

STOCHASTIC HYBRID SYSTEMS MODELING AND ESTIMATION WITH
APPLICATIONS TO AIR TRAFFIC CONTROL

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Jooyoung Lee

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

August 2019

Purdue University

West Lafayette, Indiana

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF DISSERTATION APPROVAL

Dr. Inseok Hwang, Chair

School of Aeronautics and Astronautics

Dr. Martin Corless

School of Aeronautics and Astronautics

Dr. Dengfeng Sun

School of Aeronautics and Astronautics

Dr. Jianghai Hu

School of Electrical and Computer Engineering

Approved by:

Dr. Weinong Wayne Chen

Head of the School Graduate Program

This dissertation is dedicated to my family.

ACKNOWLEDGMENTS

First of all, I would like to thank my advisor, Professor Inseok Hwang, for his constant guidance and invaluable advice during my Ph.D. study. I greatly appreciate the ample opportunities and supports he provided me, from which I have experienced the various roles required in my career goal and achieved balanced growth in every required aspect. Thanks to his patience and encouragement, I could manage many difficulties during my Ph.D. study.

I would like to thank my committee members, Professor Martin Corless, Professor Jianghai Hu, and Professor Dengfeng Sun, for their invaluable comments and suggestions on my research. I would like to express my special gratitude to Professor Martin Corless for his enthusiastic encouragement which gave me a willingness to keep moving forward during times of difficulties.

I appreciate the thoughtful and generous supports provided by Jeongsong Culture Foundation and Zonta International Foundation. They allowed me to concentrate on my Ph.D. study and to successfully carry out independent researches.

I would like to acknowledge my mentors during my bachelor's degree at Inha University, Professor Oh Kyu Kwon and Professor Keeyoung Choi, and my advisor during my master's degree at Seoul National University, Professor Youdan Kim, for their invaluable advice and encouragement on my Ph.D. study.

I am grateful to my friends for their steadfast friendship and kindness. I would like to extend my heartfelt thanks to my family, especially my parents, for being my greatest supporters with unconditional love and prayers. Last but not the least, I would like to give my sincerest gratitude to my husband, Sangjin, for being my best supporter with his tremendous love, patience, and efforts.

Thank God for being with me always and caring me with your endless love and countless blessings. I give all praise, honor and glory to you, my almighty God.

TABLE OF CONTENTS

	Page
LIST OF TABLES	vii
LIST OF FIGURES	viii
ABSTRACT	x
1 INTRODUCTION	1
1.1 Background and Motivations	1
1.2 Objectives and Contributions	3
1.2.1 Algorithm Development	3
1.2.2 Applications	4
1.3 Outline of Thesis	5
2 NONLINEAR HYBRID STATE ESTIMATION FOR JUMP MARKOV SYS- TEMS USING GAUSSIAN SUM FILTERS	6
2.1 Background and Motivations	6
2.2 Problem Formulation	9
2.3 Nonlinear State Estimation Algorithms for Jump Markov System based on Gaussian Sum Approximation	11
2.4 Numerical Simulation	16
3 NONLINEAR HYBRID STATE ESTIMATION FOR A GENERAL CLASS OF STOCHASTIC HYBRID SYSTEMS USING GAUSSIAN SUM FILTERS	25
3.1 Background and Motivations	25
3.2 Problem Formulation	26
3.3 Nonlinear Hybrid State Estimation based on Gaussian Sum Approxi- mation	27
3.4 Numerical Simulation	37
4 APPLICATION TO AIR TRAFFIC CONTROL: ESTIMATED TIME OF ARRIVAL PREDICTION	41
4.1 Background and Motivations	41
4.2 Stochastic Hybrid System Model for Aircraft in the Descent Flight Procedure	43
4.2.1 Continuous State Dynamics of Aircraft	46
4.2.2 Flight Mode Transitions	52
4.3 Aircraft Tracking and ETA Prediction Algorithm	55
4.4 Numerical Simulations	56
4.4.1 Example 1: CDA Procedure	57

	Page
4.4.2 Example 2: Conventional Approach Procedure	62
5 APPLICATION TO UAS TRAFFIC MANAGEMENT: UAS SURVEIL- LANCE IN LOW-ALTITUDE AIRSPACE WITH GEOFENCING	67
5.1 Background and Motivations	67
5.2 Constrained Stochastic Hybrid System Model for UAS	70
5.2.1 Stochastic Linear Hybrid System with Continuous-state-dependent Mode Transition	70
5.2.2 UAS Behavior Modeling using Stochastic Linear Hybrid System	71
5.3 UAS Tracking Algorithm	78
5.4 Numerical Simulation	83
6 SUMMARY	88
REFERENCES	90
VITA	95

LIST OF TABLES

Table	Page
2.1 Steps for the Gaussian components reduction techniques with pruning and merging	14
2.2 Performance comparison (500 Monte Carlo runs)	21
2.3 Average number of the Gaussian mixture components (500 Monte Carlo runs)	23
3.1 Mean RMS error for x (500 Monte Carlo simulations)	39
3.2 Discrete state estimation mismatch (500 Monte Carlo simulations)	40
4.1 Medium-range fully procedured flight profile [46]	45
4.2 Nominal flight profile of a Boeing 767 following a CDA procedure	59
4.3 Comparison of ETA prediction accuracy for Example 1: ETA error for 100 runs	61
4.4 Nominal flight profile of a Boeing 767 following a conventional approach procedure	64
4.5 Comparison of ETA prediction accuracy for Example 2: ETA error for 100 runs	66
5.1 Set of guard conditions of left turn at FMCP _{ij}	74
5.2 Set of guard conditions at FMCP ₁₃	75
5.3 Comparison of estimation performance (100 Monte Carlo runs)	86

LIST OF FIGURES

Figure	Page
2.1 An illustration of the jump Markov system with two modes	10
2.2 Structure of the proposed algorithm	17
2.3 Comparison of state estimation performance (500 Monte Carlo runs) . . .	20
2.4 Approximation results of the mixed posterior pdf (at time step $k = 1$) . . .	22
2.5 Evolution of the posterior pdf of the state for a single run	24
3.1 The overall structure of the proposed algorithm	36
3.2 Actual history of x and measurements z for a single run ($x(0) > 0$)	39
3.3 RMS estimation error for x (500 Monte Carlo simulations)	40
4.1 Different phases during a flight (source: FAA)	44
4.2 Discrete mode sequence during the descent stage	46
4.3 Typical flight profile for descent, approach, and landing phases	53
4.4 Structure of the proposed algorithm	57
4.5 Nominal trajectory of a Boeing 767 along a CDA procedure	58
4.6 Comparison of tracking accuracy for Example 1: root-mean-square values of state estimation error for 100 runs	60
4.7 Nominal trajectory of a Boeing 767 along a conventional approach procedure	63
4.8 Comparison of tracking accuracy for Example 2: root-mean-square values of state estimation error for 100 runs	65
5.1 Illustration of geofence given a set of waypoints (WPs)	68
5.2 Flight mode transition [63]	73
5.3 Left turn transition	74
5.4 Mode transition of acceleration from CV (mode 1) to CA (mode 3)	75
5.5 Circle between two segments	77
5.6 Structure of the proposed UAS tracking algorithm (CSDTHE)	82
5.7 Flight plan of the UAS [65]	83

Figure	Page
5.8 A left turn example around WP1	84
5.9 Actual and estimated trajectories of the UAS (a single run)	85
5.10 Comparison of mode-estimation accuracy (a single run)	86
5.11 RMS position and velocity estimation errors with 100 Monte Carlo runs . .	87

ABSTRACT

Lee, Jooyoung Ph.D., Purdue University, August 2019. Stochastic Hybrid Systems Modeling and Estimation with Applications to Air Traffic Control. Major Professor: Inseok Hwang.

Various engineering systems have become rapidly automated and intelligent as sensing, communication, and computing technologies have been increasingly advanced. The dynamical behaviors of such systems have also become complicated as they need to meet requirements on performance and safety in various operating conditions. Due to the heterogeneity in its behaviors for different operating modes, it is not appropriate to use a single dynamical model to describe its dynamics, which motivates the development of the stochastic hybrid system (SHS). The SHS is defined as a dynamical system which contains interacting time-evolving continuous state and event-driven discrete state (also called a mode) with uncertainties. Due to its flexibility and effectiveness, the SHS has been widely used for modeling complex engineering systems in many applications such as air traffic control, sensor networks, biological systems, and etc.

One of the key research areas related to the SHS is the inference or estimation of the states of the SHS, which is also known as the hybrid state estimation. This task is very challenging because both the continuous and discrete states need to be inferred from noisy measurements generated from mixed time-evolving and event-driven behavior of the SHS. This becomes even more difficult when the dynamical behavior or measurement contains nonlinearity, which is the case in many engineering systems that can be modeled as the SHS.

This research aims to 1) propose a stochastic nonlinear hybrid system model and develop novel nonlinear hybrid state estimation algorithms that can deal with the aforementioned challenges, and 2) apply them to safety-critical applications in

air traffic control systems such as aircraft tracking and estimated time of arrival prediction, and unmanned aircraft system traffic management.

1. INTRODUCTION

1.1 Background and Motivations

Many systems have been increasingly automated and complicated as they have been equipped with advanced sensing, communication, and computing components. In order to assure its safe and efficient operation, it is very important to infer the accurate knowledge of the system's state from various types of noisy measurements, and this requires to develop appropriate dynamical models for the system and corresponding state estimation methods.

One of motivating examples is air traffic management system where aircraft's state information such as position, velocity, flight mode, etc. are crucial for safe operation of the national airspace system. Under the development of the advanced communication, navigation, and surveillance systems, various types of real-time data can be exchanged and shared among controllers and aircraft, which include aircraft's on-board sensing data, flight intent information, airspace and weather conditions, etc [1]. This information is then integrated to precisely extract the state information for the complex behavior of aircraft with navigation uncertainties. This task is, however, very challenging due to 1) interacting physical and logical behaviors of the aerospace systems (e.g., the characteristics of aircraft's physical behavior change according to different flight modes) and 2) heterogeneity of measurement data consisting of descriptive (e.g., published flight plan), categorical (e.g., binary sensors) and continuous (e.g., barometric pressure) information. These challenges have received much attention not only in aerospace systems but also in complex systems in other engineering fields such as chemical process systems, robotic systems, etc. As illustrated, many systems have heterogeneity in its dynamical behaviors and/or measurements, and therefore it is difficult to describe them by using only a single dynamical model,

which motivates the development of multiple model dynamical system also known as hybrid system (or stochastic hybrid system if uncertainties are considered).

The stochastic hybrid system (SHS) is defined as a dynamical system which contains interacting time-evolving continuous state and event-driven discrete state (also called a mode) with uncertainties [2]. The discrete state represents which operating condition the system is running on, and its transition is usually modeled by the Markov process. The continuous state describes the physical motion of the system given an operating mode (i.e., discrete state), and its evolution over time is characterized by stochastic difference or differential equations. Due to its flexibility and effectiveness, the SHS has been widely used for modeling complex engineering systems in many applications such as air traffic control [3, 4], sensor networks [5], biological systems [6], human-machine interaction [7], space surveillance [8], and etc.

In general, the estimation of the state of the SHS (called hybrid state estimation) is not straightforward due to the following reasons. First, since the exact discrete state (i.e., which mode the system is operating at a given time) is usually not known, multiple hypotheses need to be maintained over all possible histories of the discrete state to obtain the optimal solution, which is impractical due to the exponentially increasing number of hypotheses over time. Second, the nonlinearity and/or non-Gaussianity in the continuous state dynamics would make the estimation problem even more difficult as well-known approximate filters (e.g., single Gaussian approximation filters) could not accurately represent the actual non-Gaussian distribution. Lastly, the discrete state transition can be in general dependent on the continuous state. When computing discrete state transition probabilities, the dependency causes numerical integration that is intractable to compute.

Recalling the fact that the estimation of state is crucial for safe and efficient operation of the system, it is necessary to develop hybrid state estimation algorithms that can effectively handle the challenges discussed above.

1.2 Objectives and Contributions

The contributions of this research are twofold: 1) theoretical development of new nonlinear hybrid state estimation algorithms that can deal with the aforementioned difficulties, and 2) application of the developed algorithms to various state estimation problems in air traffic control systems.

1.2.1 Algorithm Development

First, a nonlinear state estimation algorithm for the jump Markov system, which is a type of the SHS that considers the discrete mode transition with constant probabilities irrespective of the continuous state evolution, is proposed using Gaussian sum approximation that can approximate any probability density function (pdf) accurately with a set of Gaussians. Gaussian components reduction techniques based on pruning and merging are applied to the mixed posterior pdf in order to deal with an exponentially increasing number of Gaussian mixture components while minimizing information loss. It has been demonstrated that the proposed algorithm produces the better approximation of the posterior pdf of the state and thus improves the state estimation accuracy.

Second, the proposed hybrid state estimation algorithm based on Gaussian sum approximation is extended to a class of nonlinear stochastic hybrid systems where the discrete state transitions are governed by stochastic guard conditions dependent on the continuous state. Finding an exact solution to the corresponding hybrid state estimation problem is very challenging because the discrete state transition probabilities need to be computed by integrating the probability density function of the continuous state which is non-Gaussian in general due to the nonlinearity in the system. To deal with the difficulty, it is proposed to use the Gaussian mixture that can effectively approximate the non-Gaussian distribution while preserving the elegant analytical properties of Gaussians. Based on those properties, a succinct and closed-form expression is derived for the discrete state transition probabilities that

account for the continuous-state-dependent guard conditions. In addition, to avoid the exponentially growing number of hypotheses in the discrete state history, the interacting multiple model approach is used where a mixing technique is implemented to keep the number of hypotheses constant. It has been shown that the evolution of the probability density function, which is non-Gaussian due to the nonlinearity and continuous-state-dependent mode transition, can be accurately described by the proposed algorithm.

1.2.2 Applications

The proposed hybrid state estimation algorithms have been applied to safety-critical applications in air traffic control systems: 1) aircraft tracking and estimated time of arrival prediction, and 2) unmanned aircraft system traffic management.

First, an aircraft trajectory estimation and estimated time of arrival (ETA) prediction algorithm is proposed based on the SHS modeling and hybrid state estimation. Given a flight plan or procedure, the behavior of an aircraft in the descent phase can be described by both discrete transitions between flight modes and the aircraft's continuous motion corresponding to a specific flight mode. In this sense, the proposed algorithm uses a stochastic nonlinear hybrid system to accurately model the aircraft's dynamics during the descent phase along its flight plan. In the proposed stochastic hybrid system, nonlinear dynamics including wind disturbance for the aircraft's continuous motion are derived for each flight mode, and a continuous state-dependent transition model is developed for the flight mode transitions. Using the proposed stochastic hybrid system model, an algorithm which can solve both the trajectory estimation and ETA prediction in a unified framework is proposed based on the State-Dependent-Transition Hybrid Estimation approach. The proposed algorithm is then demonstrated with several ETA prediction examples.

Second, a Unmanned Aircraft Systems (UAS) tracking algorithm is proposed based on the SHS modeling to improve the safety and efficiency of UAS operation

under the UAS traffic management (UTM). To account for the complex behavior of UAS with predefined flight plans and geofence (virtual boundary), a constrained stochastic hybrid system model is proposed, in which the continuous motions of a UAS with geofence can be described using a set of stochastic continuous state dynamics with state constraints and its flight plan can be incorporated as the continuous-state-dependent discrete mode transitions. Using the proposed model, the constrained state-dependent-transition hybrid estimation (CSDTHE) algorithm is developed for UAS tracking, which can explicitly incorporate the constraints on the motion of UAS imposed by geofence into the hybrid state estimation. The proposed UAS tracking algorithm is demonstrated with an illustrative UTM example for the tracking of a UAS used for delivery in the urban area.

1.3 Outline of Thesis

The thesis is organized as follows. In Chapter 2, a class of nonlinear SHS is introduced, where the discrete state transitions are based on constant transition probabilities, and the corresponding nonlinear hybrid state estimation algorithm is proposed based on Gaussian sum approximation and Gaussian components reduction. In Chapter 3, the work in Chapter 2 is extended to a more general class of nonlinear SHS, where the discrete state evolves based on stochastic guard conditions dependent on the continuous state. In Chapters 4 and 5, the proposed nonlinear SHS modeling and hybrid state estimation algorithms are applied to 1) aircraft tracking and ETA prediction and 2) unmanned aircraft system traffic management, respectively. A summary and future works are discussed in Chapter 6.

2. NONLINEAR HYBRID STATE ESTIMATION FOR JUMP MARKOV SYSTEMS USING GAUSSIAN SUM FILTERS

In this chapter, a mathematical model for a class of stochastic nonlinear hybrid systems, called jump Markov systems, is first introduced. The hybrid state estimation is then discussed and the corresponding estimation algorithm based on Gaussian sum approximation is proposed.

2.1 Background and Motivations

The jump Markov system (JMS) is a dynamical system which involves interaction of time-evolving states and event-driven modes in a Markov process [2]. The JMS has been widely used to describe practical systems with multiple modes in various fields such as target tracking [9], air traffic management [3], sensor networks [5], biological processes [6], etc. The state estimation task for the JMS is challenging due to the need to jointly estimate the state and mode from a set of noisy measurement data, so it demands a more sophisticated approach than that for systems with a single dynamic model. The standard approaches to solve this problem are the multiple model approaches based on Bayesian framework [10] that are further categorized into two classes depending on the type of the JMS considered in their formulations.

The first class is the linear state estimation for the JMS in which the system behavior is described by multiple linear dynamic models switched according to a Markov chain and observed through linear measurements. Many state estimation algorithms in this class have been developed based on the multiple model Kalman filters approach where the state estimates and associated covariances are computed

from a bank of Kalman filters, each of which matches to each mode history, and then combined by applying the total probability theorem, yielding the posterior probability density function (pdf) of the state as a Gaussian mixture [10]. However, as the number of the mode histories grows with time, this approach needs an exponentially increasing number of Kalman filters, which makes the optimal estimation impractical. Hence, the suboptimal techniques such as hypothesis pruning and merging were introduced to reduce the number of hypotheses (i.e., the number of mode histories) and successfully applied to infer the sufficiently accurate states. The representative suboptimal algorithms are the first-order generalized pseudo-Bayesian (GPB1) algorithm, the second-order generalized pseudo-Bayesian (GPB2) algorithm, and the interacting multiple model (IMM) algorithm [10]. The GPB n algorithm ($n = 1, 2$) considers the possible mode histories only in the last n time steps. The IMM algorithm considers the possible mode histories in the two most recent time steps and merges the hypotheses by applying the mixing technique at the beginning of each filtering cycle, which achieves comparable performance as the GPB2 algorithm with efficient computation as the GPB1 algorithm. The IMM algorithm has received significant attention in many applications [8, 11–14] because of its good compromise between accuracy and computational cost.

The second class is the nonlinear state estimation for the JMS in which nonlinearities are exhibited in the system dynamics and/or measurement model. In general, for the most nonlinear state estimation problems, it is difficult to explicitly determine the posterior pdf of the state in a closed form, which led to the development of approximate nonlinear filters that use the approximate description of the state with a tractable form. Many nonlinear state estimation algorithms in this class have been developed in the IMM algorithm framework by employing the approximate nonlinear filters as the mode-matched filters. The most common algorithms are the IMM-EKF algorithm [15] and the IMM-UKF algorithm [16] that substitute the Kalman filters in the IMM algorithm scheme with the extended Kalman filters (EKFs) or the unscented Kalman filters (UKFs) which approximate the posterior pdf of the state as

a single Gaussian density [17]. However, these algorithms might suffer from performance degradation under high nonlinearities as the single Gaussian approximation could result in significant loss of the information contained in the true non-Gaussian pdf. A filtering algorithm that could work for highly nonlinear systems is the particle filter (PF) which uses numerical approximations of the pdf of the state represented by a set of random particles with associated weights [18]. The PF can be directly applied to the JMS by having a single particle represent both the state and mode [19]. However, this algorithm might confront the problem of numerical instability in computations caused by particle shortage as the number of the particles resided in each mode is not controlled. To remedy this problem, the algorithms that maintain a fixed number of particles in each mode have been proposed by adopting the PFs in the IMM algorithm framework (called the IMM-PF) [20, 21]. However, these algorithms still have the major disadvantage of extensive computational requirements, which makes them computationally infeasible for high-dimensional systems.

The Gaussian sum filter (GSF), a well-known approximate nonlinear filter, has been developed based on the idea of approximating the pdf of the state with a weighted sum of Gaussian density functions, i.e., Gaussian sum approximation [22]. It has been successful in representing any complex distributions (e.g., multimodal pdfs) as closely as desired. Whereas several researches on the GSF have been carried out in the areas of nonlinear filtering, data association, localization and mapping, etc. [23–25], only one study [26] has focused on the state estimation problem for the JMS so far. In [26], a Gaussian sum filtering algorithm for the jump Markov nonlinear systems was developed within the framework of the IMM algorithm. The authors proposed a minor Gaussian-set design (MGSD) technique to deal with the issue of an exponentially growing number of the Gaussian mixture components of the mixed posterior pdf over iterations. By the MGSD technique, the mixed posterior pdf represented as a Gaussian mixture with a large number of components is approximated using moment matching and then a new Gaussian sum pdf with a reduced number of the Gaussian components is generated, which effectually limits computational complexity of the

algorithm. However, while this algorithm achieves a balance between computational complexity and performance, it poses a distinct drawback that the MGSD technique involves significant loss of meaningful information held in the original mixed posterior pdf. In addition, the new Gaussian sum pdf is restricted to having a constant number of Gaussian components depending on the dimension of the state and they are deterministically chosen without considering the physical meaning propagated through the previous mode transitions. This research is motivated to improve the existing algorithm by developing novel Gaussian components reduction techniques in the mixing step of the IMM algorithm. The proposed techniques control the number of the Gaussian mixture components by pruning and merging based on the weight of each Gaussian component, in a way that most informative components are preserved to minimize the loss of meaningful information in the original probability distribution. Detailed description of the proposed techniques will be discussed in a later algorithm section.

2.2 Problem Formulation

We consider a discrete-time jump Markov system which is composed of the state $\mathbf{x}(k) = [x_1(k), \dots, x_n(k)]^T \in \mathbb{R}^n$, the mode $q(k) \in \mathcal{Q} = \{1, 2, \dots, r\}$, and the measurement vector $\mathbf{z}(k) = [z_1(k), \dots, z_p(k)]^T \in \mathbb{R}^p$, where k is the discrete-time index. For each mode $q(k)$, the system dynamics and the measurement equation are given by

$$\begin{aligned}\mathbf{x}(k+1) &= \mathbf{f}_{q(k)}(\mathbf{x}(k)) + \mathbf{w}_{q(k)}(k) \\ \mathbf{z}(k) &= \mathbf{h}_{q(k)}(\mathbf{x}(k)) + \mathbf{v}_{q(k)}(k)\end{aligned}\tag{2.1}$$

where $\mathbf{f}_{q(k)} : \mathbb{R}^n \rightarrow \mathbb{R}^n$ and $\mathbf{h}_{q(k)} : \mathbb{R}^n \rightarrow \mathbb{R}^p$ are (piecewise) smooth bounded nonlinear functions, and $\mathbf{w}_{q(k)}(k)$ and $\mathbf{v}_{q(k)}(k)$ are zero-mean white Gaussian noises with covariances $\mathbf{Q}_{q(k)}$ and $\mathbf{R}_{q(k)}$, respectively. The mode evolution is modeled by a Markov

chain with constant mode transition probabilities. The mode transition probability from $q(k-1) = i$ to $q(k) = j$ is defined by

$$\lambda_{ij} := p(q(k) = j | q(k-1) = i, \mathbf{Z}^{k-1}) \quad (2.2)$$

where $p(\cdot|\cdot)$ denotes a conditional probability and $\mathbf{Z}^{k-1} \equiv \{\mathbf{z}(1), \mathbf{z}(2), \dots, \mathbf{z}(k-1)\}$ denotes a set of measurements up to time $k-1$. Figure 2.1 depicts an example of the described jump Markov system.

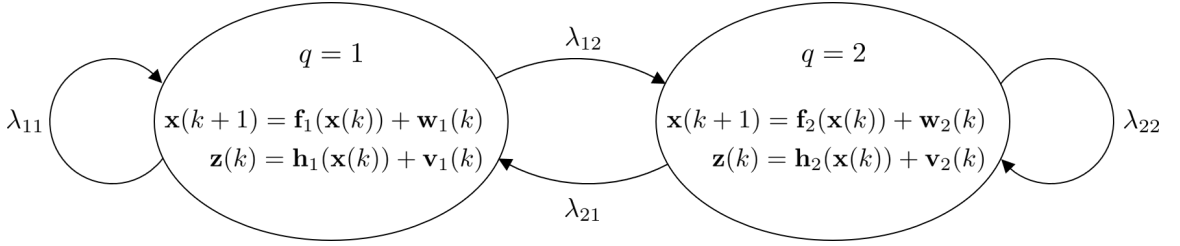


Figure 2.1. An illustration of the jump Markov system with two modes

The state estimation task of the jump Markov system involves computations of the posterior probability distribution of both the state $p(\mathbf{x}(k)|\mathbf{Z}^k)$ and the mode $p(q(k)|\mathbf{Z}^k)$. Using the total probability theorem, $p(\mathbf{x}(k)|\mathbf{Z}^k)$ can be computed by

$$p(\mathbf{x}(k)|\mathbf{Z}^k) = \sum_{i=1}^r p(\mathbf{x}(k)|q(k) = i, \mathbf{Z}^k) p(q(k) = i|\mathbf{Z}^k) \quad (2.3)$$

where $p(\mathbf{x}(k)|q(k) = i, \mathbf{Z}^k)$ is the mode-conditioned probability distribution of $\mathbf{x}(k)$ and $p(q(k) = i|\mathbf{Z}^k)$ is the mode probability for $i = 1, 2, \dots, r$. Then, using the Bayesian framework and Eq. (2.3), the state estimation of the JMS is reformulated in a recursive way that given $p(\mathbf{x}(k-1)|q(k-1) = i, \mathbf{Z}^{k-1})$ and $p(q(k-1) = i|\mathbf{Z}^{k-1})$ for each mode $i \in \mathcal{Q}$ at time $k-1$, $p(\mathbf{x}(k)|q(k) = i, \mathbf{Z}^k)$ and $p(q(k) = i|\mathbf{Z}^k)$ are computed for all the modes $i \in \mathcal{Q}$ using the new measurement vector $\mathbf{z}(k)$ generated at time k . Then, the state estimate and mode estimate can be computed by

$$\begin{aligned} \hat{\mathbf{x}}(k) &:= E[\mathbf{x}(k)|\mathbf{Z}^k] \\ \hat{q}(k) &:= \underset{j}{\operatorname{argmax}} p(q(k) = j|\mathbf{Z}^k) \end{aligned} \quad (2.4)$$

where $E[\cdot|\cdot]$ denotes the conditional expectation of a random variable.

2.3 Nonlinear State Estimation Algorithms for Jump Markov System based on Gaussian Sum Approximation

In this research, new nonlinear state estimation algorithms for the JMS are developed based on the IMM algorithm framework with mixing technique to achieve both good performance and efficient computation. We approximate the non-Gaussian pdf of the state with a weighted sum of Gaussian density functions based on the Gaussian sum approximation and use a bank of Gaussian sum filters (GSFs) for mode-conditioned filtering in order to effectively deal with nonlinearities in the system dynamics and/or measurements. It should be noted that the direct use of the GSFs along with the mixing operation causes an exponentially growing number of the Gaussian mixture components, which brings a challenge on the computational load. To overcome the challenge, we propose Gaussian components reduction techniques which control the number of the components in a Gaussian mixture by pruning and merging based on the facts that the Gaussian components with high weight have more contribution to the mixture and the closely spaced Gaussian components carry similar information. It is observed that the Gaussian components reduction techniques with pruning and merging provide good approximation quality and achieve low computational cost. The details of the proposed algorithms formulated in a recursive way are discussed as follows.

