

Active Estimation for Jump Markov Linear Systems

Lars Blackmore*, Senthooan Rajamanoharan and Brian C. Williams

Abstract— Jump Markov Linear Systems are convenient models for systems that exhibit both continuous dynamics and discrete mode changes. Estimating the hybrid discrete-continuous state of these systems is important for control and fault detection. Existing solutions for hybrid estimation approximate the belief state by maintaining a subset of the possible discrete mode sequences. This approximation can cause the estimator to lose track of the true mode sequence when the effects of discrete mode changes are subtle.

In this paper we present a method for *active* hybrid estimation, where control inputs can be designed to discriminate between possible mode sequences. By probing the system for the purposes of estimation, such a sequence of control inputs can greatly reduce the probability of losing the true mode sequence compared to a nominal control sequence. Furthermore, by using a constrained finite horizon optimization formulation, we are able to guarantee that a given control task is achieved, while optimally detecting the hybrid state.

In order to achieve this, we present three main contributions. First, we develop a method by which a sequence of control inputs is designed in order to discriminate optimally between a finite number of linear dynamic system models. These control inputs minimize a novel, tractable upper bound on the probability of model selection error. Second, we extend this approach to develop an active estimation method for Jump Markov Linear Systems by relating the probability of model selection error to the probability of losing the true mode sequence. Finally, we make this method tractable using a principled pruning technique.

Simulation results show that the new method applied to an aircraft fault detection problem significantly decreases the probability of a hybrid estimator losing the true mode sequence.

I. INTRODUCTION

STOCHASTIC hybrid discrete-continuous models have been used to represent a large number of physical and biological systems, from Mars rovers to dancing bees[1], [2], [3], [4]. In these models, the system dynamics depend on which discrete mode the system is in, and discrete mode transitions occur stochastically. Typically the continuous and discrete state is only partially observable, which means that estimation of the hybrid system state is a challenging problem.

*Corresponding author. L. Blackmore is with the NASA Jet Propulsion Laboratory, California Institute of Technology, 4800 Oak Grove Drive, Pasadena, CA, 91109 USA e-mail: lars@jpl.nasa.gov.

S. Rajamanoharan is with the Theoretical Physics Department, Cambridge University, Cambridge, UK.

B. Williams is with the Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA.

The work described in this paper was performed at the Massachusetts Institute of Technology, sponsored by NASA Award NNA04CK91A. The writing and publication of this paper were supported by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

However since tasks such as robot fault detection and pilot intent recognition can be posed as hybrid state estimation problems, it is a topic of great interest.

Exact state estimation in such systems is, in general, intractable[5]. A number of tractable algorithms have been proposed that approximate the true belief state[6], [7], [8]. One common approach is to store a finite subset of the possible discrete mode sequences[9], [10]. However, by approximating the true belief state it is possible to lose track of the true mode sequence, at which point the estimator diverges. Previous work has highlighted this problem and suggested a number of solutions, for example [2], [9], [11], [12], [13], [14].

These approaches are ‘passive’ in the sense that they attempt to do the best possible with the observations that are made available during nominal operation. In many cases, however, it is possible to obtain a great deal more information about the state of a hybrid system by issuing appropriate control inputs. For example, in the case of detecting a fault in a drive motor, a change in the motor dynamics will not be apparent in the observations unless some effort is requested from that motor.

In this paper we introduce an *active* hybrid estimation approach that generates control inputs to minimize the probability of the estimator losing the true mode sequence. This approach applies to Jump Markov Linear Systems; here the system is described by a discrete-time stochastic linear dynamic model whose parameters depend on the discrete mode. The system switches at random between modes; the discrete mode is governed by a Markov process. Jump Markov Linear Systems are an important class of hybrid discrete-continuous systems that have been used in a number of applications, for example [3], [4]. In order to develop the active hybrid estimation capability approach we provide three main technical contributions.

First, we develop a method by which a finite, constrained sequence of control inputs is designed in order to discriminate optimally between a finite number of linear dynamic system models. The problem of control design for model discrimination has received a great deal of attention[15], [16], [17], [18], [19], [20], [21]. Typical approaches, for example [16], [18], [19], design *auxiliary* signals that are added to the nominal control signal for the purpose of model discrimination. The auxiliary signal has low power so that its effect on the system state is limited. This, however, also restricts the discrimination power of the signal. Our new approach, by contrast, designs control inputs with respect to hard constraints. These constraints can be used to ensure that a certain task, defined in terms of the system state, is fulfilled, or that hard constraints

such as actuator saturation are not violated. By optimizing subject to hard constraints, the method can generate signals that are far more effective in discrimination than a limited power auxiliary signal. In addition, control inputs are chosen from a continuous set. This is in contrast with approaches such as [21] that choose control inputs from a finite set.

Previous approaches to the problem of control design for model discrimination have suggested a number of different criteria for optimization; for example information gain, or the distance between the observation distributions conditioned on different models[18], [19]. These criteria typically do not have a meaningful interpretation in the context of the model selection problem. By contrast, consistent with a Bayesian approach, we use an upper bound on the probability of model selection error as the optimization criterion. While the probability of model selection error cannot be calculated in closed form, we derive a novel upper bound on this value that applies to an arbitrary number of models. This extends existing bounds that apply to selection between only two models[22]. Then we pose the problem of designing a finite sequence of control inputs to minimize this bound, subject to constraints, as a finite horizon trajectory design problem. We show that in the case of linear constraints this problem can be solved using existing nonlinear optimization techniques such as Sequential Quadratic Programming[23].

Our second contribution is to extend this multiple-model discrimination method to develop an active estimation capability for Jump Markov Linear Systems. The key insight is that for a given discrete mode sequence, the system dynamics, although time-varying, are fully known. By extending the error bound derived for discrimination between different models to time-varying systems, we create a tractable upper bound on the probability of a hybrid estimator losing track of the true mode sequence. We then use a constrained finite horizon control design approach to ensure that a given control task is achieved, conditioned on nominal system operation.

A finite horizon control approach such as this suffers from the fact that the number of possible mode sequences is exponential in the number of discrete modes and in the length of the design horizon. In practice this means that an active hybrid estimation approach can only consider a subset of the possible mode sequences. Our third contribution is therefore to introduce an efficient pruning method. This method ensures that the control design only takes into account mode sequences that are *a priori* likely to contribute to the probability of losing the true mode sequence. The result is a tractable optimization problem that can be solved using Sequential Quadratic Programming[23], for example.

The paper is organized as follows. In Section II we derive the new method for multiple-model discrimination. In Section III we extend this to Jump Markov Linear Systems. In section IV we demonstrate the multiple-model discrimination approach in simulation using an aircraft fault detection scenario, and show empirically that the new method significantly reduces the probability of model selection error. Finally, in Section V we demonstrate the active estimation approach for Jump Markov Linear Systems and show that the new method reduces the probability of losing the true mode sequence.

II. FINITE HORIZON CONTROL DESIGN FOR OPTIMAL MODEL DISCRIMINATION

A. Problem Statement

In this section we consider the linear discrete-time dynamic system described by:

$$\begin{aligned}\mathbf{x}_{c,\tau+1} &= A\mathbf{x}_{c,\tau} + B\mathbf{u}_\tau + \omega_\tau \\ \mathbf{y}_{\tau+1} &= C\mathbf{x}_{c,\tau+1} + D\mathbf{u}_\tau + \nu_\tau,\end{aligned}\quad (1)$$

where $\mathbf{x}_c \in \mathbb{R}^{n_x}$ is the system state and $\mathbf{y} \in \mathbb{R}^{n_y}$ are the observations. The variables $\omega \in \mathbb{R}^{n_x}$ and $\nu \in \mathbb{R}^{n_y}$ are process and observation noise, respectively, which we restrict to be zero-mean, Gaussian white noise with covariance Q and R , respectively. The initial distribution $p(\mathbf{x}_{c,0})$ is a Gaussian random variable with mean $\bar{\mathbf{x}}_0$ and covariance V , and is uncorrelated with the noise variables.

Definition 1. We use subscript notation to denote the value of a variable at a given time, e.g. \mathbf{x}_τ is the value of \mathbf{x} at time step τ , where τ is an integer. We use \mathbf{x}' to denote the transpose of \mathbf{x} . We use $\mathbf{x}_{0:N}$ to denote the finite sequence $\langle \mathbf{x}_0, \dots, \mathbf{x}_N \rangle$. We use $P(A)$ to denote the probability of event A . If the variable \mathbf{x} takes continuous values, we use $p(\mathbf{x})$ to denote its probability density function. If \mathbf{x} takes discrete values, $p(\mathbf{x})$ denotes its probability mass function. We use $|M|$ to denote the determinant when M is a matrix, and the number of members of a set when M is a set.

In the Multiple Model selection problem, the parameters $\{A, B, C, D, Q, R, \bar{\mathbf{x}}_0, V\}$ are unknown, but we assume that they take values from a finite set \mathcal{H} of ‘models’ or ‘hypotheses’. Under model $H_i \in \mathcal{H}$, the system parameters are $\{A(i), B(i), C(i), D(i), Q(i), R(i), \bar{\mathbf{x}}_0(i), V(i)\}$. We assume that for each i the parameters are fully known and that the true model persists indefinitely. In other words, in this section we do not allow switching between the different models; switching dynamics are considered in Section III. Multiple Model selection uses Bayesian hypothesis selection to determine the most likely system parameters given a sequence of observations $\mathbf{y}_{0:h}$, a sequence of control inputs $\mathbf{u}_{0:h-1}$ and a prior distribution over models. We review Multiple Model selection in Section II-B. In this paper we aim to design control inputs to aid model selection. The problem is stated formally as follows:

Definition 2. At time step zero, given a set of models \mathcal{H} and a prior probability for each model $H_i \in \mathcal{H}$, the Optimal Model Discrimination Problem consists of designing a finite sequence of control inputs $\mathbf{u}_{0:h-1}$ that minimizes the probability of selection error when using Bayesian model selection.

For notational simplicity we assume, without loss of generality, that optimal model discrimination is invoked at time step zero. In many scenarios it may be useful to start optimal discrimination at some later time τ , for example when there is a high level of uncertainty about which model is the true one. In this case the prior probability of each model in Def. 2 is replaced by the probability (or *belief state*) of each model given the observations $\mathbf{y}_{0:\tau}$. This belief state is generated online using a Multiple Model estimation scheme, for example [24], [25], [26], [27], [28].

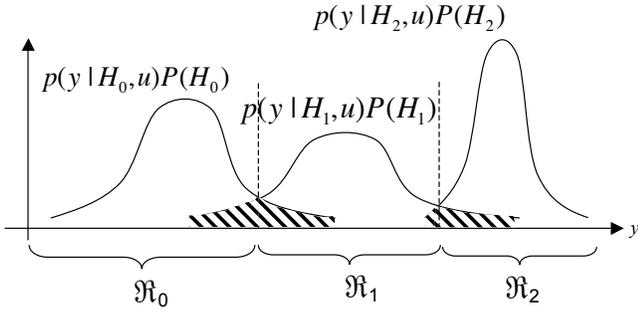


Fig. 1. Bayesian hypothesis selection between multiple models given an observation y , a control input u , and a prior [22]. In general, Bayesian selection between multiple hypotheses yields a number of *decision regions* in the space of possible observations; in each decision region a particular hypothesis is most likely. If the observation y falls into set \mathfrak{R}_i then the classifier selects H_i . Even with Bayes optimal selection there is a finite probability of error given by the Bayes Risk, denoted by the shaded region.

