Laboratório de Modelagem, Análise e Controle de Sistemas Não-Lineares
Departamento de Engenharia Eletrônica
Universidade Federal de Minas Gerais
Av. Antônio Carlos 6627, 31270-901 Belo Horizonte, MG Brasil
Fone: +55 3409-3470

State Estimation of Hybrid Dynamic Systems with Markov Jumps and State Constraints

Wendy Yadira Eras Herrera

Doctoral dissertation submitted to the Graduate Program in Electrical Engineering of the Federal University of Minas Gerais in partial fulfillment of the requirements for the degree of Doctor in Electrical Engineering.

Advisors: Prof. Bruno Otávio Soares Teixeira Dr.
Prof. Alexandre Rodrigues Mesquita Ph.D.

Belo Horizonte, February, 2018

Universidade Federal de Minas Gerais

Escola de Engenharia

Programa de Pós-Graduação em Engenharia Elétrica

STATE ESTIMATION OF HYBRID DYNAMIC SYSTEMS WITH MARKOV JUMPS AND STATE CONSTRAINTS

Wendy Yadira Eras Herrera

Tese de Doutorado submetida à Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em Engenharia Elétrica da Escola de Engenharia da Universidade Federal de Minas Gerais, como requisito para obtenção do Título de Doutor em Engenharia Elétrica.

Orientador: Prof. Bruno Otávio Soares Teixeira

Belo Horizonte - MG

Fevereiro de 2018

E65s

Eras Herrera, Wendy Yadira.
State estimation of hybrid dynamic systems with Markov jumps and state constraints [manuscrito] / Wendy Yadira Eras Herrera. - 2018.
xv, 115 f., enc. : il.

Orientador: Bruno Otávio Soares Teixeira.
Coorientador: Alexandre Rodrigues Mesquita.

Tese (doutorado) - Universidade Federal de Minas Gerais, Escola de Engenharia.

Bibliografia: f. 117-125.

1. Engenharia elétrica - Teses. 2. Markov, Processos de - Teses. 3. Processo estocástico - Teses. 4. Teoria da estimativa - Teses. I. Teixeira, Bruno Otávio Soares. II. Mesquita, Alexandre Rodrigues. III. Universidade Federal de Minas Gerais. Escola de Engenharia. IV. Título.

CDU: 621.3(043)

TESE DE DOUTORADO Nº 268

STATE ESTIMATION OF HYBRID DYNAMIC SYSTEMS WITH MARKOV JUMPS AND STATE CONSTRAINTS

Wendy Yadira Eras Herrera

DATA DA DEFESA: 28/02/2018

"State Estimation of Hybrid Dynamic Systems with Markov Jumps and State Constraints"

Wendy Yadira Eras Herrera

Tese de Doutorado submetida à Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em Engenharia Elétrica da Escola de Engenharia da Universidade Federal de Minas Gerais, como requisito para obtenção do grau de Doutor em Engenharia Elétrica.

Aprovada em 28 de fevereiro de 2018.

Por:

Prof. Dr. Bruno Otávio Soares Teixeira
DELT (UFMG) - Orientador

Prof. Dr. Alexandre Rodrigues Mesquita
DELT (UFMG) - Coorientador

Prof. Dr. Leonardo Antônio Borges Tôrres
DELT (UFMG)

Prof. Dr. Eduardo Mazoni Andrade Marçal Mendes
DELT (UFMG)

Prof. Dr. Geovany Araújo Borges
DEE (UnB)

Prof. Dr. Alessandro do Nascimento Vargas
Dep. Eletrotécnica (UTFPR)

Agradecimentos

A Deus, pelas oportunidades que surgem todos os dias na minha vida, pelas pessoas que colocou no meu caminho e que sem elas tudo seria muito mais difícil.

A minha família, pelo incansável apoio, compreensão, amor e incentivo. Vocês são o suporte primordial durante estes anos de trabalho.

Ao meu namorado Petrus, por todo o incentivo, apoio, carinho, paciência e dedicação durante toda esta longa caminhada.

Aos meus orientadores, professor Bruno Otávio, pela paciência, pela clareza, pela simplicidade e dedicação com que conduziu a orientação deste trabalho. Ao professor Alexandre, pela disposição, pela paciência, por dividir comigo seus conhecimentos. Vocês são parte fundamental neste trabalho.

Aos meus amigos do Laboratório de Modelagem, Análise e Controle de Sistemas Não-Lineares (MACSIN) pela colaboração durante a realização deste trabalho. Em especial, ao Victor, Dimas e Tales pela parceria e ao Marcus, Rodrigo, Mateus e a Ana e Anny pelo apoio nos momentos difíceis.

Aos meus amigos do Laboratório de *Mechatronics Control and Robotics* (MACRO) pelo auxilio computacional neste trabalho. Em especial, ao Ernesto, Daniel, Brenner e Juan José.

Aos meus amigos Rogério, Ludimila, Jullierme e Cristiane pelo suporte contínuo e pelo companheirismo dentro e fora da universidade.

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), FAPEMIG e CNPq pelo apoio financiero.

Resumo

Em algumas aplicações de estimação de estados, o sistema dinâmico pode ser representado por um modelo híbrido que é determinado pela interação entre estados analógicos e digitais (modos de operação). O problema de filtragem estocástica híbrida consiste em fornecer estimativas para ambos estados analógicos e digitais a partir de uma sequência de medições amostradas ruidosas e um modelo híbrido. Para esses sistemas, o filtro deve rastrear um número exponencialmente crescente de trajetórias possíveis, o que configura um desafio prático para resolver esse problema. Portanto, soluções aproximadas são comumente buscadas, procurando um compromisso entre precisão e tempo de processamento do filtro.

Neste trabalho, investigam-se duas questões relacionadas à estimatição de estados de sistemas híbridos. Primeiro, apresenta-se uma versão modificada do algoritmo de múltiplos modelos e múltiplas hipóteses (M^3H) para resolver de forma sub-ótima o problema de estimação de estado para sistemas não lineares com saltos Markovianos. Empregam-se métodos de redução de misturas Gaussianas como uma alternativa para a fusão de hipóteses do M^3H clássico. Portanto, informações de ambos os estados analógicos e digitais são empregadas para fundir as hipóteses, enquanto que apenas a informação do estado digital é empregada no método original. Como contribuição, a abordagem proposta fornece um mecanismo eficaz para que os usuários explorem o compromisso entre precisão e tempo de processamento do filtro. Os usuários estabelecem suas preferências definindo o número máximo de componentes da mistura. A sintonia desse parâmetro na abordagem proposta é mais eficiente do que escolher a profundidade de fusão no M^3H quando melhoria de precisão é requerida.

Em segundo lugar, considera-se o problema de estimação de estados para sistemas híbridos com restrições de igualdade nos estados. Investigam-se casos especiais desse problema para ambos sistemas lineares e não-lineares. Categoriza-se tal problema em três grupos de acordo com a restrição de igualdade linear ou não linear, bem como com a dependência da restrição no modo de operação. Para o caso de restrições de igualdade independentes do modo, apresentam-se as condições necessárias na inicialização e dinâmica para o clássico algoritmo de múltiplos modelos interativos (IMM) para produzir estimativas de estado satisfazendo uma restrição de igualdade linear para sistemas lineares. No entanto, para sistemas lineares e não-lineares, as restrições de igualdade dependentes do modo devem ser reforçadas ao longo do tempo. Apresenta-se uma versão modificada do filtro IMM para impor a restrição de igualdade nas estimativas de estado.

Abstract

In some state estimation applications, the dynamic system can be represented by a hybrid model that is determined by the interaction between analog and digital (mode) states. The hybrid stochastic filtering problem is to provide estimates for both analog and digital states from a sequence of noisy sample measurements and such hybrid model. For these systems, the filter should track an exponentially increasing number of possible trajectories, posing a practical challenge to solve this problem. Therefore, approximate solutions are often pursued, trading off the filter precision for processing time.

In this work, we investigate two issues related to the state estimation of hybrid systems. First, we present a modified version of the multiple models and multiple hypotheses (M^3H) algorithm to suboptimally solve the problem of state estimation for Markov jump nonlinear systems. We employ Gaussian mixture reduction methods as an alternative for the merging of hypotheses of the original M^3H. Thus, information from both the analog and digital states are employed to merge the hypotheses, while only information from the digital state is employed in the original M^3H method. As a contribution, the proposed approach provides an effective mechanism for users to explore the tradeoff between filter precision and processing time. Users set their preferences by defining the maximum number of mixture components. Setting this number in our proposed approach is more efficient than choosing the merging depth in M^3H when increased precision is demanded.

Second, we consider the problem of state estimation for hybrid systems with state equality constraints. We investigate special cases of this problem for both linear and nonlinear systems. We categorize such a problem into three groups according to the linear or nonlinear equality constraint as well as to the dependence of the constraint on the operating mode. For the mode-independent equality constraints case, we present the necessary conditions on the initialization and dynamics for the classical interacting multiple models (IMM) algorithm to yield state estimates satisfying a linear equality constraint for linear systems. However, for linear and nonlinear systems, the mode-dependent equality constraints must be enforced along time. We present a modified version of the IMM filter to enforce the equality constraint on the state estimates.

Contents

List of Figures

List of Tables

List of Symbols

$\mathbb{N}$	nonnegative integers;
$\mathbb{R}$	real numbers;
$\in$	is an element of;
$\triangleq$	equals by definition;
A^T	transpose of A;
A^{-1}	inverse of A;
$\text{diag}(A)$	diagonal of A;
$\mathcal{N}(A)$	null space of A;
$\mathcal{P}_{\mathcal{N}(A)}$	projector with range $\mathcal{N}(A)$;
δ	Dirac-delta function at x;
$\sin(x)$, $\cos(x)$, $\tan(x)$	sine, cossine, and tangent of x;
$\rho(x \mid y)$	conditional probability density function of x given y;
$\rho(x \mid m, y)$	probability density function of x conditional on the mode m given y;
$\rho(m \mid y)$	conditional probability mass function of x given m;
$\mathbb{E}[\cdot]$	mathematical expectation;
k	discrete-time index;
T	sampling interval;

f	dynamic or process nonlinear model in state space;
h	observation nonlinear model in state space;
g	nonlinear equality state constraint function;
A_{k-1}, B_{k-1}, G_{k-1}, C_k	matrices of the linear model in state space;
D_{k-1} and d_{k-1}	matrix and vector of the linear equality state constraint;
x_k	analog state vector;
m_k	digital state;
u_{k-1}	input vector;
w_{k-1}	process noise vector;
y_k	measurements or output vector;
v_k	measurement noise vector;
$\tilde{x}_k$	augmented state vector;
$J(x)$	cost function of x;
$\hat{x}_k$	state estimate;
$\hat{x}^s_{k-1}$	sth *a priori* state estimate of the filter bank;
$\hat{x}^s_k$	sth *a posteriori* state estimate of the filter bank;
$\hat{y}_k$	output estimate;
$\hat{x}^{\mathrm{c}}_k$	constrained state estimate;
$\hat{x}^{\mathrm{p}}_k$	projected state estimate;
$\hat{m}_k$	mode estimate;
w^s_{k-1}, $\tilde{p}^s_k$	sth *a priori* probability of modes;
γ^s_k, p^s_k	sth *a posteriori* probability of modes;
Π	transition probability matrix;
η^s_k	sth innovation of the filter bank;
$\rho_s(\eta^s_k)$	likelihood function of innovation;
P^{xx}_k	error covariance matrix;
P^{yy}_k	innovation covariance matrix;
P^{xy}_k	cross-covariance matrix;
$P^{xx\mathrm{c}}_k$	constrained error covariance matrix;
$P^{xx\mathrm{p}}_k$	projected error covariance matrix;

$P_k^{\tilde{x}\tilde{x}}$	augmented error covariance matrix;
Q_{k-1}	process noise covariance matrix;
R_k	measurement noise covariance matrix;
K_k	Kalman gain;
$\mathcal{X}_{j,k-1}$	jth column of the sigma-point matrix $\mathcal{X}_{k-1}$;
UT	unscented transform function.

List of Acronyms

CIMM Equality-constrained interacting multiple models;

EKF Extended Kalman filter;

KF Kalman filter;

IMM Interacting multiple model;

IMM_0 Interacting multiple model with projected initial condition;

ISE Integral squared error;

MAP Maximum *a posteriori*;

MM Multiple model;

M^3H Multiple model multiple hypothesis;

M^3HR Multiple model multiple hypothesis with Gaussian mixture reduction;

MCMC Markov chain Monte Carlo;

MHMF Multiple-hypothesis mixing filter;

MJS Markov jump system;

MMSE Minimum mean-square error;

PDF Probability density function;

PF Particle filter;

PMF Probability mass function;

GRMC Gaussian mixture reduction via clustering;

TPM Transition probability matrix;

UKF Unscented Kalman filter;

Chapter 1

Introduction

1.1 Motivation and Overview of the Field

In recent years, research on the state estimation problem has increased considerably due to numerous applications in areas such as engineering, computer science, geophysics, economics, biology, among others. Engineering problems that can be solved using state estimation include vehicle tracking [Fortmann, T., Bar-Shalom, Y., Scheffe, M., 1983], aircraft navigation systems [Nordlund, T., Gustafsson, F., 2001], fault detection and isolation [He, X., Wang, Z., Liu. Y., Zhou, D. H., 2013] and air traffic control systems [Lymperopoulos, I., Lygeros, J., 2009]. Bayesian methods may be employed as possible solutions to these problems and the most important algorithms are based on the Kalman filter.

The Kalman filter (KF) has been used in the literature for state estimation for linear and Gaussian systems [Kalman, 1960], whereas, for nonlinear systems, Gaussian approximation methods based on KF are generally used such as the extended Kalman filter (EKF) [Jazwinski, A. H., 1970; Maybeck, P. S., 1979], the unscented Kalman filter (UKF) [Julier, S. J., Uhlmann, J. K., Durrant-Whyte, H. F., 2000], as well as particle filtering methods (PF) [Arulampalam, M. S., Maskell, S., Gordon, N., Clapp, T., 2002].

In many state estimation applications, dynamical systems are described only by analog states, for example, the estimation of aircraft position, speed or attitude [Bach R., 1991; Blackman, S., Popoli, R., 1999]. In other cases, a dynamic system can be represented by a hybrid dynamic model whose behavior is determined by the interaction between the analog state x_k and the digital state m_k. This type of model is known as a hybrid system [Rong Li, X., 1996; Boers, Y., Driessen, H, 2000; Hwang, I. , Balakrishnan H., Tomlin, C., 2006; Goebel, R., Sanfelice, R. G., Teel, A. R., 2009]. In the literature, some authors refer to the analog state as the continuous state and the digital state as the discrete state or operating mode.

In the literature it is relatively common to find research on control [Choi, H. H., 2010; Liu, K., Yao, Y., Sun, D., Balakrishnan, V., 2012], identification [Juloski, A. Lj., Weiland, S., Heemels, W. P. M. H., 2005; Tian, Y., Floquet, T., Belkoura, L.,Perruquetti, W., 2011] and state estimation of hybrid systems [Sigalov, D., Leiter, N., Kalish, N., Oshman, Y., 2012; Suzdaleva, E., Nagy, I., 2011]. Over the past few years, a number of engineering applications was modeled as hybrid systems including, for instance, target tracking [Li, X. R., 2000; Hallouzi, R., Verhaegen, M., Kanev, S., 2009; Zhang, L., Pan, Q., Chen, T., 2010], fault detection and isolation [Hofbaur, M. W., Williams, B. C. , 2004], navigation systems using global navigation satellite systems [Liu, W., Hwang, I., 2012] and air traffic control systems [Antsaklis, P. J., 2000], among others.

In order to verify the relevance of the state estimation problem for hybrid systems in the world scenario, we present a research in the *Web of Science* platform to analyze the number of publications that address this problem. Figure 1.1 shows that the research on state estimation of hybrid systems has attracted large interest in recent years. In the national scenario, it is possible to mention some applications, such as aircraft tracking using a radar in air traffic control systems [Santana, P. H. R. Q. A., Menegaz, H. M., Borges G. A., Ishihara, J. Y., 2010] and the estimation of the downhole pressure of gas-lifted oil wells [Teixeira, B.O. S., Barbosa, B. H. G., Gomes, L. P., Teixeira, A. F., Aguirre, L. A., 2012].

The goal of the hybrid state estimation problem is to provide estimates for both

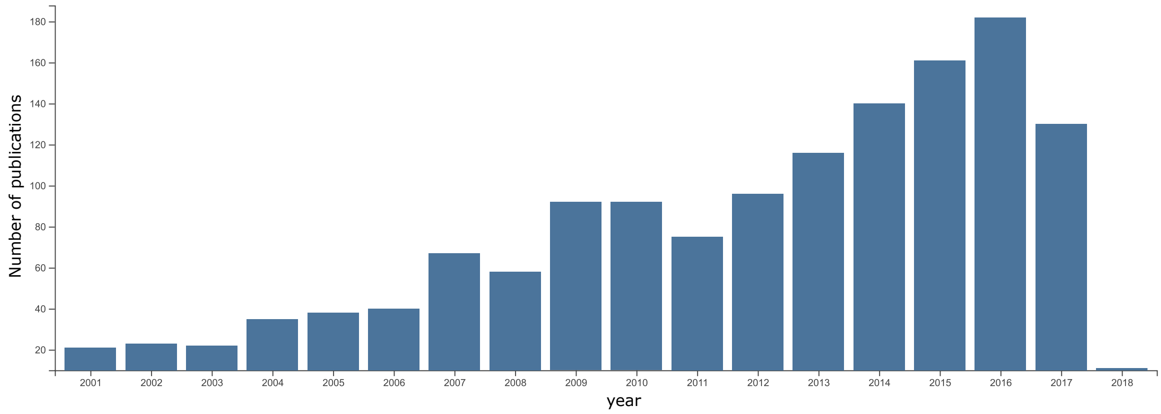

Figure 1.1: Number of publications with the topic *state estimation hybrid systems* searched on the Web of Science platform. Date: 12/01/2018. This research was refined considering the period from 2001-2018 and selecting the following research areas: *engineering, computer science* and *automation control systems*.

the analog states and the digital states from a sequence of noisy sampled measurements, $y_{1:k} \triangleq \{y_1, \ldots, y_k\}$, and a hybrid stochastic dynamic model. The solution to this problem may be obtained through the joint *a posteriori* probability density function (PDF), $\rho\left(x_k, m_k | y_{1:k}\right)$, of the analog and digital states. The main hurdle for solving such a problem is that both the number of possible sequences of modes and the number of possible analog trajectories (and, consequently, the computational cost) grows exponentially over time. This exponential growth in the number trajectories poses a challenge in solving the hybrid state estimation problem. Approximate methods circumvent the exponential growth problem by managing the number of multiple hypotheses via different approaches. For example, trajectories may be merged when they are similar and discarded when they are unlikely. Next, we present an example of the problem of exponential growth of the number of hypotheses over time.

Example 1. Consider, for example, the time evolution of a dynamical system with a digital state (mode) that can assume $M = 3$ possible values: white, black or gray, as illustrated in Figure 1.2. Assume that the initial condition of the operating mode is known at the time $k = 0$. At time $k = 1$, we have three possible operating modes ($M^k = 3^1$). If we want to keep track of the system trajectory, since we do not know which mode is active at $k = 1$, the possible modes give rise to another set of three modes. Consequently, at time $k = 2$, we have nine possible operating modes and the corresponding analog trajectories. At time $k = 3$, we have twenty-seven system

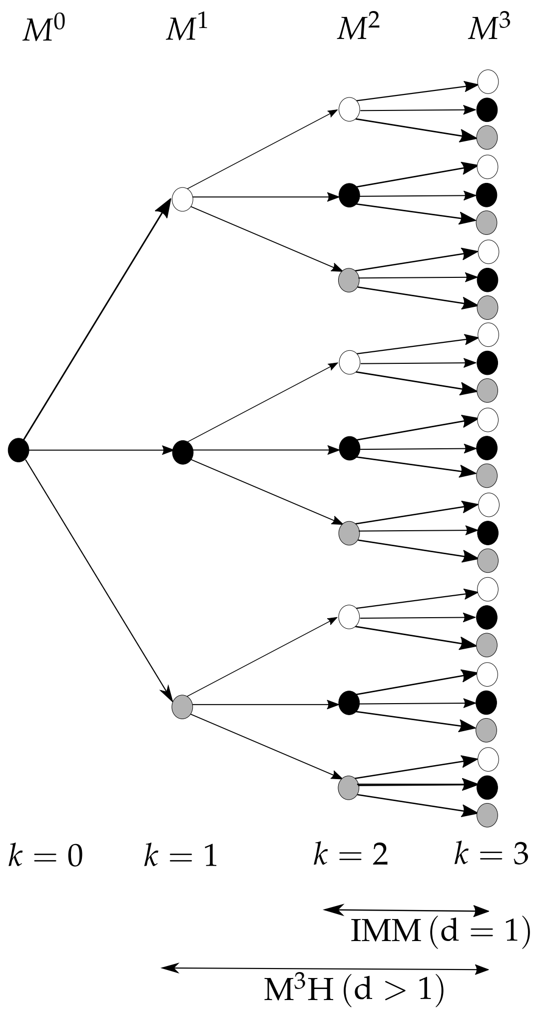

Figure 1.2: Exponential growth of the possible discrete trajectories of a stochastic hybrid system with three possible modes: white, black or gray. In the state estimation process, the IMM algorithm only employs information from the previous time while the M^3H algorithm employs a limited history of d steps behind.

trajectories. □

Existing methods for state estimation in hybrid systems are based on the multiple-model (MM) or Monte Carlo approaches. The estimator of MMs assumes that the dynamic system can be characterized by a set of M models that capture the possible operating modes of the system [Bar-Shalom, Y., Challa, S.,Blom, H. A., 2005]. The estimate provided by the MM estimator is achieved by running M filters (one for each mode) in parallel and combining their estimates [Hofbaur, M. W., Williams, B. C. , 2004]. Alternatively, particle filtering (PF) methods approximate the joint *a posteriori* PDF of the hybrid system using sampled trajectories [Boers, Y., Driessen, H, 2000;

Doucet, A., Gordon, N., Krishnamurthy, V., 2001; Koutsoukos, L., Williams, B., 2003]. The PF employs a set of particles with corresponding weights to provide estimates for both analog and digital states. The Table 1.1 shows how the hybrid stochastic filtering problem can be classified considering the type of dynamic model used to represent the hybrid system, or the approaches employed to estimate the analog and digital states and mode transitions of the system.

Several techniques based on MM approaches have been used to estimate the states of hybrid systems such as the generalized pseudo-Bayesian (GPB) algorithm [Ackerson, G. A., Fu, K. S., 1970], the detection and estimation method [Tugnait, J., 1982], the residual correlation Kalman filter bank approach [Hanlon, P., Maybeck, P., 2000] and the interacting multiple models (IMM) algorithm [Blom, H. A.P., Bar-Shalom, Y., 1988; Mazor, E., Averbuch, A., Bar-Shalom, Y., Dayan, J., 1998]. These methods control multiple hybrid estimates over time and require the running of a filter for each mode. In particular, in order to address the exponential growth problem, the IMM algorithm mixes the previous analog state estimates to generate new initial values for each filter in the next time step. Other methods of adaptive MM estimator have been proposed to reduce the number of hypotheses, so that they adapt the set of modes to a subset of modes that are more likely to occur in a given scenario [Li, X. R., Bar-Shalom, Y., 1996; Li, X. R., Zhi, X., Zhang, Y., 1999; Li, X. R., 2000]. In a more elaborate fashion, the multiple models and multiple hypotheses filter (M^3H) [Driessen, H., Boers, Y., 2001; Boers, Y., Driessen, H, 2004] keeps a limited buffer of hypotheses by means of merging and discarding hypotheses. In the M^3H filter, merging occurs when two or more hypotheses share the same sequence of modes for the previous d time steps. In addition, hypotheses are discarded whenever their likelihood is below a given threshold. The IMM algorithm can be seen as a special case of the M^3H algorithm with merging depth $d = 1$ [Driessen, H., Boers, Y., 2001]; see Figure 1.2.

Table 1.1: Research of the ten main bibliographical references that address the problem of hybrid stochastic filtering. We present the approaches employed to obtain estimates for both analog states, $\hat{x}_k$, and the digital states, $\hat{m}_k$, in each investigation. In these investigations, the Kalman filter bank (KFB), the particle filter (PF) or the search algorithm (SA) are used to estimate the analog state, whereas the Bayes' theorem (BT) or the particle filter are employed to estimate the digital state. Furthermore, we show whether the transitions between operating modes, $m_{k-1} \rightarrow m_k$, depend on the analog state (AS) or only on the digital state (DS) of the system. Citations indicate the number of citations for each investigation.

Reference	Citations	$\hat{x}_k$			$\hat{m}_k$		$m_{k-1} \rightarrow m_k$	
		KFB	PF	SA	BT	PF	AS	DS
Problems described in discrete time								
Mazor, E., Averbuch, A., Bar-Shalom, Y., Dayan, J. [1998]	1186	⋆			⋆			⋆
Li, X. R., Bar-Shalom, Y. [1996]	660	⋆			⋆			⋆
Hofbaur, M. W., Williams, B. C. [2004]	157			⋆	⋆		⋆	
Hwang, I. , Balakrishnan H., Tomlin, C. [2006]	77	⋆			⋆			⋆
Bar-Shalom, Y., Challa, S.,Blom, H. A. [2005]	76	⋆			⋆			⋆
Seah, C. E., Hwang, I. [2009]	70	⋆			⋆		⋆	
Blackmore, L., Williams, B. [2008]	59	⋆			⋆			⋆
Doucet, A., Ristic, B. [2002]	36		⋆		⋆			⋆
Koutsoukos, L., Williams, B. [2003]	35		⋆			⋆		⋆
Boers, Y., Driessen, H [2004]	32	⋆			⋆			⋆

In the context of hybrid stochastic filtering, it is necessary to describe in which way occurs the transitions between the different digital dynamics. In the MM approach, we consider the static and dynamic case for the transitions between the operating modes of the system. In the first case, we assume that a dynamic system is composed of various operating modes, but without a transition between them, while, in the second case, we present a dynamic system with transitions between the operating modes [Ristic, B., Arulampalam, S., Gordon, N., 2004; Hofbaur, M. W., 2005]. Thus, the dynamic MM approach considers that a dynamic system can be represented by several analog dynamics and that a certain switching logic chooses a given dynamics. Such systems are called as switched systems [Liberzon, D., 2003; Margaliot, M., 2006; Goebel, R., Sanfelice, R. G., Teel, A. R., 2009]. In the case of the dynamic MM approach, we consider that the transitions between the operating modes of the system depend on the analog state variable and/or the digital state variable. For example, in the investigations performed by [Seah, C. E., Hwang, I., 2009; Hwang, I. , Balakrishnan H., Tomlin, C., 2006; Benazera, E., Travé-Massuyès, L., 2009], the transitions between the modes depend on the analog state of the system, whereas, in the studies developed by [Mazor, E., Averbuch, A., Bar-Shalom, Y., Dayan, J., 1998; Blom, H. A.P., Bar-Shalom, Y., 1988; Boers, Y., Driessen, H, 2004], the transitions between the modes depend only on the digital state variable. The latter approach is known as Markovian jump systems (MJSs) [Costa, O. L., Guerra, S., 2002].

In this work, we are interested in investigating the problem of state estimation of hybrid systems, proposing approximate solutions to solve practical problems. This research topic has been receiving increasing attention for a variety of applications in the last years. In this sense, preliminary studies performed in the author's Master dissertation allowed the possibility of applying hybrid stochastic filtering algorithms to the problem of detection of the potential related to the imagination of the movement [Eras, W. Y., Erazo-Costa, F. J., Tierra-Criollo, C. J., Teixeira, B. O., 2012; Eras-Herrera, W. Y., 2012]. In that work, we employed an interactive bank based on the IMM estimator with two Kalman filters (IBKF) in parallel.

In the next subsections, we justify the investigation of two problems in the field of state estimation of hybrid systems. First, we investigate an alternative method for the merging step of the M^3H algorithm to suboptimally solve the problem of state estimation for Markov jump nonlinear systems. The proposed approach incorporates the information from the analog state in performing the merging step, rather than using only information from the digital states as in the M^3H filter. Second, we address the problem of state estimation for hybrid systems with equality constraints on the analog states. In the literature, few works address the constrained state estimation for the hybrid case. In this context, some issues are little explored in these investigations.

1.1.1 M^3H with Gaussian Mixture Reduction

The hybrid stochastic filtering problem is to provide estimates for both analog and digital states from a sequence of noisy sample measurements and such hybrid model. For these systems, the filter should track an exponentially increasing number of possible trajectories, posing a practical challenge to solve this problem. Therefore, approximate solutions are often pursued, trading off the filter precision for processing time. In particular, to address the exponential growth problem, the M^3H algorithm merges the hypotheses with the same sequence of modes in the last d steps. Thus, only information from the digital state is employed to merge the hypotheses.

In the context of Markov state transitions, the mode sequence provides no useful information to the task of predicting future states given that the current mode is known. This suggests that one should merge hypotheses based on the current state (both analog and digital) rather than based on the sequence of digital states. In the particular case of Markov jump linear systems, where dynamics is linear and mode transition is Markovian and independent of analog states [Costa, O.L.V., Fragoso, M.D., Marques, R.P., 2006], the exact posterior probability is given by a Gaussian mixture. This suggests that available techniques for reduction of Gaussian mixtures may be well suited to perform the merging operation.

The Gaussian mixture reduction by clustering (GMRC) approach optimizes the parameters of the reduced mixture according to the *integral quadratic distance* (ISD) criterion [Schieferdecker, D., Huber, M. F., 2009] and, to our knowledge, it is the best performing Gaussian mixture reduction method in approximation terms. Thus, we investigate the use of Gaussian reduction methods as an alternative for the merging step of the M^3H algorithm.

1.1.2 Constrained State Estimation

The Kalman filter provides optimal state estimates for linear and Gaussian systems. However, additional information about the system in the form of state constraints may be useful in improving the state estimates [Simon, D., 2010]. Various engineering applications regard dynamic systems satisfying certain constraints that arise from physical laws, mathematical properties or geometric considerations. For instance, the tracking of a ground vehicle in which the vehicle performs a constant velocity motion with a fixed heading determined by the physical road the vehicle is on [Simon, D. J., 2006; Ko, S., Bitmead, R.R., 2007; Teixeira, B. O. S., Chandrasekar, J., Palanthandalam-Madapusi, H. J., Tôrres, L. A. B., Aguirre, L. A., Bernstein, D. S., 2008; Chao-Yang, J., Yong-An, Z., 2013].

The problem of state estimation for non-hybrid linear and nonlinear constrained systems has been widely discussed in the literature. As examples consider the scenarios in which the dynamics and the disturbances are such that the state vector of the system satisfies an inequality constraint [Rao, C. V., Rawlings, J. B., Lee, J. H., 2001] or an equality constraint [Teixeira, B. O. S., Chandrasekar, J., Tôrres, L. A. B., Aguirre, L. A., Bernstein, D. S., 2009]. The nonlinear Kalman filtering algorithms [Teixeira, B. O. S., Tôrres, L. A. B., Aguirre, L. A., Bernstein, D. S., 2010; Kottakki, K. K., Bhushan, M., Bhartiya, S., 2014] and the particle filtering (PF) methods [Shao, X., Huang, B., Lee, J. M., 2010; Ebinger, B., Bouaynaya, N., Polikar, R., Shterenberg, R., 2015] provide approximate solutions for constrained state estimation.

For the hybrid systems case, few works address the problem of constrained state estimation in the literature. For instance, recent works of [Mann, G., Hwang, I., 2013; Kwon, C., Hwang, I., 2016] addresses this problem for linear hybrid systems. In this work, we investigate the problem of state estimation for hybrid systems with state equality constraints.

1.2 Problem Statement

The discrete-time hybrid stochastic systems in this dissertation follow the general dynamic model given by

$$x_k = f_{m_k}(x_{k-1}, u_{k-1}, w_{k-1}, k-1), \tag{1.1}$$

$$\pi_{s|r} = \Pr\{m_k = s | m_{k-1} = r\}, \tag{1.2}$$

$$y_k = h_{m_k}(x_k, v_k, k), \tag{1.3}$$

where $x_k \in \mathbb{R}^n$ is the analog state vector and $m_k \in \mathbb{M} \triangleq \{1, \ldots, M\}$ is the digital state, where M is the number of operating modes of the system. The inputs are given by $u_k \in \mathbb{R}^p$ and the measurement vector is given by $y_k \in \mathbb{R}^m$. The functions $f_{m_k} : \mathbb{R}^n \times \mathbb{R}^p \times \mathbb{R}^q \times \mathbb{N} \to \mathbb{R}^n$ and $h_{m_k} : \mathbb{R}^n \times \mathbb{R}^r \times \mathbb{N} \to \mathbb{R}^m$ are, respectively, the time-varying process and observation models with relation to the mode m_k. All M pairs of models (1.1) and (1.3) are assumed to be known. It is assumed that the process noise $w_k \in \mathbb{R}^q$ and the measurement noise $v_k \in \mathbb{R}^r$ are mutually independent, white random vectors with zero mean and with known covariance matrices Q_{m_k} and R_{m_k} respectively. It is assumed that the transition probability matrix (TPM) $\Pi \in \mathbb{R}^{M \times M}$, whose elements are given by $\pi_{s|r}$, is known.

