$$v(t)=b^{-1}a^{-t}\alpha(t)\Delta a a^{t+1}+\epsilon(t)$$
$$=b^{-1}\Delta a\cdot a\cdot\alpha(t)+\epsilon(t)$$

with $\epsilon(t)$ a linear combination of exponentially decaying terms. □

REFERENCES

[1] G. C. Goodwin and K. S. Sin, *Adaptive Filtering, Prediction and Control.* Englewood Cliffs, NJ: Prentice-Hall, 1984.

[2] H. Elliott, "Direct adaptive pole placement with application to non-minimum phase systems," *IEEE Trans. Automat. Contr.*, vol. AC-27, pp. 720-722, 1982.

[3] G. C. Goodwin and E. K. Teoh, "Persistency of excitation in the presence of possibly unbounded signals," *IEEE Trans. Automat. Contr.*, vol. AC-30, pp. 589-592, 1985.

[4] M. Heymann, "Persistency of excitation results for structured nonminimal models," *IEEE Trans. Automat. Contr.*, vol. 33, pp. 112-116, 1988.

[5] R. L. Leal and G. C. Goodwin, "A globally convergent adaptive pole placement algorithm without a persistency of excitation requirement," in *Proc. 23rd Conf. Decision Contr.*, Las Vegas, NV, Dec. 1984.

[6] I. Mareels, "Sufficiency of excitation," *Syst. Contr. Lett.*, vol. 5, pp. 159-163, 1984.

[7] R. D. Nussbaum, "Some remarks on a conjecture in adaptive control," *Syst. Contr. Lett.*, vol. 3, pp. 243-246, Nov. 1983.

[8] S. Boyd and S. S. Sastry, "Necessary and sufficient conditions for parameter convergence in adaptive control," *Automatica*, vol. 22, pp. 629-639, 1986.

[9] S. Dasgupta, B. D. O. Anderson, and A. Tsoi, "Input conditions for continuous-time adaptive systems," in *Proc. CDC*, 1983, pp. 211-216.

[10] B. D. O. Anderson and R. Johnstone, "Global adaptive pole positioning," *IEEE Trans. Automat. Contr.*, vol. AC-30, pp. 11-21, 1985.

[11] H. Elliott, R. Cristi, and M. Das, "Global stability of adaptive pole-placement algorithms," *IEEE Trans. Automat. Contr.*, vol. AC-30, pp. 348-356, 1985.

The Interacting Multiple Model Algorithm for Systems with Markovian Switching Coefficients

HENK A. P. BLOM AND YAAKOV BAR-SHALOM

Abstract—**An important problem in filtering for linear systems with Markovian switching coefficients (dynamic multiple model systems) is the one of management of hypotheses, which is necessary to limit the computational requirements. A novel approach to hypotheses merging is presented for this problem. The novelty lies in the timing of hypotheses merging. When applied to the problem of filtering for a linear system with Markovian coefficients this yields an elegant way to derive the interacting multiple model (IMM) algorithm. Evaluation of the IMM algorithm makes it clear that it performs very well at a relatively low computational load. These results imply a significant change in the state of the art of approximate Bayesian filtering for systems with Markovian coefficients.**

Manuscript received June 24, 1987; revised October 21, 1987. This paper is based on a prior submission of October 20, 1986. The work of the second author was supported by the Air Force Office of Scientific Research under Grant 84-0112.

H. A. P. Blom is with the National Aerospace Laboratory, NLR, Amsterdam, The Netherlands.

Y. Bar-Shalom is with the University of Connecticut, Storrs, CT 06268.

IEEE Log Number 8821022.