In Section II-B we define Bayesian model selection and the probability of model selection error. In Sections II-C and II-D we show that, while the probability of model selection error cannot be evaluated in closed form, it can be upper bounded. In our approach to solving the Optimal Model Discrimination Problem (Def. 2) we therefore minimize an upper bound on the model selection error. The approach is described in detail in Section II-E.

B. Hypothesis Selection and Bayes Risk

Bayesian hypothesis selection between an arbitrary number of models, given a general vector of observations \mathbb{Y} and a vector of control inputs \mathbb{U} , can be expressed as follows:

Select H_i where $i = \arg \max_j P(H_j | \mathbb{Y}, \mathbb{U})$.

Using Bayes' rule, this selection is equivalently given by:

Select H_i where $i = \arg \max_j p(\mathbb{Y} | H_j, \mathbb{U}) P(H_j)$.

The term $P(H_j)$ represents the prior probability of model j . As shown in Figure 1, Bayesian selection yields a number of decision regions \mathfrak{R}_i , in which H_i is the most probable hypothesis:

$$\mathfrak{R}_i = \{ \mathbb{Y} | p(H_i, \mathbb{Y} | \mathbb{U}) > p(H_l, \mathbb{Y} | \mathbb{U}) \quad \forall l \neq i \}. \quad (2)$$

Model i is selected if the observation \mathbb{Y} falls in region \mathfrak{R}_i . This Bayesian selection rule minimizes the likelihood of selecting an incorrect model given the available information. As shown in Figure 1, the Bayesian optimal classifier has a finite probability of selecting the incorrect model, known as the *Bayes Risk*. The Bayes risk is given by:

$$\begin{aligned} P(\text{error}) &= \sum_i \sum_{j \neq i} P(\mathbb{Y} \in \mathfrak{R}_j, H_i | \mathbb{U}) \\ &= \sum_i \sum_{j \neq i} P(\mathbb{Y} \in \mathfrak{R}_j | H_i, \mathbb{U}) P(H_i) \\ &= \sum_i \sum_{j \neq i} \int_{\mathfrak{R}_j} p(\mathbb{Y} | H_i, \mathbb{U}) P(H_i) d\mathbb{Y}. \end{aligned} \quad (3)$$

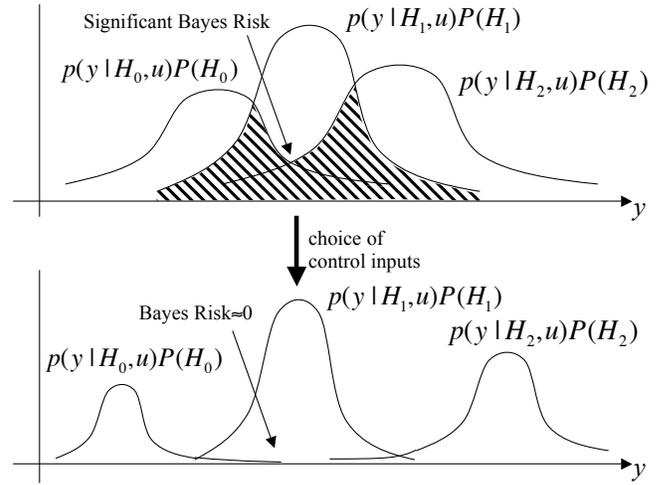


Fig. 2. Graph showing $p(y|H_0, u)$, $p(y|H_1, u)$ and $p(y|H_2, u)$ for two different choices of u . In the upper figure, the predicted distributions overlap significantly, leading to a large Bayes risk. In the lower figure, a different selection of u has separated the distributions, meaning that when the observation y is made, the correct model can be selected with high confidence. The Bayes risk is very low, meaning that the probability of error is very low.

Since the Bayes Risk is the probability of error when using the optimal classifier, we would like to optimize our control inputs to the system to minimize this measure. The effect of input choice on the Bayes Risk is illustrated in Figure 2.

C. Bounding the Bayes Risk for Two Models

The Bayes Risk is unsuitable as an optimization criterion, since the finite integral in (3) cannot, in general, be evaluated in closed form. It is, however, possible to bound the Bayes Risk in closed form. For the special case of two models H_0 and H_1 , the *Battacharyya Bound* [22] applies, and we show in this section that this leads to a quadratic cost function for model discrimination. In Section II-D we develop a novel bound that applies to more than two models.

The Battacharyya Bound is given by the integral:

$$P(\text{error}) \leq P(H_0)^{\frac{1}{2}} P(H_1)^{\frac{1}{2}} \int \sqrt{p(\mathbb{Y} | H_0) p(\mathbb{Y} | H_1)} d\mathbb{Y}. \quad (4)$$

If the distributions are Gaussian such that $p(\mathbb{Y} | H_0)$ has mean $\mu(0)$ and variance $\Sigma(0)$, and $p(\mathbb{Y} | H_1)$ has mean $\mu(1)$ and variance $\Sigma(1)$, the above value can be calculated analytically to give:

$$P(\text{error}) \leq P(H_0)^{\frac{1}{2}} P(H_1)^{\frac{1}{2}} \exp\{-k\},$$

where:

$$\begin{aligned} k &= \frac{1}{4} (\mu(1) - \mu(0))' [\Sigma(0) + \Sigma(1)]^{-1} (\mu(1) - \mu(0)) \\ &\quad + \frac{1}{2} \ln \frac{|\frac{\Sigma(0) + \Sigma(1)}{2}|}{\sqrt{|\Sigma(0)| |\Sigma(1)|}}. \end{aligned} \quad (5)$$

Since the logarithm is a monotonically increasing function, the value of x that optimizes $f(x)$ is also the value that optimizes $\ln[f(x)]$. We therefore take the logarithm of the

Battacharyya bound for Gaussian distributions to yield the following cost function:

$$J = \frac{1}{2} \ln[P(H_0)P(H_1)] - \frac{1}{2} \ln \frac{|\frac{\Sigma(0)+\Sigma(1)}{2}|}{\sqrt{|\Sigma(0)||\Sigma(1)|}} - \frac{1}{4}(\mu(1) - \mu(0))' [\Sigma(0) + \Sigma(1)]^{-1} (\mu(1) - \mu(0)). \quad (6)$$

In Section II-E we use (6) to perform finite horizon control design for discrimination between two models. This use of the Battacharyya Bound in model discrimination is novel, and furthermore the criterion is different from existing criteria for optimal model discrimination that have been proposed by other authors. The most similar criterion is the Kullback-Leibler (KL) divergence, which was used by [18]. The KL divergence from the distribution $\mathcal{N}(\mu(0), \Sigma(0))$ to the distribution $\mathcal{N}(\mu(1), \Sigma(1))$ is given by:

$$\frac{1}{2} \ln \left(\frac{|\Sigma(1)|}{|\Sigma(0)|} \right) + \frac{1}{2} \text{tr}(\Sigma(1)^{-1} \Sigma(0)) - \frac{N}{2} + \frac{1}{2} (\mu(1) - \mu(0))' \Sigma(1)^{-1} (\mu(1) - \mu(0)), \quad (7)$$

where N is the dimensionality of the distribution. The ‘symmetrized’ KL divergence between two distributions is:

$$\frac{1}{2} \text{tr}(\Sigma(1)^{-1} \Sigma(0)) + \frac{1}{2} \text{tr}(\Sigma(0)^{-1} \Sigma(1)) - 2N + (\mu(1) - \mu(0))' [\Sigma(1)^{-1} + \Sigma(0)^{-1}] (\mu(1) - \mu(0)). \quad (8)$$

Clearly, both of these criteria are different from (6).

D. Bounding the Bayes Risk for Multiple Models

In this section we introduce a new bound on the probability of model selection error that applies to an arbitrary number of models, or hypotheses.

Theorem 1. *When performing hypothesis selection between an arbitrary number of hypotheses, for Gaussian observation distributions such that $p(\mathbb{Y}|H_i) = \mathcal{N}(\mu(i), \Sigma(i))$ and $p(\mathbb{Y}|H_j) = \mathcal{N}(\mu(j), \Sigma(j))$, the Bayes Risk is upper bounded as follows:*

$$P(\text{error}) \leq \sum_i \sum_{j>i} P(H_i)^{\frac{1}{2}} P(H_j)^{\frac{1}{2}} e^{-k(i,j)}, \quad (9)$$

where:

$$k(i,j) = \frac{1}{4} (\mu(j) - \mu(i))' [\Sigma(i) + \Sigma(j)]^{-1} (\mu(j) - \mu(i)) + \frac{1}{2} \ln \frac{|\frac{\Sigma(i)+\Sigma(j)}{2}|}{\sqrt{|\Sigma(i)||\Sigma(j)|}}. \quad (10)$$

Proof: The key idea is to express the n -hypothesis Bayes Risk in terms of the Bayes Risk between pairs of hypotheses, and then to bound the term for each pair in a manner analogous to the Battacharyya bound. The n -hypothesis Bayes Risk given by (3) can be expressed in terms of hypothesis pairs as follows:

$$P(\text{error}) = \sum_i \sum_{j \neq i} P(\mathbb{Y} \in \mathfrak{R}_j, H_i | \mathbb{U}) = \sum_i \sum_{j>i} \left(P(\mathbb{Y} \in \mathfrak{R}_j, H_i | \mathbb{U}) + P(\mathbb{Y} \in \mathfrak{R}_i, H_j | \mathbb{U}) \right). \quad (11)$$

Here, each term in the summation is the Bayes Risk between hypothesis i and hypothesis j . This can be written exactly as:

$$\begin{aligned} & \left(P(\mathbb{Y} \in \mathfrak{R}_j, H_i | \mathbb{U}) + P(\mathbb{Y} \in \mathfrak{R}_i, H_j | \mathbb{U}) \right) \\ &= \int_{\mathfrak{R}_j} p(\mathbb{Y}|H_i, \mathbb{U}) P(H_i) d\mathbb{Y} + \int_{\mathfrak{R}_i} p(\mathbb{Y}|H_j, \mathbb{U}) P(H_j) d\mathbb{Y}. \end{aligned} \quad (12)$$

We now define two regions \mathfrak{R}_A and \mathfrak{R}_B as follows:

$$\begin{aligned} \mathfrak{R}_A &= \{ \mathbb{Y} | p(H_i, \mathbb{Y} | \mathbb{U}) > p(H_j, \mathbb{Y} | \mathbb{U}) \} \\ \mathfrak{R}_B &= \{ \mathbb{Y} | p(H_j, \mathbb{Y} | \mathbb{U}) > p(H_i, \mathbb{Y} | \mathbb{U}) \}. \end{aligned} \quad (13)$$

We can relate these regions to the decision regions \mathfrak{R}_i and \mathfrak{R}_j . From the definition of hypothesis selection given in Section II-B, decision region \mathfrak{R}_i is where the likelihood of hypothesis i is greater than all other hypotheses:

$$\begin{aligned} \mathfrak{R}_i &= \{ \mathbb{Y} | p(H_i, \mathbb{Y} | \mathbb{U}) > p(H_l, \mathbb{Y} | \mathbb{U}) \quad \forall l \neq i \} \\ \mathfrak{R}_j &= \{ \mathbb{Y} | p(H_j, \mathbb{Y} | \mathbb{U}) > p(H_l, \mathbb{Y} | \mathbb{U}) \quad \forall l \neq j \}. \end{aligned} \quad (14)$$