The hybrid stochastic filtering problem seeks to provide state estimates $\hat{x}_k$ and $\hat{m}_k$ given by meaningful statistics (such as the mean or the mode) from the joint *a posteriori* probability density function (PDF) of x_k and m_k given a sequence of noisy measurements, $y_{1:k} \triangleq \{y_1, \ldots, y_k\}$, see Figure 1.3. The basis to the solution of this problem lies in the following decomposition

$$\rho(x_k, m_k \mid y_{1:k}) = \sum_{\text{all } m_{1:k}} \rho(x_k \mid m_{1:k}, y_{1:k}) \rho(m_{1:k} \mid y_{1:k}) \ , \tag{1.4}$$

where $\rho\left(x_k \mid m_{1:k}, y_{1:k}\right)$ is the *a posteriori* PDF of x_k conditional on the mode sequence $m_{1:k}$ and $\rho\left(m_{1:k} \mid y_{1:k}\right)$ is the conditional probability mass function (PMF) of the mode. Here the first term on the right-hand side of (1.4) may be computed by a classical nonlinear filter given that the mode sequence is known. The second term is computed applying Bayes' rule to the result of the nonlinear filter. A key property that makes this computation efficient is that mode transitions do not depend on the analog states.

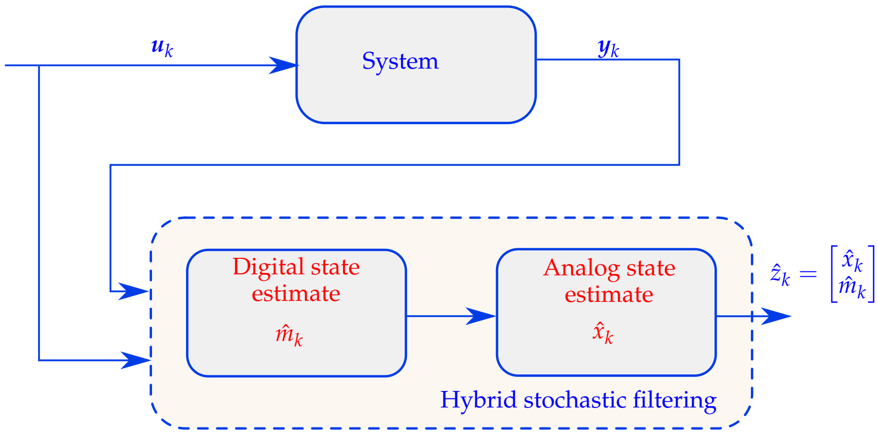

Figure 1.3: The hybrid stochastic filtering problem involves two steps: analog $\hat{x}_k$ and digital $\hat{m}_k$ state estimates. The hybrid state vector estimate can be represented by $\hat{z}_k \triangleq \begin{bmatrix} \hat{x}_k^T & \hat{m}_k^T \end{bmatrix}^T$. Adapted from [Hofbaur, M. W., 2005].

Otherwise, one would have to integrate over the prior state distribution when applying Bayes' rule.

In addition, we consider a special case of the hybrid stochastic filtering mentioned above. The problem of state estimation for hybrid systems with state equality constraints seeks to provide analog state vector x_k that satisfy the equality constraint

$$g_{m_k}(x_k, k) = d_{m_k}. \tag{1.5}$$

where the function $g_{m_k} : \mathbb{R}^n \times \mathbb{N} \to \mathbb{R}^s$, and $d_{m_k} \in \mathbb{R}^s$, is assumed to be known.

1.3 Research Objectives

This work aims at investigating the problem of state estimation of hybrid systems, proposing approximate solutions that attend the demands of practical problems. In order to address the general research goal, the following specific objectives are identified:

1. To study the main methods of state estimation of hybrid systems presented in Section 2.3. Compare suboptimal state estimation algorithms based on the multiple-model and Monte Carlo approaches for hybrid systems.

2. To investigate the use of Gaussian reduction methods to solve the exponential growth problem in the number of hypotheses of the M^3H algorithm. We also want to explore the trade off between filter precision and processing time.

3. To investigate state estimation algorithms for hybrid systems with state equality constraints. We want to discuss the special cases of hybrid constrained stochastic filtering for both linear and nonlinear systems and categorize into groups according to the linear or nonlinear equality constraints as well as to the dependence of the constraints on the operating mode.

1.4 Thesis Outline

This thesis is organized in five chapters and one appendix described as follows. Chapter 1 presents the introduction to the research topic, describing the motivation and justification of this study, formulating the problem, describing the general research goal and specific objectives and enumerating the contributions of this thesis.

Chapter 2 provides a literature review on non-hybrid stochastic filtering methods and approximate methods for state estimation in hybrid systems. First, we present the state estimation methods for non-hybrid systems. KF is presented as the optimal solution for linear and Gaussian systems. For the nonlinear systems, UKF and PF are reviewed. In addition, we present constrained state estimation methods for linear and nonlinear systems. Second, we discuss an approximate solution to the hybrid stochastic filtering problem, in the particular case of Markov jump systems. Furthermore, we review several techniques based on multiple models approaches such as the IMM and M^3H algorithms, and methods based on particle filtering.

The next two chapters present the contributions of this thesis. Chapter 3 presents a modified version of the M^3H algorithm. We use Gaussian reduction methods as an alternative for the merging step of the M^3H algorithm. We present a simulated example to illustrate the application of the proposed approach. Next, Chapter 4 considers the problem of state estimation for hybrid systems with state equality constraints. We investigate three special cases of practical interest of this problem. We present a modified version of the IMM filter to enforce the equality constraint on the state estimates. We present three simulated examples illustrate the application of the proposed method.

Finally, concluding remarks and suggestions of continuity are discussed in Chapter 5.

1.5 Contributions of this Work

This thesis presents two contributions in the field of state estimation of hybrid systems. First, we employ an alternative method for the merging step of the M^3H algorithm. Second, we investigate special cases of state estimation for hybrid systems with state equality constraints. We summarize each contribution and cite the corresponding thesis chapters as follows:

1. M^3H with Gaussian mixture reduction (Chapter 3):

   - We present a modified version of the multiple models and multiple hypotheses (M^3H) algorithm to suboptimally solve the problem of state estimation for Markov jump nonlinear systems. We employ Gaussian reduction methods as an alternative for the merging step of the M^3H algorithm. In our method, information from both the analog, $\hat{x}_k$, and digital, $\hat{m}_k$, estimates at time k are employed to define a metric to merge and eliminate hypotheses. This comes as an extension of the original M^3H, that only uses the information of the mode sequence, $\hat{m}_{k-d:k}$, from time $k-d$ to time k to merge the hypotheses.
   - We provide an effective mechanism for users to explore the tradeoff between filter precision and processing time. Users may set their preferences by defining the maximum number of mixture components N_{m}. Likewise, a similar tradeoff may be observed for M^3H by manipulating the merging depth d. A target tracking problem is used to illustrate the application of this algorithm.

  - Publication related to this contribution: [Eras-Herrera, W. Y., Mesquita, A. R., Teixeira, B. O. S., 2017].

2. Constrained state estimation (Chapter 4):

  - We address the problem of state estimation for hybrid systems with state equality constraints. We investigate special cases of this problem for both linear and nonlinear systems. We categorize such a problem into three groups according to the linear or nonlinear equality constraint as well as to the dependence of the constraint on the operating mode m_k.
  - We present two suboptimal algorithms for equality constrained state estimation for hybrid systems. For the mode-independent equality constraints case, we present the necessary conditions on the initialization and dynamics for the classical interacting multiple models (IMM) algorithm to yield state estimates satisfying a linear equality constraint for linear systems. However, for linear and nonlinear systems, the mode-dependent equality constraints must be enforced along time. We present a modified version of the IMM filter to enforce the equality constraint on the state estimates. We compare these algorithms by means of two examples, namely, a water tank system in which the sum of the levels of the two tanks is constrained so that mass is conserved, and the tracking of a ground vehicle in which the vehicle performs a constant velocity motion with a fixed heading.
  - Manuscript to be submitted [Eras-Herrera, W. Y., Mesquita, A. R., Teixeira, B. O. S., 2018].

Chapter 2

A Review on Non-Hybrid and Hybrid Stochastic Filtering Methods

In this chapter, we review the theoretical and technical bases for the state estimation in non-hybrid and hybrid systems. First, we present the state estimation methods for non-hybrid systems and we review the constrained state estimation methods for non-hybrid systems in Section 2.1. Then, we discuss the need to find an approximate solution to the state estimation problem of hybrid systems and we review the recursive solution for hybrid systems under the structure of Markov jump systems in Section 2.2. Finally, in Section 2.3, we present several hybrid stochastic filtering methods. We categorize this suboptimal solution into two groups, namely, multiple-model approaches and particle filtering methods.

2.1 Non-Hybrid Stochastic Filtering Methods

2.1.1 Recursive Bayesian Approach

Consider the discrete-time non-hybrid linear system given by

$$x_k = f(x_{k-1}, u_{k-1}, w_{k-1}, k-1), \tag{2.1}$$

$$y_k = h(x_k, v_k, k-1), \tag{2.2}$$

where $x_k \in \mathbb{R}^n$ is the state vector, $u_{k-1} \in \mathbb{R}^p$ is the input vector and $y_k \in \mathbb{R}^m$ is the measurement vector. The functions $f : \mathbb{R}^n \times \mathbb{R}^p \times \mathbb{R}^q \times \mathbb{N} \to \mathbb{R}^n$ and $h : \mathbb{R}^n \times \mathbb{R}^r \times \mathbb{N} \to \mathbb{R}^m$ are, respectively, the process and observation models. These models (2.1) and (2.2) are assumed to be known. It is assumed that the process noise $w_k \in \mathbb{R}^q$ and the measurement noise $v_k \in \mathbb{R}^r$ are mutually independent, white random vectors with zero mean and with known covariance matrices Q_k and R_k respectively.

The state estimation problem consists of finding a recursive estimate $\hat{x}_k$ for the state x_k of a dynamic system from a set of measurements $y_{1:k} \triangleq \{y_1, \ldots, y_k\}$ and a dynamic model. The recursive Bayesian approach estimates the *a posteriori* probability density function $\rho(x_k|y_{1:k})$ recursively along time using a set of measurements $y_{1:k}$ and a dynamic model. This PDF incorporates all the statistical information characterizing a complete solution to the state estimation problem. We employ the recursive Bayesian approach to obtain the *a posteriori* PDF $\rho(x_k|y_{1:k})$ of the state vector. The *a posteriori* PDF can be obtained in two steps: the *forescast* step and the *data-assimilation* step [Bar-Shalom Y., Li X. R., Kirubarajan T., 2001; Simon, D. J., 2006; Candy, J.V., 2009]. These steps are described below.

In the *forescast* step is obtained the *a priori* PDF $p(x_k|y_{1:k-1})$ of the vector of states x_k using the Chapman-Kolmogorov equation given by

$$\rho(x_k|y_{1:k-1}) = \int \rho(x_k|x_{k-1})\rho(x_{k-1}|y_{1:k-1})dx_{k-1}, \tag{2.3}$$

where $\rho(x_{k-1}|y_{1:k-1})$ is the a priori PDF in time $k-1$, $\rho(x_k|x_{k-1})$ is the PDF of the state transition obtained using the process model (2.1).

In the *data-assimilation* step, the information y_k is incorporated to update the *a priori* PDF $\rho(x_k|y_{1:k-1})$ using Bayes' theorem. This PDF is given by

$$\rho(x_k|y_{1:k}) = \frac{\rho(y_k|x_k)\rho(x_k|y_{1:k-1})}{\rho(y_k|y_{1:k-1})}, \tag{2.4}$$

where $p(y_k|x_k)$ is the likelihood funtion of innovations, $p(x_k|y_{1:k-1})$ is the *a priori* PDF in time k obtained in the *forescast* step and $p(y_k|y_{1:k-1})$ is a normalization constant.

The *a posteriori* PDF $p(x_k|y_{1:k})$ obtained using the equations (2.3) and (2.4) provides an optimal state estimate following some optimality criterion, for instance, the minimum mean-square-error (MMSE) state estimator or the maximum *a posteriori* (MAP) estimator, respectively, described by

$$\hat{x}_k^{MMSE} \triangleq \mathbb{E}\left\{x_k|y_{1:k}\right\}, \tag{2.5}$$

$$\hat{x}_k^{MAP} \triangleq \arg\max_{x_k} \quad \rho(x_k|y_{1:k}). \tag{2.6}$$

In this section, we present some stochastic filtering techniques described in the literature. The Kalman filter algorithm is presented as an optimal solution for the state estimation problem of linear and Gaussian systems [Kalman, 1960]. However, particle filtering methods [Arulampalam, M. S., Maskell, S., Gordon, N., Clapp, T., 2002] and Gaussian approximation methods based on the KF such as the unscented Kalman filter [Julier, S. J., Uhlmann, J. K., Durrant-Whyte, H. F., 2000] can be used as suboptimal algorithms for the state estimation problem of nonlinear systems. These algorithms are described below.

2.1.2 State Estimation for Linear Systems

Consider as a special case of (2.1)-(2.2) the time-varying discrete-time linear system given by

$$x_k = A_{k-1}x_{k-1} + B_{k-1}u_{k-1} + w_{k-1}, \tag{2.7}$$

$$y_k = C_k x_k + v_k, \tag{2.8}$$

where $A_{k-1} \in \mathbb{R}^{n\times n}$, $B_{k-1} \in \mathbb{R}^{n\times p}$ and $C_k \in \mathbb{R}^{m\times n}$ are assumed to be known.

The Kalman filter

The Kalman filter (KF) is the optimal recursive state estimator for such linear and Gaussian system [Bar-Shalom Y., Li X. R., Kirubarajan T., 2001]. This algorithm makes two assumptions: Gaussianity and linearity. The first assumption states that the *a posteriori* PDF can be represented by a Gaussian, $p(x_k|y_{1:k}) = \mathcal{N}\left(\hat{x}_k, P_k^{xx}\right)$, being completely characterized by two parameters, the mean and covariance [Kalman, 1960], whereas the second assumption states that the process and observation models, described in the equations (2.1) e (2.2), are known linear functions and w_{k-1} and v_k are assumed to be white, Gaussian, zero-mean, and mutually independent with known covariance matrices Q_k and R_k , respectively [Ho, Y.C., Lee, R.C.K., 1964]. The KF comprises two steps: the *forescast* step and the *data-assimilation* step. The KF algorithm is described as follows.

Algorithm 2.1.1. *The Kalman filter [Kalman, 1960]*
Initialize the filter with the state estimate $\hat{x}_0$ *and the corresponding covariance matrix* $P_0^{xx} = E[(x_0 - \hat{x}_0)(x_0 - \hat{x}_0)^T]$.

1. *Forescast step. Obtain the a priori state estimate* $\hat{x}_{k|k-1}$ *and the covariance matrix* $P_{k|k-1}^{xx}$ *from the information* $\hat{x}_{k-1}$ *and* P_{k-1}^{xx}. *This step is given by*

$$\hat{x}_{k|k-1} = A_{k-1}\hat{x}_{k-1} + B_{k-1}u_{k-1}, \tag{2.9}$$

$$P_{k|k-1}^{xx} = A_{k-1}P_{k-1}^{xx}A_{k-1}^T + Q_{k-1}, \tag{2.10}$$

$$\hat{y}_k = C_k\hat{x}_{k|k-1}, \tag{2.11}$$

$$P_k^{yy} = C_kP_{k|k-1}^{xx}C_k^T + R_k, \tag{2.12}$$

$$P_k^{xy} = P_{k|k-1}^{xx}C_k^T, \tag{2.13}$$

where P_k^{yy} *is the innovation covariance matrix and* P_k^{xy} *is the cross covariance matrix. These matrices are given by* $P_{k|k-1}^{xx} \triangleq E[(x_k - \hat{x}_{k|k-1})(x_k - \hat{x}_{k|k-1})^T]$, $P_k^{yy} \triangleq E[(y_k - \hat{y}_k)(y_k - \hat{y}_k)^T]$ *and* $P_k^{xy} \triangleq E[(x_k - \hat{x}_{k|k-1})(y_k - \hat{y}_k)^T]$.

2. *Data-assimilation step. Incorporate the information of a new measurement* y_k*, and obtain*

the a posteriori state estimate $\hat{x}_k$ and the covariance matrix P_k^{xx}. This step is given by

$$K_k = P_k^{xy}(P_k^{yy})^{-1}, \quad (2.14)$$

$$\hat{x}_k = \hat{x}_{k|k-1} + K_k(y_k - \hat{y}_k), \quad (2.15)$$

$$P_k^{xx} = P_{k|k-1}^{xx} - K_k P_k^{yy} K_k^T. \quad (2.16)$$

where K_k is the Kalman gain matrix and $P_k^{xx} \triangleq E[(x_k - \hat{x}_k)(x_k - \hat{x}_k)^T]$. Increment k and return to step 1.

□

The notation $\hat{x}_{k|k-1}$ indicates an estimate of $\hat{x}_k$ at time k based on information available up to and including time $k-1$. Note that, for convenience of notation, we employ the notation $k|k-1$ in this Section to discriminate the estimates of the two steps of Kalman filtering algorithms. However, this notation is not used in hybrid filtering algorithms.

2.1.3 State Estimation for Nonlinear Systems

The solution to the state estimation problem for nonlinear systems, (2.1)-(2.2), faces the challenge that the *a posteriori* PDF $p(x_k|y_{1:k})$ cannot be completely characterized by its mean $\hat{x}_k$ and covariance P_k^{xx} [Daum F., 2005]. Hence suboptimal algorithms are employed to circumvent this problem. Some approaches approximate the *a posteriori* PDF from a small set of samples. For instance, the unscented Kalman filter uses a small number of samples chosen deterministically to approximate the mean and covariance of the random variables, whereas the particle filter uses a large number of random samples (Monte Carlo) to approximate the PDFs of the random variables. These approaches are described below.

The Unscented Kalman filter

The unscented Kalman filter (UKF) provides a suboptimal recursive solution to the state estimation problem for nonlinear systems [Julier, S.J., Uhlmann, J.K., 2004]. We present here the UKF following the proposed systematization in [Menegaz, H.M.T., Ishihara, J.Y., Borges, G.A., Vargas, A.N., 2015]. The UKF employs the unscented transformation (UT), that approximates the posterior mean $\hat{y} \in \mathbb{R}^m$ and covariance matrix $P^{yy} \in \mathbb{R}^{m\times m}$ of a random vetor y obtained from the nonlinear transformation

$$y = h(x_1, x_2, c). \tag{2.17}$$

where x_1 and x_2 are *a priori* independent random vectors, with mean $\hat{x}_1 \in \mathbb{R}^n_1$ and $\hat{x}_2 \in \mathbb{R}^n_2$ and covariance matrices $P^{x_1x_1}, P^{x_2x_2} \in \mathbb{R}^{n\times n}$ and c is a deterministic vector, are assumed to be known.

Now, we define the augmented state vector $\tilde{x} \in \mathbb{R}^{\tilde{n}}$, $\tilde{n} = n_1 + n_2$ as

$$\tilde{x} \triangleq \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \tag{2.18}$$

and the augmented covariance matrix $P^{\tilde{x}\tilde{x}} \in \mathbb{R}^{\tilde{n}\times\tilde{n}}$ as

$$P^{\tilde{x}\tilde{x}} \triangleq \begin{bmatrix} P^{x_1x_1} & 0_{n_1\times n_2} \\ 0_{n_2\times n_1} & P^{x_2x_2} \end{bmatrix}. \tag{2.19}$$

UT is based on a set of deterministically chosen samples known as sigma points $\mathcal{X}_j \in \mathbb{R}^{\tilde{n}}$ and associated weights γ_j, $j = 1, \ldots, 2\tilde{n}$. The UT algorithm is described as follows.

Algorithm 2.1.2. *Unscented transform [Menegaz, H.M.T., Ishihara, J.Y., Borges, G.A., Vargas, A.N., 2015]*

1. *Choose the sigma points $\mathcal{X}_j$ and associated weights γ_j as*

$$\mathcal{X} = \hat{\tilde{x}} 1_{1\times 2\tilde{n}} + \sqrt{\tilde{n}} \begin{bmatrix} (P^{\tilde{x}\tilde{x}})^{1/2} & -(P^{\tilde{x}\tilde{x}})^{1/2} \end{bmatrix}, \tag{2.20}$$

$$\gamma_j = \frac{1}{2\tilde{n}}, \tag{2.21}$$

where $\mathcal{X}_j$ is the jth column of the matrix $\mathcal{X} \in \mathbb{R}^{\tilde{n}\times 2\tilde{n}}$, $(.)^{1/2}$ is the Cholesky square root and $1_{1\times 2\tilde{n}} \in \mathbb{R}^{1\times 2\tilde{n}}$ is the matrix with elements equal to 1. The sigma points (2.20) can be partitioned as

$$\begin{bmatrix} \mathcal{X}^{x_1} \\ \mathcal{X}^{x_2} \end{bmatrix} \triangleq \mathcal{X}. \tag{2.22}$$

where $\mathcal{X}^{x_1}$ and $\mathcal{X}^{x_2} \in \mathbb{R}^{n\times 2\tilde{n}}$. Then, each sigma point $\mathcal{X}_j$ is propagated through h yielding

$$\mathcal{Y}_j = h\left(\mathcal{X}_j^{x_1}, \mathcal{X}_j^{x_2}, c\right). \tag{2.23}$$

2. *Obtain the state estimate $\hat{y}$ and covariances P^{yy} and P^{xy} from (2.23) as*

$$\hat{y} = \sum_{j=1}^{2\tilde{n}} \gamma_j \mathcal{Y}_j, \tag{2.24}$$

$$P^{yy} = \sum_{j=1}^{2\tilde{n}} \gamma_j \left[\mathcal{Y}_j - \hat{y}\right]\left[\mathcal{Y}_j - \hat{y}\right]^T, \tag{2.25}$$

$$P^{x_1 y} = \sum_{j=1}^{2\tilde{n}} \gamma_j \left[\mathcal{X}_j^{x_1} - \hat{x}_1\right]\left[\mathcal{Y}_j - \hat{y}\right]^T. \tag{2.26}$$

In this work, for simplicity, we define the unscented transformation as the function UT comprising the set of equations (2.20)-(2.26), that is,

$$\left[\hat{y}, P_k^{yy}, P_k^{x_1 y}\right] = \mathrm{UT}\left(\hat{\tilde{x}}, P^{\tilde{x}\tilde{x}}, c, h\right)$$

where $\tilde{x}_{k-1}$ and $P_{k-1}^{\tilde{x}\tilde{x}}$ are given by (2.18) and (2.19), respectively.

□

The UKF comprises two steps: the *forescast* step and the *data-assimilation* step. The UKF algorithm is described as follows.

Algorithm 2.1.3. *The unscented Kalman filter [Teixeira, B. O. S., Chandrasekar, J., Tôrres, L. A. B., Aguirre, L. A., Bernstein, D. S., 2009]*

Initialize the filter with the state estimate $\hat{x}_0$ and the corresponding covariance matrix $P_0^{xx} = E[(x_0 - \hat{x}_0)(x_0 - \hat{x}_0)^T]$.

1. *Forecast step. Obtain the a priori state estimate $\hat{x}_{k|k-1}$ and the covariance matrix $P_{k|k-1}^{xx}$*

$$\left[\hat{x}_{k|k-1}, P_{k|k-1}^{xx}, 0\right] = \mathrm{UT}\left(\hat{\tilde{x}}_{k-1}, P_{k-1}^{\tilde{x}\tilde{x}}, u_{k-1}, f\right), \tag{2.27}$$

$$\left[\hat{y}_k, P_k^{yy}, P_k^{xy}\right] = \mathrm{UT}\left(\hat{\tilde{x}}_{k|k-1}, P_{k|k-1}^{\tilde{x}\tilde{x}}, 0, h\right). \tag{2.28}$$

where UT refers to Algorithm 2.1.2, where the augmented state vectors and covariance matrices of the equations (2.27) and (2.28) are respectively given by

$$\hat{\tilde{x}}_{k-1} \triangleq \begin{bmatrix} \hat{x}_{k-1} \\ w_{k-1} \end{bmatrix}, \; P^{\tilde{x}\tilde{x}}_{k-1} \triangleq \begin{bmatrix} P^{xx}_{k-1} & 0_{n\times q} \\ 0_{q\times n} & Q_{k-1} \end{bmatrix}, \; \tilde{n} = n + q. \tag{2.29}$$

$$\hat{\tilde{x}}_{k|k-1} \triangleq \begin{bmatrix} \hat{x}_{k|k-1} \\ v_k \end{bmatrix}, \; P^{\tilde{x}\tilde{x}}_{k|k-1} \triangleq \begin{bmatrix} P^{xx}_{k|k-1} & 0_{n\times r} \\ 0_{r\times n} & R_k \end{bmatrix}, \; \tilde{n} = n + r. \tag{2.30}$$

2. *Data-assimilation step. Obtain the a posteriori state estimate $\hat{x}_k$ and its corresponding covariance matrix P^{xx}_k*

$$K_k = P^{xy}_k (P^{yy}_k)^{-1}, \tag{2.31}$$

$$\hat{x}_k = \hat{x}_{k|k-1} + K_k(y_k - \hat{y}_k), \tag{2.32}$$

$$P^{xx}_k = P^{xx}_{k|k-1} - K_k P^{yy}_k K^T_k. \tag{2.33}$$

Increment k and return to step 1.

□

Particle Filtering Methods

The particle filter (PF) has been used to estimate the states of nonlinear systems [Arulampalam, M. S., Maskell, S., Gordon, N., Clapp, T., 2002]. The PF is classified as a Bayesian method implemented by means of Monte Carlo (MC) simulations [Doucet, A., Johansen, A. M., 2008]. The Sequential Monte Carlo (SMC) approach uses a set of particles with their associated weights to represent the *a posteriori* PDF $p(x_k|y_{1:k})$. In particle filtering algorithms, the main limitation of this approach is to sample the true probability distribution $p(x)$ from a set of particles, $x^i_k \sim p(x)$. Thus, the sampling method can be used to generate weight particles from a candidate distribution $q(x)$ which approximates the distribution $p(x)$, that is, $x^i_k \sim q(x)$. The sequential importance sampling (SIS) algorithm is a Monte Carlo method that uses importance sampling to approximate the *a posteriori* PDF [Ristic, B., Arulampalam, S., Gordon, N., 2004].

The particle filters considers that the *a posteriori* PDF $p(x_k|y_{1:k})$ can be characterized by discrete random measurements $\{x^i_k, w^i_k\}^N_{i=1}$, where $x^i_k, i = 1, \ldots, N$ is a set of particles with associated weights $w_k^i, i = 1, \ldots, N$. Thus, the *a posteriori* PDF can be approximated by

$$p(x_k|y_{1:k}) \approx \sum_{i=1}^{N} w_k^i \delta\left(x_k - x_k^i\right), \tag{2.34}$$

where $\delta(.)$ is the Dirac delta function and the weights w_k^i are chosen using the principle of importance sampling [Gustafsson, F., 2010]. These weights are calculated recursively as follows

$$w_k^i = w_{k-1}^i \frac{p(y_k|x_k^i)p(x_k|x_{k-1}^i)}{q(x_k|x_{k-1}^i, y_k)}, \tag{2.35}$$

where $p(y_k|x_k^i)$ is the likelihood function of innovations, $p(x_k|x_{k-1}^i)$ is the *a priori* probability and $q(x_k|x_{k-1}^i, y_k)$ is the candidate distribution. The main limitation of particle filtering methods using importance sampling is the growth of the weight variance over time generating an effect known as particle degeneration [Candy, J.V., 2009]. That is, after a number of iterations, most of the weights tend to zero, resulting in particles of no significance for the distribution to be represented. Thus, the effective number of particles N_{eff} is a measure that quantifies the degeneracy of the particles. This number is given by

$$\hat{N}_{\text{eff}} = \frac{1}{\sum_{i=1}^{N}(w_k^i)^2} \tag{2.36}$$

where $\hat{N}_{\text{eff}}$ is the estimate of the effective number of particles. We employ a resampling step to circumvent the problem of particle degeneration. The resampling preserves the particles with large weights (high probability), whereas the particles with low weights are eliminated [Chen, Z., 2003]. The resampling involves a mapping of the random measurements $\{x_k^i, w_k^i\}$ to the random measurements $\left\{x_k^{i*}, \frac{1}{N}\right\}$ with uniform weights. The new set of particles $\{x_k^{i*}\}_{i=1}^{N}$ is generated by resampling of the approximate discrete representation of $p(x_k|y_{1:k})$, see equation (2.34). The criterion used to apply the resampling is $\hat{N}_{\text{eff}} < N_{\text{threshold}}$, where $N_{\text{threshold}}$ is a threshold chosen by the user [Ristic, B., Arulampalam, S., Gordon, N., 2004].

The different approaches based on the particle filters can be considered as special cases of the SIS algorithm. These special cases are derived from the SIS algorithm by varying the choice of the candidate distribution and/or the resampling step. Thus, there are several particle filter algorithms that solve the state estimation problem for nonlinear systems, such as bootstrap particle filter, auxiliary, regularized, MCMC, with annealing, among others [Doucet, A., de Freitas, J.F., Gordon, N., 2001]. Different methods of resampling can be used in particle filters such as resampling: multinomial, residual and systematic, among others [Carpenter, J., Clifford, P., Fearnhead, P., 1999; Liu, J.S., Chen, R., 1998; Kitagawa, G., 1996].

The generic PF comprises four steps: the *particle generation* step, the *weight calculation* step, the *resampling* step and *combining the estimates* step. The PF algorithm is described as follows.

Algorithm 2.1.4. *Particle filters [Arulampalam, M. S., Maskell, S., Gordon, N., Clapp, T., 2002]*

Initialize the set of particles and their associated weights, $\{x_0^i, w_0^i\}_{i=1}^N$.

1. *Particle generation. Generate a set of particles with uniform weights* $\left\{x_k^i, \frac{1}{N}\right\}_{i=1}^N$ *from the candidate distribution* $q(x_k|x_{k-1}^i, y_k)$ *which approximates the a posteriori PDF* $p(x_k|y_{1:k})$. *In the case of the bootstrap particle filter, the transition probability is used as the candidate distribution,* $q\left(x_k|x_{k-1}^i, y_k\right) = p\left(x_k|x_{k-1}^i\right)$. *The particles are given by*

$$x_k^i \sim q\left(x_k|x_{k-1}^i, y_k\right), \tag{2.37}$$

2. *Weight calculation. Incorporate the information of a new measurement* y_k *to calculate the importance weights of each particle. For the bootstrap particle filter, the weights are obtained using the likelihood function* $p(y_k|x_k^i)$ *(2.35) and then normalize the particle weights. This step is defined by*

$$\tilde{w}_k^i = w_{k-1}^i p(y_k|x_k^i), \tag{2.38}$$

$$w_k^i = \frac{\tilde{w}_k^i}{\sum_{i=1}^N \tilde{w}_k^i}, \tag{2.39}$$

3. *Particle resampling. Perform the resampling process when the effective number of particles is less than the resampling threshold,* $\hat{N}_{eff} < N_{threshold}$.

4. *Combining the estimates. Obtain the state estimate* $\hat{x}_{k|k}$ *and the covariance matrix* $P^{xx}_{k|k}$

$$\hat{x}_k = \sum_{i=1}^{N} w_k^i x_k^i, \tag{2.40}$$

$$P_k^{xx} = \sum_{i=1}^{N} w_k^i \left(x_k^i - \hat{x}_{k|k}\right)\left(x_k^i - \hat{x}_{k|k}\right)^T. \tag{2.41}$$

Increment k and return to step 1.

□

2.1.4 State Estimation for Equality Constrained Systems

The Kalman filter provides optimal state estimates for linear and Gaussian systems. However, additional information about the system in the form of state constraints may be useful for improving the state estimates [Simon, D., 2010]. There are many constraints: equality, inequality or interval, linear or nonlinear, time invariant or time varying.

Many algorithms have been developed for constrained state estimation. These algorithms can be categorized into five classes: the measurement-augmented approach [Tahk, M., Speyer, J. L. , 1990], the estimate projection approach [Simon, D., Chia, T., 2002], the projected sigma-point approach [Vachhani, P., Narasimhan, S., Rengaswamy, R., 2006], the quadratic programming approach [Rao, C. V., Rawlings, J. B., Mayne, D. Q., 2003] and the truncated probability density function approach [Simon, D. J., 2006]. The investigation developed by [Teixeira, B. O. S., Aguirre, L. A., Tôrres, L. A. B., 2010] presents a comparative view of these approaches.

The Kalman filter with Projected Initial Condition

Consider the time-invariant linear system given by

$$x_k = Ax_{k-1} + Bu_{k-1} + w_{k-1}, \tag{2.42}$$

$$y_k = Cx_k + v_k, \tag{2.43}$$

Assume that, in addition, the state vector x_k satisfies the equality constraint

$$Dx_k = d. \tag{2.44}$$

The necessary conditions for the classical Kalman filter to provide state estimates satisfying the equality constraint (2.44) are presented in [Teixeira, B. O. S., Chandrasekar, J., Tôrres, L. A. B., Aguirre, L. A., Bernstein, D. S., 2009]. The next result presents sufficient conditions for the state vector of the linear time-invariant (LTI) system characterized by A, B, and C to satisfy the equality constraint (2.44). In this case, we say that the dynamics is *compatible* with the equality constraint [Rong Li, X., 2016].

Lemma 2.1.1. *[[Teixeira, B. O. S., Chandrasekar, J., Tôrres, L. A. B., Aguirre, L. A., Bernstein, D. S., 2009], Proposition 3.1] For the non-hybrid LTI system with matrices A, B, and C, assume that*

$$DQ = 0_{s\times q}, \tag{2.45}$$

$$DA = D, \tag{2.46}$$

$$DBu_{k-1} = 0_{s\times 1}. \tag{2.47}$$

Then, for all $k \geq 1$, the equality state constraint (2.44) *is verified, where $d = Dx_0$.*

The next result presents the sufficient conditions on the initialization and dynamics for the Kalman filter to yield estimates $\hat{x}_k$ (2.15) and P_k^{xx} (2.16) satisfying $D\hat{x}_k = d$ and $DP_k^{xx} = 0_{s\times n}$.

