The investigations summarized in this report deal with: (a) adaptive dual control of systems with unknown parameters; (b) estimation and control of hybrid stochastic systems; (c) distributed estimation in systems with measurements of uncertain origin; and (d) solution of continuous-time hybrid stochastic differential equations.
DISCLAIMER NOTICE

THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.
Annual Report
Grant AFOG-89-0202
Stochastic Adaptive Control
and
Estimation Enhancement
Y. Bar-Shalom
September 1989
The main results obtained and published during the period covered by this report, August 1988 - July 1989, are described below together with references given to the corresponding publication.


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 has been developed 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.


An area of current interest is the estimation of the state of discrete-time stochastic systems with parameters which may switch among a finite set of values. The parameter switching process of interest is modeled by a class of semi-Markov chains. This class of processes is useful in that it pertains to many areas of interests such as the failure detection problem, the target tracking problem, socio-economic problems and in the problem of approximating nonlinear systems by a set of linearized models. It is shown in this paper how the transition probabilities, which govern the model switching at each time step, can be inferred via the evaluation of the conditional distribution of the sojourn time. Following this, a recursive state estimation algorithm for dynamic systems with noisy observations and changing structures, which uses the conditional sojourn time distribution, is derived and applied to a failure detection problem.

The probabilistic data association filter (PDAF) estimates the state of a target in a cluttered environment. This suboptimal Bayesian approach assumes that the exact target and measurement models are known. However, in most practical applications, there are difficulties in obtaining an exact mathematical model of the physical process. In this paper, the problem of estimating target states with uncertain measurement origins and uncertain system models in a distributed manner is considered. First, a scheme is described for local processing, then the fusion algorithm which combines the local processed results into a global one is derived. The algorithm can be applied for tracking a maneuvering target in a cluttered and low detection environment with a distributed sensor network.


An explicit adaptive dual controller has been derived for a multiinput multioutput ARMA system. The plant has constant but unknown parameters. The cautious controller with a one-step horizon and a new dual controller with a two-step horizon are examined. In many instances, the myopic cautious controller is seen to turn off and converges very slowly. The dual controller modifies the cautious control design by numerator and denominator correction terms which depend upon the sensitivity functions of the expected future cost and avoids the turn-off and slow convergence. Monte-Carlo comparisons based on parametric and nonparametric statistical analysis indicate the superiority of the dual controller over the cautious controller.

This paper develops the reversion in time of a stochastic difference equation in a hybrid space, with a Markovian solution. The reversion is obtained by a martingale approach, which previously led to reverse time forms for stochastic equations with Gauss-Markov or diffusion solutions. The reverse time equations follow from a particular non-canonical martingale decomposition, while the reverse time equations for Gauss-Markov and diffusion solutions followed from the canonical martingale decomposition. The need for the non-canonical decomposition stems from the hybrid state space situation. The non-Gaussian discrete time situation leads to reverse time equations that incorporate a Bayesian estimation step.


A realistic stochastic control problem for hybrid systems with Markovian jump parameters may have the switching parameters in both the state and measurement equations. Furthermore, both the system state and the jump states may not be perfectly observed. Prior to this work the only existing implementable controller for this problem was based upon a heuristic multiple model partitioning (MMP) and hypothesis pruning. In this paper a stochastic control algorithm for stochastic systems with Markovian jump parameters was developed. The control algorithm is derived through the use of stochastic dynamic programming and is designed to be used for realistic stochastic control problems, i.e., with noisy state observations. The state estimation and model identification is done via the recently developed Interacting Multiple Model algorithm. Simulation results show that a substantial reduction in cost can be obtained by this new control algorithm over the MMP scheme.

Piecewise Deterministic (PD) Markov processes form a remarkable class of hybrid state processes because, in contrast to most other hybrid state processes, they include a jump reflecting boundary and exclude diffusion. As such, they cover a wide variety of impulsively or singularly controlled non-diffusion processes. Because PD processes are defined in a pathwise way, they provide a framework to study the control of non-diffusion processes along same lines as that of diffusions. An important generalization is to include diffusion in PD processes, but, as pointed out by Davis, combining diffusion with a jump reflecting boundary seems not possible within the present definition of PD processes. This paper presents PD processes as pathwise unique solutions of an Itô stochastic differential equation (SDE), driven by a Poisson random measure. Since such an SDE permits the inclusion of diffusion, this approach leads to a large variety of piecewise diffusion Markov processes, represented by pathwise unique SDE solutions.
The Interacting Multiple Model Algorithm for Systems with Markovian Switching Coefficients

Henk A. P. Blom
Yaakov Bar-Shalom

Reprinted from IEEE TRANSACTIONS ON AUTOMATIC CONTROL
Vol. 33, No. 8, August 1988
I. INTRODUCTION

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

\[ x_t = A(\theta) x_{t-1} + B(\theta) w_t \]

with observations

\[ y_t = C(\theta) x_t + D(\theta) v_t \]

\( \theta \) is a finite state Markov chain taking values in \{1, \cdots, N\} according to a transition probability matrix \( P \) 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 (MGFK) 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 is the introduction 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 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 \((x_t, \theta)\). This cycle reads as follows:

\[ P(\theta_t|Y_{1:t}) \xrightarrow{\text{Marking}} P(\theta_t|Y_{1:t}) \]
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 \( \beta(t-1) \), the \( N \) means \( \bar{x}(t-1) \) and the \( N \) associated covariances \( \bar{R}(t-1) \), one computes the mixed initial condition for the filter matched to \( \theta_i = i \), according to the following equations:

\[
\beta_i(t) = \sum_{j=1}^{N} H_j \beta_j(t-1), \quad \text{if } \beta_i(t) = 0 \text{ prune hypothesis } i.
\]

\[
\bar{X}(t-1) = \sum_{j=1}^{N} H_j \bar{x}_j(t-1) \overline{R}_j \bar{R}_j(t-1)^{-1} \bar{x}_j(t-1) - \bar{X}(t-1).
\]

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

3) The \( N \) weights \( \beta_i(t) \) are updated from the innovations of the \( N \) Kalman filters,

\[
\beta_i(t) = c \cdot \beta_i(t) \cdot [Q_i(t)]^{-1/2} \exp \{-1/2 \bar{x}_i(t) Q^{-1}_i(t) \bar{x}_i(t)\}
\]

with \( c \) denoting a normalizing constant

\[
\sigma_i(t) = y_i - h(i) \bar{x}_i(t)
\]

\[
Q_i(t) = h(i) \bar{R}_i(t) h(i)^T + \gamma(t) \gamma(t)^T.
\]

4) For output purpose only, \( \bar{x} \) and \( \bar{R} \) are computed according to

\[
\bar{x} = \sum_i \beta_i(t) \bar{x}_i(t)
\]

\[
\bar{R} = \sum_i \beta_i(t) \bar{R}_i(t) + [\bar{x}(t) - \bar{x}][\bar{x}(t) - \bar{x}].
\]

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 GBB 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

\[
p[x(t) | \theta = i, Y_t] = N[\bar{x}(t-1), \bar{R}(t-1)]
\]

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

\[
\bar{X}(t-1) = \bar{X}[x(t), \theta = i, Y_t]
\]

and

\[
\bar{R}(t-1) = \bar{R}[(t-1)]
\]

the associated covariance. Finally, we introduce the approximation,

\[
p[x(t) | \theta = i, Y_t] = N[\bar{X}(t-1), \bar{R}(t-1)]
\]

which guarantees all subsequent IMM steps fit correctly.

Remark: The IMM can be approximated by the GBB algorithm by replacing \( \bar{X}(t-1) \) and \( \bar{R}(t-1) \) in step 1) by \( \bar{x}_i \) and \( \bar{R}_i \). Together with (12) this approximates (13) and (14) in step 1) by \( \bar{x}(t-1) = \bar{x}_i \) and \( \bar{R}(t-1) = \bar{R}_i \). These equations are equivalent to (13) and (14) if each component of \( H \) equals \( 1/N \), which implies that \( \theta_i \) is a sequence of mutually independent stochastic variables. The latter is hardly the case and we conclude that the reduction of the IMM to GBB leads to a significant performance degradation. Obviously, the computational loads of IMM and GBB are almost equivalent.
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 and y are scalar processes, which satisfy \( x_k = g_i(x_{k-1}) + \eta_k \) and \( y_k = h_i(x_k) + \xi_k \). In 15 cases both \( \eta_k \) and \( \xi_k \) are Gaussian; in the other four \( \eta_k \) and \( \xi_k \) are either white or colored, respectively. The parameters \( a, b, h, g \) and the average sojourn times \( r_i \) and \( \tau_i \) 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, 9, 10, 11, 12, 13, 14, 15) are interesting. For each of these 12 cases we simulated the GPBI, 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 \( \theta \) 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. 2 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 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 \( \tau_0 \) and \( \tau_1 \) increases the difference in performance between GPB1 and GPB2, but not between IMM and GPB2.

* If \( \sigma \) 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

### Table I

<table>
<thead>
<tr>
<th>CASE</th>
<th>H-VALUES</th>
<th>( \theta )-DEPENDENT VALUES</th>
</tr>
</thead>
<tbody>
<tr>
<td>#</td>
<td>( T_0 )</td>
<td>( T_1 )</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>20</td>
</tr>
<tr>
<td>2</td>
<td>40</td>
<td>20</td>
</tr>
<tr>
<td>3</td>
<td>40</td>
<td>20</td>
</tr>
<tr>
<td>4</td>
<td>200</td>
<td>100</td>
</tr>
<tr>
<td>5</td>
<td>40</td>
<td>20</td>
</tr>
<tr>
<td>6</td>
<td>40</td>
<td>20</td>
</tr>
<tr>
<td>7</td>
<td>40</td>
<td>20</td>
</tr>
<tr>
<td>8</td>
<td>40</td>
<td>20</td>
</tr>
<tr>
<td>9</td>
<td>40</td>
<td>20</td>
</tr>
<tr>
<td>10</td>
<td>40</td>
<td>20</td>
</tr>
<tr>
<td>11</td>
<td>40</td>
<td>20</td>
</tr>
<tr>
<td>12</td>
<td>40</td>
<td>20</td>
</tr>
<tr>
<td>13</td>
<td>200</td>
<td>100</td>
</tr>
<tr>
<td>14</td>
<td>40</td>
<td>20</td>
</tr>
<tr>
<td>15</td>
<td>40</td>
<td>20</td>
</tr>
<tr>
<td>16</td>
<td>40</td>
<td>20</td>
</tr>
<tr>
<td>17</td>
<td>40</td>
<td>20</td>
</tr>
<tr>
<td>18</td>
<td>50</td>
<td>5</td>
</tr>
<tr>
<td>19</td>
<td>10</td>
<td>2</td>
</tr>
</tbody>
</table>
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


Failure Detection Via Recursive Estimation for a Class of Semi-Markov Switching Systems

I. Campo1, P. Mookerjee2, and Y. Bar-Shalom3

1. Introduction

In this paper we are concerned with failure detection via recursive estimation of parameters in dynamic systems. The topic of interest is stochastic systems with abrupt changes, i.e., model jumps. The recursive state estimation algorithm for this problem developed in this paper provides the conditional model structure of parameters which signify component failures.

The abruptly changing parameters, which switch among a finite set of values, are modeled as a Markov chain. This class of processes is useful in that it pertains to many areas of interest such as the failure detection problem, the target tracking problem, socio-economic problems and the problem of approximating nonlinear systems by a set of linearized models. It is shown in this paper how the transition probabilities, which govern the model switching at each time step, can be inferred via the evaluation of the conditional distribution of the sojourn time. Following this, a recursive state estimation algorithm for dynamic systems with noisy observations and changing structures, which uses the conditional sojourn time distribution, is derived.

2. Formulation of the problem

The system is modeled by the equations

\[ x(k) = F[M(k)] x(k-1) + v(k-1), M(k) \]

\[ z(k) = H[M(k)] x(k) + w(k) \]

where \( M(k) \) denotes the model at time \( k \) and \( v(k) \) and \( w(k) \) are white and mutually uncorrelated.

The model at time \( k \) is assumed to be among the possible \( r \) models

\[ M(k) \in \{1, \ldots, r\} \]

\[ F[M(k)] = f_j \]

\[ v(k-1|M(k)] = \mu_j \]

The first treatment of estimation in a switching environment was in [A1] where the means and covariances of the process and measurement noises experienced jumps. As indicated in [11], the optimum state estimation in a multiple model environment is a function of the elemental ("model-matched") state estimates obtained via estimators tuned to all possible parameter histories. Thus, with time, the estimator must keep track of an exponentially growing number of parameter history hypotheses. Even in the case of Markov switching the estimation algorithm requires exponentially growing memory. In [12] Suboptimal algorithms like the Generalized Pseudo-Bayesian Algorithm (GPB) and the Interacting Multiple Model Algorithm (IMM) are viable approaches to obtain a real-time implementable estimation algorithm. These algorithms rely on different hypothesis merging techniques to limit the memory and computational requirements.

In [21] a Markov switching problem was considered, but the jumps were assumed to be perfectly observed. In [9] an estimation scheme for semi-Markov processes was developed based upon the detection-estimation algorithm (DEA). The approach is obtained by retaining a certain number of most likely parameter history hypotheses. The estimation schemes based upon the DEA (which discards all but a number of most likely hypothesis) and the GPB or IMM (which use hypothesis merging) algorithms represent different philosophies of algorithm design. We present an example comparing the two methods for a particular state estimation problem later in this paper.

The problem is formulated in Section 2. In Section 3 the sojourn time conditional probability mass functions and the conditional transition probabilities which we derived in [M1] are given here for clarity and ease of reference. The inclusion of Section 4, the state estimation algorithm which was developed in [M1], is for the sake of completeness.

1 University of Connecticut
Storrs, CT 06268
Supported by AFOSR Grant 84-0112

2. Villanova University
Villanova, PA 19085
Supported by the Grant from the Vice-President for Academic Affairs Office, Villanova University and AFOSR Grant 84-0112

3. Proceedings 1980
CDC - Austin

Abstract

An area of current interest is the estimation of the state of discrete-time stochastic systems with parameters which may switch among a finite set of values. The parameter switching process of interest is modeled by a class of semi-Markov chains. This class of processes is useful in that it pertains to many areas of interest such as the failure detection problem, the target tracking problem, socio-economic problems and the problem of approximating nonlinear systems by a set of linearized models. It is shown in this paper how the transition probabilities, which govern the model switching at each time step, can be inferred via the evaluation of the conditional distribution of the sojourn time. Following this, a recursive state estimation algorithm for dynamic systems with noisy observations and changing structures, which uses the conditional sojourn time distribution, is derived.

1. Introduction

In this paper we are concerned with failure detection via recursive estimation of parameters in discrete-time dynamic systems. The topic of interest is stochastic systems with abrupt changes, i.e., model jumps. The recursive state estimation algorithm for this problem developed in this paper provides the conditional model structure of parameters which signify component failures.

The abruptly changing parameters, which switch among a finite set of values, are modeled as a Markov chain. This class of processes is useful in that it pertains to many areas of interest such as the failure detection problem, the target tracking problem, socio-economic problems and the problem of approximating nonlinear systems by a set of linearized models. It is shown in this paper how the transition probabilities, which govern the model switching at each time step, can be inferred via the evaluation of the conditional distribution of the sojourn time. Following this, a recursive state estimation algorithm for dynamic systems with noisy observations and changing structures, which uses the conditional sojourn time distribution, is derived.

1. Introduction

In this paper we are concerned with failure detection via recursive estimation of parameters in discrete-time dynamic systems. The topic of interest is stochastic systems with abrupt changes, i.e., model jumps. The recursive state estimation algorithm for this problem developed in this paper provides the conditional model structure of parameters which signify component failures.

The abruptly changing parameters, which switch among a finite set of values, are modeled as a Markov chain. This class of processes is useful in that it pertains to many areas of interest such as the failure detection problem, the target tracking problem, socio-economic problems and the problem of approximating nonlinear systems by a set of linearized models. It is shown in this paper how the transition probabilities, which govern the model switching at each time step, can be inferred via the evaluation of the conditional distribution of the sojourn time. Following this, a recursive state estimation algorithm for dynamic systems with noisy observations and changing structures, which uses the conditional sojourn time distribution, is derived.

1. Introduction

In this paper we are concerned with failure detection via recursive estimation of parameters in discrete-time dynamic systems. The topic of interest is stochastic systems with abrupt changes, i.e., model jumps. The recursive state estimation algorithm for this problem developed in this paper provides the conditional model structure of parameters which signify component failures.

The abruptly changing parameters, which switch among a finite set of values, are modeled as a Markov chain. This class of processes is useful in that it pertains to many areas of interest such as the failure detection problem, the target tracking problem, socio-economic problems and the problem of approximating nonlinear systems by a set of linearized models. It is shown in this paper how the transition probabilities, which govern the model switching at each time step, can be inferred via the evaluation of the conditional distribution of the sojourn time. Following this, a recursive state estimation algorithm for dynamic systems with noisy observations and changing structures, which uses the conditional sojourn time distribution, is derived.

1. Introduction

In this paper we are concerned with failure detection via recursive estimation of parameters in discrete-time dynamic systems. The topic of interest is stochastic systems with abrupt changes, i.e., model jumps. The recursive state estimation algorithm for this problem developed in this paper provides the conditional model structure of parameters which signify component failures.

The abruptly changing parameters, which switch among a finite set of values, are modeled as a Markov chain. This class of processes is useful in that it pertains to many areas of interest such as the failure detection problem, the target tracking problem, socio-economic problems and the problem of approximating nonlinear systems by a set of linearized models. It is shown in this paper how the transition probabilities, which govern the model switching at each time step, can be inferred via the evaluation of the conditional distribution of the sojourn time. Following this, a recursive state estimation algorithm for dynamic systems with noisy observations and changing structures, which uses the conditional sojourn time distribution, is derived.

1. Introduction

In this paper we are concerned with failure detection via recursive estimation of parameters in discrete-time dynamic systems. The topic of interest is stochastic systems with abrupt changes, i.e., model jumps. The recursive state estimation algorithm for this problem developed in this paper provides the conditional model structure of parameters which signify component failures.

The abruptly changing parameters, which switch among a finite set of values, are modeled as a Markov chain. This class of processes is useful in that it pertains to many areas of interest such as the failure detection problem, the target tracking problem, socio-economic problems and the problem of approximating nonlinear systems by a set of linearized models. It is shown in this paper how the transition probabilities, which govern the model switching at each time step, can be inferred via the evaluation of the conditional distribution of the sojourn time. Following this, a recursive state estimation algorithm for dynamic systems with noisy observations and changing structures, which uses the conditional sojourn time distribution, is derived.

1. Introduction

In this paper we are concerned with failure detection via recursive estimation of parameters in discrete-time dynamic systems. The topic of interest is stochastic systems with abrupt changes, i.e., model jumps. The recursive state estimation algorithm for this problem developed in this paper provides the conditional model structure of parameters which signify component failures.

The abruptly changing parameters, which switch among a finite set of values, are modeled as a Markov chain. This class of processes is useful in that it pertains to many areas of interest such as the failure detection problem, the target tracking problem, socio-economic problems and the problem of approximating nonlinear systems by a set of linearized models. It is shown in this paper how the transition probabilities, which govern the model switching at each time step, can be inferred via the evaluation of the conditional distribution of the sojourn time. Following this, a recursive state estimation algorithm for dynamic systems with noisy observations and changing structures, which uses the conditional sojourn time distribution, is derived.

1. Introduction

In this paper we are concerned with failure detection via recursive estimation of parameters in discrete-time dynamic systems. The topic of interest is stochastic systems with abrupt changes, i.e., model jumps. The recursive state estimation algorithm for this problem developed in this paper provides the conditional model structure of parameters which signify component failures.

The abruptly changing parameters, which switch among a finite set of values, are modeled as a Markov chain. This class of processes is useful in that it pertains to many areas of interest such as the failure detection problem, the target tracking problem, socio-economic problems and the problem of approximating nonlinear systems by a set of linearized models. It is shown in this paper how the transition probabilities, which govern the model switching at each time step, can be inferred via the evaluation of the conditional distribution of the sojourn time. Following this, a recursive state estimation algorithm for dynamic systems with noisy observations and changing structures, which uses the conditional sojourn time distribution, is derived.

1. Introduction

In this paper we are concerned with failure detection via recursive estimation of parameters in discrete-time dynamic systems. The topic of interest is stochastic systems with abrupt changes, i.e., model jumps. The recursive state estimation algorithm for this problem developed in this paper provides the conditional model structure of parameters which signify component failures.

The abruptly changing parameters, which switch among a finite set of values, are modeled as a Markov chain. This class of processes is useful in that it pertains to many areas of interest such as the failure detection problem, the target tracking problem, socio-economic problems and the problem of approximating nonlinear systems by a set of linearized models. It is shown in this paper how the transition probabilities, which govern the model switching at each time step, can be inferred via the evaluation of the conditional distribution of the sojourn time. Following this, a recursive state estimation algorithm for dynamic systems with noisy observations and changing structures, which uses the conditional sojourn time distribution, is derived.

1. Introduction

In this paper we are concerned with failure detection via recursive estimation of parameters in discrete-time dynamic systems. The topic of interest is stochastic systems with abrupt changes, i.e., model jumps. The recursive state estimation algorithm for this problem developed in this paper provides the conditional model structure of parameters which signify component failures.

The abruptly changing parameters, which switch among a finite set of values, are modeled as a Markov chain. This class of processes is useful in that it pertains to many areas of interest such as the failure detection problem, the target tracking problem, socio-economic problems and the problem of approximating nonlinear systems by a set of linearized models. It is shown in this paper how the transition probabilities, which govern the model switching at each time step, can be inferred via the evaluation of the conditional distribution of the sojourn time. Following this, a recursive state estimation algorithm for dynamic systems with noisy observations and changing structures, which uses the conditional sojourn time distribution, is derived.
A semi-Markov (SM) chain [H1, H2, R1] is characterized by a fixed matrix of transition probabilities \([p_i]\) and a matrix of sojourn time probability density functions \([f_i(t)]\), which are functions of the current state \(i\) as well as the destination state \(j\) of the transition. In a SM chain first the destination of the jump is chosen according to \([p_i]\) and then the time after which the jump takes place (i.e., the sojourn time) is chosen according to \([f_i(t)]\). In this model the process can undergo a virtual transition (i.e., jump in place) \(j\). However, in this case, the sojourn time counting is still restarted even though the system has been in state \(i\) for some time.


