Robust Estimation Algorithm for A Class of Hybrid Systems with Unknown Continuous Fault Inputs

Weiyi Liu and Inseok Hwang

Abstract—In this paper, we develop a robust hybrid estimation algorithm for the Stochastic Linear Hybrid System (SLHS) with unknown continuous fault inputs. Most existing estimation algorithms for hybrid systems with fault inputs are designed such that every faulty mode is modeled as a discrete state, which incurs a large number of the discrete states in the estimation model. The proposed algorithm overcomes this problem by decoupling the unknown fault inputs from the estimation error dynamics and has the same number of the discrete states as the original hybrid system. The proposed algorithm is validated with a vertical take-off and landing (VTOL) aircraft application.

I. INTRODUCTION

State estimation is important and challenging for hybrid systems with interacting continuous and discrete dynamics. In this paper, we are interested in hybrid systems with continuous unknown fault inputs. Most existing hybrid estimation algorithms assume the nominal system is a continuous dynamical system and a fault is modeled as a discrete state, thus the resulting dynamical system with faults is a hybrid system [1]. Some authors use the IMM filter [2] [3] to realize hybrid estimation. This approach could introduce a large number discrete states to describe various faulty behaviors of the system. The direct consequences are: (a) it is hard to parameterize a complex Markov transition matrix for a hybrid system with the large number of discrete states (modes); and (b) the IMM filter’s performance degrades rapidly when there are too many discrete states [4]. The Variable-structure IMM algorithm lowers the number of the discrete states by exploring some structural properties in the discrete transition model of a dynamical system with faults [5]. This approach increases the complexity of the estimation algorithm and requires much knowledge on the properties of the transition model.

In this paper, we propose a hybrid estimation algorithm for the Stochastic Linear Hybrid System (SLHS) with unknown fault inputs. To make the continuous and discrete state estimates robust (insensitive) to the unknown fault inputs, our algorithm decouples the fault inputs from the estimation error dynamics and the residual output of each filter corresponding to each discrete state. Similar to the Kalman Filter, the algorithm minimizes the continuous state estimation error at each time step. We consider two types of discrete state transition models in the SLHS: Markov-jump transition and state-dependent jump models. We develop a hybrid estimation algorithm for both transition models.

As the IMM algorithm, our algorithm uses the merging technique to manage the exponentially increasing number of hypotheses. However, our approach does not model each fault as a discrete state. We consider only the discrete state transitions of the nominal hybrid system. Therefore, we can do estimation without incurring too many discrete states to represent different combinations of faults. This property is strongly advantageous for accurate and computationally efficient hybrid estimation [4].

To demonstrate the performance of our algorithm, we apply our algorithm to a vertical take-off and landing (VTOL) aircraft example. We model the longitudinal dynamics of the VTOL aircraft as the SLHS and use the proposed hybrid estimation algorithm for state estimation when actuator faults occur. We compare the performance of our algorithm with that of the IMM algorithm and show that our algorithm provides better estimation accuracy.

The rest of this paper is organized as follows: in Section II, we give a description of the SLHS model with the unknown fault inputs. We formulate the estimation problem as a constrained optimization problem. In Section III, we derive the hybrid estimation algorithm which is robust to the unknown fault input. In Section IV, we illustrate the performance of our algorithm through a VTOL aircraft example. Conclusions are given in Section V.

II. PROBLEM FORMULATION

In this section, we propose the Stochastic Linear Hybrid System (SLHS) model with two discrete state transition models: Markov-jump transition model and state-dependent transition model. Also, we formulate the robust estimation problem explicitly.