Let us assume that, from the last iteration at time $k - 1$, the mode probabilities $m^i(k - 1) \equiv p(q(k - 1) = i | \mathbf{Z}^{k-1})$, $i = 1, 2, \dots, r$, are computed and the mode-conditioned posterior probability density functions of the state are obtained as:

$$p(\mathbf{x}(k - 1) | q(k - 1) = i, \mathbf{Z}^{k-1}) = \sum_{\alpha=1}^N \xi^{i\alpha}(k - 1) \mathcal{N}(\mathbf{x}(k - 1); \hat{\mathbf{x}}^{i\alpha}(k - 1), \mathbf{P}^{i\alpha}(k - 1)) \quad (2.5)$$

for $i = 1, 2, \dots, r$, where N is the number of the Gaussian components in the mixture, \mathcal{N} denotes the Gaussian distribution with the mean $\hat{\mathbf{x}}^{i\alpha}(k - 1)$ and the covariance

$\mathbf{P}^{i\alpha}(k-1)$ which are computed from the α -th subfilter of the i -th GSF at time $k-1$, and $\xi^{i\alpha}(k-1)$ denotes a weight of each Gaussian distribution which satisfies $\sum_{\alpha=1}^N \xi^{i\alpha}(k-1) = 1$. Then, using the new measurement $\mathbf{z}(k)$ generated at time k , we can compute $p(\mathbf{x}(k)|q(k) = j, \mathbf{Z}^k)$ and $m^j(k)$ for all the modes $j = 1, 2, \dots, r$ as shown in the following steps.

Step 1: Mixing

The mixing probability $m^{ij}(k)$ is computed as:

$$\begin{aligned} m^{ij}(k) &= p(q(k-1) = i | q(k) = j, \mathbf{Z}^{k-1}) \\ &= \frac{p(q(k) = j | q(k-1) = i, \mathbf{Z}^{k-1}) p(q(k-1) = i | \mathbf{Z}^{k-1})}{p(q(k) = j | \mathbf{Z}^{k-1})} \\ &= \frac{\lambda_{ij} m^i(k-1)}{\sum_{l=1}^r \lambda_{lj} m^l(k-1)} \end{aligned} \quad (2.6)$$

where λ_{ij} is the mode transition probability defined in Eq. (2.2). Using the mixing probability, the mixed posterior pdf for the GSF matched to mode j is obtained as:

$$\begin{aligned} p(\mathbf{x}(k-1) | q(k) = j, \mathbf{Z}^{k-1}) &= \sum_{i=1}^r m^{ij}(k) p(\mathbf{x}(k-1) | q(k-1) = i, \mathbf{Z}^{k-1}) \\ &= \sum_{\alpha=1}^N \sum_{i=1}^r m^{ij}(k) \xi^{i\alpha}(k-1) \mathcal{N}(\mathbf{x}(k-1); \hat{\mathbf{x}}^{i\alpha}(k-1), \mathbf{P}^{i\alpha}(k-1)) \end{aligned} \quad (2.7)$$

To control the number of the Gaussian components, the mixed posterior pdf with $r \times N$ Gaussian components in Eq. (2.7) is approximated as a Gaussian mixture with a reduced number of Gaussians by applying the proposed Gaussian components reduction techniques. We first prune insignificant Gaussian components based on the weights by either keeping only a certain number of Gaussian components with the highest weights or discarding Gaussian components with low weights (for the rest of this research, we denote the algorithm using the former pruning as ‘‘Proposed I’’ and the algorithm using the latter as ‘‘Proposed II’’) and then merge them to the remaining components based on distances. The detailed procedure of the Gaussian components reduction techniques is described as follows.

By pruning, among the $r \times N$ Gaussian components in Eq. (2.7) (where the weight of each Gaussian component is $m^{ij}(k) \xi^{i\alpha}(k-1)$), the Gaussian components with N

highest weights are selected in the Proposed I algorithm (here N is a fixed number) or the Gaussian components that have a weight greater than the value of a preset threshold ϵ (e.g., $\epsilon = 0.2$) are selected in the Proposed II algorithm (where the number of the resulting Gaussian components is denoted as N also, but it varies at each iteration). If the sum of the N weights is greater than the value of a prespecified threshold δ (e.g., $\delta = 0.7$), which implies that they approximate the original $r \times N$ Gaussians with less information loss, the weights are normalized and the selected N Gaussian components become the final outcome. If the sum is below the threshold, it might be necessary to consider more Gaussian components in addition to the initially selected N Gaussians to accurately describe the original mixed posterior pdf in Eq. (2.7). In this case, the remaining $(r - 1) \times N$ Gaussian components are further examined to be combined with the initially selected N components through merging. For each of the remaining components with a weight greater than a specified threshold κ (e.g., $\kappa = 0.15$), the distances to the initially selected N Gaussian components are calculated to find the closest component with the minimum distance. If the minimum distance is less than a preset threshold γ (e.g., $\gamma = 0.02$), the corresponding pair of Gaussian components are merged by means of moment matching. The procedure described above is summarized in Table 2.1.

After applying the Gaussian components reduction techniques with pruning and merging, the initial condition for the GSF matched to mode j is obtained as a Gaussian mixture with N components as:

$$p(\mathbf{x}(k-1)|q(k) = j, \mathbf{Z}^{k-1}) = \sum_{\beta=1}^N \xi^{j\beta}(k-1) \mathcal{N}(\mathbf{x}(k-1); \hat{\mathbf{x}}^{j\beta}(k-1), \mathbf{P}^{j\beta}(k-1)) \quad (2.8)$$

where $\xi^{j\beta}(k-1)$, $\beta = 1, \dots, N$, is a new set of Gaussian weights, and $\hat{\mathbf{x}}^{j\beta}(k-1)$ and $\mathbf{P}^{j\beta}(k-1)$ are the corresponding mean and covariance. The mode-conditioned posterior pdf $p(\mathbf{x}(k)|q(k) = j, \mathbf{Z}^k)$ is computed in the j -th GSF using the initial condition for mode j in Eq. (2.8) as described in the following step.

Step 2: Mode-conditioned estimation

Table 2.1. Steps for the Gaussian components reduction techniques with pruning and merging

1. Input

- 1) $r \times N$ Gaussian components in Eq. (2.7)
- 2) Thresholds δ , κ , and γ in Proposed I
(Thresholds ϵ , δ , κ , and γ in Proposed II)

2. Pruning and Merging

Find N components with highest weights in Proposed I

(Find N components with weights $\geq \epsilon$ in Proposed II)

if $\sum\{\text{weights of the } N \text{ components}\} \geq \delta$

re-normalize the weights

return the N components

else

for each of the remaining $(r - 1) \times N$ components

if weight $\geq \kappa$

compute distances to the initially selected N components

find the closest component with minimum distance

if the minium distance $\leq \gamma$

merge to the closest component by moment matching

end

end

end

return the merged N components (with weight re-normalization)

end

3. Output

A set of N Gaussian components

In this research, a bank of r GSFs are used, each of which matches to the dynamics of each mode. The GSF is composed of N subfilters and the subfilter propagates and updates each Gaussian component of the initial condition in Eq. (2.8). The nonlinear filters based on single Gaussian approximation (e.g., EKF, UKF, etc.) can be considered as the subfilters.

For a given mode j , with the initial condition in Eq. (2.8), the mode-conditioned prior distribution $p(\mathbf{x}(k)|q(k) = j, \mathbf{Z}^{k-1})$ is computed through the j -th GSF as:

$$p(\mathbf{x}(k)|q(k) = j, \mathbf{Z}^{k-1}) = \sum_{\beta=1}^N \xi^{j\beta}(k|k-1) \mathcal{N}(\mathbf{x}(k|k-1); \hat{\mathbf{x}}^{j\beta}(k|k-1), \mathbf{P}^{j\beta}(k|k-1)) \quad (2.9)$$

where $\xi^{j\beta}(k|k-1)$ is the same as $\xi^{j\beta}(k-1)$. With the new measurement $\mathbf{z}(k)$, the mode-conditioned posterior distribution $p(\mathbf{x}(k)|q(k) = j, \mathbf{Z}^k)$ is computed in the j -th GSF as:

$$p(\mathbf{x}(k)|q(k) = j, \mathbf{Z}^k) = \sum_{\beta=1}^N \xi^{j\beta}(k) \mathcal{N}(\mathbf{x}(k); \hat{\mathbf{x}}^{j\beta}(k), \mathbf{P}^{j\beta}(k)) \quad (2.10)$$

where the weights are updated as:

$$\xi^{j\beta}(k) = \frac{1}{c} \xi^{j\beta}(k|k-1) \mathcal{N}(\mathbf{z}(k) - \mathbf{h}_j(\hat{\mathbf{x}}^{j\beta}(k|k-1)); \mathbf{0}, \mathbf{S}^{j\beta}(k)) \quad (2.11)$$

where c is a normalizing constant and $\mathcal{N}(\mathbf{z}(k) - \mathbf{h}_j(\hat{\mathbf{x}}^{j\beta}(k|k-1)); \mathbf{0}, \mathbf{S}^{j\beta}(k))$ is the likelihood function of the β -th subfilter.

Step 3: Mode probability update

The mode probability for mode j is updated using Bayes' rule as:

$$\begin{aligned} m^j(k) &= p(q(k) = j | \mathbf{Z}^k) \\ &= \frac{1}{c_n} p(\mathbf{z}(k) | q(k) = j, \mathbf{Z}^{k-1}) p(q(k) = j | \mathbf{Z}^{k-1}) \end{aligned} \quad (2.12)$$

where c_n is a normalizing constant, $p(\mathbf{z}(k) | q(k) = j, \mathbf{Z}^{k-1})$ is the mode-conditioned likelihood function given by:

$$p(\mathbf{z}(k) | q(k) = j, \mathbf{Z}^{k-1}) = \sum_{\beta=1}^N \xi^{j\beta}(k|k-1) \mathcal{N}(\mathbf{z}(k) - \mathbf{h}_j(\hat{\mathbf{x}}^{j\beta}(k|k-1)); \mathbf{0}, \mathbf{S}^{j\beta}(k)) \quad (2.13)$$

and $p(q(k) = j | \mathbf{Z}^{k-1})$ is the prior mode probability for mode j obtained by:

$$p(q(k) = j | \mathbf{Z}^{k-1}) = \sum_{i=1}^r \lambda_{ij} m^i(k-1) \quad (2.14)$$

Step 4: Output

Using the total probability theorem, the state estimate $\hat{\mathbf{x}}(k)$ and its covariance $\mathbf{P}(k)$ are obtained as:

$$\begin{aligned} \hat{\mathbf{x}}(k) &= \sum_{j=1}^r m^j(k) \hat{\mathbf{x}}^j(k) \\ \mathbf{P}(k) &= \sum_{j=1}^r m^j(k) \{ \mathbf{P}^j(k) + [\hat{\mathbf{x}}^j(k) - \hat{\mathbf{x}}(k)][\hat{\mathbf{x}}^j(k) - \hat{\mathbf{x}}(k)]^T \} \end{aligned} \quad (2.15)$$

where the mode-conditioned state estimates $\hat{\mathbf{x}}^j(k)$ and covariance $\mathbf{P}^j(k)$ are computed from the mode-conditioned posterior pdf in Eq. (2.10) as:

$$\begin{aligned} \hat{\mathbf{x}}^j(k) &= \sum_{\beta=1}^N \zeta^{j\beta}(k) \hat{\mathbf{x}}^{j\beta}(k) \\ \mathbf{P}^j(k) &= \sum_{\beta=1}^N \zeta^{j\beta}(k) \{ \mathbf{P}^{j\beta}(k) + [\hat{\mathbf{x}}^{j\beta}(k) - \hat{\mathbf{x}}^j(k)][\hat{\mathbf{x}}^{j\beta}(k) - \hat{\mathbf{x}}^j(k)]^T \} \end{aligned} \quad (2.16)$$

The discrete state estimate $\hat{q}(k)$ is then computed as:

$$\hat{q}(k) = \underset{i}{\operatorname{argmax}} m^i(k) \quad (2.17)$$

The overall structure of the proposed algorithm is illustrated in Fig. 2.2.

2.4 Numerical Simulation

In this section, the proposed nonlinear state estimation algorithms for the jump Markov system are demonstrated with an illustrative numerical example. We consider a jump Markov system with two modes ($q = 1, 2$). For each mode q , the system behavior [27, 28] is described by:

$$\dot{x}(t) = \xi_q \sin(\eta_q x(t)) + w_q(t) \quad (2.18)$$

where the system parameters are set as $\xi_1 = 1$ and $\eta_1 = 1$ for mode 1, and $\xi_2 = 2$ and $\eta_2 = 1$ for mode 2; and $w_q(t)$ is a zero-mean white Gaussian process noise with

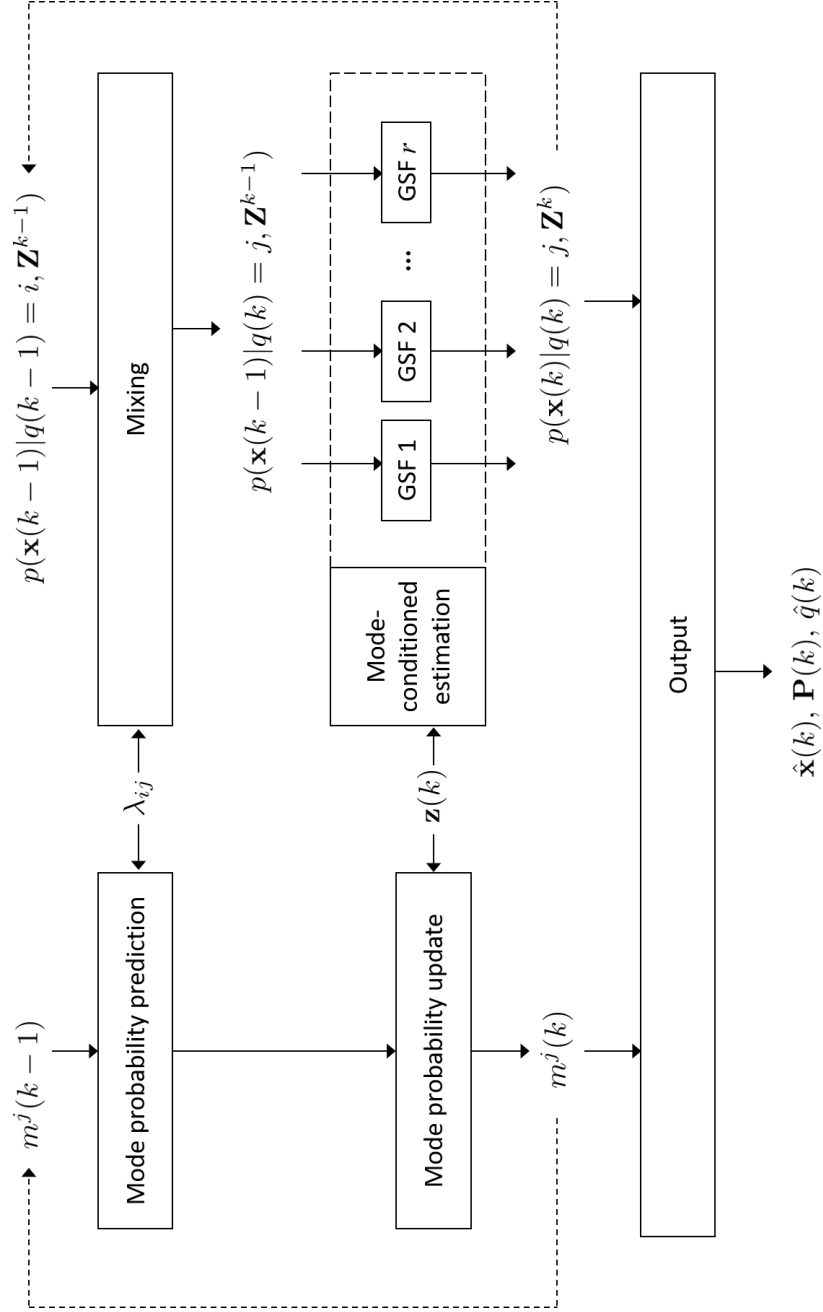


Figure 2.2. Structure of the proposed algorithm

variance $Q_q = 1$ for both modes. For simulation, we obtain the discrete-time dynamic model for each mode q based on discretization of Eq. (2.18) using the Euler-Maruyama method [29] with a sampling time $\Delta T = 0.25s$. A single simulation is carried out for 60 time steps ($0 \leq k \leq 59$). The system switches from mode 1 to mode 2 at time step $k = 29$. The mode transition is modeled as a Markov chain with a constant mode transition matrix as

$$\Lambda = \begin{bmatrix} \lambda_{11} & \lambda_{12} \\ \lambda_{21} & \lambda_{22} \end{bmatrix} = \begin{bmatrix} 0.75 & 0.25 \\ 0.25 & 0.75 \end{bmatrix} \quad (2.19)$$

The measurement equation is given for all the modes as

$$z(k) = \frac{x(k)^2}{20} + v(k) \quad (2.20)$$

where $v(k)$ is a white Gaussian measurement noise with zero mean and variance $R = 1$ which is independent of the process noise; and the measurement is taken at every time step for state estimation. The initial state $x(0) = x_0$ is independent of the noises. We assume that the initial probability distributions of the state are set as $p(x(0)|q(0) = 1) = 0.8\mathcal{N}(-2, 1) + 0.2\mathcal{N}(0.5, 1)$ and $p(x(0)|q(0) = 2) = 0.4\mathcal{N}(-1.5, 1) + 0.6\mathcal{N}(2, 1)$. The initial mode probabilities are assumed to be given as $p(q(0) = 1) = 0.7$ and $p(q(0) = 2) = 0.3$.

Given the simulation example, the proposed algorithms are applied in order to estimate the state of the JMS and evaluated by comparing the state estimate with the true state at every time step. The performances of the proposed algorithms (Proposed I and Proposed II) are compared with those of the existing nonlinear state estimation algorithms for the JMS: 1) the IMM-UKF algorithm, 2) the Gaussian sum filtering algorithm for the JMS (denoted as IMM-GSF) [26], and 3) the IMM-PF algorithm [20]. The differences between the algorithms considered are represented in the mixing step and the mode-conditioned estimation step.

The IMM-UKF algorithm approximates the mixed posterior pdf represented by a Gaussian mixture as a single Gaussian density for the initial condition of the mode-

conditioned filter and then propagates and updates the moments of the Gaussian density using the mode-conditioned filter (i.e., UKF). For simulation, the unscented transform parameters [17] that are applied to generate the sigma points are set as $\bar{\alpha} = 1$, $\bar{\kappa} = 0.6$, and $\bar{\beta} = 2$. Unless otherwise stated, the same parameter values are used for UKF in this simulation. In this example, the IMM-GSF algorithm approximates the mixed posterior pdf with a new Gaussian mixture with three Gaussian components which are deterministically chosen by the minor Gaussian-set design technique [26]. Note that the number of the Gaussian mixture components is the same for all the modes, since it depends on the dimension of the state. For simulation, the parameters that are used to compute the covariances and weights of the Gaussian mixture components in the MGSD technique are set as $\bar{\delta} = 0.9$ and $\bar{\xi}^{j0} = 0.5$, respectively. The Proposed I algorithm and the Proposed II algorithm truncate the number of the original Gaussian mixture components of the mixed posterior pdf by applying the Gaussian reduction techniques with pruning and merging strategies. In this example, the Proposed I algorithm keeps only the Gaussian mixture components with two highest weights ($N = 2$) for each mode. The Proposed II algorithm retains only the Gaussian mixture components that have a weight greater than the value of a preset threshold ϵ , which yields a variable number of the Gaussian components in the resulting Gaussian mixture at each time step. We set the pruning threshold at $\epsilon = 0.1$ for mode 1 and $\epsilon = 0.3$ for mode 2. Note that the number of the resulting Gaussian mixture components in each mode can vary. For performance comparison with the IMM-GSF algorithm, in this simulation, we set the upper and lower bounds on the number of the Gaussian mixture components as $N_u = 3$ and $N_l = 2$ for all the modes. In the both proposed algorithms, for the dropped Gaussian mixture components, the merging strategy that combines the Gaussian components with very similar means is used when they satisfy the conditions for merging in Table 2.1. The parameters of the merging strategy are $\delta = 0.7$, $\kappa = 0.15$, and $\gamma = 0.02$. In the IMM-GSF algorithm and the proposed algorithms, the GSF is used as the mode-conditioned filter and the UKFs are selected as the subfilters of the GSF. We run the IMM-PF algorithm with

two different settings in which 100 and 1000 particles, respectively, are maintained in each mode. In this simulation, 500 Monte Carlo runs are conducted and the root mean square error (RMSE) is calculated to evaluate the state estimation accuracy of the algorithms considered.

Figure 2.3 shows the RMSE of the state estimation for the respective algorithms and the average RMSE over the entire time steps is listed in Table 2.2. As shown, the simulation results clearly indicate that the proposed algorithms outperform the IMM-UKF algorithm and the IMM-GSF algorithm and achieve the performances close to that of the IMM-PF algorithm with 1000 particles. This is reasonable since the proposed algorithms minimize the information loss resulting from the approxi-

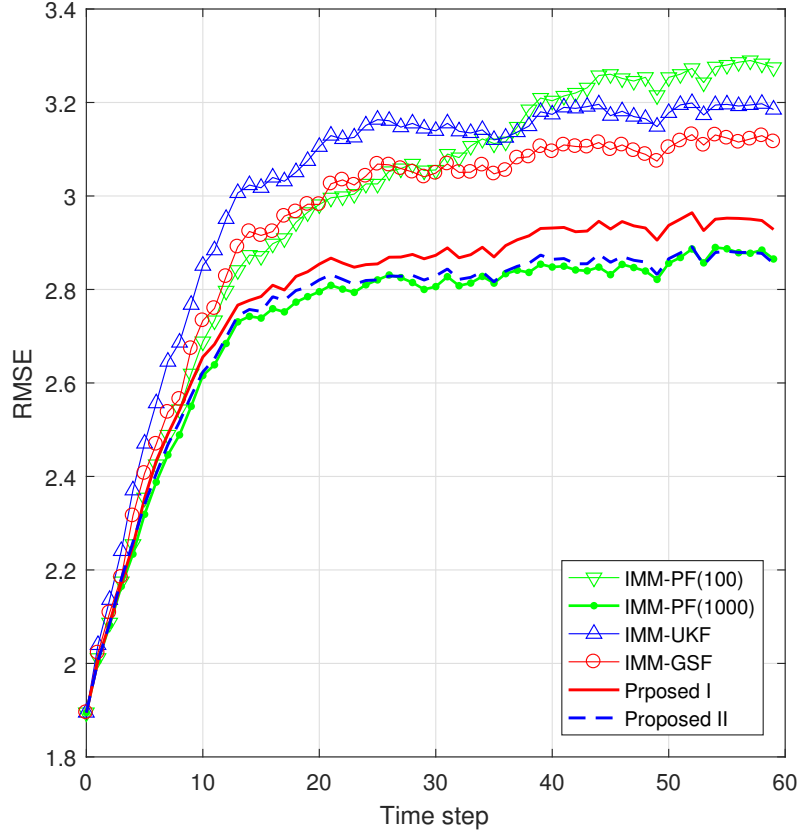
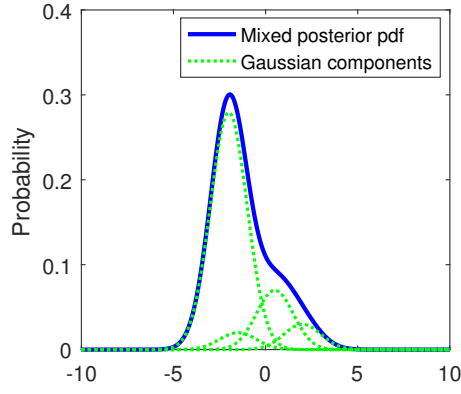


Figure 2.3. Comparison of state estimation performance (500 Monte Carlo runs)

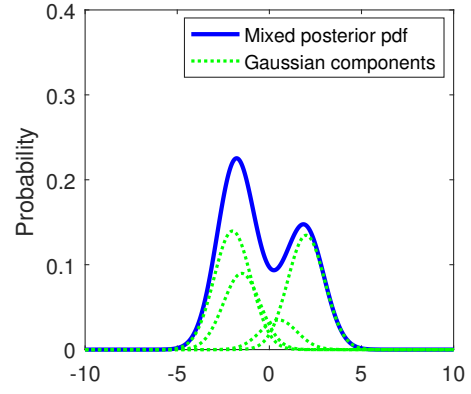
Table 2.2. Performance comparison (500 Monte Carlo runs)

	Average RMSE	Average computation time for a single step (s)
Proposed I	2.78	5.03×10^{-4}
Proposed II	2.73	6.35×10^{-4}
IMM-UKF	3.00	2.29×10^{-4}
IMM-GSF	2.92	6.84×10^{-4}
IMM-PF (100)	2.97	1.02×10^{-2}
IMM-PF (1000)	2.72	1.03×10^{-1}

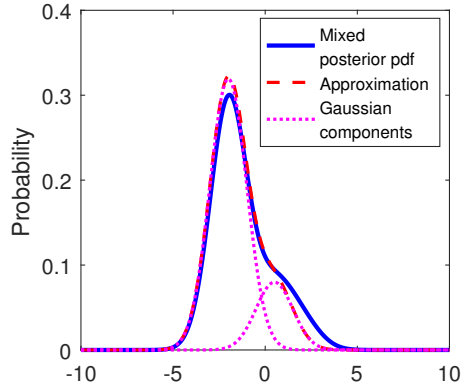
mation of the mixed posterior pdf and thus yield a better approximation of the true posterior pdf of the state which determines the state estimates. In Fig. 2.4, at time step $k = 1$, the resulting approximations of the original mixed posterior pdf by the proposed algorithms (with the Gaussian components reduction techniques) and the IMM-GSF algorithm (with the MGSD technique), respectively, are compared for each mode. The original mixed posterior pdf in the form of a Gaussian mixture and its Gaussian components for each mode are represented in Fig. 2.4(a) and Fig. 2.4(b), and the approximation results are compared with the original mixed posterior pdf in the rest subfigures. Also, the average number of the Gaussian mixture components that are used for the approximation at each time step is computed for each mode over 500 Monte Carlo runs and compared in Table 2.3. From Fig. 2.4 and Table 2.3, it is shown that the proposed Gaussian components reduction techniques with pruning and merging are accurate and notably flexible in approximating the mixed posterior pdf, while the MGSD technique is not. It is also observed that the better state estimation accuracy can be achieved with the smaller number of the Gaussian mixture components by using the proposed algorithms. Figure 2.5 shows the evolution of the posterior pdf of the state by the proposed algorithms, the IMM-GSF algorithm, and the IMM-PF algorithm, respectively, for a single run (where the posterior pdf of the



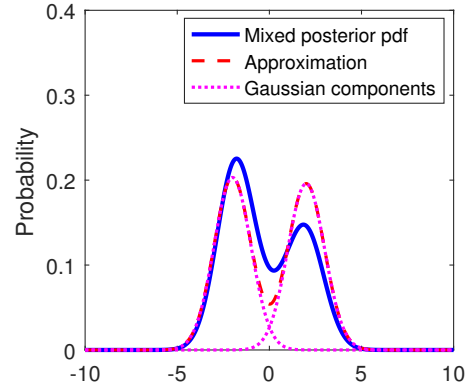
(a) Mode 1: Original mixed posterior pdf



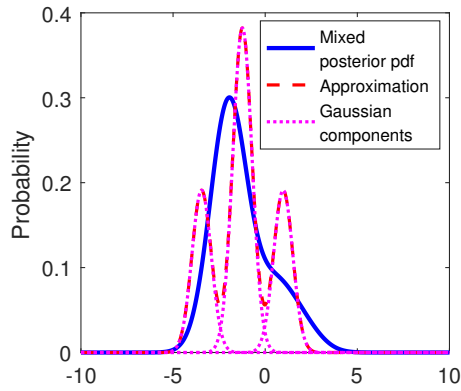
(b) Mode 2: Original mixed posterior pdf



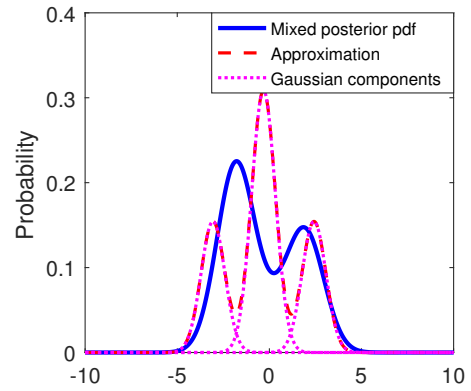
(c) Mode 1: Approximation by Proposed I & II



(d) Mode 2: Approximation by Proposed I & II



(e) Mode 1: Approximation by IMM-GSF



(f) Mode 2: Approximation by IMM-GSF

Figure 2.4. Approximation results of the mixed posterior pdf (at time step $k = 1$)

Table 2.3. Average number of the Gaussian mixture components (500 Monte Carlo runs)

	Mode 1	Mode 2
Proposed I	2.00	2.00
Proposed II	2.94	2.00
IMM-GSF	3.00	3.00

state computed by the IMM-PF algorithm is used as a proxy for the true posterior pdf of the state). We can see that the proposed algorithms more accurately track the true posterior pdf of the state with bimodal nature than the IMM-GSF algorithm. In Table 2.2, the average computation time of each time step for the respective algorithms is summarized. Through the simulation results, it is demonstrated that the proposed algorithms achieve the increased state estimation accuracies with the reduced computational efforts in solving the nonlinear state estimation problem for the JMS.