The process \(M(k)\), \(k=0,1,...\), which represents the system model, can exist in one of \(r\) possible states. The current state of transition for the STOM process [chain] are functions of the sojourn time \(\tau\) and are defined as

\[
P_i(\tau) = P(M(k)=i|M(k-1)=\tau) = p_i(\tau)\]  

where \(\tau_{i+1}\) is the sojourn time in state \(i\) at time \(k-1\). It is assumed that at \(k=0\) the sojourn time (in whatever state the system model is) is \(\tau_0\). Thus the values \(\tau\) can take are from \(0\) to the maximum, which at time \(k=1\) is then \(k\).

Let \(Z(k)\) be a noisy measurement of the state of the dynamic system whose model undergoes transitions according to the above described STOM process. Based on the available information \(Z(k)\), the probability of the model process being in state \(i\), denoted as \(u(k)\), is defined as

\[
u_i(k) = P(M(k)=i|Z(k)) \quad i=1,...,r
\]

The conditional pmf of the sojourn time in state \(M(k)\) based on the available information \(Z(k)\) at time \(k\) is

\[
g_i(\tau) = P(\tau(k)=\tau|M(k)=i,Z(k)) = P(\tau(1)=\tau|M(k-1)=\tau,Z(k-1)) = P(\tau(k)=\tau|M(k-1)=\tau,Z(k-1)) = \frac{f_i(\tau)}{\sum_{\tau'} f_i(\tau')} \]

where the perfect knowledge of the state \(M(k)\) allows one to go down to one index less in the conditioning, i.e., \(Z(k-1)\).

Following (3.1) the conditional probability of transition from \(i\) to \(j\) at time \(k-1\) given the observations \(Z(k-1)\) is, in terms of (3.3),

\[
p_{ij}(k-1) = P(M(k)=j|M(k-1)=i,Z(k-1)) = \sum_{\tau} P(M(k)=j|\tau,k=M(k-1)-i,Z(k-1)) \cdot \frac{f_i(\tau)}{\sum_{\tau'} f_i(\tau')}
\]

Note that the argument of \(p_{ij}\) defined in (3.3) is the sojourn time while the argument of \(g_i(\tau)\) defined above is the current time. The conditional probability mass function (3.3) of the sojourn time \(\tau\) in state \(i\) at time \(k\) is given by the following expressions

\[
g_i(\tau) = \sum_{\tau_{i+1}} P(M(k)=i|M(k-1)=i,Z(k-1)) \left( \frac{f_i(\tau)}{\sum_{\tau'} f_i(\tau')} \right)
\]

Expressions (3.3) are proven by induction in [M1]. The notations \(a_{ij}\) and \(b_{ij}\) used above are defined below.

The probability that the process will stay \(s\) time steps in the same state \(i\) as it is at time \(k-s\) is conditioned on the information at \(k-s\), given by the expression

\[
b_{ij}(s) = P(M(k-s)=i|M(k-s)=i,Z(k-s))
\]

Conditioned on the available information \(Z(k-s)\) at time \(k-s\), the joint probability of the system residing in the same state \(i\) for the next \(s\) time steps is denoted as

\[
a_{ij}(s) = P(M(k)=i|M(k-s)=i,Z(k-s)) = \sum_{\tau} P(M(k)=i|M(k-s)=i,Z(k-s)) \left( \frac{f_i(\tau)}{\sum_{\tau'} f_i(\tau')} \right)
\]

4. The state estimation algorithm

As indicated in Sec. 1, the optimal estimator for linear systems with Markov model jumps requires an exponentially increasing memory. Among the suboptimal approaches discussed, it appears that the IMM is the most cost-effective in implementation [B4]. In view of this, the state estimation for a linear system with sojourn-time-dependent transition probabilities is developed in the sequel based on the IMM approach.

In this approach, at time \(k\), the state estimation is computed under each possible model hypothesis using \(r\) filters (for the \(r\) possible models), with each filter using a different combination of the previous model-conditioned estimates. Each model transition probability is a known function of the sojourn time given by (3.3). Each model has a sojourn time \(\tau_{ij}\) in state \(i\) which is, however, not known. The filter has access only to the observations from which the conditional pmf of the sojourn time \(Z_{ij}\) can be obtained.
this in turn is to be used in calculation of the conditional transition probabilities (3.5).

To find the conditional pdf of the state of the dynamic system described by (2.1)-(2.3) the total probability theorem is used as follows:

\[ p(x[k]|z[k]) = \sum_{r} p(x[k]|M(k)_{j}, z[k], z_{-1}^{k-1}) p(M(k)_{j}|z_{-1}^{k-1}) \]

where \( j = 1, \ldots, r \) is the last term above. The total probability theorem is now applied to this process with the irrelevant

\[ p(x[k]|M(k)_{j}, z[k], z_{-1}^{k-1}) = \frac{1}{r} \sum_{l=1}^{r} p(x[k]|M(k)_{l}, z[k], z_{-1}^{k-1}) u_{j,l} \]

reflecting one cycle of the state estimation filtering matched to model \( j \) of the IMM. The conditional transition probabilities, \( u_{j,l} \), are given by (2.11)-(2.12) [4.1, 4.2]

\[ u_{j,l} = P(M(k)|M(k)_{l}|z_{-1}^{k-1}) \]

and

\[ u_{j,j-1} = P(M(k)|M(k)_{j-1}|z_{-1}^{k-1}) \] \hspace{1cm} (4.5)

Note that Eq. (4.3) represents a Gaussian mixture under the typical Gaussian assumptions on the noise terms in Eqs. (2.1) and (2.2). This mixture is then approximated by a single moment-matched Gaussian. The following it follows that the input to the filter matched to model \( j \), \( j = 1, \ldots, r \), is obtained from an interaction of these \( r \) filters. This interaction consists of the mixing of the estimates of \( x'[k-lk-l] \) according to the weighting probabilities \( u_{j,l} \). The evaluation of the probabilities (4.4) and (4.5) in the STOM situation, are the key results needed to obtain a recursive state estimation algorithm of this type of model switching. These probabilities are shown below to follow from the results in Section 3.

Fig. 4.1 describes the result interaction Multiple Model (IMM) algorithm, which consists of \( r \) interacting filters operating in parallel. The mixing is done at the input of the filters with the probabilities, detailed later in (4.7), conditioned on \( z_{-1}^{k-1} \).

One cycle of the algorithm consists of the following:

Starting with the model-conditioned estimate \( x'[k-lk-l] \), with associated covariance \( P(k-lk-l) \), one computes the mixed initial condition for the filter matched to \( M(k)_{j} \) according to (4.3) as follows:

\[ x'[k-lk-l] = \frac{1}{r} \sum_{i} x'(k-lk-l) u_{i,l} \] \hspace{1cm} (4.6)

From (4.5)

\[ u_{i,j} = \sum_{l} p(M(k)|M(k)_{l}|z_{-1}^{k-1} \) \]

\[ = \frac{1}{r} p(M(k)|M(k)_{j}, z_{-1}^{k-1}) \]

This is the key step of the IMM that yields an algorithm with fixed (and modest) computational requirements: using \( r \) filters it yields performance comparable to the Generalized Pseudo Bayesian algorithm with \( r^2 \) filters [84].

where the notations from (4.1) and (3.5) were used and

\[ x'[k-lk-l] \in \mathbb{R}^{r}(x[k-lk-l]|M(k)_{j}, z_{-1}^{k-1}) \]

is the model-conditioned state estimate at time \( k \). The expression of \( \hat{p} \) for the STOM case using terms involving sojourn time probabilities is the one obtained in (3.5). The covariance corresponding to (4.6) is

\[ P(k-lk-l) = \sum_{j} u_{j,l} (P(k-lk-l)) p(k-lk-l) \]

(4.7)

The state estimate \( \hat{x} \) and covariance \( P(k) \) are used as input to a standard Kalman filter matched to \( M(k)_{j} \) to yield the model-conditioned estimate

\[ \hat{x}[k] = \sum_{j} \hat{x}[k] u[j] \] \hspace{1cm} (4.8)

where the conditional transition probabilities, \( u_{j,l} \), are as given in (4.7).

Eqs. (4.7) and (4.8) in combination with \( p_{j} \) are the key results that make possible the estimation of the state for a system with sojourn-time-dependent model transitions. Finally, for output only, the latest state estimation and covariance are obtained according to Eqs (4.1) and (4.3) as

\[ \hat{x}[k] + \sum_{j} \hat{x}[k] u[j] \] \hspace{1cm} (4.9)

\[ \hat{P}(k) = \sum_{j} \hat{P}(k) u[j] \]

(4.10)

S. Simulation Results

The algorithm developed in Sec. 4 using the sojourn time pmf obtained in Sec. 3 is used to estimate the state of the system. In the first example the results of this STOM-based IMM estimation scheme are compared with results obtained from an IMM algorithm based upon a Markov model transition assumption. In the second example the STOM-based IMM estimation scheme is compared to the detection-estimation algorithm of [9]. It is assumed that an STOM process described in Sec. 2 governs the switching between models. In the following T is the sampling period and \( k \) is an integer representing the number of sampling periods since time zero.

Example 1

The estimation of a controlled double integrator system with process and measurement noises is considered with a gain failure. The two possible models are given by the following system equation

\[ x'[k+1] = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x[k] + \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} u[k] + \begin{bmatrix} 1 & 0 & \ldots & 0 \\ 0 & 1 & \ldots & 0 \end{bmatrix} v[k] \]

with measurement equation

\[ z[k] = \begin{bmatrix} 1 & 0 \end{bmatrix} x'[k] + w[k] \] \hspace{1cm} (5.1)

The models differ in the control gain parameter \( b' \). The process and measurement noises are mutually uncorrelated with zero mean and variances given by

\[ \mathbb{E}[v[k] v'[l]] = \begin{bmatrix} 4 \times 10^{-2} & 0 \\ 0 & 4 \times 10^{-2} \end{bmatrix} \]

(5.2)
and
\[ E[\xi(k+1) \xi(k)] = \delta_{ij} \]  \hspace{1cm} (5.4)

The control gain parameters were chosen to be \( b^2 \times 2 \) and \( b^2 \times 3 \). The transition probabilities \( p_{ij} \) and \( p_{ij} \) defined in (3.1) are shown in Fig. 3-1. Note that \( p_{ij} \) for \( i \neq j \) are given by
\[ p_{ij} = 1 - p_{ij} \]  \hspace{1cm} (5.5)

Thus we see that \( p_{ij} \) is initially 5 and rises rapidly to 99 and then decreases towards 1 which is its steady state value. We also see that \( p_{ij} \) has a value close to 10 for this range of \( \tau \) and thus model state 20 is essentially an absorbing state.

Figs. 5-2 through 5-4 present the results of 100 Monte Carlo runs. The true system was initially model 1 for every run and the model transitions occurred according to the probabilities of Fig. 5-1. For simplicity, since we are mainly interested in the estimation of the state, and not in the control strategy, we set \( u(k) = 3 \) for all \( k \).

The Markov based IMM used for comparison utilized the a priori average transition probabilities \( \bar{p}_{ij} \), obtained by taking the expected value of the transition probabilities shown in Fig. 5-1. In other words, the conditional probability \( p_{ij} \) is replaced by the a priori (unconditional) \( \bar{p}_{ij} \) given below in (5.7).

The probability of having a sojourn time \( \tau \) equal to \( \tau \) is the probability that model \( i \) is in effect for \( \tau - 1 \) steps, and then a transition occurs at step \( \tau \).
\[ p(\tau = \tau | \bar{p}_{ij}) = \left( \prod_{i=1}^{\tau-1} p_{ji} \right) \]  \hspace{1cm} (5.6)

Thus we get
\[ \bar{p}_{ij} = \sum_{\tau=1}^{\infty} p(\tau = \tau | \bar{p}_{ij}) \]  \hspace{1cm} (5.7a)

and
\[ \bar{p}_{ij} = 1 - p_{ij} \]  \hspace{1cm} (5.7b)

Figs. 5-2 and 5-3 are plots of the RMS error in \( x_i(k) \) and \( x_2(k) \) respectively. From Fig. 5-2 we can see that the STOM-based IMM estimator improves the RMS error in \( x_i(k) \) by as much as 20 percent. From Fig. 5-3 we see that the RMS error in \( x_2(k) \) of the STOM-based IMM estimator is as low as one third the error of the Markov-based IMM scheme. Thus the mean-square error improved by an order of magnitude.

Fig. 5-4 is a plot of the average model probability error. This is the error in the filter's determination of the correct system model.

Typical running times for the STOM-based IMM vs. the Markov-based IMM are in the ratio of 3:1. The length of the time-span over which the sojourn time pdf is computed can be truncated - it becomes negligible after 15 steps. This keeps within reasonable limits the additional calculations of the STOM-based filter and prevents any growth of the computational or memory requirements.

Example 2
In this example we make a comparison between the detection-estimation algorithm (DEA), based semi-Markov estimator of [M9] with the STOM-based IMM estimator of this paper. For this purpose the system and the semi-Markov model switching process attributes are as in [M9] example 3, and are repeated here for ease of reference.

The model process \( M(k) \) is taken as a semi-Markov chain. The scalar system is described by \[ x(k+1) = 1.04 x(k) + v(k) \]
\[ z(k) = x(k) + (0(k) + w(k)), \]  \hspace{1cm} (5.8)

where \( r = 3 \) models, \( 0(1) = 100, 0(2) = 10, \) and \( 0(3) = 1 \).

Here \( (0(1)) \) and \( (0(2)) \) are mutually independent zero-mean Gaussian white noise sequences with covariances \( Q=0.1 \) and \( R=0.1 \), respectively. The initial conditions are \( (x(0)) = (10, 2, 0), P(M(0)) = 1/3 \) for \( i=1,2,3 \) for the real system \( x(0)=1 \) in every simulation. The process \( M(k) \) is modeled by a semi-Markov chain with the imbedded Markov chain transition probabilities given by \( p_1 = 0.3, p_2 = 0.6, p_3 = 0.4, p_4 = 0.3, \) and \( p_5 = 0.7 \). The sojourn time probability mass functions \( p_{ij} \) are assumed to be
\[ p_{ij} = \exp(-1 - 3), \]  \hspace{1cm} (5.9a)
\[ p_{ij} = \exp(-1 - 6) \]  \hspace{1cm} (5.9b)
and
\[ p_{ij} = \exp(-1 - 8) \]  \hspace{1cm} (5.9c)
for \( r=0 \) with \( a \) such that
\[ \sum_{\tau=1}^{\infty} p_{ij} = 1, i=1,2,3 \]  \hspace{1cm} (5.10)

The results of 50 Monte Carlo runs average are shown in Figs. 5-5, 5-6. In Fig. 5-5 we compare the rms state errors of the two filter DEA based semi-Markov estimator of [M9] with our two filter GPB based semi-Markov approach, and with the GPB estimator using 3 filters. Note that the values for the DEA estimator are two-time-step smoothed values (see [M9], Fig. 7. M+2 most likely histories retained) whereas the values for the STOM-IMM estimator are filtered values. We can see that our estimator with two filters is stable as opposed to the unstable two-filter DEA method.

The plot of the 3 filter STOM-IMM estimator shown in Fig. 5-5 is given so that one can compare the improvement obtainable by adding an extra filter to this approach. We see that the long term trend is for the 3 filter STOM-IMM to give a smaller rms error than the version with 2 filters.

In Fig. 5-6 we compare the probability of error obtained using a 4 filter DEA estimator versus the 3 filter STOM-IMM estimator. Both curves were obtained from a filtering operation (see [M9], Fig. 10, N=0).

We can see that the present estimator gives a much clearer indication of the correct system structure and hence is preferable for failure detection.

Conclusion
We have applied the recursive state estimation algorithm for dynamic systems, whose state model experiences jumps according to a sojourn-time-dependent Markov, STOM, chain, to the problem of failure detection. The algorithm, which is of the IMM type, uses noisy state observations and the calculations are done in the following order:

1. Probability of each model being the current model
2. Sojourn time pmf in the current model
3. Model-conditioned state vector estimates and covariances
4. Overall state vector estimate and its covariance.

The first example simulated indicates that the use of the STOM-based IMM estimator can give a substantial improvement in state estimation over a Markov-based IMM. The latter relies on the a priori average transition probabilities while the former uses conditional transition probabilities obtained from the conditional sojourn time distribution. This example shows that the STOM-based scheme is substantially better than the Markov-based scheme in determining the true system model, which is beneficial for failure detection schemes.

The second example simulated shows that, for the particular system under consideration the STOM-based
IMM estimator, which is an hypothesis merging technique, compares favorably in terms of the probability of error, to the detection-estimation algorithm based estimator, which discards the unlikely parameter history hypothesis.

Acknowledgement

Comments from H.A.P. Blom are gratefully acknowledged.

References


Figure 5-1: Transition probabilities $p_{11}(k)$ and $p_{22}(k)$. $p_{11}(k)$ is shown by the dotted line.

Figure 5-2: RMS error in $x_{2}(k)$. STDM-based IMM and Markov-based IMM. Markov-based IMM errors are shown by the solid line.

Figure 5-3: RMS error in $x_{2}(k)$. STDM-based IMM and Markov-based IMM. Markov-based IMM errors are shown by the solid line.

Figure 5-4: Average model probability error magnitudes. STDM-based IMM and Markov-based IMM. Markov-based IMM errors are shown by the solid line.
Distributed Adaptive Estimation with Probabilistic Data Association*

K. C. CHANG† and Y. BAR-SHALOM‡§

A fusion algorithm for target state estimation under cluttered environment with uncertain measurement origins and uncertain system models in a distributed manner can be applied for tracking a maneuvering target in a cluttered and low detection environment.

Key Words—Distributed estimation; multiple model; target tracking; probabilistic data association; Bayesian methods; distributed sensor networks.

Abstract—The probabilistic data association filter (PDAF) estimates the state of a target in a cluttered environment. This suboptimal Bayesian approach assumes that the exact target and measurement models are known. However, in most practical applications, there are difficulties in obtaining an exact mathematical model of the physical process. In this paper, the problem of estimating target states with uncertain measurement origins and uncertain system models in a distributed manner is considered. First, a scheme is described for local processing, then the fusion algorithm which combines the local processed results into a global one is derived. The algorithm can be applied for tracking a maneuvering target in a cluttered and low detection environment with a distributed sensor network.

I. INTRODUCTION

The major difficulty in tracking a target with switching models/parameters in a cluttered environment is due to the fundamental conflict between the operations of model/parameter identification and data association, since the measurements with large innovations are considered as unlikely to have originated from the target of interest. In this paper, a multiple model approach in conjunction with the probabilistic data association (PDA) filter (Bar-Shalom and Tse, 1975; Bar-Shalom, 1978) to track a target with switching models using distributed sensors, is presented.

Several approaches have been proposed to perform the state estimation of a system together with identification of each model (out of a finite set) in a centralized framework. One of the significant schemes is the so-called generalized pseudo Bayes (GPB) method (Tugnait, 1982; Chang and Athans, 1978) and the other is the interacting multiple model (IMM) algorithm (Blom, 1984; Blom and Bar-Shalom, 1988). The general structure of these algorithms consists of a bank of filters for the state cooperating with a filter for the parameters. A GPB algorithm of order n (GPBn) needs n filters in its bank (Tugnait, 1982). The IMM algorithm performs nearly as well as the GPB2 method with notably less computation, namely, at the cost of GPB1 (Blom and Bar-Shalom, 1988). A distributed estimation scheme with uncertain models has also been derived (Chang and Bar-Shalom, 1987). However, in all the above approaches, a perfect data association was assumed, i.e. there is no uncertainty in measurement origins.

To take into account the data association problem, an adaptive PDA algorithm was presented in Gauvrit (1984) for tracking in a cluttered environment with unknown noise statistics. This algorithm identifies on line the unknown variances of the process and measurement noises but uses an earlier (static) multiple model approach (Bar-Shalom, 1988). In this paper, a distributed estimation problem which takes into account both model and measurement origin uncertainties will be derived. To handle the model uncertainty, a more general formulation with dynamic multiple models described by Markovian parameters will be adopted. These parameters may switch within a finite set of values which represent different system models. To take care of the missing and false
measurements, the PDA scheme will be employed. The probabilities of associating measurements to a target given different system models will be computed and used to weight the combination of state estimates.

The problem is formulated in Section 2. A centralized algorithm which combines the IMM algorithm and the PDA filter, resulting in the MMPDA (multiple model PDA) filter, for local processing will be described in Section 3. Then the fusion algorithm which combines the local processed results from multiple sensors into a global one will be presented in Section 4.

The algorithm can be applied for tracking a maneuvering target in a cluttered and low detection environment with a distributed sensor network (DSN).

2. PROBLEM FORMULATION

Let us consider the two-node scenario similar to that given in Chang et al. (1986), where each node processes the local measurements from its own sensor and sends the local estimates to the fusion processor periodically. The fusion processor then sends back the processed results after each communication time.

The dynamics of the target in track are modeled as

\[ x(k) = f(x(k-1), M(k), v[M(k), k-1]) \]  (1)

where \( x(k) \) is the state vector, \( v[M(k), k-1] \) the process noise vector and \( M(k) \) the system model from time \( k-1 \) to \( k \). Assume the random model process \( M(k) \) is Markov and it can only take values from a finite set \( M \), which contains \( r \) distinct models, i.e.

\[ M = \{ M_i \}_{i=1}^r \].  (2)

The measurement system is modeled as follows. If the measurement originates from the target in track, then

\[ z'(k) = h(x(k), M(k)) + w'[M(k), k] \]  (3)

where \( z'(k) \) is the measurement vector from sensor \( i \) and \( w'[M(k), k] \) is the corresponding measurement noise vector. The two noise sequences are mutually independent and independent of the initial state.