A. Hybrid System Model

We consider the following Stochastic Linear Hybrid System (SLHS) model. The continuous dynamics (in discrete time) of the SLHS with the continuous unknown fault inputs is given by:

\[ x(k) = A_q(k)x(k-1) + B_q(k)u(k) + F_q(k)f(k) + w(k) \]
\[ z(k) = C_q(k)x(k) + D_q(k)u(k) + v(k) \]

where \( x(k) \in X = \mathbb{R}^n \) is the state vector; \( u \in \mathbb{R}^m \) is the known input vector; \( z \in \mathbb{R}^p \) is the output (measurement) vector; \( f \in \mathbb{R}^{m_f} \) is the unknown fault input; \( q(k) \in Q = \{ 1, 2, \ldots, n_d \} \) is the discrete state at time \( k \); \( Q \) is a finite set of all the discrete states; \( A_q, B_q, C_q, D_q \) and \( F_q \) are the system matrices with appropriate dimensions, corresponding to each discrete
state \( q \in \mathcal{Q} \) and \( w(k) \in \mathbb{R}^n \) and \( v(k) \in \mathbb{R}^p \) are the white Gaussian process noise and sensor noise with zero mean and covariances \( Q(k) \) and \( R(k) \), respectively. Without loss of generality, we assume that \( D_q \) are zero matrices. We make the following assumptions throughout the paper:

**Assumption 1:** \( F_q \) are of full column rank [6].

Assumption 1 guarantees the existence of a solution to the fault isolation problem.

**Assumption 2:** The row rank of \( C_q \) must be no less than the column rank of \( F_q \) [7].

For Assumption 2, an equivalent statement is: \( C_qF_q \) is of full column rank. We will see later that Assumption 2 guarantees the existence of an estimator which is insensitive to the unknown fault input.

There are two types of discrete state transitions in the SLHS:

1) **Markov-jump transition model:** the discrete state transition history is a realization of a homogeneous Markov Chain. The finite state space of the Markov Chain is the discrete state space \( \mathcal{Q} \). Suppose at each time \( k \), the probability vector is given by \( \pi(k) = \{\pi_1(k)\cdots\pi_{|Q|}(k)\}^T \), with each entry of the vector \( \pi_i(k) \) denotes the probability that the system’s true discrete state is \( i \). Then, at the next time step the probability vector is:

\[
\pi(k+1) = \Gamma \pi(k)
\]

where a constant matrix \( \Gamma \) is the Markov transition matrix with \( \sum_i \Gamma_{ij} = 1 \) (we use \( \Gamma_{ij} \) to denote the scalar component in the \( i \)-th row and \( j \)-th column in the Markov transition matrix \( \Gamma \)).

2) **State-dependent transition model:** the discrete state transition is governed by:

\[
q(k+1) = \gamma(q(k),x(k),\theta)
\]

where \( \theta \in \Theta = \mathbb{R}^l \) and \( \gamma: \mathcal{Q} \times X \times \Theta \rightarrow \mathcal{Q} \) is the discrete-state transition function defined as:

\[
\gamma(i,x,\theta) = j \quad \text{if} \quad [x^T \theta^T]^T \in G(i,j)
\]

We call \( G(i,j) \) as the guard condition. For each combination of \( (i,j) \), the guard condition \( G(i,j) \) is a subset of the space \( \Omega = X \times \Theta \) with the following assumption:

**Assumption 3:** The set of guards \( \{G(i,j)|j \in \mathcal{Q}\} \) is a set of disjoint partitions of the space \( \Omega \) for any given \( i \in \mathcal{Q} \):

\[
G(i,j) \cap G(i,k) = \emptyset \quad \forall i,j,k \in \mathcal{Q} \quad \text{and} \quad j \neq k,
\]

and

\[
\bigcup_{j=1}^{\mathcal{Q}} G(i,j) = \Omega \quad \forall i \in \mathcal{Q}
\]

In this paper, we consider a specific kind of the guard condition \( \{G(i,j)|j \in \mathcal{Q}\} \) named as the stochastic linear guard condition:

\[
G(i,j) = \left\{ \begin{bmatrix} x \\ \theta \end{bmatrix} \mid x \in X, \theta \in \Theta, L_{ij} \begin{bmatrix} x \\ \theta \end{bmatrix} + b_{ij} \leq 0 \right\}
\]

where \( \theta \in \Theta = \mathbb{R}^l \) and \( \theta \sim \mathcal{N}(\theta; \Sigma_\theta) \) is a \( l \)-dimensional Gaussian random vector with mean \( \theta \) and covariance \( \Sigma_\theta \) representing uncertainties in the guard condition; \( L_{ij} \) is a \( n \times (n+l) \) matrix, \( b_{ij} \) is a constant \( n \)-dimensional vector, and \( v \) is the dimension of the vector inequality. Here, a vector inequality \( y \leq 0 \) means that each scalar element of \( y \) is non-negative.