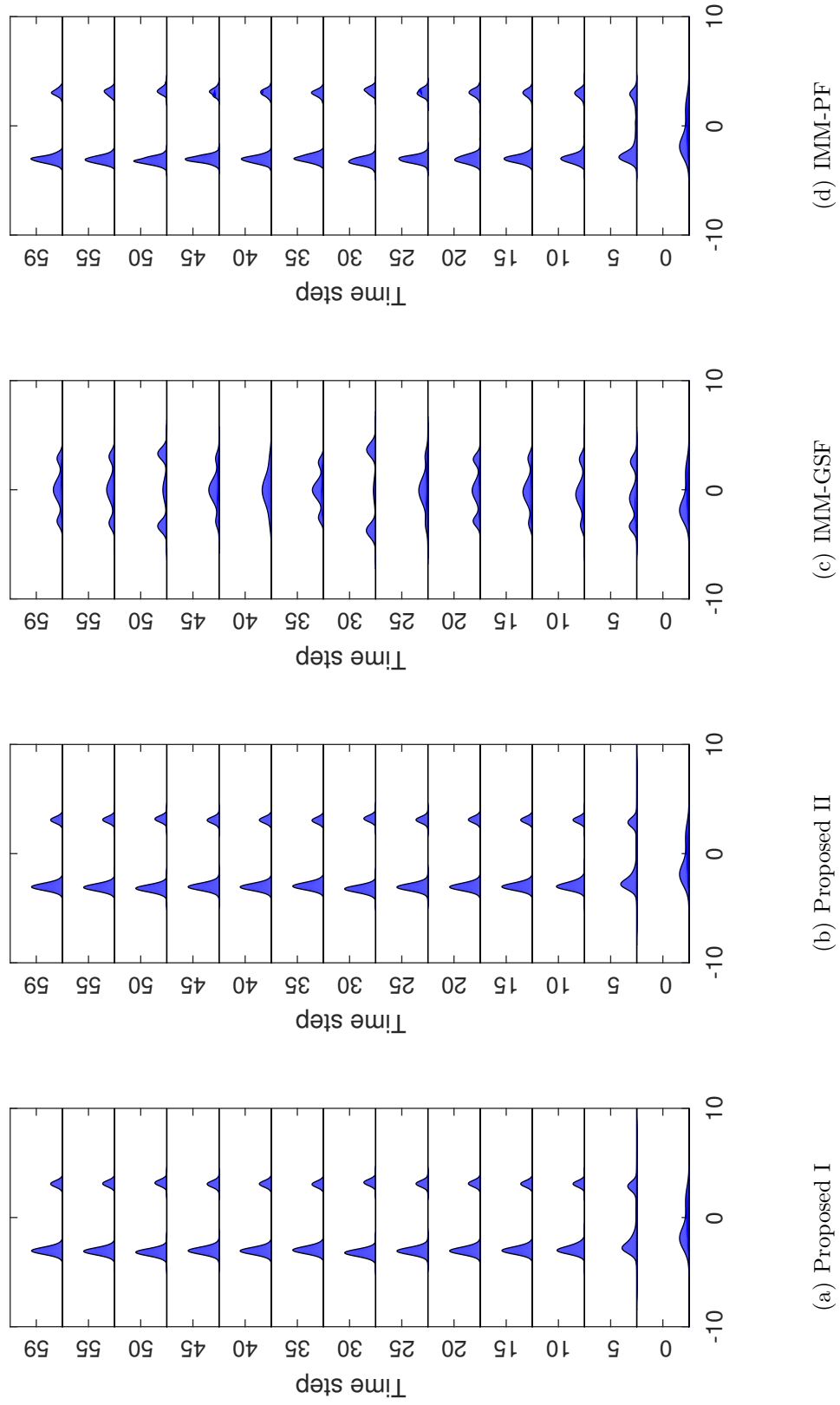


Figure 2.5. Evolution of the posterior pdf of the state for a single run

3. NONLINEAR HYBRID STATE ESTIMATION FOR A GENERAL CLASS OF STOCHASTIC HYBRID SYSTEMS USING GAUSSIAN SUM FILTERS

In this chapter, the proposed nonlinear hybrid state estimation algorithm based on Gaussian sum approximation is extended to a more general class of the SHS where the discrete state transitions are governed by stochastic guard conditions dependent on the continuous state.

3.1 Background and Motivations

The nonlinear hybrid estimation algorithms mentioned in 2.1 only consider the discrete mode transition with constant probabilities irrespective of the continuous state evolution. However, in many cases, the mode transition is conditioned on the continuous state. For example, an aircraft during descent phase along its flight plan switches its flight mode (i.e., discrete state transition) near flight mode change points that are determined by its continuous state such as the altitude and horizontal position. Another example is a modern vehicle with advanced driver-assistance systems whose behavior is a complex combination of multiple motions in different operating modes driven by various subsystems such as an adaptive cruise control system and a collision mitigating system. The transition between the subsystems is governed by the vehicle's continuous state such as speed and distance to adjacent objects. This implies that the proposed nonlinear hybrid state estimation algorithm needs to be improved to deal with the continuous state-dependent mode transition.

In this research, the nonlinear hybrid state estimation algorithm proposed in Chapter 2 is improved to deal with both nonlinearity and state-dependent mode

transition. First, to overcome the limitation in handling nonlinearity, we propose to use the GSF [22] as the mode-conditioned filter that has been proven effective for approximating a non-Gaussian distribution with low computational cost. Second, we introduce stochastic guard conditions to model the continuous state-dependent mode transition and derive a closed-form mode transition probability by analytically integrating the continuous state probability distribution over the guard conditions. For the analytic integration, the analytical properties of the Gaussian sum are exploited. The proposed GSF and state-dependent mode transition probability are then integrated into the IMM framework as an attempt to deal with the increasing number of hypotheses on the discrete state history.

3.2 Problem Formulation

A class of nonlinear stochastic hybrid systems is considered in this paper as follows. The discrete state denoted by $s(k)$ represents the mode of the system at time k , where the number of modes is q , i.e., $s(k) \in \mathcal{S} = \{1, 2, \dots, q\}$. The continuous state and measurement vectors are denoted by $\mathbf{x}(k) \in \mathbb{R}^n$ and $\mathbf{z}(k) \in \mathbb{R}^m$, respectively. For each mode $s(k) \in \mathcal{S}$, the dynamic behavior of the continuous state is governed by

$$\mathbf{x}(k+1) = \mathbf{f}_{s(k)}(\mathbf{x}(k)) + \boldsymbol{\omega}_{s(k)}(k) \quad (3.1)$$

where $\mathbf{f}_{s(k)}$ is a smooth bounded nonlinear function and $\boldsymbol{\omega}_{s(k)}(k)$ is a zero-mean white Gaussian process noise with covariance $\mathbf{Q}_{s(k)}$. The measurement is generated by a mode-specific nonlinear mapping $\mathbf{g}_{s(k)} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ as

$$\mathbf{z}(k) = \mathbf{g}_{s(k)}(\mathbf{x}(k)) + \boldsymbol{\nu}_{s(k)}(k) \quad (3.2)$$

where $\boldsymbol{\nu}_{s(k)}(k)$ is a zero-mean Gaussian measurement noise with covariance $\mathbf{R}_{s(k)}$. The discrete state also evolves over time and its transition is governed by a set of stochastic linear guard conditions $G(i, j)$, $i, j = 1, 2, \dots, q$ such that, if $s(k) = i$ and

$\mathbf{x}(k)$ satisfies $G(i, j)$ at time k , then the discrete state evolves to $s(k+1) = j$ at time $k+1$. The guard conditions are modeled as linear inequalities as

$$G(i, j) = \{\mathbf{x} | \mathbf{L}_{\mathbf{x},ij}\mathbf{x} + \mathbf{L}_{\boldsymbol{\eta},ij}\boldsymbol{\eta} \leq \mathbf{0}\} \quad (3.3)$$

where $\mathbf{L}_{\mathbf{x},ij} \in \mathbb{R}^{l \times n}$ and $\mathbf{L}_{\boldsymbol{\eta},ij} \in \mathbb{R}^{l \times r}$ are constant matrices that determine the shape of the guard condition in \mathbb{R}^n , and $\boldsymbol{\eta} \in \mathbb{R}^r$ is a Gaussian random variable

$$\boldsymbol{\eta} \sim \mathcal{N}(\boldsymbol{\eta}; \bar{\boldsymbol{\eta}}, \boldsymbol{\Sigma}_{\boldsymbol{\eta}}) \quad (3.4)$$

that accounts for uncertainties of the discrete state transition. It should be noted that the guard conditions in Eq. (3.3) explicitly present the dependency of the discrete state transitions on the continuous state \mathbf{x} (i.e., state-dependent transition), and this transition model can handle more general hybrid systems than those with constant mode transition probabilities which are a special case of the proposed model.

Given the nonlinear stochastic hybrid system defined above, the hybrid state estimation problem is formulated as computing the posterior probability distribution of both the continuous state $p(\mathbf{x}(k) | \mathbf{Z}^k)$ and discrete state $p(s(k) | \mathbf{Z}^k)$, where \mathbf{Z}^k represents a set of measurement vectors up to time k as $\mathbf{Z}^k = \{\mathbf{z}(1), \mathbf{z}(2), \dots, \mathbf{z}(k)\}$. The state estimates are then obtained by

$$\begin{aligned} \hat{\mathbf{x}}(k) &:= E[\mathbf{x}(k) | \mathbf{Z}^k] \\ \hat{s}(k) &:= \underset{i}{\operatorname{argmax}} p(s(k) = i | \mathbf{Z}^k) \end{aligned} \quad (3.5)$$

3.3 Nonlinear Hybrid State Estimation based on Gaussian Sum Approximation

The main difficulties in solving the hybrid state estimation problem defined in the previous section are twofold. First, since the exact discrete state (i.e., which mode the system is operating in at time k) is not known to the estimator, multiple hypotheses need to be maintained over all possible histories of the discrete state to obtain the optimal solution, which is impractical due to the exponentially increasing

number of hypotheses over time. Second, the computation of the state-dependent mode transition probabilities governed by the guard conditions in Eq. (3.3) is not straightforward as it involves the integration of the probability density function of the continuous state \mathbf{x} which is non-Gaussian in general due to the nonlinearity in its dynamics in Eq. (3.1).

In this research, these difficulties are overcome as follows. First, we apply the hypothesis merging approach proposed in the IMM to keep the number of hypotheses constant and thus to avoid exponentially growing computational complexity. Second, we propose to use the Gaussian mixture to effectively approximate the actual non-Gaussian probability distribution of the continuous state. The analytical properties of the Gaussian mixture are then used to derive a closed-form of the state-dependent mode transition probabilities corresponding to the guard conditions. The derived closed-form solution allows to compute the transition probabilities without computationally expensive numerical integration.

The proposed algorithm computes the posterior probability distributions recursively using Bayes' theorem. The details of the proposed approach recurring in each iteration are discussed as follows. First, at the beginning of the iteration at time k , the inputs to the proposed algorithm are the mode probabilities

$$m^i(k-1) \equiv p(s(k-1) = i | \mathbf{Z}^{k-1}) \quad (3.6)$$

and the mode-matched posterior continuous state pdfs

$$\begin{aligned} & p(\mathbf{x}(k-1) | s(k-1) = i, \mathbf{Z}^{k-1}) \\ &= \sum_{\alpha=1}^N \xi^{i\alpha}(k-1) \mathcal{N}(\mathbf{x}(k-1); \hat{\mathbf{x}}^{i\alpha}(k-1), \mathbf{P}^{i\alpha}(k-1)) \end{aligned} \quad (3.7)$$

for $i = 1, 2, \dots, q$, computed from the last iteration at time $k-1$. As shown in Eq. (3.7), for each mode i , the mode-matched pdf is approximated by a weighted sum of N Gaussians, where $\xi^{i\alpha}(k-1)$, $\hat{\mathbf{x}}^{i\alpha}(k-1)$, and $\mathbf{P}^{i\alpha}(k-1)$ are respectively the weight, mean, and covariance of the α th component of the Gaussian mixture. The weights satisfy $\sum_{\alpha=1}^N \xi^{i\alpha}(k-1) = 1$. Given these inputs and the new measurement

$\mathbf{z}(k)$ generated at time k , the proposed algorithm uses a bank of q GSFs, each of which is matched to the continuous dynamics of each mode, to compute the posterior probability distributions $p(\mathbf{x}(k)|s(k) = j, \mathbf{Z}^k)$ and $m^j(k)$ for each mode $j = 1, 2, \dots, q$, as follows:

Step 1: Mixing

Note that, for each mode j , the mode-matched posterior pdf $p(\mathbf{x}(k)|s(k) = j, \mathbf{Z}^k)$ can be represented using Bayes' theorem as

$$p(\mathbf{x}(k)|s(k) = j, \mathbf{Z}^k) = \frac{p(\mathbf{z}(k)|\mathbf{x}(k), s(k) = j, \mathbf{Z}^{k-1})}{p(\mathbf{z}(k)|s(k) = j, \mathbf{Z}^{k-1})} p(\mathbf{x}(k)|s(k) = j, \mathbf{Z}^{k-1}) \quad (3.8)$$

where the last term represents the prior pdf and can be further extended as

$$\begin{aligned} p(\mathbf{x}(k)|s(k) = j, \mathbf{Z}^{k-1}) &= \sum_{i=1}^q p(\mathbf{x}(k)|s(k) = j, s(k-1) = i, \mathbf{Z}^{k-1}) \\ &\times p(s(k-1) = i|s(k) = j, \mathbf{Z}^{k-1}) \end{aligned} \quad (3.9)$$

As shown, the prior pdf is the weighted sum of sub-priors, each of which is conditioned on one of the possible modes at time $k-1$. To compute the prior pdf, the weight $p(s(k-1) = i|s(k) = j, \mathbf{Z}^{k-1})$ (also called as the mixing probability and denoted by $m^{ij}(k)$) needs to be calculated for each component as

$$\begin{aligned} m^{ij}(k) &= p(s(k-1) = i|s(k) = j, \mathbf{Z}^{k-1}) \\ &= \frac{p(s(k) = j|s(k-1) = i, \mathbf{Z}^{k-1})p(s(k-1) = i|\mathbf{Z}^{k-1})}{p(s(k) = j|\mathbf{Z}^{k-1})} \\ &= \frac{\lambda_{ij}(k-1)m^i(k-1)}{\sum_{l=1}^q \lambda_{lj}(k-1)m^l(k-1)} \end{aligned} \quad (3.10)$$

where

$$\lambda_{ij}(k-1) \equiv p(s(k) = j|s(k-1) = i|\mathbf{Z}^{k-1}) \quad (3.11)$$

is the mode transition probability that needs to be computed using the guard condition $G(i, j)$ defined in Eq. (3.3). In this research, we provide a succinct, closed-form expression for the mode transition probability through the following lemma and theorem. To begin with, let $\Phi_l(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ be the l -dimensional Gaussian cumulative

density function defined as $\Phi_l(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \equiv p(\mathbf{a} \leq \mathbf{0})$ for a Gaussian random variable $\mathbf{a} \sim \mathcal{N}(\mathbf{a}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$.

Lemma 1 *Given a constant vector $\mathbf{a} \in \mathbb{R}^l$, a constant matrix $\mathbf{A} \in \mathbb{R}^{l \times n}$, a symmetric positive definite matrix $\mathbf{B} \in \mathbb{R}^{l \times l}$, and a Gaussian random vector $\mathbf{b} \in \mathbb{R}^n \sim \mathcal{N}(\mathbf{b}; \bar{\mathbf{b}}, \mathbf{C})$, the following holds true:*

$$\int_{\mathbb{R}^n} \Phi_l(\mathbf{A}\mathbf{b} + \mathbf{a}, \mathbf{B}) \mathcal{N}(\mathbf{b}, \bar{\mathbf{b}}; \mathbf{C}) d\mathbf{b} = \Phi_l(\mathbf{A}\bar{\mathbf{b}} + \mathbf{a}, \mathbf{D}) \quad (3.12)$$

where

$$\mathbf{D} = \mathbf{B} + \mathbf{A}\mathbf{C}\mathbf{A}^T \quad (3.13)$$

Proof The proof can be found in [30]. ■

Theorem 3.3.1 *Suppose that the mode-matched continuous state pdf $p(\mathbf{x}(k-1)|s(k-1) = i, \mathbf{Z}^{k-1})$ for the i -th mode at time $k-1$ is given as Eq. (3.7) and the mode transition from $s(k-1) = i$ to $s(k) = j$ is governed by the guard condition in Eq. (3.3). Then, the mode transition probability $\lambda_{ij}(k-1)$ can be computed in the following closed-form expression:*

$$\lambda_{ij}(k-1) = \sum_{\alpha=1}^N \xi^{i\alpha}(k-1) \Phi_l(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \quad (3.14)$$

where

$$\begin{aligned} \boldsymbol{\mu} &= \mathbf{L}_{\boldsymbol{\eta}, ij} \bar{\boldsymbol{\eta}} + \mathbf{L}_{\mathbf{x}, ij} \hat{\mathbf{x}}^{i\alpha}(k-1) \\ \boldsymbol{\Sigma} &= \mathbf{L}_{\boldsymbol{\eta}, ij} \boldsymbol{\Sigma}_{\boldsymbol{\eta}} \mathbf{L}_{\boldsymbol{\eta}, ij}^T + \mathbf{L}_{\mathbf{x}, ij} \mathbf{P}^{i\alpha}(k-1) \mathbf{L}_{\mathbf{x}, ij}^T \end{aligned} \quad (3.15)$$

Proof By the total probability theorem, $\lambda_{ij}(k-1)$ can be extended as

$$\begin{aligned} \lambda_{ij}(k-1) &= \int_{\mathbb{R}^n} p(s(k) = j | s(k-1) = i, \mathbf{x}(k-1) = \mathbf{x}, \mathbf{Z}^{k-1}) \\ &\quad \times p(\mathbf{x}(k-1) = \mathbf{x} | s(k-1) = i, \mathbf{Z}^{k-1}) d\mathbf{x} \end{aligned} \quad (3.16)$$

Using the Gaussian mixture in Eq. (3.7), Eq. (3.16) is further extended as

$$\begin{aligned} \lambda_{ij}(k-1) &= \sum_{\alpha=1}^N \xi^{i\alpha}(k-1) \left\{ \int_{\mathbb{R}^n} p(s(k) = j | s(k-1) = i, \right. \\ &\quad \left. \mathbf{x}(k-1) = \mathbf{x}, \mathbf{Z}^{k-1}) \times \mathcal{N}(\mathbf{x}; \hat{\mathbf{x}}^{i\alpha}(k-1), \mathbf{P}^{i\alpha}(k-1)) d\mathbf{x} \right\} \end{aligned} \quad (3.17)$$

From the guard condition $G(i, j)$ in Eq. (3.3), the first term in the integral in Eq. (3.17) is equivalent to

$$\begin{aligned} p(s(k) = j | s(k-1) = i, \mathbf{x}(k-1) = \mathbf{x}, \mathbf{Z}^{k-1}) \\ = \Phi_l(\mathbf{L}_{\eta, ij} \bar{\boldsymbol{\eta}} + \mathbf{L}_{\mathbf{x}, ij} \mathbf{x}, \mathbf{L}_{\eta, ij} \boldsymbol{\Sigma}_{\eta} \mathbf{L}_{\eta, ij}^T) \end{aligned} \quad (3.18)$$

Then, using Lemma 1, we have

$$\begin{aligned} \int_{\mathbb{R}^n} \Phi_l(\mathbf{L}_{\eta, ij} \bar{\boldsymbol{\eta}} + \mathbf{L}_{\mathbf{x}, ij} \mathbf{x}, \mathbf{L}_{\eta, ij} \boldsymbol{\Sigma}_{\eta} \mathbf{L}_{\eta, ij}^T) \mathcal{N}(\mathbf{x}; \hat{\mathbf{x}}^{i\alpha}(k-1), \mathbf{P}^{i\alpha}(k-1)) d\mathbf{x} \\ = \Phi_l(\mathbf{L}_{\mathbf{x}, ij} \hat{\mathbf{x}}^{i\alpha}(k-1) + \mathbf{L}_{\eta, ij} \bar{\boldsymbol{\eta}}, \mathbf{L}_{\eta, ij} \boldsymbol{\Sigma}_{\eta} \mathbf{L}_{\eta, ij}^T + \mathbf{L}_{\mathbf{x}, ij} \mathbf{P}^{i\alpha}(k-1) \mathbf{L}_{\mathbf{x}, ij}^T) \end{aligned} \quad (3.19)$$

Finally, Eqs. (3.17)-(3.19) complete the proof. ■

The initial conditions for the j -th GSF are computed using the mixing probability as

$$\begin{aligned} p(\mathbf{x}(k-1) | s(k) = j, \mathbf{Z}^{k-1}) &= \sum_{i=1}^q m^{i|j}(k) p(\mathbf{x}(k-1) | s(k-1) = i, \mathbf{Z}^{k-1}) \\ &= \sum_{\alpha=1}^N \sum_{i=1}^q m^{i|j}(k) \xi^{i\alpha}(k-1) \\ &\quad \times \mathcal{N}(\mathbf{x}(k-1); \hat{\mathbf{x}}^{i\alpha}(k-1), \mathbf{P}^{i\alpha}(k-1)) \end{aligned} \quad (3.20)$$

To keep the number of the elements of the Gaussian mixture constant, the $q \times N$ Gaussians in Eq. (3.20) is reduced to N . In this research, we first prune components of small weights (where the weight of each component is $m^{i|j}(k) \xi^{i\alpha}(k-1)$), then merge them to N components. The detailed procedure is described as follows.

To begin with, among $q \times N$ components, N components corresponding to the highest N weights are selected. If the sum of the weights of the selected components is greater than a threshold δ (e.g., $\delta = 0.8$), the weights are normalized and the initially selected N components become the final output. If the sum is less than the threshold, which means the initially selected N components are not enough to account for the information in the original pdf in Eq. (3.20), the remaining $(q-1) \times N$ components are further investigated to be merged to the initially selected N components. For each of the remaining components with the weight greater than a threshold κ (e.g.,

$\kappa = 0.1$), the distances to the initially selected N components are computed. If the distance to the nearest component is less than a threshold γ (e.g., $\gamma = 0.02$), the remaining component is merged to the nearest component by means of the moment matching approximation (the procedure discussed above is summarized in Table 2.1 as Proposed I).

After the merging and pruning, the initial conditions are approximated by a new set of N Gaussians as

$$\begin{aligned} p(\mathbf{x}(k-1)|s(k)=j, \mathbf{Z}^{k-1}) \\ = \sum_{\beta=1}^N \xi^{j\beta}(k-1) \mathcal{N}(\mathbf{x}(k-1); \hat{\mathbf{x}}^{j\beta}(k-1), \mathbf{P}^{j\beta}(k-1)) \end{aligned} \quad (3.21)$$

where $\xi^{j\beta}(k-1)$, $\beta = 1, 2, \dots, N$ is a new set of weights, and $\hat{\mathbf{x}}^{j\beta}(k-1)$ and $\mathbf{P}^{j\beta}(k-1)$ are the corresponding mean and covariance, respectively. Given the initial conditions in Eq. (3.21) for mode j , the j -th GSF computes the mode-conditioned posterior pdf $p(\mathbf{x}(k)|s(k)=j, \mathbf{Z}^k)$ as explained in the following step.

Step 2: Mode-conditioned estimation

Each component of the Gaussian sum in Eq. (3.21) can be considered as the local approximation of the actual non-Gaussian distribution. The mode-matched GSF consists of the N Gaussian sub-filters, where each sub-filter predicts and updates each Gaussian local approximation. In this research, the unscented Kalman filter is used as the sub-filter. Using the j -th GSF, the prior pdf for mode j , $p(\mathbf{x}(k)|s(k)=j, \mathbf{Z}^{k-1})$ is obtained as

$$\begin{aligned} p(\mathbf{x}(k)|s(k)=j, \mathbf{Z}^{k-1}) \\ = \sum_{\beta=1}^N \xi^{j\beta}(k|k-1) \mathcal{N}(\mathbf{x}(k); \hat{\mathbf{x}}^{j\beta}(k|k-1), \mathbf{P}^{j\beta}(k|k-1)) \end{aligned} \quad (3.22)$$

where $\hat{\mathbf{x}}^{j\beta}(k|k-1)$ and $\mathbf{P}^{j\beta}(k|k-1)$ are the predicted mean and covariance computed by the β -th UKF as

$$\begin{aligned}\hat{\mathbf{x}}^{j\beta}(k|k-1) &= \sum_{\rho=0}^{2n} \epsilon_{\rho}^m \boldsymbol{\chi}_{\rho}^{j\beta}(k|k-1) \\ \mathbf{P}^{j\beta}(k|k-1) &= \sum_{\rho=0}^{2n} \epsilon_{\rho}^c [\boldsymbol{\chi}_{\rho}^{j\beta}(k|k-1) - \hat{\mathbf{x}}^{j\beta}(k|k-1)] \\ &\quad \times [\boldsymbol{\chi}_{\rho}^{j\beta}(k|k-1) - \hat{\mathbf{x}}^{j\beta}(k|k-1)]^T + \mathbf{Q}_j\end{aligned}\quad (3.23)$$

where ϵ_{ρ}^m and ϵ_{ρ}^c , $\rho = 0, 1, \dots, 2n+1$ are the weights and the sigma point $\boldsymbol{\chi}_{\rho}^{j\beta}(k|k-1)$ is the transformation of the ρ -th column of an $n \times (2n+1)$ matrix $\boldsymbol{\chi}^{j\beta}(k-1)$ as

$$\begin{aligned}\boldsymbol{\chi}^{j\beta}(k-1) &= \begin{bmatrix} \hat{\mathbf{x}}^{j\beta}(k-1) & \hat{\mathbf{x}}^{j\beta}(k-1) \pm \sqrt{(n+\bar{\lambda})\mathbf{P}^{j\beta}(k-1)} \end{bmatrix} \\ \boldsymbol{\chi}_{\rho}^{j\beta}(k|k-1) &= \mathbf{f}_j(\boldsymbol{\chi}_{\rho}^{j\beta}(k-1)) \\ \epsilon_0^m &= \frac{\bar{\lambda}}{n+\bar{\lambda}} \\ \epsilon_0^c &= \epsilon_0^m + (1 - \bar{\alpha}^2 + \bar{\beta}) \\ \epsilon_{\rho}^m &= \epsilon_{\rho}^c = \frac{1}{2(n+\bar{\lambda})}\end{aligned}$$

where $\bar{\lambda}$, $\bar{\alpha}$, and $\bar{\beta}$ are the unscented transform parameters. The predicted weight $\xi^{j\beta}(k|k-1)$ is set as $\xi^{j\beta}(k|k-1) = \xi^{j\beta}(k-1)$.