I. INTRODUCTION

In this contribution we present a novel approach to the problem of filtering for a linear system with Markovian coefficients

$$x_t=a(\theta_t)x_{t-1}+b(\theta_t)w_t \tag{1}$$

with observations

$$y_t=h(\theta_t)x_t+g(\theta_t)v_t \tag{2}$$

θ_t is a finite state Markov chain taking values in $\{1,\cdots,N\}$ according to a transition probability matrix H, and w_t, v_t are mutually independent white Gaussian processes. The exact filter consists of a growing number of linear Gaussian hypotheses, with the growth being exponential with the time. Obviously, for filtering we need recursive algorithms whose complexity does not grow with time. With this, the main problem is to avoid the exponential growth of the number of Gaussian hypotheses in an efficient way.

This hypotheses management problem is also known for several other filtering situations [10], [5], [6], [9], and [4]. All these problems have stimulated during the last two decades the development of a large variety of approximation methods. For our problem the majority of these are techniques that reduce the number of Gaussian hypotheses, by pruning and/or merging of hypotheses. Well-known examples of this approach are the detection estimation (DE) algorithms and the generalized pseudo Bayes (GPB) algorithms. For overviews and comparisons see [14], [7], [12], and [17]. None of the algorithms discussed appeared to have good performance at modest computational load. Because of that, other approaches have been also developed, mainly by way of approximating the model (1), (2). Examples are the modified multiple model (MM) algorithms [20], [7], the modified gain extended Kalman (MGEK) filter of Song and Speyer [13], [7], and residual based methods [19], [2]. These algorithms, however, also lack good performance at modest computational load in too many situations. In view of this unsatisfactory situation and the practical importance of better solutions, the filtering problem for the class of systems (1), (2) needed further study.

One item that has not received much attention in the past is the timing of hypotheses reduction. It is common practice to reduce the number of Gaussian hypotheses immediately after a measurement update. Indeed, on first sight there does not seem to be a better moment. However, in two recent publications [3], [1], this point has been exploited to develop, respectively, the so-called IMM (interacting multiple model) and AFMM (adaptive forgetting through multiple models) algorithms. The latter exploits pruning to reduce the number of hypotheses, while the IMM exploits merging. The IMM algorithm was the reason for a further evaluation of the timing of hypotheses reduction. A novel approach to hypotheses merging is presented for a dynamic MM situation, which leads to an elegant derivation of the IMM algorithm. Next Monte Carlo simulations are presented to judge the state of the art in MM filtering after the introduction of the IMM algorithm.

II. TIMING OF HYPOTHESES REDUCTION

To show the possibilities of timing the hypothesis reduction, we start with a filter cycle from one measurement update up to and including the next measurement update. For this, we take a cycle of recursions for the evolution of the conditional probability measure of our hybrid state Markov process (x_t, θ_t). This cycle reads as follows:

$$P\{\theta_{t-1}|Y_{t-1}\}\xrightarrow{\text{Mixing}}P\{\theta_t|Y_{t-1}\} \tag{3}$$

if $P\{\theta_t | Y_{t-1}\} = 0$ prune hypothesis θ_t,

$$p[x_{t-1}|\theta_{t-1}, Y_{t-1}] \xrightarrow{\text{Mixing}} p[x_{t-1}|\theta_t, Y_{t-1}] \tag{4}$$

$$p[x_{t-1}|\theta_t, Y_{t-1}] \xrightarrow{\text{Evolution}} p[x_t|\theta_t, Y_{t-1}] \tag{5}$$

$$P\{\theta_t | Y_{t-1}\} \xrightarrow{\text{Bayes}} P\{\theta_t | Y_t\} \tag{6}$$

$$p[x_t|\theta_t, Y_{t-1}] \xrightarrow{\text{Bayes}} p[x_t|\theta_t, Y_t]. \tag{7}$$

For output purposes, we can use the law of total probability

$$p[x_t| Y_t] = \sum_i p[x_t|\theta_t = i, Y_t]P\{\theta_t = i | Y_t\}. \tag{8}$$