### B. Problem formulation of the robust estimation problem

We first design an estimator for the LTI system \( \{A_q,B_q,C_q,F_q|q \in \mathcal{Q}\} \) for each discrete state:

\[
\dot{x}(k+1) = A_q \dot{x}(k) + B_q u(k)
\]

\[
+ L_q[z(k+1) - C_q A_q \dot{x}(k) - C_q B_q u(k)]
\]

\[
r(k) := T_q[z(k) - C_q A_q \dot{x}(k-1) - C_q B_q u(k)]
\]

where \( \dot{x} \in \mathbb{R}^n \) is the continuous state; \( r(k) \in \mathbb{R}^m \) is the residual generated by the estimator, and matrices \( L_q \) and \( T_q \) are the design parameters. We define the estimation error as:

\[
e(k) := x(k) - \hat{x}(k)
\]

Subtracting (8) from (1) and using (10), we can get the error dynamics for discrete state \( q \):

\[
e(k+1) := (A_q - L_q C_q A_q) e(k) + (F_q - L_q C_q F_q)
\]

\[
f(k) + (I - L_q C_q) w(k) - L_q v(k+1)
\]

\[
+ T_q C_q w(k-1) + T_q v(k)
\]

To estimate the continuous state \( x(k) \) at time \( k \) regardless of \( f(k) \), the robust estimator must be such that \( f(k) \) is completely decoupled from the error dynamics (11) and the residual (12), which is equivalent to:

\[
F_q - L_q C_q F_q = 0
\]

and

\[
T_q C_q F_q = 0
\]

At the same time, in the robust estimation algorithm for each discrete state, we want to minimize the estimation error at each step, with (13) and (14) being imposed as constraints of the optimization problem. The proposed robust hybrid estimation algorithm is composed of a set of robust estimators in (8) and (9), each matched to a discrete state.

### III. ROBUST HYBRID ESTIMATION ALGORITHM

In this section, we propose a robust hybrid estimation algorithm for the SLHS with unknown fault inputs. An estimation algorithm for a hybrid system with unknown fault inputs typically has more discrete states than the nominal hybrid system, because the resulting hybrid system has extra discrete states representing individual faults and/or combination of them. However, too many discrete states degrades the estimation accuracy rapidly in the hybrid estimation algorithm [4]. We do not model each fault as a discrete state. We consider the transition of the nominal (normal) hybrid system only. This goal is achieved by decoupling the unknown fault input from the continuous dynamics corresponding to each discrete state.
A. Individual Robust Estimator Design

We first design a bank of robust estimators (RE), each matched to a discrete state as discussed in Section II-B. The task is actually a constrained optimization problem: in a state estimator for a LTI system \[\{A_q, B_q, C_q, D_q, F_q \mid q \in \mathcal{Q}\}\], we want to minimize the estimation error, \(tr[P_q(k+1)]\) of the estimate \(\hat{x}_q(k+1)\) by choosing proper \(L_q\) and \(T_q\) matrices which also decouple the unknown fault input from the error dynamics. Assumptions 1 and 2 guarantee that (13) and (14) are solvable. The solution of (13) can be parameterized as:

\[L_q = F_q(C_q F_q)^+ + L_q \Psi_q\] (15)

where \((C_q F_q)^+ = [C_q F_q]^T C_q F_q]^{-1}[C_q F_q]^T\) denotes the Moore-Penrose pseudoinverse of the matrix \(C_q F_q\). \(\Psi_q\) is a full row rank matrix whose rows span the space which is orthogonal to the space spanned by the columns of \(C_q F_q\). We can assume \(\Psi_q = \Psi_q(I - C_q F_q (C_q F_q)^+)\), where \(\Psi_q \in \mathbb{R}^{n \times (p-m)}\) is any full row rank matrix of appropriate dimension. In (15), \(\bar{L}_q\) is a design parameter.