As in the PDA filter, it is assumed that a rule of validation of the candidate measurements is available such that it guarantees that the current return will be retained with a given probability. For each sensor, denote the validated measurements at time \( k \) as

\[ Z^i(k) = \{ z'_i(k) \}_{i=1}^{m_i^k} \]  (4)

where \( m_i^k \) is the number of validated measurements of sensor \( i \) at time \( k \), and

\[ Z^i = \{ Z'(l) \}_{l=1}^{k-1}. \]  (5)

The local model-conditioned state pdfs at sensor \( i \) are

\[ p(x(k) \mid M_i(k), Z^i, Y^i). \]

\[ i = 1, 2; j = 1, \ldots, r \]  (6)

with the corresponding model probabilities

\[ P(M_i(k) \mid Z^i, Y^i), \]

\[ i = 1, 2; j = 1, \ldots, r \]  (7)

where \( Y^i = \{ Y(1), \ldots, Y(r) \} \)

and \( Y(i) \) denotes the information received by node \( i \) during the sampling period ending at time \( k \), which is defined as the fusion result (namely, global conditional pdf) up to time \( k-1 \). Assuming lossless communication and that the information communicated is the sufficient statistics, i.e. the information contained in \( Y^i \) is equivalent to the information in \( Z^i \), then we have the following equality:

\[ p(x(k) \mid Z^i, Y^i) = p(x(k) \mid Z^i, Z^i) \]

\[ = p(x(k) \mid Z^i) \]  (9)

where \( i \) represents all sensors other than sensor \( i \) and \( Z^i = \{ Z(l) \}_{l=1}^{k-1} \), where \( Z(l) \) represents measurements from all sensors at time \( l \).

Given the above models, the question now is how the global conditional pdf can be constructed by fusing together the local ones. Specifically, we shall investigate what is the necessary and sufficient information that has to be transmitted between nodes. The derivations will be carried out for arbitrary pdfs; however, the simulations assume linear models with Gaussian random variables, in which case the state's model-conditioned pdf (6) is Gaussian and the overall conditional pdf of the state is a Gaussian mixture (Bar-Shalom, 1986).

3. CENTRALIZED ALGORITHM FOR LOCAL PROCESSING

For each local node, the centralized algorithm where all measurements are sent to and processed with one processor is described below. The goal is to compute the conditional state...
Distributed adaptive estimation 361

distribution given the local accumulated measurements. With only model uncertainty, the local conditional pdf at sensor \(i\) can be obtained as

\[
p(x(k) \mid Z_{i}^{k}, Y_{i}^{k}) = \sum_{j=1}^{c} p(x(k) \mid M_{j}(k), Z_{i}^{k}, Y_{i}^{k}) \times P(M_{j}(k) \mid Z_{i}^{k}, Y_{i}^{k}).
\]

(10)

When the additional measurement origin uncertainties are present, the above equation becomes

\[
p(x(k) \mid Z_{i}^{k}, Y_{i}^{k}) = \sum_{j=1}^{c} \left( \sum_{i} p(x(k) \mid M_{j}(k), \theta_{i}^{k}, Z_{i}^{k}, Y_{i}^{k}) \times P(\theta_{i}^{k} \mid M_{j}(k), Z_{i}^{k}, Y_{i}^{k}) \right) \times P(M_{j}(k) \mid Z_{i}^{k}, Y_{i}^{k}).
\]

(11)

where \(\theta_{i}^{k}\) is the event that \(z_{i}^{k}(k)\) is the correct measurement and \(\theta_{0}^{k}\) denotes no correct measurement.

The first term on the right-hand side of equation (11) is the standard PDA filter based on model \(M_{j}\), where for each \(\theta_{i}^{k}\):

\[
p(x(k) \mid M_{j}(k), \theta_{i}^{k}, Z_{i}^{k}, Y_{i}^{k}) = \frac{1}{c_{i}[M_{j}(k), \theta_{i}^{k}]} p(Z(k) \mid x(k), M_{j}(k), \theta_{i}^{k}, Z_{i}^{k-1}, Y_{i}^{k-1}) \times p(x(k) \mid M_{j}(k), Z_{i}^{k}, Y_{i}^{k}).
\]

(12)

where \(c_{i}[M_{j}(k), \theta_{i}^{k}]\) has been omitted in the last term above (since it is irrelevant) and

\[
c_{i}[M_{j}(k), \theta_{i}^{k}] = \int_{x} p(Z(k) \mid x(k), M_{j}(k), \theta_{i}^{k}, Z_{i}^{k-1}, Y_{i}^{k-1}) \times p(x(k) \mid M_{j}(k), Z_{i}^{k-1}, Y_{i}^{k-1}) dx(k)
\]

(13)

Using Bayes' rule, the second term on the right-hand side of equation (11) is

\[
P(\theta_{i}^{k} \mid M_{j}(k), Z_{i}^{k}, Y_{i}^{k}) = \frac{1}{c_{i}[M_{j}(k), \theta_{i}^{k}]} p(Z(k) \mid M_{j}(k), \theta_{i}^{k}, Z_{i}^{k-1}, Y_{i}^{k-1}) \times P(\theta_{i}^{k} \mid M_{j}(k), Z_{i}^{k-1}, Y_{i}^{k-1})
\]

(14)

where

\[
c_{i}[M_{j}(k), \theta_{i}^{k}] = \sum_{\theta_{i}^{k}} c_{i}[M_{j}(k), \theta_{i}^{k}] \times P(\theta_{i}^{k} \mid M_{j}(k), Z_{i}^{k-1}, Y_{i}^{k-1})
\]

(15)

In equation (13), the joint measurement density is (see, e.g. Bar-Shalom (1988))

\[
p(Z(k) \mid M_{j}(k), \theta_{i}^{k}, Z_{i}^{k-1}, Y_{i}^{k})
\]

(16)

where \(V_{a}\) is the volume of the validation region, because our assumption on the incorrect measurements being uniformly distributed, independent from each other and from the correct measurement, and

\[
p(z_{i}^{k}(k) \mid M_{j}(k)) = P_{G}^{-1} p(z_{i}^{k}(k) \mid M_{j}(k), \theta_{i}^{k}, Z_{i}^{k-1}, Y_{i}^{k})
\]

(17)

is the truncated density which is zero outside the validation region where \(P_{G}\) is the probability that the correct return will lie in the validation region.

In equation (14), \(P(\theta_{i}^{k} \mid M_{j}(k), Z_{i}^{k-1}, Y_{i}^{k})\) is the prior probability of the event \(\theta_{i}^{k}\) based on model \(M_{j}\) to be correct at time \(k\). By choosing a large enough validation threshold, this probability becomes independent of \(M_{j}(k)\) and is assumed to be the same for all \(\theta_{i}^{k}\) unless target signature information can be used. If no such information is available, then

\[
P(\theta_{i}^{k} \mid M_{j}(k), Z_{i}^{k-1}, Y_{i}^{k}) = \begin{cases} 1 - P_{G} P_{D} & \text{if } i = 0 \\ P_{G} P_{D} & \text{otherwise} \end{cases}
\]

(18)

where \(P_{D}\) is the probability that the correct return will be detected.

For each model \(M_{j}(k)\) and event \(\theta_{i}^{k}\), equation (12) is the standard filtering equation. In that equation, by using the IMM approach (Blom and Bar-Shalom, 1988), the extrapolated pdf is obtained by combining the extrapolations of the

* For more elaborate models see Bar-Shalom (1988).
prior pdfs (independent of the event \( \theta_i^k \))

\[
p(x(k) | M_i(k), Z_i^{k-1}, Y^{i,k}) = \sum_{i=1}^r p(x(k) | M_i(k), M_i(k-1), Z_i^{k-1}, Y^{i,k}) 
\times P(M_i(k-1) | M_i(k), Z_i^{k-1}, Y^{i,k}) 
\times P(M_i(k) | M_i(k-1))
\]

where \( p(x(k) | M_i(k), M_i(k-1), Z_i^{k-1}, Y^{i,k}) \) is the extrapolation of the conditional state pdf given \( Z_i^{k-1} \) and \( Y^{i,k} \) from model \( M_i(k-1) \) to model \( M_i(k) \) and

\[
c_i[M_i(k)] = P(M_i(k) | Z_i^{k-1}, Y^{i,k})
= \sum_{i=1}^r P(M_i(k) | M_i(k-1)) 
\times P(M_i(k-1) | Z_i^{k-1}, Y^{i,k}) 
\times \rho(M_i(k) | Z_i^{k-1}, Y^{i,k}). \tag{19}
\]

The last term of equation (11) is the a posteriori model probability, which is obtained as

\[
P(M_i(k) | Z_i^{k-1}, Y^{i,k}) = \frac{1}{c_4} P(Z_i^{k-1} | M_i(k), Z_i^{k}, Y^{i,k}) 
\times P(M_i(k) | Z_i^{k-1}, Y^{i,k})
\]

\[
= \frac{1}{c_4} c_i[M_i(k)]c_i^r[M_i(k)] \tag{20}
\]

4. FUSION ALGORITHM

With the local conditional pdfs obtained in Section 3, we can now derive the fusion algorithm to obtain global pdf. Similar to equations (10) and (11), the global conditional pdf can be obtained as

\[
p(x(k) | Z^k) = \sum_{i=1}^r p(x(k) | M_i(k), Z^k)P(M_i(k) | Z^k)
\]

\[
= \sum_{i=1}^r \left\{ \sum_{\theta_i^1, \theta_i^2} p(x(k) | M_i(k), \theta_i^1, \theta_i^2, Z^k) \times P(\theta_i^1, \theta_i^2 | M_i(k), Z^k) \right\}P(M_i(k) | Z^k) \tag{23}
\]

Assuming measurements from different sensors are independent given the target state, then the first term on the right-hand side of equation (23) can be obtained as

\[
p(x(k) | M_i(k), \theta_i^1, \theta_i^2, Z^k)
= \frac{1}{c[M_i(k), \theta_i^1, \theta_i^2]} 
\times p(Z(k) | x(k), M_i(k), \theta_i^1, \theta_i^2, Z^{k-1}) 
\times p(x(k) | M_i(k), \theta_i^1, \theta_i^2, Z^{k-1})
\]

\[
= \frac{1}{c[M_i(k), \theta_i^1, \theta_i^2]} 
\times \prod_{i=1}^n [p(Z(k) | x(k), M_i(k), \theta_i^1, Z^{k-1})] 
\times p(x(k) | M_i(k), Z^{k-1})
\]

\[
= \frac{1}{c[M_i(k), \theta_i^1, \theta_i^2]} 
\times \prod_{i=1}^n [p(Z(k) | x(k), M_i(k), \theta_i^1, Z^{k-1})] 
\times \frac{p(x(k) | M_i(k), Z^{k-1})}{p(x(k) | M_i(k), Z^{k-1})} \tag{24}
\]
Distributed adaptive estimation

Assuming \( \theta_1^i \) and \( \theta_2^i \) are independent given the target state, then similarly to Chang et al. (1986), the second term of equation (23) can be obtained as

\[
P(\theta_1^i, \theta_2^i \mid M(k), Z^k)
\]

\[
= \frac{1}{c_1[M(k)]} \int p(\theta_1^i, \theta_2^i, Z(k) \mid x(k), M(k), Z^{k-1}) p(x(k) \mid M(k), Z^{k-1}) \, dx(k)
\]

\[
= \frac{1}{c_1[M(k)]} \prod_{i=1}^{2} \int p(\theta_1^i, Z'(k) \mid M(k), Z^{k-1}) p(x(k) \mid M(k), Z^{k-1}) \, dx(k)
\]

\[
= \frac{1}{c_1[M(k)]} \prod_{i=1}^{2} P(\theta_1^i \mid M(k), Z^k, Z^{k-1})
\]

\[
\times \int p(x(k) \mid M(k), Z^k) \, dx(k)
\]

where

\[
c_1[M(k)] = p(Z(k) \mid M(k), Z^{k-1})
\]

(31)

Equation (24) can be rewritten as

\[
p(x(k) \mid M(k), \theta_1^i, \theta_2^i, Z^k) = \frac{1}{c[M(k), \theta_1^i, \theta_2^i]}
\]

\[
\times \prod_{i=1}^{2} p(x(k) \mid M(k), \theta_1^i, Z^{k-1}, Y^{i,k})
\]

\[
\times p(x(k) \mid M(k), Z^{k-1})
\]

\[
= \frac{1}{c_0[M(k), \theta_1^i, \theta_2^i]}
\]

\[
\times \prod_{i=1}^{2} p(x(k) \mid M(k), \theta_1^i, Z^{k-1}, Y^{i,k})
\]

\[
\times p(x(k) \mid M(k), Z^{k-1})
\]

(27)

Equation (24) can be rewritten as

\[
p(x(k) \mid M(k), \theta_1^i, \theta_2^i, Z^k) = \frac{1}{c[M(k), \theta_1^i, \theta_2^i]}
\]

\[
\times \prod_{i=1}^{2} p(x(k) \mid M(k), \theta_1^i, Z^{k-1}, Y^{i,k})
\]

(26)

where the denominator can be derived as

\[
p(x(k) \mid M(k), Z^{k-1}) = \frac{p(x(k) \mid M(k), Z^{k-1})}{p(M(k) \mid Z^{k-1})}
\]

\[
= \frac{1}{c_0[M(k), \theta_1^i, \theta_2^i]}
\]

\[
\times \prod_{i=1}^{2} p(x(k) \mid M(k), \theta_1^i, Z^{k-1}, Y^{i,k})
\]

(28)

\[
= \frac{1}{c_0[M(k), \theta_1^i, \theta_2^i]}
\]

\[
\times \prod_{i=1}^{2} p(x(k) \mid M(k), \theta_1^i, Z^{k-1}, Y^{i,k})
\]

(29)

and

\[
c_0[M(k), \theta_1^i, \theta_2^i] = \frac{c[M(k), \theta_1^i, \theta_2^i]}{\prod_{i=1}^{2} c_1[M(k), \theta_1^i]}
\]

\[
= \frac{\prod_{i=1}^{2} p(x(k) \mid M(k), \theta_1^i, Z^{k-1}, Y^{i,k})}{p(x(k) \mid M(k), Z^{k-1})} \, dx(k)
\]

(30)

is the new normalization constant.

From equations (27) and (33), equation (23) can
be written as
\[
p(x(k) \mid Z^k) = \sum_{j=1}^{c} \frac{1}{\hat{c}[M_j(k)]} \times \left\{ \prod_{i=1}^{2} \left[ p(x(k) \mid M_i(k), \theta_i, Z^{i,k}, Y^{i,k}) \right] \times P(M_i(k) \mid Z^{i-1}) \right\} \times P(M_j(k) \mid Z^k).
\]
(34)

The last term of equation (34) is the global a posteriori pdf and model probabilities. With equations (31) and (32) we have
\[
P(M_j(k) \mid Z^k) = \frac{\hat{c}}{\hat{c}} c_i[M_j(k)] P(M_j(k) \mid Z^{i-1})
\]
\[
= \frac{\hat{c}}{\hat{c}} \prod_{i=1}^{2} \left[ c_i[M_j(k)] P(M_j(k) \mid Z^{i,k}) \right] P(M_j(k) \mid Z^{k-1}) \times P(M_j(k) \mid Z^k)
\]
(35)

and \(\hat{c}\) are
\[
\hat{c} = p(Z(k) \mid Z^{k-1})
\]
and
\[
\hat{c}' = \frac{\hat{c}}{\prod_{i=1}^{2} p(Z^{i,k}) \prod_{i=1}^{2} c_i}
\]
(37)

4.1. Overview of the fusion algorithm

From the above, it follows that the global a posteriori pdf and model probabilities are obtained by combining (multiplying) the local a posteriori pdfs and model probabilities and removing (dividing) the common a priori pdf and model probabilities. From equation (34), we can see that for each model, the conditional global pdf given that this model is correct is obtained by the sum of global fused pdfs given all possible global event pairs \(\theta_1^i, \theta_2^i\). The overall global a posteriori pdf is then obtained by the sum of global pdfs of each model weighted by the global a posteriori model probabilities. Equations (34) and (35) represent the complete cycle of fusion processing. From them it follows that the information needed to be communicated from local nodes to the fusion node consists of:

- (a) the model probabilities;
- (b) the association event probabilities; and
- (c) the corresponding pdfs (mean and covariance for Gaussian case).

A summary flow diagram of the fusion algorithm with two models is given in Fig. 2. For
5. SIMULATION RESULTS

A two-dimensional single target tracking problem will be considered. Two target dynamic models will be assumed, one with (nearly) constant velocity and the other with (nearly) constant acceleration. The Markov transition matrix of the models is known and given. The initial target state estimate is given and the initial probabilities of the two target models are assumed equal.

The target dynamic models with discretization over time intervals of length $T$ are

$$x(k) = F[M(k)]x(k - 1) + G[M(k)]v(k - 1)$$  \hspace{1cm} (38)

where for model 1, the nearly constant velocity model, the state is

$$x = [x \ x \ y \ y]'$$  \hspace{1cm} (39)

and

$$F = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}$$  \hspace{1cm} (40)

$$G = \begin{bmatrix} T^{2}/2 & 0 \\ T & 0 \\ 0 & T^{2}/2 \\ 0 & T \end{bmatrix}$$  \hspace{1cm} (41)

The process noise $v(k) = [v_x \ v_y]'$ representing the acceleration during one period is a zero mean Gaussian white noise vector with covariance

$$\begin{bmatrix} q_{1x} & 0 \\ 0 & q_{1y} \end{bmatrix}$$  \hspace{1cm} (42)

For models 2 (with acceleration), the state is

$$x = [x \ x \ x \ y \ y \ y]'$$  \hspace{1cm} (43)

and

$$F = \begin{bmatrix} 1 & T & T^{2}/2 & 0 & 0 & 0 \\ 0 & 1 & T & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & T & T^{2}/2 \\ 0 & 0 & 0 & 0 & 1 & T \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$  \hspace{1cm} (44)

where the process noise $v(k)$ representing here the acceleration increment over one period is a zero mean Gaussian white noise vector with covariance

$$\begin{bmatrix} q_{2x} & 0 \\ 0 & q_{2y} \end{bmatrix}$$  \hspace{1cm} (45)

Assuming only position measurements to be available, then, for node $i$

$$z_i'(k) = H'x(k) + w_i(k)$$  \hspace{1cm} (46)

and $w_i(k)$ is a zero mean Gaussian white noise vector with covariance

$$\begin{bmatrix} r_i & 0 \\ 0 & r_i \end{bmatrix}$$  \hspace{1cm} (47)

To overcome the fact that one has different state dimensions the lower dimension vector was augmented with suitable zero components (which then have mean and variance zero) to make it compatible with the higher dimension state.

With sampling interval $T = 1$ s, the true target is simulated with constant velocity for the first seven scans, then switches to constant acceleration for the next seven scans, and finally returns to constant velocity for another seven scans. The initial target state is assumed to be $[100 \text{ m}, 30 \text{ m s}^{-1}, 0, 100 \text{ m}, 15 \text{ m s}^{-1}, 0]$ and the acceleration is assumed to be $5$ and $-5$ m s$^{-2}$ for the $x$ and $y$ coordinates, respectively.

The variances of the process noise are taken as $q_{1x} = q_{1y} = 0.1$ (m s$^{-2}$) for model 1, the nearly constant velocity model, and $q_{2x} = q_{2y} = 1.0$ (m s$^{-2}$) for model 2, the nearly constant acceleration model. The detection probabilities for both sensors are equal to 0.67 and the false alarm rates are 0.0001 m$^{-2}$. The standard deviations of the measurement errors are assumed to be $\sqrt{10} \text{ m}$ for both $x$ and $y$ coordinates of the two sensors. The Markov transition matrix for the model parameters is assumed to be

$$\begin{bmatrix} 0.9 & 0.1 \\ 0.1 & 0.9 \end{bmatrix}$$  \hspace{1cm} (48)

The initial state estimate is generated randomly with mean the same as the true target state and covariance matrix equal to

$$\text{diag} [100, 1, 0.1, 100, 1, 0.1].$$

Three different configurations will be tested. First, each sensor will be simulated independently using the MMPDA algorithm described in Section 3. Second, a centralized processing with
measurements from both sensors will be simulated using the same MMPDA algorithm. Finally, the distributed case will be simulated. In this case, the two nodes will communicate every scan. At each scan, each node will process its own sensor measurements first, then send the local processed results to the fusion node. After receiving the information from both local nodes, the fusion node will use the fusion algorithm derived in the previous section to construct the global estimates and send the results back to each local node.

Simulations were carried out with 50 Monte Carlo runs. The results of one sample run are shown in Figs 3–5. Figures 3 and 4 show the estimated and true trajectories of the target with sensors 1 and 2, respectively. Figure 5 shows the results for the distributed case where the two sensors interchanged their processed results. As can be seen from the figures, in both single sensor cases the algorithm fails to detect clearly the switches of the target between two models. The distributed algorithm not only responds faster in detecting the first jump of the target from the constant velocity mode to the constant acceleration mode, but also successfully detects the end of the acceleration. The centralized algorithm, which is not shown in the figures, performs exactly the same as the distributed one.

The average performances for the three configurations for 50 runs are given in Table 1. The centralized and distributed algorithms successfully track the target in 43 out of 50 runs (“successful tracking” is defined when the estimated target position is within 30 m of the true target position for the last three scans). However, out of 50 runs, sensor 1 alone and sensor 2 alone only track the target successfully in 27 and 30 runs, respectively. The r.m.s. position errors for those successful runs are also calculated. Similarly, the centralized and distributed algorithms perform better than the single sensor configurations. Note that the quality of the estimation using two sensors in terms of mean square error is significantly better than using a single sensor.

The centralized case yields the upper bound of the performance for the distributed configur-
**Fig. 4.** Tracking results with sensor 2 only (one sample run).

**Fig. 5.** Tracking results of distributed case (one sample run).
atation when the nodes communicate every scan.
The simulation shows that the results of
the distributed algorithm are the same as in the
centralized algorithm, which confirms the theo-
retical equivalence.  