Proposition 2.1.1. *[[Teixeira, B. O. S., Chandrasekar, J., Tôrres, L. A. B., Aguirre, L. A., Bernstein, D. S., 2009], Proposition 4.3] For LTI dynamic systems, assume that the conditions of Lemma 2.1.1 hold and that the initial estimates satisfy $D\hat{x}_0 = d$ and $DP_0^{xx} = 0_{s\times n}$. Then, the classical Kalman filter yields estimates, $\hat{x}_k$ (2.15) and P_k^{xx} (2.16) satisfying $D\hat{x}_k = d$ and $DP_k^{xx} = 0_{s\times n}$, $\forall k$.*

Remark 2.1.1. *In order to initialize the KF such that $D\hat{x}_0 = d$ and $DP_0^{xx} = 0_{s\times n}$ are verified, one may choose arbitrary values $\bar{x}_0$ and $\bar{P}_0^{xx}$ and project them by*

$$\hat{x}_0 = \mathcal{P}_{\mathcal{N}(D)}\bar{x}_0 + \bar{d}, \tag{2.48}$$

$$P_0^{xx} = \mathcal{P}_{\mathcal{N}(D)}\bar{P}_0^{xx}, \tag{2.49}$$

where $\bar{d} = D^T(DD^T)^{-1}d$ is an offset and the projector is given by

$$\mathcal{P}_{\mathcal{N}(D)} \triangleq I_{n\times n} - WD^T(DWD^T)^{-1}D. \tag{2.50}$$

If $W = I_{n\times n}$, then the projector is orthogonal.

The Equality-constrained Kalman Filter

Consider again the time-varying linear system given by the equations

$$\begin{aligned} x_k &= A_{k-1}x_{k-1} + B_{k-1}u_{k-1} + w_{k-1}, \\ y_k &= C_k x_k + v_k, \end{aligned}$$

Assume that the state vector x_k is known to satisfy the time-varying equality constraint

$$D_k x_k = d_k. \tag{2.51}$$

The equality-constrained Kalman filter (ECKF) projects the state estimate $\hat{x}_k$ given by (2.15) onto the hyperplane defined by (2.51) by means of the *projection* step [Simon, D., Chia, T., 2002; Teixeira, B. O. S., Chandrasekar, J., Tôrres, L. A. B., Aguirre, L. A., Bernstein, D. S., 2009]. The ECKF algorithm is described as follows.

Algorithm 2.1.5. *The equality-constrained Kalman filter [Simon, D., Chia, T., 2002]*
Initialize the filter with the state estimate $\hat{x}_0$ and the corresponding covariance matrix P_0^{xx} and the parameters of the equality constraint D_k and d_k.

1. *Perfom steps 1 to 2 of the KF filter (Algorithm 2.1.2) with $\hat{x}_{k-1}=\hat{x}_{k-1}^{\mathrm{p}}$ and $P_{k-1}^{\mathrm{xx}}=P_{k-1}^{\mathrm{xxp}}$.*

2. *Projection step. Obtain the constrained state estimate $\hat{x}_k^{\mathrm{p}}$ and the covariance matrix $P_k^{xx\mathrm{p}}$*

$$\hat{d}_k = D_k \hat{x}_k, \tag{2.52}$$

$$P_k^{dd} = D_k W_k D_k^T, \tag{2.53}$$

$$P_k^{xd} = W_k D_k^T, \tag{2.54}$$

$$K_k^{\mathrm{p}} = P_k^{xd}(P_k^{dd})^{-1}, \tag{2.55}$$

where $\hat{x}_k$ is given by (2.15) and $W_k \in \mathbb{R}^{n\times n}$ is assumed to be positive definite and is often set to $W_k = P_k^{xx}$, where P_k^{xx} is given by (2.16). Then $\hat{x}_k^{\mathrm{p}}$ and $P_k^{xx\mathrm{p}}$ are given by

$$\hat{x}_k^{\mathrm{p}} = \hat{x}_k + K_k^{\mathrm{p}}(d - \hat{d}_k), \tag{2.56}$$

$$P_k^{xx\mathrm{p}} = P_k^{xx} - K_k^{\mathrm{p}}(P_k^{dd})(K_k^{\mathrm{p}})^T. \tag{2.57}$$

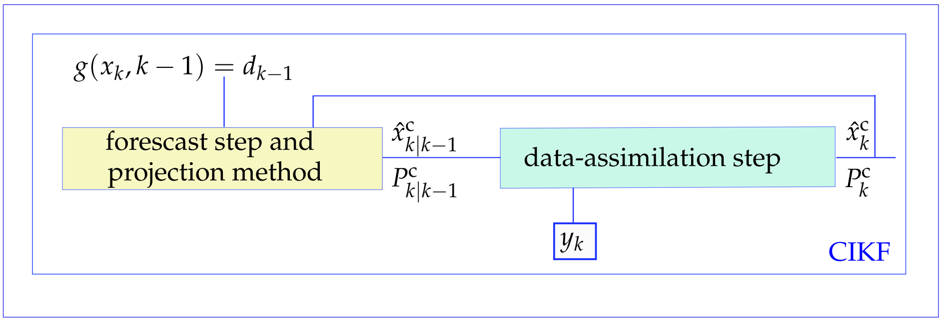

Figure 2.1: Diagram of the constrained innovation Kalman filter. The projection step is incorporated during the forecast step.

Increment k and return to step 1.

□

The Constrained Innovation Kalman filter

The constrained innovation Kalman filter (CIKF) [Mann, G., Hwang, I., 2013] enforces the equality constraint (2.51) by projecting the unconstrained state prediction onto the constraint hyperplane. The equality constraint on state estimate is incorporated during the forecast step, see Figure 2.1. The CIKF comprises two steps: the *forescast and projection* step and the *data-assimilation* step. The CIKF algorithm is described as follows.

Algorithm 2.1.6. *The constrained innovation Kalman filter [Mann, G., Hwang, I., 2013] Initialize the filter with the state estimate $\hat{x}_0$ and the corresponding covariance matrix P_0^{xx} and the parameters of the equality constraint D_k and d_k.*

1. *Forescast and projection step. Obtain the a priori constrained state estimate $\hat{x}^c_{k|k-1}$ and the covariance matrix $P^{xxc}_{k|k-1}$*

$$\begin{aligned} \hat{x}^c_{k|k-1} &= A_{k-1}\hat{x}^c_{k-1} + B_{k-1}u_{k-1} - \\ &\quad J_k\left(D_k\left(A_{k-1}\hat{x}^c_{k-1} + B_{k-1}u_{k-1}\right) - d_k\right), \end{aligned} \tag{2.58}$$

$$P^{xxc}_{k|k-1} = (I - J_k D_k)\left(A_{k-1}P^{xxc}_{k-1}A^T_{k-1} + Q_{k-1}\right)(I - J_k D_k)^T, \tag{2.59}$$

$$\hat{y}_k = C_k \hat{x}^c_{k|k-1}, \tag{2.60}$$

$$P^{yy}_k = C_k P^{xxc}_{k|k-1} C^T_k + R_k, \tag{2.61}$$

$$P^{xy}_k = P^{xxc}_{k|k-1} C^T_k, \tag{2.62}$$

where $J_k = W^{-1}D_k^T\left(D_kW^{-1}D_k^T\right)^{-1}$, *W is a positive definite matrix. For details on the equations (2.58)-(2.59), see [Mann, G., Hwang, I., 2013].*

2. *Data-assimilation step. Obtain the a posteriori constrained state estimate* $\hat{x}_k^c$ *and its corresponding covariance matrix* P_k^{xxc}

$$K_k = P_k^{xy}(P_k^{yy})^{-1}, \tag{2.63}$$

$$\hat{x}_k^c = \hat{x}_{k|k-1}^c + K_k(y_k - \hat{y}_k), \tag{2.64}$$

$$P_k^{xxc} = P_{k|k-1}^{xxc} - K_kP_k^{yy}K_k^T. \tag{2.65}$$

Increment k and return to step 1.

□

The Equality-constrained Unscented Kalman Filter

Consider again the nonlinear discrete-time system given by the equations

$$x_k = f(x_{k-1}, u_{k-1}, w_{k-1}, k-1),$$
$$y_k = h(x_k, v_k, k-1),$$

Now assume that the state vector x_k satisfies the nonlinear equality constraint

$$g(x_k, k) = d_k. \tag{2.66}$$

The algorithms based on the unscented transformation provide suboptimal solutions to the equality-constrained state estimation problem for nonlinear systems. These approaches do not guarantee that the nonlinear equality constraint (2.66) is exactly satisfied, but they provide approximate solutions.

The equality-constrained unscented Kalman filter (ECUKF) yields state estimates that approximately satisfy the nonlinear equality constraint (2.66) [Teixeira, B. O. S., Chandrasekar, J., Tôrres, L. A. B., Aguirre, L. A., Bernstein, D. S., 2009]. In this case, the projection step of ECKF given by (2.52)-(2.57) is extended to the nonlinear case by means of UT, which is given by

$$\left[\hat{d}_k, P_k^{dd}, P_k^{xd}\right] = \mathrm{UT}\left(\hat{x}_k, P_k^{xx}, 0, g\right). \tag{2.67}$$

where the state estimate $\hat{x}_k \in \mathbb{R}^n$ and the covariance matrix $P_k^{xx} \in \mathbb{R}^{n\times n}$ are given by (2.32) and (2.33), respectively. From (2.67), K_k^{p}, $\hat{x}_k^{\mathrm{p}}$ and $\hat{P}_k^{xx\mathrm{p}}$ are given by (2.55), (2.56) and (2.57), respectively. The ECUKF algorithm is described as follows.

Algorithm 2.1.7. *The equality-constrained unscented Kalman filter [Teixeira, B. O. S., Chandrasekar, J., Tôrres, L. A. B., Aguirre, L. A., Bernstein, D. S., 2009]*

Initialize the filter with the state estimate $\hat{x}_0$ and the corresponding covariance matrix P_0^{xx} and the parameters of the equality constraint $g(x_k, k)$ and d_k.

1. *Forecast step. Obtain the a priori state estimate $\hat{x}_{k|k-1}$ and the corresponding covariance matrix $P_{k|k-1}^{xx}$*

$$\begin{aligned} \left[\hat{x}_{k|k-1}, P_{k|k-1}^{xx}, 0\right] &= \mathrm{UT}\left(\hat{\tilde{x}}_{k-1}, P_{k-1}^{\tilde{x}\tilde{x}}, u_{k-1}, f\right), \quad (2.68)\\ \left[\hat{y}_k, P_k^{yy}, P_k^{xy}\right] &= \mathrm{UT}\left(\hat{\tilde{x}}_{k-1}, P_{k-1}^{\tilde{x}\tilde{x}}, c, h\right), \quad (2.69) \end{aligned}$$

where $\hat{\tilde{x}}_{k-1}$ and $P_{k-1}^{\tilde{x}\tilde{x}}$ are respectively given by

$$\hat{\tilde{x}}_{k-1} \triangleq \begin{bmatrix} \hat{x}_{k-1}^{\mathrm{p}} \\ w_{k-1} \end{bmatrix}, \; P_{k-1}^{\tilde{x}\tilde{x}} \triangleq \begin{bmatrix} P_{k-1}^{xx\mathrm{p}} & 0_{n\times q} \\ 0_{q\times n} & Q_{k-1} \end{bmatrix}, \quad (2.70)$$

2. *Data-assimilation step. Obtain the a posteriori state estimate $\hat{x}_k$ and the corresponding covariance matrix P_k^{xx}*

$$\begin{aligned} K_k &= P_k^{xy}(P_k^{yy})^{-1}, \quad (2.71)\\ \hat{x}_k &= \hat{x}_{k|k-1} + K_k(y_k - \hat{y}_k), \quad (2.72)\\ P_k^{xx} &= P_{k|k-1}^{xx} - K_k P_k^{yy} K_k^T, \quad (2.73) \end{aligned}$$

3. *Projection step. Obtain the constrained state estimate $\hat{x}_k^{\mathrm{p}}$ and the covariance matrix $P_k^{xx\mathrm{p}}$*

$$\begin{aligned} \left[\hat{d}_k, P_k^{dd}, P_k^{xd}\right] &= \mathrm{UT}\left(\hat{x}_k, P_k^{xx}, 0, g\right), \quad (2.74)\\ K_k^{\mathrm{p}} &= P_k^{xd}(P_k^{dd})^{-1}, \quad (2.75)\\ \hat{x}_k^{\mathrm{p}} &= \hat{x}_k + K_k^{\mathrm{p}}(d - \hat{d}_k), \quad (2.76)\\ P_k^{xx\mathrm{p}} &= P_k^{xx} - K_k^{\mathrm{p}}(P_k^{dd})(K_k^{\mathrm{p}})^T. \quad (2.77) \end{aligned}$$

Increment k and return to step 1.

□

2.2 Steps of the Hybrid Stochastic Filtering Problem

In this work, we address approximate solutions to the state estimation problem of hybrid systems. In the particular case of Markov jump systems, where the mode transitions at time k are assumed to be independent of $m_{0:k-1}$ and $x_{0:k}$ given the current mode m_k. The solution to this problem may be obtained through the joint *a posteriori* PDF of x_k and m_k (1.4) given a sequence of noisy measurements, the whose equation is repeated here for convenience, that is,

$$\rho(x_k, m_k \mid y_{1:k}) = \sum_{\text{all } m_{1:k}} \rho(x_k \mid m_{1:k}, y_{1:k}) \rho(m_{1:k} \mid y_{1:k}) \ ,$$

where $\rho\left(x_k \mid m_{1:k}, y_{1:k}\right)$ is the *a posteriori* PDF of x_k conditional on the mode m_k and $\rho\left(m_{1:k} \mid y_{1:k}\right)$ is the conditional probability mass function (PMF) of the mode sequence $m_{1:k}$. Figure 2.2 shows how the two terms of the right-hand side of (1.4) are obtained. The approaches used to treat each term of the joint *a posteriori* PDF are presented below.

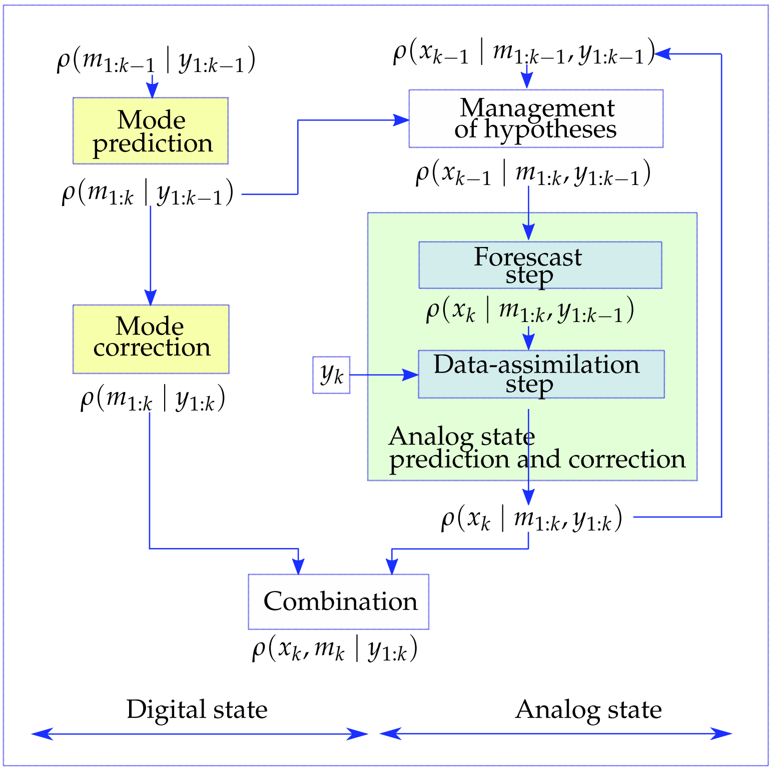

Figure 2.2: Diagram of the steps of the hybrid stochastic filtering problem shows how the two terms of the right-hand side of (1.4) are obtained.

2.2.1 Analog State Estimate

From the first PDF of (1.4), $\rho(x_k \mid m_{1:k}, y_{1:k})$, it is possible to estimate the analog state of the hybrid system. This PDF can be obtained using a filter bank (FB), for example. The FB is based on the dynamic MM approach [Hide C., 2004] and is classified as a Bayesian method in [Mehra R.K., 1972]. The filter bank is compounded by a set of filters that runs in parallel. Different algorithms can be used to compose each bank filter [[Bar-Shalom Y., Li X. R., Kirubarajan T., 2001], p. 441-465], [Chaer W. S., Bishop R. H., Ghosh J., 1997; Mazor, E., Averbuch, A., Bar-Shalom, Y., Dayan, J., 1998]. For example, for linear systems, we employ the Kalman (KF) filter [Kalman, 1960], whereas, for nonlinear systems, we employ Gaussian approximation methods based on the KF as the extended Kalman filter (EKF) [Jazwinski, A. H., 1970; Maybeck, P. S., 1979] and the unscented Kalman filter (UKF) [Julier, S. J., Uhlmann, J. K., Durrant-Whyte, H. F., 2000], as well as particle filtering methods [Arulampalam, M. S., Maskell, S., Gordon, N., Clapp, T., 2002]. With relation to particle filtering, there are several algorithms based on this method, for example, particle filters: Bootstrap, auxiliary, regularized, Markov chain Monte Carlo (MCMC), with annealing, among others [Doucet, A., de Freitas, J.F., Gordon, N., 2001; Boers, Y., Driessen, H, 2000].

In this work, we employ the unscented Kalman filter to compose the bank because such a filter uses a statistical linearization technique that uses a reduced number of samples chosen deterministically [Lefebvre, T., Bruyninckx, H., De Schutter, J., 2002]. In the case of the Extended Kalman filter we employ an analytical linearization of the system model. Thus, the UKF approach is presented as a more efficient algorithm to estimate the states of nonlinear systems [Simon, D. J., 2006]. The KF, UKF and PF approaches are reviewed in Section 2.1.

2.2.2 Digital State Estimate

The second term of (1.4) estimates the digital state of the hybrid system. For such, the conditional PMF of the mode, $\rho(m_{1:k} \mid y_{1:k})$, is obtained using the Bayes' theorem given by

$$\begin{aligned} \rho(m_{1:k} \mid y_{1:k}) &= \frac{\rho\left(y_k|m_{1:k}, y_{1:k-1}\right) p\left(m_{1:k}|y_{1:k-1}\right) p(y_{1:k-1})}{\rho\left(y_k|y_{1:k-1}\right) p(y_{1:k-1})}, \\ &= \frac{\rho\left(y_k|m_{1:k}, y_{1:k-1}\right) p\left(m_{1:k}|y_{1:k-1}\right)}{\rho\left(y_k|y_{1:k-1}\right)}, \end{aligned} \tag{2.78}$$

where $\rho\left(y_k|m_{1:k}, y_{1:k-1}\right)$ is the likelihood function of innovations, $p\left(m_{1:k}|y_{1:k-1}\right)$ is the discrete modal probability and $\rho\left(y_k|y_{1:k-1}\right)$ is a normalization constant. The elements of the equation (2.78) are analyzed as follows. We consider that the likelihood function of innovations is a measure that quantifies the difference between the current observation y_k, and the estimated observation $\hat{y}_k$. This measure is given (at least approximately, if non-Gaussian) by

$$\rho\left(y_k|m_{1:k}, y_{1:k-1}\right) = \frac{1}{\sqrt{(2\pi)^{n_y} det(P_k^{yy})}} \exp\left[-\frac{1}{2}(\eta_k)^T (P_k^{yy})^{-1} \eta_k\right], \tag{2.79}$$

where $\eta_k \triangleq y_k - \hat{y}_k$ is the innovation and P_k^{yy} is the innovation covariance matrix. For the second term of (2.78), the modal probability is defined by

$$p\left(m_{1:k}|y_{1:k-1}\right) = \pi_{j|i} p(m_{1:k-1}|y_{1:k-1}), \tag{2.80}$$

where $\pi_{j|i}$ is the transition probability of the mode $m_{k-1} = i$ to the mode $m_k = j$ and $p(m_{1:k-1}|y_{1:k-1})$ is the *a priori* probability of the modes. The third term of the equation (2.78), $\rho\left(y_k|y_{1:k-1}\right)$, is independent of x_k and m_k and can be eliminated in the process of normalization of probabilities [Bar-Shalom, Y., Challa, S.,Blom, H. A., 2005; Ristic, B., Arulampalam, S., Gordon, N., 2004; Hofbaur, M. W., 2005].

Transitions between modes

In the context of hybrid stochastic filtering is necessary to describe the way in which the occurrence of transitions between different digital dynamics. We consider that the transitions between the operating modes of the system depend on the analog state variable and/or the digital state variable. For example, in the investigations performed by [Seah, C. E., Hwang, I., 2009; Hwang, I. , Balakrishnan H., Tomlin, C., 2006; Benazera, E., Travé-Massuyès, L., 2009], the transitions between modes depend on the analog state of the system, whereas, in the studies developed by [Mazor, E., Averbuch, A., Bar-Shalom, Y., Dayan, J., 1998; Blom, H. A.P., Bar-Shalom, Y., 1988; Boers, Y., Driessen, H, 2004], the transitions between modes depend only on the digital state variable. The types of mode transitions of the system can be illustrated in the example below.

Example 2. Consider the dynamic system given by a bouncing ball under the action of acceleration of gravity. This system has two analog states, the position and the velocity of the ball, and the digital states are the jumping count. In this system, the transitions between modes depend on the digital state variable because the mode transition occurs when the ball touches the ground. On the other hand, we consider the dynamic system of heating a house. This system has an analog state, the temperature of the house, and two operating modes controlled by a thermostat, natural cooling and forced heating. Depending on the preset maximum and minimum temperature limits, the thermostat switches the heater on and off. In this system, the transitions between modes depend on the analog state variable. □

In this work, we consider that the mode transitions are independent of analog states. A hybrid system with such property is known in the literature as a discrete-time Markov jump system. Thus, we assume that the transitions between modes are modeled by a Markov chain and occur according to the transition probability matrix Π, as defined in Section 1.2.

2.2.3 Limitation of the Hybrid Stochastic Filtering Problem

The main hurdle of solving the hybrid stochastic filtering problem is that both the number of possible sequences of modes and the number of possible analog trajectories (and, consequently, the computational cost) grow exponentially over time, as shown in Figure 1.2. In order to analyze this question, we employ the first PDF of the equation

(1.4) at the time $k-1$ using the Total Probability Theorem[2] given by

$$\rho(x_{k-1}|m_{1:k}{=}j, y_{1:k-1}) = \sum_{i=1}^{M} \rho(x_{k-1}|m_{1:k-1}{=}i, m_{1:k}{=}j, y_{1:k-1}) \\ p(m_{1:k-1}{=}i|m_{1:k}{=}j, y_{1:k-1}), \quad (2.81)$$

where the two terms in the equation (2.81) are described below. In the first term, the mode m_k is independent of x_{k-1} if $m_{1:k-1}$ is known, then this term can be rewritten as

$$\rho(x_{k-1}|m_{1:k-1}{=}i, m_{1:k}{=}j, y_{1:k-1}) = \rho(x_{k-1}|m_{1:k-1}{=}i, y_{1:k-1}), \quad (2.82)$$

where $\rho(x_{k-1}|m_{1:k-1}{=}i, y_{1:k-1})$ is a posteriori PDF of x_{k-1} conditional on the mode $m_{1:k-1}$. In the second term of the equation (2.81), $p(m_{1:k-1}{=}i|m_{1:k}{=}j, y_{1:k-1})$ is the conditional probability of the mode. Using Bayes' Theorem, this probability can be expressed as

$$\begin{aligned} p(m_{1:k-1}{=}i|m_{1:k}{=}j, y_{1:k-1}) &= \frac{p(m_{1:k}{=}j|m_{1:k-1}{=}i, y_{1:k-1})p(m_{1:k-1}{=}i|y_{1:k-1})}{p(m_{1:k}{=}j|y_{1:k-1})}, \\ &= \pi_{j|i} \frac{p(m_{1:k-1}{=}i|y_{1:k-1})}{p(m_{1:k}{=}j|y_{1:k-1})}, \end{aligned} \quad (2.83)$$

where $p(m_{1:k}{=}j|m_{1:k-1}{=}i, y_{1:k-1}) = \pi_{j|i}$. Finally, the equations (2.82) and (2.83) are replaced in (2.81) yielding

$$\rho(x_{k-1}|m_{1:k}{=}j, y_{1:k-1}) = \sum_{i=1}^{M} w_k^i \rho(x_{k-1}|m_{1:k-1}{=}i, y_{1:k-1}), \quad (2.84)$$

where

$$w_k^i \triangleq \pi_{j|i} \frac{p(m_{1:k-1}{=}i|y_{1:k-1})}{p(m_{1:k}{=}j|y_{1:k-1})}. \quad (2.85)$$

In the equation (2.84), the PDF $\rho(x_{k-1}|m_{1:k-1}{=}i, y_{1:k-1})$ is a sum of M weighted PDFs. This sum of PDFs is related to a tree of possible mode sequences due to the lack of knowledge of the transitions between modes of the system (Figure 1.2). As a

[2] The Total Probability Theorem can be defined for conditional probabilities such as

$$P(A|C) = \sum_{s=1}^{n} P(A|C \cap B_s)P(B_s|C),$$

where B_s are events that partition a sample space and C is an event independent of B_s.

consequence, the number of parameters required to characterize the *a posteriori* PDF grows exponentially over time [Ristic, B., Arulampalam, S., Gordon, N., 2004]. To circumvent this practical limitation, hybrid stochastic filtering methods employ different mechanisms to yield approximate solutions.

2.3 Hybrid Stochastic Filtering Methods

Several techniques have been used to estimate the states of hybrid systems, such as the interacting multiple-model (IMM) algorithm [Blom, H. A.P., Bar-Shalom, Y., 1988], the multiple-model and multiple-hypothesis (M^3H) algorithm [Driessen, H., Boers, Y., 2001; Boers, Y., Driessen, H, 2004], the multiple-hypothesis mixing filter (MHMF) [Santana, P. H. R. Q. A., Menegaz, H. M., Borges G. A., Ishihara, J. Y., 2010], the bootstrap particle filter, [Boers, Y., Driessen, H, 2000], the Rao-Blackwellized particle filter [Hendeby, G., Karlsson, R., Gustafsson, F., 2010], and the combination of the IMM approach with the particle filter (IMMPF) [Wang, X., Xu, M., Wang, H., Wu, Y., Shi, H., 2012]. Some of these approaches are reviewed below.

2.3.1 The Interacting Multiple-Model Algorithm

The Interacting Multiple-Model (IMM) filter provides an approximate recursive solution to the state estimation problem for unconstrained hybrid systems [Blom, H. A.P., Bar-Shalom, Y., 1988]. The IMM approach has been one of the main approaches to estimate states of hybrid systems [Blom, H. A.P., Bar-Shalom, Y., 1988; Mazor, E., Averbuch, A., Bar-Shalom, Y., Dayan, J., 1998]. In order to address the exponential growth problem, the IMM algorithm mixes the previous analog state estimates to generate new initial values for each filter in the next time step. For such, we approximate (2.82) as

$$\rho(x_{k-1}|m_{1:k-1} = s, y_{1:k-1})) \simeq \mathcal{N}\left(\hat{x}^s_{k-1}, P^{xx,s}_{k-1}\right), \tag{2.86}$$

where $\mathcal{N}$ represents a Gaussian distribution with mean $\hat{x}^s_{k-1}$ and its corresponding covariance, $P^{xx,s}_{k-1}$, $s = 1, \ldots, M$. Thus, we replace the approximation of the equation

(2.86) into (2.84) yielding

$$\rho\left(x_{k-1}|m_{1:k}, y_{1:k}\right) \simeq \sum_{s=1}^{M} w_k^s \mathcal{N}\left(\hat{x}_{k-1}^s, P_{k-1}^{xx,s}\right). \tag{2.87}$$

The IMM is a five-step algorithm: the *mode probability prediction* step, the *mixing of estimates* step, the *analog state prediction and correction* step, the *mode probability correction* step and the *combining the estimates* step, as shown in Figure 2.3. In this algorithm, the two terms of the right-hand side of (1.4) are treated as follows. The first term, $\rho(x_k \mid m_{1:k}, y_{1:k})$, is addressed in the *mixing of estimates* and in the *prediction and correction of estimates* steps, whereas, the second term, $\rho(m_{1:k} \mid y_{1:k})$, is processed in the *mode probability prediction* step and updated in the *mode probability correction* step. We assume that the analog state estimates are obtained from the weighted combination of the recursive estimates of the filter bank, whereas, the digital state estimate is obtained as being the most likely *a posteriori* mode.

Figure 2.3 summarizes the IMM algorithm in five recursive steps. In the first step of the IMM, the *a priori* probability of the mode, $w_{k-1}^s, s = 1, \ldots, M$, are obtained. In the second step, the previous state vector estimates $\hat{x}_{k-1}^r$, the matrix covariance $P_{k-1}^{\text{xx},r}$ and the mixing probabilities $\mu_{k-1}^{s|r}, r = 1, \ldots, M$, are mixed to generate new initial values for each filter, the state vector estimates $\bar{x}_{k-1}^s$ and the corresponding covariance $\bar{P}_{k-1}^{\text{xx},s}$. In the third step, the filter bank provides the estimate of the state vector $\hat{x}_k^s$, the covariance matrix $P_k^{\text{xx},s}$, the innovation $\eta_k^s \triangleq y_k^s - \hat{y}_k^s$ and the innovation covariance matrix $P_k^{yy,s}$. In the fourth step, the *a posteriori* probability of the mode γ_k^s, is obtained using Bayes' theorem. In the fifth step, the estimates provided by the filter bank and the *a posteriori* probability of the mode are combined to obtain the estimate of the state vector $\hat{x}_k$, and associated covariance P_k^{xx} at time k. The IMM algorithm is described as follows.

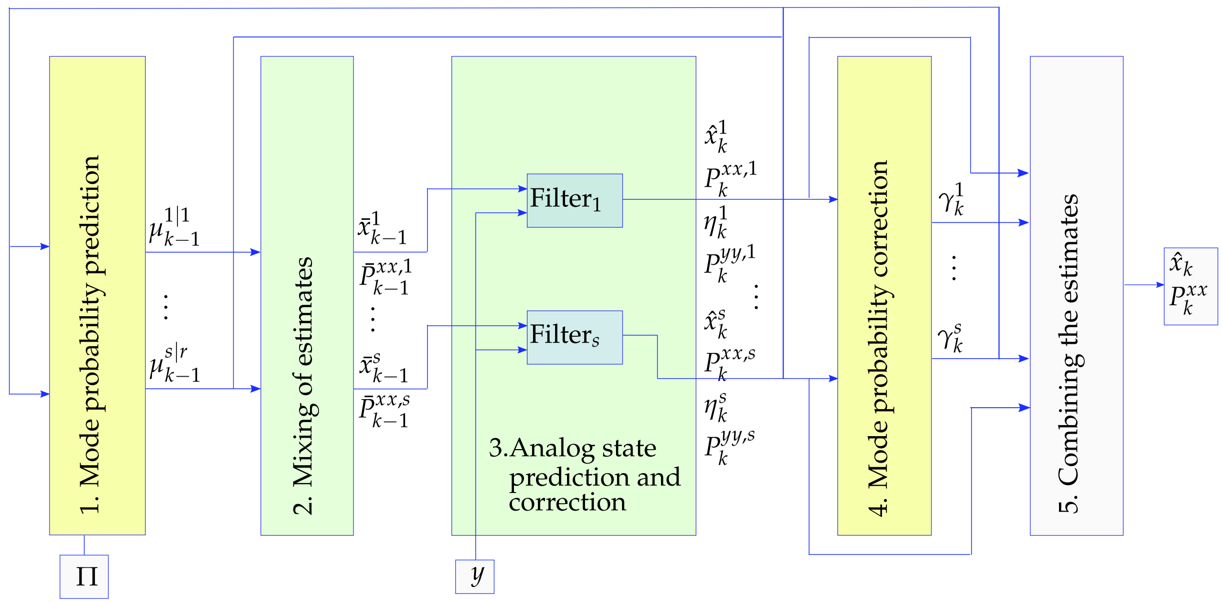

Figure 2.3: Diagram of the steps of the IMM algorithm. The IMM algorithm comprises five steps: in the *mode probability prediction* step, the *a priori* probabilities of the modes are obtained; in the *mixing of estimates* step, the estimates of the state vector and associated covariance matrix for each filter are obtained; in the *analog state prediction and correction* step, the M filters are employed to estimate the state vector and the associated covariance matrix; in the *mode probability correction* step, the *a posteriori* probabilities of the modes are obtained using Bayes' theorem and in the *combining the estimates* step, the M recursive estimates of each filter and their respective *a posteriori* probabilities are combined. Note that, the first and second term of (1.4) are respectively treated in the green and yellow steps in the IMM algorithm.