Let us take a closer look at the derivation of the above cycle. As v_t and w_t are mutually independent, the Bayes formula, which represents (6) and (7), follows easily from (2). From the evolution of system (1) follows (5). The Chapman-Kolmogorov equation for the Markov *chain* θ_t

$$P\{\theta_t = i | Y_{t-1}\} = \sum_j H_{ij} P\{\theta_{t-1} = j | Y_{t-1}\} \tag{9}$$

which represents (3), can be seen as a "mixing." To derive a representation of (4) we first introduce the following equation on the basis of the law of total probability:

$$p[x_{t-1}|\theta_t = i, Y_{t-1}] = \sum_j [p[x_{t-1}|\theta_{t-1} = j, \theta_t = i, Y_{t-1}] \cdot P\{\theta_{t-1} = j | \theta_t = i, Y_{t-1}\}]. \tag{10}$$

As θ_t is *independent* of x_{t-1} *if* θ_{t-1} is known, we easily obtain

$$p[x_{t-1}|\theta_{t-1} = j, \theta_t = i, Y_{t-1}] = p[x_{t-1}|\theta_{t-1} = j, Y_{t-1}].$$

Substitution of this and of the following:

$$P\{\theta_{t-1} = j | \theta_t = i, Y_{t-1}\} = H_{ij} P\{\theta_{t-1} = j | Y_{t-1}\}/P\{\theta_t = i | Y_{t-1}\}$$

in (10) yields the desired representation of transition (4)

$$p[x_{t-1}|\theta_t = i, Y_{t-1}] = \sum_j H_{ij} P\{\theta_{t-1} = j | Y_{t-1}\} \cdot p[x_{t-1}|\theta_{t-1} = j, Y_{t-1}]/P\{\theta_t = i | Y_{t-1}\}. \tag{11}$$

Notice that the mixing of the densities in (11) is explicitly related to the above-mentioned Markov properties of θ_t and the conditional independence of θ_t and x_{t-1}, given θ_{t-1}. According to the above filtering cycle there are at any moment in time N densities on R^n and N scalars. The densities on R^n are rarely Gaussian. Even if $p[x_0| Y_0]$ is Gaussian, then $p[x_t|\theta_t = i, Y_t]$ is in general a sum of N^{t-1} weighted Gaussians (Gaussian mixture). Explicit recursions for these N^t individual Gaussians and their weights can simply be obtained from the above filter cycle. Obviously, the N times increase of the number of Gaussians during each filter cycle is caused by (4) only.

In the sequence of elementary transitions, (3) through (7), we can apply a hypotheses reduction either after (4), after (5), or after (7). We review these reduction timing possibilities for the fixed depth merging hypotheses reduction. This fixed depth merging approach implies that the Gaussian hypotheses, for which the Markov chain paths are equivalent during the recent past of some fixed depth, are merged to one moment-matched Gaussian hypothesis. The degrees of freedom in applying this fixed depth merging approach are the choice of the depth, $d\ (\geq 1)$, and the moment of application. If the application is immediately after each measurement update pass (7), it yields the GPB (d + 1) algorithms [14], [16]. In the next section we derive the IMM algorithm by applying the fixed depth merging approach with depth, $d = 1$, after each pass of (4). It can easily be verified that all other timing possibilities yield disguised versions of IMM and GPB algorithms. Merging after (5) with $d = 1$ yields a disguised but more complex IMM algorithm. Merging either after (4) or after (5) with $d \geq 2$ yields a disguised but more complex GPBd algorithm.

III. The IMM Algorithm

The IMM algorithm cycle consists of the following four steps, of which the first three steps are illustrated in Fig. 1.