6. CONCLUSION

A recursive estimation algorithm that accounts
for the uncertainties of both measurement
origins and system models in a distributed
framework has been derived. The distributed
estimation technique has been adopted together
with the probabilistic data association (PDA)
filter in conjunction with the interactive multiple
model (IMM) scheme. The resulting algorithm
can be applied to track a maneuvering target in a
cluttered environment with distributed sensors.
Simulation results show the expected perform-
ance of the algorithm. With full communication
rate, the distributed case performs exactly the
same as the centralized case, which confirms the
theoretical equivalence, but has the advantages
of increased reliability.

REFERENCES

Bar-Shalom, Y. and E. Tse (1975). Tracking in a cluttered
environment with probabilistic data association.
*Automatica*, 11, 451.


data association tracker, interactive software.

UCLA Extension and University of Maryland Short
Course Notes.

Bar-Shalom, Y. and T. E. Fortmann (1988). *Tracking and


<table>
<thead>
<tr>
<th>Table 1. Simulation results (50 runs)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Decentralized case</strong></td>
</tr>
<tr>
<td>Node 1</td>
</tr>
<tr>
<td>-----------------</td>
</tr>
<tr>
<td>Number of successful tracks</td>
</tr>
<tr>
<td>r.m.s. Position error</td>
</tr>
</tbody>
</table>


Substituting these into (3.10), and using (E-2), we obtain
\[
\lim_{t \to \infty} e(t) = 0. \tag{3.13}
\]

The estimation property (E-3), the uniform boundedness of \( y(t) \) and \( u(t) \), and (2.5) the definition of \( \epsilon^*_k \) imply that
\[
\lim_{t \to \infty} \epsilon(t) = 0. \tag{3.14}
\]

Substituting this into (3.11) and, again, using (E-2) we obtain
\[
\lim_{t \to \infty} e_0(t) = 0. \tag{3.15}
\]

Since \( E(z^{-1}) \) is a stable polynomial, we can establish ii) by substituting (3.13) and (3.14) into (2.12).

**Remark 3.1:** The multirate sampling estimation algorithm in general does not have the property that \( e(t)/(1 + \|e(t - 1)\|^2) \in l_2 \), which is required in the stability proof of conventional adaptive control algorithms. However, we still prove the stability using property (E-3) and the relation \( |e(t_{j+1})| \geq |e(t)| \) for \( t_{j+1} \). The dual controller is more than the cautious controller with a one-step horizon and a new dual controller with a two-step horizon are examined. In many instances, the myopic cautious controller is seen to turn off and converges very slowly. The dual controller modifies the cautious control design by numerator and denominator correction terms which depend upon the sensitivity functions of the expected future cost and avoids the turn-off and slow convergence. Monte-Carlo comparisons based on parametric and nonparametric statistical analysis indicate the superiority of the dual controller over the cautious controller.

**IV. CONCLUSIONS**

In this note, we have developed a multirate sampling adaptive control algorithm which allows a fast sampling rate of feedback control to be used even if the computation of parameter estimate and controller coefficient may take a relatively long period of time.

The key idea to achieve this is to record the plant input and output prior to the currently obtained estimate and use them to compute the coming estimate and controller coefficients. Thus, the computation is not dependent upon the inputs and outputs appearing during the updating process. The closed-loop system is shown to be stable.

**Remark 4.1:**

i) One may further extend the algorithm to consider \( t_j - t_{j-1} > n + m + d - \alpha \). In this case, a relation
\[
e(t_{j} + \alpha + k) \leq C_1 \max_{t_j - k \leq t \leq t_{j+1}} |e(t)| + C_2
\]

\((k < \infty, C_1 > \infty, C_2 > 0)\), can be used, and the algorithm only needs to compute \( e(t) \) for \( t_j - 1 \leq t \leq t_j + \alpha \) but not for every \( t \) in \( t_j - 1 \leq t \leq t_j \).

ii) Instead of the ARMA model, one can use \( b \)-model [8] in the algorithm, which retains the key features of the continuous-time model and allows a wide bandwidth MRAC system to be achieved.

iii) The multirate sampling adaptive control is presented for an indirect MRAC system. However, the method covers a wide class of direct and indirect adaptive control algorithms of certainty equivalence type such as pole-assignment, LQ-optimal, etc.

iv) Various methods developed for improving adaptive control system performance are applicable to the presented multirate sampling adaptive algorithm. These methods include: a) various modifications of parameter estimator for improving convergence rate; b) noise and disturbance filtering techniques; c) robustness techniques with respect to disturbances and unmodeled dynamics, such as deadzone, normalization, etc.; d) internal model principle for deterministic disturbance rejection, etc.

**REFERENCES**


**An Adaptive Dual Controller for a MIMO-ARMA System**

P. MOOKERJEE AND Y. BAR-SHALOM

Abstract—An adaptive dual controller is presented here for a multiinput multioutput ARMA system. The plant has constant but unknown parameters. The cautious controller with a one-step horizon and a new dual controller with a two-step horizon are examined. In many instances, the myopic cautious controller is seen to turn off and converges very slowly. The dual controller modifies the cautious control design by numerator and denominator correction terms which depend upon the sensitivity functions of the expected future cost and avoids the turn-off and slow convergence. Monte-Carlo comparisons based on parametric and nonparametric statistical analysis indicate the superiority of the dual controller over the cautious controller.

I. INTRODUCTION

Multiinput multioutput systems with unknown parameters are encountered in many practical situations, and their control poses a great challenge to the stochastic control theory. It is not possible to obtain an optimal solution for such systems because of the dimensionality involved in the stochastic dynamic programming [6]. In such situations, emphasis is on obtaining a suboptimal solution that incorporates the intrinsic properties of the optimal solution. For stochastic systems, the control has in general a dual effect [1], [11]: it affects the system's state as well as the future state and/or parameter uncertainty. Thus, the dual controller offers significant improvement potential for the control of uncertain linear plants. In multi-stage problems it 'probes' the system to enhance real-time identification of the system's parameters in order to increase the accuracy of the subsequent control decisions and regulates the system at the same time [4], [9].

Two classes of dual controllers exist presently [14]. In the first class [10], [12], [18], the control minimizes a one-step ahead criterion augmented by a second term which penalizes for poor identification. This approach is simple but often requires tuning of some parameters. The second class (developed for SISO systems in [3], [16], [17]) used the stochastic dynamic programming equation and expands the future cost about a nominal trajectory. Using first- and second-order Taylor series expansions of the expected future cost about a nominal trajectory, dual controllers for MIMO static systems are developed in [5] and [14]. A second-order Taylor series expansion of the future expected cost is performed about a nominal trajectory and a dual controller based on a two-step horizon is developed in this note for a MIMO dynamic (ARMA) model. The cautious [14], [16], [18] and the new dual controller are applied to a MIMO-ARMA system. Monte Carlo simulations, based on parametric and nonparametric statistical analysis, indicate that the dual controller...
controller prevents the turn-off phenomenon and slow convergence prevalent with a cautious solution.

Section II gives the problem formulation. The approximate dual controller with a two-step horizon for the MIMO system is derived in Section III. The control solution is obtained by approximating the solution of the stochastic dynamic programming equation. A second-order Taylor series expansion of the expected future cost is performed about a nominal trajectory and this leads to a dual control solution in a closed form. Following the derivations of the controller, a summary of the algorithm is given. Section IV describes the simulation of the plant and compares the performances of the cautious and the dual solutions. Section V concludes the note.

II. PROBLEM FORMULATION

The MIMO system to be controlled is described by

\[ y(k) = -Ay(k-1) + Bu(k-1) + e(k) \]  

where

\[ E[e(k)] = 0; E[e(k) e'(j)] = W \delta_{ij}. \]  

Here \( y(k) \) is the output of the plant, \( u(k) \) is the input to the plant, and \( e(k) \) is the measurement noise.

The parameter matrices \( A \) and \( B \) are unknown. This model describes some industrial processes like an ore crushing plant, or a heat exchanger [1]. The unknown elements of \( A \) and \( B \) comprise the parameter vector \( \theta(k) \) whose estimate at time \( k \) is \( \hat{\theta}(k) \) with covariance matrix \( P(k) \). The parameter vector is designated as

\[ \theta(k) \triangleq [a_1; b_1; a_2; b_2; \cdots; a_n; b_n]' \]  

where \( n \) is the dimension of the output vector \( y(k) \) and \( a_i; b_i \) are the \( i \)th row of the matrices \( A \) and \( B \), respectively. Assuming the parameters are time-invariant, we have

\[ \theta(k+1) = \theta(k). \]  

A measurement matrix \( H(k) \) is defined as

\[ H(k) \triangleq \text{diag}[-y'(k)u'(k), -y'(k)u'(k), \cdots] \]  

where \( H(k) \) has \( n \) rows, and \( y'(k), u'(k) \) are the measurement and control vectors transposed.

With these definitions, the measurement model is

\[ y(k) = H(k-1)\theta(k-1) + e(k). \]  

The performance criterion to be minimized is \( J(0) \), i.e., the conditional expected value of the cost \( C(0) \) from step 0 to \( N \), denoted by

\[ J(0) = E[C(0)|I^1] = E \left[ \sum_{k=0}^{N-1} \{ y(k+1) - y_r \}'Q(k)\{ y(k+1) - y_r \} | I^k \right] \]  

where \( Q(k) \) is the diagonal weighting matrix, \( I^k \) is the cumulated information at time \( k \), and \( y_r \) is the desired output.

III. DUAL CONTROL WITH A TWO-STEP HORIZON

First the controller is derived and then a summary of the algorithm is provided.

A dual control solution with a two-step horizon is obtained by minimizing (2.7) with respect to the control \( u(0) \) for the multidimensional plant (2.1)-(2.4). This is obtained by solving the general equation of stochastic dynamic programming [3], [7], [8].

\[ J^*(k) = \min_{u(k)} E[C(k) + J^*(k+1)|I^k] \quad k = N-1, \cdots, 1, 0 \]  

where \( J^*(k) \) is the optimal expected cost to go from \( k \) to \( N \). \( C(k) \) is the cost to go from \( k \) to \( N \), and \( I^k \) is the cumulated information at time \( k \) when the control \( u(k) \) is to be applied. The information \( I^k \) is the set of all past controls until time \( k-1 \) and outputs until time \( k \). Thus, for a two-step horizon we have

\[ J^*_{k+1,k+2} = \min_{u(k)} E[C(k) + J^*_{k+1,k+2}|I^k] \]

\[ = \min_{u(k)} E[(y(k+1) - y_r)'Q(k)(y(k+1) - y_r) + J^*_{k+1,k+2}|I^k] \]

where \( J^*_{k+1,k+2} \) is the optimal expected cost at the last step with one-step horizon and is obtained by minimization of \( J_{k+1,k+2} \), and \( J_{k+1,k+2} \) is the cost to go from \( k+1 \) to \( k + 2 \).

The cautious control at \( k + 1 \) with one-step horizon is given by

\[ u(k+1) = [E(B'Q(k+1)B'I^{k+1}]^{-1} \cdot E[B'Q(k+1)(Ay(k+1) + y_r)]|I^{k+1} \]. \]  

The cost from step \( k + 1 \) to \( k + 2 \) is

\[ J_{k+1,k+2} = \text{tr} Q(k+1)W \]

\[ + E[(Ay(k+1) + y_r)'Q(k+1)(Ay(k+1) + y_r)]|I^{k+1} \]

\[ - E[(Ay(k+1) + y_r)'Q(k+1)B'I^{k+1}] \]

\[ - E[B'Q(k+1)(Ay(k+1) + y_r)]|I^{k+1} \]

and inserting (3) into (4) the optimal cost at the last step is

\[ J^*_{k+1,k+2} = \text{tr} Q(k+1)W \]

\[ + E[(Ay(k+1) + y_r)'Q(k+1)(Ay(k+1) + y_r)]|I^{k+1} \]

\[ - E[(Ay(k+1) + y_r)'Q(k+1)B'I^{k+1}] \]

\[ - E[B'Q(k+1)(Ay(k+1) + y_r)]|I^{k+1} \]

where \( E[|I^{k+1}] \) is the conditional expectation given the available information \( I^{k+1} \).

The unknown parameters will be chosen from the Gaussian family and thus their estimate \( \hat{\theta}(k+1) \) and associated error covariance \( P(k+1) \) are the sufficient statistic. The parameter vector estimate \( \hat{\theta}(k+1) \) and the associated covariance matrix \( P(k+1) \) are obtained from a Kalman filter according to

\[ K(k+1) = P(k)H'(k)[H(k)P(k)H'(k) + W]^{-1} \]

\[ \hat{\theta}(k+1) = \hat{\theta}(k) + K(k+1)[y(k+1) - H(k)\hat{\theta}(k)] \]

\[ P(k+1) = P(k) - P(k)H'(k)[H(k)P(k)H'(k) + W]^{-1}H(k)P(k) \]. \]  

Here \( \hat{\theta}(k+1) \) is the innovation of the process.

From (5) it is clear that \( J_{k+1,k+2} \) is a nonlinear function of the estimated parameter vector \( \hat{\theta}(k+1) \) and covariance \( P(k+1) \). But the estimated vector \( \hat{\theta}(k+1) \) and the covariance \( P(k+1) \) are not known until the control \( u(k) \) is applied. A control \( u(k) \) with a two-step horizon can be obtained from (2) if a second-order Taylor series expansion of \( J_{k+1,k+2} \) is performed about a suitable nominal trajectory. Here the nominal trajectory is defined by

1) a nominal parameter estimate \( \hat{\theta}(k+1) = \hat{\theta}(k) \)
2) a nominal control \( u(k) \)
3) a nominal covariance \( P(k+1) \) obtained by using \( u(k) \)
4) a nominal measurement \( y(k+1) \) obtained by using \( u(k) \) and \( \hat{\theta}(k) \), i.e., \( y(k+1) = H(k)\hat{\theta}(k) \).
Expansion of (5) about this nominal trajectory results in
\[ J_{k+1, k+2} = J_{k+1} + J_{k+1}(k+1)(y(k+1) - y^*(k+1)) \]
\[ + \frac{1}{2} y(k+1) - y^*(k+1) [J_{k+1}(k+1)(y(k+1) - y^*(k+1)) + J_{k+1}(k+1)[\theta(k+1) - \theta(k)] + \frac{1}{2} \theta(k+1) - \theta(k)]^\top \]
\[ \cdot \sigma(k+1) - \theta(k)] + \frac{1}{2} \] \[ \cdot [J_{k+1}(k+1)P(k+1) - P(k)] \]
(9)
where \( J_i \) is the zeroth-order term and the cost sensitivities are
\[ J_{k+1} \equiv \left[ \frac{\partial J_{k+1}}{\partial y(k+1)} \right] \]
(10)
\[ J_{k+1} \equiv \left[ \frac{\partial J_{k+1}}{\partial y(k+1)} \right] \]
(11)
\[ J_{k+1} \equiv \left[ \frac{\partial J_{k+1}}{\partial \theta(k+1)} \right] \]
(12)
\[ J_{k+1} \equiv \left[ \frac{\partial J_{k+1}}{\partial \theta(k+1)} \right] \]
(13)
\[ J_{k+1} \equiv \left[ \frac{\partial J_{k+1}}{\partial \theta(k+1)} \right] \]
(14)
The above sensitivities are evaluated at \( \theta(k), P(k+1), \) and \( y(k+1); \)
and \( P(k+1) \) is the \( i \)-th element of the covariance matrix associated
with the parameter estimates \( \theta(k+1) \) and \( \theta(k+1). \)
Under the Gaussian assumption for the zero mean noise
\[ y(k+1) - y^*(k+1) \sim N(\mu, \Sigma) \]
(15)
where the conditional mean is
\[ \mu = E[H(k)P(k+1) + e(k+1) - \hat{H}(k)\hat{\theta}(k)] \]
\[ = \{H(k) - \hat{H}(k)\}\hat{\theta}(k) \]
(16)
and the conditional covariance is
\[ \Sigma = E[(y(k+1) - y^*(k+1) - \mu)(y(k+1) - y^*(k+1) - \mu)^\top] \]
\[ = H(k)P(k+1)H^\top(k) + \Sigma. \]
(17)
With the choice of the nominal path as defined earlier and using (6),
(16), and (17), the conditional expected value of (9) is
\[ E[J_{k+1}(k+1) + J_{k+1}(k+1)[H(k) - \hat{H}(k)]\hat{\theta}(k) \]
\[ = \frac{1}{2} \mu \cdot J_{k+1}(k+1)\mu + \frac{1}{2} \text{tr} \{J_{k+1}(k+1)\Sigma\} \]
\[ + \frac{1}{2} \text{tr} \{J_{k+1}(k+1)[P(k+1) - P(k)]\} \]
\[ + \frac{1}{2} \text{tr} \{J_{k+1}(k+1)[P(k+1) - P(k)]\} \]
(18)
The above expected future cost (18) is a function of the nominal
parameters multiplied by appropriate sensitivity functions \( J_{k+1}, \)
\( J_{k+1}(k+1), J_{k+1}(k+1), \) and \( J_{k+1}(k+1). \) These sensitivities introduce
the dual effect into (2) which is then used to yield \( u(k). \) It must also be noted
that the covariance \( P(k+1) \) is nonlinear in \( u(k) \) and is not yet known.
Hence, a second-order expansion of \( P(k+1) \) is proposed about
a nominal control \( \tilde{u}(k) \) and a nominal covariance \( \tilde{P}(k+1) \) in order to
obtain a (suboptimal) dual solution \( u(k) \) in a closed form from (2).
This expansion is performed as follows:
\[ P(k+1) = \tilde{P}(k+1) + \sum_{i=1}^{r} e_i e_i^\top \{ P_{u_i}(k+1)[u(k) - \tilde{u}(k)] \}
+ \frac{1}{2} u(k) - \tilde{u}(k) \}
\[ + \frac{1}{2} \text{tr} \{ J_{k+1}(k+1)[P(k+1) - P(k)]\} \]
(19)
IV. Simulation Results

Performance is evaluated from 500 Monte Carlo runs for the following controllers:
1) heuristic certainty equivalence [3] (with a one-step horizon);
2) one-step ahead cautious controller; and
3) dual controller based upon sensitivity functions (with a two-step horizon) derived in Section III.

The plant equations for a two-input two-output system are

\[ y_1(k+1) = -a_{11}y_1(k) - a_{12}y_2(k) + b_{11}u_1(k) + b_{12}u_2(k) + e_1(k) + 1 \]  
\[ y_2(k+1) = -a_{21}y_1(k) - a_{22}y_2(k) + b_{21}u_1(k) + b_{22}u_2(k) + e_2(k) + 1 \]  

where

\[ E[z(k)z(j)] = W_0 = \text{diag}(W_1, W_2); \]
\[ W_1 = 7.52^2; W_3 = 43^2. \]

The true values of the parameters are

\[ a_{11} = 0.8 \quad b_{11} = -74.84 \]
\[ a_{12} = 0.1 \quad b_{12} = -51.04 \]
\[ a_{21} = 0.2 \quad b_{21} = 53.31 \]
\[ a_{22} = 0.75 \quad b_{22} = -82.56. \]

Only the gain parameters \((B\) matrix) are considered unknown for testing the dual effect and their initial estimates were generated as \(2(\hat{b}_1, \hat{b}_2)\). This choice of system was motivated by the helicopter vibration study [13].

A large initial uncertainty is chosen in the parameter estimates in order to test the learning capabilities of the various adaptive algorithms. The cost weighting matrices are

\[ Q(k) = \text{diag}(q_1, q_2); \quad q_1 = 1.0, q_2 = 1.0. \]

The desired response is
\[ y = [-18 \ 80]^T. \]

For the model chosen (1)-(6) the optimal control solution in order to reach a steady-state value of \(y\) in (6) is

\[ u_1^* = 1.0, u_2^* = -1.0. \]

In terms of the notation of (1) and (2)

\[ \theta(k) \triangleq [a_{11} \ a_{12} \ b_{11}(k) \ b_{12}(k) \ a_{21} \ a_{22} \ b_{21}(k) \ b_{22}(k)]^T \]

and

\[ H(k) \triangleq \begin{bmatrix} -y_1(k) & -y_2(k) & u_1(k) & u_2(k) & 0 & 0 & 0 & 0 & -y_1(k) & -y_2(k) & u_1(k) & u_2(k) \end{bmatrix}. \]

The controllers are implemented with a sliding horizon for a total of 40 time steps. The evaluation criterion is

\[ C_y = \sum_{k=1}^{40} \frac{1}{J} \| y(k+1) - y \|_2^2, \]

where

\[ C_y = \sum_{k=1}^{40} \frac{1}{J} \| y(k+1) - y \|_2^2, \]

A. Analysis of the Monte Carlo Average Costs

Comparisons are made between the performances of the cautious and the dual algorithm on the system and a statistical significance analysis is done using the normal theory approach (i.e., it is assumed that the central limit theorem holds for the sample mean from a large number of runs) [14]. Tables I-IV contain the results of the simulation runs. Table I compares the average cost \(C_y\) over 500 Monte Carlo runs for the first 40 time steps for HCE, cautious, and the dual algorithms, with a control limiter \(|u| \leq 2, i = 1, 2\).

Clearly it is seen that the cumulative average cost is the lowest for the dual controller. The HCE incurs an excessive penalty in time step 1) because of lack of caution. The cautious controller is overly cautious and exhibits slow convergence. However, the dual controller incurs less penalty in time step 1) than the HCE and makes a judicious choice of caution and probing to learn the parameters fast.

Table II provides a statistical significance test and shows the improved performances of the dual solution from time step 40 onwards with at least 98 percent confidence.

Table III indicates the percentage of runs where the cost exceeds 2000 for the two algorithms. This threshold of 2000 is selected from a sample distribution study of the cost at each time step. Table IV shows the percentile test [14], [15] comparing the cautious and the dual solution. They clearly indicate from time step 40 onwards the light tailed nature of the distribution of the cost yielded by the new dual control algorithm.