Given a measurement $\mathbf{z}(k)$, the posterior pdf for mode j , $p(\mathbf{x}(k)|s(k)=j, \mathbf{Z}^k)$, is then computed as

$$p(\mathbf{x}(k)|s(k)=j, \mathbf{Z}^k) = \sum_{\beta=1}^N \xi^{j\beta}(k) \mathcal{N}(\mathbf{x}(k); \hat{\mathbf{x}}^{j\beta}(k), \mathbf{P}^{j\beta}(k)) \quad (3.24)$$

where $\hat{\mathbf{x}}^{j\beta}(k)$ and $\mathbf{P}^{j\beta}(k)$ are the updated mean and covariance obtained by the β -th UKF as

$$\begin{aligned}\hat{\mathbf{x}}^{j\beta}(k) &= \hat{\mathbf{x}}^{j\beta}(k|k-1) + \mathbf{K}_{j\beta}(k)(\mathbf{z}(k) - \hat{\mathbf{z}}^{j\beta}(k|k-1)) \\ \mathbf{P}^{j\beta}(k) &= \mathbf{P}^{j\beta}(k|k-1) - \mathbf{K}_{j\beta}(k)\mathbf{P}_{j\beta}^{zz}(k|k-1)\mathbf{K}_{j\beta}(k)^T\end{aligned}\quad (3.25)$$

where

$$\begin{aligned}
\bar{\mathbf{Z}}_\rho^{j\beta}(k|k-1) &= \mathbf{g}_j(\bar{\boldsymbol{\chi}}_\rho^{j\beta}(k|k-1)) \\
\hat{\mathbf{z}}^{j\beta}(k|k-1) &= \sum_{\rho=0}^{2n} \epsilon_\rho^m \bar{\mathbf{Z}}_\rho^{j\beta}(k|k-1) \\
\mathbf{P}_{j\beta}^{zz}(k|k-1) &= \sum_{\rho=0}^{2n} \epsilon_\rho^c [\bar{\mathbf{Z}}_\rho^{j\beta}(k|k-1) - \hat{\mathbf{z}}^{j\beta}(k|k-1)] \\
&\quad \times [\bar{\mathbf{Z}}_\rho^{j\beta}(k|k-1) - \hat{\mathbf{z}}^{j\beta}(k|k-1)]^T + \mathbf{R}_j \\
\mathbf{P}_{j\beta}^{xz}(k|k-1) &= \sum_{\rho=0}^{2n} \epsilon_\rho^c [\bar{\boldsymbol{\chi}}_\rho^{j\beta}(k|k-1) - \hat{\mathbf{x}}^{j\beta}(k|k-1)] \\
&\quad \times [\bar{\mathbf{Z}}_\rho^{j\beta}(k|k-1) - \hat{\mathbf{z}}^{j\beta}(k|k-1)]^T \\
\mathbf{K}_{j\beta}(k) &= \mathbf{P}_{j\beta}^{xz}(k|k-1) \mathbf{P}_{j\beta}^{zz}(k|k-1)^{-1}
\end{aligned} \tag{3.26}$$

The weights are updated as

$$\begin{aligned}
\xi^{j\beta}(k) &= \frac{1}{c_{j\beta}} \xi^{j\beta}(k|k-1) \\
&\quad \times \mathcal{N}(\mathbf{z}(k) - \hat{\mathbf{z}}^{j\beta}(k|k-1); \mathbf{0}, \mathbf{P}_{j\beta}^{zz}(k|k-1))
\end{aligned} \tag{3.27}$$

where $\mathcal{N}(\mathbf{z}(k) - \hat{\mathbf{z}}^{j\beta}(k|k-1); \mathbf{0}, \mathbf{P}_{j\beta}^{zz}(k|k-1))$ is the likelihood of $\mathbf{z}(k)$ in the β -th UKF, and $c_{j\beta}$ is a normalizing constant that imposes

$$\sum_{\beta=1}^N \xi^{j\beta}(k) = 1 \tag{3.28}$$

Step 3: Mode probability estimation

First, the prior mode probability for mode j , $p(s(k) = j | \mathbf{Z}^{k-1})$, is computed as

$$\begin{aligned}
p(s(k) = j | \mathbf{Z}^{k-1}) &= \sum_{i=1}^q p(s(k) = j | s(k-1) = i, \mathbf{Z}^{k-1}) \\
&\quad \times p(s(k-1) = i | \mathbf{Z}^{k-1}) \\
&= \sum_{i=1}^q \lambda_{ij}(k-1) m^i(k-1)
\end{aligned} \tag{3.29}$$

Then, based on Bayes' theorem, the posterior mode probability is computed for mode j as

$$\begin{aligned} m^j(k) &= p(s(k) = j | \mathbf{Z}^k) \\ &= \frac{1}{c_j} p(\mathbf{z}(k) | s(k) = j, \mathbf{Z}^{k-1}) p(s(k) = j | \mathbf{Z}^{k-1}) \end{aligned} \quad (3.30)$$

where c_j is a normalizing constant and $p(\mathbf{z}(k) | s(k) = j, \mathbf{Z}^{k-1})$ is the mode-conditioned likelihood given by

$$\begin{aligned} &p(\mathbf{z}(k) | s(k) = j, \mathbf{Z}^{k-1}) \\ &= \sum_{\beta=1}^N \xi^{j\beta}(k | k-1) \mathcal{N}(\mathbf{z}(k) - \hat{\mathbf{z}}^{j\beta}(k | k-1); \mathbf{0}, \mathbf{P}_{j\beta}^{zz}(k | k-1)) \end{aligned} \quad (3.31)$$

Step 4: Output

Using the total probability theorem, it is shown that

$$p(\mathbf{x}(k) | \mathbf{Z}^k) = \sum_{i=1}^q p(\mathbf{x}(k) | s(k) = i, \mathbf{Z}^k) p(s(k) = i | \mathbf{Z}^k) \quad (3.32)$$

From Eqs. (3.5) and (3.32), the continuous state estimate and its covariance are computed as

$$\begin{aligned} \hat{\mathbf{x}}(k) &= \sum_{i=1}^q m^i(k) \hat{\mathbf{x}}^i(k) \\ \mathbf{P}(k) &= \sum_{i=1}^q m^i(k) \{ \mathbf{P}^i(k) + [\hat{\mathbf{x}}^i(k) - \hat{\mathbf{x}}(k)][\hat{\mathbf{x}}^i(k) - \hat{\mathbf{x}}(k)]^T \} \end{aligned} \quad (3.33)$$

where the mode-conditioned state estimates $\hat{\mathbf{x}}^i(k)$ and covariance $\mathbf{P}^i(k)$ are computed from the mode-conditioned posterior distributions in Eq. (3.24) as

$$\begin{aligned} \hat{\mathbf{x}}^i(k) &= \sum_{\beta=1}^N \xi^{i\beta}(k) \hat{\mathbf{x}}^{i\beta}(k) \\ \mathbf{P}^i(k) &= \sum_{\beta=1}^N \xi^{i\beta}(k) \{ \mathbf{P}^{i\beta}(k) + [\hat{\mathbf{x}}^{i\beta}(k) - \hat{\mathbf{x}}^i(k)][\hat{\mathbf{x}}^{i\beta}(k) - \hat{\mathbf{x}}^i(k)]^T \} \end{aligned} \quad (3.34)$$

The discrete state estimate $\hat{s}(k)$ is computed as

$$\hat{s}(k) = \underset{i}{\operatorname{argmax}} m^i(k) \quad (3.35)$$

The overall structure of the proposed algorithm is illustrated in Fig. 3.1.

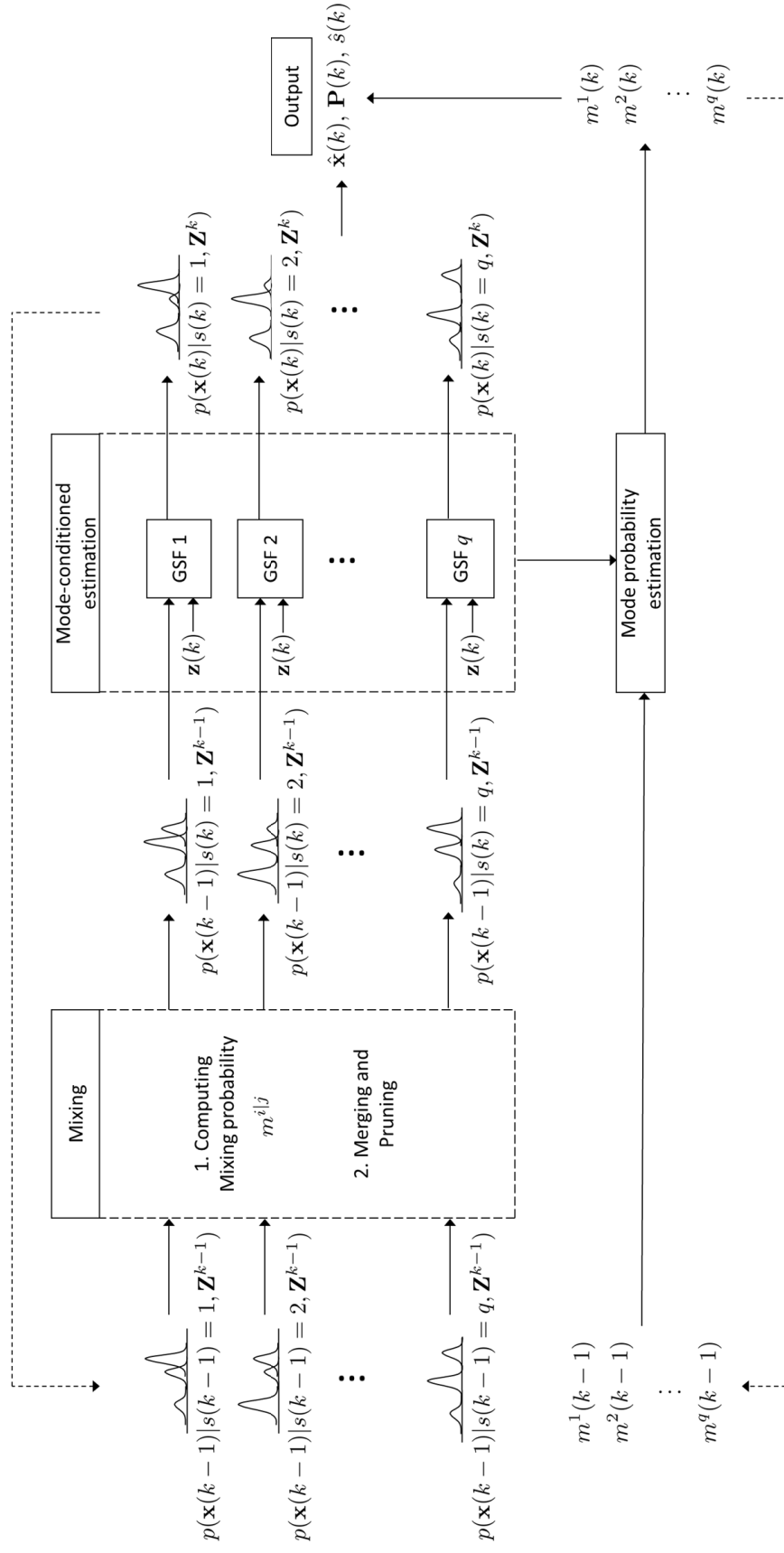


Figure 3.1. The overall structure of the proposed algorithm

3.4 Numerical Simulation

In this section, the effectiveness of the proposed algorithm is demonstrated using an illustrative numerical example with a multimodal system. We consider a nonlinear stochastic hybrid system with two modes (i.e., $s \in \{1, 2\}$) for which the continuous dynamics is given as

$$\dot{x}(t) = \xi_s x(t)(1 - \eta_s x(t)^2) + \omega_s(t), \quad s \in \{1, 2\} \quad (3.36)$$

where ξ_s and η_s are system parameters that characterize the behavior of the system for each mode, given as $\xi_1 = 1.25$, $\eta_1 = 0.02$ for $s = 1$ and $\xi_2 = 1.5$, $\eta_2 = 0.01$ for $s = 2$; and $\omega_s(t)$ is a zero-mean white Gaussian process noise with strength $Q_s = 0.5$ for both $s = 1, 2$. In this example, Eq. (3.36) is discretized using the Euler-Maruyama method [29] with a sampling time $\Delta T = 0.1$ sec to obtain the corresponding discrete-time dynamic model. The measurement equation is given for all the modes as

$$z(k) = \frac{x(k)^2}{10} + \nu(k) \quad (3.37)$$

where $\nu(k)$ is a white Gaussian measurement noise with zero mean and covariance $R = 0.2$, which is independent of the process noise; and the measurement is taken at every time step for state estimation. The system is initialized at mode 1 ($s = 1$) and switches to mode 2 ($s = 2$) when the following guard condition is satisfied:

$$|x| \leq 6.5 \quad (3.38)$$

The parameters in Eqs. (3.3) and (3.4) corresponding to the guard condition are given as

$$\begin{aligned} \mathbf{L}_x &= [-1, 1]^T \\ \mathbf{L}_\eta &= \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \\ \bar{\eta} &= [-6.5, 6.5]^T \\ \Sigma_\eta &= \begin{bmatrix} 0.1^2 & 0 \\ 0 & 0.1^2 \end{bmatrix} \end{aligned} \quad (3.39)$$

The mode-conditioned GSFs of the proposed algorithm are initialized as (the number of Gaussians, N , is set to 2, and the parameters for the pruning and merging are set as $\delta = 0.7$, $\kappa = 0.15$, and $\gamma = 0.02$)

$$\begin{aligned} p(x(0)|s(0) = 1) &= 0.8 \mathcal{N}(-1.2, 0.2^2) + 0.2 \mathcal{N}(1.2, 0.2^2) \\ p(x(0)|s(0) = 2) &= 0.4 \mathcal{N}(-1.5, 0.2^2) + 0.6 \mathcal{N}(1.5, 0.2^2) \end{aligned} \quad (3.40)$$

with the initial mode probabilities given by

$$\begin{aligned} p(s(0) = 1) &= 0.7 \\ p(s(0) = 2) &= 0.3 \end{aligned} \quad (3.41)$$

For performance evaluation, the proposed algorithm, called the state-dependent-mode-transition hybrid estimation with the GSF (denoted as ‘SD-GSF’), is compared to two other algorithms that also use the same proposed framework for the state-dependent mode transition but use the EKF (denoted as ‘SD-EKF’) or UKF (denoted as ‘SD-UKF’) as the mode-matched filter, respectively.

The actual history of x and corresponding measurements z are depicted in Fig. 3.2 for a single run where $x(0) > 0$ and the mode changes around $k = 20$ (when the guard condition Eq. (3.38) is satisfied). All the three algorithms (SD-GSF, SD-EKF, and SD-UKF) are applied and their performances are evaluated by calculating the estimation error for both the continuous and discrete states.

The root mean square (RMS) estimation error of the continuous state x is computed by the 500 Monte Carlo simulations as presented in Fig. 3.3. The mean of the RMS over the simulation time is also computed and presented in Table 3.1. As shown, the proposed algorithm produces more accurate estimation results than the other two algorithms. The estimation performance is also compared for the discrete state estimation as presented in Table 3.2 where the average, maximum, and minimum number of mismatch between the actual and estimated modes (i.e., $s(k) \neq \hat{s}(k)$) are computed along the simulation time (50 steps) for the 500 Monte Carlo simulations. It is shown that the proposed algorithm also outperforms the other algorithms in discrete state estimation.

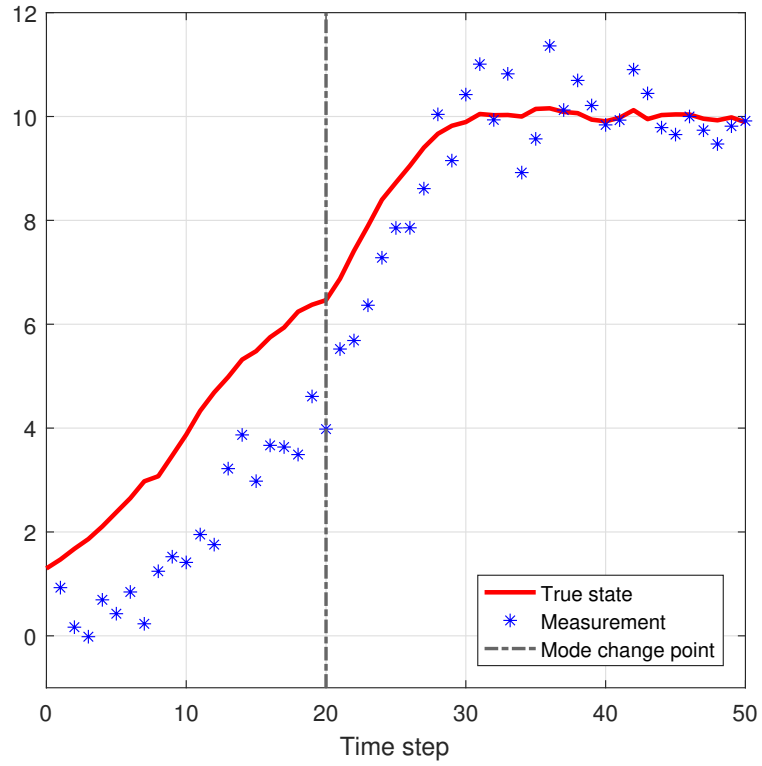


Figure 3.2. Actual history of x and measurements z for a single run ($x(0) > 0$)

Table 3.1. Mean RMS error for x (500 Monte Carlo simulations)

	Average RMS continuous state estimation error
Proposed (SD-GSF)	7.42
SD-EKF	8.88
SD-UKF	8.46

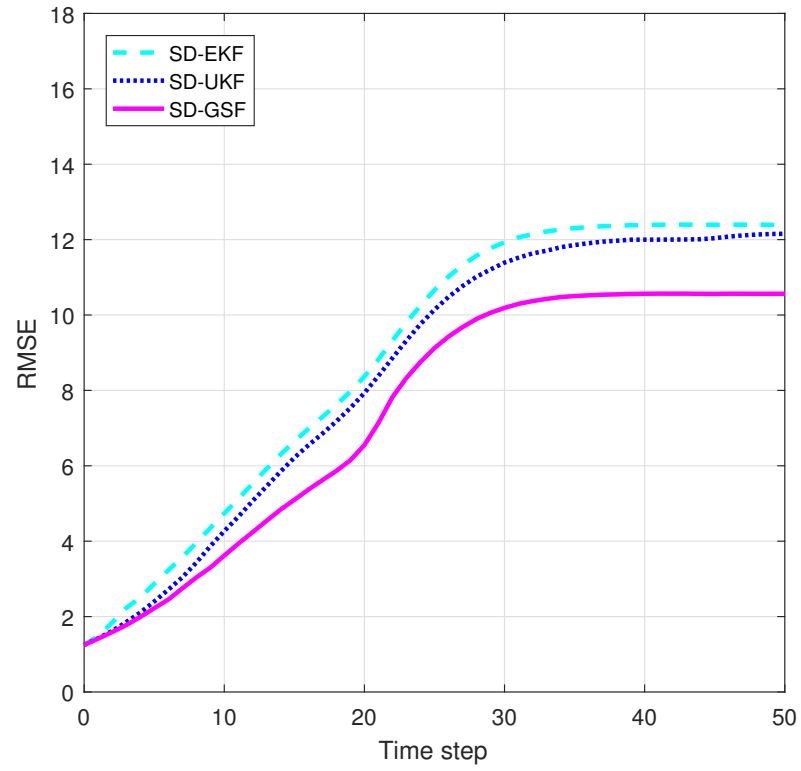


Figure 3.3. RMS estimation error for x (500 Monte Carlo simulations)

Table 3.2. Discrete state estimation mismatch (500 Monte Carlo simulations)

	Average	Min	Max
Proposed (SD-GSF)	1.05	0	3
SD-EKF	1.09	0	3
SD-UKF	7.49	0	30

4. APPLICATION TO AIR TRAFFIC CONTROL: ESTIMATED TIME OF ARRIVAL PREDICTION

In this chapter, an aircraft trajectory estimation and estimated time of arrival (ETA) prediction algorithm is proposed based on the hybrid system modeling and hybrid state estimation presented in the previous chapters.

4.1 Background and Motivations

As the National Airspace System (NAS) has been facing the pressure of steadily increasing air travel demand, air traffic congestion and flight delays around airports have become major issues in air traffic management. To mitigate such issues, a new framework known as the Next Generation Air Transportation System (NextGen) [1] has been proposed, under which aircraft operate with reduced separation thresholds and have the flexibility of changing their routes for the safe and efficient use of a given airspace (e.g., 4D trajectory-based operations). Under NextGen, a key requirement for safe and efficient air traffic flow management in terminal airspace is accurate knowledge of the aircraft's states (e.g., position, velocity, and flight mode) and accurate prediction of the aircraft's estimated time of arrival (ETA). Using the accurate state information of the aircraft, more efficient airborne spacing with reduced separation thresholds can be achieved, and thereby, air traffic flow near an airport can be effectively managed within its capacity. In addition, the accurate prediction of ETA can play an important role in enhancing the efficiency of airport surface operations, since it can reduce unnecessary delays in taxi times caused by inaccurate ETA of arrival aircraft [31].

A lot of research has been done for aircraft tracking and trajectory prediction, and [32] and [33] provide comprehensive literature reviews. In general, aircraft tracking

and trajectory prediction can be performed in a unified framework, but one of the most important differences between these tasks is that the aircraft tracking problem has regularly updated observations to estimate the aircraft's states, while the trajectory prediction problem needs to predict the future trajectory without observations. So, in most cases, the aircraft's states are first estimated in the tracking step, and then in the prediction step, the estimated state information is used as an initial condition to compute the aircraft's future trajectory. Many trajectory prediction algorithms have been developed using kinematic models due to their simplicity [34–38]. In these algorithms, the future trajectory is projected forward based only on the estimated velocity and acceleration information. Since these models cannot accurately capture the maneuver uncertainty of the aircraft, they could cause large errors and only work for short look-ahead times. To improve the prediction accuracy, there have been efforts to incorporate flight intent information into prediction models [39–42]. If a specific flight intent is inferred from the current estimated states, then a specific prediction model corresponding to the inferred flight intent is chosen to predict the future trajectory. However, only short-term flight intents have been exploited in the intent-based prediction models, and therefore, their prediction accuracy is degraded for long look-ahead times. In addition, under the 4D trajectory-based operation concept of NextGen, the aircraft's intent could frequently change due to the air traffic or weather conditions in the terminal airspace, which makes trajectory prediction more challenging.

Due to the development of advanced flight data communication systems, long-term flight intent information such as flight plans or airline procedures have become available for more accurate trajectory prediction [43–45]. In particular, the long-term flight intent information can benefit the trajectory prediction of the aircraft during the descent phase. This is because the aircraft is subject to strictly follow its flight procedure during descent for safe and efficient terminal airspace operation. In fact, under NextGen, an aircraft's onboard data including its flight management system (FMS) setting information (i.e., flight plan) becomes available to air traffic controllers

(ATC) and is frequently updated through data communication [1]. This implies that a more accurate prediction model can be developed which can exploit the flight plan information explicitly, and thereby predict future trajectories more accurately.

In this research, we propose a stochastic hybrid system model to describe the behavior of an aircraft along its flight plan (with the focus on the descent phase as related to ETA computation) since the aircraft's behavior is composed of both discrete transitions between a number of flight modes (discrete states) and continuous motion corresponding to a specific flight mode (continuous states). In each flight mode, we derive a nonlinear dynamic model of the aircraft's continuous motion and also derive a wind model to incorporate the effects of wind disturbance on the aircraft's motion. Then, we model the discrete transitions between the flight modes by using the continuous state-dependent transition probabilities. This is reasonable since the flight mode changes along a given flight plan are triggered when some conditions on the continuous states (e.g., position and velocity) are satisfied. Based on the obtained stochastic hybrid system model containing the nonlinear continuous dynamics and multiple flight modes with continuous state-dependent transitions, we then develop an algorithm for both aircraft tracking and ETA prediction based on the nonlinear hybrid state estimation algorithm proposed in Chapters 2 and 3. The proposed algorithm first estimates the aircraft's continuous and discrete states using available measurements (aircraft tracking) and then propagates the estimates using the stochastic hybrid system model to predict the future trajectory and compute the corresponding ETA (ETA prediction).

4.2 Stochastic Hybrid System Model for Aircraft in the Descent Flight Procedure

The Federal Aviation Administration (FAA) has summarized the general flight procedures in a typical flight as shown in Fig. 4.1. A flight typically consists of a collection of different flight modes and trigger conditions that govern the switches from

one flight mode to another. A typical medium-range fully procedured flight profile is listed in Table 4.1, where the operational procedures T_{max} , T_{min} , M , CAS , and $PATH$ represent the maximum thrust procedure, the minimum thrust procedure, the constant Mach procedure, the constant calibrated airspeed procedure, and the constant flight path angle procedure, respectively [46]. In different flight modes, the aircraft's behaviors are governed by different continuous dynamics and subject to different constraints. The aircraft can also switch among different flight modes, which is governed by the trigger conditions. Note that the trigger conditions are based on the aircraft's continuous states such as airspeed (V , V_{CAS}) and altitude (h). So, given a flight procedure or plan, the behavior of the aircraft can be described by a stochastic hybrid system having interacting continuous states and discrete states (modes) with continuous state-dependent mode transitions. In this section, the aircraft's motion in the descent stage along its flight plan is modeled as a stochastic hybrid system using the hybrid system model introduced in Chapter 3.



Figure 4.1. Different phases during a flight (source: FAA)

From the general flight phases and their operational procedures shown in Table 4.1, the flight modes most relevant to the descent stage are Constant Mach, Constant Calibrated Airspeed, Constant Deceleration, Approach, and Landing. These flight modes are then connected as a mode sequence with trigger conditions (guard conditions) which govern the timing of the switches between one flight mode and the

Table 4.1. Medium-range fully procedured flight profile [46]

Mode	Name	Aircraft Configuration	Trigger Condition	Operational Procedure
0	Takeoff	Takeoff	$V = 1.3V_{SIC}$	T_{max}
1	Initial Climb	Initial Climb	$V = 1.3V_{SCR}$	T_{max}
2	Free climb	Cruise	$h = 10,000 \text{ ft}$	T_{max}
3	Climb accel	Cruise	$V_{CAS} = 300 \text{ kt}$	$h = 10,000 \text{ ft}, T_{max}$
4	Climb CAS	Cruise	$M = 0.78$	$CAS = 300 \text{ kt}, T_{max}$
5	Climb Mach	Cruise	$h = FL360$	$M = 0.78, T_{max}$
6	Cruise	Cruise	-	$h = FL360, M = 0.78$
7	Descent Mach	Cruise	$V_{CAS} = 300 \text{ kt}$	$M = 0.78, T_{max}$
8	Descent CAS	Cruise	$h = 10,000 \text{ ft}$	$CAS = 300 \text{ kt}, T_{max}$
9	Descent decel	Cruise	$V_{CAS} = 250 \text{ kt}$	$h = 10,000 \text{ ft}, T_{max}$
10	Free descent	Cruise	$h = 6000 \text{ ft}$	T_{min}
11	Approach	Approach	$h = 2000 \text{ ft}$	$PATH = -3^\circ$
12	Landing	Landing	-	$PATH = -3^\circ$

following mode (see Fig. 4.2). In Fig. 4.2, TCs denote triggering conditions, for example, TC1: $V_{CAS} = 300$ kt, TC2: $h = 10,000$ ft, TC3: $V_{CAS} = 250$ kt, TC4: $h = 6,000$ ft, and TC5: $h = 2,000$ ft, where V_{CAS} is the calibrated airspeed and h is the altitude of the aircraft. Note that the flight mode sequence and the trigger conditions can vary according to specific flight plans or procedures. This variability can be easily dealt with by the stochastic hybrid system by appropriately defining the guard conditions in Eq. (3.3). In the following sections, we present the nonlinear continuous dynamics for the aircraft's continuous motion in each flight mode, and derive a continuous state-dependent flight mode transition model using a set of guard conditions.

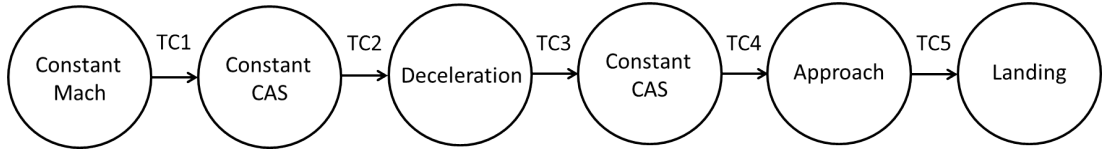


Figure 4.2. Discrete mode sequence during the descent stage

4.2.1 Continuous State Dynamics of Aircraft

We consider the longitudinal motion of an aircraft in the vertical plane with the assumptions of a flat earth and constant gravitational acceleration [47]. For the wind disturbance, we assume the standard atmosphere where the atmospheric properties vary with altitude, and take into account along-track wind effects. Because the vertical component of the wind along the track is relatively small compared to the horizontal component, we consider only the horizontal component of the wind along the track W_u in the equations of motion (the dynamic model for W_u will be presented shortly). Assuming that the angle of attack and the angle of the engine

thrust relative to the aircraft's body axis are small, the equations of the longitudinal motion of an aircraft are obtained as follows [47]:

$$\dot{u} = V \cos \gamma + W_u \quad (4.1)$$

$$\dot{h} = V \sin \gamma \quad (4.2)$$

$$\dot{V} = \frac{T - D}{m} - g \sin \gamma - \dot{W}_u \cos \gamma \quad (4.3)$$

$$\dot{\gamma} = \frac{1}{V} \left(\frac{L}{m} - g \cos \gamma + \dot{W}_u \sin \gamma \right) \quad (4.4)$$

$$\dot{m} = -Q_f \quad (4.5)$$

where u denotes the horizontal position, h is the altitude, V is the true airspeed, γ is the flight path angle, m is the mass of the aircraft, T denotes the thrust, D is the drag, L is the lift, Q_f is the fuel consumption rate, and g is the gravitational acceleration. In general, to solve the above equations, T , L , and D need to be specified either directly by operating constraints on themselves (such as idle thrust) or indirectly by motion constraints such as motion at constant Mach (these constraints are derived from the characteristics of each flight mode).