It is clear from (13) and (14) that the decision regions are subsets of the regions \mathfrak{R}_A and \mathfrak{R}_B , such that $\mathfrak{R}_i \subseteq \mathfrak{R}_A$ and $\mathfrak{R}_j \subseteq \mathfrak{R}_B$. We can therefore bound the 2-hypothesis Bayes Risk term in (12) as follows:

$$\begin{aligned} & \int_{\mathfrak{R}_j} p(\mathbb{Y}|H_i, \mathbb{U}) P(H_i) d\mathbb{Y} + \int_{\mathfrak{R}_i} p(\mathbb{Y}|H_j, \mathbb{U}) P(H_j) d\mathbb{Y} \leq \\ & \int_{\mathfrak{R}_B} p(\mathbb{Y}|H_i, \mathbb{U}) P(H_i) d\mathbb{Y} + \int_{\mathfrak{R}_A} p(\mathbb{Y}|H_j, \mathbb{U}) P(H_j) d\mathbb{Y}. \end{aligned} \quad (15)$$

Note that the union of the regions \mathfrak{R}_A and \mathfrak{R}_B is the entire space of \mathbb{Y} . Using the definitions in (13), we can therefore write the integrals in (15) as a single integral over the entire space of \mathbb{Y} :

$$\begin{aligned} & \int_{\mathfrak{R}_B} p(\mathbb{Y}|H_i, \mathbb{U}) P(H_i) d\mathbb{Y} + \int_{\mathfrak{R}_A} p(\mathbb{Y}|H_j, \mathbb{U}) P(H_j) d\mathbb{Y} \\ &= \int_{\mathbb{Y}} \min\{p(\mathbb{Y}|H_i, \mathbb{U}) P(H_i), p(\mathbb{Y}|H_j, \mathbb{U}) P(H_j)\} d\mathbb{Y}. \end{aligned} \quad (16)$$

We now use the following inequality[22]:

$$\min\{a, b\} \leq a^{\frac{1}{2}} b^{\frac{1}{2}}, \quad (17)$$

which means that the integral over \mathbb{Y} can be bounded as follows:

$$\begin{aligned} & \int_{\mathbb{Y}} \min\{p(\mathbb{Y}|H_i, \mathbb{U}) P(H_i), p(\mathbb{Y}|H_j, \mathbb{U}) P(H_j)\} d\mathbb{Y} \\ & \leq P(H_i)^{\frac{1}{2}} P(H_j)^{\frac{1}{2}} \int_{\mathbb{Y}} p^{\frac{1}{2}}(\mathbb{Y}|H_i, \mathbb{U}) p^{\frac{1}{2}}(\mathbb{Y}|H_j, \mathbb{U}) d\mathbb{Y}. \end{aligned} \quad (18)$$

Furthermore, if the distributions are Gaussian, such that $p(\mathbb{Y}|H_i) = \mathcal{N}(\mu(i), \Sigma(i))$ and $p(\mathbb{Y}|H_j) = \mathcal{N}(\mu(j), \Sigma(j))$, then this integral can be evaluated in closed form to give:

$$\int_{\mathbb{Y}} p^{\frac{1}{2}}(\mathbb{Y}|H_i, \mathbb{U}) p^{\frac{1}{2}}(\mathbb{Y}|H_j, \mathbb{U}) d\mathbb{Y} = e^{-k(i,j)}, \quad (19)$$

where $k(i, j)$ is defined in (10). We therefore have:

$$\begin{aligned} & P(\mathbb{Y} \in \mathfrak{R}_j, H_i | \mathbb{U}) + P(\mathbb{Y} \in \mathfrak{R}_i, H_j | \mathbb{U}) \\ & \leq P(H_i)^{\frac{1}{2}} P(H_j)^{\frac{1}{2}} e^{-k(i,j)}, \end{aligned} \quad (20)$$

which leads to a new upper bound on the probability of model selection error:

$$P(\text{error}) \leq \sum_i \sum_{j>i} P(H_i)^{\frac{1}{2}} P(H_j)^{\frac{1}{2}} e^{-k(i,j)}, \quad (21)$$

where $k(i, j)$ is defined in (10). \square

We have therefore introduced a new upper bound on the probability of model selection error between an arbitrary number of models, which can be evaluated in closed form. Notice that in the special case of only two hypotheses, the bound reduces to the Battacharyya bound mentioned in Section II-C. We discuss the properties of this bound in Appendix I. In Section II-E we describe how the new bound can be used as an optimization criterion for finite horizon control design.

E. Finite Horizon Formulation

In this section we pose the problem of model discrimination between an arbitrary number of models as a finite horizon trajectory design problem, and show that it can be solved using Sequential Quadratic Programming (SQP)[23]. For the special case of two models, the optimization problem can be solved using Quadratic Programming (QP) to global optimality.

1) *A Tractable Cost Function for Model Discrimination:* Optimal Model Discrimination Problem (Def. 2) consists of planning a finite sequence of inputs in order to minimize the probability of error. This form of planning is known as *finite horizon* planning. In this case, if the horizon is of length h , we are concerned with a sequence of observations $\mathbf{y}_1, \dots, \mathbf{y}_h$ and a sequence of inputs $\mathbf{u}_0, \dots, \mathbf{u}_{h-1}$. Since evaluation of the probability of error is intractable, we instead minimize an upper bound on the probability of error, described in Section II-D. In this section we describe how this bound can be used in a finite-horizon context. We define:

$$\begin{aligned} \mathbb{Y} &= [\mathbf{y}_1' \quad \mathbf{y}_2' \quad \dots \quad \mathbf{y}_h']' \\ \mathbb{U} &= [\mathbf{u}_0' \quad \mathbf{u}_1' \quad \dots \quad \mathbf{u}_{h-1}']'. \end{aligned} \quad (22)$$

We assume that model selection is carried out by a Bayes optimal classifier that makes its decision based on all h observations within the horizon. Under the assumptions in Section II-A, \mathbf{y}_τ is a normally distributed random variable given a sequence of inputs \mathbb{U} and given a model H_l . We define $\mu_\tau(l)$ and $\Sigma_\tau(l)$ for time steps $\tau = 1, \dots, h$ and models H_l such that:

$$p(\mathbf{y}_\tau | H_l, \mathbb{U}) = \mathcal{N}(\mu_\tau(l), \Sigma_\tau(l)). \quad (23)$$

Then the block vector of all observations $\mathbb{Y} = [\mathbf{y}'_1, \dots, \mathbf{y}'_h]'$ is a vector of normally distributed random variables, given a sequence of inputs and a model. We define $\mu(l)$ and $\Sigma(l)$ to be the mean and covariance of the vector of all observations such that:

$$p(\mathbb{Y} | H_l, \mathbb{U}) = \mathcal{N}(\mu(l), \Sigma(l)). \quad (24)$$

From the above definitions the distribution of $\mathbb{Y}(l)$ is given by:

$$\mu(l) = [\mu_1(l)', \dots, \mu_h(l)']' \quad (25)$$

$$[\Sigma(l)]_{i,j} = E \left[([\mathbb{Y}]_i - [\mu(l)]_i) ([\mathbb{Y}]_j - [\mu(l)]_j) | H_l \right]. \quad (26)$$

Here $[\cdot]_i$ denotes the value at index i into the vector, and similarly $[\cdot]_{i,j}$ denotes the value at index (i, j) into the matrix.

Having defined $\mu(l)$ and $\Sigma(l)$ for all l , the bound given in (9) provides an upper bound for the probability of error when using the entire sequence of observations from time 1 to time h . We therefore use this bound as a cost function on the finite horizon optimization formulation:

$$J = \sum_i \sum_{j>i} P(H_i)^{\frac{1}{2}} P(H_j)^{\frac{1}{2}} e^{-k(i,j)}, \quad (27)$$

where $k(i, j)$ is defined in (10).

Given a model H_l , the system equations (1) are fully known. Hence the distribution $p(\mathbb{Y} | H_l, \mathbb{U})$ can be calculated for all l . Applying the system equations (1) recursively we have:

$$\begin{aligned} \mathbf{y}_\tau(l) &= C(l)A(l)^\tau \mathbf{x}_0 + D(l)\mathbf{u}_{\tau-1} + \nu_{\tau-1} \\ &+ C(l) \sum_{\gamma=0}^{\tau-1} A(l)^{\tau-\gamma-1} (B(l)\mathbf{u}_\gamma + \omega_\gamma). \end{aligned} \quad (28)$$

Using (28) we can derive explicit expressions for the mean $\mu(h)$ and covariance $\Sigma(h)$ as defined in (25) and (26). First, define:

$$f = n_y(p-1) + q, \quad (29)$$

where:

$$1 \leq q \leq n_y \quad 1 \leq p \leq h \quad p, q \in \mathbb{Z}. \quad (30)$$

Then following from (28):

$$\begin{aligned} [\mu(l)]_f &= [\mu_p(l)]_q = \left[C(l)A(l)^p \bar{\mathbf{x}}_0(l) + D(l)u_{p-1} \right. \\ &\left. + C(l) \sum_{\gamma=0}^{p-1} A(l)^{p-\gamma-1} B(l)u_\gamma \right]_q \end{aligned} \quad (31)$$

Defining $g = n_y(r-1) + s$ in a similar manner to (29), the expression for the covariance is:

$$\begin{aligned} [\Sigma(l)]_{f,g} &= [\mathcal{R}(l)(p, r) + C(l)A(l)^p V(l)A(l)^{r'} C(l)']_{q,s} \\ &+ \left[\sum_{\gamma=0}^{m-1} C(l)A(l)^{(p-\gamma-1)} Q(l)A(l)^{(r-\gamma-1)} C(l)' \right]_{q,s} \end{aligned} \quad (32)$$

where $m = \min\{p, r\}$ and:

$$\mathcal{R}(l)(p, r) = \begin{cases} R(l) & p = r \\ 0 & p \neq r. \end{cases} \quad (33)$$

The distribution of \mathbb{Y} has two important properties:

- The equation for the mean of the predicted distribution of \mathbb{Y} is linear in the control inputs \mathbb{U} .
- The covariance of the predicted distribution of \mathbb{Y} is not a function of the control inputs \mathbb{U} .

These properties mean that the multiple model criterion in (27) has a tractable form, enabling it to be used in a constrained optimization formulation. Furthermore, for the two-model case, these properties mean that the criterion in (6) can be simplified to:

$$J' = -(\mu(1) - \mu(0))' [\Sigma(0) + \Sigma(1)]^{-1} (\mu(1) - \mu(0)). \quad (34)$$

Since both $\mu(1)$ and $\mu(0)$ are linear functions of \mathbb{U} , (34) is quadratic in the control inputs \mathbb{U} . This means that the two-model discrimination problem subject to linear constraints (Section II-E.2) can be solved using Quadratic Programming.