1) Starting with the N weights $\hat{p}_i(t-1)$, the N means $\hat{x}_i(t-1)$ and the N associated covariances $\hat{R}_i(t-1)$, one computes the mixed initial condition for the filter matched to $\theta_t = i$, according to the following equations:

$$\bar{p}_i(t) = \sum_j H_{ij}\hat{p}_j(t-1), \text{ if } \bar{p}_i(t) = 0 \text{ prune hypothesis } i, \tag{12}$$

$$\hat{x}^i(t-1) = \sum_j H_{ij}\hat{p}_j(t-1)\hat{x}_j(t-1)/\bar{p}_i(t), \tag{13}$$

$$\hat{R}^i(t-1) = \sum_j H_{ij}\hat{p}_j(t-1)[\hat{R}_j(t-1) + [\hat{x}_j(t-1) - \hat{x}^i(t-1)][.\ .]^T]/\bar{p}_i(t). \tag{14}$$

2) Each of the N pairs $\hat{x}^i(t-1)$, $\hat{R}^i(t-1)$ is used as input to a Kalman filter matched to $\theta_t = i$. Time-extrapolation yields, $\bar{x}_i(t)$, $\bar{R}_i(t)$, and then, measurement updating yields, $\hat{x}_i(t)$, $\hat{R}_i(t)$.

3) The N weights $\bar{p}_i(t)$ are updated from the innovations of the N Kalman filters,

$$\hat{p}_i(t) = c \cdot \bar{p}_i(t) \cdot \|Q_i(t)\|^{-1/2} \exp\{-1/2\vartheta_i^T(t)Q_i^{-1}(t)\vartheta_i(t)\} \tag{15}$$

with c denoting a normalizing constant

$$\vartheta_i(t) = y_t - h(i)\bar{x}_i(t) \tag{16}$$

$$Q_i(t) = h(i)\bar{R}_i(t)h^T(i) + g(i)g^T(i). \tag{17}$$

4) For output purpose only, $\hat{x}_t$ and $\hat{R}_t$ are computed according to

$$\hat{x}_t = \sum_i \hat{p}_i(t)\hat{x}_i(t) \tag{18}$$

$$\hat{R}_t = \sum_i \hat{p}_i(t)[\hat{R}_i(t) + [\hat{x}_i(t) - \hat{x}_t][.\ .]^T]. \tag{19}$$

Only step 1) is typical for the IMM algorithm. Specifically, the mixing represented by (13) and (14) and by the interaction box in Fig. 1, cannot be found in the GPB algorithms. This is the key of the novel approach to the timing of fixed depth hypotheses merging that yields the IMM algorithm. We give a derivation of the key step 1).

Application of fixed depth merging with $d = 1$ implies that

$$p[x_{t-1}|\theta_{t-1} = i, Y_{t-1}] \sim N\{\hat{x}_i(t-1), \hat{R}_i(t-1)\}.$$

Substitution of this in (11) immediately yields (13) and (14), with

$$\hat{x}^i(t-1) \triangleq E\{x_{t-1}|\theta_t = i, Y_{t-1}\}$$

and

$$\hat{R}^i(t-1)$$

the associated covariance. Finally, we introduce the approximation,

$$p[x_{t-1}|\theta_t = i, Y_{t-1}] \sim N\{\hat{x}^i(t-1), \hat{R}^i(t-1)\}$$

which guarantees that all subsequent IMM steps fit correctly.

Remark: The IMM can be approximated by the GPB1 algorithm by replacing $\hat{x}_j(t-1)$ and $\hat{R}_j(t-1)$ in step 1) by $\hat{x}_{t-1}$ and $\hat{R}_{t-1}$. Together with (12) this approximates (13) and (14) in step 1) by, $\hat{x}^i(t-1) \simeq \hat{x}_{t-1}$ and $\hat{R}^i(t-1) \simeq \hat{R}_{t-1}$. These equations are equivalent to (13) and (14) if each component of H equals $1/N$, which implies that θ_t is a sequence of mutually independent stochastic variables. The latter is hardly ever the case and we conclude that the reduction of the IMM to GPB1 leads to a significant performance degradation. Obviously, the computational loads of IMM and GPB1 are almost equivalent.

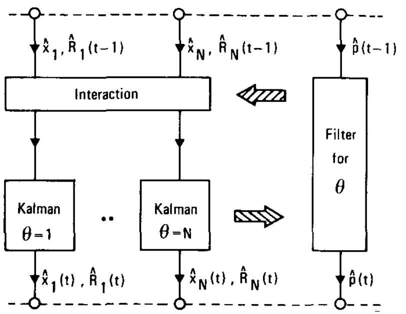

Fig. 1. The IMM algorithm.