To describe the effect of wind disturbance, we consider a wind model \mathbf{W} which contains the deterministic \mathbf{W}_d and stochastic \mathbf{W}_s components as follow:

$$\mathbf{W} = \mathbf{W}_d + \mathbf{W}_s \quad (4.6)$$

The deterministic component of wind can be computed from the meteorological prediction which is available in Rapid Update Cycle (RUC) data from the National Oceanic and Atmospheric Administration (NOAA). The wind forecast data is updated with a one hour assimilation cycle, thus we can consider them as time-invariant during descent/approach flight phases. So, from the data, we can build a model describing deterministic wind profiles (both in speed and direction) for a given airport and specific time. For example, for the one hour period between UTC 5:00 and UTC 6:00 on September 22, 2004 at the Louisville International Airport (KSDF), we build a wind model where the wind direction is computed as constant ($\varphi_W = 275^\circ$ from

true North), and the wind speed is represented as a linear or logarithmic function of altitude h as (we will use this model in the simulation section) [48]:

$$W_d(h) = \begin{cases} \frac{40.6159}{26000}h + 8.7626 & [kt], & 10000 \text{ } ft \leq h \leq 36100 \text{ } ft \\ -11.4800 + 3.8939 \ln(h) & [kt], & h < 10000 \text{ } ft \end{cases} \quad (4.7)$$

For the stochastic component of the wind disturbance, we consider an integral-state model [49] which has the stochastic wind component W_s and its time derivative \dot{W}_s as the states, under the assumption that the stochastic wind acceleration \ddot{W}_s is an independent random process:

$$\begin{bmatrix} \dot{W}_s \\ \ddot{W}_s \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} W_s \\ \dot{W}_s \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} w(t) \quad (4.8)$$

where $w(t)$ is a zero-mean white Gaussian noise process with

$$E[w(t + \tau)w(t)] = Q(t)\delta(t - \tau)$$

(here E denotes the expectation and δ is the Dirac delta function). Then, based on the deterministic and stochastic wind models, the along-track wind speed profile W_u which we consider in this study can be computed as:

$$W_u = -(W_d + W_s) \cos(\psi - \varphi_w) \quad (4.9)$$

where ψ is the heading angle of the aircraft, which is usually constant in each flight mode. In the following subsections, the constraints for each flight mode in the descent and approach phase are specified, and the corresponding continuous dynamics of the aircraft with wind disturbance are derived from the general equations of motion (Eqs. (4.1)-(4.5)) [50, 51].

Constant Mach Mode

In this mode, the constant Mach number is used as a constraint. Using the constant Mach constraint and the definition of the true airspeed $V = Ma(h)$ (where

M is the Mach number and $a(h)$ is the speed of sound which varies with the altitude), the time derivative of the true airspeed can be obtained as $\dot{V} = M(da/dh)\dot{h}$. The equations of motion are then written as:

$$\dot{u} = V \cos \gamma + W_u \quad (4.10)$$

$$\dot{h} = V \sin \gamma \quad (4.11)$$

$$\dot{\gamma} = \frac{1}{V} \left(\frac{L}{m} - g \cos \gamma + \dot{W}_u \sin \gamma \right) \quad (4.12)$$

$$\dot{m} = -Q_f \quad (4.13)$$

with the equilibrium condition for the time derivative of the true airspeed \dot{V} :

$$\frac{T - D}{m} - g \sin \gamma - \dot{W}_u \cos \gamma = \frac{da}{dh} a M^2 \sin \gamma \quad (4.14)$$

Constant Calibrated Airspeed (CAS) Mode

In this mode, the constant calibrated airspeed is used as a constraint. The time derivative of the true airspeed is $\dot{V} = (\partial V / \partial V_{CAS}) \dot{V}_{CAS} + (\partial V / \partial h) \dot{h}$, where V_{CAS} denotes the calibrated airspeed. Using $\dot{V}_{CAS} = 0$ and Eq. (4.2), the equations of motion are obtained as:

$$\dot{u} = V \cos \gamma + W_u \quad (4.15)$$

$$\dot{h} = V \sin \gamma \quad (4.16)$$

$$\dot{\gamma} = \frac{1}{V} \left(\frac{L}{m} - g \cos \gamma + \dot{W}_u \sin \gamma \right) \quad (4.17)$$

$$\dot{m} = -Q_f \quad (4.18)$$

with the equilibrium condition for the time derivative of the true airspeed \dot{V} :

$$\frac{T - D}{m} - g \sin \gamma - \dot{W}_u \cos \gamma = \frac{\partial V}{\partial h} V \sin \gamma \quad (4.19)$$

Constant Deceleration Mode

In this mode, the constant deceleration rate is used as a constraint. For the constant deceleration constraint, we have $\dot{V} = c$, where c is a negative constant which is determined by operating conditions when this mode is active. The corresponding equations of motion are represented as:

$$\dot{u} = V \cos \gamma + W_u \quad (4.20)$$

$$\dot{h} = V \sin \gamma \quad (4.21)$$

$$\dot{\gamma} = \frac{1}{V} \left(\frac{L}{m} - g \cos \gamma + \dot{W}_u \sin \gamma \right) \quad (4.22)$$

$$\dot{m} = -Q_f \quad (4.23)$$

with the equilibrium condition for the time derivative of the true airspeed \dot{V} :

$$\frac{T - D}{m} - g \sin \gamma - \dot{W}_u \cos \gamma = c \quad (4.24)$$

Horizontal Deceleration Mode

In this mode, the constant altitude is used as a constraint. For the constant altitude constraint, we have $\dot{h} = 0$ and $\gamma = 0$ from Eq. (4.2). Thus, the time derivative of the flight path angle, $\dot{\gamma}$, is zero and the lift is $L = mg$ from Eq. (4.4). The corresponding equations of motion are written as:

$$\dot{u} = V \cos \gamma + W_u \quad (4.25)$$

$$\dot{h} = V \sin \gamma \quad (4.26)$$

$$\dot{V} = \frac{T - D}{m} - g \sin \gamma - \dot{W}_u \cos \gamma \quad (4.27)$$

$$\dot{m} = -Q_f \quad (4.28)$$

with the equilibrium conditions for the flight path angle and the lift:

$$\gamma = 0, \quad L = mg \quad (4.29)$$

Approach Mode

In this mode, the constant ground path angle γ_g is used as a constraint. The ground path angle is represented as $\tan \gamma_g = V \sin \gamma / (V \cos \gamma + W_u)$, and from this, we can derive the derivative of the flight path angle $\gamma = \gamma(V, W_u, \gamma_g)$ as $\dot{\gamma} = \frac{\partial \gamma}{\partial V} \dot{V} + \frac{\partial \gamma}{\partial W_u} \dot{W}_u + \frac{\partial \gamma}{\partial \gamma_g} \dot{\gamma}_g$. The corresponding equations of motion are written as:

$$\dot{u} = V \cos \gamma + W_u \quad (4.30)$$

$$\dot{h} = V \sin \gamma \quad (4.31)$$

$$\dot{V} = \frac{T - D}{m} - g \sin \gamma - \dot{W}_u \cos \gamma \quad (4.32)$$

$$\dot{m} = -Q_f \quad (4.33)$$

with the equilibrium condition for the time derivative of the flight path angle $\dot{\gamma}$ (where $\dot{\gamma}_g = 0$):

$$\frac{1}{V} \left(\frac{L}{m} - g \cos \gamma + \dot{W}_u \sin \gamma \right) = \frac{\partial \gamma}{\partial V} \dot{V} + \frac{\partial \gamma}{\partial W_u} \dot{W}_u \quad (4.34)$$

Let us define a continuous state vector $\mathbf{x}(t)$ as:

$$\mathbf{x} = [u, h, V, \gamma, m, W_s, \dot{W}_s]^T \quad (4.35)$$

and define a discrete state q as $q = 1$: “Constant Mach Mode”, $q = 2$: “Constant Calibrated Airspeed Mode”, $q = 3$: “Constant Deceleration Mode”, $q = 4$: “Horizontal Deceleration Mode”, and $q = 5$: “Approach Mode”. Then, for each flight mode q , the equations of motion under the wind disturbance can be represented as the following mode-specific continuous dynamics:

$$\dot{\mathbf{x}}(t) = \tilde{\mathbf{f}}_q(\mathbf{x}(t)) + \tilde{\mathbf{w}}_q(t) \quad (4.36)$$

where $\tilde{\mathbf{f}}_q$ is given by the equations of motion corresponding to each mode (Eqs. (4.10)-(4.34)) and the wind disturbance model (Eqs. (4.7)-(4.9)); and $\tilde{\mathbf{w}}_q$ is a mode-dependent

zero-mean white Gaussian noise whose spectral density is $\tilde{\mathbf{Q}}_q$. Then, through discretization using the Euler-Maruyama method with a sampling time T [29], we finally get the discrete-time continuous dynamics for each mode q as:

$$\mathbf{x}(k+1) = \mathbf{f}_q(\mathbf{x}(k)) + \mathbf{w}_q(k) \quad (4.37)$$

where \mathbf{f}_q is the discretized dynamics of $\tilde{\mathbf{f}}_q$; $\mathbf{w}_q(k)$ are now a zero-mean white Gaussian noise with covariance \mathbf{Q}_q .

Under NextGen, ATC can access an aircraft's onboard sensor data, such as GPS measurements, through data communication, which implies that most of the aircraft's continuous state information is available for more accurate state estimation. In this research, it is assumed that noise-corrupted measurements of u , h , V , and m are available from the aircraft's onboard sensors. The measurement vector \mathbf{z} is then defined for all the flight modes as:

$$\mathbf{z}(k) = \mathbf{h}(\mathbf{x}(k)) + \mathbf{v}(k) \quad (4.38)$$

where

$$\mathbf{h}(\mathbf{x}(k)) := [u(k) \ h(k) \ V(k) \ m(k)]^T \quad (4.39)$$

and $\mathbf{v}(k)$ is a zero-mean white Gaussian noise with covariance \mathbf{R} . Note that the aircraft's mode-specific continuous dynamics (Eq. (4.37)) and measurement model (Eq. (4.38)) correspond to Eq. (2.1) in the general stochastic hybrid system model. The next step to complete the stochastic hybrid system modeling is to mathematically describe the flight mode transitions (i.e., the discrete state transitions), which correspond to the design of a set of guard conditions $G(i, j)$ in Eq. (3.3).

4.2.2 Flight Mode Transitions

Given a flight plan or procedure, the flight usually consists of several different flight modes and transitions. For example, Fig. 4.3 shows a typical flight profile for

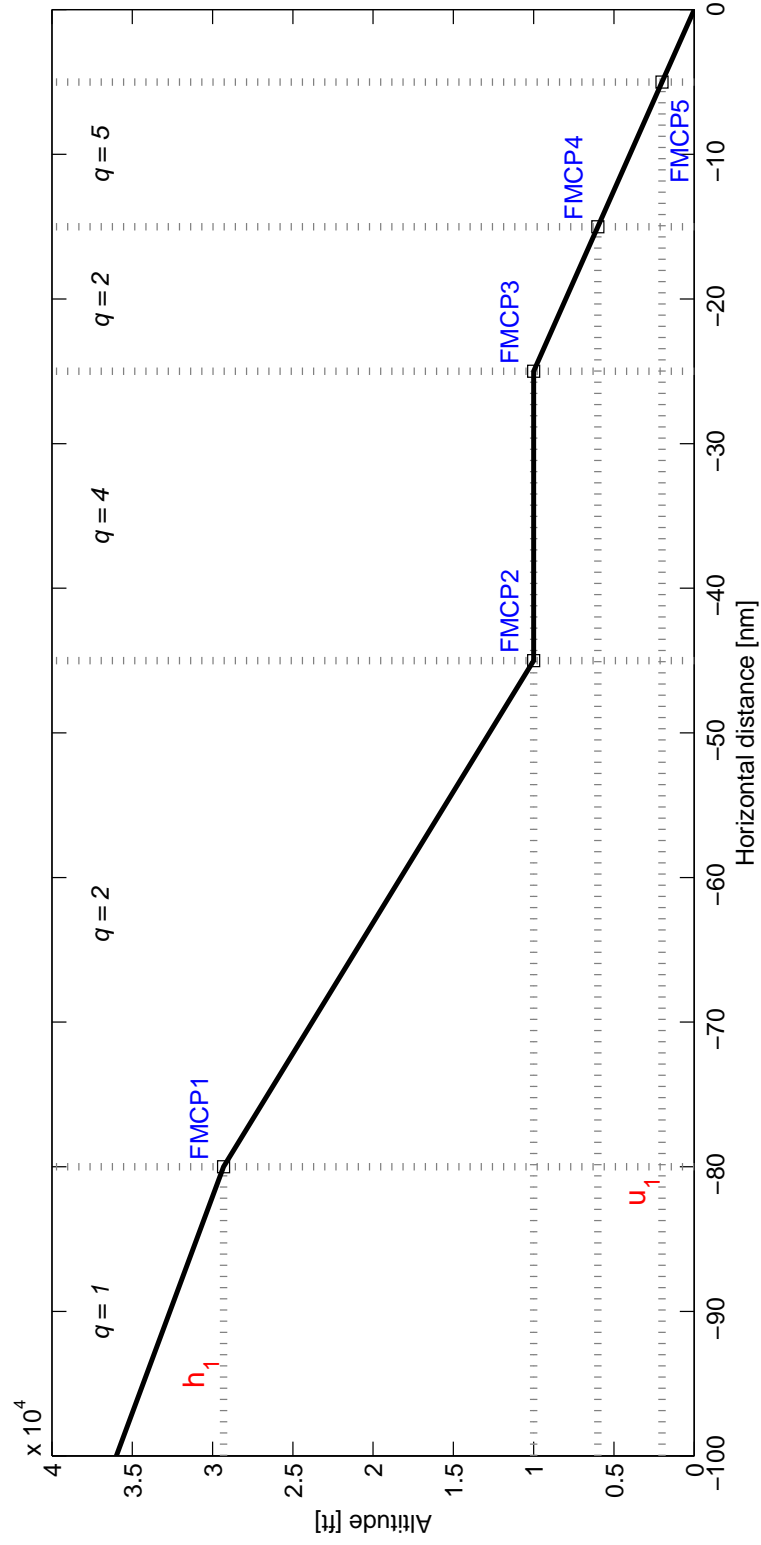


Figure 4.3. Typical flight profile for descent, approach, and landing phases

the descent, approach, and landing phases which are further divided into several flight modes. The flight mode transitions occur at various points which are marked as flight mode change points (FMCPs). The switching condition for each FMCP is determined by the given flight plan, and is dependent on the aircraft's continuous states such as the altitude and horizontal position. These continuous state-dependent discrete state transitions are then mathematically described by using guard conditions in the stochastic hybrid system framework. As an example, we present how to construct a guard condition corresponding to the flight mode transition at FMCP₁ in Fig. 4.3 (even though the guard conditions can be described in the general structure (Eq. (3.3)), its specific forms vary with specific flight plans). In this example, the aircraft initially descends at $M = 0.78$ in Constant Mach Mode ($q = 1$) approaching FMCP₁. When the speed of the aircraft reaches $V_{CAS} = 300$ knot calibrated airspeed (the corresponding altitude h_1^* can be computed which satisfying both $M = 0.78$ and $V_{CAS} = 300$ knot), i.e., when the aircraft reaches h_1^* , the aircraft's flight mode changes to Constant Calibrated Airspeed Mode ($q = 2$). In addition to that, due to a horizontal distance constraint imposed by FMCP₁, the flight mode transition ($q = 1$ to $q = 2$) is also triggered when the horizontal position of the aircraft reaches a specific distance u_1^* from the airport. That is, if $h < h_1^*$ or $u < u_1^*$, then the flight mode transition occurs, and the corresponding guard condition can be represented as:

$$\begin{aligned}
G(1, 1) &= G(2, 1) = G(3, 1) = G(4, 1) = \{[\mathbf{x}^T \ \theta^T]^T | L_{\mathbf{x},11}\mathbf{x} + L_{\theta,11}\boldsymbol{\theta} \leq \mathbf{0}\} \\
G(1, 2) &= G(2, 2) = G(3, 2) = G(4, 2) = G(1, 1)^c \\
G(1, 3) &= G(2, 3) = G(3, 3) = G(4, 3) = \emptyset \\
G(1, 4) &= G(2, 4) = G(3, 4) = G(4, 4) = \emptyset
\end{aligned} \tag{4.40}$$

where

$$L_{\mathbf{x},12} = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad L_{\theta,12} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{\theta}; \begin{bmatrix} u_1^* \\ h_1^* \end{bmatrix}, \Sigma_{\boldsymbol{\theta}}) \tag{4.41}$$

\emptyset denotes the empty set; and $G(\cdot, \cdot)^c$ denotes the complement of $G(\cdot, \cdot)$. Note that due to the navigation uncertainty, there is probabilistic variation of the actual flight profiles, and the actual FMCPs deviate from the nominal ones (e.g., h_1^* and u_1^* for FMCP₁). This uncertainty can be accounted for by the parameter θ in the guard condition, which has a multivariate Gaussian pdf with mean $\bar{\theta}$ and covariance Σ_θ . The mean $\bar{\theta}$ can be defined by the given nominal flight profile, and the covariance Σ_θ can be determined from the aircraft's navigation performance. Similarly to FMCP₁ presented as an example, the guard conditions can be derived for other FMCPs using the flight mode switching conditions imposed by the flight procedures.

4.3 Aircraft Tracking and ETA Prediction Algorithm

In this section, we propose an algorithm that solves both the aircraft tracking problem and the ETA prediction problem in a unified framework. The ETA prediction is based on the aircraft's continuous states, flight mode, and a priori knowledge of the future flight mode sequence. Since the mode sequence and switching conditions are already defined in the stochastic hybrid system model with a given flight plan, we only need to estimate the current aircraft's states (e.g., position and flight mode), and propagate the estimated states through the stochastic hybrid system model to predict the descent trajectory and ETA. So, the first step of the proposed algorithm is the hybrid state estimation where the pdfs of both the continuous states (e.g., position and velocity) and the discrete state (flight mode) are computed using noisy measurement data. Then, the state estimates obtained from the computed pdfs are propagated using the stochastic hybrid system model along the time horizon of prediction without measurement data.

It should be noted that the state estimates, $\hat{\mathbf{x}}(k)$ and $\hat{q}(k)$, can be computed using the nonlinear hybrid state estimation algorithm proposed in Chapter 3. In order to predict the future trajectory of the aircraft, the state estimates ($\hat{\mathbf{x}}(k)$ and $\hat{q}(k)$)

obtained using the measurements up to the current time k need to be propagated through the stochastic hybrid system model. The propagation (or prediction) can be performed through a process similar to the hybrid state estimation presented in the previous section. The only difference is that the propagation is performed without measurements. So, there is no measurement update of the pdfs of both the continuous states and the discrete state. That is, in the mode-conditioned estimation (*Step 2*) and the mode probability update (*Step 3*) steps, the posterior distributions, $p(\mathbf{x}(k')|q(k') = j, \mathbf{Z}^{k'})$ and $p(q(k') = j|\mathbf{Z}^{k'})$, are now computed as the same as the prior distribution, $p(\mathbf{x}(k')|q(k') = j, \mathbf{Z}^{k'-1})$ and $p(q(k') = j|\mathbf{Z}^{k'-1})$, respectively for time $k' > k$. The estimated time of arrival t_{eta} is then computed by

$$t_{eta} = k_{eta}T_s \quad (4.42)$$

where T_s is a sampling time used to obtain the discrete-time dynamics in Eq. (4.37) (in this study, T_s is chosen as $T_s = 1 \text{ sec}$, since 1 sec is one of the normal rates of onboard sensors (e.g., GPS) in practice) and

$$k_{eta} = \inf \left\{ k' | \hat{h}(k') < h_{runway}, k' > k \right\} \quad (4.43)$$

where \hat{h} and h_{runway} are the altitude estimate and altitude of runway, respectively. The overall structure of the proposed algorithm is illustrated in Fig. 4.4.

4.4 Numerical Simulations

In this section, the proposed aircraft tracking and ETA prediction algorithm is demonstrated with two illustrative examples: 1) a continuous descend approach (CDA) case and 2) a conventional approach (standard stair-case descending) case. In both cases, the actual flight descent procedures at the Louisville International Airport [48] are considered. For comparison, we consider an algorithm consisting of the IMM [52, 53] and dead reckoning [54, 55] which have been extensively used for the aircraft tracking (i.e., hybrid state estimation) and ETA prediction, respectively (for

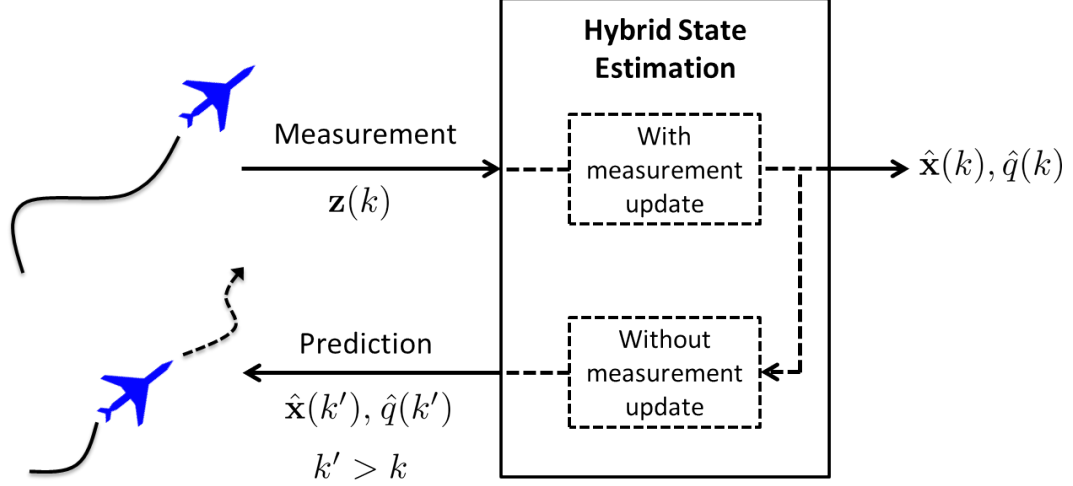


Figure 4.4. Structure of the proposed algorithm

the rest of the paper, we denote this algorithm as “IMM-DR”). The performances of the proposed algorithm and the IMM-DR algorithm are compared in terms of the aircraft tracking accuracy and the ETA prediction accuracy. One of the traditional approaches to aircraft tracking is a single EKF which uses a single dynamic model to describe the motion of an aircraft along its flight profile. Since the single dynamic model cannot accurately describe the different dynamics of the aircraft operating in different flight modes, it can be easily expected that the resulting tracking and prediction accuracies are worse than those of the IMM-based approach or of the proposed approach. In this sense, we do not consider the single EKF-based approach in the comparison study.

4.4.1 Example 1: CDA Procedure

In this example, the tracking and ETA prediction of a Boeing 767 following a CDA procedure at the Louisville International Airport are considered. The aircraft starts to descend at 36,000 *ft* and 96 nautical miles (*nm*) from the airport with an initial speed of 273 *kt* in Constant Mach mode. Then, the aircraft changes its flight

mode at 8 different FMCPs during the descending procedure. The operational procedures and trigger conditions (i.e., guard conditions) corresponding to each mode are summarized in Table 4.2 (also see Fig. 4.5 for the nominal trajectory along the CDA procedure). For simulation, we use the proposed stochastic hybrid system model where the parameters in the aircraft dynamics are obtained from the aircraft performance model in the Base of Aircraft Data (BADA) [56]. For all the modes, the covariance \mathbf{Q} of the process noise for the continuous dynamics is set as $\mathbf{Q} = \text{diag}[(10^{-3})^2 \text{ ft}^2, (10^{-3})^2 \text{ ft}^2, (0.2)^2 (\text{ft/s})^2, (10^{-3})^2 \text{ lb}^2]$, and the covariance of measurement noise is set as $\mathbf{R} = \text{diag}[5^2 \text{ ft}^2, 5^2 \text{ ft}^2, 1.7^2 (\text{ft/s})^2, 63.6^2 \text{ lb}^2]$. Then, the proposed algorithm and the IMM-DR algorithm are applied to estimate the aircraft's states and predict the ETA using the simulated sensor measurements \mathbf{z} . The tracking and ETA prediction accuracies are compared in Fig. 4.6 and Table 4.3, respectively. In Table 4.3, the ETA is predicted from the top of descent (TOD), meter fix (MF), initial approach fix (IAF), and final approach fix (FAF).