Remark 1. *Since the covariance matrices $\Sigma(0)$ and $\Sigma(1)$ are positive definite, the cost function given by (34) is a concave function of $(\mu(1) - \mu(0))$. Both $\mu(1)$ and $\mu(0)$ are linear functions of \mathbb{U} , and hence (34) is also a concave function of the control inputs \mathbb{U} . This concavity makes (34) a particularly tractable cost function for optimization, and guarantees that a global optimum can be found in bounded time [29].*

2) *Linear Constraints:* A powerful aspect of the constrained finite horizon formulation is that optimal input sequences can be found subject to hard constraints. This can be used to model actuator saturation, for example, by constraining $u_{min} < u_\tau < u_{max}$. The expected system state conditioned on a model H_l is a linear function of the control inputs:

$$E[\mathbf{x}_\tau | H_l] = A(l)^\tau \bar{\mathbf{x}}_0 + \sum_{j=0}^{\tau-1} A(l)^{\tau-j-1} (B(l) \mathbf{u}_j). \quad (35)$$

Hence constraints on the expected system state of the form $E[\mathbf{x}_\tau | H_l] = goal$ or $\mathbf{x}_{min} < E[\mathbf{x}_\tau | H_l] < \mathbf{x}_{max}$ are linear constraints in the control inputs. By imposing such constraints, we can:

- Ensure that a certain task, defined in terms of the expected system state, is fulfilled
- Ensure that the expected system state stays within a ‘safe’ operating region or within a valid linearization region
- Ensure that the system state ends the experiment in the same region as it started.

Here, we have restricted our attention to linear constraints, since these are straightforward to encode, and are guaranteed to be convex; convexity simplifies the optimization problem greatly. The general formulation, however, applies to nonlinear constraints.

3) *Summary:* We have shown that the problem of designing a sequence of optimal control inputs to discriminate between an arbitrary number of models can be posed as a finite-horizon trajectory design problem. The resulting AE-MM algorithm, summarized in Table I, works by minimizing a novel, closed form upper bound on the probability of model selection error, and imposing constraints on the expected system state and control inputs to ensure that a defined task is fulfilled and that actuator limits are not violated. In this sense, the approach uses constraints to perform control, while optimizing with regard to estimation. The optimization can be solved using existing methods such as Sequential Quadratic Programming. In the case of discrimination between two models, the optimization can be solved using Quadratic Programming; furthermore in this case the global optimum can be found in finite time.

III. ACTIVE ESTIMATION FOR JUMP MARKOV LINEAR SYSTEMS

We now extend the multiple-model discrimination method to develop an active estimation capability for Jump Markov Linear Systems, which we call the AE-JMLS algorithm. By

Function	ACTIVE MULTIPLE MODEL MAIN($\mathcal{H}, p(\mathcal{H}), h$)	returns
	$\mathbf{u}_{0:h-1}$	
1)	For each model H_i , calculate the mean $\mu(h)$ as a linear function of the control inputs, according to (31).	
2)	For each H_i calculate the covariance $\Sigma(h)$ according to (32).	
3)	Using SQP, minimize over $\mathbf{u}_{0:h-1}$:	
	$\sum_i \sum_{j>i} P(H_i)^{\frac{1}{2}} P(H_j)^{\frac{1}{2}} e^{-k(i,j)}, \quad (36)$	
	where $k(i, j)$ is defined in (10), in terms of $\mu(h)$ and $\Sigma(h)$, subject to:	
	<ul style="list-style-type: none"> • Constraints on the expected state, for example $\mu_\tau(l) \leq \mu_{max}$ or $\mu_\tau(l) = \mu_{eq}$. • Constraints on the control inputs, for example $\mathbf{u}_\tau \leq \mathbf{u}_{max}$. 	

TABLE I. AE-MM Algorithm for Optimal Discrimination between Multiple Linear Models.

extending the error bound derived for discrimination between different models, to time-varying systems, we create a closed form upper bound on the probability of the true mode sequence being pruned. We then use a constrained finite horizon control design approach to minimize this bound, while ensuring that a given control task is achieved.

A. Problem Statement

In this section we define a Jump Markov Linear System (JMLS) and describe how approximate state estimation can be carried out for such systems. The continuous dynamics of a JMLS \mathcal{M} are defined by:

$$\begin{aligned} \mathbf{x}_{c,\tau+1} &= A(\mathbf{x}_{d,\tau}) \mathbf{x}_{c,\tau} + B(\mathbf{x}_{d,\tau}) \mathbf{u}_\tau + \omega_\tau \\ \mathbf{y}_{\tau+1} &= C(\mathbf{x}_{d,\tau}) \mathbf{x}_{c,\tau+1} + D(\mathbf{x}_{d,\tau}) \mathbf{u}_\tau + \nu_\tau, \end{aligned} \quad (37)$$

where $\mathbf{x}_c \in \mathbb{R}^{n_x}$ is the continuous system state and $\mathbf{y} \in \mathbb{R}^{n_y}$ are the observations. The discrete system state (or *mode*) $\mathbf{x}_{d,\tau} \in \{1, \dots, |\mathcal{X}_d|\}$ is a Markov chain that evolves according to a transition matrix \mathcal{T} such that:

$$p(\mathbf{x}_{d,\tau+1} = j | \mathbf{x}_{d,\tau} = i) = [\mathcal{T}]_{ij}. \quad (38)$$

The variables $\omega_\tau \in \mathbb{R}^{n_x}$ and $\nu_\tau \in \mathbb{R}^{n_y}$ are zero-mean Gaussian white noise processes with covariance $Q(\mathbf{x}_{d,\tau})$ and $R(\mathbf{x}_{d,\tau})$, respectively. The initial state distribution $p(\mathbf{x}_{c,0}, \mathbf{x}_{d,0})$ is defined such that $p(\mathbf{x}_{d,0}) = \rho(\mathbf{x}_{d,0})$ and $p(\mathbf{x}_{c,0} | \mathbf{x}_{d,0}) \sim \mathcal{N}(\bar{\mathbf{x}}_0(\mathbf{x}_{d,0}), V(\mathbf{x}_{d,0}))$, and is uncorrelated with the noise variables. In the JMLS formulation, as opposed to the multiple-model formulation, switching between models is allowed; for example stochastic jumps can represent component failures.

We define the problem of hybrid estimation in a JMLS as that of estimating $p(\mathbf{x}_{c,\tau}, \mathbf{x}_{d,\tau} | \mathbf{y}_{1:\tau}, \mathbf{u}_{0:\tau-1})$, the probability distribution of the hybrid discrete-continuous state, conditioned on the sequence of all observations and control inputs. This probability can be written as a sum over all possible mode sequences that end in the mode $\mathbf{x}_{d,\tau}$:

$$\begin{aligned} & p(\mathbf{x}_{c,\tau}, \mathbf{x}_{d,\tau} | \mathbf{y}_{1:\tau}, \mathbf{u}_{0:\tau-1}) \\ &= \sum_{\mathbf{x}_{d,0:\tau-1}} p(\mathbf{x}_{c,\tau}, \mathbf{x}_{d,0:\tau-1}, \mathbf{x}_{d,\tau} | \mathbf{y}_{1:\tau}, \mathbf{u}_{0:\tau-1}). \end{aligned} \quad (39)$$

Each summand can be further expanded as a product of the posterior probability of the discrete mode sequence $\mathbf{x}_{d,0:\tau}$ and

the posterior distribution over the continuous state, conditioned on this mode sequence:

$$\begin{aligned} p(\mathbf{x}_{c,\tau}, \mathbf{x}_{d,0:\tau-1}, \mathbf{x}_{d,\tau} | \mathbf{y}_{1:\tau}, \mathbf{u}_{0:\tau-1}) \\ = p(\mathbf{x}_{d,0:\tau} | \mathbf{y}_{1:\tau}, \mathbf{u}_{0:\tau-1}) p(\mathbf{x}_{c,\tau} | \mathbf{x}_{d,0:\tau}, \mathbf{y}_{1:\tau}, \mathbf{u}_{0:\tau-1}). \end{aligned} \quad (40)$$

For a given mode sequence, the system dynamics are fully known, although time-varying. This means that the probability distribution $p(\mathbf{x}_{c,\tau} | \mathbf{x}_{d,0:\tau-1}, \mathbf{y}_{1:\tau}, \mathbf{u}_{0:\tau-1})$ can be calculated exactly using the Kalman Filter recursion [30]. The probability of a given mode sequence $p(\mathbf{x}_{d,0:\tau-1} | \mathbf{y}_{1:\tau}, \mathbf{u}_{0:\tau-1})$ can also be calculated using the residuals in the Kalman filter equations. In principle, therefore, it is possible to calculate the distribution over the hybrid state $p(\mathbf{x}_{c,\tau}, \mathbf{x}_{d,\tau} | \mathbf{y}_{1:\tau}, \mathbf{u}_{0:\tau-1})$ exactly, yielding a sum-of-Gaussians expression. In practice, however, this *exact* hybrid state estimation is infeasible since the number of mode sequences $\mathbf{x}_{d,1:\tau}$ grows exponentially with time and with the number of possible modes.

B. Approximate Hybrid Estimation

A large number of methods have been proposed that make the hybrid estimation problem tractable by approximating the probability $p(\mathbf{x}_{c,\tau}, \mathbf{x}_{d,\tau} | \mathbf{y}_{1:\tau}, \mathbf{u}_{0:\tau-1})$ [4], [28], [31]. One common approach is to discard mode sequences that have a low posterior probability $p(\mathbf{x}_{d,0:\tau} | \mathbf{y}_{1:\tau}, \mathbf{u}_{0:\tau-1})$. Such *pruning* approaches typically ensure that a fixed number of mode trajectories are tracked. In this paper, we assume that a pruning approach is used so that K individual mode sequences are tracked; this is called *K-best hybrid estimation*. While pruning is usually carried out at every time step, for the purposes of finite-horizon control design, we assume that pruning is carried out at the end of the control horizon; this is discussed in Section VII. Figure 3 shows the pruning process for a time horizon of h time steps and $K = 4$ tracked mode sequences.

It is possible for the true mode sequence to be discarded in this pruning process. If this occurs, the hybrid estimator typically diverges and the approximated state distribution no longer resembles the true distribution. The goal of active estimation for JMLS is to use control inputs to minimize the probability of the true mode sequence being pruned.

Definition 3. *At time step 0, given a JMLS \mathcal{M} , the Active Hybrid Estimation Problem consists of designing a finite sequence of control inputs $\mathbf{u}_{0:h-1}$ that minimizes the probability of K -best hybrid estimation pruning (i.e. discarding) the true mode sequence $\mathbf{x}_{d,0:h-1}^*$.*

As in Section II we assume that Active Hybrid Estimation is invoked at time step zero, without loss of generality. The true mode sequence is pruned at time step h if and only if its posterior probability is not in the top K posteriors. We denote this event *prune*:

$$\begin{aligned} \text{prune} \iff \{p(\mathbf{x}_{d,0:h-1}^* | \mathbf{y}_{1:h}, \mathbf{u}_{0:h-1}) < \\ p(\mathbf{x}_{d,0:h-1}(i) | \mathbf{y}_{1:h}, \mathbf{u}_{0:h-1}) \text{ for } K \text{ or more } i\}. \end{aligned} \quad (41)$$

We must find the control inputs $\mathbf{u}_{0:h-1}$ that minimize the probability of the event *prune*. We write this probability as $P(\text{prune} | \mathbf{u}_{0:h-1})$. In order to calculate this value, we must

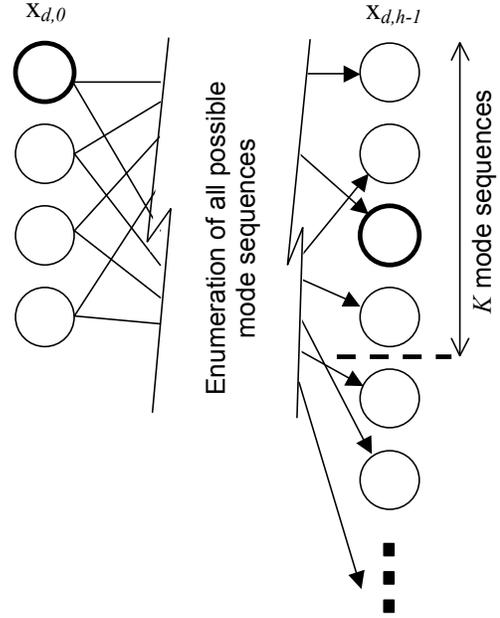


Fig. 3. Pruning approach to approximate hybrid estimation. At time step 0, the estimator is tracking $K = 4$ distinct mode sequences. We assume that at time step h , the posterior probabilities of all possible mode sequences $\mathbf{x}_{d,0:h-1}$ are calculated. The top K sequences are retained, while the remaining sequences are pruned. The true mode sequence is shown in bold.