From (14), \(T_q\) can be written as:

\[T_q = \Psi_q\] (16)

From the estimator dynamics in (8) and (9) at each time \(k\), the covariance matrix of the estimate \(\hat{x}_q(k+1)\) can be written as:

\[P_q(k+1) = (A_q - L_q C_q A_q) P_q(k) (A_q - L_q C_q A_q)^T + (I - L_q C_q) Q(k)(I - L_q C_q)^T + L_q R L_q^T\] (17)

We define:

\[\tilde{P}_q(k) := A_q P_q(k) A_q^T + Q(k)\] (18)

and

\[S_q(k) := C_q \tilde{P}_q(k) C_q^T + R(k)\] (19)

Then (17) can be rewritten as:

\[P_q(k+1) = \tilde{P}_q(k) - L_q C_q \tilde{P}_q(k) - \tilde{P}_q(k) C_q^T + L_q S_q(k) L_q^T\] (20)

Substituting (15) into (20) and letting \(\partial tr[P_q(k+1)]/\partial \bar{L}_q = 0\), we get:

\[\bar{L}_q = [\tilde{P}_q(k) - C_q^T \Psi_q^{-1} - F_q (C_q F_q)^+ S_q(k) \Psi_q^{-1} - \Psi_q S_q(k) \Psi_q^{-1}]^{-1}\] (21)

Thus, we can find \(\bar{L}_q\) which minimizes \(tr[P_q(k+1)]\). Note that in (21), \(\bar{L}_q\) is actually changing with time \(k\). Substituting (21) into (15), we get filter gain matrix \(L_q(k)\) which minimizes \(tr[P_q(k+1)]\) and decouples \(f(k)\) from the error dynamics:

\[L_q(k) = F_q (C_q F_q)^+ + [\tilde{P}_q(k) - C_q^T \Psi_q^{-1} - F_q (C_q F_q)^+ S_q(k) \Psi_q^{-1} - \Psi_q S_q(k) \Psi_q^{-1}]^{-1}\Psi_q\] (22)

B. Robust Hybrid Estimation Algorithm

In this section, we present our robust hybrid estimation algorithm based on the results in the previous sections. Our algorithm is named as the Unknown Fault Input Hybrid Estimation (UFIHE) algorithm. To estimate the continuous state and the discrete state of the hybrid system (1) and (2) with the discrete transition model (3) or (4), we use a bank of robust estimators (RE), each matched to a discrete state. The dynamics of each robust estimator is given by (8) and (9). The filter gain \(L_q(k)\) and the matrix \(T_q\) at each time \(k\) are given by (21) and (16). Then, we combine the robust estimators to complete the hybrid estimation algorithm. Our algorithm uses the “mixing” step which is similar to the IMM algorithm [8] to keep the exponentially growing computational complexity constant. As the fault is decoupled from the error dynamics of each estimator, the estimation accuracy of the hybrid estimation algorithm is not influenced by the unknown fault input. For the Markov-jump transition model, the \textit{a priori} knowledge about the discrete state transition is given by the Markov transition matrix, which assumes the transition probabilities \(Pr\{q(k+1) = jq(k) = i, Z^k\}\) to be constant\(^1\). In the state-dependent transition model, the discrete state transition probabilities are computed from the continuous state estimates. Finally, the estimates of the continuous state is given by a weighted sum of the output of each robust estimator, and the discrete state estimate is given by the discrete state of the highest probability among all discrete states. The structure of our algorithm is described as follows:

\textbf{Step 1. Mixing (merging) probabilities:} The mixing probabilities \(Pr\{q(k) = jq(k+1) = j, Z^k\}\) for all \(i, j \in \mathcal{Q}\) are defined as:

\[\mu_{ji}(k) := Pr\{q(k) = j | q(k+1) = j, Z^k\}\] (23)