Algorithm 2.3.1. *The interacting multiple-model filter [Blom, H. A.P., Bar-Shalom, Y., 1988] Initialize each sth filter with the state estimate $\hat{x}_0^s$ and corresponding covariance $P_0^{xx,s}$, and the mode probability $\gamma_0^s, s = 1, \ldots, M$.*

1. *Mode probability prediction. Obtain the a priori probability of the modes $w_{k-1}^s, s = 1, \ldots, M$, and obtain the mixing probabilities $\mu_{k-1}^{s|r}$,*

$$\mu_{k-1}^{s|r} = \frac{\pi_{s|r}\gamma_{k-1}^r}{w_{k-1}^s}, \tag{2.88}$$

where $\pi_{s|r}$ is given by (1.2) and γ_{k-1}^r is the r-th a posteriori probability of the modes, $r = 1, \ldots, M$. The a priori probability of the modes is given by

$$w_{k-1}^s \triangleq \sum_{r=1}^{M} \pi_{s|r}\gamma_{k-1}^r. \tag{2.89}$$

2. *Mixing of estimates. Obtain the mixed estimates $\bar{x}_{k-1}^s$ and $\bar{P}_{k-1}^{xx,s}$, $s = 1, \ldots, M$ from the interaction of the M filters by mixing the previous state vector estimates $\hat{x}_{k-1}^r$ and covariance matrix $P_{k-1}^{xx,r}$ with the respective mixing probabilities $\mu_{k-1}^{s|r}$,*

$$\bar{x}_{k-1}^s = \sum_{r=1}^{M} \mu_{k-1}^{s|r}\hat{x}_{k-1}^r, \tag{2.90}$$

$$\bar{P}_{k-1}^{xx,s} = \sum_{r=1}^{M} \mu_{k-1}^{s|r}\left[P_{k-1}^{xx,r} + \left(\hat{x}_{k-1}^r - \bar{x}_{k-1}^s\right)\left(\hat{x}_{k-1}^r - \bar{x}_{k-1}^s\right)^T\right]. \tag{2.91}$$

3. *Analog state prediction and correction. For each estimate $\bar{x}_{k-1}^s$, run a filter to estimate the state vector $\hat{x}_k^s$, the covariance matrix $P_k^{xx,s}$, the innovation η_k^s, and the innovation covariance matrix $P_k^{yy,s}$. Each s-th filter is given by*

$$\left\{\hat{x}_k^s, P_k^{xx,s}, \eta_k^s, P_k^{yy,s}\right\} = \mathit{Filter}\left(\bar{x}_{k-1}^s, \bar{P}_{k-1}^{xx,s}, y_k^s, f_s, h_s, Q_s, R_s\right), \tag{2.92}$$

where each "Filter" uses one of the M possible pairs of stochastic models f_s, h_s, Q_s and R_s. In principle, any filter algorithm with prediction-correction structure can be used.

4. *Mode probability correction. Obtain the a posteriori probability γ_k^s of each estimate using the Bayes' theorem as follows:*

$$\gamma_k^s = \frac{\rho_s(\eta_k^s)w_{k-1}^s}{\sum_{r=1}^{M}\rho_r(\eta_k^r)w_{k-1}^r}, \tag{2.93}$$

where w_{k-1}^s is given by (2.89) and $\rho_s(\eta_k^s)$ is the likelihood function of innovations, given (at least approximately, if non-Gaussian) by

$$\rho_s(\eta_k^s) = \frac{1}{\sqrt{(2\pi)^m |P_k^{yy,s}|}} \, exp\left[-\frac{1}{2}(\eta_k^s)^T (P_k^{yy,s})^{-1} \eta_k^s\right] \, . \tag{2.94}$$

5. *Combining the estimates. Combine the M estimates to obtain the state estimate $\hat{x}_k$ and its corresponding covariance P_k^{xx} at time k as*

$$\hat{x}_k = \sum_{s=1}^{M} \gamma_k^s \hat{x}_k^s, \tag{2.95}$$

$$P_k^{xx} = \sum_{s=1}^{M} \gamma_k^s \left[P_k^{xx,s} + (\hat{x}_k^s - \hat{x}_k)\,(\hat{x}_k^s - \hat{x}_k)^T\right]. \tag{2.96}$$

and obtain the estimate $\hat{m}$ as being the most likely mode a posteriori. Increment k and return to step 1.

□

Recent work [Santana, P. H. R. Q. A., Borges G. A., Ishihara, J. Y., 2010] illustrates a modified version of the IMM algorithm. In this work, the transition probability matrix correction step, Π_k, is incorporated after the fifth step of the classical IMM algorithm.

2.3.2 The Multiple-Model and Multiple Hypothesis Algorithm

The multiple models and multiple hypotheses (M^3H) filter provides an approximate solution to the state estimation problem of hybrid systems [Driessen, H., Boers, Y., 2001; Boers, Y., Driessen, H, 2004]. In order to address the exponential growth problem, the M^3H keeps a limited buffer of hypotheses by means of merging and discarding hypotheses. In the M^3H filter, merging occurs when 2 or more hypotheses share the same sequence of modes for the previous d time steps. In addition, hypotheses are discarded whenever their likelihood is below a given threshold. The IMM algorithm can be seen as a special case of the M^3H algorithm with merging depth $d = 1$, see Figure 1.2. Note that, the merging depth d is a tuning parameter.

At this point, we define the buffer of hypotheses $\mathcal{H}_k \triangleq \{H_k^r,\ r = 1, \ldots, N\}$, $N < M$, where each hypothesis H_k^r is defined by a tuple $(\hat{m}_{k-d:k}^r, \hat{x}_k^r, P_k^{xx,r})$ composed, respectively, of a sequence of the last d modes, a mean vector and a covariance matrix for the analog state PDF and the hypothesis probability $p_k^r \triangleq p(H_k^r)$.

The M^{3}H is a six-step algorithm: the *hypotheses probability prediction* step, the *hypotheses merging* step, the *hypotheses pruning* step, the *analog state prediction and correction* step, the *hypotheses probability correction* step and the *combining the estimates* step, as shown in Figure 2.5. In this algorithm, the two terms of the right-hand side of (1.4) are treated as follows. The first term, $\rho(x_k \mid m_{1:k}, y_{1:k})$, is addressed in the *prediction and correction of estimates* step, whereas, the second term, $\rho(m_{1:k} \mid y_{1:k})$, is processed in the *hypotheses probability prediction* step and updated in the *hypotheses probability correction* step. In addition, the M^{3}H approach incorporates the *hypotheses merging* and *hypotheses pruning* steps instead of the *mixing of estimates* step in the IMM approach. We assume that the analog state estimates are obtained from the weighted combination of the recursive estimates of the filter bank, whereas, the digital state estimate is obtained as being the most likely *a posteriori* mode.

Figure 2.5 summarizes the M^{3}H algorithm in six recursive steps. In the first step of M^{3}H, the s-th sequences of modes $\hat{m}_{k-d:k}^s$, is propagated one step ahead into all possible future sequences and the resulting *a priori* probabilities $\tilde{p}_k^s$ are obtained, as illustrated in the example provided in Figure 2.4. In the second step, from the set of resulting hypotheses, those hypotheses with the same sequence of modes in the last d steps are merged (see Figure 2.4). Note that the number of hypotheses grows by a factor of M in the propagation step. The merging step may reduce this number up to M times and guarantees this number is upper bounded by M^d. In the third step, the following criteria for hypothesis pruning are considered: i) the hypotheses with probability lower than the elimination threshold given by $0 \leq \varepsilon \leq 1$ are eliminated and ii) the number of hypotheses is kept under $N_{\max}$ by eliminating hypotheses in increasing order of probability. The threshold ε and $N_{\max}$ are tuning parameters to be chosen by the user. In the fourth step, the filter bank provides the estimate of the state vector $\hat{x}_k^s$, the covariance matrix $P_k^{xx,s}$, the innovation η_k^s, and the innovation covariance matrix $P_k^{yy,s}$, associated with each one of the hypothesis $H_k^s \in \mathcal{H}_k$. In this step, a filter is run for each one of the N_p hypothesis. In the fifth step, the *a posteriori* probability of the sequence of modes p_k^s, is obtained from $\tilde{p}(\hat{m}_{k-d:k}^s)$ using Bayes' theorem. In the sixth step, the estimates provided by the filter bank and the *a posteriori* probability of the hypotheses are combined. The M^{3}H algorithm is described as follows.

Example 3. Consider the temporal evolution of a dynamical system with a digital state (mode) that can assume $M = 2$ possible values: white or black, as illustrated in Figure 2.4. We assume that the merging depth is $d = 2$. In the propagation of the hypotheses at the time $k-2$, we have four hypotheses. At time $k-1$, we have eight hypotheses, while at the time k, we have sixteen possible hypotheses of the system. Note that, the hypotheses H_k^8 and H_k^{16} are merged because they have the same sequence of modes (in red). □

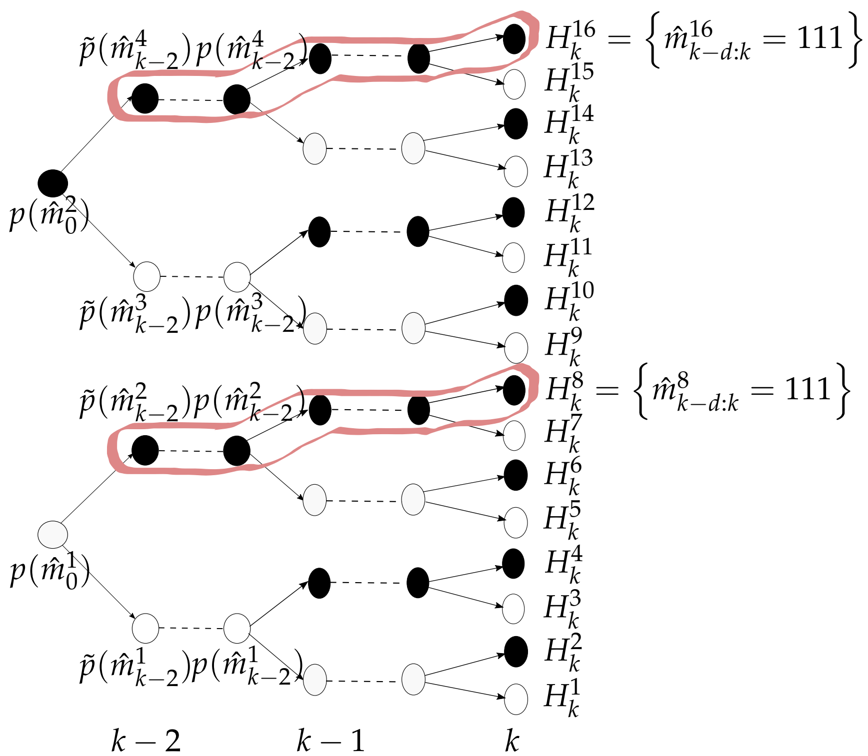

Figure 2.4: Evolution of the sequence of modes of hypotheses from time $k-d$ to time k for a system with two operating modes: white (0) and black (1). The binary code of each hypothesis represents the sequence of modes. The merging depth is $d = 2$. $\tilde{p}(\hat{m}_{k-d:k}^s)$ is the probability of the sequence of modes. In this illustrative example, the hypotheses H_k^8 and H_k^{16} are merged because they have the same sequence of modes (in red). The dashed line (- -) indicate the hypothesis probability correction (step 5).

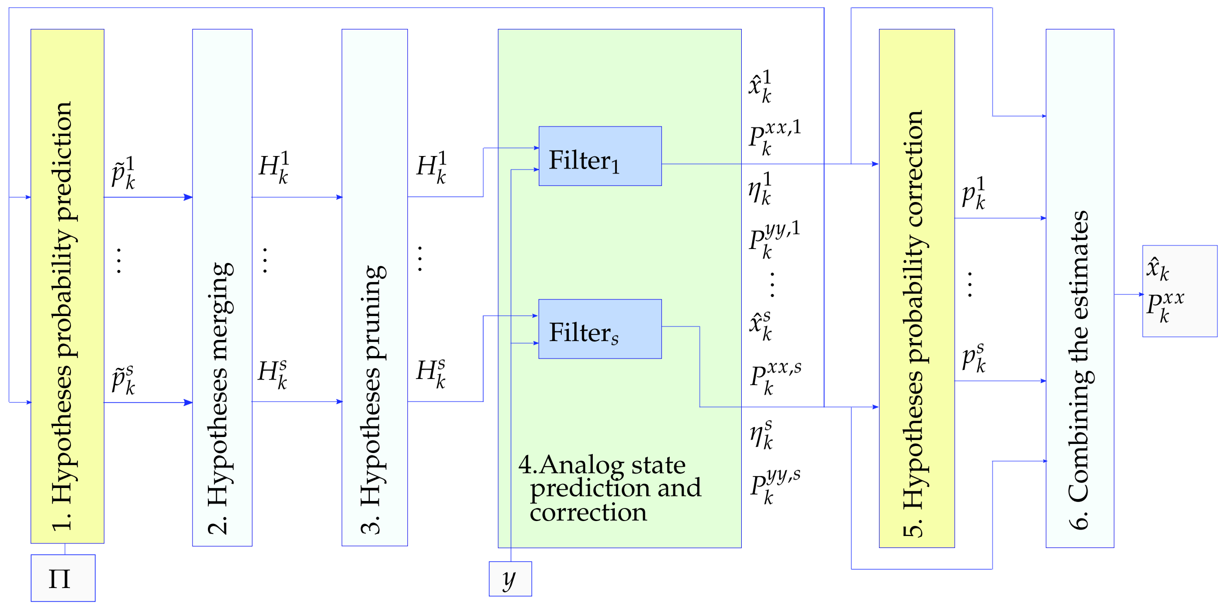

Figure 2.5: Diagram of the steps of the M^{3}H algorithm. The M^{3}H algorithm comprises six steps: in the *hypotheses probability prediction* step, the s-th sequences of modes are generated and the *a priori* probabilities are obtained; in the *hypotheses merging* step, the hypotheses with the same sequence of modes in the last d steps are merged; in the *hypotheses pruning* step, the hypotheses with probability lower are eliminated; in the *analog state prediction and correction* step, the state vector and the associated covariance matrix are estimated; in the *hypotheses probability correction* step, the *a posteriori* probability of the sequences of modes are obtained using Bayes' theorem and in the *combining the estimates* step, the recursive estimates of each filter and their respective *a posteriori* probabilities are combined. Note that, the first and second term of (1.4) are respectively treated in the green and yellow steps in the M^{3}H algorithm.

Algorithm 2.3.2. *The multiple models and multiple hypotheses filter [Boers, Y., Driessen, H, 2004]*

Initialize the initial buffer of hypotheses $\mathcal{H}_0$ and the initial hypothesis probability p_k^0 and set the tuning parameters as follows: the merging depth d and the pruning threshold ε and the maximum number of hypotheses $N_{\max}$.

1. *Hypotheses probability prediction. For all $H_{k-1}^r \in \mathcal{H}_{k-1}$, generate all sequences $\hat{m}_{k-d:k}^s$ such that $\hat{m}_{k-d:k-1}^s = \hat{m}_{k-d:k-1}^r$, $\hat{m}_k^s = j$ and $s = (r-1)M + j$, $j = 1, \ldots, M$; store the new hypotheses in a buffer $\mathcal{H}_k = \{(\hat{m}_{k-d:k}^s, \hat{x}_{k-1}^r, P_{k-1}^{xx,r})\}$ and update their* a priori *probability to*

$$\tilde{p}_k^s = \pi_{j|i} p_{k-1}^r, \tag{2.97}$$

   where $i = \hat{m}_{k-1}^r$.

2. *Hypotheses merging. Merge the hypotheses from $\mathcal{H}_k$ with identical sequences $\hat{m}_{k-d:k}^s$ by keeping that one with the largest* a priori *probability $\tilde{p}_k^s$. Normalize the probabilities of the remaining hypotheses.*

3. *Hypotheses pruning. Discard the hypotheses H_k^s from buffer $\mathcal{H}_k$ with $\tilde{p}_k^s < \varepsilon$. Next, if the size of the resulting buffer of hypotheses is larger than $N_{\max}$, the less likely hypotheses are eliminated to comply with the desired maximum size. Normalize the probabilities of the remaining hypotheses.*

4. *Analog state prediction and correction. For each hypothesis $H_k^s \in \mathcal{H}_k$, run a filter to estimate the state vector $\hat{x}_k^s$, the covariance matrix $P_k^{xx,s}$, the innovation η_k^s, and the innovation covariance matrix $P_k^{yy,s}$ and make $H_k^s = (\hat{m}_{k-d:k}^s, \hat{x}_k^s, P_k^{xx,s})$. Each filter is given by*

$$\left\{\hat{x}_k^s, P_k^{xx,s}, \eta_k^s, P_k^{yy,s}\right\} = \mathit{Filter}\left(\hat{x}_{k-1}^s, P_{k-1}^{xx,s}, y_k^s, f_s, h_s, Q_s, R_s\right), \tag{2.98}$$

   Here each "Filter" uses one of the M possible pairs of stochastic models f_s, h_s, Q_s and R_s.

5. *Hypotheses probability correction. Obtain the a posteriori probability* p_k^s *of hypothesis* H_k^s *using the Bayes' theorem as follows:*

$$p_k^s = \frac{\rho_s(\eta_k^s)\tilde{p}_k^s}{\sum_{H_k^j \in \mathcal{H}_k} \rho_j(\eta_k^j)\tilde{p}_k^j}, \tag{2.99}$$

where $\rho_s(\eta_k)$ *is the likelihood function of innovations, given (at least approximately, if non-Gaussian) by*

$$\rho_s(\eta_k^s) = \frac{1}{\sqrt{(2\pi)^m |P_k^{yy,s}|}} exp\left[-\frac{1}{2}(\eta_k^s)^T (P_k^{yy,s})^{-1} \eta_k^s\right] . \tag{2.100}$$

6. *Combining the estimates. Combine the estimates from the buffer of hypothesis to obtain the mean and covariance of the corresponding Gaussian mixture*

$$\hat{x}_k = \sum_{H_k^s \in \mathcal{H}_k} p_k^s \hat{x}_k^s, \tag{2.101}$$

$$P_k^{xx} = \sum_{H_k^s \in \mathcal{H}_k} p_k^s \left[P_k^{xx,s} + (\hat{x}_k^s - \hat{x}_k)(\hat{x}_k^s - \hat{x}_k)^T\right]. \tag{2.102}$$

and return $\hat{x}_k$ *and* P_k^{xx} *as the state estimate and its corresponding covariance at time k and the estimate* $\hat{m}$ *as being the most likely mode a posteriori. Increment k and return to step* 1.

□

2.3.3 The Multiple-Hypothesis Mixing Filter

The Multiple-hypothesis mixing filter (MHMF) provides state estimates for hybrid systems. This approach proposes to improve the performance of hybrid stochastic filtering through the ability to track multiple hypotheses. The main features of the MHMF approach are the generalization of the *mixing of estimates* step of the IMM algorithm considering $d > 1$, and the correction of the transition probability matrix (TPM) [Santana, P. H. R. Q. A., Menegaz, H. M., Borges G. A., Ishihara, J. Y., 2010]. To update the TPM is used the Quasi-Bayesian algorithm [Jilkov, V., Li, X., 2004].

The MHMF approach generalizes the *mixing of estimates* step of the IMM algorithm considering $d > 1$. This step is treated as follows. First, we consider that in the *mixing of estimates* step of the classical IMM the merging depth is $d = 1$. This step is given by the equation (2.87), repeated here for convenience,

$$\rho(x_{k-1}|m_{1:k} = j, y_{1:k-1}) = \sum_{i=1}^{M} w_k^i \rho(x_{k-1}|m_{1:k-1} = i, y_{1:k-1}), \tag{2.87}$$

where w_k^i is given into (2.85) and $\rho(x_{k-1}|m_{1:k-1} = i, y_{1:k-1})$ is given in (2.86). Consider now that the *mixing of estimates* step of the IMM (2.87) can be modified. Then, we assume that $\rho(x_{k-1}|m_{1:k-1} = i, y_{1:k-1})$ is given by

$$\begin{aligned}\rho(x_{k-1}|m_{1:k-1} = i, y_{1:k-1}) = \sum_{s=1}^{M^d} &\rho(x_{k-1}|m_{1:k-1} = i, H_{k-1}^s, y_{1:k-1})\\ &p(H_{k-1}^s|m_{1:k-1} = i, y_{1:k-1}),\end{aligned} \tag{2.103}$$

where H_{k-1}^s is a particular hypothesis being tracked between two mixing steps of estimates. The second term of the equation (2.103) is given by

$$p(H_{k-1}^s|m_{1:k-1} = i, y_{1:k-1}) = \frac{p(m_{1:k-1} = i|H_{k-1}^s, y_{1:k-1})p(H_{k-1}^s|y_{1:k-1})}{p(m_{1:k-1} = i|y_{1:k-1})}. \tag{2.104}$$

The MHMF is a seven-step algorithm: the *hypotheses probability prediction* step, the *hypotheses pruning* step, the *mixing of hypotheses* step, the *analog state prediction and correction* step, the *hypotheses probability correction* step, the *combining the estimates* step and the *transition probability matrix correction* step, as shown in Figure 2.6. In this algorithm, the two terms of the right-hand side of (1.4) are treated as follows. The first term, $\rho(x_k \mid m_{1:k}, y_{1:k})$, is addressed in the *hypotheses merging* and *prediction and correction of estimates* steps, whereas the second term, $\rho(m_{1:k} \mid y_{1:k})$, is processed in the *hypotheses probability prediction* step and updated in the *hypotheses probability correction* step.

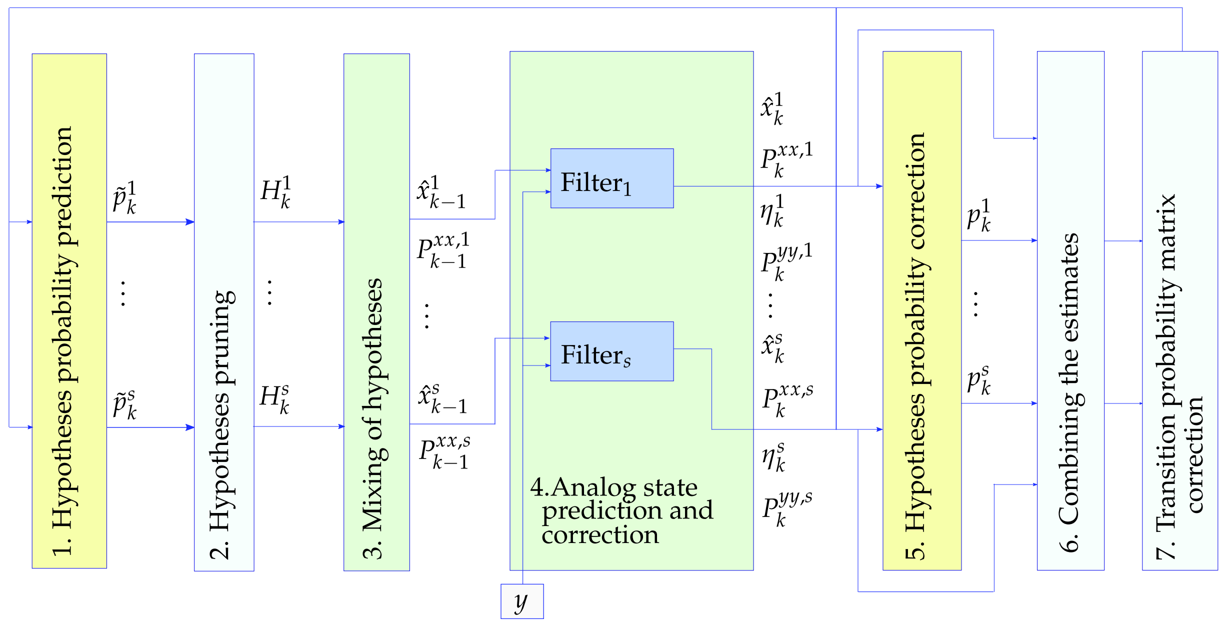

Figure 2.6: Diagram of the steps of the MHMF algorithm. The MHMF algorithm comprises seven steps: in the *hypotheses probability prediction* step, the s-th sequences of modes are generated and the *a priori* probabilities are obtained; in the *hypotheses pruning* step, the hypotheses with low probability are eliminated; in the *mixing of hypotheses* step, the estimates of the state vector and associated covariance matrix for each filter are obtained; in the *analog state prediction and correction* step, the state vector and the associated covariance matrix are estimated; in the *hypotheses probability correction* step, the *a posteriori* probabilities of the sequences of modes are obtained using Bayes' theorem; in the *combining the estimates* step, the recursive estimates of each filter and their respective *a posteriori* probabilities are combined and the *transition probability matrix correction* step, the Quasi-Bayesian algorithm is employed to estimate the transition probability matrix. Note that, the first and second term of (1.4) are respectively treated in the green and yellow steps in the MHMF algorithm.

Figure 2.6 summarizes the MHMF algorithm in seven recursive steps. In the first step of MHMF, the s-th sequences of modes $\hat{m}^s_{k-d:k}$, is propagated one step ahead into all possible future sequences and the resulting *a priori* probabilities $\tilde{p}^s_k$ are obtained. In the second step, the hypotheses with probability lower than the elimination threshold given by $0 \leq \varepsilon \leq 1$ are eliminated. In the third step, the previous state vector estimates $\hat{x}^r_{k-1}$, the matrix covariance $P^{xx,r}_{k-1}$ and the mode probability predictions $w^r_{k-1}, r = 1, \dots, M$, are mixed to generate new initial values for each filter, the state vector estimates $\hat{x}^s_{k-1}$ and the corresponding covariance $P^{xx,s}_{k-1}$. In the fourth step, the filter bank provides the estimate of the state vector $\hat{x}^s_k$, the covariance matrix $P^{xx,s}_k$, the innovation η^s_k, and the innovation covariance matrix $P^{yy,s}_k$, associated to each one of the hypothesis $H^s_k \in \mathcal{H}_k$. In this step, a filter is run for each one of the N_p hypothesis. In the fifth step, the *a posteriori* probability of the sequence of modes p^s_k, is obtained from $\tilde{p}(\hat{m}^s_{k-d:k})$ using Bayes' theorem. In the sixth step, the estimates provided by the filter bank and the *a posteriori* probability of the hypotheses are combined. In the seventh step, the estimation of the transition probability matrix, $\hat{\Pi}_k$, is obtained using the Quasi-Bayesian algorithm. The MHMF algorithm is described as follows.

Algorithm 2.3.3. *The multiple-hypothesis mixing filter [Santana, P. H. R. Q. A., Menegaz, H. M., Borges G. A., Ishihara, J. Y., 2010]*

Initialize the initial buffer of hypotheses $\mathcal{H}_0$, the initial hypothesis probability p^0_k, the inicial transition probability matrix Π_0, and set the tuning parameters as follows: the merging depth d and the pruning threshold ε and the maximum number of hypotheses $N_{\max}$.

1. *Hypotheses probability prediction. For all $H^r_{k-1} \in \mathcal{H}_{k-1}$, generate all sequences $\hat{m}^s_{k-d:k}$ such that $\hat{m}^s_{k-d:k-1} = \hat{m}^r_{k-d:k-1}$, $\hat{m}^s_k = j$ and $s = (r-1)M + j$, $j = 1, \dots, M$; store the new hypotheses in a buffer $\mathcal{H}_k = \{(\hat{m}^s_{k-d:k}, \hat{x}^r_{k-1}, P^{xx,r}_{k-1})\}$ and update their* a priori *probability to*

$$\tilde{p}^s_k = \pi_{j|i} p^r_{k-1}, \tag{2.105}$$

   where $i = \hat{m}^r_{k-1}$.

2. *Hypotheses pruning. Discard the hypotheses H^s_k from buffer $\mathcal{H}_k$ with $\tilde{p}^s_k < \varepsilon$. Normalize the probabilities of the remaining hypotheses.*

3. *Mixing of hypotheses. Use the state vector estimates $\hat{x}^r_{k-1}$, the covariance matrix $P^{xx,r}_{k-1}$, the probability of the modes estimate, γ^r_{k-1}, and the probability of the mode sequence, $\tilde{p}^s_k$, to obtain the estimates of the state vector, $\hat{x}^s_{k-1}$, and the covariance matrix $P^{xx,s}_{k-1}$ for the s filter, $s = 1, \ldots, M$. For such, the probability of mixing the hypotheses, w^r_{k-1}, is given by*

$$w^r_{k-1} = \frac{\hat{\pi}_{j|i}\gamma^r_{k-1}}{c_s}, \tag{2.106}$$

$$c_s = \sum_{s=1}^{R} p(m_{1:k} = j|H^s_k, y_{1:k-1})\tilde{p}(\hat{m}^s_{k-d:k}), \tag{2.107}$$

where $r = 1, \ldots, M$ e $s = 1, \ldots, R$. The mixture of hypotheses is given by

$$\hat{x}^s_{k-1} = \sum_{r=1}^{M} w^r_{k-1}\tilde{x}^r_{k-1}, \tag{2.108}$$

$$\tilde{x}^r_{k-1} = \sum_{l=1}^{n(\mathcal{H}^e_{k-1})} \hat{x}^l_{k-1} p(H^l_{k-1}|m_{1:k-1} = i, y_{1:k-1}), \tag{2.109}$$

$$\triangle^r_{k-1} = \sum_{l=1}^{n(\mathcal{H}^e_{k-1})} P^{xx,l}_{k-1} p(H^l_{k-1}|m_{1:k-1} = i, y_{1:k-1}), \tag{2.110}$$

$$P^{xx,s}_{k-1} = \sum_{r=1}^{M} w^r_{k-1}\left[\triangle^r_{k-1} + \left(\tilde{x}^r_{k-1} - \hat{x}^s_{k-1}\right)\left(\tilde{x}^r_{k-1} - \hat{x}^s_{k-1}\right)^T\right], \tag{2.111}$$

where $n(\mathcal{H}^e_{k-1})$ is the number of hypotheses at time $k-1$, e $p(H^l_{k-1}|m_{1:k-1} = i, y_{1:k-1})$ is given by (2.104).

4. *Analog state prediction and correction. For each hypothesis $H^s_k \in \mathcal{H}_k$, run a filter to estimate the state vector $\hat{x}^s_k$, the covariance matrix $P^{xx,s}_k$, the innovation η^s_k, and the innovation covariance matrix $P^{yy,s}_k$ and make $H^s_k = (\hat{m}^s_{k-d:k}, \hat{x}^s_k, P^{xx,s}_k)$. Each filter is given by*

$$\left\{\hat{x}^s_k, P^{xx,s}_k, \eta^s_k, P^{yy,s}_k\right\} = Filter\left(\hat{x}^s_{k-1}, P^{xx,s}_{k-1}, y^s_k, f_s, h_s, Q_s, R_s\right), \tag{2.112}$$

Here each "Filter" uses one of the M possible pairs of stochastic models f_s, h_s, Q_s and R_s.

5. *Hypotheses probability correction. Obtain the a posteriori probability p^s_k of hypothesis H^s_k using the Bayes' theorem:*

$$p^s_k = \frac{\rho_s(\eta^s_k)\tilde{p}^s_k}{\sum_{H^j_k \in \mathcal{H}_k} \rho_j(\eta^j_k)\tilde{p}^j_k}, \tag{2.113}$$

where $\rho_s(\eta_k)$ is the likelihood function of innovations, given (at least approximately, if non-Gaussian) by

$$\rho_s(\eta_k^s) = \frac{1}{\sqrt{(2\pi)^m |P_k^{yy,s}|}} exp\left[-\frac{1}{2}(\eta_k^s)^T (P_k^{yy,s})^{-1}\eta_k^s\right] . \tag{2.114}$$

6. *Combining the estimates. Combine the estimates from the buffer of hypothesis to obtain the mean and covariance of the corresponding Gaussian mixture*

$$\hat{x}_k = \sum_{H_k^s \in \mathcal{H}_k} p_k^s \hat{x}_k^s, \tag{2.115}$$

$$P_k^{xx} = \sum_{H_k^s \in \mathcal{H}_k} p_k^s \left[P_k^{xx,s} + (\hat{x}_k^s - \hat{x}_k)(\hat{x}_k^s - \hat{x}_k)^T\right]. \tag{2.116}$$

and return $\hat{x}_k$ and P_k^{xx} as the state estimate and its corresponding covariance at time k and the estimate $\hat{m}$ as being the most likely mode a posteriori.

7. *Transition probability matrix correction. Use the quasi-Bayesian algorithm to estimate the transition probability matrix, $\hat{\Pi}_k$, from a sequence of noisy measurements [Jilkov, V., Li, X., 2004]. We assume that the TPM whose row is given by $row_i(\Pi_k)$, where row means the i-th row of Π_k, can be modeled by a Dirichlet distribution. In this algorithm, we assume that the probability of the mode is given by*

$$\mu_{i,k-1} \triangleq p(m_{1:k} = j|\hat{\Pi}_{k-1}, y_{1:k-1}), \tag{2.117}$$

$$\theta_{k-1} = [\mu_{1,k} \dots \mu_{M,k}]^T, \tag{2.118}$$

where $i, j = 1, \dots, M$. The likelihood function of innovations is given by

$$\lambda_{i,k} \triangleq p(y_k|m_{1:k} = j, \hat{\Pi}_{k-1}, y_{1:k-1}), \tag{2.119}$$

$$\Lambda_k = [\lambda_{1,k} \dots \lambda_{M,k}]^T, \tag{2.120}$$

$$\eta_{i,k} = \frac{\mu_{i,k-1}}{\theta_{k-1}^T \hat{\Pi}_{k-1} \Lambda_k}. \tag{2.121}$$

We assume that the Dirichlet distribution is defined by

$$\alpha_{i,k} = \left[\alpha_{1|i,k} \dots \alpha_{j|i,k}\right], \tag{2.122}$$

where j is the j-th column of Π_k, $i, j = 1, \ldots, M$. *In the initialization step, we assume the following conditions*

$$\alpha_{i,0} = \left[\alpha_{1|i,0} \ldots \alpha_{M|i,0}\right], \tag{2.123}$$

$$\gamma_{i,0} = \sum_{j=1}^{M} \alpha_{j|i,0}, \tag{2.124}$$

$$\hat{\pi}_{i,0} = \frac{1}{\gamma_{i,0}} \alpha_{i,0}, \tag{2.125}$$

if the a priori information of the TPM is unknown, we can choose $\alpha_{i,0} = [1, \ldots 1]$, *as a consequence,* $\hat{\pi}_{i,0} = \frac{1}{M}$. *Finally, the recursive estimate of the transition probability matrix,* $\hat{\Pi}_k$, *is given by*

$$g_{i,k} = \sum_{j=1}^{M} 1 + \eta_{i,k} \left[\lambda_{j,k} - \hat{\pi}_{i,k-1}^{T} \Lambda_k,\right], \quad i = 1, \ldots, M, \tag{2.126}$$

$$\alpha_{i,k} = \sum_{j=1}^{M} \alpha_{j,k-1} + \frac{\alpha_{j,k-1} g_{i,k}}{\sum_{l=1}^{M} \alpha_{l,k-1} g_{l,k}}, \tag{2.127}$$

$$\hat{\pi}_{i,k} = \sum_{j=1}^{M} \frac{1}{k + \gamma_{i,0}} \alpha_{j,k}. \tag{2.128}$$

Increment k and return to step 1.