B. Individual Time History Runs

Analysis of the Monte Carlo average cost indicates the improvement offered by the dual solution; it provides no information about the cautious control's turning-off phenomenon [16], [18]. Hence, a careful investigation of the individual runs is required to examine these occurrences.
TABLE III
COMPARISON OF THE TAILS USING THE CAUTIOUS AND THE DUAL
ALGORITHMS IN THE SIMULATION WITH A LIMITER
(|u_l| ≤ 2.0, |u_u| ≤ 2.0) (500 MONTE CARLO RUNS)

<table>
<thead>
<tr>
<th>Time Step k</th>
<th>Cautious Percentage of runs which exceed 2000</th>
<th>Dual</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>64</td>
<td>72</td>
</tr>
<tr>
<td>2</td>
<td>60</td>
<td>32</td>
</tr>
<tr>
<td>3</td>
<td>43</td>
<td>40</td>
</tr>
<tr>
<td>4</td>
<td>33</td>
<td>25</td>
</tr>
<tr>
<td>5</td>
<td>31</td>
<td>17</td>
</tr>
<tr>
<td>6</td>
<td>22</td>
<td>10</td>
</tr>
<tr>
<td>7</td>
<td>22</td>
<td>8</td>
</tr>
<tr>
<td>8</td>
<td>19</td>
<td>7</td>
</tr>
<tr>
<td>9</td>
<td>16</td>
<td>5</td>
</tr>
<tr>
<td>10</td>
<td>12</td>
<td>2</td>
</tr>
<tr>
<td>11</td>
<td>12</td>
<td>1.2</td>
</tr>
<tr>
<td>12</td>
<td>10</td>
<td>1.4</td>
</tr>
<tr>
<td>13</td>
<td>11</td>
<td>1.4</td>
</tr>
<tr>
<td>14</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>15</td>
<td>8</td>
<td>0.4</td>
</tr>
<tr>
<td>16</td>
<td>6</td>
<td>0.4</td>
</tr>
<tr>
<td>17</td>
<td>6</td>
<td>0.2</td>
</tr>
<tr>
<td>18</td>
<td>6</td>
<td>0.4</td>
</tr>
<tr>
<td>19</td>
<td>3</td>
<td>0.4</td>
</tr>
<tr>
<td>20</td>
<td>5</td>
<td>0.2</td>
</tr>
</tbody>
</table>

TABLE IV
PERCENTILE TEST FOR COMPARISONS OF THE CAUTIOUS AND THE DUAL
ALGORITHMS IN THE SIMULATION WITH A LIMITER
(|u_l| ≤ 2.0, |u_u| ≤ 2.0) (500 MONTE CARLO RUNS)

<table>
<thead>
<tr>
<th>Time Step k</th>
<th>( \chi^2 ) test statistics at ( K_{0.9} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>--</td>
</tr>
<tr>
<td>2</td>
<td>--</td>
</tr>
<tr>
<td>3</td>
<td>--</td>
</tr>
<tr>
<td>4</td>
<td>--</td>
</tr>
<tr>
<td>5</td>
<td>10</td>
</tr>
<tr>
<td>6</td>
<td>19</td>
</tr>
<tr>
<td>7</td>
<td>23</td>
</tr>
<tr>
<td>8</td>
<td>33</td>
</tr>
<tr>
<td>9</td>
<td>35</td>
</tr>
<tr>
<td>10</td>
<td>37</td>
</tr>
<tr>
<td>11</td>
<td>40</td>
</tr>
<tr>
<td>12</td>
<td>40</td>
</tr>
<tr>
<td>13</td>
<td>40</td>
</tr>
<tr>
<td>14</td>
<td>16</td>
</tr>
<tr>
<td>15</td>
<td>32</td>
</tr>
<tr>
<td>16</td>
<td>11</td>
</tr>
<tr>
<td>17</td>
<td>16</td>
</tr>
<tr>
<td>18</td>
<td>14</td>
</tr>
<tr>
<td>19</td>
<td>15</td>
</tr>
<tr>
<td>20</td>
<td>25</td>
</tr>
</tbody>
</table>

Fig. 1. Time history of the average cost using the heuristic certainty equivalence, cautious, and the dual controllers. (500 Monte Carlo runs; |u_l| ≤ 2.0, |u_u| ≤ 2.0.) The superior rate of adaptation of the dual algorithm is demonstrated here.

Fig. 2. Time history of output 1 using the cautious and the dual algorithms for run 90 (500 Monte Carlo runs; |u_l| ≤ 2.0, |u_u| ≤ 2.0).

Fig. 3. Time history of output 2 using the cautious and the dual algorithms for run 90 (500 Monte Carlo runs; |u_l| ≤ 2.0, |u_u| ≤ 2.0).

Fig. 4. Time history of control 1 using the cautious and the dual algorithms for run 90 (500 Monte Carlo runs; |u_l| ≤ 2.0, |u_u| ≤ 2.0).
The turn-off phenomenon is observed in many runs among the 500 Monte Carlo simulations while using the cautious controller; run 90 is a typical example of it. Both components are almost off between time steps 0 and 20 during which the dual controller already identified the parameters and reached the desired trajectory. Figs. 2-5 portray this result.

V. Conclusions

A new adaptive dual control solution with a two-step sliding horizon has been developed for an ARMA-MIMO system. The control law is derived by solving the stochastic dynamic programming equation. This solution utilizes the dual effect by performing a second-order Taylor series expansion of the expected future cost and does not need any tuning for any of the runs in the example. It modifies the cautious solution by explicit numerator and denominator correction terms. The controller in its present form is the first of its kind in a closed form for a system with unknown parameters. The controller is tested on a MIMO system in a systematic Monte Carlo fashion. Conclusions are based on 500 Monte Carlo runs. Analysis of the simulation runs has shown that this new dual control solution applied to a multinput multoutput model improves over the cautious controller. The key improvement is in the avoiding of situations like turn-off and slow convergences, typical of the cautious solution.

REFERENCES

TIME-REVERSION OF A HYBRID STATE STOCHASTIC DIFFERENCE SYSTEM
Hank A.P. Blov
National Aerospace Lab. NLK
Amsterdam, The Netherlands

Yaskov Bar-Shalom
University of Connecticut
Storrs, USA

ABSTRACT
The reversion in time of a stochastic difference equation in a hybrid space, with a Markovian solution, is presented. The reversion is obtained by a martingale approach, which previously led to reverse time forms for stochastic equations with Gauss-Markov or diffusion solutions. The reverse time equation is derived from a particular non-canonical martingale decomposition, while the reverse time equations for Gauss-Markov and diffusion solutions followed from the canonical martingale decomposition. The need for this non-canonical decomposition stems from the hybrid state space situation. Moreover, the non-Gaussian discrete time situation leads to reverse time equations that incorporate a Bayesian estimation step.

1. INTRODUCTION
This paper addresses the problem of time-reversion of a hybrid state Markov process which is given as the solution of a stochastic difference equation. The desired result is a similar equation but running in reverse-time direction while having a solution that is respectively pathwise and in probability law equivalent to the solution of the forward equation.

The motivation to study this problem stems from two different kinds of application. The first is to approach the solution of a nonlinear smoothing problem by a merging of the estimates of two nonlinear filters: one filter matches the original model and is applied in the usual time direction while the other filter matches the time-reversed model and is applied in the reverse-time direction. The second application is the determination of a rate distortion theory lower bound for a discrete-time nonlinear filtering problem by the method of Gallgo. This method is based on Bucy's representation formula and requires a Monte Carlo simulation in reverse-time direction of model matching trajectories, starting from a prespecified end point (Gallgo, 1981; Vashburn et al., 1983). For both of these applications it is necessary to have a time-reversed difference equation for which the Markovian solutions are in probability law equivalent to the original solution.

Our problem falls in the category of how to reverse a Markov process in time. The Markov property implies that the past and the future are independent under the condition that the present state is known (Mantzafl, 1981). This invariance with respect to the time direction is the key property used in time-reversion studies. There are two types of studies that deal with this problem; a classical type and a systems-type. The classical type of study assumes that the transition measure or the generator of a Markov process is given and then tries to characterize the transition measure in reverse-time direction (Maggi, 1964; Kunita and Watanabe, 1966; Chung and Walsh, 1969; Akma, 1973; Hasuga, 1976; Dynkin, 1978; Williams, 1979). The systems-type of study assumes that a stochastic equation with a Markovian solution is given for which it tries to characterize the time-reversed equation. The first time-reversed equations were obtained by orthogonality arguments, for the linear Gaussian situation (Ljung and Kailath, 1976; Lainiotis, 1976). For general diffusions, it has already been pointed out by Stratonovich (1960) how to obtain the reversed-time equations by actually following the classical approach: from a stochastic equation via the generator and the time-reversed generator back to time-reversed equations. A truly systems-type of study has been started by Verghese and Kailath (1979), by showing how for a linear Gaussian system a more direct martingale approach leads in a simpler way to time-reversed equations. Moreover, by this approach it was possible to obtain a reversed-time equation with a pathwise equivalent solution. Early elaborations of these ideas led, along different routes, to time-reversed equations with pathwise equivalent solutions (Anderson, 1982; Castanon, 1982; Pardoux, 1983). During subsequent studies, quite large classes of stochastic differential equations and their reversed-time equations have been identified (Elliott and Anderson, 1985; Pardoux, 1985; Elliott, 1986a, 1986b; Haussmann and Pardoux, 1986; Pardoux, 1986). Recently these results have been extended by using the Girsanov transformation of Brownian motion (Picard, 1986; Protter, 1987). Obviously, this Girsanov approach can not be applied to discontinuous or discrete-time processes.

To give an idea of why there is an additional problem in using a martingale approach to the reverse of an equation with a discontinuous solution, we give a brief outline of the approach. The martingale approach roughly consists of checking if the time-reversed driving noise
sequence can be decomposed in a suitable reverse-time martingale part and its complement and next, if such a decomposition exists (Jacod and Shiryaev, 1987; Jacob and Protter, 1988), selecting such a decomposition. The final step is to characterize both the martingale part and its complement. In contrast with a continuous process such a decomposition is not unique for a discontinuous process (see for example, Jacod and Shiryaev, 1987). This makes the selection of a suitable martingale decomposition far from trivial in the hybrid state space situation, because a less costly choice yields unnecessarily complicated reverse-time equations. This complication is presently unsolved, neither in continuous-time nor in discrete-time. It will be solved in the sequel for quite general difference equations in a hybrid space. With that result we subsequently reverse the considered difference equation in time.

The paper is organized as follows. In section 2 we define the hybrid state stochastic difference equation that will be considered and shortly compare its time-reversion with the time-reversion of a linear Gaussian equation. In section 3 we specify the time-reversion requirements. Next, in sections 4 and 5 we consider, respectively, the pathwise time-reversion and the in probability law equivalent time-reversion. In section 6 we discuss the results obtained.

2. THE STOCHASTIC DIFFERENCE EQUATION CONSIDERED

The stochastic difference equation we consider in the sequel is the following system, on an appropriate stochastic basis and a discrete time interval \([0,T]\) \(\times\) \(\{\omega\}\),

\[
X_{t+1} = a'(t_{t+1}, X_t, X_{t-1}, Y_t),
\]

\[
\theta_{t+1} = b'(t_{t+1}, Y_t),
\]

\[
Y_t = C(t_{t}, X_t, Y_t),
\]

where \((X_{t-1})\) \((\theta_{t-1})\) and \((Y_{t-1})\) are i.i.d. standard Gaussians distributed with mean \(0\) and \(1\) respectively, the initial distribution of \((X_0, \theta_0)\) has the density mass function \(p_{\theta_0} \theta_0\) and \((Y_0, Y_{-1})\) is independent of \((X_0, \theta_0)\). Further \(X_t, \theta_t\) and \(Y_t\) have respectively \(\mathbb{R}^n, \mathbb{R}^n\) and \(\mathbb{R}^m\) valued realizations (with \(m\) a countable set), while \(a, b\) and \(c\) are measurable mappings of appropriate dimensions such that system \(1\) has a unique solution for each initial \((X_0, \theta_0)\) with \(p_{\theta_0} \theta_0\). The mappings \(a, b\) and \(c\) are time-invariant for notational simplicity only.

The second order dependence of \((a)\) on \((\theta_t)\) is quite uncommon (Blom, 1985). Obviously, \((a)\) reduces to the more common situation of first order dependence, only if \(a(t_{t+1}, X_{t+1}, Y_{t+1})\) is invariant w.r.t. either \(\theta_t\) or \(\theta_{t-1}\). The interpretation of \((a)\) as an equation with a second order dependence on \((\theta_t)\) suggests the substitution of \(\theta_{t+1} \theta_{t+1}\) in \((a)\). On doing this \((a)\) reduces to the more common equation, and it follows immediately that \((\theta_t)\) and \((\theta_{t-1})\) are Markov processes. However, as the state space of \((\theta_t)\) is significantly larger than the state space of \((\theta_{t+1})\), this is a rather brute force transformation of \((a)\). A more elegant transformation of \((a)\) to the more common equation consists of substituting \((b)\) in \((a)\), which yields an equation of the following form,

\[
X_{t+1} = a'(t_{t+1}, X_t, X_{t-1}, Y_t),
\]

Instead of a state space expansion, there appears an additional noise term, \(Y_t\). From the latter representation, it follows immediately that the processes \((X_t, X_{t+1})\) and \((\theta_t)\) are Markov processes. The latter transformation clearly shows that \((a)\) is indeed more general than the more commonly studied equation with first order dependence of \((\theta_t)\). With the study of this more general equation, we also anticipate the time-reversion results obtained. In the sequel it will turn out that a reverse-time equation of \((a)\) has, in general, a second order dependence on the time-reversed \((\theta_t)\), even when \(a(t_{t+1}, X_{t+1}, Y_{t+1})\) is \(\theta_t\)-invariant. In view of this, it is natural to study the above more general form.

In the sequel we consider the time-reversion of system \((1)\) under the following assumptions:

\[
A_1
\]

\[
\theta(t_{t+1}, X_t, X_{t-1}, \theta_{t+1}) \text{ has an inverse } a^*:X^\times X^\times X^\times \mathbb{R}^m, \text{ such that for any } (a, b, c) \in \mathbb{R}^n, \mathbb{R}^n, \mathbb{R}^m
\]

\[
a^*(a, b, c)=a(b^*, c) = \text{ all } \mathbb{R}^n, \mathbb{R}^n, \mathbb{R}^m.
\]

Assumptions \(A_1\) and \(A_2\) suggest to transform \((a)\) to the following time-reversed model,

\[
X_t = a^*(t_{t+1}, X_{t+1}, X_{t-1}, Y_t),
\]

\[
\theta_t = b^*(t_{t+1}, Y_t),
\]

\[
Y_t = C(t_{t}, X_t, Y_t),
\]

Because \((Y_t, Y_{t-1})\) and the future \((= \text{ reverse-time past})\) \(\theta_{t+1} = \theta(t_{t+1}, X_t, X_{t-1}, \theta_{t+1})\) are dependent, this is not the time-reversed equation we should look for. Unfortunately, it is not clear how to continue from here. To develop some insight, we take a quick look at the time-reversion of a linear Gaussian system.

Linear Gaussian example

Consider the following linear Gaussian system

\[
X_{t+1} = A X_t + B \theta_t,
\]

Assumption \(A_1\) implies that \(A\) is invertible, \(b\), which

\[
X_t = A^{-1}(X_{t+1} - B \theta_t).
\]

Obviously \(\theta_t\) and the future \(\theta_{t+1}\) are dependent, which requires a martingale decomposition of \(\theta_t\). In this linear Gaussian case the canonical martingale decomposition is the appropriate one. It consists of decomposing \(\theta_t\) in its reverse-time predictable part, \(E(\theta_t|\theta_{t+1})\), and its complement \(\theta^*\):

\[
\theta_t = E(\theta_t|\theta_{t+1}) + \theta^*.
\]

The problem is now to write the predictable part as a function of \(\theta_{t+1}\) (if possible) and to
characterize the covariance of $v'_t$. As pointed out by Verghese and Kailath (1979) it follows readily from orthogonality arguments that
$$E[v'_t|x_{t+1}] = E[v'_t|x_{t+1}],$$
while the fundamental formula for LLSE estimation yields
$$E[v'_t|x_{t+1}] = E[v'_t|x_{t+1}],$$
where $R(t+1)$ is the covariance of $x_{t+1}$.

By a straightforward substitution of these results we obtain
$$x_t = A^{-1} \{ x_{t+1} - B B^T R^{-1}(t+1)x_{t+1} - B v'_t \},$$
which yields the desired reverse-time system:
$$x_t = A^{-1} \{ x_{t+1} - B B^T R^{-1}(t+1)x_{t+1} - B v'_t \}.$$  

The orthogonality arguments and the LLSE estimation step, used in the above procedure, prevent a straightforward extension of that procedure to equation (1). In the sequel we replace the orthogonality arguments and the LLSE estimation step respectively by Markov duality arguments and a Bayesian estimation step. Besides this, we have to select an appropriate martingale decomposition. Following the linear Gaussian case, the canonical martingale decomposition seems a good candidate:
$$(v'_t,v'_t) = (v'_t,v'_t) + E(v'_t,v'_t|x_{t+1}).$$
Unfortunately, this decomposition leads to very complicated elaborations of the Bayesian estimation step. To avoid these complications, we use in this paper the following decomposition:
$$v'_t = E(v'_t|x_{t+1}) = E'_t,$$
where we have to select an appropriate martingale decomposition and to elaborate on the Bayesian estimation step, respectively for equation (1).

We want to obtain a time-reversed version of system (1), such that its solution, $(\hat{y}_t, \hat{R}_t, \hat{S}_t)$, is in some sense equivalent to $(y_t, R_t, S_t)$. To make this objective explicit it needs both a specification of what we mean by a time-reversion of (1), and a specification of the desired sense of process equivalence.

By a reverse-time system we mean a stochastic difference equation which starts at time $T$ and runs in negative time direction on the interval $[0,T]$. We require from a time-reversion of system (1) that it does not change the state space and that the solution of the resulting reverse-time system represents the process $(\hat{R}_t, \hat{R}_t, \hat{S}_t)$. More specifically, $(\hat{R}_t, \hat{R}_t, \hat{S}_t)$ must be the solution of the following system of stochastic difference equations, all $t \in [0,T]$:  
$$R_t = R(t, \hat{R}_{t+1}, \hat{R}_t, \hat{S}_{t+1}, \hat{S}_t),$$
$$S_t = S(t, \hat{R}_{t+1}, \hat{R}_t, \hat{S}_{t+1}, \hat{S}_t),$$
$$y_t = \hat{y}(t, \hat{R}_{t+1}, \hat{R}_t, \hat{S}_{t+1}, \hat{S}_t).$$

where $A$, $B$ and $C$ are deterministic mappings of appropriate dimensions and $(\hat{R}_t, \hat{S}_t)$ is a noise sequence to be specified. For a better understanding of (4) notice that the substitutions in (4.a) in (4.c) and of (4.b) in (4.a,c) transform (4) to a reverse-time system of the more common form:
$$\hat{y}_t = \hat{y}(t, \hat{R}_{t+1}, \hat{R}_t, \hat{S}_{t+1}, \hat{S}_t),$$
$$\hat{R}_t = R(t, \hat{R}_{t+1}, \hat{R}_t, \hat{S}_{t+1}, \hat{S}_t),$$
$$\hat{S}_t = \hat{S}(t, \hat{R}_{t+1}, \hat{R}_t, \hat{S}_{t+1}, \hat{S}_t).$$

To be a useful reverse-time system, $(\hat{R}_t, \hat{S}_t)$ should, as much as possible, be independent of the future (= reverse-time past) information field
$$\hat{y}_{t+1} = \hat{y}(t, \hat{R}_{t+1}, \hat{R}_t, \hat{S}_{t+1}, \hat{S}_t).$$

A minimal requirement is then, that the conditional expectation of $(\hat{R}_t, \hat{S}_t)$, given $\hat{y}_{t+1}$, should be zero. Because $\hat{y}_t$ is a decreasing sequence of sigma algebras, the latter can most easily be put in martingale language (see Elliott, 1982; Kumar and Varaiya, 1979; and the definitions below):

$(\hat{R}_t, \hat{S}_t)$ in (4) should be a reverse-time Martingale Difference sequence w.r.t. $\hat{y}_t$.

1. Definition  
Assume $(\hat{y}_t; \hat{y}(t; \hat{y}_{t+1}))$ is an increasing sequence of information fields, i.e. $\hat{y}_{t+1} \subseteq \hat{y}_t$, any $t \in [0,T]$. 
A random sequence $(\hat{t}_t)$ is said to be a Martingale Difference sequence w.r.t. $\hat{y}_t$ iff for all $t \in [0,T]$, 
(i) $\hat{t}_t$ is $\hat{y}_t$-measurable, 
(ii) $E[|\hat{t}_t|]<\infty$, 
(iii) $E[\hat{t}_t; \hat{y}_{t+1}]=0$ a.s.; for all $t \in [0,T-1]$.  

2. Definition  
Assume $(\hat{y}_t; \hat{y}(t; \hat{y}_{t+1}))$ is a decreasing sequence of information fields, i.e. $\hat{y}_{t+1} \subseteq \hat{y}_t$, any $t \in [0,T]$. 
A random sequence $(\hat{t}_t)$ is said to be a reverse-time Martingale Difference sequence w.r.t. $\hat{y}_t$ iff for all $t \in [0,T]$, 
(i) $\hat{t}_t$ is $\hat{y}_t$-measurable, 
(ii) $E[|\hat{t}_t|]<\infty$, 
(iii) $E[\hat{t}_t; \hat{y}_{t+1}]=0$ a.s.; for all $t \in [0,T-1]$. 