By the Bayes’ Theorem,

\[Pr\{q(k) = jq(k+1) = j, Z^k\} = \frac{1}{c_j} Pr\{q(k+1) = jq(k) = i, Z^k\} p(q(k)|Z^k)\] (24)

where \(c_j\) is a normalizing constant. To evaluate (23) and (24), we use the following approach to compute the discrete state transition probability \(Pr\{q(k+1) = jq(k) = i, Z^k\}\):

1) Markov-jump transition: The discrete state transition probability in (24) can be written as:

\[Pr\{q(k+1) = jq(k) = i, Z^k\} = \Gamma_{ij} = \text{const.}\] (25)

2) State-dependent transition: With the stochastic linear guard condition given in (7), we compute the discrete state transition probability in (24) as

\(^1\)Through out this paper, we use \(Pr\{\bullet\}\) to denote the probability of an event and \(p\{\bullet\}\) for the probability density function (pdf).
Step 2. Initial conditions for each RE: At each time

$k$, we approximate the initial condition of each robust estimator by a single Gaussian distribution. The initial conditions (mean $\hat{x}_j(0)$ and covariance $P_j(0)$) for the $j$-th robust estimator are given by:

$$\hat{x}_j(0) = \sum_{i=1}^{n_j} \mu_{ji}(k) \hat{x}_i(k)$$

$$P_j(0) = \sum_{i=1}^{n_j} \mu_{ji}(k) \{ P_i(k) + \Sigma_j \}$$

Step 3. Mode-matched filtering: Each RE computes the posterior mean and covariance $\hat{x}_{j1}(k+1)$, $P_{j1}(k+1)$ conditioned on $q(k) = j$. The structure of REs is given by (8) where the filter gain $L_j(k)$ is calculated from (22) at each time $k$. The REs also give residuals $r_j(k+1)$ by (9) where the residual weighting matrix $T_j$ is parameterized by (16) and the residual covariance matrix $S_j(k+1)$ is given by (19).

Step 4. Discrete-state pdf update: For each RE, the likelihood function $p[z(k+1)|q(k+1) = j, Z^k]$ is

$$\Lambda_j(k+1) := p[z(k+1)|q(k+1) = j, Z^k] = N_p(r_j(k+1); 0, S_j(k+1))$$

where $r_j$ and $S_j$ are defined in (9) and (19) respectively. By the Bayes’ Theorem, the probability $\alpha_j(k+1|k+1) := Pr\{q(k+1) = j|Z^{k+1}\}$ is given by

$$\alpha_j(k+1|k+1) = \frac{1}{\delta_j} \Lambda_j(k+1)$$

where $\delta_j$ is a normalizing constant. Substituting (29) into (30) and using the total probability theorem on the term $Pr\{q(k+1) = j|Z^k\}$ in (30), we get

$$\alpha_j(k+1|k+1) = \frac{1}{\delta_j} \Lambda_j(k+1)$$

$$\sum_{i=1}^{n_j} Pr\{q(k+1) = j|q(k) = i, Z^k\} Pr\{q(k) = i|Z^k\}$$

Step 5. Output: By the total probability theorem, the continuous state pdf at time $k+1$ is given by

$$p[x(k+1)|Z^{k+1}] = \sum_{j=1}^{n_d} p[x(k+1)|q(k+1) = j, Z^{k+1}]$$

We approximate the sum of the $n_d$ terms in (32) via moment matching by a single Gaussian pdf [10]:

$$p[x(k+1)|Z^{k+1}] \approx N_a(x; \hat{x}(k+1), P(k+1))$$

where

$$\hat{x}(k+1) = \sum_{j=1}^{n_d} \alpha_j(k+1|k+1) \hat{x}_j(k+1|k+1)$$

$$P(k+1) = \sum_{j=1}^{n_d} \{ P_j(k+1|k+1) + \alpha_j(k+1|k+1)$$

$$[\hat{x}_j(k+1|k+1) - \hat{x}(k)] [\hat{x}_j(k+1|k+1) - \hat{x}(k)]^T \}$$

The probability that discrete state is $j$ conditioned on the observation up to time $k+1$ is given by