□

2.3.4 Particle Filters

The particle filter (PF) approach has been used to estimate states of hybrid systems. Particle filtering methods approximate the joint *a posteriori* PDF of the hybrid system using sampled trajectories [Boers, Y., Driessen, H, 2000; Doucet, A., Gordon, N., Krishnamurthy, V., 2001; Koutsoukos, L., Williams, B., 2003]. The PF employs a set of particles with corresponding weights to provide estimates for both analog and digital states. One of the limitations of particle filtering is the choice of the candidate distribution, $q(x_k|x_{k-1}^i, y_k)$, $i = 1, \ldots, N$, that represents the true joint *a posteriori* PDF.

In the hybrid stochastic filtering, the PF approximates a joint *a posteriori* PDF of x_k and m_k, $\rho(x_k, m_k \mid y_{1:k})$. In this case, the PF provides the estimates of the analog and digital states. On the other hand, to deal with the non-hybrid stochastic filtering, the

PF aims to generate an approximation of the *a posteriori* PDF of x_k, $\rho\left(x_k|y_{1:k}\right)$. In this case, the PF only provides the analog state estimate. A brief review on particle filtering in the non-hybrid context is presented in Section 2.1.3. We recommend to read section 2.1.3 before reading this Section.

Among the various particle filtering algorithms, we employ the bootstrap particle filter with systematic resampling, for being the most feasible algorithm compared to other PF approaches due to its simplicity and performance [Candy, J.V., 2009]. The bootstrap particle filter employs the state transition PDF, $\rho\left(x_k|x^i_{k-1}\right)$, $i = 1, \ldots, N$, as a candidate distribution. In the filtering process, this approach uses a high number of particles, implying a high computational cost [Ristic, B., Arulampalam, S., Gordon, N., 2004; Gustafsson, F., 2010].

The particulate filter can be considered as a hybrid stochastic filtering algorithm, in which both analog and digital state estimates are obtained [Boers, Y., Driessen, H, 2000; Doucet, A., Gordon, N., Krishnamurthy, V., 2001; Koutsoukos, L., Williams, B., 2003]. Alternatively, some particle filtering methods combine the particle filter and the interacting multiple-model estimator (IMMPF) to estimate the analogue and digital state, respectively [Boers, Y., Driessen, H., 2003; Hendeby, G., Karlsson, R., Gustafsson, F., 2010; Kawamoto, K., 2010; Wang, X., Xu, M., Wang, H., Wu, Y., Shi, H., 2012]. Thus, the IMMPF approach treats the two terms of (1.4) as follows: the analog state estimate is obtained from a filter bank composed of particle filters, whereas the digital state estimate is obtained using the Bayes Theorem, as described in Section 2.2. In the Rao-Blackwellized particle filter case, the first term of (1.4) is obtained employing the Kalman filter bank, whereas the second term is obtained using the particle filter [Hendeby, G., Karlsson, R., Gustafsson, F., 2010]. Details of the equations of the IMMPF and Rao-Blackwellized algorithms are not presented in this work but can be found in [Wang, X., Xu, M., Wang, H., Wu, Y., Shi, H., 2012; Hendeby, G., Karlsson, R., Gustafsson, F., 2010].

The PF approach proposed by [Doucet, A., Gordon, N., Krishnamurthy, V., 2001] divides the hybrid stochastic filtering problem into four steps: the *particle generation* step, the *weight calculation* step and the *particle resampling* step and the *combining the* *estimates* step, as shown in Figure 2.7. At this point, we assume that a particle, x_k, and the particle mode, m_k, are defined by $z_k \triangleq \left[x_k^T m_k^T\right]^T$. Thus, the set of particles with their respective modes and weights is given by $\{z_k^i, w_k^i\}_{i=1}^N$, where N is the number of particles.

Figure 2.7 summarizes the PF algorithm in four recursive steps. In the first step, we generate a set of particles with their respective modes from the candidate distribution, $\{z_{k-1}^i\}_{i=1}^N$. In this step, we assume that the particle weights are uniform, $\left\{w_{k-1}^i = \frac{1}{N}\right\}_{i=1}^N$. In the second step, the information of a new measurement, y_k, is incorporated to calculate the importance weights of each particle, $\{w_{k-1}^i\}_{i=1}^N$. In the third step, we performed the resampling process from random measures $\{z_{k-1}^i, w_{k-1}^i\}_{i=1}^N$ to random measures $\{z_k^i, w_k^i\}_{i=1}^N$ when the effective number of particles, $\hat{N}_{\text{eff}}$, is less than the threshold chosen by the user, $N_{\text{threshold}}$. In this step, the particles with higher probability are preserved. In the fourth step, the set of particles and their respective weights, $\{z_k^i, w_k^i\}_{i=1}^N$, are combined to obtain the analog state estimate, $\hat{x}_k$, and the corresponding covariance P_k^{xx} and the mode estimate $\hat{m}$ is obtained as being the most likely *a posteriori* mode. The PF algorithm is described as follows.

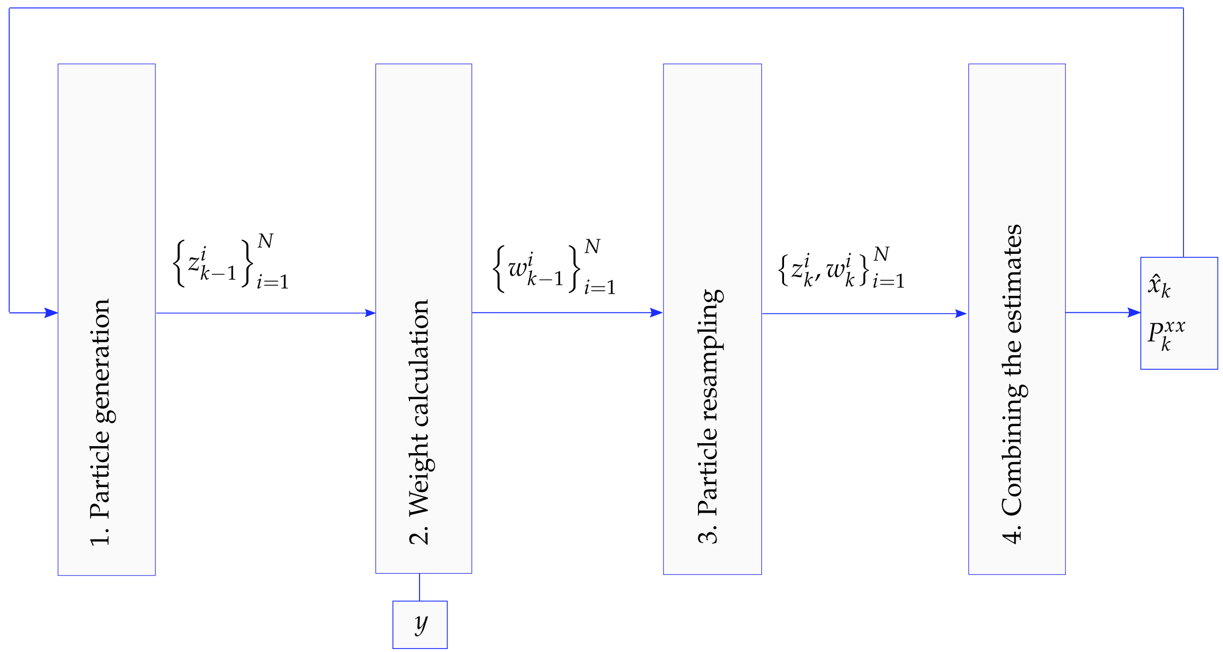

Figure 2.7: Diagram of the steps of the PF algorithm. The PF comprises four steps: in the *particle generation* step, a set of particles with their respective modes is generated from the candidate distribution; in the *weight calculation* step, the information of a new measurement is incorporated to calculate the importance weight of each particle; in the *particle resampling* step, the resampling process is performed and in the *combining the estimates* step, the set of particles and their respective weights are combined to obtain the analog state estimate and the corresponding covariance.

Algorithm 2.3.4. *Particle filters [Doucet, A., Gordon, N., Krishnamurthy, V., 2001]*
Initialize the set of particles with their respective modes and associated weights ,$\{z_0^i, w_0^i\}_{i=1}^N$ *and set the tuning parameters as follows: the number of particles N and the the resampling threshold* $N_{\text{threshold}}$.

1. *Particle generation. Generate a set of particles with their respective modes,* $\{z_{k-1}^i\}_{i=1}^N$, *from the candidate distribution,* $q(x_k|x_{k-1}^i, y_k)$. *In particular, the bootstrap particle filter employs the state transition PDF,* $\rho\left(x_k|x_{k-1}^i\right)$, *as the distribution candidate. This step is given by*

$$z_{k-1}^i \sim \rho\left(x_k|x_{k-1}^i\right), \tag{2.129}$$

where $\rho\left(x_k|x_{k-1}^i\right)$ *is obtained using the process model (1.1). Note that, the particle weights are uniform,* $\left\{w_{k-1}^i = \frac{1}{N}\right\}_{i=1}^N$.

2. *Weight calculation. Incorporate the information of a new measurement,* y_k, *to calculate the importance weights of each particle. In the bootstrap particle filter case, the weights are obtained using the likelihood function,* $p(y_k|x_k^i)$ *and then normalize the particle weights. This process is given by*

$$\tilde{w}_{k-1}^i = w_{k-1}^i p(y_k|x_k^i), \tag{2.130}$$

$$w_{k-1}^i = \frac{\tilde{w}_{k-1}^i}{\sum_{i=1}^N \tilde{w}_{k-1}^i}, \tag{2.131}$$

3. *Particle resampling. Perform the resampling process from random measurements* $\{z_{k-1}^i, w_{k-1}^i\}_{i=1}^N$ *to random measurements* $\{z_k^i, w_k^i\}_{i=1}^N$ *when the effective number of particles,* $\hat{N}_{eff}$, *is less than the threshold chosen by the user,* $N_{threshold}$. *For such, we employ the systematic resampling method. This method selects the most important particles (high probability) to generate the random measurements* $\{z_k^i, w_k^i\}_{i=1}^N$ *with uniform weights.*

   First, a uniform distribution is generated, $U\left(0, \frac{1}{N}\right)$. *Then, an integer less than 1 is selected. This number corresponds to the reference value of the distribution. From this reference the distribution is divided into intervals uniformly. The resampling process is performed until the effective number of particles,* N_{eff}, *is less than the resampling threshold,* N_{limiar}. *The details of the equations in the systematic resampling method are* *not presented in this work but can be found in [Arulampalam, M. S., Maskell, S., Gordon, N., Clapp, T., 2002].*

4. *Combining the estimates. Combine the set of particles and their respective weights, $\{z_k^i, w_k^i\}_{i=1}^N$, to obtain the analog state estimate, $\hat{x}_k$, and the associated covariance matrix P_k^{xx}. These estimates are given by*

$$\hat{x}_k = \sum_{i=1}^{N} w_k^i x_k^i, \tag{2.132}$$

$$P_k^{xx} = \sum_{i=1}^{N} w_k^i \left(x_k^i - \hat{x}_k\right)\left(x_k^i - \hat{x}_k\right)^T. \tag{2.133}$$

Increment k and return to step 1.

□

2.3.5 Constrained Hybrid Methods

The hybrid stochastic filtering problem is to provide estimates for both analog and digital states from a sequence of noisy sample measurements and such hybrid model. However, additional information about the system in the form of state constraints may be useful for improving the state estimates.

Existing methods for constrained state estimation in hybrid systems are based on the multiple-model or Monte Carlo. Some methods based on MM approaches are presented in [Pannetier, B., Benameur, K., Nimier, V., Rombaut, M., 2005; Zhang, M., Knedlik, S., Loffeld, O., 2008; Mann, G., Hwang, I., 2013; Kwon, C., Hwang, I., 2016] Alternatively, particle filtering method is presented in [Kravaritis, G.., Mulgrew, B., 2008].

Some investigations address the problem of equality-constrained state estimation in hybrid systems. For instance, in the study developed by [Mann, G., Hwang, I., 2013], the constrained innovation hybrid estimator (CIHE) is proposed to address the problem of equality-constrained state estimation for linear hybrid systems using the IMM approach. The proposed approach considers that the equality constraint vary with the operating mode. The CIHE algorithm is described as follows.

The CIHE is a six-step algorithm: the *mode probability prediction* step, the *mixing of estimates* step, the *analog state prediction and correction* step, the *mode probability correction* step, the *combining the estimates* step and the *mode transition probability correction* step, as shown in Figure 2.8. In this algorithm, the two terms of the right-hand side of (1.4) are treated as follows. The first term, $\rho(x_k \mid m_{1:k}, y_{1:k})$, is addressed in the *estimate mixing* and the *filter bank* steps, whereas, the second term, $\rho(m_{1:k} \mid y_{1:k})$, is processed in the *mode probability prediction* step and updated in the *mode probability correction* step. We assume that the analog state estimates are obtained from the weighted combination of the recursive estimates of the filter bank, whereas, the digital state estimate is obtained as being the most likely *a posteriori* mode.

Figure 2.8 summarizes the CIHE algorithm in six recursive steps. In the first step of the CIHE, the *a priori* probability of the mode, $w^s_{k-1}, s = 1, \ldots, M$, are obtained. In the second step, the previous state vector estimates $\hat{x}^{rc}_{k-1}$, the matrix covariance $P^{xx,rc}_{k-1}$ and the mode probability predictions $w^r_{k-1}, r = 1, \ldots, M$, are mixed to generate new initial values for each filter, the state vector estimates $\hat{x}^{sc}_{k-1}$ and the corresponding covariance $P^{xx,sc}_{k-1}$. In the third step, the filter bank provides the estimate of the state vector $\hat{x}^{sc}_k$, the covariance matrix $P^{xx,sc}_k$, the innovation η^s_k and the innovation covariance matrix $P^{yy,s}_k$. In the fourth step, the *a posteriori* probability of the mode γ^s_k, is obtained using Bayes' theorem. In the fifth step, the estimates provided by the filter bank and the *a posteriori* probability of the mode are combined to obtain the estimate of the state vector $\hat{x}^c_k$, and associated covariance P^{xxc}_k at time k. In the sixth step, the mode transition probabilities $\pi_{j|i}$, are obtained using the stochastic linear guard conditions. The CIHE algorithm is described as follows.

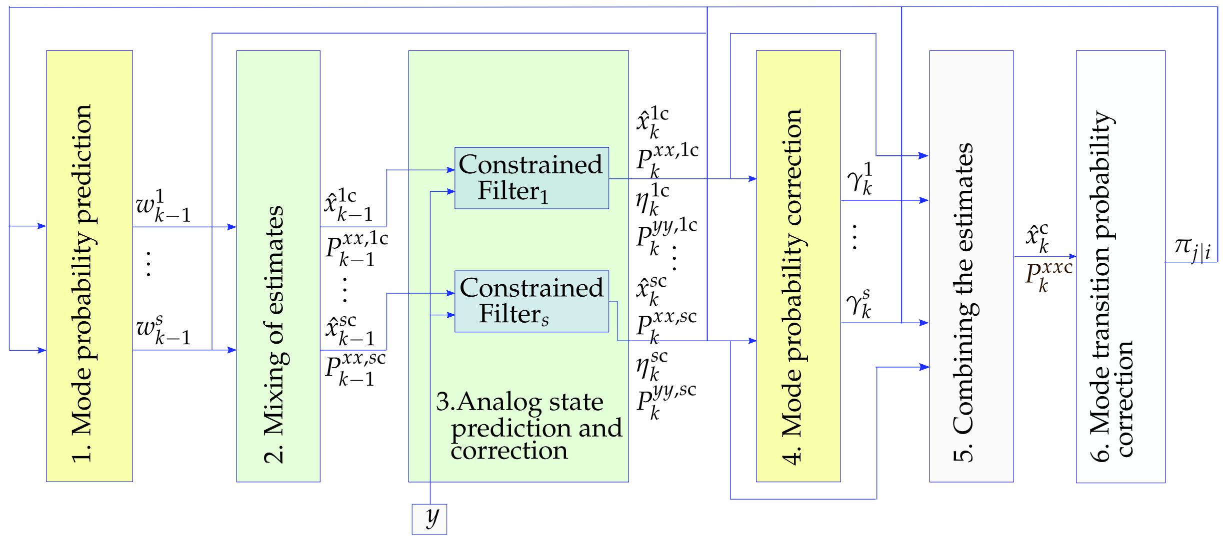

Figure 2.8: Diagram of the steps of the CIEH algorithm. The CIHE algorithm comprises six steps: in the *mode probability prediction* step, the *a priori* probabilities of the modes are obtained; in the *mixing of estimates* step, the estimates of the state vector and associated covariance matrix for each filter are obtained; in the *analog state prediction and correction* step, the M filters are employed to estimate the state vector and the associated covariance matrix; in the *mode probability correction* step, the *a posteriori* probabilities of the modes are obtained using Bayes' theorem; in the *combining the estimates* step, the M recursive estimates of each filter and their respective *a posteriori* probabilities are combined and the *mode transition probability correction* step, the stochastic linear guard conditions are employed to estimate the mode transition probabilities. Note that, the first and second term of (1.4) are respectively treated in the green and yellow steps in the IMM algorithm.

Algorithm 2.3.5. *The CIHE filter with state equality constraint [Mann, G., Hwang, I., 2013]* *Initialize each sth filter with the state vector estimate $\hat{x}_0^s$, the covariance matrix $P_0^{xx,s}$, the mode probability $\gamma_0^s, s = 1, \dots, M$ and the initial transition probability matrix, Π_0.*

1. *Mode probability prediction. Obtain the a priori probability of the modes $w_{k-1}^s, s = 1, \dots, M$, and obtain the mixing probabilities $\mu_{k-1}^{s|r}$,*

$$\mu_{k-1}^{s|r} = \frac{\pi_{s|r}\gamma_{k-1}^r}{w_{k-1}^s}, \tag{2.134}$$

where $\pi_{s|r}$ is given by (1.2) and γ_{k-1}^r is the r-th a posteriori probability of the mode, $r = 1, \dots, M$. The a priori probability of the modes is given by

$$w_{k-1}^s \triangleq \sum_{r=1}^{M} \pi_{s|r}\gamma_{k-1}^r. \tag{2.135}$$

2. *Mixing of estimates. Obtain the mixed estimates $\bar{x}_{k-1}^{sc}$ and $\bar{P}_{k-1}^{xx,sc}$, $s = 1, \dots, M$ from the interaction of the M filters by mixing the previous state vector estimates $\hat{x}_{k-1}^{rc}$ and covariance matrix $P_{k-1}^{xx,rc}$ with the respective mixing probabilities $\mu_{k-1}^{s|r}$,*

$$\bar{x}_{k-1}^{sc} = \sum_{r=1}^{M} \mu_{k-1}^{s|r}\hat{x}_{k-1}^{rc}, \tag{2.136}$$

$$\bar{P}_{k-1}^{xx,sc} = \sum_{r=1}^{M} \mu_{k-1}^{s|r}\left[P_{k-1}^{xx,rc} + \left(\hat{x}_{k-1}^{rc} - \bar{x}_{k-1}^{sc}\right)\left(\hat{x}_{k-1}^{rc} - \bar{x}_{k-1}^{sc}\right)^T\right]. \tag{2.137}$$

3. *Analog state prediction and correction. For each estimate $\bar{x}_{k-1}^{sc}$, run a filter to estimate the state vector $\hat{x}_k^{sc}$, the covariance matrix $P_k^{xx,sc}$, the innovation η_k^s, and the innovation covariance matrix $P_k^{yy,s}$. Each s-th filter is given by*

$$\left\{\hat{x}_k^{sc}, P_k^{xx,sc}, \eta_k^s, P_k^{yy,s}\right\} = \textit{Constrained Filter}\left(\hat{x}_{k-1}^{sc}, P_{k-1}^{xx,sc}, y_k^s, f_s, h_s, Q_s, R_s\right), \tag{2.138}$$

Here each "Constrained Filter" uses one of the M possible pairs of stochastic models f_s, h_s, Q_s and R_s. Note that in the CIHE, the filter bank consists of constrained innovation Kalman filter (CIKF). The CIKF enforces the equality constraint (4.5) on state estimate during the forecast step, see Section 2.1.4.

4. *Mode probability correction. Obtain the a posteriori probability γ_k^s of each estimate using the Bayes' theorem as follows:*

$$\gamma_k^s = \frac{\rho_s(\eta_k^s)w_{k-1}^s}{\sum_{r=1}^{M}\rho_r(\eta_k^r)w_{k-1}^r}, \tag{2.139}$$

where w^s_{k-1}, *is given by (2.135) and* $\rho_s(\eta^s_k)$ *is the likelihood function of innovations, given (at least approximately, if non-Gaussian) by*

$$\rho_s(\eta^s_k) = \frac{1}{\sqrt{(2\pi)^m |P^{yy,s}_k|}} \, exp\left[-\frac{1}{2}(\eta^s_k)^T (P^{yy,s}_k)^{-1}\eta^s_k\right] \quad . \tag{2.140}$$

5. *Combining the estimates. Combine the M estimates to obtain the state estimate* $\hat{x}^c_k$ *and its corresponding covariance* P^{xxc}_k *at time k*

$$\hat{x}^c_k = \sum_{s=1}^{M} \gamma^s_k \hat{x}^{sc}_k, \tag{2.141}$$

$$P^{xxc}_k = \sum_{s=1}^{M} \gamma^s_k \left[P^{xx,sc}_k + (\hat{x}^{sc}_k - \hat{x}^c_k)\,(\hat{x}^{sc}_k - \hat{x}^c_k)^T\right]. \tag{2.142}$$

and obtain the estimate $\hat{m}$ *as being the most likely mode a posteriori.*

6. *Mode transition probability correction. Use the stochastic linear guard conditions to estimate the mode transition probabilities,* $\pi_{s|r}$, *[Seah, C. E., Hwang, I., 2009]. The evolution of the mode is a Markov chain described by a analog state dependent mode transition matrix*

$$\Pi_{x_k} = \pi_{s|r,x_k}, \tag{2.143}$$

where $\pi_{s|r,x_k}$ *is the mode transition probability conditioned on the analog state* x_k *given by*

$$\pi_{s|r,x_k} = Pr\left\{\begin{bmatrix} x^T_k & \theta^T \end{bmatrix}^T \in G_{s,r} | m_{k-1} = r\right\}. \tag{2.144}$$

where $\theta \in \Theta$ *is a random variable that models uncertainties in* $G_{s,r}$ *represents the guard conditions associated with the mode transition from mode* $m_{k-1} = r$ *to mode* $m_k = s$, *which partitions out the analog state. The stochastic linear guard conditions are given by*

$$G_{s,r} = \left\{\begin{bmatrix} x_k \\ \theta \end{bmatrix} \mid x_k \in X, \theta \in \Theta, L_{s|r}\begin{bmatrix} x_k \\ \theta \end{bmatrix} \leqslant 0\right\} \tag{2.145}$$

where $L_{s|r} \in \mathbb{R}^{M\times n}$ *is the matrix defining the inequality and* θ *is a Gaussian random vector with mean* $\bar{\theta}$ *and covariance* Σ_θ.

From (2.145), the mode transition probability (2.144) can be rewritten as

$$\pi_{s|r,x_k} = \Phi\left(0; L_{s|r}\begin{bmatrix} \hat{x}^{rc}_{k-1} \\ \theta_{s|r} \end{bmatrix}; L_{j|i}\begin{bmatrix} P^{xx,rc}_{k-1} & 0 \\ 0 & \Sigma_{\theta_{s|r}} \end{bmatrix} L^T_{s|r}\right) \tag{2.146}$$

where $\Phi(\mu, \bar{\mu}, \Sigma)$ *denotes the Gaussian cumulative density function with mean* $\bar{\mu}$ *and covariance* Σ*. Increment* k *and return to step* 1.

□

In a recent work, [Kwon, C., Hwang, I., 2016] presents a simplified version of constrained stochastic filtering algorithm for linear hybrid systems, in which the road map information employed by the equality constraint is useful for improving the state estimation process. Likewise, [Mann, G., Hwang, I., 2012] considers the constrained stochastic filtering problem for linear hybrid systems. This approach addresses aircraft taxiway conformance-monitoring problem.

2.4 Concluding Remarks

The hybrid stochastic filtering problem is to provide estimates for both analog and digital states from a sequence of noisy sample measurements and such hybrid model. The solution to this problem is given by the joint *a posteriori* PDF of the analog state, x_k, and digital, m_k. This joint *a posteriori* PDF can be divided into two parts, the first term on the right-hand side of (1.4) may be computed by a classical nonlinear filter given that the mode sequence is known. The second term is computed applying Bayes' rule to the result of the nonlinear filter.

In this chapter, we present approximate solutions to the hybrid stochastic filtering problem. In the particular case of Markov jump linear systems where dynamics is linear and mode transition is Markovian and independent of analog states. Thus, we present some hybrid stochastic filtering methods such as the IMM approach, the M3H approach, PF methods and the combination of some of these approaches.

The IMM approach is the main approach that addresses the problem of estimation of states of hybrid systems. The IMM algorithm mixes the previous analog state estimates to generate new initial values for each filter in the next time step, to circumvent the problem of exponential growth of possible hypotheses. However, this method control multiple hybrid estimates over time and require the running of a filter for each mode.

Alternatively, the M^3H algorithm keeps a limited buffer of hypotheses by means of merging and discarding hypotheses to address the exponential growth problem. In the M^3H filter, merging occurs when two or more hypotheses share the same sequence of modes for the previous d time steps. In addition, hypotheses are discarded whenever their likelihood is below a given threshold. The hypotheses with the same sequence of modes in the last d steps are merged. The merging step employs only information from the digital state.

The MHMF filter proposes the generalization of the *mixing of estimates* step of the IMM algorithm considering the merging depth $d > 1$, and the correction of the transition probability matrix. This approach addresses the tracking of multiple hypotheses between two instants of mixture of estimates. In this approach, the correction of TPM avoids inaccurate estimates of the analog and digital state vector.

On the other hand, particle filtering methods approximate the joint *a posteriori* PDF of the hybrid system using sampled trajectories. The PF employ a set of particles with corresponding weights to provide estimates for both analog and digital states. However, the PF has limitations such as the use of high number of particles implying a high computational cost and the choice of the candidate distribution causing an imprecise approximation of the joint *a posteriori* PDF. Some particle filtering methods combine the particle filter with other approaches to address the hybrid stochastic filtering problem such as the IMMPF approach and the Rao-Blackwellized filter.

Table 2.1: Hybrid stochastic filtering methods. We present the approaches employed to obtain estimates for both the analog state, $\hat{x}_k$, and digital state, $\hat{m}_k$, in each of the methods. To estimate the analog state is employed a Kalman filter bank (KFB), the particle filter bank (PFB), the particle filter (PF) or Constrained Kalman filter bank (CKFB), whereas to estimate the digital state is used the Bayes' theorem (BT) or the particle filter.

Methods	$\hat{x}_k$				$\hat{m}_k$	
	KFB	PFB	PF	CKFB	TB	PF
IMM	$\star$				$\star$	
M3H	$\star$				$\star$	
MHMF	$\star$				$\star$	
PF			$\star$			$\star$
IMMPF		$\star$			$\star$	
Rao-Blackwellized PF	$\star$					$\star$
CIHE				$\star$	$\star$	

Constrained state estimation methods for hybrid systems incorporate additional information about the system in the form of state constraints to improve the state estimates. The CIHE algorithm addresses the problem of equality-constrained state estimation for linear hybrid systems, whose equality constraints vary with the operating mode. This approach employs a filter bank based on the IMM estimator. The filter bank uses constrained Kalman filters.

The hybrid stochastic filtering methods employ different approaches to obtain estimates for both the analog state, $\hat{x}_k$, and digital state, $\hat{m}_k$. Table 2.1 shows how each of these methods provides estimates of the analog and digital state. In these methods, to estimate the analog state is employed a filter bank composed of Kalman filters or particle filters, whereas to estimate the digital state is used the Bayes' theorem or particle filtering.

Chapter 3

The Multiple-Model and Multiple Hypothesis Algorithm with Gaussian Mixture Reduction

The present chapter adresses the problem of state estimation for Markov jump systems. First, Section 3.1 defines this problem. In Section 3.2, we present a modified version of the multiple models and multiple hypotheses (M^3H) algorithm to suboptimally solve the problem of state estimation for Markov jump nonlinear systems. Then, we investigate the use of Gaussian mixture reduction methods as an alternative for the merging step of the M^3H algorithm in Section 3.3. Finally, in Section 3.4, the proposed approach is compared to classical M^3H by means of one example. The contribution presented in this chapter was published in [Eras-Herrera, W. Y., Mesquita, A. R., Teixeira, B. O. S., 2017].

3.1 Problem Statement

We consider discrete-time hybrid stochastic system given by (1.1)-(1.3), whose equations are repeated here for convenience, that is,

$$\begin{aligned} x_k &= f_{m_k}(x_{k-1}, u_{k-1}, w_{k-1}, k-1), \\ \pi_{s|r} &= \Pr\{m_k = s | m_{k-1} = r\}, \\ y_k &= h_{m_k}(x_k, v_k, k). \end{aligned} \tag{3.1}$$

We assume that the transition probability matrix (TPM) $\Pi \in \mathbb{R}^{M\times M}$, whose elements are given by $\pi_{s|r}$, is known. Mode transitions at time k are assumed to be independent of $m_{0:k-1}$ and $x_{0:k}$ given the current mode m_k, i.e., transition probabilities depend only on the current mode and not on the process history or on the analog state. A hybrid system with such properties is known in the literature as a discrete-time Markov jump system (MJS).

The solution to this problem may be obtained through the joint *a posteriori* PDF (1.4), of the analog and digital states. In the case of linear dynamics and linear observations, the first term of (1.4) is simply the density of a Gaussian PDF. Since the second term does not depend on x_k, we conclude that the marginal posterior density of x_k is a Gaussian mixture. Unfortunately, the number of components of this mixture is as large as M^k.

3.2 M^3H using Gaussian Mixture Reduction

We investigate the use of Gaussian reduction methods as an alternative for the merging step of the M^3H algorithm. The Gaussian mixture reduction by clustering (GMRC) approach [Schieferdecker, D., Huber, M. F., 2009] is used here to reduce the number of hypotheses propagated from the *hypothesis probability prediction* step of the M^3H. The proposed approach is here called M^3H using Gaussian mixture reduction (M^3HR). In our method, information from both the analog, $\hat{x}_k$, and digital, $\hat{m}_k$, estimates at time k are employed to define a metric to merge and eliminate hypotheses. This comes as an extension of the original M^{3}H, that only uses the information of the mode sequence, $\hat{m}_{k-d:k}$, from time $k-d$ to time k to merge the hypotheses.

We now redefine hypotheses as components of a Gaussian mixture (where the mode history no longer plays a role): $I_k^s \triangleq (\tilde{p}_k^s, \hat{m}_k^s, \hat{x}_k^s, P_k^{xx,s})$ comprised, respectively, of a hypothesis probability, an operating mode, a mean vector and a covariance matrix for the analog state PDF.

Figure 3.1 shows that the proposed M^{3}H filter has the same form of the original M^{3}H except for the replacement of the merging step by the GMRC algorith. The M^{3}HR algorithm is composed of six recursive steps. In the first step of M^{3}HR, the *a priori* probabilities $\tilde{p}_k^s$ are obtained. In the second step, the GMRC approach is used here to reduce the number of hypotheses. In this work, we propose an alternative method for the *hypotheses merging* step (orange) of the M^{3}H, see Figure 3.1. In the third step, the hypotheses with probability lower than the elimination threshold given by $0 \leq \varepsilon \leq 1$ are eliminated. The threshold ε is a tuning parameter to be chosen by the user. In the fourth step, the filter bank provides the estimate of the state vector $\hat{x}_k^s$, the covariance matrix $P_k^{xx,s}$, the innovation η_k^s, and the innovation covariance matrix $P_k^{yy,s}$, associated to each one of the hypothesis $I_k^s \in \mathcal{I}_k$. In the fifth step, the *a posteriori* probability p_k^s, is obtained from $\tilde{p}_k^s$ using Bayes' theorem. In the sixth step, the estimates provided by the filter bank and the *a posteriori* probability of the hypotheses are combined. Note that the two terms of the right-hand side of (1.4) are treated as follows. The first term, $\rho(x_k \mid m_{1:k}, y_{1:k})$, is addressed in the *prediction and correction of estimates* step, while the second term, $\rho(m_{1:k} \mid y_{1:k})$, is processed in the *hypothesis probability prediction* step and updated in the *hypothesis probability correction* step.