marginalize over the observations $\mathbf{y}_{1:h}$ and the true mode sequence, since both are unknown at time step 0:

$$\begin{aligned} P(\text{prune} | \mathbf{u}_{0:h-1}) = \sum_{\mathbf{x}_{d,0:h-1}^*} \int_{\mathbf{y}_{1:h}} P(\text{prune} | \mathbf{y}_{1:h}, \mathbf{u}_{0:h-1}) \\ * p(\mathbf{y}_{1:h} | \mathbf{x}_{d,0:h-1}^*) d\mathbf{y}_{1:h} p(\mathbf{x}_{d,0:h-1}^*). \end{aligned} \quad (42)$$

We now look at the two terms in the integrand. First, the observation probability $p(\mathbf{y}_{1:h} | \mathbf{x}_{d,0:h-1}^*)$ can be calculated using repeated application of the Kalman Filter prediction equations (see Section III-C for details), and will yield a Gaussian function of $\mathbf{y}_{1:h}$. Second, we can evaluate $P(\text{prune} | \mathbf{y}_{1:h}, \mathbf{u}_{0:h-1})$, in principle, since it is unity if condition (41) holds and zero otherwise; however this involves calculating the posterior probability of every possible mode sequence given the observations $\mathbf{y}_{1:h}$. Note also that this will be a discontinuous function of $\mathbf{y}_{1:h}$, moreover calculating even the location of the discontinuities is non-trivial. Hence the integral (42) cannot be evaluated in closed form. In the same spirit as the AE-MM approach developed in Section II, we therefore derive a tractable *upper bound* on the probability of pruning the true mode sequence. We then approximate the Active Hybrid Estimation Problem (Def. 3) by minimizing this bound instead of the true probability of pruning.

C. Bounding the Probability of Pruning

In this section we extend the bound (9) to create a bound on the probability of approximate hybrid estimation pruning the true mode sequence.

Theorem 2. *Denote the number of possible mode sequences $\mathbf{x}_{d,0:h-1}$ as N_{seqs} . Enumerate all mode sequences $\mathbf{x}_{d,0:h-1}(i)$*

for $i = 1, \dots, N_{seqs}$. Define:

$$\begin{aligned} \mathbb{Y} &= [\mathbf{y}'_1 \quad \mathbf{y}'_2 \quad \dots \quad \mathbf{y}'_h]' \\ \mathbb{U} &= [\mathbf{u}'_0 \quad \mathbf{u}'_1 \quad \dots \quad \mathbf{u}'_{h-1}]' \\ \mu(i) &= [\mu_1(i)', \dots, \mu_h(i)']' \text{ where } \mu_j(i) = E[\mathbb{Y}|\mathbf{x}_{d,0:h-1}(i)] \\ [\Sigma(i)]_{f,g} &= E\left[\left([\mathbb{Y}]_f - [\mu(i)]_f\right)\left([\mathbb{Y}]_g - [\mu(i)]_g\right) \middle| \mathbf{x}_{d,0:h-1}(i)\right]. \end{aligned} \quad (43)$$

Then:

$$\begin{aligned} &P(\text{prune}|\mathbf{u}_{0:h-1}) \\ &\leq \sum_i \sum_{j>i} P(\mathbf{x}_{d,0:h-1}(i))^{\frac{1}{2}} P(\mathbf{x}_{d,0:h-1}(j))^{\frac{1}{2}} e^{-k(i,j)}, \end{aligned} \quad (44)$$

where $k(i, j)$ is defined in (10) in terms of $\mu(\cdot)$ and $\Sigma(\cdot)$.

Proof: The following implication holds:

$$\begin{aligned} &\{\exists i \ p(\mathbf{x}_{d,0:h-1}^*|\mathbf{y}_{1:h}, \mathbf{u}_{0:h-1}) < p(\mathbf{x}_{d,0:h-1}(i)|\mathbf{y}_{1:h}, \mathbf{u}_{0:h-1})\} \\ &\iff \{p(\mathbf{x}_{d,0:h-1}^*|\mathbf{y}_{1:h}, \mathbf{u}_{0:h-1}) < p(\mathbf{x}_{d,0:h-1}(i)|\mathbf{y}_{1:h}, \mathbf{u}_{0:h-1}) \\ &\text{for } K \text{ or more } i\} \\ &\iff \text{prune}. \end{aligned} \quad (45)$$

Now consider each mode sequence as a hypothesis in the sense of Bayesian hypothesis selection, as in Section II-B. Then the event $\{p(\mathbf{x}_{d,0:h-1}^*|\mathbf{y}_{1:h}, \mathbf{u}_{0:h-1}) < p(\mathbf{x}_{d,0:h-1}(i)|\mathbf{y}_{1:h}, \mathbf{u}_{0:h-1})$ for any $i\}$ is identical to the event that Bayesian hypothesis selection makes an error, denoted *error*. From (45) we have $\text{prune} \implies \text{error}$, and it follows that:

$$P(\text{prune}|\mathbf{u}_{0:h-1}) \leq P(\text{error}). \quad (46)$$

Combining this with Theorem 1 we have:

$$\begin{aligned} &P(\text{prune}|\mathbf{u}_{0:h-1}) \leq P(\text{error}) \\ &\leq \sum_i \sum_{j>i} P(\mathbf{x}_{d,0:h-1}(i))^{\frac{1}{2}} P(\mathbf{x}_{d,0:h-1}(j))^{\frac{1}{2}} e^{-k(i,j)}, \end{aligned} \quad (47)$$

which completes the proof. \square

The mean and covariance expressions defined in (43) can be calculated explicitly in terms of the control inputs using repeated application of the Kalman Filter prediction equations. We first define:

$$f = n_y(p-1) + q, \text{ where } 1 \leq q \leq n_y \quad 1 \leq p \leq h \quad p, q \in \mathbb{Z}. \quad (48)$$

Then the mean is given by:

$$\begin{aligned} &[\mu(i)]_f = [\mu_p(i)]_q \\ &= \left[C(\mathbf{x}_{d,p-1}(i)) \left(\prod_{l=0}^{p-1} A(\mathbf{x}_{d,l}(i)) \right) \bar{\mathbf{x}}_0(i) + D(\mathbf{x}_{d,p-1}(i)) \mathbf{u}_{p-1} \right. \\ &\left. + C(\mathbf{x}_{d,p-1}(i)) \sum_{l=0}^{p-1} \left(\prod_{v=l+1}^{p-1} A(\mathbf{x}_{d,v}(i)) \right) B(\mathbf{x}_{d,l}(i)) \mathbf{u}_l \right]_q. \end{aligned} \quad (49)$$

Defining $g = n_y(r-1) + s$ in the same manner as (48), the expression for the covariance is:

$$\begin{aligned} &[\Sigma(i)]_{f,g} = \left[\mathcal{R}(\mathbf{x}_{d,p})(p, r) + C(\mathbf{x}_{d,p-1}(i)) \left(\prod_{v=0}^{p-1} A(\mathbf{x}_{d,v}(i)) \right) \right. \\ &\quad \left. * V(\mathbf{x}_{d,0}(i)) \left(A(\mathbf{x}_{d,w}(i))' \prod_{w=0}^{r-1} C(\mathbf{x}_{d,r-1}(i))' \right) \right]_{q,s} \\ &\quad + \left[\sum_{l=0}^{m-1} C(\mathbf{x}_{d,p-1}(i)) \left(\prod_{v=l+1}^{p-1} A(\mathbf{x}_{d,v}(i)) \right) \right. \\ &\quad \left. * Q(\mathbf{x}_{d,l}(i)) \left(A(\mathbf{x}_{d,w}(i))' \prod_{w=l+1}^{r-1} C(\mathbf{x}_{d,r-1}(i))' \right) \right]_{q,s}, \end{aligned} \quad (50)$$

where $m = \min\{p, r\}$ and:

$$\mathcal{R}(\mathbf{x}_{d,\tau})(p, r) = \begin{cases} R(\mathbf{x}_{d,\tau}) & p = r \\ 0 & p \neq r. \end{cases} \quad (51)$$

We use the following notation regarding matrix products. Repeated right matrix products are denoted:

$$\left(\prod_{i=1}^p A(\mathbf{x}_{d,i}) \right) = A(\mathbf{x}_{d,1})A(\mathbf{x}_{d,2}), \dots, A(\mathbf{x}_{d,p-1})A(\mathbf{x}_{d,p}), \quad (52)$$

while repeated left matrix products are denoted:

$$\left(A(\mathbf{x}_{d,i}) \prod_{i=1}^p \right) = A(\mathbf{x}_{d,p})A(\mathbf{x}_{d,p-1}), \dots, A(\mathbf{x}_{d,2})A(\mathbf{x}_{d,1}). \quad (53)$$

In principle, therefore, we can use the bound in (44) as an optimization criterion for active hybrid estimation. This is in contrast to the exact value (42), which cannot be evaluated in closed form.

However the new criterion requires evaluating $\mathcal{O}(|\mathcal{X}_d|^{2h})$, so even for a relatively short time horizon and a modest number of possible modes, evaluating (44) is intractable. In Section III-D we overcome this problem using a principled bound relaxation approach.

D. Considering a Subset of Possible Mode Sequences

We aim to find a looser bound on the probability of pruning that requires the evaluation of a fixed number of terms. In deriving the looser bound, we make use of the Minkowski's Inequality[32]:

Lemma 1 (Minkowski's Inequality). For all symmetric, positive definite Σ_i and Σ_j :

$$\frac{|\frac{\Sigma_i + \Sigma_j}{2}|}{\sqrt{|\Sigma_i| |\Sigma_j|}} \geq 1.$$

Theorem 3. Define:

$$F_{ij}(\mathbb{U}) = P(H_i)^{\frac{1}{2}} P(H_j)^{\frac{1}{2}} e^{-k(i,j)}, \quad G_{ij} = P(H_i)^{\frac{1}{2}} P(H_j)^{\frac{1}{2}}. \quad (54)$$

Then for any set of mode sequences S :

$$P(\text{prune}) \leq \sum_{i \in S} \sum_{j>i, j \in S} F_{ij}(\mathbb{U}) + \sum_{i \notin S} \sum_{j>i, j \notin S} G_{ij}. \quad (55)$$

Proof: Following from Lemma 1, and the definition in (10), it is clear that $k(i, j) \geq 0$, and hence $G_{ij} \geq F_{ij}(\mathbb{U})$. We write the bound on the probability of pruning (42) as:

$$P(\text{prune}) \leq \sum_i \sum_{j>i} F_{ij}(\mathbb{U}). \quad (56)$$

The inequality (55) follows from (56). \square

Note that the G_{ij} terms in (55) do not depend on the control inputs \mathbb{U} and hence can be dropped from the cost function. By replacing $F_{ij}(\mathbb{U})$ terms in (56) with G_{ij} terms, we have obtained a looser upper bound on the probability of pruning the true mode sequence with fewer terms. S is the set of mode sequences for which the full bound is calculated as a function of \mathbb{U} . We would like the tightest such bound for a given size of S . We achieve this as follows.