$$Pr\{q(k+1) = j|Z^{k+1}\} = \alpha_j(k+1|k+1)$$

and the discrete state estimate is

$$\hat{q}(k+1) = \arg \max \{ q(k+1) = j|Z^{k+1} \}$$

IV. SIMULATIONS

In this section, we demonstrate the performance of our algorithm with an example of a vertical take-off and landing (VTOL) aircraft. We use the SLHS with state-dependent transitions to model the aircraft’s longitudinal flight dynamics with multiple flight modes [11]. In the simulation, actuator faults are added to the system, including the fault of the collective input and longitudinal cyclic input. Our algorithm does not need any a priori information about the magnitude of the fault input and we consider different and time-varying degrees of the actuator faults.

A. Aircraft Model

In this example, we consider the nonlinear longitudinal dynamics of a VTOL aircraft [11]: some parameters are dependent on the continuous state $x(k)$. Here we use the SLHS model with state-dependent transitions to describe the behavior of the VTOL aircraft longitudinal dynamics by dividing it into three flight modes: Hovering Mode (HM), Cruise Mode (CM) and Transition Mode (TM). TM describes the aircraft’s transition between HM and CM. We have two control inputs: collective and longitudinal cyclic, both of which are associated with the blades of the aircraft’s rotor(s). For each flight mode, the evolution of the continuous dynamics of the aircraft is given by the following difference equations:

$$x(k) = A_{q(k)} x(k-1) + B_{q(k)} u(k) + F_{q(k)} f(k) + w(k)$$

$$z(k) = x(k) + v(k)$$
where \( x(k) = [x_1 \ x_2 \ x_3 \ x_4]^T \in \mathbb{R}^4 \) is the continuous state in which \( x_1 \) and \( x_2 \) are the horizontal and the vertical velocity, and \( x_3 \) and \( x_4 \) are the pitch angle and the pitch angle rate; \( z(k) \in \mathbb{R}^4 \) is the measurement vector; \( q(k) \in \mathcal{Q} = \{1, 2, 3\} \) is the discrete state corresponding to the three different flight modes: HM, TM and CM, respectively at a given time \( k \); \( u(k) \in \mathbb{R}^2 \) is the control input and \( f(k) \in \mathbb{R}^2 \) is the unknown fault input; \( A_q, B_q \) and \( F_q \) are the system matrices corresponding to each flight mode and \( w(k) \) and \( v(k) \) are white Gaussian noise with zero mean and covariances \( Q(k) = 0.01I \) and \( R(k) = 0.2^2I \). Other system parameters are given in Appendix I.

We model the discrete state transition using the state-dependent transition model: we have two threshold velocity values: \( v_1^* \) and \( v_2^* \) (\( v_1^* < v_2^* \)); when the horizontal velocity is smaller than both \( v_1^* \) and \( v_2^* \), it means that the aircraft is hovering and its continuous dynamics evolution is described by (33) with \( q = 1 \); when the horizontal velocity is between \( v_1^* \) and \( v_2^* \), the aircraft’s behavior is represented by (33) with \( q = 2 \); when the horizontal velocity is beyond both \( v_1^* \) and \( v_2^* \), the aircraft is cruising and we use (33) with \( q = 3 \) to describe the corresponding aircraft dynamics. Figure 1 shows the transition model of the hybrid system for the VTOL aircraft. Note that in the figure, we use the symbol \( \land \) to denote logical “and”.

![Flight mode transition model of the hybrid system for a VTOL aircraft.](image)

Fig. 1. Flight mode transition model of the hybrid system for a VTOL aircraft.