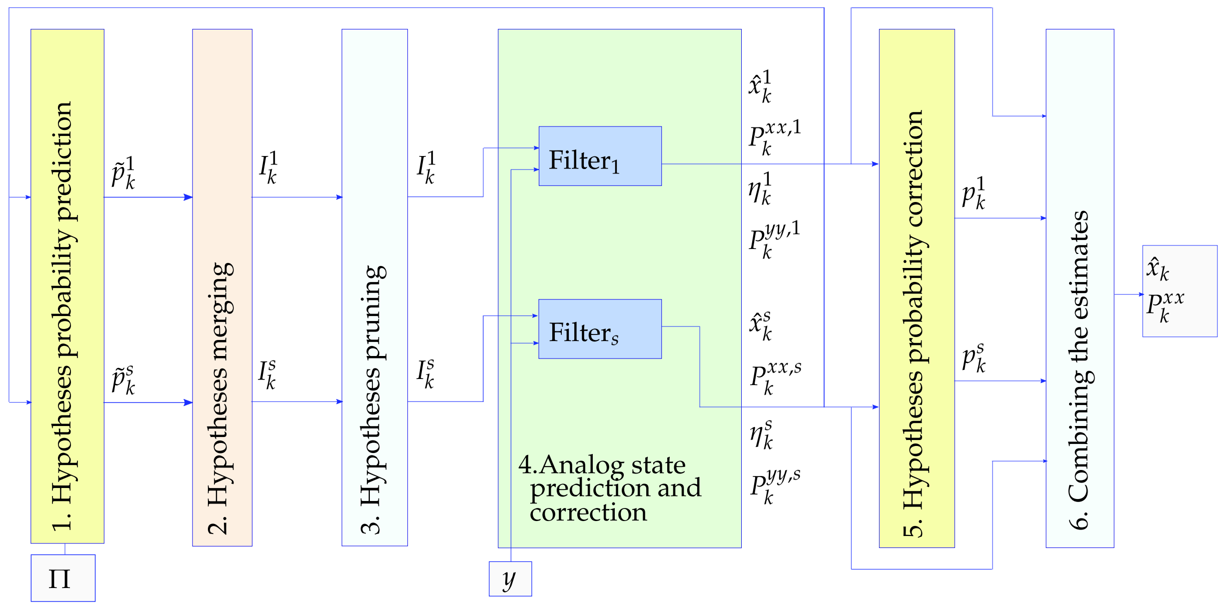

Figure 3.1: Diagram of the steps of the M^3HR algorithm. The M^3HR algorithm comprises six steps: in the *hypotheses probability prediction* step, the *a priori* probabilities are obtained; in the *hypotheses merging* step, the hypotheses are merged using the Gaussian mixture reduction; in the *hypotheses pruning* step, the hypotheses with probability lower are eliminated; in the *analog state prediction and correction* step, the state vector and the associated covariance matrix are estimated; in the *hypotheses probability correction* step, the *a posteriori* probabilities are obtained using Bayes' theorem and in the *combining the estimates* step, the recursive estimates of each filter and their respective *a posteriori* probabilities are combined. In this work, we propose an alternative method for the *hypotheses merging* step (orange) of the M^3H. Note that, the first and second term of (1.4) are respectively treated in the green and yellow steps in the M^3HR algorithm.

3.3 Gaussian Mixture Reduction via Clustering

Following an approach similar to [Crouse, D., Willett, P., Svensson, L., Svensson, D., Guerriero, M., 2011] that employs a clustering algorithm to perform Gaussian mixture reduction for smoothed state estimation avoiding track coalescence, we apply here the Gaussian mixture reduction via clustering (GMRC) approach [Schieferdecker, D., Huber, M. F., 2009] to the *hypotheses merging* step of the M^3H algorithm.

The GMRC optimizes the parameters of the reduced mixture according to the *integral quadratic distance* (ISD) [Scott, 1999] criterion and, to our knowledge, it is the best performing Gaussian mixture reduction method in approximation terms. In M^3H, instead of grouping hypotheses by their mode history, we look only at the current mode of each hypothesis and group them according to the analog state. Using the GMRC approach, our filter reduces all hypotheses with the same current mode m_k to a set of N_m hypotheses, where N_m is a tunning parameter, thus making up for a total of $N_m M$ hypotheses. An example is presented as follows.

Example 4. Consider the hybrid system with $M = 3$ possible operating modes, illustrated in Figure 3.2, we have the original set of hypotheses $\mathcal{I}_k = \{I_k^1, \ldots, I_k^N\}$, $N = 27$, each one of the white, black and gray balls represents one of these 27 hypotheses. The GMRC is employed to obtain the reduced set of these hypotheses, yielding $\mathcal{I}_k^{\mathrm{r}} = \{\tilde{I}_k^1, \ldots, \tilde{I}_k^{N_{\mathrm{m}}M}\}$, where we chose $N_{\mathrm{m}} = 3$. □

We now consider that the set of all N possible hypotheses at time k is defined by $\mathcal{I}_k \triangleq \{I_k^1, \ldots, I_k^N\}$. We see that the hypotheses set $\mathcal{I}_k$ represents a Gaussian mixture with N components, yielding the PDF

$$\rho(x_k|\mathcal{I}_k) \triangleq \sum_{s=1}^{N} \tilde{p}_k^s \mathcal{N}\left(\hat{x}_k^s, P_k^{xx,s}\right), \tag{3.2}$$

with weights $\tilde{p}_k^s$, means $\hat{x}_k^s$ and covariance matrices $P_k^{xx,s}$, $s = 1, \ldots, N$, where $\mathcal{N}$ accounts for the Gaussian PDF as in (2.100). Similarly, the desired reduced set of hypotheses $\mathcal{I}_k^{\mathrm{r}}$ represents a Gaussian mixture with $N_{\mathrm{m}}M$ components, yielding the PDF

$$\rho(x_k|\mathcal{I}_k^{\mathrm{r}}) \triangleq \sum_{r=1}^{N_{\mathrm{m}}M} \bar{p}_k^r \mathcal{N}\left(\hat{\bar{x}}_k^r, \bar{P}_k^{xx,r}\right), \tag{3.3}$$

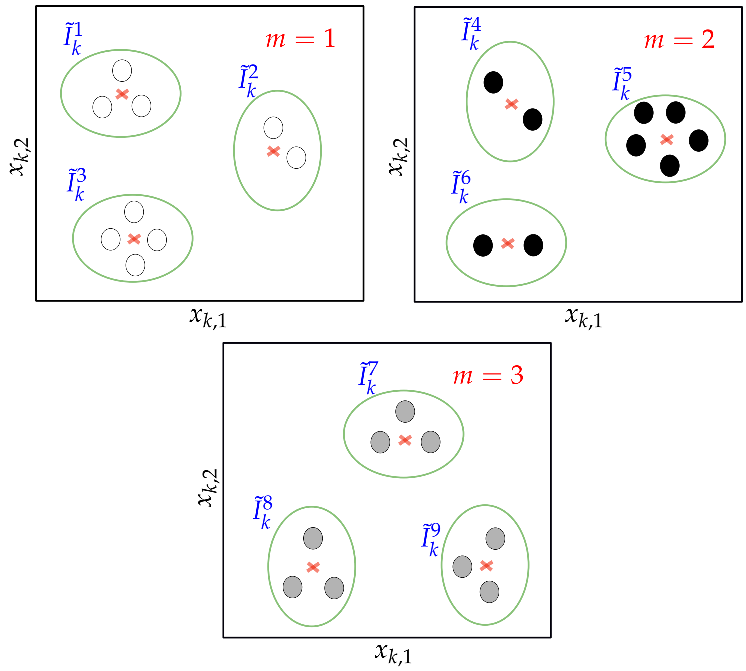

Figure 3.2: The Gaussian mixture reduction by clustering (GMRC) approach is employed in this example to obtain the reduced set of hypotheses of the stochastic hybrid system with $M = 3$ possible operating modes: white, black and gray. $\mathcal{I}_k = \{I_k^1, \ldots, I_k^{27}\}$ indicates the original set of hypotheses, each hypothesis is represented by a white, black or gray ball. Note that, this coloring is related to the last mode in each sequence corresponding to each hypothesis. $\mathcal{I}_k^{\mathrm{r}} = \{\tilde{I}_k^1, \ldots, \tilde{I}_k^{N_{\mathrm{m}}M}\}$ indicates the reduced set of hypotheses. The number of hypotheses for each mode is $N_{\mathrm{m}} = 3$ in this example. The centroids of the Gaussian components are illustrated by the $\times$ red marks.

with weights $\bar{p}_k^r$, mean $\hat{\bar{x}}_k^r$ and covariance matrix $\bar{P}_k^{xx,r}$, $r = 1, \ldots, N_{\mathrm{m}}M$.

In order to obtain the parameters $\bar{p}_k^r, \hat{\bar{x}}_k^r, \bar{P}_k^{xx,r}$ of the reduced mixture $\rho(x_k|\mathcal{I}_k^{\mathrm{r}})$ that maximizes its similarity to the original mixture $\rho(x_k|\mathcal{I}_k)$, we seek to minimize the ISD metric given by

$$
\begin{aligned}
J(\mathcal{I}_k^{\mathrm{r}}) &\triangleq \int [\rho(x_k|\mathcal{I}_k) - \rho(x_k|\mathcal{I}_k^{\mathrm{r}})]^2 \, dx_k, \\
&= \int \rho(x_k|\mathcal{I}_k)^2 - 2\rho(x_k|\mathcal{I}_k)\rho(x_k|\mathcal{I}_k^{\mathrm{r}}) + \rho(x_k|\mathcal{I}_k^{\mathrm{r}})^2 dx_k, \\
&= J_{NN} - 2J_{NN_{\mathrm{m}}} + J_{N_{\mathrm{m}}N_{\mathrm{m}}},
\end{aligned} \tag{3.4}
$$

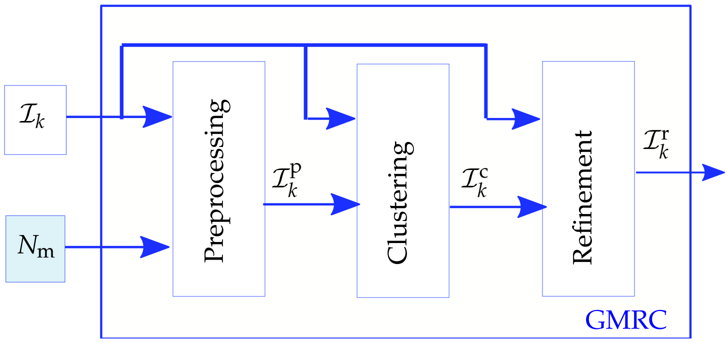

Figure 3.3: Diagram GMRC approach. The GMRC comprises three main sub-steps: the *preprocessing step*, the *clustering step* and the *refinement step*.

where

$$
\begin{aligned}
J_{NN} &\triangleq \int \rho(x_k|\mathcal{I}_k)^2 dx_k, \\
J_{NN_\mathrm{m}} &\triangleq \int \rho(x_k|\mathcal{I}_k)\rho(x_k|\mathcal{I}_k^\mathrm{r}) dx_k, \\
J_{N_\mathrm{m}N_\mathrm{m}} &\triangleq \int \rho(x_k|\mathcal{I}_k^\mathrm{r})^2 dx_k.
\end{aligned} \tag{3.5}
$$

In some texts, the ISD metric is referred to as *integral square error* (ISE) [Williams, 2003]. For details on the ISD, the reader is referred to ([Williams, 2003], pp. 3-19 - 3-23).

The GMRC approach comprises three basic sub-steps: the *preprocessing step*, the *clustering step* and the *refinement step*, as illustrated in Figure 3.3. In the first step, the initial reduced set with N_m components is obtained as $\mathcal{I}_k^\mathrm{p}$ using the *greedy Runnalls'* algorithm. In the second step, the k-means clustering algorithm (with maximum number of iterations i^max) is employed to obtain the reduced set of possible hypotheses $\mathcal{I}_k^\mathrm{c}$. In the third step, iterative optimization over the ISD metric is performed to refine $\mathcal{I}_k^\mathrm{c}$, yielding $\mathcal{I}_k^\mathrm{r}$. The GMRC algorithm is described as follows.

Algorithm 3.3.1. *Gaussian mixture reduction by clustering algorithm [Schieferdecker, D., Huber, M. F., 2009]*

Initialize the tuning parameters as follows: the merging depth d and the pruning threshold ε and the maximum number of hypotheses N_max.

1. *Preprocessing step. The initial reduced set of hypotheses* $\mathcal{I}_k^\mathrm{p}$*, is obtained applying the greedy Runnalls' algorithm to the hypotheses set* $\mathcal{I}_k$*. The greedy Runnalls' algorithm*

minimizes an upper bound on the Kullback-Leibler (KL) divergence between the original set of N hypotheses and the reduced set of $\mathrm{N_m}$ *hypotheses [Runnalls, A. R., 2007]. Initially, the merged hypotheses* $I_k^{\mathrm{m}_{i,j}}$ *are obtained from the combination of all possible hypotheses pairs* $\{I_k^i, I_k^j\}, \forall i, j$ *of the N hypotheses. Each merged hypothesis is defined by* $I_k^{\mathrm{m}_{i,j}} \triangleq \{p_\mathrm{m}, \hat{x}_\mathrm{m}, P_\mathrm{m}\}$*, whose elements are given by*

$$p_\mathrm{m} \triangleq p_i + p_j, \tag{3.6}$$

$$\hat{x}_\mathrm{m} \triangleq \frac{1}{p_\mathrm{m}} \left[p_i \hat{x}_k^i + p_j \hat{x}_k^j \right], \tag{3.7}$$

$$\begin{aligned} P_\mathrm{m} \triangleq & \left[\frac{p_i}{p_\mathrm{m}} \left[P_k^{xx,i} + \left(\hat{x}_k^i - \hat{x}_\mathrm{m} \right) \left(\hat{x}_k^i - \hat{x}_\mathrm{m} \right)^T \right] + \right. \\ & \left. \frac{p_j}{p_\mathrm{m}} \left[P_k^{xx,j} + \left(\hat{x}_k^j - \hat{x}_\mathrm{m} \right) \left(\hat{x}_k^j - \hat{x}_\mathrm{m} \right)^T \right] \right], \end{aligned} \tag{3.8}$$

where p_m *is the probability,* $\hat{x}_\mathrm{m}$ *is the state estimate and* P_m *is the associated covariance matrix and* $p_i = \tilde{p}(I_k^i)$ *and* $p_j = \tilde{p}(I_k^j)$*. Next, the measure of dissimilarity* $D^\mathrm{p}_\mathrm{i,j}$ *is obtained for each pair of the N hypotheses as*

$$D^\mathrm{p}_\mathrm{i,j} = \frac{1}{2} \left(p_\mathrm{m} \log \left[|P_\mathrm{m}| \right] - p_i \log \left[|P_k^{xx,i}| \right] - p_j \log \left[|P_k^{xx,j}| \right] \right). \tag{3.9}$$

A new mixture is then created by replacing, in the original mixture, the pair with smallest dissimilarity by its merged hypothesis. This reduces the number of components by one. This procedure is then applied recursively to the resulting mixtures until the mixture size is $\mathrm{N_m}$*.*

2. *Clustering step. The reduced set of hypotheses* $\mathcal{I}_k^\mathrm{c}$ *is obtained using the k-means clustering algorithm as in [Crouse, D., Willett, P., Pattipati, K., Svensson, L., 2011] initialized with the* $\mathrm{N_m}$ *cluster centers from* I_k^p*. The first step in the k-means algorithm is to associate each component* $I_s \in \mathcal{I}_k$ *to the closest cluster center in* $I_r \in \mathcal{I}_k^\mathrm{c}$*. To this purpose, we use the KL divergence as a pseudodistance as follows*

$$\begin{aligned} D^\mathrm{c}_\mathrm{s,r} = trace & \left[(\bar{P}_k^{xx,r})^{-1} \left(P_k^{xx,s} - \bar{P}_k^{xx,r} + (\hat{x}_k^s - \hat{\bar{x}}_k^r)(\hat{x}_k^s - \hat{\bar{x}}_k^r)^T \right) \right] \\ & + \log \left(\frac{|\bar{P}_k^{xx,r}|}{|P_k^{xx,s}|} \right), \end{aligned} \tag{3.10}$$

where $\hat{x}_k^s$ *and* $\hat{\bar{x}}_k^r$ *are the state estimate and* $P_k^{xx,s}$ *and* $\bar{P}_k^{xx,r}$*,* $s = 1, \ldots, N$*,* $r = 1, \ldots, \mathrm{N_m}$*, are the associated covariance matrices of* $I_s \in \mathcal{I}_k$ *and* $I_r \in \mathcal{I}_k^\mathrm{c}$*, respectively. Thus,* $\mathcal{I}_k^\mathrm{c}$ *is* *obtained from the merging of hypotheses associated with each of the* $\mathrm{N_m}$ *centroids. The merged hypotheses for a given cluster* C_r *are*

$$p_{\mathrm{c}} \triangleq \sum_{s \in C_r} p_k^s, \tag{3.11}$$

$$\hat{x}_{\mathrm{c}} \triangleq \frac{1}{p_{\mathrm{c}}} \sum_{s \in C_r} p_k^s \hat{x}_k^s, \tag{3.12}$$

$$P_{\mathrm{c}} \triangleq \sum_{s \in C_r} \frac{p_k^s}{p_{\mathrm{c}}} \left[P_k^{xx,s} + (\hat{x}_k^s - \hat{x}_{\mathrm{c}})(\hat{x}_k^s - \hat{x}_{\mathrm{c}})^T \right]. \tag{3.13}$$

This step is performed until the stop criterion given by the maximum number of iterations, $i^{\max}$*, is reached. At the end of this step, we have the reduced set of hypotheses* $\mathcal{I}_k^{\mathrm{c}} = \{I_k^1, \ldots, I_k^{N_\mathrm{m}}\}$.

3. *Refinement step. Starting from the reduced set* $\mathcal{I}_k^{\mathrm{c}}$*, we search the parameter space* $(\bar{p}_k^r, \hat{\bar{x}}_k^r, \bar{P}_k^{xx,r})$ *for a local minimum of the ISD, thus yielding the refined set of hypotheses* $\mathcal{I}_k^{\mathrm{r}}$*. The gradient of the ISD distance metric is used to obtain these parameters as*

$$\nabla J(\mathcal{I}_k^r) \;=\; -2\nabla J_{NN_\mathrm{m}} + \nabla J_{N_\mathrm{m}N_\mathrm{m}}, \tag{3.14}$$

Note that, since the element J_{NN} *in (3.4) does not depend on the parameter of the reduced set* $\mathcal{I}_k^r$*, it was discarded. The parameters of* $\mathcal{I}_k^{\mathrm{r}}$ *are then obtained through a two-stage optimization algorithm. We minimize in the parameters* $\hat{\bar{x}}_k^r$ *and* $\bar{P}_k^{xx,r}$ *employing a Quasi Newton method, whereas the optimization of parameter* $\bar{p}_k^r$ *is treated as a quadratic programming problem (see [Crouse, D., Willett, P., Pattipati, K., Svensson, L., 2011] for details). At the end of this step, we have the refined set of hypotheses* $\mathcal{I}_k^{\mathrm{r}} \triangleq \{I_k^1, \ldots, I_k^{N_\mathrm{m}}\}$*. Increment k and return to step* 1.

□

Remark 3.3.1. *Whereas the analog filter bank runs in linear time in the number of components* $\mathrm{N}_m M$*, Runnalls', k-means and the refinement step are quadratic in* $\mathrm{N}_m M$*. In contrast, the original* M^3H *may be implemented in linear time in the number of hypotheses. Despite this apparent disadvantage, our results show that it is only prevalent when low precision is required from the filter. When higher precision is demanded,* M^3HR *will need less components than* M^3H *to control the approximation error in the merging step.*

Remark 3.3.2. *Alternatively, West's algorithm can also be used in the preprocessing step of GMRC yielding the so-called GMRC-West algorithm. Simulation results from [Schieferdecker, D., Huber, M. F., 2009] show that, the processing time of the GMRC-West is faster than the original GMRC; however, error performance can be worse compared to the original GMRC that uses the Runnall's algorithm in the preprocessing step.*

3.4 Simulated Example: Target Tracking using a Radar

Mixture reduction techniques have already been applied in the context of target tracking [Salmond, D.J, 1990; Williams, 2003; Crouse, D., Willett, P., Svensson, L., Svensson, D., Guerriero, M., 2011]. However, in this domain of application, the digital states correspond exclusively to observation models. Here we extend these techniques to the dynamics as well, thus comprising the full hybrid system. Although, the application example in this work also addresses the problem of target tracking under multiple dynamic models, our approach is applicable to general hybrid systems under the structure of Markov jump systems.

3.4.1 Problem Description

We now consider the target tracking problem of [Boers, Y., Driessen, H, 2004]. Let an aircraft have three operating modes, $m_k \in \mathbb{M} = \{1, 2, 3\}$, where $m_k = 1$ corresponds to a straight maneuver with constant velocity and altitude, $m_k = 2$ corresponds to a circular maneuver (coordinated turn) and $m_k = 3$ corresponds to an accelerated straight line maneuver. Let d_x, d_y and d_z be the position components, v_x, v_y and v_z be the linear velocity components, ω be the angular velocity on the x-y plane and a_z be the accelaration along the z-axis. Our goal is to obtain estimates for both the digital state (mode) $\hat{m}_k$ and analog state $\hat{x}_k$ with corresponding covariance P_k^{xx}.

For $m_k = 1$, the linear process model f_1 is given by

$$x_k = \begin{bmatrix} 1 & 0 & 0 & T & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & T & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & T & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} x_{k-1} + \begin{bmatrix} \frac{1}{2}a_s T^2 & 0 & 0 \\ 0 & \frac{1}{2}a_s T^2 & 0 \\ 0 & 0 & \frac{1}{2}a_s T^2 \\ a_s T & 0 & 0 \\ 0 & a_s T & 0 \\ 0 & 0 & a_s T \\ 0 & 0 & 0 \end{bmatrix} w_{k-1}, \tag{3.15}$$

where $x_k = \begin{bmatrix} d_x & d_y & d_z & v_x & v_y & v_z & \omega \end{bmatrix}^T$, $T = 5$ s is the sampling period. The process noise modeling disturbing accelerations, $w_{k-1} \sim \mathcal{N}(0, Q_{m_k})$, is assumed to be a zero-mean white noise with covariance Q_{m_k} that depends on the operating mode. The acceleration parameter is set to $a_s = 1.5$ m/s^2.

For $m_k = 2$, the nonlinear process model f_2 is given by

$$x_k = \begin{bmatrix} 1 & 0 & 0 & \sin(\omega_{k-1}T)/\omega_{k-1} & (\cos(\omega_{k-1}T) - 1)/\omega_{k-1} & 0 & 0 \\ 0 & 1 & 0 & (1 - \cos(\omega_{k-1}T))/\omega_{k-1} & \sin(\omega_{k-1}T)/\omega_{k-1} & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & T & 0 \\ 0 & 0 & 0 & \cos(\omega_{k-1}T) & -\sin(\omega_{k-1}T) & 0 & 0 \\ 0 & 0 & 0 & \sin(\omega_{k-1}T) & \cos(\omega_{k-1}T) & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} x_{k-1} + G_{k-1} w_{k-1}, \tag{3.16}$$

where

$$G_{k-1} = \begin{bmatrix} \text{ca} & -\text{sa} & 0 & 0 & 0 & 0 & 0 \\ \text{sa} & \text{ca} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \text{ca} & -\text{sa} & 0 & 0 \\ 0 & 0 & 0 & \text{sa} & \text{ca} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & L_{k-1} \end{bmatrix} \begin{bmatrix} \frac{1}{2}a_{\text{long}} T^2 & 0 & 0 \\ 0 & \frac{1}{2}a_{\text{lat}} T^2 & 0 \\ 0 & 0 & \frac{1}{2}a_{\text{vert}} T^2 \\ a_{\text{long}} T & 0 & 0 \\ 0 & a_{\text{lat}} T & 0 \\ 0 & 0 & a_{\text{vert}} T \\ 0 & a_{\text{lat}} & 0 \end{bmatrix},$$

where $x_k = \begin{bmatrix} d_x & d_y & d_z & v_x & v_y & v_z & \omega \end{bmatrix}^T$, $\text{ca} \triangleq \cos(\text{atan2}(v_{y,k-1}, v_{x,k-1}))$, $\text{sa} \triangleq \sin(\text{atan2}(v_{y,k-1}, v_{x,k-1}))$ and $L_{k-1} = \frac{1}{\sqrt{(v_{x,k-1})^2 + (v_{y,k-1})^2 + (v_{z,k-1})^2}}$. For the coordinated turn model, we set $a_{\text{long}} = 15$ m/s^2 for the longitudinal acceleration, $a_{\text{vert}} = 15$ m/s^2 for the vertical acceleration and $a_{\text{lat}} = 20$ m/s^2 for the lateral acceleration.

For $m_k = 3$, the linear process model f_3 is given by

$$x_k = \begin{bmatrix} 1 & 0 & 0 & T & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & T & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & T & \frac{1}{2}T^2 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & T \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} x_{k-1} + \begin{bmatrix} \frac{1}{2}a_{az}T^2 & 0 & 0 \\ 0 & \frac{1}{2}a_{az}T^2 & 0 \\ 0 & 0 & \frac{1}{2}a_{az}T^2 \\ a_{az}T & 0 & 0 \\ 0 & a_{az}T & 0 \\ 0 & 0 & a_{az}T \\ 0 & 0 & a_{az} \end{bmatrix} w_{k-1}, \quad (3.17)$$

where $x_k = \begin{bmatrix} d_x & d_y & d_z & v_x & v_y & v_z & a_z \end{bmatrix}^T$, a_{az} =20 m/s^2. Note that, for $m_k = 1,2$, $a_z = 0$ and, for $m_k = 3$, $\omega = 0$.

For all operating modes, the nonlinear observation model h is given by

$$y_k = \begin{bmatrix} \sqrt{d_{x,k}^2 + d_{y,k}^2 + d_{z,k}^2} \\ \operatorname{atan}\left(\frac{d_{y,k}}{d_{x,k}}\right) \\ \operatorname{atan}\left(\frac{d_{z,k}}{\sqrt{d_{x,k}^2+d_{y,k}^2}}\right) \\ \frac{d_{x,k}v_{x,k}+d_{y,k}v_{y,k}+d_{z,k}v_{z,k}}{\sqrt{d_{x,k}^2+d_{y,k}^2+d_{z,k}^2}} \end{bmatrix} + \nu_k, \quad (3.18)$$

where the components of y are the radar measurements for range, bearing, elevation and Doppler speed. The measurement noise, $v_k \sim \mathcal{N}(0, R_{m_k})$, is assumed to be zero-mean white noise with constant covariance $R_{m_k} = \operatorname{diag}([20^2, (8 \times 10^{-3})^2, (8 \times 10^{-3})^2, 10^2])$. We assume that the transition probability matrix is given by

$$\Pi = \begin{bmatrix} 0.96 & 0.02 & 0.02 \\ 0.02 & 0.96 & 0.02 \\ 0.02 & 0.02 & 0.96 \end{bmatrix}. \quad (3.19)$$

We also assume the aircraft starts from $x_0 = \begin{bmatrix} 100000 & 0 & 2000 & -400 & 0 & 0 & -0.1 \end{bmatrix}^T$, with a coordinated turn ($m_0 = 2$). That is, the aircraft starts at about 100 km from the radar with constant altitude of 2000 m. The aircraft is flying inbound at a speed of 400 m/s and an angular velocity of 0.1 rad/s.

3.4.2 Numerical Experiments on State Estimation

For M^{3}H, we set the merging depth $d = 3$, the pruning threshold $\epsilon = 0.01$ and $N_{\max} = 27$. In M^{3}HR, we set the reduced hypotheses number $N_{\mathrm{m}} = 3$. The filter bank of the M^3H and M^3HR algorithms is composed by three unscented Kalman filters (UKF) [Menegaz, H.M.T., Ishihara, J.Y., Borges, G.A., Vargas, A.N., 2015]. We performed 50 distinct realizations using IMM, M^3H and M^3HR in a desktop Dell Inspiron Intel Core i3 with 3.2 GHz processor and 4GB RAM memory running Matlab. We also tested M^3H with partial execution of GMRC, for example, executing only up to the first step or up to the second one.

a) Comparison within a single realization

We begin by presenting results for the tracking of a single target trajectory. Figure 3.4 shows the simulation results for one realization of the target tracking problem using the original M^3H and the modified M^3HR. Figure 3.4a shows the aircraft flight trajectory. Figure 3.4b shows that, after the merging step, the proposed approach M^3HR (black line) reduces the number of hypotheses more significantly than does the original algorithm M^3H (red line).

Figure 3.4c shows that M^3HR was more accurate in the estimation of the operating modes. For example, in the mid-plot from about time $t = 240$s until time $t = 300$s, we observe that the mode 2 (circular maneuver) is correctly estimated by M^3HR while M^3H wrongly detects that mode 3 is active. Figure 3.4d shows that the M^3H yields erroneous estimates for the operating mode more often than M^3HR. Figure 3.4e shows the estimated aircraft position coordinates from radar measurements using the M^3HR (black line) and M^3H (red line) approaches. We observe that M^3HR yields more accurate estimates for the position compared to M^3H estimates. Finally, Figure 3.4f illustrates the error estimates for the position coordinate d_x using both M^3HR (black line) and M^3H (red line). Note that, in addition to yielding a smaller error, M^3HR provides uncertainty estimates (plus and minus three standard deviations) consistent with the true position.

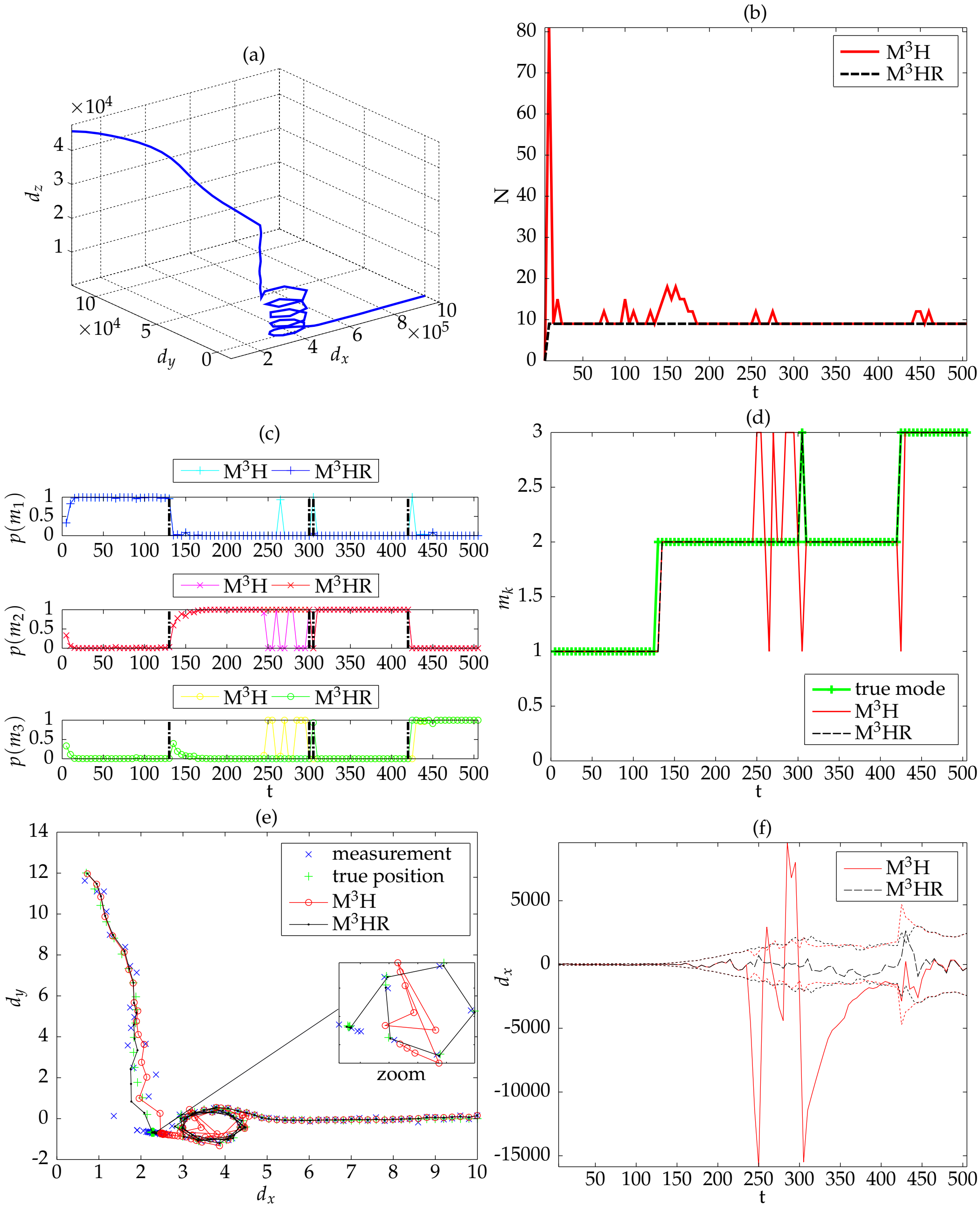

Figure 3.4: Tracking the aircraft flight trajectory using a radar: (a) true aircraft trajectory, (b) number of hypotheses after the merging step using M^{3}HR (- -) and M^{3}H (—), (c) probability of the modes: mode 1 (—|—) using M^{3}HR (blue) and M^{3}H (light blue), mode 2 (—×—) using M^{3}HR (red) and M^{3}H (pink) and mode 3 (—o—) using M^{3}HR (green) and M^{3}H (yellow) and the black vertical line indicate actual mode transitions, (d) estimate of the digital state using M^{3}HR (- -) and M^{3}H (—) in comparison with the true value (—|—), (e) estimates of the position coordinates using M^{3}HR (—•—) and M^{3}H (—o—) in comparison with the measurements ($\times$) and the true position ($+$) and (f) estimation error of the position component d_x using M^{3}HR (- -) and M^{3}H (—) algorithm. The dotted line (- -) indicate plus and minus three standard deviations around the error estimates.