The difference between the bound G_{ij} and the bound $F_{ij}(\mathbb{U})$ is largest when the control inputs \mathbb{U} drive the value of $F_{ij}(\mathbb{U})$ to zero, at which point the difference is $P(H_i)^{\frac{1}{2}}P(H_j)^{\frac{1}{2}}$. Since we do not have knowledge of \mathbb{U} when choosing which mode sequences to include in S , we assume this *worst case* difference between G_{ij} and F_{ij} . In order to find the tightest bound of the form (55), we therefore include in S the mode sequences that maximize:

$$L = \sum_{i \in S} \sum_{j>i, j \in S} P(H_i)^{\frac{1}{2}}P(H_j)^{\frac{1}{2}}. \quad (57)$$

Intuitively, this means that optimization will concentrate on reducing terms where the control inputs can have the greatest effect on $p(\text{prune})$, and will ignore terms that do not contribute significantly. It can be seen that in order to maximize L , the set S must be chosen to contain hypotheses with the greatest probability $P(H_i)$; replacing any $P(H_i)$ value with a lower one can only reduce, or have no effect on, terms in the summation (57).

Although we consider only the $|S|$ mode sequences with highest prior probabilities in the process of control design for Active Hybrid Estimation, this is not to say that we are ignoring available observation data. First, $p(\text{prune})$ does not, by definition, depend on the observations after time step zero. Second, the observations after time step zero *are* used in estimation (rather than control design), when they become available. Third, observations made before $t = 0$ are incorporated into the prior probabilities $p(H_i)$.

In this section we have shown how to derive a tractable upper bound on the probability of pruning the true mode sequence that involves a fixed number $|S|$ of mode sequences for which the observation statistics (49) and (50) need to be calculated. By choosing the $|S|$ most likely mode sequences, we achieve the tightest such bound. Choosing the $|S|$ mode sequences with highest priors $\mathbf{x}_{d,0:h-1}$ is a challenging problem in itself given an exponential number of possible sequences. Prior work has, however, shown that this can be posed as a tree search problem[7]. This enables the best $|S|$ mode sequences to be found efficiently using a best-first informed search approach[33]. For the sake of brevity, we refer the interested reader to [7] for details of this approach.

E. Summary

The AE-JMLS algorithm is summarized in Table II. The algorithm works in parallel with approximate hybrid estimation, which calculates the probability of each of the K tracked mode sequences. Starting from these sequences, AE-JMLS then enumerates the $|S|$ most likely future mode sequences over the horizon using best-first search. AE-JMLS forms an upper bound on the probability of approximate hybrid estimation losing the true mode sequence, which involves only the $|S|$ most likely future sequences. AE-JMLS minimizes this upper bound subject to constraints on the expected system state using Sequential Quadratic Programming. This yields an optimized sequence of control inputs that is applied to the system, while hybrid estimation continues to estimate the hybrid state.

Function	ACTIVEHYBRIDMAIN($\mathcal{M}, P(\cdot), K, h$)	returns
		$(\mathbf{u}_{0:h-1}^*)$
1)	Perform K -best hybrid estimation. Calculate the probabilities of the K tracked mode sequences, as well as the distribution over the continuous state conditioned on each mode sequence.	
2)	Starting from the K tracked mode sequences, enumerate the $ S $ most likely future mode sequences over the horizon $1, \dots, h$ using best-first search.	
3)	Form the upper bound on the probability of pruning the true mode sequence involving only terms corresponding to the $ S $ most likely future mode sequences:	
		$J = \sum_{i \in S} \sum_{j>i, j \in S} P(H_i)^{\frac{1}{2}}P(H_j)^{\frac{1}{2}}e^{-k(i,j)}. \quad (58)$
4)	Using Sequential Quadratic Programming, minimize (58) subject to constraints on the expected system state and control inputs to find the optimal control sequence $\mathbf{u}_{0:h-1}^*$.	
5)	Execute optimal control sequence $\mathbf{u}_{0:h-1}^*$ while estimating hybrid state.	

TABLE II. AE-JMLS for Active Hybrid Estimation algorithm with JMLS.

IV. SIMULATION RESULTS: MULTIPLE-MODEL DISCRIMINATION

In this section we demonstrate the AE-MM algorithm using an aircraft fault detection scenario. We use a discrete time approximation to the longitudinal aircraft dynamics, linearized about the trim state with $\Delta t = 0.5s$. Here, the state of the system consists of the vertical velocity, the horizontal velocity, the pitch rate, the pitch angle and the altitude. The observed output of the system is taken to be the pitch rate $\dot{\theta}$ and the vertical velocity V_y . The input is denoted u , and is taken to be the requested elevator angle. We assume that the elevator actuator saturates at $\pm 0.25rad$.

In the multiple-model selection task, we must determine which model $H(i)$ is most likely. Under model H_i , the system is described by $\{A(i), B(i), C(i), D(i), Q(i), R(i)\}$. For the aircraft example, we consider three single-point failures; the pitch rate sensor may fail, the vertical velocity sensor may fail, or the elevator actuator may fail. This gives four models:

- H_0 : Nominal (no faults)
- H_1 : Faulty pitch rate sensor
- H_2 : Faulty vertical velocity sensor
- H_3 : Faulty elevator actuator

The model parameters for H_0 are:

$$A(0) = \begin{bmatrix} 0.9985 & 0.1950 & 0 & -0.161 & 0 \\ -0.0325 & 0.8405 & 3.87 & 0 & 0 \\ 0.01 & -0.0505 & 0.7855 & 0 & 0 \\ 0.0 & 0 & 0.5 & 1.0 & 0 \\ 0.5 & 0 & 0 & 0 & 1.0 \end{bmatrix}$$

$$B(0) = \begin{bmatrix} 0.005 \\ -0.09 \\ -0.58 \\ 0.0 \\ 0.0 \end{bmatrix} \quad C(0) = \begin{bmatrix} 0 & 1 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}' \quad D(0) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

(59)

In the case of H_1 and H_2 , the C matrix is modified so that the sensor reading is zero mean white noise. In the case of H_3 , the B matrix is modified so that the elevator exerts no control effort. In all models, the process and observation noise matrices are given by $Q = \text{diag}([0.01 \ 0.01 \ 0.01 \ 0.01 \ 0.1])$ and $R = \text{diag}([0.01 \ 0.01])$, while the initial state distribution is given by $\bar{x}_0 = [0 \ 0 \ -0.1 \ 0 \ 100]'$ and $P = Q$. Consistent with a multiple-model fault detection framework, we assume that the system matrices $\{A, B, C, D, Q, R\}$ are fully known for each of the possible faults, and that the true model persists indefinitely. In other words, in this section we do not allow switching between the different models; switching dynamics are considered in Section III.

In each of the following discrimination tasks we control the system by constraining the expected state, conditioned on nominal operation. We evaluate the efficacy of the AE-MM approach in terms of the reduction in the probability of model selection error achieved by using the discrimination-optimal sequence. There are two different error probabilities that we consider:

- 1) *Batch Error Probability*. This error probability is based on the assumption that Bayesian model selection is carried out based on all the observations over the entire horizon. This value is calculated in closed form.
- 2) *Sequential Error Probability*. This is the probability of a sequential multiple-model estimation scheme making a model selection error. This probability is estimated by carrying out a large number of simulations.

As discussed in Section VII AE-MM minimizes an upper bound on the batch error probability although most practical multiple-model implementations are sequential. Nevertheless, the results provided here show that the new approach dramatically reduces the sequential error probability also.

A. Altitude Envelope

Figure 4 and Figure 5 show results from a fault detection scenario where the aircraft is constrained to remain within a flight envelope around an altitude of $100m$. The elevator angle is constrained to be at most $0.25rad$ in magnitude. The prior probabilities of models H_0 through H_3 are 0.65, 0.1, 0.05 and 0.2 respectively. We assume that these priors have been generated by a multiple-model estimator running up until time 0.

The optimized control input yields a batch error probability of 0.0003 and a sequential error probability of 0.0004. The *bang-bang* nature of the optimized control input highlights the fact that by optimizing up against hard constraints the discrimination power of the input signal can be much greater than a power-bounded auxiliary signal. This particular solution took $58.5s$ to generate, of which $30.7s$ was used to generate the necessary covariance matrices and of which $27.8s$ was used to perform the nonlinear optimization using Matlab's `fmincon` function.

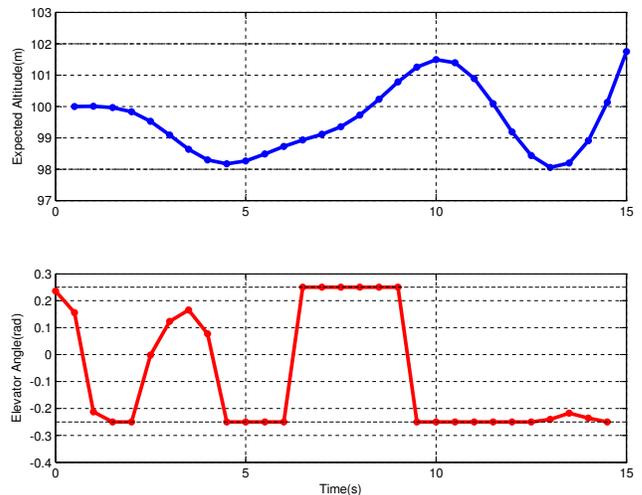


Fig. 4. Discrimination-optimized input design for aircraft flight envelope scenario. Top: Expected altitude of aircraft given nominal operation. Bottom: Optimized sequence of control inputs. The optimal control discriminates between the different models while ensuring that in the nominal case, the aircraft altitude remains between 98 and $102m$. The optimized control input yields a batch error probability of 0.0003 and a sequential error probability of 0.0004.

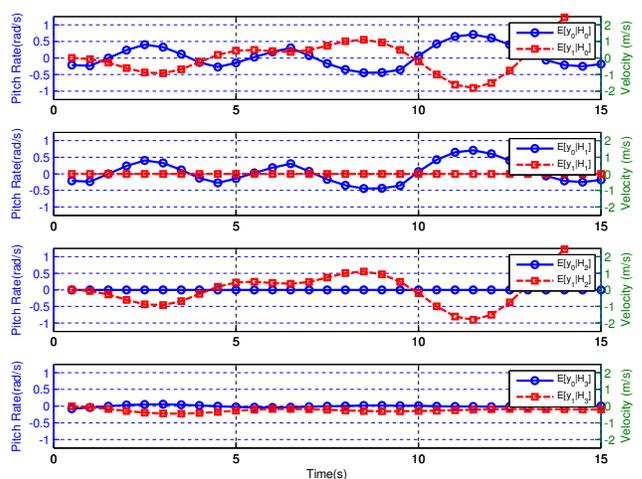


Fig. 5. Expected observations for aircraft flight envelope scenario with optimized control input. The top plot shows the nominal case, where there are no faults. In the second and third plots, the pitch rate sensor and vertical velocity sensors are faulty, respectively. In the bottom plot, the elevator actuator is faulty. The optimized control input ensures that the observation sequences in each case are as different as possible, in order to minimize the probability of model selection error.