However, due to uncertainties in the dynamics of the aircraft, such as wind gust disturbances and unmodeled dynamics, discrete state transitions are “likely” but not “exactly” to occur when the conditions are satisfied. Therefore, it is reasonable to use the stochastic linear guard conditions presented in Section II-A to model such transitions. We first define the random parameter \( \theta \in \Theta = \mathbb{R}^2 \) in the guard conditions: \( p[\theta] = \mathcal{N}_2(\bar{\theta}; \Sigma_\theta) \) in which we choose \( \bar{\theta} = [v_1^* \ \ v_2^*]^T = [90 \ \ 150]^T \) and \( \Sigma_\theta = \text{diag}(s^2 \ \ s^2) \). The guard conditions are defined as:

\[
G(i, j) = \left\{ \frac{x}{\theta} \in \mathbb{R} \times \Theta, L_j \frac{x}{\theta} \leq 0 \right\} \quad i, j = 1, 2, 3
\]

(35)

where \( L_1, L_2 \) and \( L_3 \) are given by:

\[
L_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}
\]

\[
L_2 = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\]

\[
L_3 = \begin{bmatrix} -1 & 0 & 0 & 1 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\]

**B. Simulation Results**

In the simulation, the aircraft increases its horizontal speed from \( 30 \text{knots} \) to about \( 180 \text{knots} \) and performs flight mode transitions from HM (Mode 1) to TM (Mode 2) and to CM (Mode 3) sequentially. The aircraft longitudinal dynamics is unstable naturally, so we apply a control sequence \( u(k) \), \( k = 1, 2, \cdots \) to stabilize the aircraft as well as to manipulate its speed such that the aircraft can enter the desired cruise mode eventually. In the \( 20 \text{sec} \) simulation, we assume that there are two actuator faults occurring: (a) inactive fault: the actuator cannot reach the desired position specified by the input command; and (b) overshoot fault: the actuator acts more than desired. In our simulation, the magnitudes of the two kinds of faults change continuously with time.

![Controller input and unknown fault input](image)

Fig. 2. Controller input and unknown fault input

Figure 2 shows the control input and unknown fault input histories. We can see that there is an overshoot fault with the collective input (Fault 1) which begins at \( 7 \text{ sec} \) and ends at \( 10.5 \text{ sec} \). In addition, there is an inactive fault with the longitudinal cyclic input (Fault 2) which begins at \( 6 \text{ sec} \) and ends at \( 8 \text{ sec} \). The magnitude of the fault inputs varies between \( 6 \) and \( -6 \text{ deg} \).

We implement the proposed UFIHE algorithm and the IMM algorithm. The IMM algorithm has a SLHS model with 12 discrete states. Table I summarizes all the discrete states (modes) considered by the IMM algorithm. The IMM algorithm assumes a constant transition matrix. Also, it is assumed that we have the a priori knowledge that Fault 1 stays at \( 6 \text{ deg} \) and Fault 2 stays at \( -6 \text{ deg} \) when they happen. Thus, each fault is modeled as a discrete state. For example, Mode 2 in Table I denotes the case in which the aircraft is in the HM flight mode and the collective fault input is \( 6 \text{ deg} \) while there is no longitudinal cyclic fault input.

<table>
<thead>
<tr>
<th>Flight mode</th>
<th>Fault combination</th>
</tr>
</thead>
<tbody>
<tr>
<td>HM</td>
<td>Mode 1 Mode 2 Mode 3</td>
</tr>
<tr>
<td>TM</td>
<td>Mode 5 Mode 6 Mode 7</td>
</tr>
<tr>
<td>CM</td>
<td>Mode 9 Mode 10 Mode 11</td>
</tr>
</tbody>
</table>

**TABLE I**

**DISCRETE STATE (MODE) OF THE IMM ALGORITHM**

We perform the Monte Carlo simulation with 100 simulation runs and compare the root-mean-square (RMS) of the horizontal velocity estimation errors of the two algorithms in Figure 4. We can see that when the faults occur (from \( 6 \text{ sec} \) to \( 10.5 \text{ sec} \)), the UFIHE algorithm yields better estimation accuracy, because it is robust to unknown fault input. Even though the IMM algorithm has included the faulty modes...
into its multiple model set, there is still mode mis-matching when the fault inputs do not stay at the predetermined values. During the normal operation, the IMM algorithm can identify the true mode as well as the UFIHE algorithm, however, as a robust estimation algorithm, the UFIHE does not utilize the information from some subspaces of the residual space which may be corrupted by fault inputs. Therefore, the likelihood given by UFIHE is not as accurate as that by the IMM algorithm. This is the price we have to pay. Table II shows the statistics of 100 simulation runs for the two algorithms.