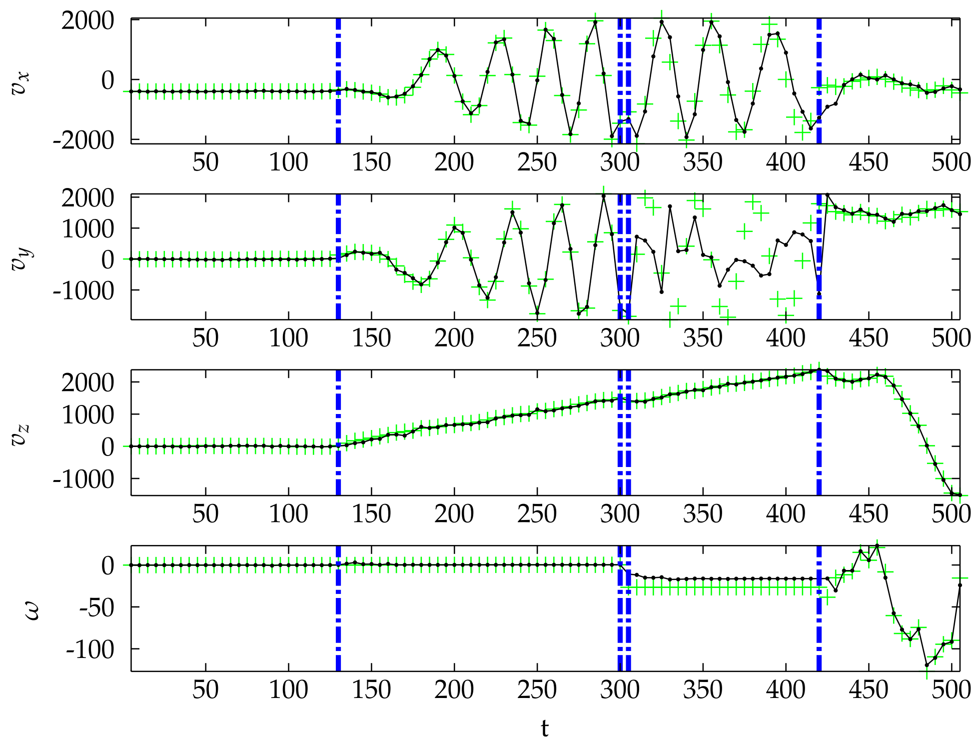

Figure 3.5: Target velocity during the flight: x, y, and z linear velocity components and angular velocity. The estimate of the analog state using M^3HR(—•—) is compared to the true value (—+—). The vertical lines indicate mode transition times.

Figure 3.5 illustrates how the operating modes influence the variations of some components of the analog state. For example, we observe that mode 2 differs from other modes by variations in the linear velocity components from time $t = 130$s to $t = 300$s, whereas mode 3 yields larger values for the angular velocity from time $t = 425$s to $t = 505$s. Thus, because the operating modes correlate strongly with the analog states, it is natural to expect that the use of information from both analog and digital states may improve the error in the merging step.

b) Comparison over multiple realizations

Overall performance was assessed averaging the results from 50 distinct process realizations. Table 3.1 compares the mean processing time per sampling period, T_{CPU}, and RMSE (regarding position coordinates only). We see that M^3H provides more accurate position estimates than IMM; however, the processing time of M^3H is twice that of IMM. These results are in accordance with the results of [Boers, Y., Driessen, H, 2004].

Table 3.1: Mean processing time to run each algorithm from time $k-1$ to time k, T_{CPU}, RMSE (for position coordinates) and frequency of error in digital state estimates for 50 distinct realizations using the IMM, M^3H and M^3HR filter. We consider three scenarios for the merging step of M^3HR, where the M^3HR (1-step) uses only the preprocessing step, M^3HR (2-step) uses the preprocessing and clustering step and M^3HR (3-step) uses preprocessing, clustering and refinement step.

Algorithms	T_{CPU} (ms)	RMSE for positon (m)	% Mode error
IMM	1.9	523	12.3
M^3H	3.9	497	11.6
M^3HR (1-step)	5.8	437	4.8
M^3HR (2-step)	6.5	435	4.5
M^3HR (3-step)	7.5	430	4.1

Compared to M^3H, the full M^3HR reduces the RMSE index by approximately 14% but with a processing time over twice than of M^3H. On the other hand, M^3HR truncated in the first step yields approximately the same improvement in accuracy but with the processing time 24% smaller compared to the full M^3HR. This suggests a diminishing returns property, in which most of the benefit from using GMRC is collected right in the first step. Moreover, we observe that the M^3HR misses the active mode in 4.1% of times compared to 11.6% of M^3H, as shown in Table 3.1.

This diminishing returns property is further evinced in Figure 3.6, where we evaluate how the filter performance is influenced by i) the variation of the measurement noise levels and ii) the variation in the three steps of the GMRC algorithm employed in the merging step of M^3HR. By rescaling the measurement noise covariance R_{m_k}, we choose three different values of signal to noise ratio (SNR), 10, 20 and 30dB. From Figure 3.6a, we observe an increase of about 16% in the position RMSE for the largest noise level (10 dB) using M^3H compared to M^3HR (3 step). Note that, for smaller measurement noise levels, 20 dB and 30 dB, the M^3HR(3 step) yields estimates about, respectively, 34% and 55% more accurate than M^3H. A similar analysis is valid for RMSE for velocity; see Figure 3.6b.

In regard to performance changes with the variation in the three steps of GMRC. Figure 3.6 shows that M^3HR (3-step) yields position estimates only 1% more accurate than M^3HR (1-step). In addition, the M^3HR (1-step) reduces processing time by approximately 24% in comparison to M^3HR (3-step). We conclude that employing the

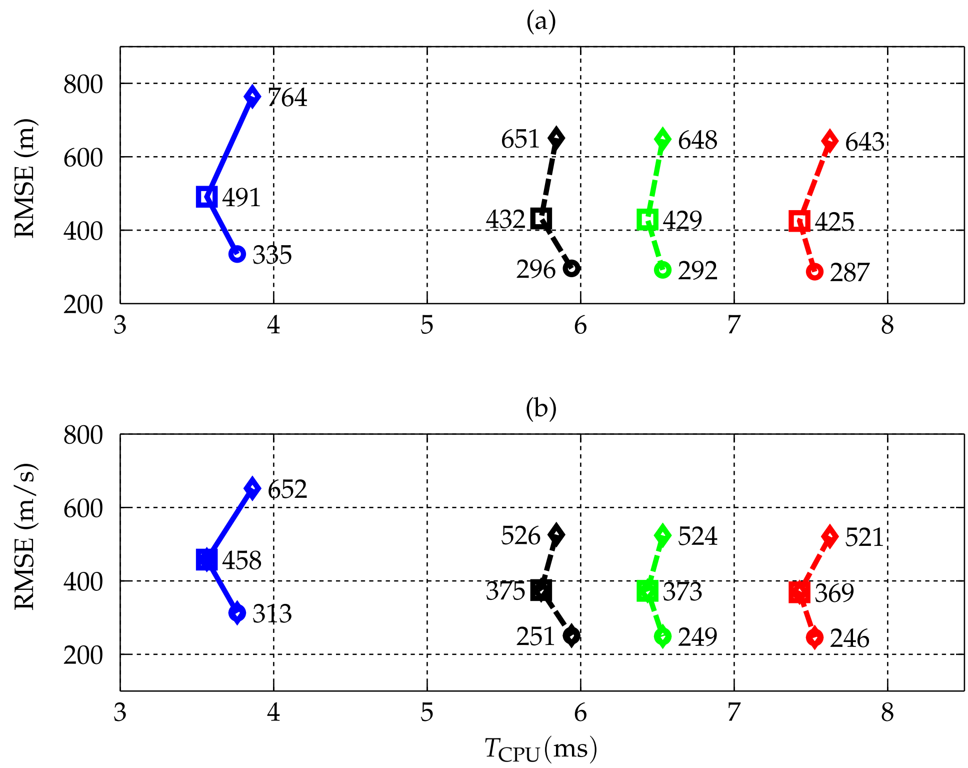

Figure 3.6: Mean normalized processing time, T_{CPU}, and RMSE (for position coordinates) for 50-run Monte Carlo simulation for M^{3}HR and M^{3}H algorithm with different merging depth: (a) position and (b) velocity. The variation of the measurement noise levels are 10 dB (◇), 20 dB (□) and 30 dB (○). The variation of the three scenarios for the merging step of M^{3}HR are M^{3}H (1 step) (black line), M^{3}HR (2 step) (green line) and M^{3}HR (3 step) (red line) compared to M^{3}H (blue line).

three steps of GMRC in the merging step of M^{3}HR may not pay off depending on the application, as the 1-step version might already provide satisfying improvement.

c) Comparison over different time-precision tradeoffs

Our main results come from comparing M^{3}H and M^{3}HR over different user preferences regarding processing time and filter approximation error. We restrict the comparison to M^{3}HR (1 step) since it had a more satisfying performance in the experiments above. We compare the performance of M^{3}HR (1 step) and M^{3}H using different values of $N_{\text{m}} = 1, 2, 3, 4, 5$ (□), and merging depth, $d = 2, 3, 4, 5, 6, 7, 8$ (○).

Results, shown in Figure 3.7, indicate that the performance of M^{3}HR dominates that of M^{3}H in all cases when the user demands RMSE below 525 m. As a conclusion, we see that M^{3}HR adds improved flexibility to the user who can now choose over a

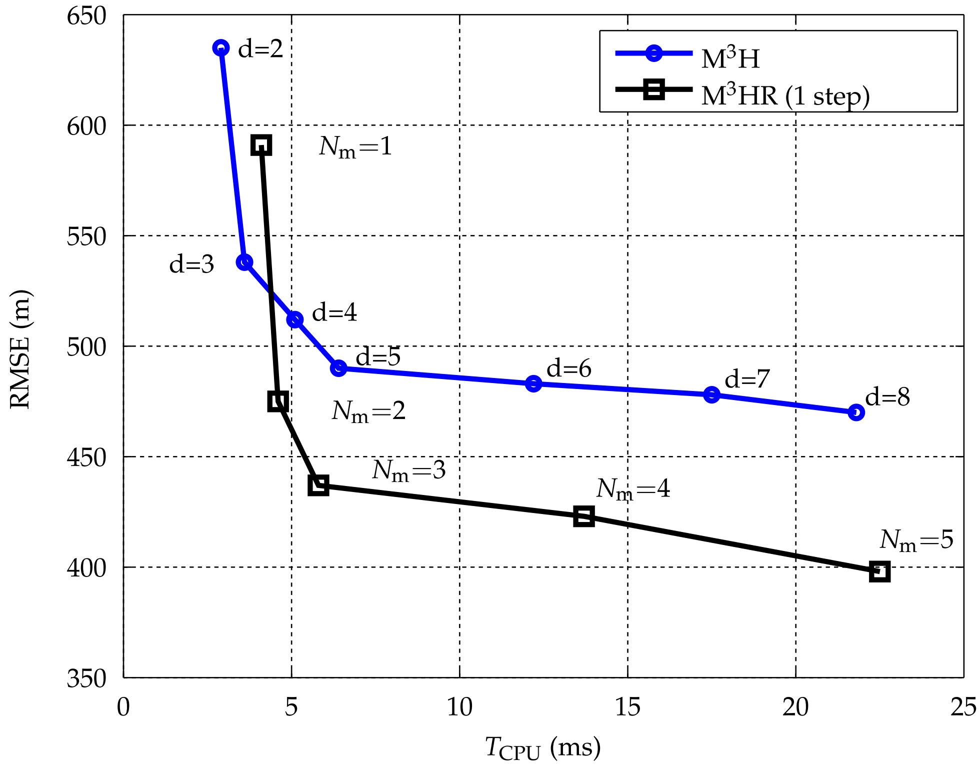

Figure 3.7: Mean processing time, T_{CPU}, and position RMSE for 50 distinct realizations using the M^{3}HR (1 step) (black line) and M^{3}H (blue line) algorithms. The values of number of components of the Gaussian mixture are 1, 2, 3, 4, 5 (□) and of merging depth are 2, 3, 4, 5, 6, 7, 8 (○). The values for each algorithm are shown above from left to right.

larger range of RMSE. It is worth noticing that very little improvement is gained by M^{3}H increasing the merging depth beyond $d = 5$. This highlights how little extra information is encoded in the mode history to help estimate analog states, which was our initial motivation for proposing M^{3}HR.

3.5 Concluding Remarks

In this chapter we present a modified version of the multiple models and multiple hypotheses (M^{3}H) algorithm to suboptimally solve the problem of state estimation for Markov jump nonlinear systems. In particular, the M^{3}H algorithm merges the hypotheses with the same sequence of modes in the last d steps. Thus, only information from the digital state is employed to merge the hypotheses.

In the context of Markov state transitions, the mode sequence provides no useful information to the task of predicting future states given that the current mode is known. This suggests that one should merge hypotheses based on the current state (both analog and digital) rather than based on the sequence of digital states. Thus, the reduction of Gaussian mixtures may be well suited to perform the merging operation.

Then, we use Gaussian reduction methods as an alternative for the merging step of the original M^3H algorithm. We employ the Gaussian mixture reduction by clustering (GMRC) approach to merge and eliminate hypotheses. Thus, information from both the analog and digital states is used to merge the hypotheses, while only information from the digital state is employed in the original method. Numerical results suggest that the proposed approach M^3HR provides improvement in the accuracy of analog and digital state estimates compared with M^3H algorithm.

In addition, the M^3HR provides an effective mechanism for users to explore the tradeoff between filter precision and processing time. Users set their preferences by defining the maximum number of mixture components N_{m}. Likewise, a similar tradeoff may be observed for M^3H by manipulating the merging depth d. Our numerical results indicate that manipulating N_{m} in M^3HR is more efficient than choosing d in M^3H when increased precision is demanded. That is, for the same processing time, improved accuracy was observed for M^3HR.

Chapter 4

Equality-constrained State Estimation for Hybrid Systems

In this chapter, we address the problem of state estimation for hybrid systems with state equality constraints. A literature review on this topic is presented in Section 2.3.5. This work deals with the extension of constrained stochastic filtering methods for the hybrid case. A brief review of these methods in the non-hybrid context is provided in Section 2.1.4. It is recommended to read it before reading this Section.

In Section 4.2, we investigate the mode-independent equality constraints case for linear systems. Then, we extend the necessary conditions for the Kalman filter yielding state estimates that satisfy the equality constraint to the hybrid case. In Section 4.3, we consider the mode-dependent equality constraints case for linear systems. Then, we present two different possibilities to enforce an equality constraint along time. In Section 4.4, we investigate the mode-dependent equality constraints case for nonlinear systems. Then, we discuss approaches that provide approximate solutions. Finally, in Section 4.5, these algorithms are compared to the classical IMM by means of three examples.

4.1 Problem Statement

We consider the discrete-time hybrid stochastic system given by (1.1)-(1.3), whose equations are repeated here for convenience,

$$
\begin{aligned}
x_k &= f_{m_k}(x_{k-1}, u_{k-1}, w_{k-1}, k-1), \\
\pi_{s|r} &= \Pr\{m_k = s | m_{k-1} = r\}, \\
y_k &= h_{m_k}(x_k, v_k, k).
\end{aligned}
$$

In addition, we assume that the analog state vector x_k is known to satisfy the equality constraint

$$g_{m_k}(x_k, k) = d_{m_k}, \tag{4.1}$$

where the function $g_{m_k} : \mathbb{R}^n \times \mathbb{N} \to \mathbb{R}^s$, and $d_{m_k} \in \mathbb{R}^s$, is assumed to be known. Note that, in the general case, such constraint may vary with both time k and operating mode m_k.

The *hybrid constrained stochastic filtering* problem seeks to provide state estimates $\hat{x}_k$ and $\hat{m}_k$ given by meaningful statistics (such as the mean or the mode) from the joint *a posteriori* PDF of x_k and m_k (1.4) given a sequence of noisy sampled measurements, $y_{1:k}$, and a hybrid stochastic dynamic model. In this chapter, the analog state estimates must satisfy the constraint (4.1).

Next we define special cases of the general hybrid constrained stochastic filtering problem that are investigated in this work.

Problem 1. *Assume that (1.1)-(1.3) is a linear hybrid system given by*

$$
\begin{aligned}
x_k &= A_{m_k} x_{k-1} + B_{m_k} u_{k-1} + w_{k-1}, && (4.2) \\
y_k &= C_{m_k} x_k + v_k. && (4.3)
\end{aligned}
$$

where $A_{m_k} \in \mathbb{R}^{n \times n}$, $B_{m_k} \in \mathbb{R}^{n \times p}$ and $C_{m_k} \in \mathbb{R}^{m \times n}$ are assumed to be known for all possible modes m_k. Also, assume that the state vector x_k satisfies the mode-independent equality constraint

$$D x_k = d, \tag{4.4}$$

where $D \in \mathbb{R}^{s \times n}$ and $d \in \mathbb{R}^{s \times 1}$ are assumed to be known. Our goal is to obtain state estimates $\hat{x}_k$ and $\hat{m}_k$ satisfying $D\hat{x}_k = d$, $\forall k$.

Problem 2. *Consider the linear hybrid system given by (4.2)-(4.3), whose state vector x_k satisfies the mode-dependent equality constraint*

$$D_{m_k} x_k = d_{m_k}. \tag{4.5}$$

where $D_{m_k} \in \mathbb{R}^{s \times n}$ and $d_{m_k} \in \mathbb{R}^{s \times 1}$ correspond to the equality constraint for the mode active at the time k. Our goal is to obtain state estimates $\hat{x}_k$ and $\hat{m}_k$ satisfying $D_{m_k}\hat{x}_k = d_{m_k}$, $\forall k$.

Note that, for the problems 1 and 2, the matrices A_{m_k}, B_{m_k} and C_{m_k} are constant for a given operating mode m_k. These matrices are time homogeneous in the sense that they do not depend directly on k.

Problem 3. *Consider the nonlinear hybrid system (1.1)-(1.3), whose state vector x_k satisfies the mode-dependent nonlinear equality constraint (1.5). Our goal is to obtain state estimates $\hat{x}_k$ and $\hat{m}_k$ satisfying $g_{m_k}(x_k, k) = d_{m_k}$, $\forall k$.*

4.2 Linear and Mode-independent Case

For the non-hybrid case, [Teixeira, B. O. S., Chandrasekar, J., Tôrres, L. A. B., Aguirre, L. A., Bernstein, D. S., 2009] present the sufficient conditions for the conventional Kalman filter to yield state estimates that satisfy the equality constraint. The authors prove that if the system obeys a condition of compatibility of the dynamics with the equality constraint (Lemma 2.1.1), then the conventional Kalman filter is able deal with time-invariant equality-constrained systems as long as a proper initialization is performed, see Proposition 2.1.1. In this case, we consider that the dynamics are compatible with the equality constraint when the dynamics remain in the hyperplane defined by the equality constraint. In other words, there is such compatibility when the conditions in Lemma 2.1.1 are verified [Rong Li, X., 2016].

Assumption 4.2.1 (Compatibility of contraints and dynamics)**.** *Lemma 2.1.1 is extended to the hybrid case. [[Teixeira, B. O. S., Chandrasekar, J., Tôrres, L. A. B., Aguirre, L. A., Bernstein, D. S., 2009], Proposition 3.1] present sufficient conditions for the non-hybrid linear time-invariant system to be compatible with the equality constraint. In this work, we consider that all those conditions are valid for all operating modes of the system, that is,*

$$D_{m_k} Q_{m_k} = 0_{s\times q}, \tag{4.6}$$

$$D_{m_k} A_{m_k} = D_{m_k}, \tag{4.7}$$

$$D_{m_k} B_{m_k} u_{k-1} = 0_{s\times 1}. \tag{4.8}$$

where the matrices of the system A_{m_k}, B_{m_k}, the noise covariance matrix Q_{m_k} and the equality constraint D_{m_k} are assumed to be known.

The next proposition extends Proposition 2.1.1 for linear hybrid systems with mode-independent equality constraints, see Problem 1. Specifically, we present the necessary conditions for the classical IMM algorithm to yield state estimates satisfying (4.4) for all $k \geq 1$ and (almost) all noise realizations.

Proposition 4.2.1. *Consider the IMM algorithm 2.3.1 initialized with $\hat{x}_0^r$ and $\hat{P}_0^{xx,r}$, for $r = 1, \ldots, M$, given by (2.48) and (2.49) as in Remark 2.1.1 and applied to the scenario of Problem 1. Assume that the conditions in Assumption 4.2.1 hold for all M mode-dependent linear process models (4.2). Then, for all $k \geq 1$,*

(i) the mixed estimates, $\bar{x}_{k-1}^s$ (2.90) and $\bar{P}_{k-1}^{xx,s}$ (2.91) satisfy $D\bar{x}_{k-1}^s = d$ and $D\bar{P}_{k-1}^{xx,s} = 0_{s\times n}$,

(ii) the filter estimates, $\hat{x}_k^s$ and $P_k^{xx,s}$ (2.92) satisfy $D\hat{x}_k^s = d$ and $DP_k^{xx,s} = 0_{s\times n}$, and

(iii) the combined estimates, $\hat{x}_k$ (2.95) and P_k^{xx} (2.96) satisfy $D\hat{x}_k = d$ and $DP_k^{xx} = 0_{s\times n}$.

Proof. Suppose that part (ii) is true at time k-1 such that the filter estimates satisfy that $D\hat{x}_{k-1}^r = d$ and $DP_{k-1}^{xx,r} = 0_{s\times n}$, multiplying (2.90) by D yields

$$\begin{aligned} D\bar{x}_{k-1}^s &= \sum_{r=1}^{M} \mu_{k-1}^{s|r} D\hat{x}_{k-1}^r, \\ &= \sum_{r=1}^{M} \mu_{k-1}^{s|r} d = d \sum_{r=1}^{M} \mu_{k-1}^{s|r} = d, \end{aligned} \tag{4.9}$$

where $\sum_{r=1}^{M} \mu_{k-1}^{s|r} = 1$. Using (4.9) and multiplying (2.91) by D yields

$$\begin{aligned} D\bar{P}_{k-1}^{\text{xx},s} &= \sum_{r=1}^{M} \mu_{k-1}^{s|r} \left[DP_{k-1}^{\text{xx},r} + \left(D\hat{x}_{k-1}^{r} - D\bar{x}_{k-1}^{s}\right) \cdot \left(D\hat{x}_{k-1}^{r} - D\bar{x}_{k-1}^{s}\right)^{T} \right], \\ &= \sum_{r=1}^{M} \mu_{k-1}^{s|r} \left[0_{s\times n} + (d-d)\,(d-d)^{T} \right] = 0_{s\times n}. \end{aligned} \tag{4.10}$$

Then we prove part (i) at time $k-1$.

For the M Kalman filter estimates, we decompose this filter given by (2.92) in two steps. First, given (4.9)-(4.10) and multiplying (2.9)-(2.10) by D, we obtain $\hat{x}_{k|k-1}^{s}$ and $P_{k|k-1}^{\text{xx},s}$ satisfying $D\hat{x}_{k|k-1}^{s} = d$ and $DP_{k|k-1}^{\text{xx},s} = 0_{s\times n}$. Second, given $D\hat{x}_{k|k-1}^{s} = d$ and $DP_{k|k-1}^{\text{xx},s} = 0_{s\times n}$ using (2.12)-(2.16), we obtain $\hat{x}_{k}^{s}$ and $P_{k}^{\text{xx},s}$ satisfying $D\hat{x}_{k}^{s} = d$ and $DP_{k}^{\text{xx},s} = 0_{s\times n}$ such that part (ii) is proved at time k. By induction, provided that the filter estimates satisfy the constraints at time 0, we have proved (i) and (ii) for all k$\geq$ 0, then by induction the part (i) and (ii) are true for all k.

Finally, for the combined estimates, multiply (2.95)-(2.96) by D and use part (ii) to obtain (iii) at time k. □

Henceforth, whenever the IMM filter (Section 2.3.1) is properly initialized as in Remark 2.1.1 in order to satisfy the equality constraint (4.4), it is referred to as IMM with projected initial condition (IMM$_0$). However, it is important to point out that this is only possible for models that are compatible with the equality constraint.

4.3 Linear and Mode-dependent Case

Consider now Problem 2. In this case, we consider that the process and observation models (4.2)-(4.3) are constant for a given mode m_k and the equality constraint (4.5) is a function of the active mode.

For Problem 2, the classical IMM does not guarantee that (4.5) is satisfied for all $k \geq 1$. In this work, the proposed approach is a modified version of the IMM filter to provide state estimates, $\hat{x}_{k}^{\text{p}}$ and P_{k}^{xxp}, that satisfy the equality constraint (4.5); see Figure 4.1. The proposed approach is here called equality-constrained interacting multiple models (CIMM). We investigate two algorithms based on the IMM filter that

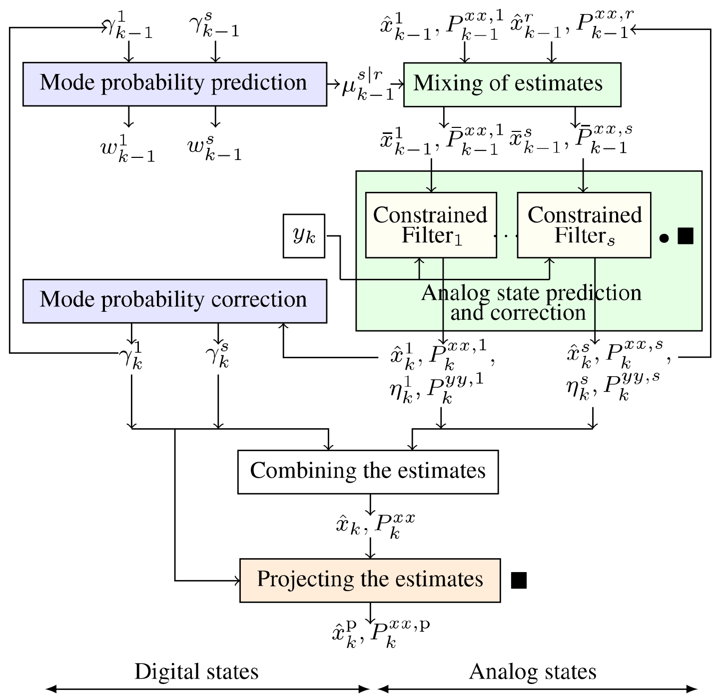

Figure 4.1: Diagram of the CIMM$_1$ algorithm. The CIMM$_1$ is a modified version of the classical IMM filter for to obtain state estimates satisfying the equality constraint. CIHE (●) and CIMM$_1$ (■) differ in the step of the filter where the equality constraints are enforced. The classical IMM has a similar structure except for cases where constraints are used. For both approaches, the first and second terms of the right-hand side of (1.4) are respectively shown in green and in blue.

enforce the equality constraint (4.5) on the state vector that vary with the mode m_k. These algorithms differ in the step of the IMM filter where the equality constraints are enforced. We consider the following cases: (i) using equality-constrained filters and (ii) using an additional step after the combined estimates and the case (i). The two cases are presented in Sections 4.3.1 and 4.3.2, respectively.

4.3.1 Constrained Filters

In the first case, we employ the approach used in [Mann, G., Hwang, I., 2013]. This work addresses the constrained stochastic filtering problem for linear hybrid systems. This approach is called constrained innovation hybrid estimator (CIHE). The CIHE provides state estimates that satisfy the equality constraint (4.5).

The CIHE employs the IMM algorithm whose filter bank is composed by equality-constrained filters, reviewed in Section 2.1.4, where the state vector $\bar{x}^s_{k-1}$ and the corresponding covariance $\bar{P}^{xx,s}_{k-1}$ are replaced by the constrained state estimates $\bar{x}^{p,s}_{k-1}$ and the covariance $\bar{P}^{xxp,s}_{k-1}$ at time $k-1$ and the filter return the constrained state estimates $\hat{x}^{p,s}_k$ and its corresponding covariance $P^{xxp,s}_k$ at time k. Note that the equality-constrained filters are employed to yield estimates $\hat{x}^{p,s}_k$ and $P^{xxp,s}_k$ (2.92) satisfying (4.5). In fact, such procedure does not guarantee that (4.5) is satisfied in the *combining the estimates* step of IMM because the estimates provided by the filter bank that satisfy different equality constraints are combined.

The CIHE employs equality-constrained filters to enforce the equality constraint (4.5) on the state vector; see marker • in Figure 4.1. For convenience, we review the CIHE algorithm [Mann, G., Hwang, I., 2013] in this section as follows.

Algorithm 4.3.1. *Constrained innovation hybrid estimator [Mann, G., Hwang, I., 2013]* *Initialize each sth filter with $\bar{x}^{p,s}_0$, $\bar{P}^{xxp,s}_0$, $\gamma^s_0, s = 1, \ldots, M$ and the parameters of the linear equality constraints D_{m_k} and d_{m_k}.*

1. *Perform steps 1 to 2 of the IMM filter (Algorithm 2.3.1).*

2. *Replace $\bar{x}^s_{k-1}$ by $\bar{x}^{p,s}_{k-1}$ and $\bar{P}^{xx,s}_{k-1}$ by $\bar{P}^{xxp,s}_{k-1}$ in (2.92) and obtain the constrained state estimates $\hat{x}^{p,s}_k$ and $P^{xxp,s}_k$ that satisfy the equality constraint (4.5) using the Constrained innovation Kalman filter algorithm; see Section 2.3.5 for details.*

3. *Perform steps 4 to 5 of the IMM filter.*

□

4.3.2 Combining Constrained Filters and Constrained Combined Estimates

Now, an issue concerning whether or not it is necessary to enforce the same constraint in others steps of IMM filter arises in the second case. We investigate the use of an additional step after the *combined estimates* step of the IMM filter to obtain constrained state estimates $\hat{x}^p_k$ and P^{xxp}_k that satisfy the equality constraint (4.5).

The proposed approach is called equality-constrained IMM (CIMM_1). The CIMM_1 algorithm projects the combined estimate $\hat{x}_k$ of the IMM filter given by (2.95) onto some of the hyperplanes defined by the linear equality constraint (4.5) by means of a *projection* step. Also, the CIMM_1 employs a filter bank composed by equality-constrained filters. Note that, for the filter estimates, we consider that even if each process model (4.2) is compatible with the equality constraint (4.5), we do not guarantee that the initial condition satisfies such constraint after a mode transition. Thus, it is necessary to enforce (4.5) after the *data-assimilation* step (in both $\hat{x}_k^{\mathrm{p},s}$ and $P_k^{\mathrm{xxp},s}$ (2.92)). Likewise, we consider that it is necessary to use the *projecting the estimates* step to obtain combined estimates, $\hat{x}_k$ (2.95) and P_k^{xx} (2.96), that satisfy the equality constraint (4.5). The CIMM_1 employs the *prediction and correction of estimates* step and the *projecting the estimates* step to enforce the equality constraint (4.5) on the state vector; see marker ■ in Figure 4.1.

In the projection step, we investigate how to choose the equality constraint (4.5) associated with each operating mode to be enforced to obtain the constrained state estimate $\hat{x}_k^{\mathrm{p}}$ and corresponding covariance P_k^{xxp}. We present two proposals to choose an equality constraint. First, we chose the equality constraint (4.5) associated with the most likely mode *a posteriori*. Next, we project the combined estimates, $\hat{x}_k$ and P_k^{xx}, in this equality constraint using the equations (2.52)-(2.57). The projected estimate is given by

$$i = \arg\max_{s}(\gamma_k^s), \tag{4.11}$$

$$\{\hat{x}_k^{\mathrm{p}}, P_k^{xx\mathrm{p}}\} = \mathrm{P}rojection_i\left(\hat{x}_k, P_k^{xx}, D_i, d_i\right), \tag{4.12}$$

where γ_k^s, $\hat{x}_k$ and P_k^{xx} are given by (2.93), (2.95) and (2.96), respectively, and each "Projection" uses the projection step of ECKF (Algorithm 2.1.5) with the corresponding equality constraint. However, for the first proposal, consider the case in which, during the transition between modes, the modes probabilities, γ_k^s, are approximately equal. In this case, this first proposal does not look very appealing. Alternatively, we project the combined estimate, $\hat{x}_k$ (2.95) and P_k^{xx} (2.96), in each one of the equality constraints (4.5). Next, we choose the final constrained estimate, $\hat{x}_k^{\mathrm{p}}$ and P_k^{xxp}, as the projected estimate closest to the combined estimate $\hat{x}_k$ (2.95). In this work, the proposed approach CIMM_1

employs the second approach to choose an equality constraint as detailed next.

Figure 4.1 summarizes the CIMM_1 algorithm in six recursive steps. The proposed CIMM_1 filter has the same form of the original IMM except for the replacement of the Kalman filers by the equality-constrained filters in the *analog state prediction and correction* step and the additional *projection* step after the *combining the estimates* step. In the CIMM_1 filter, the six step are: the *mode probability prediction* step, the *mixing of estimates* step, the *analog state prediction and correction* step, the *mode probability correction* step, the *combining the estimates* step and the *projecting the estimates*. In this algorithm, the two terms of the right-hand side of (1.4) are treated as follows. The first term, $\rho(x_k \mid m_{1:k}, y_{1:k})$, is addressed in the *mixing of estimates* and the *prediction and correction of estimates* steps, whereas, the second term, $\rho(m_{1:k} \mid y_{1:k})$, is processed in the *mode probability prediction* step and updated in the *mode probability correction* step. We assume that the analog state estimates are obtained from the weighted combination of the recursive estimates of the filter bank, whereas, the digital state estimate is obtained as being the most likely *a posteriori* mode. The CIMM_1 algorithm is described as follows.

Algorithm 4.3.2. *CIMM_1 filter*

Initialize each sth filter with the state vector estimate $\bar{x}_0^{\mathrm{p},s}$, the covariance matrix $\bar{P}_0^{\mathrm{xx},s}$, the mode probability $\gamma_0^s, s = 1, \ldots, M$ and the parameters of the linear equality constraints D_{m_k} and d_{m_k}.