IV. Performance of the IMM Algorithm

Presently a comparison of the different filtering algorithms for systems with Markovian coefficients with respect to their performance is hampered by the analytical complexity of the problem [16], [15]. Because of this, such comparisons necessarily rely on Monte Carlo simulations for specific examples. For our simulated examples we used the set of 19 cases that have been developed by Westwood [18]. To make the comparison more precise, we specify these cases and summarize the observed performance results. In all 19 cases both x_t and y_t are scalar processes, which satisfy $x_t = a(\theta_t)x_{t-1} + b(\theta_t)w_t + u(t)$ and $y_t = h(\theta_t)x_t + g(\theta_t)v_t$, with $\theta_t:\Omega \Rightarrow \{0, 1\}$, $u(t) = 10.\cos\{2\pi t/100\}$, x_0 a Gaussian variable with expectation 10 and variance 10, $P\{\theta_0 = 1\} = P\{\theta_0 = 0\} = 1/2$, while $H_{00} = (1 - 1/\tau_0)$ and $H_{11} = (1 - 1/\tau_1)$. The parameters a, b, h, g and the average sojourn times τ_0 and τ_1 of these 19 cases are given in Table I.

The results of Westwood [18] show that, in all 19 cases the differences in performance of the GPB2 and the GPB3 algorithms are negligible, while in only seven cases (5, 6, 8, 16, 17, 18, 19) the differences in performance of the GPB1 and the GPB2 algorithms are negligible. To our present comparison the other 12 cases (1, 2, 3, 4, 7, 9, 10, 11, 12, 13, 14, 15) are interesting. For each of these 12 cases we simulated the GPB1, the GPB2, and the IMM algorithms and ran Monte Carlo simulations, consisting of 100 runs from $t = 0$ to $t = 100$. For simplicity of interpretation of the results we used one fixed path of θ during all runs: $\theta = 0$ on the time interval [0, 30], $\theta = 1$ on the interval [31, 60], and $\theta = 0$ on the interval [61, 100].

The results of our simulations for the 12 interesting cases are as follows. In six cases (1, 2, 7, 12, 14, 15) both the IMM and the GPB2 performed slightly better than the GPB1, while the IMM and the GPB2 performed equally well. For typical results, see Fig. 2. In the other six cases both the IMM and the GPB2 performed significantly better than the GPB1. For typical results see Figs. 3 and 4. Of these six cases the IMM and the GPB2 performed four times equally well (cases 3, 4, 11, and 13) and two times significantly different (cases 9 and 10).

On the basis of these simulations we can conclude that the IMM performs almost as well as the GPB2, while its computational load is about that of GPB1. We can further differentiate this overall conclusion.

- Increasing the parameters τ_0 and τ_1 increases the difference in performance between GPB1 and GPB2, but not between IMM and GPB2.
- If a is being switched, then the IMM performs as well as the GPB2, while the GPB1 sometimes stays significantly behind.
- If the white noise gains, b or g, are being switched, then the IMM performs as well as the GPB2, while the GPB1 sometimes stays significantly behind.
- If only h is being switched, then in some cases the IMM, and even more often, the GPB1 tend to diverge while the GPB2 works well.

Another interesting question is how the IMM compares to the modified MM algorithm and the MGEK filter. Apart from the GPB algorithms, Westwood [18] also evaluated four more filters, the MM, the modified MM, the MGEK, and a MGEK with a "postprocessor." For the 19 cases there was only one algorithm that outperformed the GPB1 algorithm in some cases. It was the MGEK filter in the cases 1, 3, and 4. He also found that the MGEK filter performed in these cases marginally or significantly less good than the GPB2 algorithm. As the above experiments showed that for cases 1, 3, and 4 the GPB2 and the IMM algorithm performed equally well, one can conclude that the MM, the modified MM, the MGEK, the MGEK with "postprocessor," and the GPB1 are in all 19 cases outperformed by the IMM algorithm.