It is shown that the proposed algorithm produces more accurate state estimates and predicts more accurate ETA compared to the IMM-DR approach. In particular,

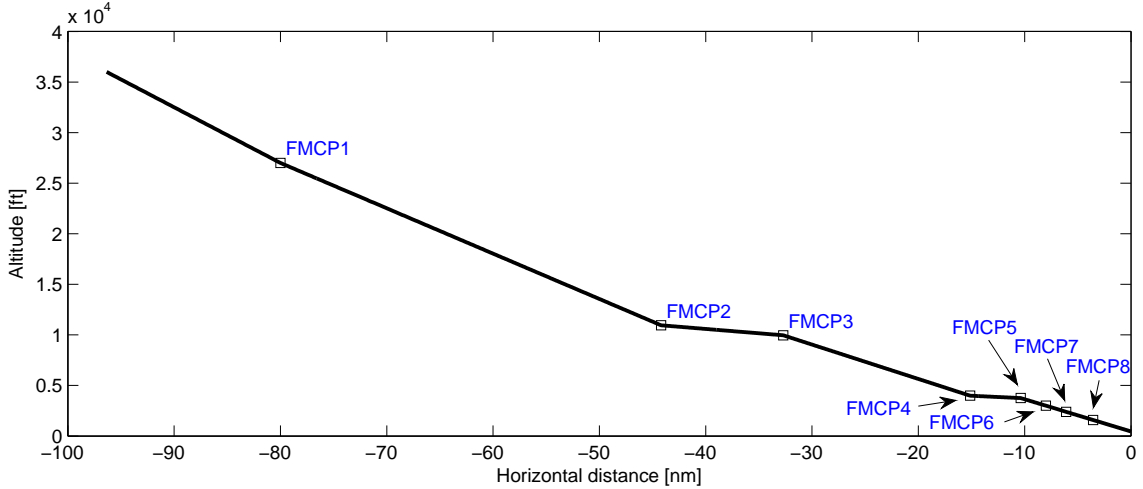


Figure 4.5. Nominal trajectory of a Boeing 767 along a CDA procedure

Table 4.2. Nominal flight profile of a Boeing 767 following a CDA procedure

Sequence	Phase	Mode	Operational procedure	Trigger condition
1	Descent	Constant Mach	$M = 0.82$	$V_{CAS} = 332.79 \text{ kt}$
2	Descent	Constant CAS	$V_{CAS} = 332.79 \text{ kt}$	$h = 11,000 \text{ ft}$
3	Descent	Constant Deceleration	$\dot{V} = -0.9504 \text{ kt/s}$	$V_{CAS} = 240 \text{ kt}$
4	Descent	Constant CAS	$V_{CAS} = 240 \text{ kt}$	$h = 4,000 \text{ ft}$
5	Descent	Constant Deceleration	$\dot{V} = -0.91 \text{ kt/s}$	$V_{CAS} = 180 \text{ kt}$
6	Approach	Constant CAS	$V_{CAS} = 180 \text{ kt}$	$h = 3,000 \text{ ft}$
7	Approach	Constant Deceleration	$\dot{V} = -0.295 \text{ kt/s}$	$V_{CAS} = 170 \text{ kt}$
8	Final Approach	Constant Deceleration	$\dot{V} = -0.6292 \text{ kt/s}$	$V_{CAS} = 135 \text{ kt}$
9	Final Approach	Constant CAS	$V_{CAS} = 135 \text{ kt}$	$h = 453 \text{ ft}$

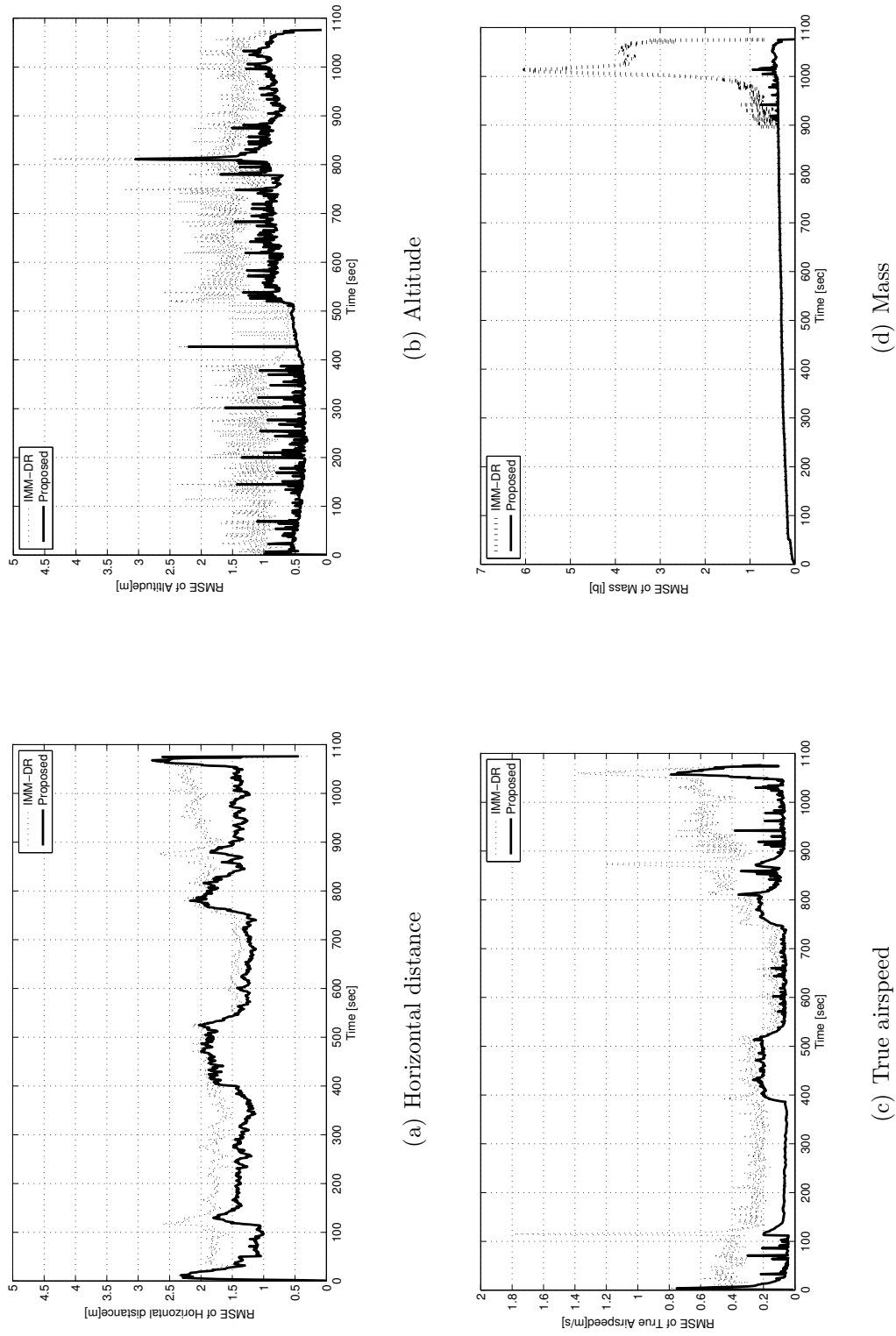


Figure 4.6. Comparison of tracking accuracy for Example 1: root-mean-square values of state estimation error for 100 runs

Table 4.3. Comparison of ETA prediction accuracy for Example 1: ETA error for 100 runs

Prediction point	Location	ETA prediction error (mean \pm standard deviation)	
		Proposed	IMM-DR
TOD	$u = -100 \text{ nm}, h = 36,000 \text{ ft}$	$11.33 \pm 5.03 \text{ s}$	$377.31 \pm 21.71 \text{ s}$
MF	$u = -44 \text{ nm}, h = 11,000 \text{ ft}$	$6.00 \pm 5.20 \text{ s}$	$257.21 \pm 17.17 \text{ s}$
IAF	$u = -11 \text{ nm}, h = 4,000 \text{ ft}$	$3.67 \pm 2.31 \text{ s}$	$56.71 \pm 4.37 \text{ s}$
FAF	$u = -6 \text{ nm}, h = 2,400 \text{ ft}$	$1.01 \pm 1.00 \text{ s}$	$28.99 \pm 1.90 \text{ s}$

the accuracy of the ETA prediction is significantly improved by using the proposed algorithm. This is because the proposed algorithm can explicitly utilize the flight procedure information (characterized by flight mode transitions and triggering conditions) for future trajectory prediction. On the contrary, the flight procedure information cannot be systematically incorporated in the IMM-DR approach. Since the IMM assumes constant mode transition probabilities, information on the continuous state-dependent mode transitions cannot be incorporated, and thus the algorithm cannot accurately predict the mode transitions. The performance of the proposed algorithm can also be indirectly compared to that of [53]. The result of the proposed algorithm shows that the largest ETA error from the TOD is less than 17 sec for about 20 min duration, which is more accurate than the roughly 30 sec error for about 20 min duration obtained by [53]. The accurate state estimates and ETA of the aircraft obtained from the proposed algorithm can play an important role in efficient terminal airspace operation. With more accurate ETA information for aircraft near an airport, the ATC can control the sequence and separation between the aircraft more tightly, increasing the throughput of the airport [57].

4.4.2 Example 2: Conventional Approach Procedure

In this example, the tracking and ETA prediction of a Boeing 767 following a conventional approach procedure (standard stair-case descending) at the Louisville International Airport are considered. For each mode, the operational procedures and mode transition conditions are summarized in Table 4.4, and the nominal trajectory along the conventional approach procedure is illustrated in Fig. 4.7. The same values of \mathbf{Q} and \mathbf{R} as in Example 1 are used for simulation. From the numerical simulation, it is also found that the proposed algorithm outperforms the IMM-DR method in terms of both tracking and ETA prediction (see Fig. 4.8 and Table 4.5). The ETA prediction error is similar or slightly larger when compared to Example 1, which is

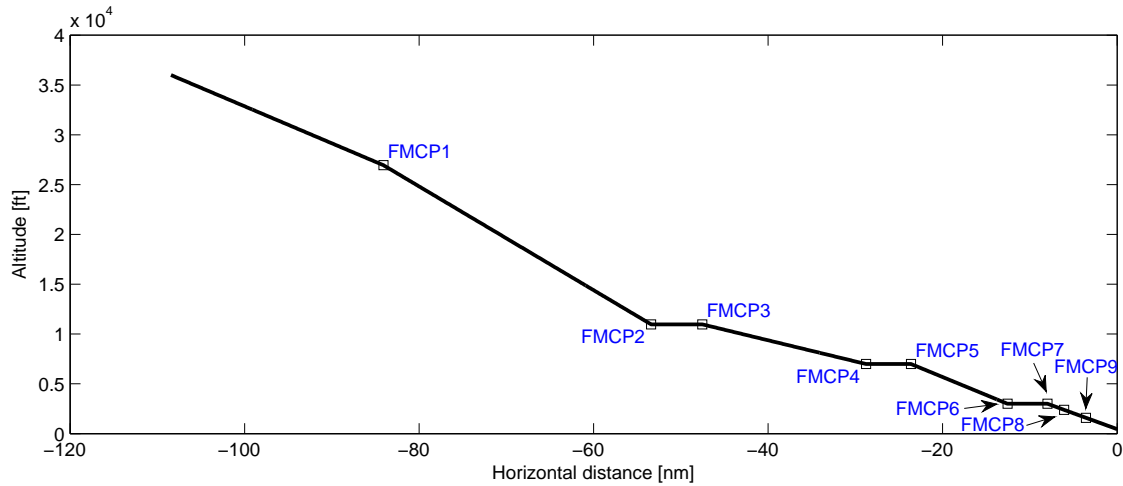


Figure 4.7. Nominal trajectory of a Boeing 767 along a conventional approach procedure

because the locations of the prediction points are similar for both examples, and the numbers of planned flight mode transitions are similar. As illustrated in both examples, the proposed algorithm can be applied to both the CDA procedure and the conventional approach procedure by appropriately modeling those procedures (i.e., the sequence of flight mode transitions and triggering conditions). Note that even though the proposed aircraft tracking and ETA prediction algorithm has been demonstrated with aircraft landing examples, it is general enough to be applied to the other phases of flight.

Table 4.4. Nominal flight profile of a Boeing 767 following a conventional approach procedure

Sequence	Phase	Mode	Operational procedure	Trigger condition
1	Descent	Constant Mach	$M = 0.82$	$V_{CAS} = 332.79 \text{ kt}$
2	Descent	Constant CAS	$V_{CAS} = 332.79 \text{ kt}$	$h = 11,000 \text{ ft}$
3	Descent	Horizontal Deceleration	$h = 11,000 \text{ ft}$	$V_{CAS} = 250 \text{ kt}$
4	Descent	Constant CAS	$V_{CAS} = 250 \text{ kt}$	$h = 7,000 \text{ ft}$
5	Descent	Horizontal Deceleration	$h = 7,000 \text{ ft}$	$V_{CAS} = 210 \text{ kt}$
6	Descent	Constant CAS	$V_{CAS} = 210 \text{ kt}$	$h = 3,000 \text{ ft}$
7	Descent	Horizontal Deceleration	$h = 3,000 \text{ ft}$	$V_{CAS} = 170 \text{ kt}$
8	Approach	Constant CAS	$V_{CAS} = 170 \text{ kt}$	$h = 2,400 \text{ ft}$
9	Final Approach	Constant Deceleration	$\dot{V} = -0.6292 \text{ kt/s}$	$V_{CAS} = 135 \text{ kt}$
10	Final Approach	Constant CAS	$V_{CAS} = 135 \text{ kt}$	$h = 453 \text{ ft}$

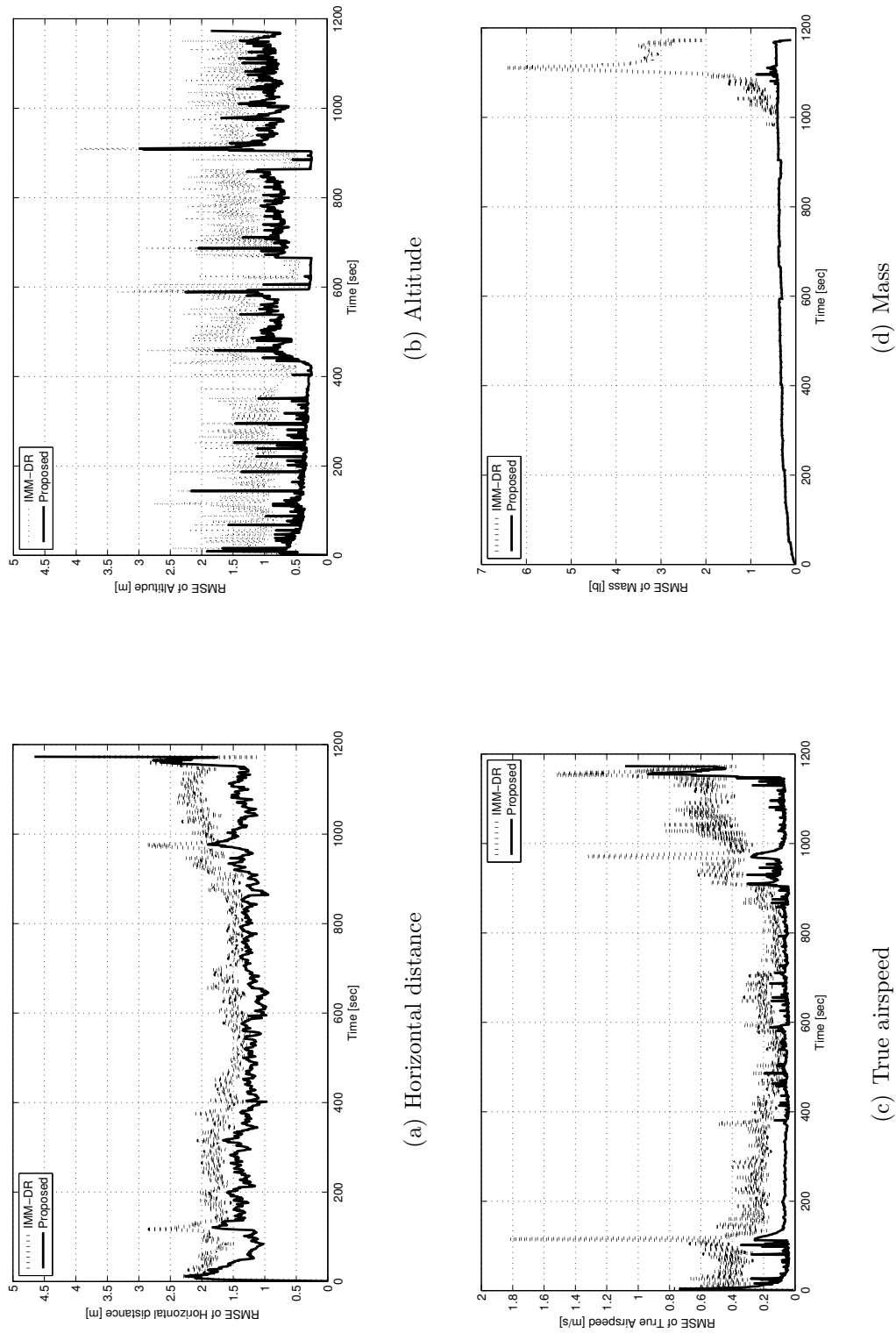


Figure 4.8. Comparison of tracking accuracy for Example 2: root-mean-square values of state estimation error for 100 runs

Table 4.5. Comparison of ETA prediction accuracy for Example 2: ETA error for 100 runs

Prediction point	Location	ETA prediction error (mean \pm standard deviation)	
		Proposed	IMM-DR
TOD	$u = -108 \text{ nm}, h = 36,000 \text{ ft}$	$11.84 \pm 5.86 \text{ s}$	$382.58 \pm 19.02 \text{ s}$
MF	$u = -47 \text{ nm}, h = 11,000 \text{ ft}$	$8.18 \pm 4.40 \text{ s}$	$261.02 \pm 19.02 \text{ s}$
IAF	$u = -8 \text{ nm}, h = 3,000 \text{ ft}$	$3.23 \pm 2.17 \text{ s}$	$48.18 \pm 9.03 \text{ s}$
FAF	$u = -6 \text{ nm}, h = 2,400 \text{ ft}$	$1.02 \pm 1.01 \text{ s}$	$28.41 \pm 2.02 \text{ s}$

5. APPLICATION TO UAS TRAFFIC MANAGEMENT: UAS SURVEILLANCE IN LOW-ALTITUDE AIRSPACE WITH GEOFENCING

In this chapter, a UAS tracking algorithm is proposed to improve the safety and efficiency of UAS operation under the UAS traffic management (UTM) based on the hybrid system modeling and hybrid state estimation.

5.1 Background and Motivations

In the UTM system [58], a new concept called *geofence* has been proposed and studied as a method to improve the safety and efficiency of UAS operation given an increasing density of UAS in low-altitude airspace. The geofence [58, 59] represents the boundary of a region in airspace assigned around a planned path of each UAS, which restricts the UAS from deviating from its approved flight plan during a pre-specified time interval (see Fig. 5.1). Once a flight plan is submitted to the UTM system by a UAS operator, including the origin, destination, way points, flight time, UAS type, operational capability, etc., the corresponding geofence is computed by the UTM system to reserve a volume of airspace for the planned flight of the UAS. When another UAS submits a request with its flight plan, the UTM system generates a geofence for that UAS such that it does not intersect with the existing geofences that were already assigned to other UAS currently operating in airspace. This allows the UTM system to safely and effectively manage many UAS flights, reducing the risk of potential collision.

In practice, due to many factors such as weather, traffic demand, control system failure, change of flight intents, etc., the actual trajectory of a UAS can deviate from

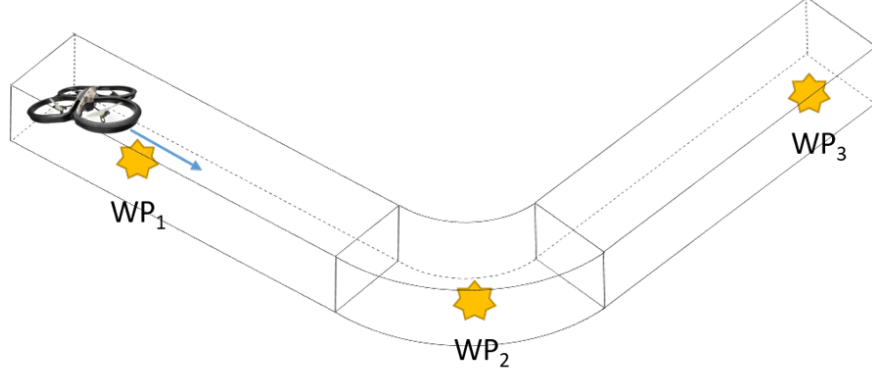


Figure 5.1. Illustration of geofence given a set of waypoints (WPs)

a predefined geofence and this may cause undesired use of airspace and/or conflicts with other UAS, which degrades the safety and efficiency of the UTM system. This implies that the actual trajectories of UAS should be accurately tracked in real-time and monitored to check if they remain within the geofences assigned to them, which leads to UAS tracking and conformance monitoring problems. In this research, our efforts focus on the development of a UAS tracking problem and the technical details will be discussed with illustrative numerical simulation results. The result will be extended to solving the conformance monitoring problem as future work.

There are two key points to consider for the UAS tracking: 1) the motion of a UAS is complex as it is composed of, for example, a sequence of straight flights and turning motions along its planned trajectory connecting waypoints, and 2) the trajectory of a UAS is restricted to be within a region in airspace bounded by an assigned geofence as regulated by the UTM system. The goal of this research is to develop a UAS tracking algorithm that can effectively account for these points so that it can produce more accurate tracks of UAS than the existing tracking algorithms, and thus support the safe and efficient operation of the UTM system.

Many tracking algorithms have been developed to estimate the position and velocity of moving targets. A simplest way is to use a single Kalman filter based on a single dynamics that assumes a certain type of behavior of the target. However,

it could cause significant tracking errors as the single dynamics cannot effectively describe the complex behavior of targets that can take various types of maneuvers. To account for this, the multiple model algorithms [60] have been developed based on the Bayesian framework, and they have been widely used in many applications of maneuvering target tracking such as air traffic surveillance [52]. The interacting multiple model (IMM) algorithm [10] is one of the most popular multiple model algorithms, and it is well known to produce good performances and cause less computational costs. However, the IMM algorithm models the discrete mode transitions as a Markov process with constant mode transition probabilities, and therefore its applications are limited to a certain type of the stochastic hybrid systems. In recent years, a new hybrid state estimation algorithm, called the state-dependent-transition hybrid estimation (SDTHE) [30], has been developed and applied to many aerospace applications [3, 8]. The SDTHE algorithm is based on the IMM framework, but it exploits the information of the discrete mode switching conditions dependent on the continuous states of the stochastic hybrid systems and computes the continuous-state-dependent mode transition probabilities explicitly, which is applicable to a more general class of the stochastic hybrid systems compared to the IMM algorithm. However, these approaches also have limitations in that they do not use additional information (e.g., constraints on the state) that needs to be incorporated for more accurate state estimation.

In this research, we first propose to model the behavior of a UAS as a stochastic hybrid system in which a set of stochastic continuous dynamical models is used to describe different physical motions of a UAS with uncertainties such as wind disturbance, and the transitions between these stochastic continuous dynamical models are described by the discrete dynamics. We also propose to use the constrained Kalman filtering technique [61] in hybrid state estimation framework as an attempt to explicitly incorporate the constraints on the motion of a UAS (imposed by geofence) into state estimation (tracking). It is shown that successful UAS tracking is achieved by

the effective extraction and incorporation of useful information about the UAS state.

5.2 Constrained Stochastic Hybrid System Model for UAS

Given a flight plan (e.g., a set of waypoints), a physically flyable trajectory can be generated, which in general is composed of a set of flight modes, for example, constant velocity mode, turning mode, constant acceleration mode, etc. Thus, the motion of a UAS can be described by 1) the physical behavior corresponding to each flight mode characterized by the continuous state dynamics and 2) discrete state (or mode) transitions between the different flight modes that are usually dependent on the continuous state of the UAS (e.g., a UAS is likely to begin a turning motion around the waypoints, i.e., based on the distance to waypoints). This nature of UAS behavior can be well modeled by a stochastic hybrid system having interacting continuous states and discrete states (modes) with continuous state-dependent mode transitions. In this section, a class of stochastic hybrid systems called the stochastic linear hybrid system (SLHS) is introduced, and its application to UAS behavior modeling is briefly discussed.

5.2.1 Stochastic Linear Hybrid System with Continuous-state-dependent Mode Transition

Consider the continuous state $\mathbf{x}(k) = [x_1(k), \dots, x_n(k)]^T \in \mathbb{R}^n$, the discrete state (mode) $q(k) \in \mathcal{Q} = \{1, 2, \dots, r\}$, and the measurement vector $\mathbf{z}(k) = [z_1(k) \dots z_p(k)]^T \in \mathbb{R}^p$, where k is the discrete-time index. In each mode $q(k)$, the continuous state dynamics and the measurement equation are given by,

$$\begin{aligned}\mathbf{x}(k+1) &= \mathbf{A}_{q(k)}\mathbf{x}(k) + \mathbf{B}_{q(k)}\mathbf{w}_{q(k)}(k) \\ \mathbf{z}(k) &= \mathbf{H}_{q(k)}\mathbf{x}(k) + \mathbf{v}_{q(k)}(k)\end{aligned}\tag{5.1}$$

where $\mathbf{A}_{q(k)} \in \mathbb{R}^{n \times n}$, $\mathbf{B}_{q(k)} \in \mathbb{R}^{n \times m}$ and $\mathbf{H}_{q(k)} \in \mathbb{R}^{p \times n}$ are the system, input and measurement matrices, and $\mathbf{w}_{q(k)}(k) \in \mathbb{R}^m$ and $\mathbf{v}_{q(k)}(k) \in \mathbb{R}^p$ are the process and measurement noises modeled as zero-mean white Gaussian noises with covariance $\mathbf{Q}_{q(k)}$ and $\mathbf{R}_{q(k)}$, respectively. The discrete state transitions are governed by a set of guard conditions $G(i, j)$, $i, j = 1, 2, \dots, r$, such that if the continuous state $\mathbf{x}(k)$ satisfies $G(i, j)$, the discrete state changes from $q(k) = i$ to $q(k + 1) = j$. The guard condition $G(i, j)$ is mathematically represented by a set as:

$$G(i, j) = \{[\mathbf{x}^T \ \boldsymbol{\theta}^T]^T | L_{\mathbf{x},ij} \mathbf{x} + L_{\boldsymbol{\theta},ij} \boldsymbol{\theta} \leq \mathbf{0}\} \quad (5.2)$$

where $L_{\mathbf{x},ij} \in \mathbb{R}^{l \times n}$ and $L_{\boldsymbol{\theta},ij} \in \mathbb{R}^{l \times s}$ are constant matrices characterizing the guard condition (i.e., the corresponding transition), and $\boldsymbol{\theta} \in \mathbb{R}^s$ is a random vector describing the uncertainties in the guard condition modeled by the Gaussian distribution

$$\boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{\theta}; \bar{\boldsymbol{\theta}}, \Sigma_{\boldsymbol{\theta}}) \quad (5.3)$$

with the mean $\bar{\boldsymbol{\theta}}$ and covariance $\Sigma_{\boldsymbol{\theta}}$. The discrete state transition from $q(k) = i$ to $q(k + 1) = j$ happens when $[\mathbf{x}(k)^T \ \boldsymbol{\theta}^T]^T \in G(i, j)$ (i.e., the transition is dependent on the continuous state $\mathbf{x}(k)$). The above stochastic hybrid system is called as the *stochastic linear hybrid system* [30].

5.2.2 UAS Behavior Modeling using Stochastic Linear Hybrid System

In what follows, we will first present the continuous dynamics of a UAS which represents the continuous motion corresponding to each flight mode, and then illustrate how the continuous-state-dependent mode transition model (i.e., a set of guard conditions) can be derived for a given set of waypoints. For simplicity, we consider two-dimensional motion and thus define the continuous state of a UAS as $\mathbf{x} = [\zeta, \eta, \dot{\zeta}, \dot{\eta}]^T$, where ζ and η are the Cartesian coordinates of the UAS's position in a given local frame. Note that this can be easily extended to the general three-dimensional motion.

Continuous State Dynamics

As discussed, the UASs trajectory is composed of a set of flight modes [62]: 1) Constant Velocity (CV) Mode (cruise mode):

$$\mathbf{x}(k+1) = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}(k) + \begin{bmatrix} \frac{T^2}{2} & 0 \\ 0 & \frac{T^2}{2} \\ T & 0 \\ 0 & T \end{bmatrix} \mathbf{w}(k) \quad (5.4)$$

where T is the sampling time, 2) Coordinated Turn (CT) Mode (turning mode):

$$\mathbf{x}(k+1) = \begin{bmatrix} 1 & 0 & \frac{\sin \omega T}{\omega} & -\frac{1-\cos \omega T}{\omega} \\ 0 & 1 & \frac{1-\cos \omega T}{\omega} & \frac{\sin \omega T}{\omega} \\ 0 & 0 & \cos \omega T & -\sin \omega T \\ 0 & 0 & \sin \omega T & \cos \omega T \end{bmatrix} + \begin{bmatrix} \frac{T^2}{2} & 0 \\ 0 & \frac{T^2}{2} \\ T & 0 \\ 0 & T \end{bmatrix} \mathbf{w}(k) \quad (5.5)$$

where ω is the turn rate of the UAS of which sign determines the turn direction (left/right), and 3) Constant Acceleration (CA) Mode (along a straight line):

$$\mathbf{x}(k+1) = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}(k) + \begin{bmatrix} \frac{T^2}{2} & 0 \\ 0 & \frac{T^2}{2} \\ T & 0 \\ 0 & T \end{bmatrix} \mathbf{u}(k) + \begin{bmatrix} \frac{T^2}{2} & 0 \\ 0 & \frac{T^2}{2} \\ T & 0 \\ 0 & T \end{bmatrix} \mathbf{w}(k) \quad (5.6)$$

where $\mathbf{u}(k) = [a_\zeta, a_\eta]^T$ is the acceleration of which direction can be determined based on the waypoint configuration.

In this research, it is assumed that noise-corrupted measurements of the position of the UAS are available by a surveillance system (e.g., radar). Therefore, the measurement vector $\mathbf{z}(k)$ is defined for all the flight modes as:

$$\mathbf{z}(k) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \mathbf{x}(k) + \mathbf{v}(k) \quad (5.7)$$

Continuous-state-dependent Flight Mode Transition

The flight path of the UAS can be considered as a sequence of waypoints, and the waypoints are chosen such that the UAS has flight mode change points (FMCP) in the area near the waypoints. The FMCP is stochastic due to uncertainties such as navigation errors or wind, and for each nominal FMCP, there exists a set of guard conditions $G(i, j)$ for the flight mode transition from mode i to mode j . A general structure of the flight mode transition model is shown in Fig. 5.2, and here, each guard condition from mode i to mode j is denoted as C_{ij} .