<table>
<thead>
<tr>
<th>TABLE II</th>
<th>COMPARISON OF PERFORMANCE: STATISTICS OF 100 SIMULATION RUNS</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Estimation algorithm</strong></td>
<td><strong>UFIHE</strong></td>
</tr>
<tr>
<td>RMS velocity error (knots)</td>
<td>0.1515</td>
</tr>
<tr>
<td>Overall average</td>
<td>0.0616</td>
</tr>
<tr>
<td>Average during the fault</td>
<td>0.0622</td>
</tr>
<tr>
<td>Average no. of discrete state estimation error</td>
<td>0.58</td>
</tr>
<tr>
<td>Total computation time of 100 runs (sec)</td>
<td>184.00</td>
</tr>
</tbody>
</table>

V. CONCLUSIONS

In this paper, we have proposed a robust hybrid estimation (HE) algorithm which is insensitive (robust) to unknown continuous fault inputs. Our algorithm can be applied to the Stochastic Linear Hybrid System (SLHS) with Markov-jump transitions or state-dependent-jump transitions. To demonstrate the performance of our algorithm, we have applied it to state estimation for a vertical take-off and landing (VTOL) aircraft. Simulation results show that the continuous and discrete state estimates of our algorithm is robust to the unknown fault input.

VI. ACKNOWLEDGEMENT

The authors would like to acknowledge that this work is supported by NSF CAREER Award CNS-0746299 and to thank Dr. Helen Gill for her support.

APPENDIX I

The continuous dynamics of the aircraft without fault inputs is given by

$$x(t) = Ax(t) + Bu(t) + w(t)$$  (36)

The values of the matrices in (36) are given by [11]. Some parameters in $A$ and $B$ are changing with the horizontal speed $V_h$. We discretize the system with sampling time $T_s = 0.1s$ at three points: $V_h = 60, 110$ and $170 \text{knots}$. To model the actuator fault, we set $F_q = B_q$ for each $q$, and get the following difference equation:

$$x(k+1) = A_q x(k) + B_q u(k) + F_q f(k) + w(k)$$  (37)

where $q = 1, 2, 3$ and

$$A_1 = \begin{bmatrix} 0.9641 & 0.0025 & -0.0004 & -0.0452 \\ 0.0044 & 0.9039 & -0.0118 & -0.3826 \\ 0.0095 & 0.0061 & 0.9322 & 0.0101 \\ 0.0005 & 0.0003 & 0.0966 & 1.0005 \end{bmatrix}, \quad B_1 = F_1 = \begin{bmatrix} 0.0434 \\ 0.0965 \\ -0.5325 \\ 0.0269 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} 0.9641 & 0.0025 & -0.0004 & -0.0452 \\ 0.0044 & 0.9038 & -0.0118 & -0.3843 \\ 0.0096 & 0.0063 & 0.9379 & 0.1223 \\ 0.0005 & 0.0013 & 0.0967 & 1.0063 \end{bmatrix}, \quad B_2 = F_2 = \begin{bmatrix} 0.0437 \\ 0.2932 \\ -0.5297 \\ 0.0268 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} 0.9641 & 0.0025 & -0.0004 & -0.0452 \\ 0.0044 & 0.9036 & -0.0118 & -0.3843 \\ 0.0096 & 0.0065 & 0.9435 & 0.2350 \\ 0.0005 & 0.0024 & 0.0969 & 1.0120 \end{bmatrix}, \quad B_3 = F_3 = \begin{bmatrix} 0.4898 \\ -0.7249 \\ -0.5227 \\ 0.0214 \end{bmatrix}$$

REFERENCES