Having specified the desired type of reverse-time system, the next step is to specify the types of equivalence of solutions of systems (1) and (4), in which we are interested. For stochastic processes several useful types of equivalence have been defined and named in the past. We restrict ourselves to the two most important types of equivalence and their unambiguous names (Elliot, 1982; Jacod and Shiryaev, 1987):
- indistinguishable, 
- equivalent in law. 
Definitions are given below.
1 Definition

Two processes \((\xi_t, \eta_t)\) and \((\xi'_t, \eta'_t)\), \(t \in [0,T]\), are said to be indistinguishable if they are defined on the same probability space \((\Omega, \mathcal{F}, P)\) and

\[ P(\xi_t = \xi'_t, \forall t \in [0,T]) = 1. \]  \hspace{1cm} (5)

4 Definition

Two processes \((\xi_t, \eta_t)\) and \((\xi'_t, \eta'_t)\), \(t \in [0,T]\), are said to be equivalent in law if they have the same state space, \(\Omega\), and for all \(0 \leq t_1 < \ldots < t_k \leq T\)

\[ P(\xi(t_1), \ldots, \xi(t_k)) = P(\eta(t_1), \ldots, \eta(t_k)) \]  \hspace{1cm} (6)

For discrete-time processes (5) is satisfied if and only if, for all \(t \in [0,T]\), \(t_{t+1} = t_t\) almost surely. Our objective in the sequel is to obtain time-reversed systems of type (4), with solutions that are respectively indistinguishable and equivalent in law w.r.t. the solution of (1).

4 INDISTINGUISHABLE TIME-REVERSION

In this section we derive a type (4) version of system (1), such that their solutions, \((\xi_t, \xi'_t, \eta_t)\) and \((\xi'_t, \xi'_t, \eta'_t)\), are indistinguishable, and illustrate these results for a jump-linear example.

The first step of our derivation consists of a substitution of (2) and (3) in (1), to arrive at the in section 2 discussed time-reversed system,

\[ \xi_t = \xi_t^{\bullet}(\theta_t, \xi_{t+1}^{\bullet}, \xi_{t+1}^{\bullet}), \]  \hspace{1cm} (7.a)
\[ \eta_t = \eta_t^{\bullet}(\theta_t, \xi_{t+1}^{\bullet}, \eta_{t+1}^{\bullet}), \]  \hspace{1cm} (7.b)
\[ \gamma_t = \gamma_{t}(\theta_t, \xi_{t+1}^{\bullet}, \eta_{t+1}^{\bullet}). \]  \hspace{1cm} (7.c)

Although (7) and (4) look similar, one requirement is not met: the driving noise in (7) is not a reverse-time Martingale Difference sequence w.r.t. the future information field

\[ \gamma_t = \gamma_t^{\bullet}(\theta_t, \xi_{t+1}^{\bullet}, \eta_{t+1}^{\bullet}, \xi_{t+1}^{\bullet}, \eta_{t+1}^{\bullet}); \]  \hspace{1cm} (8)

Therefore our next step is to introduce a particular reverse-time Martingale Difference sequence, \((\xi_t^{\bullet}, \eta_t^{\bullet})\), as follows,

\[ \xi_t^{\bullet} = (\xi_t^{\bullet}, \xi_{t+1}^{\bullet}), \eta_t^{\bullet} = (\eta_t^{\bullet}, \eta_{t+1}^{\bullet}). \]  \hspace{1cm} (9.a)

with

\[ \xi_t^{\bullet} = E(\xi_t | \mathcal{F}_{t-1}), \]  \hspace{1cm} (9.b)
\[ \eta_t^{\bullet} = E(\eta_t | \mathcal{F}_{t-1}); \]  \hspace{1cm} (9.c)

and \((\xi_t^{\bullet}, \eta_t^{\bullet}) = 0 \) almost surely.

Notice that the definition of \(\xi_t^{\bullet}\) differs significantly from the reverse-time predictable process \(E(\xi_t | \mathcal{F}_{t-1})\). As such the decomposition in (9) is not the unique canonical decomposition (see Jacod and Shiryaev, 1987). The introduction of this non-canonical decomposition is a crucial step necessary for obtaining the time-reversion of hybrid state system (1).

In the sequel we verify that \((\xi_t^{\bullet}, \eta_t^{\bullet})\) is indeed a reverse-time Martingale Difference sequence w.r.t. \(\xi_t\), and thus also w.r.t. \(\xi_t^{\bullet} \in \xi_t^{\bullet} \cup \xi_t^{\bullet} \cup (\xi_t^{\bullet}, \eta_t^{\bullet}); \mathcal{F}_t \cup \mathcal{F}_t \cup \mathcal{F}_t \). Moreover we show that, due to the duality of the Markov property, \((\xi_t^{\bullet}, \eta_t^{\bullet})\) is conditionally independent of \(\xi_{t+2}\) (given \((\xi_t^{\bullet}, \eta_t^{\bullet})\)).

5 Theorem

Assume \((\xi_t, \eta_t, \gamma_t)\), \((\xi_t^{\bullet}, \eta_t^{\bullet})\) and \((\xi_t^{\bullet}, \eta_t^{\bullet})\) satisfy (1) and (9). Then \((\xi_t^{\bullet}, \eta_t^{\bullet})\) is a reverse-time Markingale Difference sequence w.r.t. \(\xi_t^{\bullet}\), while \(\bar{\xi}_t^{\bullet}\) and \(\bar{\eta}_t^{\bullet}\) satisfy:

\[ \bar{\xi}_t^{\bullet} = E(\xi_{t+1}^{\bullet} | \mathcal{F}_{t+1}), \]  \hspace{1cm} (10.a)
\[ \bar{\eta}_t^{\bullet} = E(\eta_{t+1}^{\bullet} | \mathcal{F}_{t+1}); \]  \hspace{1cm} (10.b)

Proof: See Blom and Bar-Shalom (1989).

Theorem 5 implies that \(\bar{\xi}_t^{\bullet}\) and \(\bar{\eta}_t^{\bullet}\) can be written as

\[ \bar{\xi}_t^{\bullet} = \xi_t^{\bullet} - \xi_t^{\bullet} \cdot \xi_{t+1}^{\bullet} \gamma_{t+1}^{\bullet}, \]  \hspace{1cm} (11.a)
\[ \bar{\eta}_t^{\bullet} = \bar{\eta}_t^{\bullet} - \bar{\eta}_t^{\bullet} \cdot \xi_{t+1}^{\bullet} \gamma_{t+1}^{\bullet}. \]  \hspace{1cm} (11.b)

Substitution of (9.a) and (11.a,b) in (7,a,b,c) yields

\[ \xi_t = \xi_t^{\bullet} \cdot \xi_{t+1}^{\bullet} \gamma_{t+1}^{\bullet} \]  \hspace{1cm} (12.a)
\[ \eta_t = \eta_t^{\bullet} \cdot \xi_{t+1}^{\bullet} \gamma_{t+1}^{\bullet} \]  \hspace{1cm} (12.b)
\[ \gamma_t = \gamma_t^{\bullet} \cdot \xi_{t+1}^{\bullet} \gamma_{t+1}^{\bullet}. \]  \hspace{1cm} (12.c)

The above result is summarized by the following corollary.

6 Corollary

Under assumptions A.1 and A.2, the solution \((\bar{\xi}_t, \bar{\eta}_t)\) of the reverse-time system (4) is indistinguishable from the solution \((\xi_t, \xi_t, \xi_t, \xi_t)\) of system (1), if

(I) \((\xi_t, \xi_t, \xi_t) = (\xi_t, \xi_t, \xi_t)\) a.s.,

(II) \(\bar{\xi}_t^{\bullet} = \bar{\xi}_t^{\bullet} \cdot \xi_{t+1}^{\bullet} \gamma_{t+1}^{\bullet} \) a.s.,

(III) \((\xi_t, \eta_t) = (\xi_t, \eta_t)\) a.s; \(\xi_t \in \xi_0 \cup \xi_0 \cup \xi_0 \cup \xi_0 \), and \(\bar{\xi}_t^{\bullet} \cdot \eta_{t+1}^{\bullet} \cdot \xi_{t+1}^{\bullet} \gamma_{t+1}^{\bullet} \), and \(\bar{\xi}_t^{\bullet} \cdot \eta_{t+1}^{\bullet} \cdot \xi_{t+1}^{\bullet} \gamma_{t+1}^{\bullet} \), of the above result is summarized by the following corollary.

Jump-linear example

To illustrate the results obtained so far, let us consider the particular situation of a linear system with first order Markovian switching coefficients and observation noise independent of the state driving noise. Both \(a(s, x, w)\) and \(c(s, x, u)\) are then linear in \((s, x, w, u)\), while the first is \(v\)-invariant and the second is \(v\)-invariant, by which system (1) simplifies to

\[ \xi_{t+1} = A(\theta_{t+1}) \xi_t + B(\theta_{t+1}) \eta_t, \]
\[ \xi_{t+1} = b(\theta_{t+1}) \xi_t, \]
\[ \eta_t = G(\theta_t) \xi_t + H(\theta_t) u_t. \]

Then from Corollary 6 we readily find the indistinguishable time-reversed system

\[ \xi_t = A^{-1}(\theta_{t+1}) \xi_{t+1} - B(\theta_{t+1}) \bar{\eta}_{t+1} \xi_{t+1}, \]
\[ \eta_t = b(\theta_{t+1}) \xi_{t+1}, \]
\[ \eta_t = G(\theta_t) \xi_t + H(\theta_t) u_t. \]
where \((\tilde{w}^T_4, \tilde{v}^T_4)\) is the reverse-time HD-sequence of Theorem 5. \(\hat{u}_\infty(t, \hat{x}_{t+1}, \hat{x}_t, x_{t+1})\) and \(\hat{u}_\infty(t, \hat{x}_{t+1}, \hat{x}_t, x_{t+1})\) are according to (11) and (13.b). The difference equation for \(\hat{u}_\infty\) is similar to the one for the linear Gaussian example in section 2.

But due to \(\hat{u}_\infty\), it may even be nonlinear in \(\hat{x}_{t+1}\). At the end of the next section we will show that there are some further simplifications possible for this example, in case of in probability law equivalence.

5. EQUIVALENT IN LAW TIME-REVERSION

In this section we derive conditions under which the solutions of (1) and (4) are equivalent in law, and discuss these results for a jump-linear example. So far our line of reasoning is quite similar to the martingale approach of time-reversing a diffusion. However, things are quite different now we require equivalence in law only. The reason is that while in the diffusion situation this requires that \(\tilde{u}_\infty\) and \(\tilde{u}_\infty\) are equivalent in law, no similar simple results hold in the discrete-time situation. Instead of this, we identify the relation between conditional laws of \(\tilde{u}_\infty\) and \(\tilde{u}_\infty\) by a Bayesian estimation step. Next we characterize \(f\) and the required law of \(w^*_T\).

7. Theorem

Under assumption A.1 the solution \((X_t, R_t, X_t)\) of reverse-time system (4) is equivalent in law w.r.t. the solution \((\hat{x}_t, \hat{r}_t, \hat{x}_t)\) of system (1) if,

1. \(P((\tilde{y}_t, \tilde{R}_t, \tilde{y}_t)\in d\tilde{x}) = P((\tilde{y}_t, \tilde{R}_t, \tilde{y}_t)\in d\tilde{x})\), for any measurable \(d\tilde{x}\) if

2. \(\tilde{w}\) and \(\tilde{c}\) satisfy (13.a),

3. \(P(\tilde{Y}_t = \tilde{y}_t | \tilde{x}_{t+1} = s, \tilde{R}_t = x)\),

4. \(P(\tilde{Q}_{t+1} = \tilde{q}_{t+1} | \tilde{X}_{t+1} = x, \tilde{R}_{t+1} = x)\), all \((x, \tilde{q}_{t+1}, \tilde{y}_{t+1}, \tilde{x}_{t+1}, \tilde{w}_{t+1}, \tilde{w}_{t} \in \tilde{w}_{t} \in \tilde{v}_{t}\) and \(f\) satisfying (9.a), (10.a) and (11.a).

Proof: See Blom and Bar-Shalom (1989).

Our remaining problem is the characterization of the conditional law of \(\tilde{w}_{t}\). As this is actually a discrete-time nonlinear filtering problem, it can be done by applying Bayes formula. We do this under the following additional assumptions:

A.2. The a priori distribution of \((\hat{x}_t, \hat{y}_t)\) permits a density-mass function for all \(t \in [0, T]\).

A.3. \(\tilde{a}^\hat{c}(x, \tilde{y}, \tilde{w})\) is once differentiable in \(x \in \tilde{X}\) for all \((x, \tilde{y}, \tilde{w}) \in \tilde{X}\).

If the distributions in (iv) of Theorem 7 have density-mass functions then it can easily be verified that (iv) implies,

\[
P(\tilde{X}_{t+1} = \tilde{x}_{t+1} | \tilde{X}_t = x_t, \tilde{R}_t = x_t) = P(\tilde{X}_{t+1} = \tilde{x}_{t+1} | \tilde{X}_t = x_t, \tilde{R}_t = x_t)
\]

where \(\tilde{u}_\infty\) satisfies (10.a). With this our remaining step is to characterize the density at the right-hand side of (14.a) by applying Bayes formula.

8. Proposition

Under assumptions A.2 and A.3, the distribution in (iv) of Theorem 7 permits a density which is characterized by (14.a) and,

\[
P(\tilde{X}_{t+1} = \tilde{x}_{t+1} | \tilde{X}_t = x_t, \tilde{R}_t = x_t) = \int P(\tilde{X}_{t+1} = \tilde{x}_{t+1} | \tilde{X}_t = x_t, \tilde{R}_t = x_t)
\]

with \(\tilde{w}_t\) the gradient and \(\tilde{c}\) either a normalizing factor or zero iff \(P(\tilde{X}_{t+1} = \tilde{x}_{t+1} | \tilde{X}_t = x_t, \tilde{R}_t = x_t) = 0\).

Moreover,

\[
P(\tilde{X}_{t+1} = \tilde{x}_{t+1} | \tilde{X}_t = x_t, \tilde{R}_t = x_t) = \int P(\tilde{X}_{t+1} = \tilde{x}_{t+1} | \tilde{X}_t = x_t, \tilde{R}_t = x_t)
\]

Proof: See Blom and Bar-Shalom (1989).

9. Jump-linear example

For a linear system with first order Markovian switching coefficients we arrived in section 4, at the following reversed-time equation:

\[
x_{t+1} = A^{-1}(x_t) + x_t = B(x_t)x_t^w_t
\]

with \(w^*_t\) the reverse-time HD sequence and \(w^*_t\) of Theorem 7. Because \(a^*\) is linear in \((x, w)\), its gradient w.r.t. \(x\) is \(w\)-invariant, by which proposition 8 yields

\[
P(\tilde{w}_t = \tilde{w}_t^{*} | \tilde{x}_t = x_t, \tilde{R}_t = x_t)
\]

In spite of the simplification this is a form which is in general quite complex, by which \(\tilde{w}_t\) still may be a nonlinear function of \(x_{t+1}\). Obviously, this type of complexity could have been expected, as it is well known that a discrete-time Bayesian estimation step leads to nonlinear equations, unless the prior densities involved are Gaussian.

6. CONCLUDING REMARKS

We considered the problem of reversing the Markov solution of a nonlinear stochastic difference equation in time. The nonlinearities were due to nonlinear coefficients and a hybrid state space, i.e. a product of an Euclidean space and a discrete set. For simplicity, it was assumed that the process in the discrete set satisfies the Markov property. Subsequently we gave a precise description of our time reversal objectives: the development of time reversed difference equations, of forms similar to the original equation, but driven by reversed-time martingale difference sequences, such that their solutions are respectively indistinguishable from and in
probability law equivalent to the solution of the original equation. Following this the derivation of the indistinguishable reverse-time equation was performed. The main new theoretical result is the introduction and evaluation of a non-canonical (Jacod and Shiryaev, 1987) reverse-time martingale decomposition, which is appropriate to the hybrid state space situation. In contrast with this, all previous reverse-time equations are based on a canonical martingale decomposition. After that, it was shown how the in probability law equivalent to the solution of the R.J. Elliott, B.D.O. Anderson, Reverse time diffusion equations, Stochastic Processes and their Applications, Vol. 19 (1985), pp. 277-339.


P.R. Kumar, P. Varaiya, Stochastic systems, Prentice Hall, 1966.


ACKNOWLEDGEMENT

The authors like to thank Robert Washburn (Alphatech Inc.) for his suggestion to study time-reversal for the hybrid state space situation, and an anonymous reviewer for pointing out an error in the preprint.

REFERENCES


A NEW CONTROLLER FOR DISCRETE-TIME STOCHASTIC SYSTEMS WITH MARKOVIAN JUMP PARAMETERS

L. Campo and Y. Bar-Shalom

Univ. of Connecticut, Storrs, CT 06269-3157, USA

Abstract. A realistic stochastic control problem for hybrid systems with Markovian jump parameters may have the switching parameters in both the state and measurement equations. Furthermore, both the system state and the jump states may not be perfectly observed. Currently, only existing implementable controller for this problem is based on a heuristic multiple model partitioning (MMP) and hypothesis pruning. In this paper we present a stochastic control algorithm for stochastic systems with Markovian jump parameters. The control algorithm is derived through the use of stochastic dynamic programming and is designed to be used for realistic stochastic control problems, i.e., with noisy state observations. The state estimation and model identification is done via the recently developed interacting multiple model algorithm. Simulation results show that a substantial reduction in cost can be obtained by this new control algorithm over the (MMP) scheme.

Keywords. Stochastic control: Dynamic programming: Hybrid systems: Multiple model partitioning; Markovian jump parameters.

1. INTRODUCTION

An important problem of engineering concern is the control of discrete-time stochastic systems with parameters that may switch among a finite set of values. In this paper we present the development of a controller for discrete-time hybrid jump-linear Gaussian systems. Here the state and measurement equations have parameter matrices which are functions of a Markov switching process. The jump states are not observed and only the state is observed in the presence of noise.

Along with presenting a desirable practical control algorithm we also point out an interesting theoretical phenomenon. We show that there is a natural connection between the interacting multiple model (IMM) state estimation algorithm [81] and the control of jump-linear systems. Thus the IMM is the state estimation algorithm of choice for use in these types of control problems.

Systems which pertain to the jump-linear modelling methodology are found in many areas. Systems of a highly nonlinear nature can be approximated by a set of linearized models [M3, V1, V2]. A failure in a component of a dynamical system (or subsequent repair) can be represented by a sudden change in the systems parameters [B2, S1, V1]. Also economic problems, which can be modelled by parameters that are subject to sudden changes due to shortages in important materials [G2]. And as is noted in [M6] there also exist applications to the design of control systems for large flexible structures in space.

There has been an extensive amount of work done in this area and on the related problem of controlling stochastic dynamic systems with unknown, time-invariant parameters. We refer the reader to the [T3] and [G3] for a list of references and a discussion of their scope and applications.

Research sponsored under Grant AFOSR-88-0202.

More recently in [S2] a feedforward/feedback controller was presented for the continuous-time problem with a completely observed system state and where the "modal indicator" is measured with a high quality sensor. In [M6] the continuous-time jump-linear problem is considered where the system state and "modal processes" are perfectly observed. The optimal regulator was obtained and notions of stochastic stabilizability and detectability were introduced to characterize the behavior of the optimal system on long time intervals. In [M7] the continuous-time jump-linear problem with additive and multiplicative noises and noisy measurements of the plant state was considered with the plant mode assumed perfectly observed.

In [E1] a sufficient stability test is given for checking the asymptotic behavior of the error introduced by the averaging of hybrid systems. In [M8] the continuous-time jump-linear problem with non-Markovian regime changes was considered. A control scheme was presented for the case of perfect observations of the system state and plant regime.

In [C3] a discrete-time Markovian jump optimal control problem was considered. The controller is for the case of perfect system state observations and known form process. They derive necessary and sufficient conditions for the existence of optimal constant control laws which stabilize the controlled system as the time horizon becomes infinite. Through examples they show the interesting result that stabilizability of the system in each form is neither necessary nor sufficient for the existence of a stable steady-state closed-loop system.

In [Y1] a discrete-time system with perfect state and mode information was considered. A controller was presented which is stabilizing in the mean square exponential sense.

As pointed out in [G2], we generally cannot determine the optimal jump-linear quadratic Gaussian closed-loop control law analytically.
even for a two-step problem. In order to compute the optimal control, extensive numerical search methods must be employed and thus one would like to find simpler suboptimal control schemes.

Currently the only existing implementable controller for this problem (switching parameters in the system state and measurement equations and noisy state observations), is the one discussed in [T3] and is of the OLOF class. This algorithm is based upon a heuristic multiple model partitioning (MMP) and hypothesis pruning. The MMP approach, being simple and straightforward to implement, is a reasonable choice for the unknown parameter identification problem [L1], and as shown in [T3] it works well for applications involving switching parameters in the state measurement equation only. For the non-switching parameter problem the operating mode is determined to a high probability in a relatively short period of time and the MMP approach gives the linear quadratic Gaussian optimal control.

For switching parameter problems a different situation exists. Here because of the switching the operating mode may not be determined to a high probability. The proposed approach to deriving a suboptimal control scheme is to start with the solution to the optimal control problem via the use of stochastic dynamic programming. By utilizing dynamic programming and making appropriate suboptimal assumptions the use of numerical search methods has been avoided. We thus have developed a multiple model control scheme which has the following desirable properties: (a) It gives the optimal final control, (b) the algorithm utilizes the IMM state estimation scheme, and (c) it has the same high property as the MMP approach in that it gives the optimal linear quadratic control under the assumption of a perfectly known model history sequence (which is however an unrealistic assumption for this class of problems).

For comparison purposes we implement the "switching parameters in the system state equation" controller, proposed but not tested in [T3]. We show via example that a statistically significant reduction in cost can be achieved through the use of our controller which also belongs to the OLOF class.

The paper is outlined as follows. In section 2 the problem formulation is given. In section 3 an interesting connection between the IMM state estimation algorithm and the control of multiple model systems is shown to exist. In section 4 we obtain the control algorithm. A new "full-tree" control algorithm is derived which utilizes all possible future parameter history sequences. In section 5 we use simulations to compare the MMP control algorithm with the full-tree controller.

2. PROBLEM FORMULATION

The problem to be solved, is discussed next. We took the pragmatic approach of starting with the available mathematical and statistical tools found to yield success in solving similar problems of this type in the past i.e., use is made of the stochastic dynamic programming method and the total probability theorem, etc.). As we shall see, not only does this practical engineering approach yield an improved multiple model control algorithm, but it also leads to the interesting theoretical observation of a direct connection between the IMM state estimation algorithm and jump-linear control.