On the basis of these comparisons one can conclude that for practical filtering applications with $N = 2$, the IMM algorithm is the best first choice. As the IMM algorithm has been developed on the basis of some general hypotheses reduction principles, which are N-invariant, one can reasonably expect that this is also true for larger N. But it is unlikely that the IMM performs in all applications almost as good as the exact filter.

TABLE I
The Parameters of the 19 Cases of Westwood [18]

CASE	H-VALUES		θ-DEPENDENT VALUES			
#	τ_0	τ_1	a(0), a(1)	b(0), b(1)	h(0), h(1)	g(0), g(1)
1	40	20	.995,.990	1.0	1.0	1.0
2	40	20	.995,.990	.5	1.0	.5
3	40	20	.995,.990	.1	1.0	5.0
4	200	100	.995,.990	.1	1.0	5.0
5	40	20	.995,.990	8.0	1.0	1.0
6	40	20	.995,.990	1.0	1.0	.3
7	40	20	.995,.900	.5	1.0	2.0
8	40	20	.995,.750	1.0	1.0	.6
9	40	20	.995	2.0	1.0,.95	.5
10	40	20	.995	1.0	1.0,.80	.2
11	40	20	.995	.5	1.0,.80	.8
12	4	2	.995	.5	1.0,.80	.8
13	200	100	.995	.5	1.0,.80	.8
14	40	20	.995	.1,5.0	1.0	1.0
15	40	20	.995	1.0	1.0	.1,5.0
16	10	2	.95	.5	1.0,0.0	1.0,2.0
17	200	5	.950,0.0	1.0	1.0	1.0
18	50	5	.950,1.2	1.0	1.0	1.0
19	10	2	.95	.5	1.0	1.0,40.0

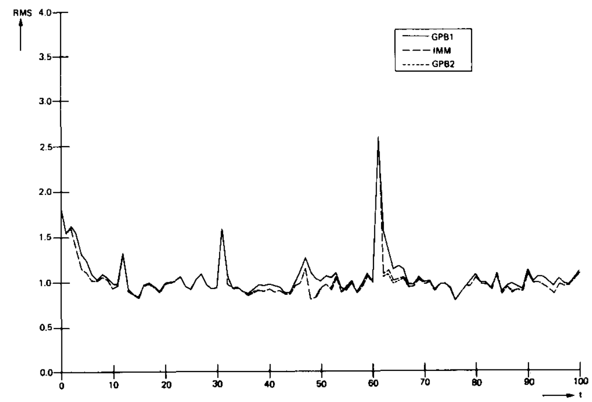

Fig. 2. rms error for case 7, illustrative of the six cases (1, 2, 7, 12, 14, 15) where both IMM and GPB2 perform slightly better than GPB1.

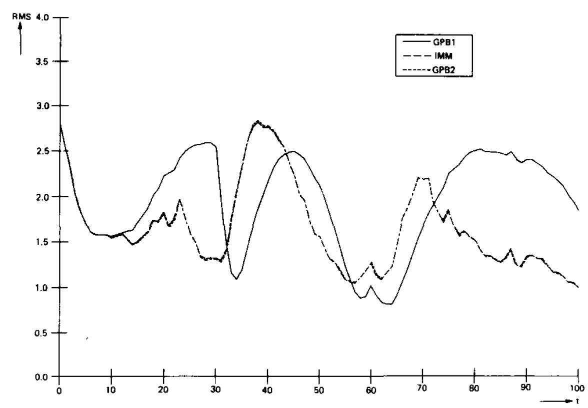

Fig. 3. rms error for case 3, illustrative of the four cases (3, 4, 11, 13) where both IMM and GPB2 perform better than GPB1, while IMM and GPB2 perform equally well.