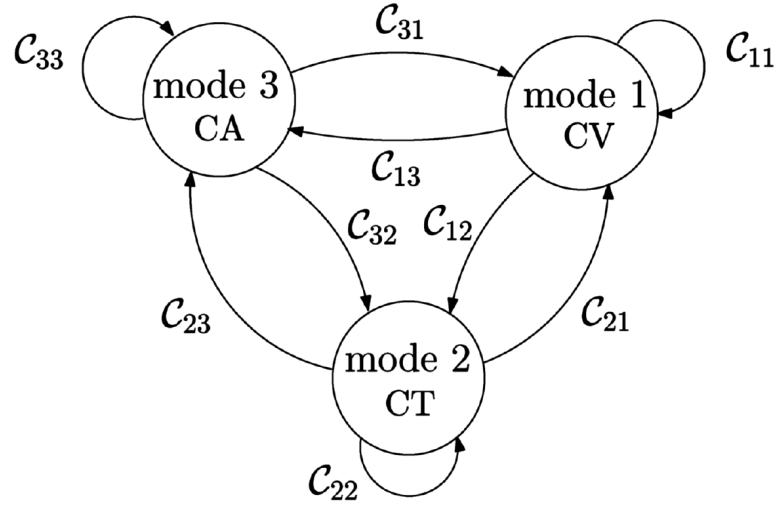


Figure 5.2. Flight mode transition [63]

We can consider the four different flight patterns of the UAS which are composed of the flight mode transitions between two modes in a basic set of discrete modes (CV, CT and CA): 1) left turn, 2) right turn, 3) acceleration, and 4) deceleration.

a. Left/Right Turn The left turn is corresponding to the mode transition between CV mode to CT mode. Figure 5.3 shows the UAS entering a left turn along the flight path bounded by the geofence and presents the parameters defining the guard conditions for each FMCP. The unit direction vector of the predefined flight path is

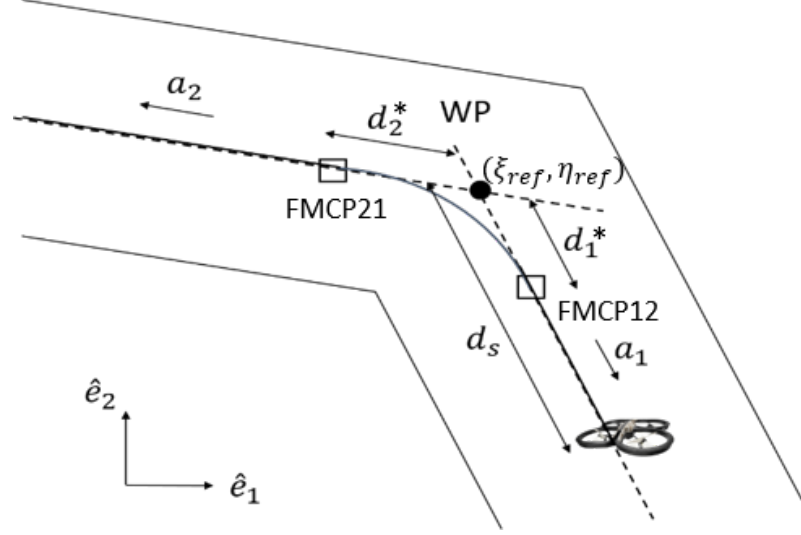


Figure 5.3. Left turn transition

denoted \mathbf{a}_i , the projected distance of the UAS from the waypoint along the dotted predefined flight path (the center line of geofence) is noted as d_s ; d_1^* and d_2^* are Gaussian random variables with a given mean and covariance, and they are defined as the distance from each actual FMCP at which the UAS initiates or finishes the left turn. The UAS remains in CT mode when the inequality condition ($d_s \leq d_i^*$) is satisfied, and the corresponding set of guard conditions for the left turn is given in Table 5.1. The set of guard conditions for the right turn can also be defined in the same way with that of the left turn.

Table 5.1. Set of guard conditions of left turn at FMCP_{ij}

$$C_{1i} = C_{2i} = C_{3i} = \neg C_{1j}$$

$$C_{1j} = C_{2j} = C_{3j} = L_{\mathbf{x}}\mathbf{x} + L_{\boldsymbol{\theta}}\boldsymbol{\theta} \leq \mathbf{0}$$

$$C_{13} = C_{23} = C_{33} = \emptyset$$

$$L_{\mathbf{x}} = [a_i^T \quad 0 \quad 0]$$

$$L_{\boldsymbol{\theta}} = -1$$

$$\boldsymbol{\theta}^* = d_1^* + \mathbf{a}_i^T [\zeta_{ref} \quad \eta_{ref}]$$

b. Acceleration/Deceleration For the acceleration/deceleration behavior of the UAS, the mode transition for the acceleration consists of CV mode and CA mode: the UAS in acceleration is likely to change the mode from CV mode to CA mode in the area near the waypoint. Fig. 5.4 shows the parameters of guard conditions for the FMCP₁₃, and the set of guard conditions is set up similarly to the left turn in Table 5.2. The set of guard conditions for the mode transition for deceleration can be defined similarly with that of the acceleration behavior.

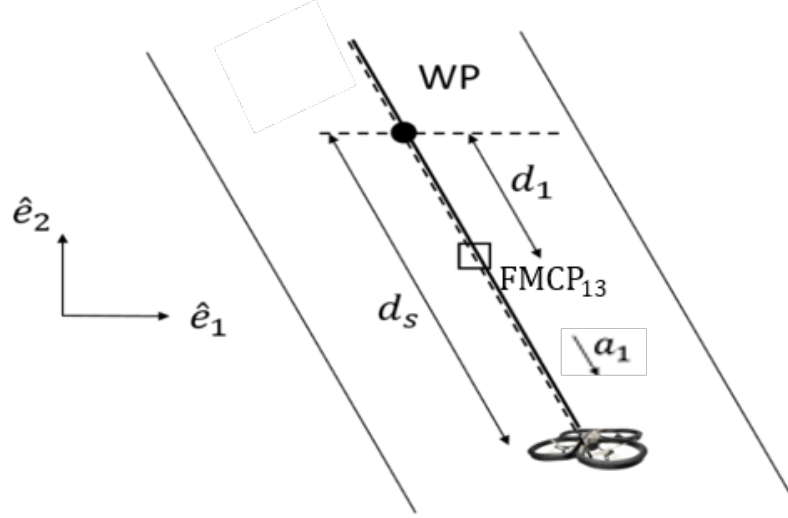


Figure 5.4. Mode transition of acceleration from CV (mode 1) to CA (mode 3)

Table 5.2. Set of guard conditions at FMCP₁₃

$C_{11} = C_{21} = C_{31} = \neg C_{13}$
$C_{13} = C_{23} = C_{33} = L_{\mathbf{x}}\mathbf{x} + L_{\boldsymbol{\theta}}\boldsymbol{\theta} \leq \mathbf{0}$
$C_{12} = C_{22} = C_{32} = \emptyset$
$L_{\mathbf{x}} = [a_1^T \ 0 \ 0]$
$L_{\boldsymbol{\theta}} = -1$
$\boldsymbol{\theta}^* = d_1^* + \mathbf{a}_i^T [\zeta_{ref} \ \eta_{ref}]$

Constraints on UAS Motion

As discussed, the assigned geofence provides constraints on the motion of a UAS in each flight mode, since the UAS should remain within the geofence provided by the UTM system. Also, we can access the performance data of the UAS and it can be used as constraints on the state of UAS during flight. From the geofence configuration, we can enforce the constraints on the direction of the velocity of the UAS to be aligned with the center line of geofence and/or on the position of the UAS to be within the boundary, which are represented in forms of an equality or an inequality condition on the state of the UAS. From the performance data, we can use the cruise and/or maximum speed as constraints in forms of an equality or an inequality condition of the velocity state of the UAS. The equality and inequality constraints are defined as:

$$\begin{aligned} D_{e_q(k)}(k)\mathbf{x}(k) &= \beta_{e_q(k)}(k) \\ D_{i_q(k)}(k)\mathbf{x}(k) &\leq \beta_{i_q(k)}(k) \end{aligned} \tag{5.8}$$

where $D_{e_q(k)}(k) \in \mathbb{R}^{h \times n}$, $D_{i_q(k)}(k) \in \mathbb{R}^{p \times n}$, $\beta_{e_q(k)}(k) \in \mathbb{R}^h$ and $\beta_{i_q(k)}(k) \in \mathbb{R}^p$ form the state constraints. In this research, we consider the velocity constraints for the expected direction of the UAS from the assigned geofence configuration and focus on the equality conditions. There are different constraints depending on the flight mode. When the UAS is flying along a straight line (e.g., CA and CV modes), the constraints are found by enforcing the velocity vector to be orthogonal to the unit normal vector of the segment \mathbf{b}_i of the flight path in Fig. 5.5, which is represented explicitly as

$$[0 \quad 0 \quad \mathbf{b}_i^T]\mathbf{x}(k) = 0 \tag{5.9}$$

The other case is the constraint for the turning mode. The center line of geofence is modeled as the circle that is tangent to the current segment and the next segment of flight path Fig. 5.5. The corresponding velocity constraints are obtained by requiring

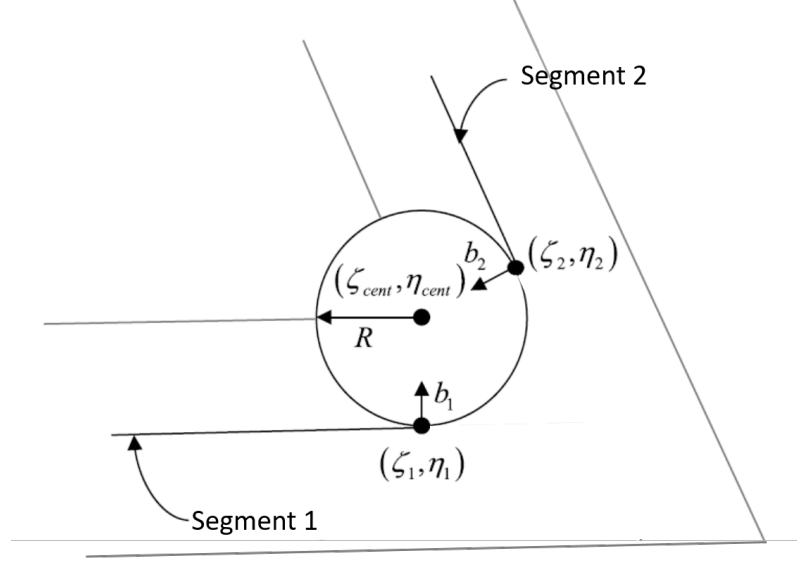


Figure 5.5. Circle between two segments

the velocity to be orthogonal to the gradient vector of the circle [64]. The equations of the circle and the gradient vector $\nabla \mathbf{h}$ are as follows:

$$\begin{aligned} h(\mathbf{x}(k)) &= (\zeta - \zeta_{cent})^2 + (\eta - \eta_{cent})^2 = R^2 \\ \nabla \mathbf{h}(\mathbf{x}(k)) &= [2(\zeta - \zeta_{cent}) \quad 2(\eta - \eta_{cent})] \end{aligned} \quad (5.10)$$

The equality constraint of the state of the UAS in the turning mode is defined as:

$$[0 \quad 0 \quad \nabla \mathbf{h}^T] \mathbf{x}(k) = 0 \quad (5.11)$$

This constraint information is very useful for accurate UAS tracking, as our UAS model can better represent the actual behavior of the UAS regulated by the UTM system with geofencing, and thus produce more accurate tracking performance. So, it is desirable to incorporate the constraint information into UAS tracking in a systematical way. To achieve this, we propose to use the constrained Kalman filtering technique that can efficiently handle various types of state constraints. The technical details of the constrained Kalman filter will be described when it is introduced as a building block in the proposed UAS tracking algorithm that will be discussed in the

following sections.

5.3 UAS Tracking Algorithm

To develop a hybrid estimation algorithm (i.e., UAS tracking algorithm) for the constrained stochastic linear hybrid system model, we use a bank of constrained Kalman filters, each of which is matched to the continuous dynamics with the state constraints in each mode. Let $\mathbf{Z}^k \equiv \{\mathbf{z}(1), \mathbf{z}(2), \dots, \mathbf{z}(k)\}$ denote a set of measurements up to time k . Let us assume that, from the last iteration at time $k - 1$, the mode probabilities $m^i(k - 1) \equiv p(q(k - 1) = i | \mathbf{Z}^{k-1})$, $i = 1, 2, \dots, r$, are computed and the mode-matched continuous state probability distribution functions (pdfs) are obtained Gaussian distributions as:

$$p(\mathbf{x}(k - 1) | q(k - 1) = i, \mathbf{Z}^{k-1}) = \mathcal{N}(\mathbf{x}(k - 1); \hat{\mathbf{x}}^i(k - 1), \mathbf{P}^i(k - 1)) \quad (5.12)$$

for $i = 1, 2, \dots, r$, where $\hat{\mathbf{x}}^i(k - 1)$ and $\mathbf{P}^i(k - 1)$ are the mean and covariance computed from the i th constrained Kalman filter at time $k - 1$. Then, using the new measurement $\mathbf{z}(k)$ generated at time k , we can compute $p(\mathbf{x}(k) | q(k) = i, \mathbf{Z}^k)$ and $m^i(k)$ for all the modes as shown in the following steps.

Step 1: Mixing

First, we compute the mixing probability $m^{i|j}(k)$ as:

$$\begin{aligned} m^{i|j}(k) &= p(q(k - 1) = i | q(k) = j, \mathbf{Z}^{k-1}) \\ &= \frac{p(q(k) = j | q(k - 1) = i, \mathbf{Z}^{k-1}) p(q(k - 1) = i | \mathbf{Z}^{k-1})}{p(q(k) = j | \mathbf{Z}^{k-1})} \\ &= \frac{\lambda_{ij}(k - 1) m^i(k - 1)}{\sum_{l=1}^r \lambda_{lj}(k - 1) m^l(k - 1)} \end{aligned} \quad (5.13)$$

where $\lambda_{ij}(k-1) := p(q(k) = j|q(k-1) = i, \mathbf{Z}^{k-1})$ is the mode transition probability and computed as:

$$\begin{aligned}
\lambda_{ij}(k-1) &= \int_{\mathbb{R}^n} p(q(k) = j|q(k-1) = i, \mathbf{x}(k-1) = \mathbf{x}, \mathbf{Z}^{k-1}) \\
&\quad \times p(\mathbf{x}(k-1) = \mathbf{x}|q(k-1) = i, \mathbf{Z}^{k-1}) d\mathbf{x} \\
&= \int_{\mathbb{R}^n} \Phi_l(L_{\mathbf{x},ij}\mathbf{x} + L_{\boldsymbol{\theta},ij}\bar{\boldsymbol{\theta}}, L_{\boldsymbol{\theta},ij}\Sigma_{\boldsymbol{\theta}}L_{\boldsymbol{\theta},ij}^T) \mathcal{N}(\mathbf{x}; \hat{\mathbf{x}}^i(k-1), \mathbf{P}^i(k-1)) d\mathbf{x} \\
&= \Phi_l(L_{\mathbf{x},ij}\hat{\mathbf{x}}^i(k-1) + L_{\boldsymbol{\theta},ij}\bar{\boldsymbol{\theta}}, L_{\boldsymbol{\theta},ij}\Sigma_{\boldsymbol{\theta}}L_{\boldsymbol{\theta},ij}^T + L_{\mathbf{x},ij}\mathbf{P}^i(k-1)L_{\mathbf{x},ij}^T)
\end{aligned} \tag{5.14}$$

where $\Phi_l(\mu, \Sigma)$ is the l -dimensional Gaussian cumulative density function for $\mathbf{y} \sim \mathcal{N}(\mu, \Sigma)$ defined as $\Phi_l(\mu, \Sigma) \equiv p(\mathbf{y} \leq \mathbf{0})$. Using the mixing probability, the initial conditions for the constrained Kalman filter matched to mode j are obtained as:

$$\begin{aligned}
\hat{\mathbf{x}}^{0j}(k-1) &= \Sigma_{i=1}^r m^{ij}(k) \hat{\mathbf{x}}^i(k-1) \\
\mathbf{P}^{0j}(k-1) &= \Sigma_{i=1}^r m^{ij}(k) \{ \mathbf{P}^i(k-1) \\
&\quad + [\hat{\mathbf{x}}^i(k-1) - \hat{\mathbf{x}}^{0j}(k-1)][\hat{\mathbf{x}}^i(k-1) - \hat{\mathbf{x}}^{0j}(k-1)]^T \}
\end{aligned} \tag{5.15}$$

Note that this mixing allows the exponentially growing computational complexity of hybrid estimation to be constant, thus enabling online application.

Step 2: Mode-Conditioned Estimation (Constrained Kalman Filtering)

Typically, the mode-matched estimation can be performed using the traditional Kalman filters. However, in this research, we need to incorporate the constraints imposed on the state of the UAS, and thus will apply the constrained Kalman filtering technique, where we incorporate the state constraints by restricting the prior distribution of the traditional Kalman filters. For mode j , the well-known Kalman filter has the following prediction equations:

$$\begin{aligned}
\hat{\mathbf{x}}^j(k|k-1) &= \mathbf{A}_j \hat{\mathbf{x}}^{0j}(k-1) \\
\mathbf{P}^j(k|k-1) &= \mathbf{A}_j \mathbf{P}^{0j}(k-1) \mathbf{A}_j^T + \mathbf{B}_j \mathbf{Q}_j \mathbf{B}_j^T
\end{aligned} \tag{5.16}$$

With the new measurement $\mathbf{z}(k)$, the prior mean and covariance are updated as:

$$\begin{aligned}\hat{\mathbf{x}}^j(k) &= \hat{\mathbf{x}}^j(k|k-1) + \mathbf{K}(k)(\mathbf{z}(k) - \mathbf{H}_j\hat{\mathbf{x}}^j(k|k-1)) \\ \mathbf{P}^j(k) &= (\mathbf{I} - \mathbf{K}(k)\mathbf{H}_j)\mathbf{P}^j(k|k-1)\end{aligned}\tag{5.17}$$

where

$$\mathbf{K}(k) = \mathbf{P}^j(k|k-1)\mathbf{H}_j^T(\mathbf{H}_j\mathbf{P}^j(k|k-1)\mathbf{H}_j^T + \mathbf{R}_j)^{-1}\tag{5.18}$$

We impose the constraints by projecting the unconstrained state estimates onto the constraint surface. If we consider only the equality constraints, this is identical to solving the following minimization problem.

$$\hat{\mathbf{x}}_c^j(k) = \arg \max_x \{[\mathbf{x} - \hat{\mathbf{x}}^j(k)]^T \mathbf{W}[\mathbf{x} - \hat{\mathbf{x}}^j(k)] : \mathbf{D}_j \mathbf{x} = \beta_j\}\tag{5.19}$$

where \mathbf{W} is a positive definite weighting matrix. The solution to this constrained optimization problem can be obtained using the Lagrange multiplier approach. Let $\Psi(\mathbf{x})$ be a scalar cost function defined as follows:

$$\Psi(\mathbf{x}) := [\mathbf{x} - \hat{\mathbf{x}}^j(k)]^T \mathbf{W}[\mathbf{x} - \hat{\mathbf{x}}^j(k)] + 2\lambda^T(\mathbf{D}_j \mathbf{x} - \beta_j)\tag{5.20}$$

where $\lambda \in \mathbb{R}^h$ is a vector of Lagrange multipliers. Taking the partial derivatives and setting them equal to zero yields

$$\begin{aligned}\frac{\partial \Psi}{\partial \mathbf{x}} &= 2(\mathbf{x} - \hat{\mathbf{x}}^j(k))^T \mathbf{W} + 2\lambda^T \mathbf{D}_j = 0 \\ \frac{\partial \Psi}{\partial \lambda} &= \mathbf{D}_j \mathbf{x} - \beta_j = 0\end{aligned}\tag{5.21}$$

After applying the constraints and rearranging, this yields

$$\hat{\mathbf{x}}_c^j(k) = \hat{\mathbf{x}}^j(k) - \mathbf{J}_j(\mathbf{D}_j \hat{\mathbf{x}}^j(k) - \beta_j)\tag{5.22}$$

where \mathbf{J}_j is defined as:

$$\mathbf{J}_j \equiv \mathbf{W}^{-1} \mathbf{D}_j^T (\mathbf{D}_j \mathbf{W}^{-1} \mathbf{D}_j^T)^{-1}\tag{5.23}$$

The corresponding prior covariance is given by

$$\mathbf{P}_c^j(k) = (\mathbf{I} - \mathbf{J}_j \mathbf{D}_j)(\mathbf{A}_j \mathbf{P}^j(k) \mathbf{A}_j^T + \mathbf{B}_j \mathbf{Q}^j \mathbf{B}_j^T)(\mathbf{I} - \mathbf{J}_j \mathbf{D}_j)^T\tag{5.24}$$

Step 3: Mode Probability Update

The mode probability which denotes how probable a mode is the true mode is updated using Bayes rule as:

$$\begin{aligned} m^j(k) &= p(q(k) = j | \mathbf{Z}^k) \\ &= \frac{1}{c} p(\mathbf{z}(k) | q(k) = j, \mathbf{Z}^{k-1}) p(q(k) = j | \mathbf{Z}^{k-1}) \end{aligned} \quad (5.25)$$

where c is a normalizing constant; $p(\mathbf{z}(k) | q(k) = j, \mathbf{Z}^{k-1})$ is the mode-conditioned likelihood function given by

$$p(\mathbf{z}(k) | q(k) = j, \mathbf{Z}^{k-1}) = \mathcal{N}(\mathbf{z}(k) - \mathbf{H}_j \hat{\mathbf{x}}^j(k|k-1); \mathbf{0}, \mathbf{S}_j(k)) \quad (5.26)$$

where

$$\mathbf{S}_j(k) = \mathbf{H}_j(k) \mathbf{P}^j(k|k-1) \mathbf{H}_j(k)^T + \mathbf{R}_j \quad (5.27)$$

and $p(q(k) = j | \mathbf{Z}^{k-1})$ is the prior mode probability computed as:

$$p(q(k) = j | \mathbf{Z}^{k-1}) = \sum_{i=1}^r \lambda_{ij}(k-1) m^i(k-1) \quad (5.28)$$

Step 4: Output

Using the mode-conditioned state estimates $\hat{\mathbf{x}}_c^j(k)$, the covariance $\mathbf{P}_c^j(k)$, and the mode probability $m^j(k)$ for $j = 1, 2, \dots, r$, the continuous state estimate $\hat{\mathbf{x}}(k)$ and its covariance $\mathbf{P}(k)$ are computed as:

$$\begin{aligned} \hat{\mathbf{x}}(k) &= \sum_{j=1}^r \hat{\mathbf{x}}_c^j(k) m^j(k) \\ \mathbf{P}(k) &= \sum_{j=1}^r \{ \mathbf{P}_c^j(k) + [\hat{\mathbf{x}}_c^j(k) - \hat{\mathbf{x}}(k)][\hat{\mathbf{x}}_c^j(k) - \hat{\mathbf{x}}(k)]^T \} m^j(k) \end{aligned} \quad (5.29)$$

The discrete state estimate $\hat{q}(k)$ is then computed as:

$$\hat{q}(k) = \underset{i}{\operatorname{argmax}} m^i(k) \quad (5.30)$$

The proposed algorithm is named as constrained state-dependent-transition hybrid estimation (CSDTHE) and summarized in Figure 5.6.

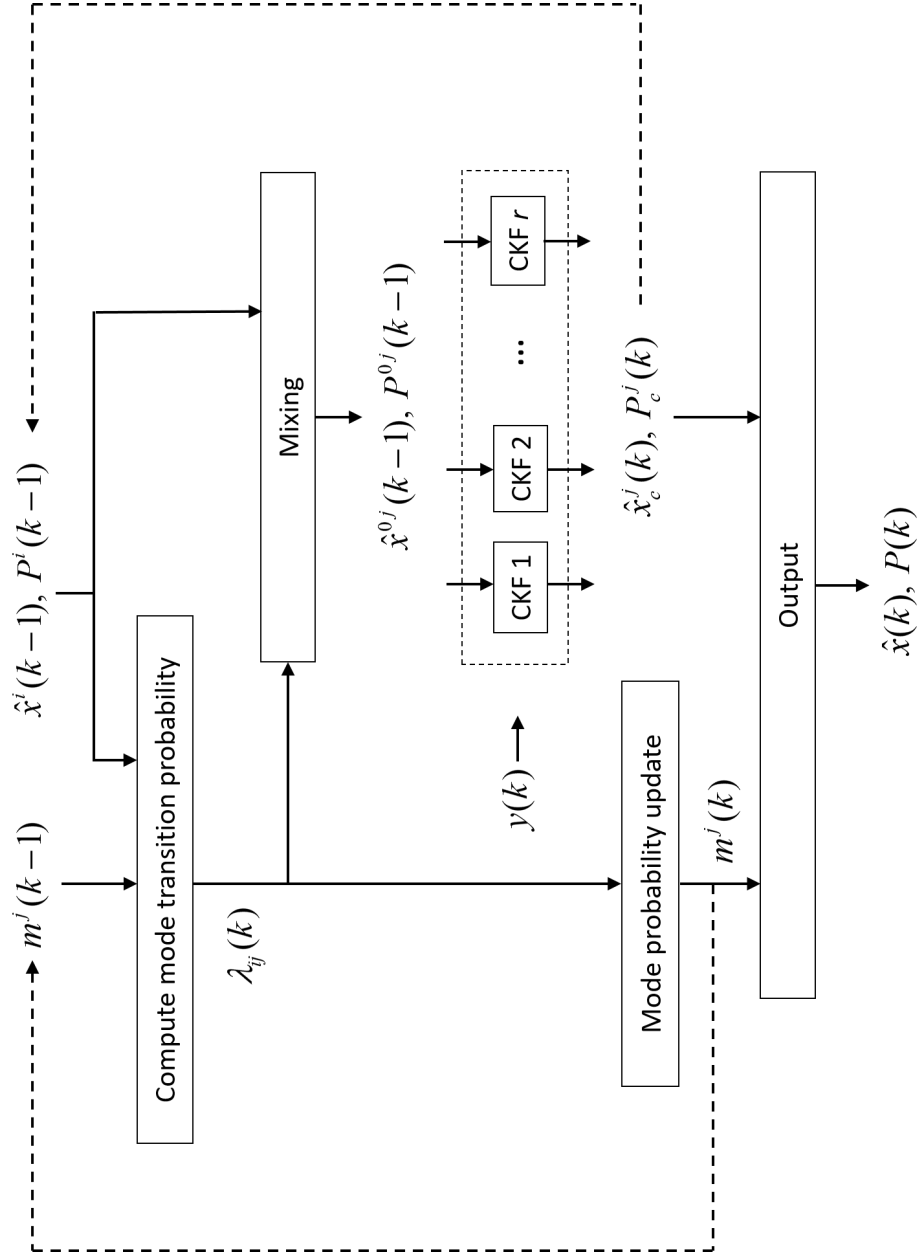


Figure 5.6. Structure of the proposed UAS tracking algorithm (CSDTHE)

5.4 Numerical Simulation

In this section, the proposed UAS tracking algorithm is demonstrated with an illustrative UTM example: the tracking of a UAS flying along the predefined flight plan in the urban area. Figure 5.7 shows a two-dimensional 1.2 *km* flight path of the UAS which consists of a series of waypoints (WP1, WP2, and WP3) from the origin to the FedEx office center in Indianapolis, Indiana, U.S. Here, we focus on a left turn example around the waypoint WP1 given in Fig. 5.8 to demonstrate the proposed UAS tracking algorithm. The UAS starts to flight toward WP1 at a constant velocity (about 10 *m/s*) from the origin, maintaining a constant altitude. Then, the UAS changes its flight mode from CV mode to CT mode at the stochastic FMCP₁ in the area near WP1 to go to the next waypoint WP2, and complete the turn at the next stochastic FMCP₂, which corresponds to the flight mode transition from CT mode to CV mode. To emulate the actual UAS trajectories, we generate 100 trajectories using the UAS models discussed in Section 5.2.2, which are randomly distributed about the center line of the geofence with 24 *m* width. These trajectories include uncertainties of the UAS flight due to navigation error and wind. The sampling time is chosen to be $T = 1$ *sec*, the nominal speed of UAS is 10 *m/s*, and its nominal turning rate during a coordinated turn ω is 0.2 *rad/s*. The process noise covariance for each mode,



Figure 5.7. Flight plan of the UAS [65]