It is desired to find a sequence of causal control values to minimize the cost functional

\[ J = E \left[ \left( x(0) + x(0) \right)^T Q^{0.5} x(0) + u(0)^T R^{0.5} u(0) \right] \]

where the cost functional is chosen to be

for each \( k = 0, 1, \ldots, N \) and and it is sufficient that \( R^{0.5} > 0 \) for each \( k = 0, 1, \ldots, N-1 \).

The discrete-time, system state and measurement modeling equations are

\[ x(k+1) = F_{[M(k)]} x(k) + G_{[u(k)]} u(k-1) + v_{[k-1, M(k)]} \]

\[ z(k) = H_{[M(k)]} x(k) + w_{[k, M(k)]} \]

where \( x(k) \) is an \( nx1 \) system state vector, \( u(k) \) is an \( px1 \) control input, and \( z(k) \) is an \( nx1 \) system state observation vector. The argument \( M(k) \) denotes the model at time \( k \) - in effect during the sampling period ending at \( k \). The process and measurement noise sequences, \( v_{[k-1, M(k)]} \) and \( w_{[k, M(k)]} \), are white and mutually uncorrelated.

The model at time \( k \) is assumed to be among several models

\[ M(k) \in \{ 1, 2, \ldots, r \} \]

for example

\[ f_{[M(k)=i]} = f_i \]

\[ v_{[k-1, M(k)=1]} = N(0, V_{[1,1]}) \]

\[ w_{[k, M(k)=1]} = N(0, W_{[1,1]}) \]

i.e., the structure of the system and/or the statistics of the noises might be different from model to model.

The model switching process to be considered here is of the Markov type. The process is specified by a transition matrix with elements \( p_{ij} \). Let

\[ \pi^t := \{ x(0), z(1), \ldots, z(k), u(0), u(1), \ldots, u(k-1) \} \]

denote the information available to the controller at time \( k \) (i.e., the control is causal).

3. THE LAST STAGE CONTROL AND THE CONNECTION WITH THE IMM ESTIMATOR

An integral part of any control algorithm for this class of problems is the system state estimator. In this section we show that there exists an interesting connection between the control of multiple model stochastic systems and the IMM system state estimator [B1]. To this end we start by solving for the time \( N-1 \) optimal control. The optimal control at time \( N-1 \) is the value of \( u(N-1) \) which minimizes

\[ J(N-1) = E \left[ x(N-1)^T Q^{0.5} x(N-1) + u(N-1)^T R^{0.5} u(N-1) \right] \]

\[ \times \left( x(N-1)^T Q^{0.5} x(N-1) + u(N-1)^T R^{0.5} u(N-1) \right) \]

\[ \times \sum_{M} E \left[ x(N-1)^T Q^{0.5} x(N-1) + u(N-1)^T R^{0.5} u(N-1) \right] \]

\[ \times \left( x(N-1)^T Q^{0.5} x(N-1) + u(N-1)^T R^{0.5} u(N-1) \right) \]

\[ \times p[M(N)] \pi^{N-1} \]

\[ p[M(N)] \pi^{N-1} \]

\[ 3.1 \]
\[
J(N-1) = \frac{\delta}{\beta_0} E\{\mathbf{x}(N-1)|\mathbf{Q}(N-1)|\mathbf{f}(N-1)|\mathbf{x}(N-1)\}
\]
\[
+ \sum_{\beta_0} \mathbf{G}_0 \mathbf{Q}(N)|\mathbf{f}(N-1)|\mathbf{x}(N-1)\}
\]
\[
\cdot \mathbf{f}(N-1)|\mathbf{H}(N-1)|\mathbf{u}(N-1)
\]
\[
+ \sum_{\beta_0} \text{tr} \mathbf{Q}(N)|\mathbf{f}(N-1)|\mathbf{u}(N-1)
\]
\[
(3.3)
\]

Taking the partial of (3.3) w.r.t. \(u(N-1)\) and setting it to zero yields
\[
u^*(N-1) = \frac{\delta}{\beta_0} \frac{\partial}{\partial u(N-1)} E\{\mathbf{x}(N-1)|\mathbf{H}(N-1)|\mathbf{u}(N-1)\}
\]
\[
\cdot \mathbf{f}(N-1)|\mathbf{H}(N-1)|\mathbf{u}(N-1)
\]
\[
\cdot \mathbf{f}(N-1)|\mathbf{H}(N-1)|\mathbf{u}(N-1)
\]
\[
(3.4)
\]

Notice that
\[
E\{\mathbf{x}(N-1)|\mathbf{H}(N-1)|\mathbf{u}(N-1)\} = \sum_{\beta_0} E\{\mathbf{x}(N-1)|\mathbf{H}(N-1)|\mathbf{u}(N-1)\}
\]
\[
(3.5)
\]

where, since \(\mathbf{H}(N-1)|\mathbf{u}(N-1)\) is irrelevant, the expectation inside the summation is
\[
E\{\mathbf{x}(N-1)|\mathbf{H}(N-1)|\mathbf{u}(N-1)\} = \sum_{\beta_0} E\{\mathbf{x}(N-1)|\mathbf{H}(N-1)|\mathbf{u}(N-1)\}
\]
\[
(3.6)
\]

which is the IMM mixed initial estimate (BI).

Thus using (3.6) in (3.1) we get
\[
u^*(N-1) = \frac{\delta}{\beta_0} \frac{\partial}{\partial u(N-1)} E\{\mathbf{x}(N-1)|\mathbf{H}(N-1)|\mathbf{u}(N-1)\}
\]
\[
\cdot \mathbf{f}(N-1)|\mathbf{H}(N-1)|\mathbf{u}(N-1)
\]
\[
(3.7)
\]

4. THE CONTROL ALGORITHM

We will derive a full-tree control algorithm (FT) which computes control values by taking into account all possible future model histories. As will be seen by our example this method offers improved performance over the existing scheme [T3].

The 1-th future history of models is denoted as
\[
\mathbf{M}^{(1)} = \{\mathbf{M}(k)=\mathbf{I}_1, \ldots, \mathbf{M}(N)=\mathbf{I}_1\} \quad I = \ldots, \ldots, 1^{N-1} \quad (4.1)
\]

where \(\mathbf{I}_1\) is the model at time \(I\) from history 1 and
\[
1 \leq I \leq \mathbf{I} \quad \mathbf{I} = 1, \ldots, \mathbf{N} \quad (4.2)
\]

The control that minimizes an approximation to (4.4) is derived in the Appendix, and is given by
\[
u^*(k) = \min_{\beta_0} \mathbf{E}\{\mathbf{x}(k)|\mathbf{Q}(k)|\mathbf{u}(k) \cdot \mathbf{f}(k)|\mathbf{R}(k)|\mathbf{u}(k) \}
\]
\[
\cdot \mathbf{E}\{\mathbf{x}(k-1)|\mathbf{Q}(k-1)|\mathbf{u}(k) \cdot \mathbf{f}(k-1)|\mathbf{R}(k-1)|\mathbf{u}(k) \}
\]
\[
(4.3)
\]

where \(\mathbf{J}(k)^4\) is the optimal cost-to-go from time \(k\) to the end. Now applying the total probability theorem to (4.3) yields
\[
\mathbf{J}(k,N) = \min_{\beta_0} \mathbf{E}\{\mathbf{x}(k)|\mathbf{Q}(k)|\mathbf{u}(k) \cdot \mathbf{f}(k)|\mathbf{R}(k)|\mathbf{u}(k) \}
\]
\[
\cdot \mathbf{E}\{\mathbf{x}(k-1)|\mathbf{Q}(k-1)|\mathbf{u}(k) \cdot \mathbf{f}(k-1)|\mathbf{R}(k-1)|\mathbf{u}(k) \}
\]
\[
(4.4)
\]

5. SIMULATION RESULTS

The FT controller developed in Sec. 4 is used to control the state trajectory of the system. The performance of this algorithm, as determined by (2.4), is compared to the cost obtainable by using the MMP controller discussed in [T3]. In order to obtain a meaningful comparison we use the rigorous statistical analysis technique presented in [85, W3].

The control of a double integrator system with process and measurement noises is considered with a gain failure. The two possible models are given by the following system equation
\[
x(k+1) = \begin{bmatrix} T^2/2 & 0 \\ T/2 & 1 \end{bmatrix} x(k) + \begin{bmatrix} 0 \\ b^2 \end{bmatrix} u(k)
\]
\[
(5.1)
\]

with measurement equation
\[
z(k) = \begin{bmatrix} 1 & 0 \end{bmatrix} x(k) + w(k)
\]
\[
(5.2)
\]

The models differ in the control gain parameter \(b^2\). The process and measurement noises are mutually uncorrelated with zero mean and variances given by
\[
\mathbf{E}\{w(k)|w(k)\} = 0.16 \delta_{ij}
\]
\[
(5.3)
\]

and
\[
\mathbf{E}\{w(k)|w(k)\} = \delta_{ij}
\]
\[
(5.4)
\]

The control gain parameters were chosen to be
\[
b^2 = 2 \quad \text{and} \quad b^2 = 0.5
\]

The Markov transition matrix was selected to be
For this example $N=7$, and the cost parameters \(R(k)\) and \(Q(k)\), (see (2.1)), were selected as
\[
R(k) = 5.0 \quad k=1,2,\ldots,N-1
\]
and
\[
Q(0) = \begin{bmatrix} 0.1 & 0.9 \\ 0.9 & 0.1 \end{bmatrix}
\]
For this example $T=10$, and the cost parameters \(R(k)\) and \(Q(k)\), (see (2.1)), were selected as
\[
R(k) = 5.0 \quad k=1,2,\ldots,N-1
\]
and
\[
Q(0) = \begin{bmatrix} 8 & 8 \\ 8 & 8 \end{bmatrix}
\]
where the last matrix, \(Q(7)\), reflects our desire to drive \(x(7)\) vigorously to zero. Also note that for this example $T=10$.

The real system was initialized with \(x(0)=[30.0, 0.0]'\) and a random selection was done for choosing the initial model with \(P(0|0)=I+0.5, 1=1,2\). The Kalman filters each received an initial state covariance of
\[
P(0|0) = \begin{bmatrix} 1.0 & 1.0 \\ 1.0 & 2.0 \end{bmatrix}
\]
and the initial state estimate was selected as
\[
\begin{bmatrix} \hat{x}_1(0|0) \\ \hat{x}_2(0|0) \end{bmatrix} = \begin{bmatrix} z(0) \\ z(0) - z(-1) \end{bmatrix}
\]
where \(z(-1) = 30.0 + w(-1)\) and \(z(0) = 30.0 + w(0)\).

Statistical tests were made on the results of 50 Monte Carlo runs. Sample means and variances of the Monte Carlo costs \(C_1\) defined in (2.1) were computed for the FT, MMP, and "known model-history" (i.e. optimum linear quadratic) controllers.

Table I contains the results. The FT algorithm shows a clear reduction in cost as compared with the MMP scheme. However in order to provide a rigorous argument that the actual performance is order as Table I indicates we apply the statistical test presented in [85, W3].

Table II contains the results. The sample standard deviation \(\sigma_\Delta\) of the mean of the cost differences, \(C_{1}^{\text{FT}}-C_{1}^{\text{MMP}}\), are shown.

The hypothesis that the FT controller is better than the MMP scheme can be accepted only if the probability of error \(\alpha\) is less than, say, 1 percent. Then the threshold against which we compare the test statistic \(\Delta/\sigma_\Delta\) is \(\mu=2.33\). This test statistic has to exceed the threshold in order to accept the hypothesis.

### Table I

<table>
<thead>
<tr>
<th>Sample Average Costs and Standard Deviations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model-History</td>
</tr>
<tr>
<td>Sample Mean</td>
</tr>
<tr>
<td>Sample Standard Deviation</td>
</tr>
</tbody>
</table>

### Table II

<table>
<thead>
<tr>
<th>Statistical Test for Algorithm Comparisons</th>
</tr>
</thead>
<tbody>
<tr>
<td>FT-MMP</td>
</tr>
</tbody>
</table>

### 6. CONCLUSION

The development of a new control algorithm for discrete-time hybrid stochastic systems with Markovian jump parameters has been presented. This controller was derived through the use of stochastic dynamic programming and by taking into account all possible future "histories of models". This scheme uses the IMM state estimation algorithm. We show that there is an interesting connection between the IMM state estimator and control of jump-linear hybrid systems. This new controller is of the OLOF class and has off-line computable control gain parameters.

From the example it is seen that this scheme can achieve a statistically significant reduction in cost when compared to the multiple model partitioning approach.

### APPENDIX

#### 1. Derivation of (4.5)

Note that given the future history of models \(H^{k+1,k}\), the optimal cost-to-go \(J^{\text{FT}}(k+1,k^{*+1})\) is easily computed and is denoted.

\[
J^{\text{FT}}(k+1,k^{*+1}) = \left[ \mathbb{E}\left\{ x(k+1)'P(k+1|x(k+1), k^{*+1}, H^{k+1,k}) \right\} \right]^{1/2}
\]

where the notation from [84] is used for \(P(k+1)\) and \(\alpha(k+1)\).

Since the expectation in (4.4) is conditioned on \(H^{k+1,k}\), we obtain our of approximation by replacing \(J^{\text{FT}}(k+1,k^{*+1})\) inside the expectation with (A.1) and (4.4) becomes

\[
J^{\text{FT}}(k+1) = \min_{k^{*+1}} \left[ \sum_{k} \mathbb{E}\left\{ x(k)'Q(k|x(k), u(k)'R(k)u(k) \right\} \right]^{1/2}
\]

\[
+ \left[ \mathbb{E}\left\{ x(k+1)'P(k+1|x(k+1), k^{*+1}, H^{k+1,k}) \right\} \right]^{1/2} + \alpha(k+1) |H^{k+1,k}|^{1/2} \right\} \right]^{1/2}
\]

\[
\text{APPENDIX}
\]
where
\[ u_t[N-k+1] & \in P[X^{t+k} \mid X^t] \tag{A.3} \]

Now use (2.2a) and apply the smoothing property of expectation to (A.2) to get
\[
J'(k') = \sum_{n_i} \left[ E \left[ x(k) \mid X^{t+k} \right] u_t'[R(k) u_t] \right]
\]
\[
+ \left[ F_t - F_{k-t} u_t(k) \right] \delta(k-1,1) p_t[-] - \left[ \alpha(k+1) \right] 1 \mid X_{k} ] \right] u_t[N-k+1] \tag{A.4} \]

Take the partial w.r.t. \( u_t(k) \) of (A.4) and set to zero to solve for
\[
u_t(k) = \left[ - \sum_{n_i} \left[ G_t \cdot p_t[-] G_t \cdot p_t[-] u_t[N-k+1] \right] \right]^{-1} \cdot \left[ \sum_{n_i} \left[ G_t \cdot p_t[-] F_t - F_{k-t} u_t(k) \right] \delta(k-1,1) p_t[-] - \left[ \alpha(k+1) \right] 1 \mid X_{k} ] \right] u_t[N-k+1] \tag{A.5} \]

We still need to evaluate the expectation in (A.5). This is done as follows. Note that \( x(k) \) is independent of \( M(t) = 1 \) if \( k \leq 2 \), ..., \( N \) if \( M(k+1) \) is known, thus
\[
E \left[ x(k) \mid X^{t+k} \right] = E \left[ x(k) \mid X^{t+k} \right] 1 \mid X_{t} ] \right] \] \tag{A.6} \]

But (A.6) is \( x \circ (k, k) \), the IMM mixed initial estimate (see (3.6)), thus using (A.6) in (A.5), we get (A.4).

REFERENCES

FROM PIECEWISE DETERMINISTIC TO PIECEWISE DIFFUSION MARKOV PROCESSES

Henk A.P. Blom
University of Connecticut, ESE Dept.

ABSTRACT

Piecewise Deterministic (PD) Markov processes form a remarkable class of hybrid state Markov processes, in contrast to most other hybrid state processes, they include a jump reflecting boundary and exclude diffusion. As such, they cover a wide variety of piecewise, locally controlled non-diffusion processes. Because PD processes are defined in a pathwise way, they provide a framework to study the control of non-diffusion processes along the same lines as that of diffusions. An important generalization is to include diffusion in PD processes, but, as pointed out by Davis, combining diffusion with a jump reflecting boundary seems not possible within the present definition of PD processes. This paper presents PD processes as pathwise unique solutions of an Itô stochastic differential equation (SDE), driven by a Poisson random measure. Since such an SDE permits the inclusion of diffusion, this approach leads to a large variety of piecewise diffusion Markov processes, represented by pathwise unique SDE solutions.

1. INTRODUCTION

Because many of the stochastic processes that we meet in nature have a state space that is a product of a continuous space and a discrete set, we often need pathwise models on such a hybrid state space. As a result, several classes of hybrid state space models have been developed, such as systems with Markovian switching coefficients, doubly stochastic counting processes and Markov decision drift processes. These models are used in quite different fields of applications, by which their studies have often evolved separately. One reason to study hybrid state space processes within a common framework is that their martingale parts in general are jumps. This property has attracted a lot of attention, and is by now very well documented (Jacod, 1979; Cinler et al., 1980; Bremaud, 1981; Elliott, 1984; Bensoussan and Lions, 1984; Cinlar, 1971; Jacod and Shiryaev, 1987). It is quite clear from these results that, to study hybrid state Markov processes along the same lines as diffusions, we need both pathwise representations and strong Markov (martingale) characterizations of those processes. Unfortunately, for hybrid state Markov processes there is presently a lacuna of pathwise representations with strong Markov characterizations. This lacuna is apparent if we depict the main classes of hybrid state Markov processes in the form of a Venn-diagram (Fig. 1).

There exist pathwise representations with strong Markov characterizations of counting processes with diffusion intensity (Snyder, 1975; Marcus, 1978), of diffusions with Markovian switching coefficients (Wong, 1970; Brockett and Blankenship, 1977) and of Piecewise Deterministic (PD) Markov processes (Davis, 1984). For many other Markov processes in figure 1, there exist only strong Markov characterizations (Kingman, 1975; Anulova, 1979, 1982; Bensoussan and Lions, 1984; Belbas and Lenhart, 1986). Actually, PD Markov processes seem the most interesting of all processes in figure 1, as they provide pathwise representations with a strong Markov characterization of all major non-diffusion Markov processes. As such, PD Markov processes provide a framework to study Markov decision drift processes (Kordijk and Van der Duyn Schouten, 1983; Yushkevich, 1983) along the same line as diffusions (Verhove, 1985). With this, an interesting generalization is to extend the spectrum of hybrid state Markov processes by including diffusion into PD Markov processes. As the present definition of PD processes does not seem to have an opening left for that inclusion (Davis, 1984), we need a different approach.

The approach that overcomes this difficulty, presented in the sequel, is to assume a stochastic differential equation (SDE) in a hybrid space and to construct a large class of piecewise diffusion Markov processes from it. With respect to the state space we restrict our attention to a hybrid subset of a Euclidean space. Then the most general SDE is of Itô type, driven by Brownian motion, W, and a Poisson random measure, \( \pi \),

\[
\text{d}t = a(t) \text{d}t + \int \int \pi (dt, du),
\]

where \( p \) generates a multivariate point \((t, u)\), then the path of \( t \) has a discontinuity:

\[
t_\ell = t_{\ell-} + * (t_{\ell-}, u).
\]

In the sequel we shall focus on pathwise unique solutions. The classical result for the existence of such solutions requires that \( * \) is sufficiently continuous (Gilman and Skorohod, 1972) which restricts the SDE essentially to systems with Markovian switching coefficients. However, there are some non-classical pathwise uniqueness results that allow a discontinuous \( * \) (Lepeltier and Marchal, 1976; Jacod and Protter, 1982; Veretennikov, 1988). Taking these results as a starting point, we introduce and evaluate a particular structure for \( * \) in section 2. This structure poses hardly any restrictions on the possible solution of the SDE, while it enables a separate evaluation of an unbounded jump intensity and a hybrid state space situation. In view of this separation, we first consider, in sections 3 and 4, the modelling of a jump reflecting boundary in \( \mathbb{R}^n \) through an unbounded jump intensity, and after that, in section 5, we consider the hybrid state situation.

Assume an open subset \( \mathcal{O} \) of \( \mathbb{R}^n \) with jump reflecting boundary \( \partial \mathcal{O} \), which means that \( (t, u) \) undergoes an
Instantaneous jump into the interior of 0 if $t_\epsilon$ tries to cross or to travel through 0. To model this with the above SDE, the Poisson random measure $\mu^t$ should normally generate a point at any time $t_\epsilon$ enters 0. However, this is not possible as a Poisson random measure generates almost surely no point when $t_\epsilon$ enters 0. Therefore, we briefly discuss the following three approaches:

1. Replace $\mu^t$ by a random measure, with a probability of one point at an arbitrary time.
2. Assume a $\lambda$ such that $\mu^t$ generates an active point during an infinitesimal small time interval after entering 0.
3. Assume a $\lambda$ such that $\mu^t$ generates an active point during an infinitesimal small time interval just before entering 0.

Approach 1 solves the instantaneous jump problem but creates many non-problems, because if $\mu^t$ is not a Poisson random measure, then the SDE cannot be analyzed within the powerful Itô framework.

Approach 2 is a modification of a PD Markov process. Approach 3 is the desired solution. However, the problem with approach 3 is that it is not known how to carry it out.

A constructive answer to this will be given in the sequel. It is clear that approach 3 needs a kind of boundary condition of the time that $t_\epsilon$ might enter 0. Actually, PD Markov processes are presently the only processes for which this problem is solved (Davis, 1984). As such, we first outline that some of the present results of section 3 are known Borel set 0 $\in R^d$.

Assumptions

**A.1.** There is a constant $K$ such that, for all $t \in R^d$, $|f(t)|^2 + |g(t)|^2 + \|{\mu^t}\|^2 \leq K(1+|t|^2)$.

**A.2.** For all $k \in N$ there exists a constant $K_k$ such that, for all $t \in R^d$ and $y$ in the ball $B_k=(0,2^{k+1})$, $|f(t)\cdot y|^2 + |g(t)\cdot y|^2 + \|{\mu^t}\|^2 \leq K_k|y|^2$.