B. Manually Generated Sequence

In order to identify the longitudinal dynamics of an aircraft, pilots typically use a doublet control input[34]. Figure 6 shows such a control input sequence, with the same actuator limits as for the optimized control sequence. This sequence yields a batch error probability of 0.0269 and a sequential error probability of 0.0258. Hence the optimized sequence in Figure 4 has significantly greater discrimination power than the manually generated sequence.

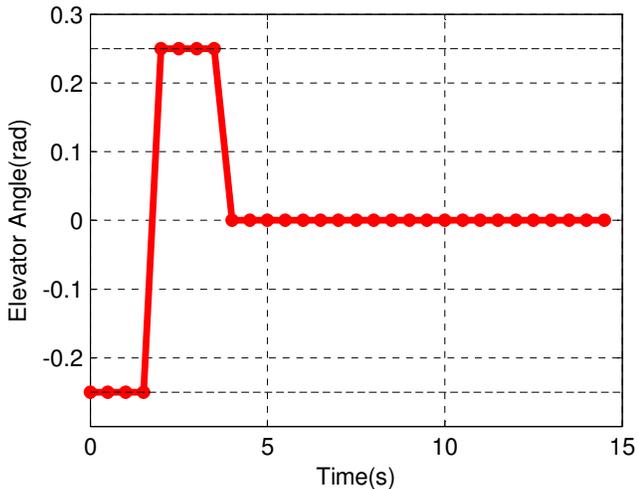


Fig. 6. Typical manually generated identification sequence. This doublet form is used by pilots to perform aircraft system identification. This sequence yields a batch error probability of 0.0269 and a sequential error probability of 0.0258.

C. Auxiliary Signal

A number of existing approaches to control design for discrimination and model identification add a low power auxiliary signal to the nominal control sequence[16], [18], [19]. The power of the signal is small so that the effect on the system state is small. A typical approach to auxiliary signal design uses a Pseudo-Random Binary Signal (PRBS)[35].

Figure 7 shows an altitude-hold control signal with a typical PRBS auxiliary signal added. The PRBS signal was constrained to have a maximum elevator angle of 0.01 in order to ensure that the aircraft altitude did not deviate significantly from 100m. Averaged over a number of randomly-generated signals, the resulting batch error probability was 0.3455, and the sequential error probability was 0.3510, which is far worse than the values obtained using the new discrimination approach.

D. Altitude Change Maneuver

The AE-MM algorithm can use constraints to ensure that a given control task is performed, while optimizing with respect to discrimination. This is demonstrated in Figure 8, where the aircraft carries out a maneuver that changes its altitude from 100m to 120m, conditioned on the elevator actuator being functional. The discrimination-optimal control sequence is compared to the fuel-optimal one. Optimizing with respect

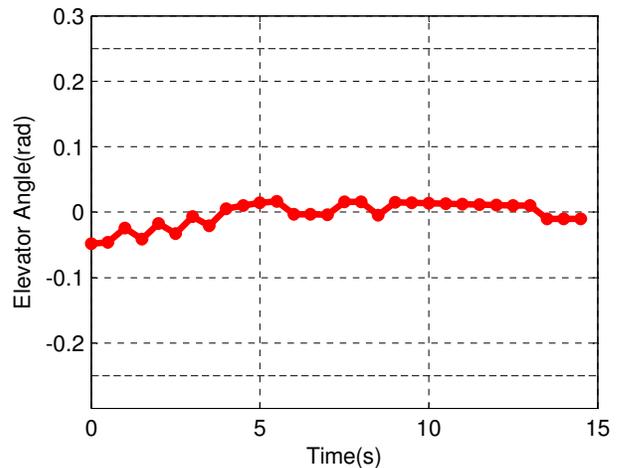


Fig. 7. Typical control sequence with added Pseudo-Random Binary auxiliary signal. This sequence yields a batch error probability of 0.3360 and a sequential error probability of 0.3406.

to discrimination yields a batch error probability of 0.0006 and a sequential error probability of 0.0001. Optimizing with respect to fuel yields a batch error probability of 0.1914 and a sequential error probability of 0.1833. Hence a dramatic improvement in fault detection can be achieved by using control inputs designed for model discrimination, rather than those designed to optimize some other criterion and employing only passive model selection. This particular solution took 51.4s to generate, of which 30.9s was spent calculating the necessary covariances and 20.5s was spent performing the nonlinear optimization.

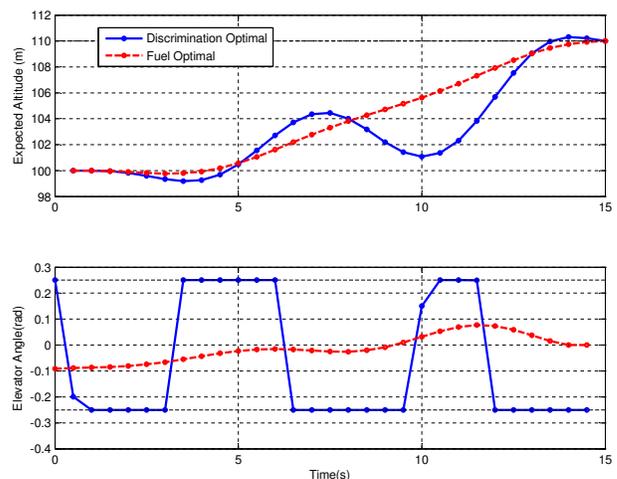


Fig. 8. Discrimination-optimal and fuel-optimal control design for altitude change maneuver. The discrimination-optimal sequence gives a batch error probability of 0.0006 and a sequential error probability of 0.0001, while the fuel-optimal sequence gives a batch error probability of 0.1914 and a sequential error probability of 0.1833.

V. SIMULATIONS RESULTS: JMLS DISCRIMINATION

In this section we demonstrate the AE-JMLS algorithm in simulation. We consider the aircraft described in Section IV, except we now model the system as a JMLS with the following modes:

- Mode 1: Nominal (no faults)
- Mode 2: Faulty pitch rate sensor
- Mode 3: Faulty vertical velocity sensor
- Mode 4: Faulty elevator actuator

These modes are the same as the models described in Section IV. The key difference is that the JMLS model explicitly models stochastic jumps between the modes; stochastic jumps represent component failures or recoveries from failure. The transition probability matrix is:

$$T = \begin{bmatrix} 0.97 & 0.01 & 0.01 & 0.01 \\ 0.0 & 1.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 1.0 \end{bmatrix}. \quad (60)$$

Notice that once a fault occurs, it persists indefinitely. The initial belief state is uniform across the four discrete modes. As in Section IV-D we constrain the expected system state, conditioned on nominal operation, to make the aircraft carry out an altitude change maneuver.

Given the designed control sequence we simulate the JMLS aircraft model and carry out approximate hybrid estimation as described in Section III-A. Hybrid estimation is sequential, and at every time step mode sequences that are not in the 20 most likely are discarded. While in designing the control sequence we assume that sequences are discarded at the end of the planning horizon, we evaluate the new control design approach in terms of the benefits afforded to *sequential* hybrid estimation, since this is the most common implementation. The two metrics used are:

- 1) *Probability of discarding the true mode sequence.* This is estimated by carrying out a large number of simulations and recording occurrences of true mode sequence loss.
- 2) *Probability of Maximum A Posteriori (MAP) mode sequence estimation error.* Correctly estimating the MAP mode sequence is important for control, in particular. This value is again estimated through a large number of simulations.

The control design algorithm considered the *a priori* most likely 10 sequences in the set S , as defined in (44). The elevator angle was constrained to have a maximum magnitude of $0.25rad$. Figure 9 shows the designed active hybrid estimation control input, as well as the fuel-optimal sequence, for comparison. Active hybrid estimation yields a probability of losing the correct mode sequence of 0.0570 and a probability of MAP mode sequence error of 0.1540. Optimizing with respect to fuel yields a probability of losing the correct mode sequence of 0.0970 and a MAP mode sequence error probability of 0.3040. Again, a significant improvement in fault detection can be achieved by using control inputs designed for model discrimination, rather than those designed to optimize some other criterion and employing only passive hybrid estimation. This solution took $269.7s$ to generate, of which $1.02s$ was spent finding the *a priori* most likely 10 sequences, $76.3s$ was spent calculating the necessary covariances, and $192.4s$ was spent carrying out the nonlinear optimization.

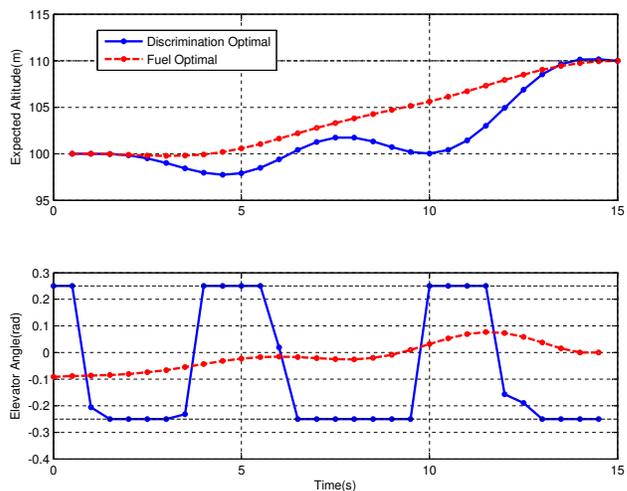


Fig. 9. Active hybrid estimation for altitude change maneuver with JMLS aircraft model. The active estimation sequence yields a probability of losing the correct mode sequence of 0.0570 and a probability of MAP mode sequence error of 0.1540. Optimizing with respect to fuel yields a probability of losing the correct mode sequence of 0.0970 and a MAP mode sequence error probability of 0.3040.

VI. CONCLUSION

This paper introduced a novel method for active state estimation in Jump Markov Linear Systems. The method designs finite sequences of control inputs that reduce the probability of pruning the true mode sequence while ensuring that a given control task is achieved. We first presented a method for constrained optimal discrimination between a finite number of linear dynamics systems. By extending this approach we then derived a tractable active hybrid estimation method. Simulation results showed that the new method significantly reduces the probability of hybrid estimation losing the true mode sequence.

VII. DISCUSSION

In the derivation of the new methods we assumed that both model selection and mode sequence pruning are carried out at the end of the finite horizon control sequence, rather than at every time step. However in most Multiple Model and hybrid estimation schemes, selection occurs at every time step. While it would be possible, and in fact simpler, to formulate the control design problem for one time step, we did not do so for two reasons. First, in the discrete-time formulation, in many systems the effect of control inputs at time τ is not manifested in the observations until some time after $\tau + 1$. Hence a one-step design approach will be severely limited. Second, by constraining the system state over a long horizon, the optimization has much greater latitude in designing a powerful sequence for the purposes of estimation; the system state can be driven far from its initial value, while being brought back to its goal value by the end of the time horizon. Hence by considering the probability of pruning over a horizon, rather than over a single time step, the approach yields sequences that are more powerful with regard to discrimination. Furthermore, we provided empirical results that show that the new methods do indeed reduce the probability of error when a sequential estimation method is employed.