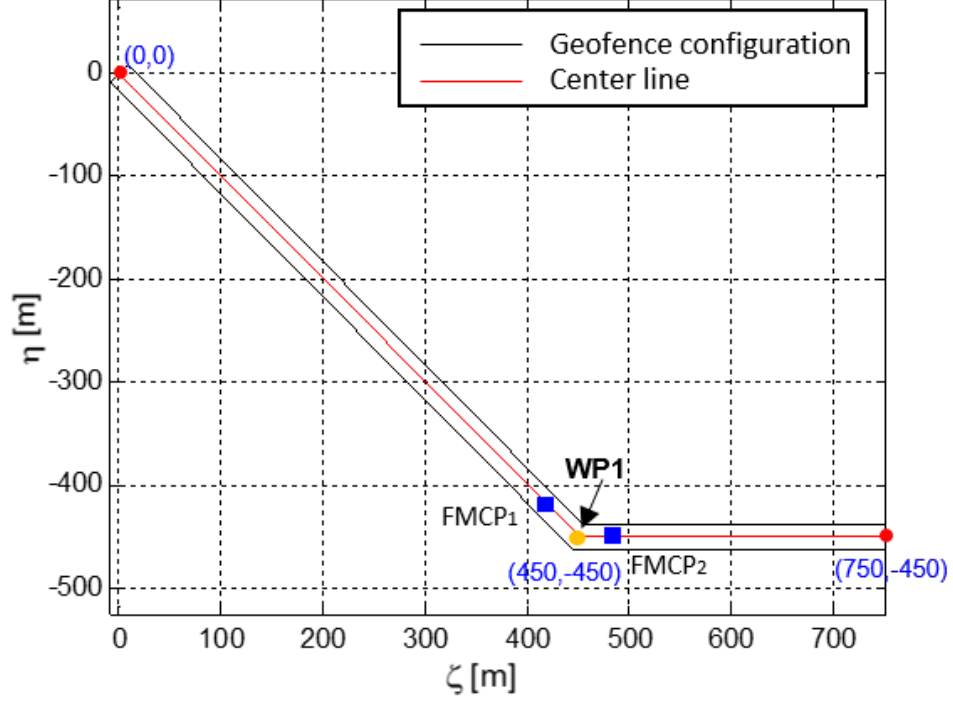


Figure 5.8. A left turn example around WP1

the measurement noise covariance, and initial condition $\mathbf{x}_0 \sim \mathcal{N}(\boldsymbol{\mu}_0, \mathbf{Q}_0)$ are set as follows.

$$\mathbf{Q}_{cv} = \begin{bmatrix} 0.003^2 & 0 \\ 0 & 0.003^2 \end{bmatrix}, \mathbf{Q}_{ct} = \begin{bmatrix} 0.001^2 & 0 \\ 0 & 0.001^2 \end{bmatrix}, \mathbf{R} = \begin{bmatrix} 2.5^2 & 0 \\ 0 & 2.5^2 \end{bmatrix} \quad (5.31)$$

$$\boldsymbol{\mu}_0 = \begin{bmatrix} 0 \text{ m} & 7 \text{ m/s} & 0 \text{ m} & -7 \text{ m/s} \end{bmatrix}^T \quad (5.32)$$

$$\mathbf{Q}_0 = \text{diag} \left(\begin{bmatrix} (1\text{m})^2 & (0.01\text{m/s})^2 & (1\text{m})^2 & (0.01\text{m/s})^2 \end{bmatrix} \right) \quad (5.33)$$

We compare the performance of the proposed CSDTHE algorithm with that of the IMM algorithm which has been shown to give an excellent performance in aircraft tracking applications and the SDTHE algorithm which can account for the state-dependent transition but no constraint information. Since IMM cannot incorporate

neither the state-dependent transition nor constraint information, we use the below constant matrix to model the discrete mode transition in the IMM algorithm.

$$\lambda_{ij} = \begin{bmatrix} 0.8 & 0.2 \\ 0.2 & 0.8 \end{bmatrix} \quad (5.34)$$

Figure 5.9 shows the result of a single run including the true trajectory of the UAS with the estimated trajectories obtained by each estimation method. In Fig. 5.10, the discrete mode estimation accuracies of the algorithms are compared. Figure 5.11 shows the RMS position and velocity estimation errors of each method, respectively. The other statistics of the estimation results are summarized in Table 5.3. It is obvious that the proposed CSDTHE algorithm produces more accurate estimates compared to both IMM and SDTHE. This is reasonable as the proposed CSDTHE algorithm can systematically incorporate both the state-dependent transition and state constraint

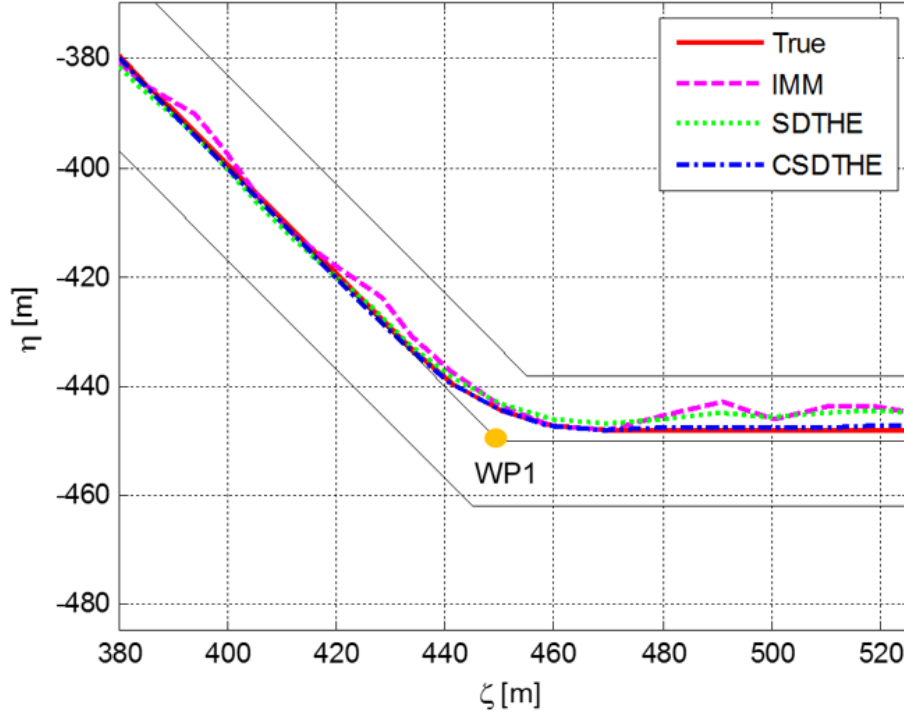


Figure 5.9. Actual and estimated trajectories of the UAS (a single run)

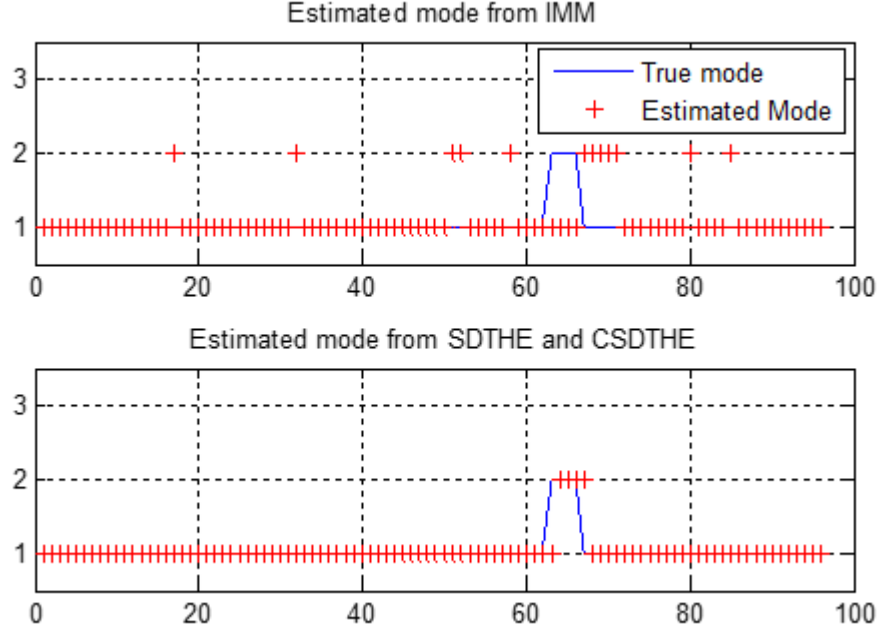


Figure 5.10. Comparison of mode-estimation accuracy (a single run)

Table 5.3. Comparison of estimation performance (100 Monte Carlo runs)

	RMS position error [m]	RMS velocity error [m/s]	Average mode estimation error (number of time steps)
IMM	2.22	1.31	9.56
SDTHE	1.12	0.10	2.00
CSDTHE	0.89	0.04	2.00

information, while the others cannot. These numerical simulation results demonstrate that the proposed CSDTHE algorithm can enhance the UAS tracking performance in the UTM with geofence, which is a crucial element of the UTM for highly dense UAS operations in the urban environment.

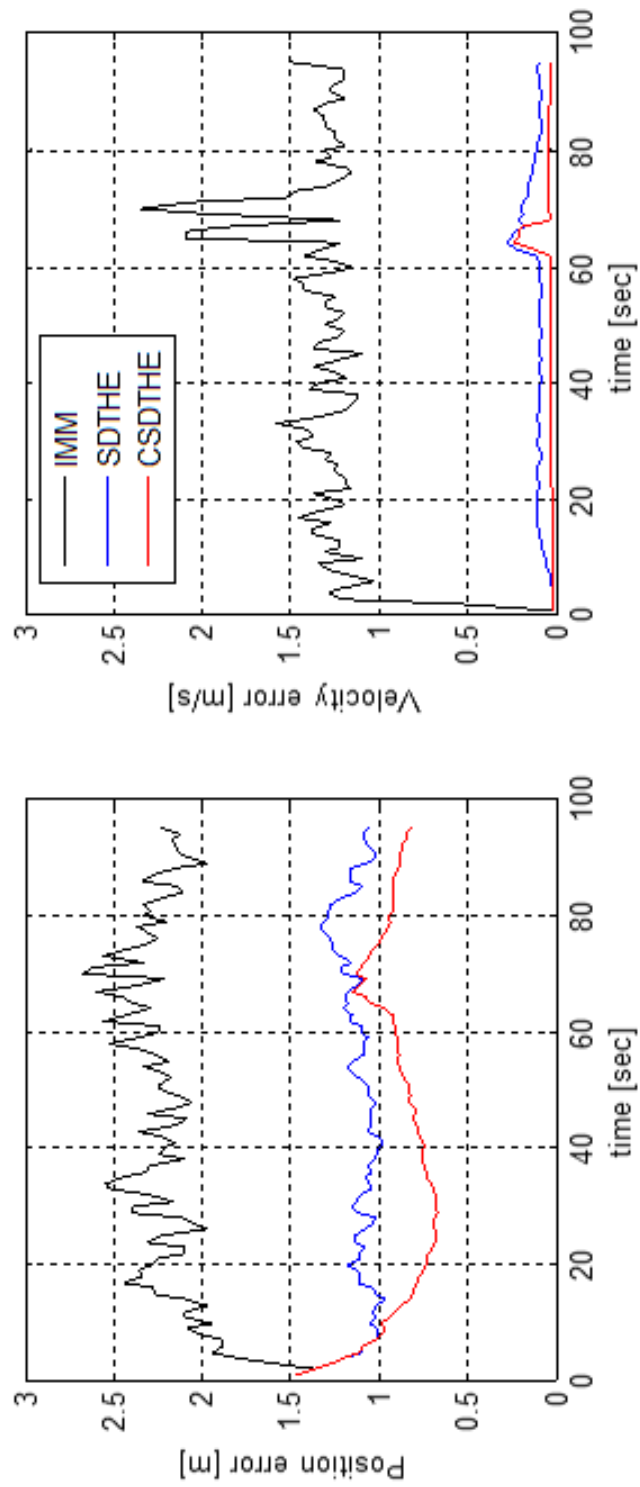


Figure 5.11. RMS position and velocity estimation errors with 100 Monte Carlo runs

6. SUMMARY

In this thesis, novel hybrid state estimation algorithms have been developed that can effectively solve the complex state estimation problems of the stochastic hybrid system. The developed hybrid state estimation algorithms have been applied to safety-critical applications in air traffic control systems: 1) aircraft tracking and estimated time of arrival prediction, and 2) unmanned aircraft system traffic management.

We have first developed new nonlinear state estimation algorithms for the jump Markov system based on Gaussian sum approximation which effectively deal with nonlinearities in the system dynamics and/or measurements. The proposed algorithms address the problem of the exponential growth in the number of the Gaussian mixture components through the Gaussian components reduction techniques developed in this research and successfully incorporate the Gaussian sum filters in the interacting multiple model algorithm framework. It has been demonstrated in the numerical simulation that the proposed algorithms improve the representation of the non-Gaussian posterior probability density function of the state by minimizing the loss of useful information and provide better state estimation performance with efficient computation over the existing multiple model based nonlinear state estimation algorithms for the jump Markov system.

The developed hybrid estimation algorithms have been extended to a more general class of stochastic hybrid systems consisting of nonlinear mode-matched continuous state dynamics with state-dependent mode transition. The Gaussian sum filter has been integrated to the proposed hybrid estimation framework as the mode-matched filter to cope with the nonlinearity of the continuous state dynamics. A closed-form expression of the state-dependent mode transition probability has been derived using the analytical properties of the posterior distribution represented by Gaussian mixtures. The derived mode transition probability has been used to compute the

mixing probability for merging of the hypotheses on the discrete state history. The effectiveness of the proposed hybrid estimation algorithm has been demonstrated using an illustrative hybrid state estimation problem.

We have investigated the flight phases and operational procedures of a typical flight, summarized and abstracted different operational procedures into flight modes, and derived a stochastic nonlinear hybrid model to describe the aircraft's behavior in the descent stage. Under the framework of the stochastic nonlinear hybrid model, an aircraft tracking and estimated time of arrival (ETA) prediction algorithm has been proposed based on the state-dependent-transition hybrid estimation algorithm which can account for continuous state-dependent flight mode transitions. Since the flight plan and procedure information can be effectively incorporated into the proposed algorithm, more accurate ETA prediction can be achieved. The performance of the proposed algorithm has been validated using two actual descending approach scenarios.

A new UAS tracking algorithm has been also proposed as a supporting tool for safe and efficient UAS traffic management (UTM) operation with geofence. The proposed algorithm can explicitly exploit useful information derived from the geofence, one of the key concepts of UTM, and effectively handle the complex behavior of a UAS with the stochastic linear hybrid system model. The improvement of the tracking accuracy obtained by the proposed algorithm has been demonstrated with an illustrative UTM example. The future work is to extend the current research to conformance monitoring of UAS operations, in which the position and velocity information estimated by the proposed algorithm can be used to check whether the actual trajectories of UAS satisfy the regulation imposed by the UTM system (e.g., planned waypoints, required time of arrival, assigned geofence, planned airspeed profile, etc.).

REFERENCES

REFERENCES

- [1] Concept of operations for the next generation air transportation system. Technical report, Joint Planning and Development Office, Washington, D. C., 2011.
- [2] J. Hu, J. Lygeros, and S. Sastry. Towards a theory of stochastic hybrid systems. In *Hybrid Systems: Computation and Control*, volume 1790 of the series Lecture Notes in Computer Science, pages 160–173. Springer Verlag, 2000.
- [3] J. Lee, S. Lee, and I. Hwang. Hybrid system modeling and estimation for estimated time of arrival prediction in terminal airspace. *Journal of Guidance, Control, and Dynamics*, 39(4):903–910, 2016.
- [4] C. E. Seah, A. Aligawesa, and I. Hwang. Algorithm for conformance monitoring in air traffic control. *Journal of Guidance, Control, and Dynamics*, 33(2):500–509, 2010.
- [5] S. Lee and I. Hwang. Event-based state estimation for stochastic hybrid systems. *IET Control Theory and Applications*, 9(13):1973–1981, 2015.
- [6] J. Hu, W.-C. Wu, and S. Sastry. Modeling subtilin production in bacillus subtilis using stochastic hybrid systems. In *Hybrid Systems: Computation and Control*, volume 2993 of the series Lecture Notes in Computer Science, pages 417–431. Springer Verlag, 2004.
- [7] S. Lee and I. Hwang. Intent inference-based flight-deck human-automation mode-confusion detection. *Journal of Aerospace Information Systems*, 12(8):503–518, 2015.
- [8] S. Lee, J. Lee, and I. Hwang. Maneuvering spacecraft tracking via state-dependent adaptive estimation. *Journal of Guidance, Control, and Dynamics*, 39(9):2034–2043, 2016.
- [9] S. McGinnity and G. W. Irwin. Multiple model bootstrap filter for maneuvering target tracking. *IEEE Transactions on Aerospace and Electronic Systems*, 36(3):1006–1012, 2000.
- [10] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan. *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley and Sons, 2004.
- [11] S. Lee and I. Hwang. Interacting multiple model estimation for spacecraft maneuver detection and characterization. In *Proceedings of AIAA Guidance, Navigation, and Control Conference*, Kissimmee, FL, January 2015.
- [12] H.A.P. Blom and Y. Bar-Shalom. The interacting multiple model algorithm for systems with Markovian switching coefficients. *IEEE Transactions on Automatic Control*, 33(8):780–783, 1988.

- [13] C. Gao and G. Duan. Fault diagnosis and fault tolerant control for nonlinear satellite attitude control systems. *Aerospace Science and Technology*, 33(1):9–15, 2014.
- [14] I. Hwang, C. E. Seah, and S. Lee. A study on stability of the interacting multiple model algorithm. *IEEE Transactions on Automatic Control*, 62(2):901–906, 2017.
- [15] R. Toledo-Moreo, M. A. Zamora-Izquierdo, B. Úbeda Miñarro, and A. F. Gómez-Skarmeta. High-integrity IMM-EKF-based road vehicle navigation with low-cost GPS/SBAS/INS. *IEEE Transactions on Intelligent Transportation Systems*, 8(3):491–511, 2007.
- [16] Y.-S. Kim and K.-S. Hong. An IMM algorithm for tracking maneuvering vehicles in an adaptive cruise control environment. *International Journal of Control, Automation, and Systems*, 2(3):310–318, 2004.
- [17] E. A. Wan and R. V. Merwe. The unscented kalman filter for nonlinear estimation. In *IEEE Proceedings of Adaptive Systems for Signal Processing, Communications, and Control Symposium*, Lake Louise, Alberta, Canada, August 2000.
- [18] N. J. Gordon, D. J. Salmond, and A. F. M. Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings F (Radar and Signal Processing)*, 140(2):107–113, 1993.
- [19] A. Doucet, N. de Freitas, and N. Gordon. *Sequential Monte Carlo Methods in Practice*. Springer, New York, NY, 2001.
- [20] Y. Boers and J. N. Driessen. Interacting multiple model particle filter. *IEE Proceedings of Radar Sonar Navigation*, 150(5):344–349, 2003.
- [21] H. Driessen and Y. Boers. Efficient particle filter for jump Markov nonlinear. *IEE Proceedings of Radar Sonar Navigation*, 152(5):323–326, 2005.
- [22] D. L. Alspach and H. W. Sorenson. Nonlinear Bayesian estimation using Gaussian sum approximation. *IEEE Transactions on Automation Control*, 17(4):439–448, 1972.
- [23] G. Terejanu, P. Singla, T. Singh, and P.D. Scott. Uncertainty propagation for nonlinear dynamic systems using Gaussian mixture models. *Journal of Guidance, Control, and Dynamics*, 31(6):1623–1633, 2008.
- [24] B. Vo and W. Ma. The Gaussian mixture probability hypothesis density filter. *IEEE Transactions on Signal Processing*, 54(11):4091–4104, 2006.
- [25] N. M. Kwok, G. Dissanayake, and Q. P. Ha. Bearing-only SLAM using a SPRT based Gaussian sum filter. In *IEEE Proceedings of International Conference on Robotics and Automation*, Barcelona, Spain, April 2005.
- [26] L. Wang, Y. Liang, X. Wang, and L. Xu. Gaussian sum filter of Markov jump nonlinear system. *IET Signal Processing*, 9(4):335–340, 2015.
- [27] G. Terejanu, P. Singla, T. Singh, and P.D. Scott. Adaptive Gaussian sum filter for nonlinear Bayesian estimation. *IEEE Transactions on Automation Control*, 56(9):2151–2156, 2011.

- [28] G. Terejanu, P. Singla, T. Singh, and P.D. Scott. A novel Gaussian sum filter method for accurate solution to the nonlinear filtering problem. In *11th International Conference on Information Fusion*, Cologne, Germany, September 2008.
- [29] D. J. Higham. An algorithmic introduction to numerical simulation of stochastic differential equations. *Society for Industrial and Applied Mathematics Review*, 43(3):525–546, 2001.
- [30] C. E. Seah and I. Hwang. State estimation for stochastic linear hybrid systems with continuous-state-dependent transitions: An IMM approach. *IEEE Transactions on Aerospace and Electronic Systems*, 45(1):376–392, 2009.
- [31] Y. Jung and G. Monroe. Development of surface management system integrated with CTAS arrival tool. In *Proceedings of AIAA Aviation, Technology, Integration, and Operations Conference*, Arlington, VA, September 2005.
- [32] Y. Bar-Shalom, X.-R. Li, and T. Kirubarajan. *Estimation with Applications to Tracking and Navigation*. John Wiley & Sons, 2001.
- [33] B. Musialek, C. F. Munafo, H. Ryan, and M. Paglione. Literature survey of trajectory predictor technology. Technical report, Federal Aviation Administration William J. Hughes Technical Center, 2010.
- [34] Q. M. Zheng and Y. J. Jhao. Probabilistic approach to trajectory conformance monitoring. *Journal of Guidance, Control, and Dynamics*, 35(6):1888–1898, 2012.
- [35] S. M. McGovern, S. B. Cohen, M. Truong, and G. Fairley. Kinematics-based model for stochastic simulation of aircraft operating in the national airspace system. In *Proceedings of IEEE/AIAA Digital Avionics Systems Conference*, Dallas, TX, December 2007.
- [36] I. Lympieropoulos and J. Lygeros. Improved multi-aircraft ground trajectory prediction for air traffic control. *Journal of Guidance, Control, and Dynamics*, 33(2):347–362, 2010.
- [37] U. Maeder, M. Morari, and T. I. Baumgartner. Trajectory prediction for light aircraft. *Journal of Guidance, Control, and Dynamics*, 34(4):1112–1119, 2011.
- [38] G. B. Chatterji. Short-term trajectory prediction methods. In *Proceedings of AIAA Guidance, Navigation, and Control Conference and Exhibit*, Portland, OR, August 1999.
- [39] W. Liu and I. Hwang. Probabilistic trajectory prediction and conflict detection. *Journal of Guidance, Control, and Dynamics*, 34(6):1779–1789, 2011.
- [40] J. L. Yepes, I. Hwang, and M. Rotea. New algorithms for aircraft intent inference and trajectory prediction. *Journal of Guidance, Control, and Dynamics*, 30(2):370–382, 2007.
- [41] J. Krozel and D. Andrisani. Intent inference with path prediction. *Journal of Guidance, Control, and Dynamics*, 29(2):225–236, 2006.
- [42] Y. Liu and X. R. Li. Intent based trajectory prediction by multiple model prediction and smoothing. In *Proceedings of AIAA Guidance, Navigation, and Control Conference*, Kissimmee, FL, January 2015.

- [43] R. A. Coppenbarger. En route climb trajectory prediction enhancement using airline flight-planning information. In *Proceedings of AIAA Guidance, Navigation, and Control Conference and Exhibit*, Portland, OR, August 1999.
- [44] D. P. Thipphavong, C. A. Schultz, A. G. Lee, and S. H. Chan. Adaptive algorithm to improve trajectory prediction accuracy of climbing aircraft. *Journal of Guidance, Control, and Dynamics*, 36(1):15–24, 2013.
- [45] Y. Zhao and S. Vaddi. Algorithms for FMS reference trajectory synthesis to support NextGen capability studies. In *Proceedings of AIAA Aviation, Technology, Integration, and Operations Conference*, Los Angeles, CA, August 2013.
- [46] M. Soler, A. Olivares, E. Saffetti, and D. Zapata. Framework for aircraft trajectory planning toward an efficient air traffic management. *Journal of Aircraft*, 49(1):341–348, 2012.
- [47] D. G. Hull. *Fundamentals of Airplane Flight Dynamics*. Springer, 2007.
- [48] J. P. Clarke, D. Bennett, K. Elmer, J. Firth, R. Hilb, N. Ho, S. Johnson, S. Lau, L. Ren, D. Senechal, and N. Sizov. Development, design and flight test evaluation of a continuous descent approach procedure for nighttime operation at Louisville International Airport. Technical report, Partnership for Air Transportation Noise and Emissions Reduction, 2006.
- [49] S. S. Mulgund and R. F. Stengel. Optimal nonlinear estimation for aircraft flight control in wind shear. *Automatica*, 32(1):3–13, 1996.
- [50] D. Rivas, A. Valenzuela, and J. L. de Augusto. Computation of global trajectories of commercial transport aircraft. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 227(1):142–158, 2013.
- [51] M. Norsell. Multistage trajectory optimization with radar-range constraints. *Journal of Aircraft*, 42(4):849–857, 2005.
- [52] X.-R. Li and Y. Bar-Shalom. Design of an interacting multiple model algorithm for air traffic control tracking. *IEEE Transactions on Control Systems Technology*, 1(3):186–194, 1993.
- [53] K. Roy, B. Levy, and C. J. Tomlin. Target tracking and estimated time of arrival (ETA) prediction for arrival aircraft. In *Proceedings of AIAA Guidance, Navigation, and Control Conference and Exhibit*, Keystone, CO, August 2006.
- [54] Q. Zheng, Y. J. Zhao, and B. Capozzi. Time-of-arrival taxi conformance monitoring for surface operations. *Journal of Guidance, Control, and Dynamics*, 34(3):750–760, 2011.
- [55] C. Gong and A. Sadosky. A final approach trajectory model for current operations. In *Proceedings of AIAA Aviation, Technology, Integration, and Operations Conference*, Fort Worth, TX, September 2010.
- [56] User manual for the base of aircraft data (BADA) revision 3.11. Eurocontrol Experimental Centre, 2013.

- [57] J. K. Klooster, K. D. Wichman, and O. F. Bleeker. 4D trajectory and time-of-arrival control to enable continuous descent arrivals. In *Proceedings of AIAA Guidance, Navigation, and Control Conference and Exhibit*, Honolulu, HI, August 2008.
- [58] P. Kopardekar et al. Unmanned aircraft system traffic management (UTM) concept of operations. In *Proceedings of AIAA Aviation, Technology, Integration, and Operations Conference*, Washington, D.C., June 2016.
- [59] G. Zhu and P. Wei. Low-altitude UAS traffic coordination with dynamic geofencing. In *Proceedings of AIAA Aviation, Technology, Integration, and Operations Conference*, Washington, D.C., June 2016.
- [60] X. R. Li and V. P. Jilkov. Survey of maneuvering target tracking. part V: Multiple-models methods. *IEEE Transactions on Aerospace and Electronic Systems*, 41(4):1255–1321, 2005.
- [61] D. Simon. Kalman filtering with state constraints: a survey of linear and non-linear algorithms. *IET Control Theory and Applications*, 4(8):1303–1318, 2010.
- [62] X. R. Li and V. P. Jilkov. Survey of maneuvering target tracking. part I: Dynamic models. *IEEE Transactions on Aerospace and Electronic Systems*, 39(4):1333–1364, 2003.
- [63] C. E. Seah and I. Hwang. Terminal-area aircraft tracking using hybrid estimation. *Journal of Guidance, Control, and Dynamics*, 32(3):836–849, 2009.
- [64] G. Mann and I. Hwang. Four-dimensional aircraft taxiway conformance monitoring with constrained stochastic linear hybrid systems. *AIAA Journal of Guidance, Control and Dynamics*, 35(5):1593–1604, 2012.
- [65] Google Map. <https://www.google.com/maps>.

VITA

VITA

Jooyoung Lee received a bachelor's degree in Aerospace Engineering from Inha University, Incheon, South Korea, in 2007 and a master's degree in Mechanical and Aerospace Engineering from Seoul National University, Seoul, South Korea, in 2010. Her research during the master's degree was specialized in decentralized fusion filter for state estimation of UAV. After completing the master's degree, she worked on mobile robot control as a research engineer at LG Electronics. She began to pursue a doctoral degree in school of Aeronautics and Astronautics at Purdue University in 2012. Her research is focused on stochastic hybrid systems modeling and estimation with applications to air traffic control.