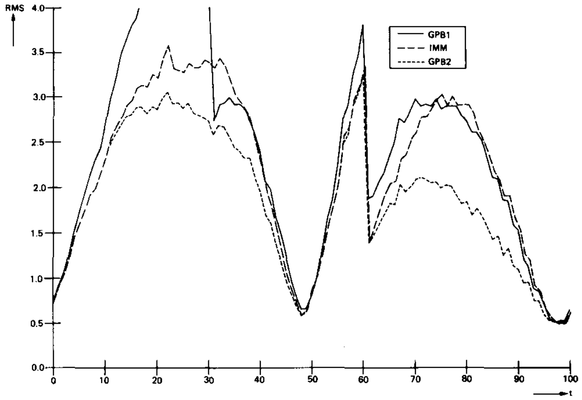

Fig. 4. rms error for case 9, illustrative of the two cases (9 and 10) where IMM performs better than GPB1, but slightly worse than GPB2 (in these two cases only h jumps).

Therefore, if the IMM performs not well enough in a particular application one should consider using a suitable GPB (≥ 2) or DE algorithm [14], or one might try to design a better algorithm by using adaptive merging techniques [16]. The DE algorithm might possibly be improved by the novel timing of hypotheses reduction [1]. If for a particular application the performance of the selected algorithm has a too high computational load, then it is best to try to exploit some geometrical structure of the problem considered [2], [11].

In situations where estimation has to be done outside some time-critical control loop, it is usually preferable to use a smoothing algorithm instead of a filtering algorithm [8], [14], [21]. In view of the above filtering results, this suggests that the ideas that underly the IMM algorithm can be exploited to develop better smoothing algorithms.

REFERENCES

[1] P. Andersson, "Adaptive forgetting in recursive identification through multiple models," *Int. J. Contr.*, vol. 42, pp. 1175–1193, 1985.
[2] M. Basseville and A. Benveniste, *Detection of Abrupt Changes in Signals and Dynamical Systems*. New York: Springer-Verlag, 1986.
[3] H. A. P. Blom, "An efficient filter for abruptly changing systems," in *Proc. 23rd IEEE Conf. Decision Contr.*, 1984, pp. 656–658.
[4] ——, "Overlooked potential of systems with Markovian coefficients," in *Proc. 25th IEEE Conf. Decision Contr.*, Athens, Greece, 1986, pp. 1758–1764.
[5] C. Y. Chong, S. Mori, E. Tse, and R. P. Wishner, "A general theory for Bayesian Multitarget tracking and classification," Advanced Decision Systems, Rep. TR-1015-1, 1982.
[6] A. M. Makowski, W. S. Levine, and M. Asher, "The nonlinear MMSE filter for partially observed systems driven by non-Gaussian white noise, with applications to failure estimation," in *Proc. 23rd IEEE Conf. Decision Contr.*, Las Vegas, NV, 1984, pp. 644–650.
[7] S. I. Marcus and E. K. Westwood, "On asymptotic approximation for some nonlinear filtering problems," in *Proc. 9th IFAC Triennial World Congress*, 1984, pp. 811–816.
[8] V. J. Mathews, and J. K. Tugnait, "Detection and estimation with fixed lag for abruptly changing systems," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 19, pp. 730–739, 1983.
[9] S. Mori, C. Y. Chong, E. Tse, and R. P. Wishner, "Tracking and classifying multiple targets without *a priori* identification," *IEEE Trans. Automat. Contr.*, vol. 31, pp. 401–409, 1986.
[10] K. R. Pattipati and N. R. Sandell, Jr., "A unified view of state estimation in switching environments," in *Proc. 1983 Amer. Contr. Conf.*, 1983, pp. 458–465.
[11] J. Raisch, "Comments on 'A multiple-model adaptive predictor for stochastic processes with Markov switching parameters,' " *Int. J. Contr.*, vol. 45, pp. 1489–1490, 1986.
linear systems," in *Analysis and Optimization of Stochastic Systems*, O. L. R. Jacobs *et al.* Eds. New York: Academic, 1980, pp. 333–345.
[13] T. L. Song and J. L. Speyer, "A stochastic analysis of a modified gain extended kalman filter with applications to estimation with bearings only measurements," in *Proc. 22nd IEEE Conf. Decision Contr.*, 1983, pp. 1291–1296.
[14] J. K. Tugnait, "Detection and estimation for abruptly changing systems," *Automatica*, vol. 18, pp. 607–615, 1982.
[15] R. B. Washburn, T. G. Allen, and D. Teneketzis, "Performance analysis for hybrid state estimation problems," in *Proc. 1985 Amer. Contr. Conf.*, 1985, pp. 1047–1053.
[16] J. L. Weiss, "A comparison of finite filtering methods for status directed processes," Master's thesis, Charles Stark Draper Lab., Mass. Inst. Technol., Rep. CSDL-T-819, 1983.
[17] J. L. Weiss, T. N. Upadhyay, and R. Tenney, "Finite computable filters for linear systems subject to time-varying model uncertainty," in *Proc. NAECON, 1983*, pp. 349–355.
[18] E. K. Westwood, "Filtering algorithms for the linear estimation problem with switching parameters," M.S. thesis, Univ. of Texas at Austin, 1984.
[19] A. S. Willsky, "Detection of abrupt changes in dynamic systems," Rep. MIT-LIDS-P-1351, 1984.
[20] A. S. Willsky, E. Y. Chow, S. B. Gershwin, C. S. Greene, P. K. Houpt, and A. L. Kurkjian, "Dynamic model-based techniques for the detection of incidents on freeways," *IEEE Trans. Automat. Contr.*, vol. 25, pp. 347–360, 1980.
[21] J. W. Woods, S. Dravida, and R. Mediavilla, "Image estimation using doubly stochastic Gaussian random field models," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-9, pp. 245–253, 1987.