- | |
|--|
| <ol style="list-style-type: none"> 1) <i>Randomly generate M models.</i> Each model H_i has a prior probability $p(H_i)$, a mean $\mu(i)$ and a covariance $\Sigma(i)$ for the observation distribution. 2) <i>Evaluate upper bound on error probability.</i> The upper bound (9) is evaluated for the generated model set. 3) <i>Simulate observations.</i> The true model is chosen at random, according to the prior distribution $p(H_i)$. Then an observation \mathbf{y} is drawn from the probability distribution $\mathcal{N}(\mu(i), \Sigma(i))$. 4) <i>Select most likely model.</i> The probability $p(H_i \mathbf{y})$ is evaluated for each H_i, and the most likely model is identified. 5) <i>Record errors.</i> If the most likely model is not the true model, record the selection as an error. 6) <i>Repeat and calculate error probability.</i> Steps 1 through 5 are repeated a large number of times. The probability of model selection error is approximated as the fraction of errors recorded. |
|--|

TABLE III. Experimental process for analyzing tightness of bound on probability of Multiple Model selection error.

APPENDIX I BOUND TIGHTNESS ANALYSIS

We now discuss some properties of the bound on multiple model selection error in (9). In the AE-MM algorithm we minimize this bound in order to minimize the probability of model selection error. We do not prove analytically that designing control inputs to reduce this upper bound necessarily reduces the probability of model selection error. Instead, we motivate the use of an upper bound minimization approach in three ways. First, note that the bound (9) is lower-bounded by zero. Driving the bound to zero will trivially reduce the probability of error (unless the probability of error was already zero). Second, we showed empirically in Section IV with a fault detection scenario that minimizing the upper bound does, indeed, minimize the probability of error. Finally, in this section, we analyze the tightness of the bound and show empirically that the bound follows the same trend as the true probability of error, in randomly generated instances.

For the empirical tightness analysis, we use the process in Table III. The observations in this case are scalar. Figure 10 shows the upper bound and the true probability of error for $M = 5$, where M is the number of hypotheses, and a variety of randomly generated mean values $\mu(i)$. Here the true probability of error was estimated using 10^5 Monte Carlo simulations. To aid visualization, in this case we have fixed $\Sigma(i)$ and $p(H_i)$. The x -axis of Figure 10 is the average Euclidean distance between the observation means; we choose this measure to enable visualization on a single figure.

Figure 10 shows that the bound is not particularly tight. This is not surprising, since prior work has shown that the Battacharyya Bound is relatively loose[22], and the new bound (9) is at least as loose as the Battacharyya Bound. The bound does, however, follow the same trend as the true probability of error. This is empirical motivation for using the bound as an optimization criterion in the place of the probability of error.

Figure 11 shows the ratio between the upper bound and the true probability of error as a function of M , the number of models. In this case all of $\mu(i)$, $\Sigma(i)$ and $p(H_i)$ are generated randomly. Figure 11 shows that the bound becomes less tight as the number of models increases, and that the relationship is approximately linear after $M = 4$.

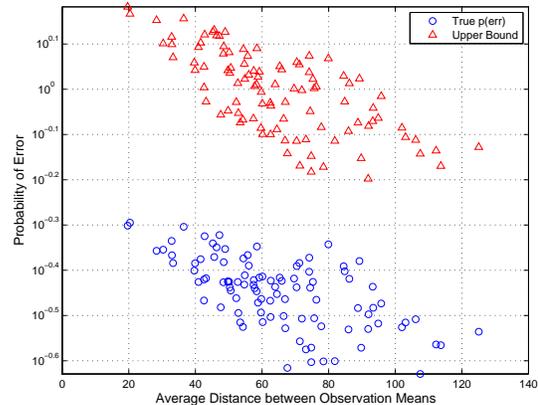


Fig. 10. Tightness of new upper bound on the probability of Multiple Model selection error. Here there are 5 models, with randomly generated observation means. The true probability of error is estimated using a large number of simulations. The bound is not particularly tight, but follows the same trend as the probability of error.

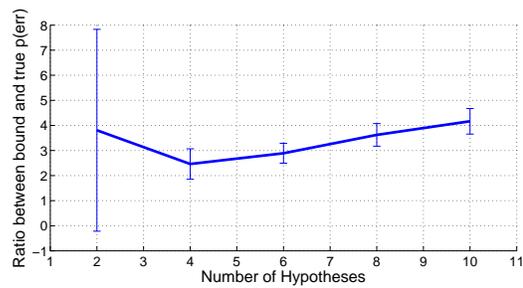


Fig. 11. Ratio of new upper bound to the probability of Multiple Model selection error.

REFERENCES

- [1] R. Dearden and D. Clancy, "Particle filters for real-time fault detection in planetary rovers," in *Proceedings of the 13th International Workshop on Principles of Diagnosis (DX02)*, May 2002, pp. 1–6.
- [2] L. Blackmore, S. Funiak, and B. C. Williams, "Combining stochastic and greedy search in hybrid estimation," in *Proc. 20th National Conference on Artificial Intelligence (AAAI05)*, Pittsburgh, PA, 2005.
- [3] V. Pavlovic, J. Rehg, T.-J. Cham, and K. Murphy, "A dynamic Bayesian network approach to figure tracking using learned dynamic models," in *Proceedings of ICCV*, 1999.
- [4] S. M. Oh, J. M. Rehg, T. Balch, and F. Dallaert, "Data-driven MCMC for learning and inference in switching linear dynamic systems," in *Proc. 20th National Conference on Artificial Intelligence (AAAI-2005)*, Pittsburgh, PA, 2005.
- [5] U. Lerner and R. Parr, "Inference in hybrid networks: Theoretical limits and practical algorithms," in *Proceedings of the 17th Annual Conference on Uncertainty in Artificial Intelligence (UAI-01)*, Seattle, Washington, August 2001, pp. 310–318.
- [6] C. B. Chang and M. Athans, "State estimation for discrete systems with switching parameters," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-14, pp. 418–424, May 1978.
- [7] M. Hofbauer and B. C. Williams, "Mode estimation of probabilistic hybrid systems," in *Intl. Conf. on Hybrid Systems: Computation and Control*, May 2002.
- [8] A. Doucet, S. Godsill, and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," *Statistics and Computing*, vol. 10, no. 3, pp. 197–208, July 2000.
- [9] U. Lerner, R. Parr, D. Koller, and G. Biswas, "Bayesian fault detection and diagnosis in dynamic systems," in *Proc. of the 17th National Conference on A. I.*, July 2000, pp. 531–537. [Online]. Available: citeseer.nj.nec.com/lerner00bayesian.html
- [10] Y. Bar-Shalom and T. E. Fortmann, *Tracking and Data Association*. Academic Press, 1988.

- [11] U. Lerner, "Hybrid bayesian networks for reasoning about complex systems," Ph.D. dissertation, Stanford University, October 2002.
- [12] S. Funiak and B. Williams, "Multi-modal particle filtering for hybrid systems with autonomous mode transitions," in *Proceedings of SafeProcess*, June 2003.
- [13] F. Hutter and R. Dearden, "The Gaussian particle filter for diagnosis of non-linear systems," in *Proc. 14th Int. Conf. on Principles of Diagnosis*, Washington, DC, USA, June 2003, pp. 65–70.
- [14] V. Verma, S. Thrun, and R. Simmons, "Variable resolution particle filter," in *Proc. International Joint Conf. on Artificial Intelligence*. AAAI, August 2003.
- [15] V. Fedorov, *Theory of Optimal Experiments*. New York: Academic Press, 1972.
- [16] R. Nikoukhah, S. L. Campbell, K. Horton, and F. Delebecque, "Auxiliary signal design for robust multi-model identification," *IEEE Trans. Automat. Contr.*, vol. 47, pp. 158–163, 2002.
- [17] R. Nikoukhah, S. L. Campbell, and F. Delebecque, "Detection signal design for failure detection: a robust approach," *Int. J Adaptive control and Signal Processing*, vol. 14, pp. 701–724, 2000.
- [18] X. J. Zhang, "Auxiliary signal design in fault detection and diagnosis," *NASA STI/Recon Technical Report A*, vol. 90, pp. 24257–+, 1989.
- [19] F. Kerestecioglu, *Change Detection and Input Design in Dynamical Systems*. New York: Wiley, 1993.
- [20] R. J. Patton, P. M. Frank, and R. N. Clark, Eds., *Issues of Fault Diagnosis for Dynamic Systems*. New York: Springer, 2002.
- [21] M. Simandl, I. Puncoschar, and J. Kralovec, "Rolling horizon for active fault detection," in *Proc. Control and Decision Conference*, Seville, Spain, 2005.
- [22] R. Duda, P. Hart, and D. Stork, *Pattern Classification (2nd ed.)*. Wiley Interscience, 2000.
- [23] J. Nocedal and S. Wright, *Numerical Optimization*. Springer, 1999.
- [24] D. T. Magill, "Optimal adaptive estimation of sampled stochastic processes," *IEEE Transactions on Automatic Control*, vol. AC-10, pp. 434–439, 1965.
- [25] P. Hanlon and P. Maybeck, "Multiple-model adaptive estimation using a residual correlation kalman filter bank," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 36, pp. 393–406, Apr. 2000.
- [26] A. Willsky, "A survey of design methods for failure detection in dynamic systems," *Automatica*, vol. 12, pp. 601–611, 1976.
- [27] J. Chen and R. Patton, *Robust Model-Based Fault Diagnosis for Dynamic Systems*. Boston: Kluwer Academic, 1999.
- [28] J. K. Tugnait, "Detection and estimation for abruptly changing systems," *Automatica*, vol. 18, 1982.
- [29] P. Pardalos and J. Rosen, "Methods for global concave minimization: A bibliographic survey," *Society for Industrial and Applied Mathematics*, SIAM Review, Sept. 1986.
- [30] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME—Journal of Basic Engineering*, vol. 82, pp. 35–45, Apr. 1960.
- [31] H. Blom and Y. Bar-Shalom, "The interacting multiple model algorithm for systems with Markovian switching coefficients," *IEEE Transactions on Automatic Control*, vol. 33, 1988.
- [32] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge, England: Cambridge University Press, 1985.
- [33] S. J. Russell and P. Norvig, Eds., *Artificial Intelligence: A Modern Approach*. Prentice Hall, 2003.
- [34] J. Jang and C. Tomlin, "Longitudinal stability augmentation system design of the Stanford DragonFly UAV using a single GPS receiver," in *Proceedings of the AIAA GNC conference*, Austin, Texas, Aug. 2003.
- [35] G. C. Goodwin and R. L. Payne, *Dynamic System Identification: Experiment Design and Data Analysis*. New York: Academic Press, 1977.



Lars Blackmore has a Ph.D. in Control and Estimation from the Massachusetts Institute of Technology as well as B.A. and M.Eng. degrees from the University of Cambridge (UK). He is currently with the Guidance and Control Analysis Group at the NASA Jet Propulsion Laboratory, where his research interests include estimation and control under stochastic uncertainty.



Senthooran Rajamanoharan is a Ph.D. student at the Department of Applied Mathematics and Theoretical Physics, University of Cambridge, where he is working on Brane Inflation and Cosmic D-Strings.



Brian C. Williams is a professor at MIT in the department of Aeronautics and Astronautics. He received his Ph.D. from MIT in 1989. He pioneered multiple fault, model-based diagnosis at the Xerox Palo Alto Research Center, and at NASA Ames Research Center formed the Autonomous Systems Area. He co-invented the Remote Agent model-based autonomous control system, which flew on the NASA Deep Space One probe in 1999. He has served as guest editor of the *Artificial Intelligence Journal* and has been on the editorial board of the *Journal of Artificial Intelligence Research*. He is currently a member of the Advisory Council of the NASA Jet Propulsion Laboratory at Caltech. Prof. Williams' research concentrates on model-based autonomy - the creation of long-lived autonomous systems that are able to explore, command, diagnose and repair themselves using fast, commonsense reasoning.