**A.3.** For all $k \in N$, there is a constant $K_k$ such that, with $B_k$ the ball of $R^d$, $R^d=\left\{ \left( y, t, u \right) \in R^d \right\}$,

**A.4.** For all $k \in N$, there is a constant $K_k$ such that,

**A.5.** For all $k \in N$, there is a constant $N_k$, such that, for all $k \in N$, $|\mu^t| \leq N_k$.

2.2 Proposition

Given $(\mu^t)(\mu^t)du dx$ and assumptions A.1. A.2. A.3. A.5 are satisfied. Then equation (1) has for any $\Phi(t)$ a pathwise unique solution, $(\mu^t)$.

Moreover, $(\mu^t)$ is then a right continuous Markov process.

Remark: Proposition 2.1 is a version of Theorem III of Lepeltier and Marchal (1976), in the sense that they considered the situation of $\Phi^t$.

Nevertheless, for the proof we can almost follow Lepeltier and Marchal. Another recent extension of Theorem III of Lepeltier and Marchal is to the situation of a non-Lipschitzian $\Phi^t$ in turn of a sufficient non-degeneracy assumption on $\Phi$ (Veretennikov, 1988).

Proof:

If (1)'s fourth right hand term vanishes, then it is well known that A.1 and A.2 are sufficient conditions (Gihman and Skorohod, 1972). As such, we have to show that (1)'s fourth right hand term does not change that situation, under A.3. A.4 and A.5.

Due to A.3 and the definition of Itô integration a solution of (1) is CADG. Due to A.4, the discontinuities in $(\mu^t)$'s fourth right hand term, are countable. Therefore we can associate with each discontinuity a time $t_\epsilon$. 
and a multi-variate point, $u_t$, such that

$$0 < T_1 < T_2 < \ldots < T_k$$

and $|a| = T_1 = 0$. Due to the latter and $(t_1)$ being CADLAG,

$$\int_0^T \mathbb{d}a(t_1) \mathbb{d}a(t_1) - 0 = \mathbb{d}a(T_1) \mathbb{d}a(T_1).$$

If (1) is a third right hand term which, then the latter sum is finite (a.s.) for all $\Omega_k$, due to $A_0$ and $A_0^-$. With this result it is sufficient to show that (1) has a pathwise unique solution on an arbitrary finite time-interval $[0, T]$. For the existence of a solution, see the proof of Th. III, p. 62-65. We already that a solution is unique and $\mathbb{d}a(t_1)$ measurable on $(0, T_1)$. Because $t_1$ is CADLAG and $\mathbb{d}a(t_1)$ measurable, then by the definition of a Poisson random measure (Jacod and Shiryaev, 1987, p. 65-66) $u_t$ is $\mathbb{d}a(t_1)$ measurable. Therefore, we can repeat the procedure to show that pathwise uniqueness holds true on $[T_1, T_2]$ and $t_1$, $t_2$. Due to the latter, we can repeat the procedure to show that pathwise uniqueness holds true on $[T_2, T_3]$, and so on for the countable sequence of intervals. Q.E.D.

The interesting aspect of proposition 2.1 is that the coefficients of (1) is a fourth right hand term may be discontinuous in $t$. This is exactly what we need, to construct a class of hybrid state Markov processes that is larger than the class of solutions of systems with Markovian switching coefficients. The first step towards this construction is that (1) is pathwise unique solution on an extended generator, as defined in Theorem 2.2, equation (9). Obviously, (A2) is of finite variation on any finite time-interval, while $(t_1 \rightarrow \infty)$ is a local martingale $(t_1)$ is a (special) semimartingale (Jacod and Shiryaev, p. 445, Def. 4.21). This immediately implies that $(t_1)$ is a $\mathbb{d}a(t_1)$ measurable. Because $(t_1)$ is a semimartingale, the generator $A$ follows from its definition for the dual filtration process (Elliott, 1982). Q.E.D.

3. PIECEWISE DETERMINISTIC MARKOV PROCESSES

In this section, we represent Markov processes as solutions of an SDE. Therefore, we consider (2.1, b) with $\mathbb{d}a(t)$ and $\mathbb{d}a(t)$ vanishing on $\Omega^x R_+$. A measurable mapping $F$ from $[0, T]$ to $\Omega^x R_+$ is constructed from 0 and $\alpha$. The construction of $F$ will be based on the following differential equation, on $(0, \omega) R_+$:

$$\mathbb{d}a(t) = a(t) \mathbb{d}a(t) + \mathbb{d}a(t),$$

with $\mathbb{d}a(t)$ and $\mathbb{d}a(t)$ measurable on $R^2$. Then equation (2.1, b) has for any $a$, a pathwise unique solution $\mathbb{a}(t)$. Moreover $(t, t)$ is then a Markov process, of which the the sample paths are measurable on the stochastic basis $(A, \mathcal{F}, P)$.

Proof: Because, on $0 < a(t)$ is continuous in $t$ (due to $A_0^*$ and $A_0^*$), $\alpha$, $A_0^*$, $A_0^*$ are satisfied. Then equation (2.1, b) has for any $a$, a pathwise unique solution $\mathbb{a}(t)$. Moreover $(t, t)$ is then a Markov process, of which the sample paths are measurable on the stochastic basis $(A, \mathcal{F}, P)$.

Having theorem 2.2, we are prepared to consider a jump reflecting boundary (in sections 3 and 4) and the hybrid state space situation (in section 5). But first we give a strong Markov characterization of $(t_1)$ if there is no reflecting boundary.

2.3 Proposition

Given $F$ vanishes everywhere and the assumptions of theorem 2.2 are satisfied. Then for all $t \in R_+$, $(t_1)$ is a semi-martingale strong Markov process, and its extended generator is satisfied by:

$$A(t) = A(t) + A_*, \quad \forall t \geq 0,$$

and

$$F(t) = F(t) + F_*, \quad \forall t \geq 0,$$

with $F(t)$ and $F_*$ measurable on $R^2$. Then $(t_1)$ is a pathwise unique solution on an extended generator, as defined in Theorem 2.2, equation (9).
Next, we come to the main result of this section, which implies that \((t_0)\) is \textit{a Piecewise Deterministic Markov process}.

\subsection*{1.2 Theorem}

With probability one, the process \((t_0)\), of corollary 3.1, exits \(0 + 0\) zero times on \((0, a)\).

\textbf{Proof:}

By the definition of \(F\), all points of \(p\) in \(R^n\) become active as soon as \((t_0)\) exits \(0\). This situation holds on until \((t_0)\) reenters \(0\). The reentering may occur due to drift or due to a jump generated by a point of \(p\) in \(R^n\). Obviously, the cases that \((t_0)\) reenters \(0\) by drift without exit of \(0 + 0\) do not cause any difficulties. In all other cases, the probability of exit \(0 + 0\) by drift is

\[ \gamma \exp(-r/s) \, ds = \gamma \exp(-r/s) \, dr, \]

where \(r = \inf(1/t)\), \(s \geq 1\) and \(r/s\) the intensity of points of \(p\) in \(R^n\). Because \((t_0)\) exits \(0\) at most a \textit{countable number of times}, the probability of exit \(0 + 0\) at least once is then \(\gamma \exp(-r/s)\). If all points of \(p\) are active, then because \(K\),

\[ 1 + \exp(-r/s) = 0, \]

which means a zero probability to exit \(0 + 0\) on \((0, a)\). Q.E.D.

\subsection*{1.3 Theorem}

The process \((t_0)\), of corollary 3.1, is a \textit{semimartingale} strong Markov process, and its extended generator, \(d\), is given by:

\[ d = d_{t_0} + d = d_{t_0} + \dot{d} \text{, for all } \mathcal{E}(A), \]

where \(d\) and \(d_{t_0}\) are given in proposition 2.3 with \(0\) and 
\(\mathcal{D}(\mathcal{E})\) the domain of \(d_{t_0}\).

\textbf{Proof:}

Define a process \(A_{t_0}\) as follows:

\[ A_{t_0} = \int_0^{t_0} e(t_0) \, ds + \int_0^{t_0} f(t_0) \int_{t_0} \mathcal{D}(\mathcal{E}) \, d\mu_{s_0}(du), \]

with \(S_1\) the \(\mathcal{E}\)-adapted times that \((t_0)\) jumps from \(R^2 - 0\) into \(0\) at \(S_0\) and \(S_0\),

\[ S_0 = \int_0^{t_0} e(t_0) \, ds + \int_0^{t_0} f(t_0) \int_{t_0} \mathcal{D}(\mathcal{E}) \, d\mu_{s_0}(du), \]

Obviously, \((A_{t_0})\) is of finite variation on \(0\) and \(S_1\) is a \textit{local martingale}, i.e., a \textit{martingale}, which implies that \((t_0)\) is a \textit{semimartingale}. Application of Itô's differentiation rule for discontinuous (piecewise deterministic) semimartingales to \((t_0)\), with \(t < \infty\), yields:

\[ f(t_0) = f(t_0) + \frac{1}{2} f(t_0) \, ds + \int_{t_0} \mathcal{D}(\mathcal{E}) \, d\mu_{s_0}(du), \]

up to indistinguishability.

Substitution of \(A_{t_0}\),

\[ d(t_0) = d(t_0) + d = d(t_0) + \dot{d} \text{, for all } \mathcal{D}(\mathcal{E}), \]

and using \(t \in \mathcal{D}(\mathcal{E}) \in \mathcal{D}(\mathcal{E}) \setminus \mathcal{D}(\mathcal{E})\), yields

\[ f(t_0) = f(t_0) + \frac{1}{2} f(t_0) \, ds + \int_{t_0} \mathcal{D}(\mathcal{E}) \, d\mu_{s_0}(du), \]

for all \(t \in \mathcal{D}(\mathcal{E})\).

Next, we use the property that \(f(t_0) = 0\), for all \(t \in \mathcal{D}(\mathcal{E})\).

Because \(f(x)\) is a linear growth \((t_0)\) is locally bounded, \(e(t_0)\) is locally bounded. This implies that \((t_0)\) does not increase while travelling through \(0 + 0\) at \(0\), as this takes a time interval of zero duration. The latter and the assumptions that \(f(t_0) = 0\) and \(f(t_0) = 0\) for all \(t \in \mathcal{D}(\mathcal{E})\), imply that \(f(t_0) = 0\) for all \(t \in \mathcal{D}(\mathcal{E})\). With this,
given by:

\[ \mathcal{D} = \mathcal{F} + \mathcal{F}' = \mathcal{F} + \{ \text{events} \} \]

where \( x \) and \( y \) are given in proposition 2.3, while the domain of \( \mathcal{D} \) is given by:

\[ \mathcal{D} = \{ (t, f, g) \in \mathcal{C}^2 \times \mathcal{P}^2 \} \]

Proof: Similar to the proof of proposition 3.1, except that now \( f'(u) \equiv 0 \), for all \( t \geq 0 \), follows from \( \mathcal{F}(0,0,0) \).

Q.E.D.

Finally, we consider the more general situation with \( \theta(t) \) being positive semidefinite. The construction of \( F \) works according to equations (17), (18), (20), (21) and (22), but with distance function:

\[ d_0(t, \Omega) = \inf \{ r(\Omega, \Omega) \geq 0 \} \]

where the set of \( \Omega \) that is accessible by \( (t') \) from \( 0 \), is the empty set and \( \Omega \), is a closure of an n-dimensional ellipsoid, with centre \( \sum_{i=1}^{\infty} \sup_{(t\in\Omega)} \mathcal{F}(t,0) \) and shape defined by covariance \( \theta(t) \). Obviously, \( d_0(t, \Omega) \) is measurable, by which the \( t \)'s and \( \Omega \)'s are Borel sets and \( F \) is measurable, and we get:

4.4 Corollary

Given an open subset \( 0 \) of \( \mathbb{R}^n \), and a mapping \( F \), defined by (17) through (22), then, under the assumptions of theorem 2.2, equation (17) has for any \( t \geq 0 \) a pathwise unique solution \( (t', \xi) \). Moreover, \( (t', \xi) \) is a Markov process, with sample paths measurable on the stochastic basis \((\omega, \mathcal{F}, \mathcal{P})\).

Next, we come to the main result of this section. The main objective of this section is to show that the last term of (23.b) generates a particular type of jump: a jump in \( (t') \) that anticipates a simultaneous switching of \( (t') \). For short we refer to this type of jumps as hybrid jumps. Notice that these hybrid jumps are in some sense unexpected, as all coefficients of (23.a,b) are non-anticipating. To show these hybrid jumps explicitly, we need some preparation.

5.1 Lemma

Under assumptions A.1, A.2, A.3, A.4 and A.5, the pair of equations (23.a,b) has for any \( t \geq 0 \) a pathwise unique solution \( (t', t) \), where \( t \) is a multivariate counting process on \( \mathbb{R}^k \times \mathbb{R}^d \) of a predictable intensity, \( \lambda(t') \). Moreover both \( (t', t) \) and \( (t') \) are then semimartingale strong Markov processes, of which \( (t') \) is indistinguishable from the one in theorem 2.2.

Proof:

It follows from theorem 2.2, that the system of equations (23.a,b) and (23.a) has, for any Borel \( U \), a pathwise unique solution \( (t', t) \). With this system (23.a,b), (23.a) has a pathwise unique solution \( (t', t) \). Obviously all potentially active points of \( p \), that are in \( \mathbb{R}^k \times \mathbb{R}^d \), are collected by \( t' \) in a predictable way, by which we can write \( \mathcal{R} \times \mathcal{R} \) of \( (t', t) \) up to indistinguishability. This implies that the solution of (23.b) is indistinguishable from the solution of (23.b). Q.E.D.

Now we are prepared to consider the hybrid state space situation. Therefore we assume that the first component of \( t' \) is \( N \)-valued, with \( M \mathcal{N}(1,2,..) \), and we can write the first scalar equation of (23.b) as follows:

\[ (t', t) = \mathcal{R} \times \mathcal{R} \] of \( (t', t) \), with \( t' \) a mapping of \( \mathbb{R}^k \times \mathbb{R}^d \) into the integer lattice \( Z \).

Next we assume that \( \lambda \) satisfies, for all \( u \in \mathcal{O}(A(t')) \),

\[ \lambda(t') = \mathcal{R} \times \mathcal{R} \]

where \( \varphi \) is a measurable mapping of \( \mathbb{R}^k \times \mathbb{R}^d \) into \( \mathbb{R}^{k \times n} \), and \( \lambda \) is a measurable mapping of \( \mathbb{R}^k \) into \( \mathbb{R}^n \), such that \( (t') = (t) \). Moreover, we assume that for all \( \lambda \in \mathcal{L}(E R^d) \),

\[ \lambda(t') = \mathcal{R} \times \mathcal{R} \]

which together with (24) and (\( A \times 0 \)) implies that if \( t' \) is \( \mathcal{R} \times \mathbb{R}^d \) of \( t \), such that \( (t') = (t) \).

Similarly, we get the following lemma 4.2 for these cases.

5.6 Theorem

Given the assumptions of corollary 4.4 are satisfied for \( (t', t) \), then \( (t', t) \) is a semimartingale strong Markov process, and its extended generator, \( \mathcal{L} = \mathcal{L} + \mathcal{L}' \), is given by:

\[ \mathcal{L} = \mathcal{L} + \mathcal{L}' \]

where \( \mathcal{L} \) and \( \mathcal{L}' \) are the generators given in proposition 2.3, while the domain of \( \mathcal{L} \) is:

\[ \mathcal{D} = \{ (t, f, g) \in \mathcal{C}^2 \times \mathcal{P}^2 \} \]

Proof: Similar to the proofs of theorem 3.3 and proposition 4.3.

5. THE HYBRID STATE SPACE SITUATION

In this section we explicitly consider the hybrid state space situation for a system of the form (23.a,b), in such a way that they, by which we are assuming a particular \( \theta \) or \( \lambda \). As such, all jump reflecting boundary results of the former sections fit into the results of this section. For ease of notation and interpretation, we rewrite the SDE form (23.a,b), by replacing the Poisson random measure, \( \mathcal{P} \), by a multivariate counting process, \( \mathcal{P} \), such that the pathwise uniqueness of (23.a,b)'s solution is preserved. We do that by defining, for all Borel \( \mathcal{U} \times \mathbb{R}^d \),

\[ \mathcal{R} \times \mathcal{R} \]

and then rewriting (23.a) as

\[ \mathcal{R} \times \mathcal{R} \]

(23.a)

The main objective of this section is to show that the last term of (23.b) generates a particular type of jump: a jump in \( (t') \) that anticipates a simultaneous switching of \( (t') \). For short we refer to this type of jumps as hybrid jumps. Notice that these hybrid jumps are in some sense unexpected, as all coefficients of (23.a,b) are non-anticipating. To show these hybrid jumps explicitly, we need some preparation.

5.1 Lemma

Under assumptions A.1, A.2, A.3, A.4 and A.5, the pair of equations (23.a,b) has for any \( t \geq 0 \) a pathwise unique solution \( (t', t) \), where \( t \) is a multivariate counting process on \( \mathbb{R}^k \times \mathbb{R}^d \) of a predictable intensity, \( \lambda(t') \). Moreover both \( (t', t) \) and \( (t') \) are then semimartingale strong Markov processes, of which \( (t') \) is indistinguishable from the one in theorem 2.2.

Proof:

It follows from theorem 2.2, that the system of equations (23.a,b) and (23.a) has, for any Borel \( U \), a pathwise unique solution \( (t', t) \). With this system (23.a,b), (23.a) has a pathwise unique solution \( (t', t) \). Obviously all potentially active points of \( p \), that are in \( \mathbb{R}^k \times \mathbb{R}^d \), are collected by \( t' \) in a predictable way, by which we can write \( \mathcal{R} \times \mathcal{R} \) of \( (t', t) \) up to indistinguishability. This implies that the solution of (23.b) is indistinguishable from the solution of (23.b). Q.E.D.
A.5

A. For all \( (t^0) \), \( \{s, t, t^0 \} \) is continuous in \( t \), \( \lambda(t, t^0) = 0 \) for all \( t \neq t^0 \).

B. \( \{t, \} \) exits \( 0 \) at most a countable number of times.

5.2 Theorem

Given the hybrid space \( \mathbb{O} = \{0N x \mathbb{R}^n \} \),

Under assumptions A.1 through A.5, the system of equations \( (t^0) \) has a pathwise unique solution \( \{t, \} \). Moreover \( \{t, \} \) is a semimartingale strong Markov process in \( \mathbb{R}^n \).

Proof:

Due to A.1 and A.2, (24) defines \( \xi \) as a measurable mapping (see proof of theorems 2.2), by which \( (26.a, b, c) \) is a special case of \((23.a, b, c)\). Next we show that \( A.4 \) implies \( A.4 \), by which lemma 5.1 and (24) imply that the solution of \((23.a, b, c)\) is indistinguishable from the solution of \((26.a, b, c)\).

To arrive at \( A.4 \), we start from \( A.4 \) and subsequently use \( A.5 \), interchange order of integration and substitution \((24)\).

Q.E.D.

Due to its extensive form, equation \((26.a, b, c)\) hides the reachability analysis which has been carried out. Therefore, we take a closer look at it in case that \( p \) has no points in \( \mathbb{R}^n \). Then, \((26.b)\) becomes

\[
\begin{align*}
\frac{d\xi}{\mathbb{R}^n} = \xi(t, t^0) dt + \frac{1}{\mathbb{R}^n} g(t, t^0) d\xi \left( \mathbb{R}^n \times \mathbb{R}^n \right) \\
\end{align*}
\]

Moreover, to avoid the use of equations \((26.a, c)\), we go over to the common descriptive way of formulating \((t, t^0)\) and \((t, \) \).

\( \{t, \} \) is a multivariate counting process characterized by the \( \mathbb{R}^n \)-predictable intensity, \( \gamma(t, t^0) \).

\[
\begin{align*}
\gamma(t, t^0) = A(t, t^0) \left[ 1 + F(t) \right] \frac{1}{\mathbb{R}^n} / \mathbb{R}^n \}
\end{align*}
\]

and a deterministic jump measure \( d\mu(t, t^0) \).

\( \{t, \} \) is a countable state Markov process in a half-space \( \mathbb{R}^n \), with \( \mathbb{R}^n \)-predictable rate, \( r(t, t^0) \), of jumping from \( t(t-1) \) to \( t(t^0) \).

\[
\begin{align*}
\frac{1}{\mathbb{R}^n} \left[ 1 + F(t) \right] \frac{1}{\mathbb{R}^n} / \mathbb{R}^n \}
\end{align*}
\]

while \( \{t, \} \) remains \( \gamma \).

From this formulation, we easily notice the interesting effect that \( t(t^0) \) appears in the coefficient \( \xi(t) \) of \((27.a)\)'s third right hand term. This means that \( \xi(t, t^0, t) \) anticipates the switching from \( t(t-1) \) to \( t(t^0) \), and thus a jump of \( \{t, \} \).

Verify that the anticipating coefficient \( \xi(t, t^0) \) already appears in \((26.a, b, c)\), while there is no anticipating coefficient in equation \((23.a, b, c)\). As the solutions of both equations are indistinguishable, we conclude that \((23.a, b, c)\) is the canonical representation of a system with hybrid jumps, while \((26.a, b, c)\), with the anticipating coefficient, is the representation that is more useful when it comes to the realization of Markov models with hybrid jumps.

Remark: If \( \{t, \} \) is \( \{t, \} \)-invariant, then \( \{t, \} \) is a countable state Markov process. In this case \((27.a)\) can straightforwardly be obtained from a classical system like \((1)\) of which all coefficients are continuous. For the situation that \( \{t, \} \) is continuous, i.e. \( \gamma = 0 \), see Brockett and Blankenship (1977).

For some applications with hybrid jumps, i.e. \( g(t, t^0) \) (S. Vermersch (1977)), Bon (1977) and Mariton (1987).

Acknowledgement

The authors are grateful to Professor Yaakov Bar-Shalom for stimulating discussions and his hospitality at the University of Connecticut.

References