Upper Bounds and Approximate Solutions for Multidisk Problems

THOMAS TING AND KAMESHWAR POOLLA

Abstract—**The objective of this note is to consider approximate solutions of multiobjective H^∞-optimization problems, also referred to as multidisk problems. The main result is the presentation of a systematic algorithm that enables us to compute an upper bound for linear two-disk problems and also a (suboptimal) controller that achieves this bound. This algorithm involves some graphical techniques which can also be used to explicitly demonstrate the design tradeoffs inherent in problems involving competing objectives. The results presented here can easily be generalized to incorporate multidisk problems.**

I. INTRODUCTION

Recently, the single objective H^∞-optimization problem, introduced by Zames [15] as an alternative to the classical Wiener-Hopf approach to feedback synthesis, has received much attention. This problem encompasses two fundamental problems in frequency domain control; the robust stabilization problem and the uniformly or H^∞-optimization problem. Both of these problems involve a single performance criterion and, in the context of LTI controllers, both problems reduce to the same mathematical problem, a one-disk problem. In practice, however, a control system designer must often consider multiple performance criteria, i.e., one may wish to simultaneously consider both the robust stability properties and the disturbance rejection capabilities of a particular control design. Therefore, several interesting and important control problems require the solution of a multiobjective H^∞-optimization problem, also referred to as a multidisk problem (see Francis and Doyle [6]). For example, the problems of

robust stabilization with optimal nominal disturbance rejection, (1.1)

robust simultaneous stabilization, (1.2)

optimal nominal disturbance rejection
with robust stability around a failure operating point (1.3)

all reduce to (perhaps nonlinear) multidisk problems.

In complete generality, these multidisk problems are very difficult and no closed-form solution is known. For certain special classes of multidisk

Manuscript received July 7, 1987; revised November 19, 1987. This work was supported in part by the Joint Services Electronics Program under Contract N00014-84-C-0149 and by the National Science Foundation under Grant ECS-87-09265.

The authors are with the Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, Urbana, IL 61801.

IEEE Log Number 8821565.

0018-9286/88/0800-0783$01.00 © 1988 IEEE