1. *Perform steps 1 to 2 of the IMM filter (Algorithm 2.3.1).*

2. *Replace $\bar{x}_{k-1}^s$ by $\bar{x}_{k-1}^{\mathrm{p},s}$ and $\bar{P}_{k-1}^{\mathrm{xx},s}$ by $\bar{P}_{k-1}^{\mathrm{xxp},s}$ in (2.92) and obtain the constrained state estimates $\hat{x}_k^{\mathrm{p},s}$ and $P_k^{\mathrm{xxp},s}$ that satisfy the equality constraint (4.5) using constrained state estimation methods for linear systems.*

3. *Compute γ_k^s, $\hat{x}_k$ and P_k^{xx} according to equations (2.93)-(2.96).*

4. *Projecting the estimates. First, project the combined estimates, $\hat{x}_k$ (2.95) and P_k^{xx} (2.96), according to the equality constraints (4.5) yielding $\check{x}_k^{\mathrm{p},s}$ and $\check{P}_k^{\mathrm{xxp},s}$ by means of the or-*

thogonal projector $\mathcal{P}_{\mathcal{N}(D_{m_k})}$ given by (2.50) as

$$\check{x}_k^{\mathrm{p},s} = \mathcal{P}_{\mathcal{N}(D_{m_k})}\hat{x}_k, \tag{4.13}$$

$$\check{P}_k^{\mathrm{xxp},s} = \mathcal{P}_{\mathcal{N}(D_{m_k})}P_k^{\mathrm{xx}}, \tag{4.14}$$

where $\check{x}_k^{\mathrm{p},s}, \check{P}_k^{\mathrm{xxp},s}$, $s = 1, \ldots, M$. Next, choose the projected estimate, $\check{x}_k^{\mathrm{p},s}$, closest to the combined estimate $\hat{x}_k$ (2.95) and P_k^{xx} (2.96) to obtain the constrained state estimate $\hat{x}_k^{\mathrm{p}}$ and P_k^{xxp} as

$$s^* = \arg\min_s ||\hat{x}_k - \check{x}_k^{\mathrm{p},s}||^2,$$

$$\hat{x}_k^{\mathrm{p}} = \check{x}_k^{\mathrm{p},s*}, \tag{4.15}$$

$$P_k^{\mathrm{xxp}} = \check{P}_k^{\mathrm{xxp},s*} \tag{4.16}$$

Increment k and return to step 1.

□

The next proposition shows that the constrained state estimate $\hat{x}_k^{\mathrm{p}}$ given by (4.15) minimizes the mean-squarded error satisfying at least one of the equality constraint (4.5). Then, the CIMM$_1$ algorithm to yield state estimates satisfying (4.5).

Proposition 4.3.1. *Consider Problem 2 and let $p(x_k)$ denote the a posteriori probability density for x_k provided by the IMM filter as the Gaussian mixture:*

$$p(x_k) = \sum_{s=1}^{M} \gamma_k^s \mathcal{N}(\check{x}_k^{p,s}, \check{P}_k^{\mathrm{xx}p,s}).$$

where γ_k^s is given by (2.93), $\check{x}_k^{p,s}$ and $\check{P}_k^{\mathrm{xxp},s}$ are given by (4.13) and (4.14), respectively. Then, $\hat{x}_k^p$ given in (4.15) is the state estimate that minimizes the mean-squared error subject to the satisfaction of at least one equality constraint, i.e., (4.17)

$$\hat{x}_k^{\mathrm{p}} = \underset{\bar{x}_k : D_{m_k}\bar{x}_k = d_{m_k} \;\; for\ some\ m_k}{\arg\min} \mathrm{E}_p\left[||x_k - \bar{x}_k||^2\right] \tag{4.17}$$

Proof. *Recalling that $E[\|X-a\|^2] = Var(X) + \|E[X]-a\|^2$, for a constant a, we have that*

$$\underset{\bar{x}_k : D_{m_k}\bar{x}_k = d_{m_k}}{\arg\min} \mathrm{E}_p\left[||x_k - \bar{x}_k||^2\right] = Var_p(x_k) + \min_{m_k} \min_{\substack{\bar{x}_k \\ \bar{x}_k : D_{m_k}\bar{x}_k = d_{m_k}}} E_p[\|x_k - \bar{x}_k\|^2] \tag{4.18}$$

$$= Var_p(x_k) + \min_{m_k} \|E_p[x_k] - \mathcal{P}_{\mathcal{N}(D_{m_k})}E_p[x_k]\|^2 \tag{4.19}$$

Note that (4.19) has the same form of (4.15). In the last term on the right-hand side of (4.19), we have used the fact that the Euclidean norm minimizer is given by the orthogonal projection. From this we have that the minimum in (4.17) is achieved by the minimizer in (4.15).

□

4.4 General Nonlinear Case

We now consider Problem 3. We discuss an approach that provides approximate solutions to the equality constrained state estimation problem for nonlinear systems. These approaches do not guarantee that the nonlinear equality constraint (1.5) is exactly satisfied, but they provide approximate solutions.

This case is more complicated because the problem is nonlinear and the linear algebra tools used before only work in an approximate form. In this case, we obtain the nonlinear extension of CIMM_1 (Algorithm 4.3.2), here called CIMM_2, to solve this problem. That is, we use constrained state estimation methods for nonlinear systems, for example the equality-constrained unscented Kalman filter (ECUKF) (Section 2.1.4), during the *prediction and correction of estimates* step and the *projecting the estimates*. We replace the equations of the *prediction and correction of estimates* step given by (2.92) and the *projecting the estimates* step given by (4.15) by the equations of the ECUKF whose *forecast* step is given by (2.67), (2.55)-(2.57) and the *data-assimilation* step is given by (2.14)-(2.16). The nonlinear CIMM_2 algorithm is described as follows.

Algorithm 4.4.1. *CIMM_2 filter*
Initialize each sth filter with the state vector estimate $\bar{x}_0^{\mathrm{p},s}$, the covariance matrix $\bar{P}_0^{\mathrm{xxp},s}$, the mode probability $\gamma_0^s, s = 1, \ldots, M$ and the parameters of the nonlinear equality constraints $g_{m_k}(x_k, k)$ and d_{m_k}.

1. *Perform steps 1 to 2 of the IMM filter (Algorithm 2.3.1).*

2. *Prediction and correction of estimates. Use constrained state estimation methods for nonlinear systems (such as the ECUKF given by Algorithm* **??***) to obtain $\hat{x}_k^{\mathrm{p},s}$ and $P_k^{\mathrm{xxp},s}$ where the nonlinear equality constraints (1.5) is approximately satisfied.*

3. *Perform steps 4 to 5 of the IMM filter.*

4. *Perform step 4 of the CIMM$_1$ algorithm using constrained state estimation methods for nonlinear systems. Increment k and return to step 1.*

□

4.5 Simulated Examples

For the three case studies we discuss next, we performed 100 distinct realizations using the proposed approaches and the classical IMM in a desktop Dell Inspiron Intel Core i3 with 3.2 GHz processor and 4GB RAM memory running Matlab.

4.5.1 Linear and mode-independent equality constraint case: Water tank system

Problem description

Consider the water tank system of [Mann, G., Hwang, I., 2013] to illustrate an application of Problem 1. From Figure 4.2, note that it comprises two tanks coupled by one on-off valve. The tank 1 has a water input q_1 provided by the pump 1. Water flow q_2 flows from tank 1 to tank 2 and is manipulated by the valve V_1. Therefore, the reconfigurable system has two operating modes, $m_k \in \mathbb{M} = \{1, 2\}$, representing, respectively, the closed and open status of the valve. Our goal is to estimate the digital state (valve position), $\hat{m}_k$, and the analog states (water level), $\hat{x}_k$ with corresponding covariance matrix, P_k^{xx}.

Let $x_k = \begin{bmatrix} h_{1,k} & h_{2,k} \end{bmatrix}^T$ be the analog state vector associated to the reconfigurable tank system, where h_1 and h_2 are water levels in tanks 1 and 2, respectively. The process model for the mode $m_k = 1$ (closed valve) is given by

$$x_k = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x_{k-1} + \begin{bmatrix} T & 0 \\ -T & 0 \end{bmatrix} u_{k-1} + w_{k-1}. \tag{4.20}$$

where $T = 0.5$ s is the sampling period and $u_{k-1} = \begin{bmatrix} q_1 & q_2 \end{bmatrix}^T$, q_1=1 m/s and q_2=2 m/s for both modes. The process noise, $w_{k-1} \sim \mathcal{N}(0, Q_{k-1})$, is assumed to be a zero-

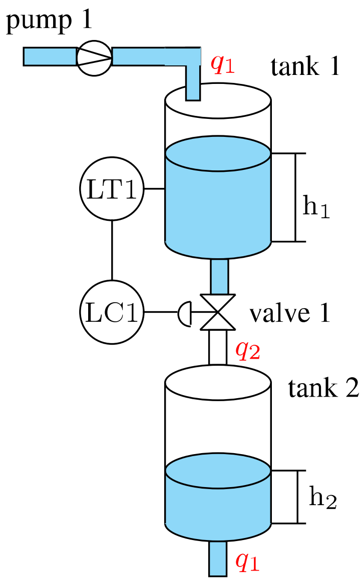

Figure 4.2: Water tank system consisting of two coupled tanks that are controlled by one on-off valve. The pump 1 provides water for the tank 1 and the tank 1 provides water for the tank 2. The level of the tank 1 is measured by the sensor LT1. Adapted from [Mann, G., Hwang, I., 2013].

mean white noise with covariance $Q_{k-1} = 0.125 \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$ for both modes. The process model for $m_k = 2$ (opened valve) is given by

$$x_k = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x_{k-1} + \begin{bmatrix} T & -T \\ -T & T \end{bmatrix} u_{k-1} + w_{k-1}. \tag{4.21}$$

Note that, the models associated with modes 1 and 2 are compatible with the equality constraint (4.22) because they satisfy the conditions of Lemma 2.1.1.

In addition, we assume that the state vector satisfies the equality constraint (4.4) for both modes, whose parameters are assumed to be known and are given by

$$D = \begin{bmatrix} 1 & 1 \end{bmatrix},\ d = 26, \tag{4.22}$$

that is, the mass conservation is observed, meaning that the sum of the levels of the two tanks remain constant.

The observation model for both modes is given by

$$y_k \;=\; \begin{bmatrix} 1 & 0 \end{bmatrix} x_k + v_k, \tag{4.23}$$

where the measurement noise, $v_k \sim \mathcal{N}(0, R_k)$, is assumed to be zero-mean white noise with constant covariance and $R_k = 0.2$. We assume that the transition probability matrix is given by

$$\Pi = \begin{bmatrix} 0.9 & 0.1 \\ 0.1 & 0.9 \end{bmatrix}. \tag{4.24}$$

We also assume the water level starts from $x_0 = \begin{bmatrix} 20 & 6 \end{bmatrix}^T$. In order to test the robustness of our filtering strategy, the actual simulated systems had state-dependent transitions rather than following the Markov jump process in (4.24). These transitions are given by the on-off controller (LC1) as: for mode 1, $h_1 \leq \bar{h} - \epsilon$ then the valve is closed and, for the mode 2, $h_1 \geq \bar{h} + \epsilon$ then the valve is opened, where we set $\bar{h} = 10$ m and $\epsilon = 0.5$. Note that, the level h_1 of the first tank initially causes the valve V_1 to open and to remain open until $h_1 = \bar{h} - \epsilon$.

State estimation

The filter bank of the classical IMM and IMM$_0$ (see Proposition 4.2.1) algorithms is composed by two Kalman filters (KF). We performed 100 distinct realizations using the classical IMM and IMM$_0$ in a desktop Dell Inspiron Intel Core i3 with a 3.2 GHz processor and a 4GB RAM memory running Matlab.

a) Comparison over multiple realizations

We begin by presenting results for multiple realizations of the water tank system. Overall performance was assessed averaging the results from 100 distinct process realizations. We obtain RMSE indices of 0.25 m and 0.11 m, respectively, for the water levels estimates using the classical IMM and IMM$_0$ (see Proposition 4.2.1) filters, with each filter bank being composed by two KFs. In this case, the IMM$_0$ provides more accurate estimates of approximately 14 cm for the water levels of tanks 1 and 2 than does the classical IMM. We observe that the additional information regarding the initial condition provided by the equality constraint is helpful to improve the IMM estimates.

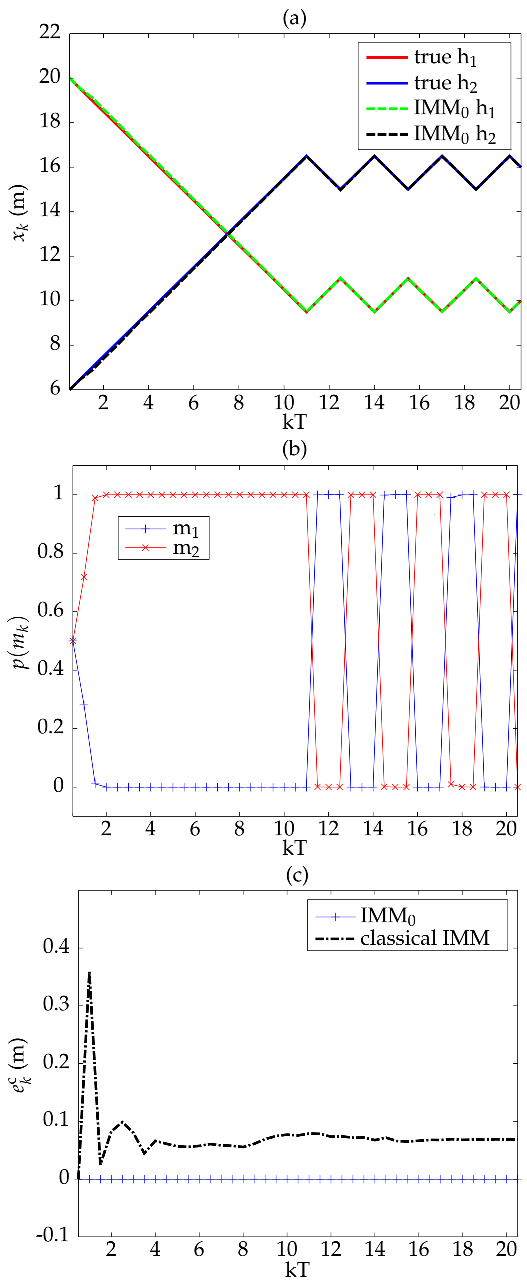

Figure 4.3: Water tank system: (a) analog state estimate using the IMM_0 (- -) in comparison with the true value (—), (b) probability of the mode estimates: mode 1 (–+–) and mode 2 (–×–) using the IMM_0 algorithm and (c) constraint error, $e_k^c = d - D\hat{x}_k$, using the IMM_0 (–+–) and the classical IMM (-.-).

b) Comparison within a single realization

Figure 4.3 shows the simulation results for one realization of the water tank system using the IMM_0. Figure 4.3a shows the estimate of the water level in tanks 1 (green) and 2 (black) compared to the true water level of the tanks. Regarding the water level estimates, the IMM_0 yields more accurate estimates than classical IMM. Note that, initially the water level of the first tank is decreasing until the level reaches $\bar{h} - \epsilon$ whereas the water level of the second tank is increasing until the level reaches $\bar{h} + \epsilon$. The water level in the tanks is controlled by the valve that switches the position on and off such that $h_1 \approx \bar{h}$. Figure 4.3b shows the probability associated to each estimated operating mode. Figure 4.3c shows that the combined estimates, $\hat{x}_k$ (2.95) using the IMM_0 guarantee that the equality constraint (4.4) is satisfied. However, the classical IMM does not yield estimates that satisfy the equality constraint.

4.5.2 Linear and mode-dependent equality constraint case: Tracking a ground vehicle

In a recent work, [Mann, G., Hwang, I., 2013] adresses the problem of hybrid stochastic filtering with mode-dependent equality constraints. However, in their simulated results, we observe that the numerical example illustrates the mode-independent equality constraints case (Problem 1). As illustrated in Section 4.5.1, we know that, for this example, the IMM_0 algorithm (Proposition 4.2.1) can be used and it is not necessary to apply the CIHE approach (Algorithm 4.3.1). Next, we investigate a ground vehicle tracking application in the hybrid context considering that the equality constraints vary with the mode m_k.

Problem description

To illustrate Problem 2, we now consider the problem of tracking a ground vehicle moving in straight roads [Teixeira, B. O. S., Chandrasekar, J., Palanthandalam-Madapusi, H. J., Tôrres, L. A. B., Aguirre, L. A., Bernstein, D. S., 2008]. Let the ground vehicle have two operating modes, $m_k \in \mathbb{M} = \{1, 2\}$, where $m_k = 1$ corresponds to a fixed heading $\theta_1 = 45°$ and $m_k = 2$ corresponds to a heading $\theta_2 = -45°$. Let $x_k = \begin{bmatrix} p_x & p_y & v_x & v_y \end{bmatrix}^T$ be the analog state vector referring to the vehicle trajectory, where p_x and p_y are the components of the position and v_x and v_y are the components of the velocity. Our goal is to obtain estimates for both the digital state, $\hat{m}_k$, and analog states, $\hat{x}_k$ with corresponding covariance matrix, P_k^{xx}.

The linear process models for the ground vehicle as a function of m_k are given by

$$x_k = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} x_{k-1} + \begin{bmatrix} 0 \\ 0 \\ T\sin\theta_{m_k} \\ T\cos\theta_{m_k} \end{bmatrix} u_{k-1} + w_{k-1}, \tag{4.25}$$

where T=2 s is the sampling period and u_{k-1} is the commanded acceleration, which is alternatively set to ± 1 m/s^2, as if the vehicle were accelerating and decelerating in traffic. The process noise, $w_{k-1} \sim \mathcal{N}(0, Q_{m_k})$, is assumed to be a zero-mean white noise with covariance Q_{m_k} that depends on the operating modes and it is given by

$$Q_{m_k} = 10 \begin{bmatrix} \mathrm{s}_\theta & \mathrm{sc}_\theta & 0 & 0 \\ \mathrm{sc}_\theta & \mathrm{c}_\theta & 0 & 0 \\ 0 & 0 & \mathrm{s}_\theta & \mathrm{sc}_\theta \\ 0 & 0 & \mathrm{sc}_\theta & \mathrm{c}_\theta \end{bmatrix}, \tag{4.26}$$

where $\mathrm{s}_\theta = \sin^2\theta_{m_k}$, $\mathrm{c}_\theta = \cos^2\theta_{m_k}$, $\mathrm{sc}_\theta = \sin\theta_{m_k}\cos\theta_{m_k}$. Note that, the models associated with modes 1 and 2 are compatible with the equality constraint (4.27). In addition, we assume that the state vector satisfies the equality constraint (4.5) for each mode, whose parameters are assumed to be known and are given by

$$D_{m_k} = \begin{bmatrix} 0 & 0 & 1 & -\tan\theta_{m_k} \end{bmatrix}, \; d_{m_k} = 0, \tag{4.27}$$

that is, the ground vehicle performs a constant velocity motion with a fixed heading.

The observation model for all modes is given by

$$y_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} x_k + \nu_k. \tag{4.28}$$

where the measurement noise, $\nu_k \sim \mathcal{N}(0, R_k)$, is assumed to be zero-mean white noise with constant covariance $R_k = diag([400, 400])$. We assume that the transition probability matrix is given by

$$\Pi = \begin{bmatrix} 0.9 & 0.1 \\ 0.1 & 0.9 \end{bmatrix}. \tag{4.29}$$

Simulated data are generated with the ground vehicle starting from
$x_0 = \begin{bmatrix} 500 & (500/\tan\theta_1) & 30 & (30/\tan\theta_1) \end{bmatrix}^T$ for the mode $m_0 = 1$.

State estimation

We performed 100 distinct realizations using the CIHE and CIMM_1 algorithms in a desktop Dell Inspiron Intel Core i3 with a 3.2 GHz processor and a 4GB RAM memory running Matlab.

a) Comparison over multiple realizations

We begin by presenting results for multiple realizations of the ground vehicle tracking. Overall performance was assessed averaging the results from 100 distinct process realizations. We compare the processing time, T_{CPU}, and RMSE of the position and velocity coordinates using the CIHE and CIMM_1 algorithms.

Table 4.1 compares the processing time per sampling period, T_{CPU}, and RMSE obtained from a 100 distinct realizations using the CIHE and CIMM_1 algorithms. We see that CIMM_1 provides more accurate estimates by approximately 13% and 19%, respectively, for the position and the velocity than CIHE; however, the processing time of CIMM_1 is 20% larger compared to CIHE.

Table 4.1: Mean processing time to run each algorithm from time $k-1$ to time k, T_{CPU}, and RMSE for 100 distinct realizations using CIHE [Mann, G., Hwang, I., 2013], CIMM_1, CIMM_2 and classical IMM [Blom, H. A.P., Bar-Shalom, Y., 1988].

Algorithms	T_{CPU} (ms)	RMSE for position (m)	RMSE for velocity (m/s)
Linear and mode-dependent case (Section 4.5.2)			
CIHE	1.1	8.25	6.97
CIMM_1	1.3	7.18	5.64
General nonlinear case (Section 4.5.3)			
IMM	0.9	8.91	5.18
CIMM_2	1.4	6.92	3.46

b) Comparison within a single realization

Figure 4.4 shows the simulation results for one realization of the ground vehicle tracking system using the CIHE and CIMM_1 approaches. Figure 4.4a shows the vehicle position estimates using the CIHE (blue line) and CIMM_1 (green line). Note that, the vehicle moves at a constant velocity with fixed heading θ_1. After 25 s, it changes the direction of the velocity, then the vehicle moves at a constant velocity with fixed heading θ_2 for 25 s. Figure 4.4b shows the probability for each estimated operating mode. For example, from $kT = 2$ s until $kT = 52$ s, we observe that the mode 1 is estimated at 97% compared to the 3% that corresponds to the vehicle executing the mode 2 using the CIHE and CIMM$_1$. Figure 4.4c shows that the constraines estimates, $\hat{x}_k^{\mathrm{p}}$, (4.15) guarantee that the equality constraint (4.5) is satisfied for each mode using the CIMM$_1$. For the CIHE and classical IMM, the state estimates $\hat{x}_k$ (2.95) does not guarantee that (4.5) is satisfied for all k.

4.5.3 General nonlinear case: Tracking a ground vehicle

Problem description

We now modify the vehicle tracking problem of Section 4.5.2 as in [Xu, L., Rong Li, X., Liang, Y., Duan, Z., 2017] to illustrate Problem 3.

To illustrate this application, we consider the linear process models for the first mode, $m_k = 1$, corresponding to the vehicle with fixed heading $\theta = 45°$ given by

$$x_k = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} x_{k-1} + \begin{bmatrix} 0 \\ 0 \\ T\sin\theta \\ T\cos\theta \end{bmatrix} u_{k-1} + w_{k-1}, \tag{4.30}$$

where T=2 s is the sampling period and u_{k-1} is the commanded acceleration, which is alternatively set to ± 1 m/s^2, as if the vehicle were accelerating and decelerating in traffic. The process noise, $w_{k-1} \sim \mathcal{N}(0, Q_1)$, is assumed to be a zero-mean white noise with covariance

$$Q_1 = 10 \begin{bmatrix} \mathrm{s}_\theta & \mathrm{sc}_\theta & 0 & 0 \\ \mathrm{sc}_\theta & \mathrm{c}_\theta & 0 & 0 \\ 0 & 0 & \mathrm{s}_\theta & \mathrm{sc}_\theta \\ 0 & 0 & \mathrm{sc}_\theta & \mathrm{c}_\theta \end{bmatrix}, \tag{4.31}$$

where $\mathrm{s}_\theta = \sin^2\theta$, $\mathrm{c}_\theta = \cos^2\theta$, $\mathrm{sc}_\theta = \sin\theta\cos\theta$. In addition, for $m_k = 1$, we assume that the state vector satisfies the linear equality constraint (4.5), whose parameters are assumed to be known and are given by

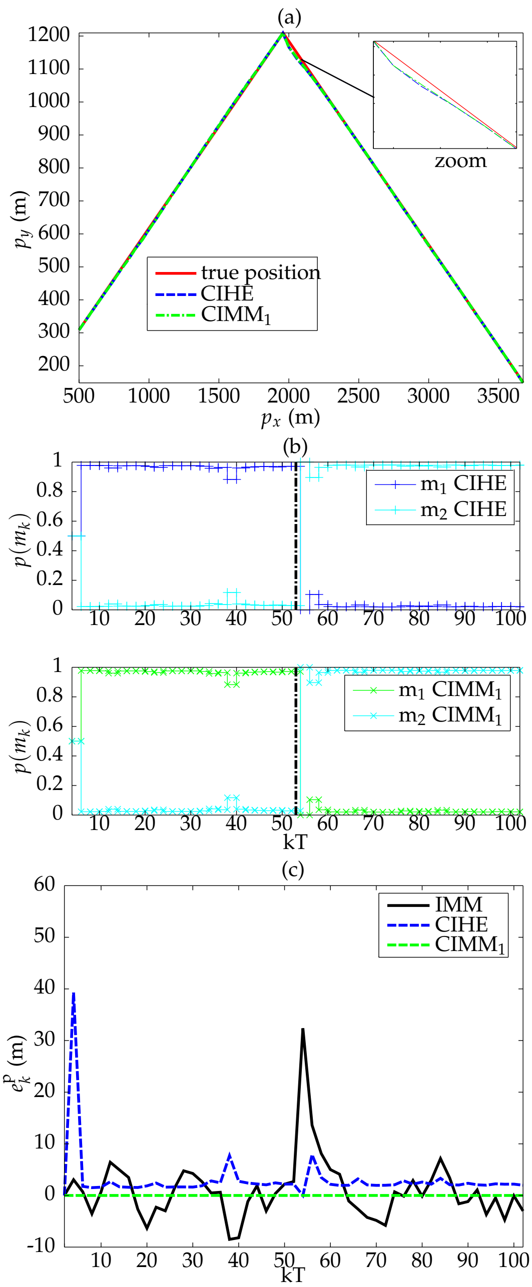

Figure 4.4: Tracking a ground vehicle: (a) estimates of the position coordinates using the CIHE (- -) and CIMM_1 (-.-) in comparison with the true value (—), (b) probability of the modes: mode 1 and mode 2 using the CIHE and CIMM_1 and (c) constraint error, $e_k^p = d - D\hat{x}_k^{\text{p}}$ using the CIMM_1 and $e_k^p = d - D\hat{x}_k$ using the CIHE and IMM.

$$D_{m_k} = \begin{bmatrix} 0 & 0 & 1 & -\tan\theta_{m_k} \end{bmatrix},\ d_{m_k} = 0, \tag{4.32}$$

that is, the ground vehicle performs a constant velocity motion with a fixed heading.

The process models for the second mode, $m_k = 2$, corresponds to the vehicle with constant speed. The process model for this mode is

$$x_k = \begin{bmatrix} p_{x,k-1} + Tv_{x,k-1} \\ p_{y,k-1} + Tv_{y,k-1} \\ v_{x,k-1}s/s_k^a \\ v_{y,k-1}s/s_k^a \end{bmatrix} + w_k. \tag{4.33}$$

where $s = 15$ m/s and $s_k^a = \sqrt{(v_{x,k-1})^2 + (v_{y,k-1})^2}$. The process noise, $w_{k-1} \sim \mathcal{N}(0, Q_2)$, is assumed to be a zero-mean white noise with covariance

$$Q_2 = 10^2 \begin{bmatrix} \frac{T^3}{3} & 0 & \frac{T^2}{2} & 0 \\ 0 & \frac{T^3}{3} & 0 & \frac{T^2}{2} \\ \frac{T^2}{2} & 0 & T & 0 \\ 0 & \frac{T^2}{2} & 0 & T \end{bmatrix}. \tag{4.34}$$

In addition, for $m_k = 2$, we assume that the state vector satisfies the nonlinear equality constraint (1.5) and is given by

$$v_x^2 + v_y^2 = s^2. \tag{4.35}$$

that is, the vehicle moves with a constant speed s.

The observation model for all modes is given by

$$y_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} x_k + \nu_k. \tag{4.36}$$

where the measurement noise, $\nu_k \sim \mathcal{N}(0, R_k)$, is assumed to be zero-mean white noise with constant covariance for each mode given by $R_1 = diag([400, 400])$ and $R_2 = diag([1225, 1225])$. We assume that the transition probability matrix is given by

$$\Pi = \begin{bmatrix} 0.9 & 0.1 \\ 0.1 & 0.9 \end{bmatrix}. \tag{4.37}$$

We also assume the vehicle starts from $x_0 = \begin{bmatrix} 500 & (500/\tan\theta) & 30 & (30/\tan\theta) \end{bmatrix}^T$ for the mode $m_0 = 1$.

[Kwon, C., Hwang, I., 2016] present a simplified version of the vehicle tracking problem to illustrate an application of Problem 3. This example shows that the road map information can be used to enforce an equality constraint on the state vector for linear systems.

For simplicity, we use two operating modes where in the first mode, the process model is linear and in the second mode, the process model is nonlinear. However, different mathematical models of target motion can be used for maneuvering target tracking, for instance, constant velocity models, constant turn models and accelerated models. We suggest the following reference for more details on such models [Rong Li, X., Jilkov, V. P., 2003].

State estimation

In this example, the filter bank of the CIMM_2 algorithm is composed by one equality-constrained Kalman filter (ECKF) and equality-constrained unscented Kalman filter (ECUKF). Note that, the equality constraint is linear for the first mode whereas the equality constraint is nonlinear for the second mode. Then, for each mode, different constrained state estimation methods are employed in the filter bank. We performed 100 distinct realizations using the CIMM_2 and IMM algorithms in a desktop Dell Inspiron Intel Core i3 with a 3.2 GHz processor and a 4GB RAM memory running Matlab.

a) Comparison over multiple realizations

We begin by presenting results for multiple realizations of the ground vehicle tracking. Overall performance was assessed averaging the results from 100 distinct process realizations. We compare the processing time, T_{CPU}, and RMSE of the position and velocity coordinates using the classical IMM and CIMM_2 algorithms.

Table 4.1 compares the T_{CPU} and RMSE obtained from a 100 distinct realizations using the classical IMM and CIMM_2 algorithms, where the filter bank of the CIMM_2 is composed of two filters, namely ECKF and ECUKF for modes 1 and 2, respectively. We observe that CIMM_2 reduces RMSE by approximately 22% and 33%, respectively, for the position and the velocity compared with the classical IMM; however, the processing time of $CIMM_2$ is 1.6 times larger than of IMM.

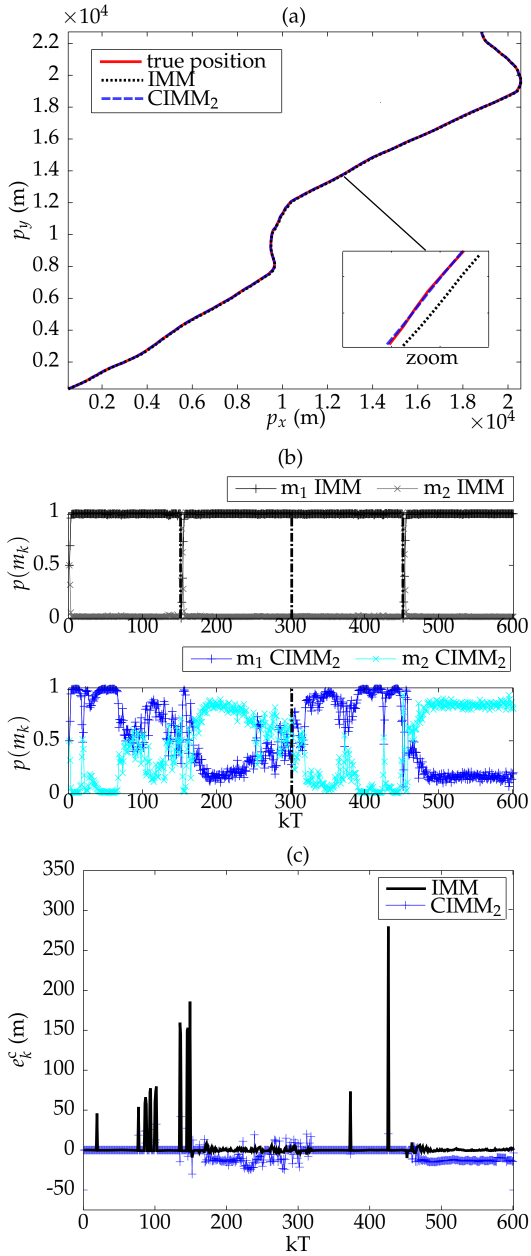

Figure 4.5: Tracking a ground vehicle: (a) analog state estimate using the $CIMM_2$ (- -) and IMM ($\cdots$) in comparison with the true value (—), (b) probability of the mode estimates: mode 1 (—+—) and mode 2 (—×—) using the IMM and $CIMM_2$. The vertical lines indicate mode transition times. (c) constraint error, $e_k^p = d_{m_k} - g_{m_k}(x_k, k)$, using the $CIMM_2$ (—+—) and IMM (—).

b) Comparison within a single realization

Figure 4.5 shows the simulation results for one realization of the ground vehicle tracking system using the IMM and $CIMM_2$. In Figure 4.5a, note that the vehicle performs a constant velocity motion with fixed heading θ and, after some time, the vehicle is moving with constant speed. Figure 4.5b shows the probability for each estimated operating mode using the IMM and $CIMM_2$. For example, from $kT = 2$ s until $kT = 150$ s, we observe that the mode 1 (dark blue line) is estimated during 80% of the time compared to the 20% that corresponds to the vehicle executing the mode 2 (light blue line) using the $CIMM_2$. After the mode transition, we observe that the mode 2 is estimated at 65% compared to the 35% of the mode 1 for 150 s. After this time interval, the modes are alternated. However, the classic IMM wrongly detects that mode 1 (black line) is active for almost all k. Note that, the IMM shows no transition between modes. Figure 4.5c shows that the constrained estimates, $\hat{x}_k^{\mathrm{p}}$ (4.15), do not guarantee that the nonlinear equality constraint (1.5) is exactly satisfied. We observe that the constraint is approximately satisfied using the $CIMM_2$. For the classical IMM, the state estimates $\hat{x}_k$ (2.95) does not satisfy (1.5) for all k.

4.6 Concluding Remarks

In this chapter we discuss the problem of state estimation for hybrid linear and nonlinear systems with state equality constraints. We show three special cases of practical interest of this problem. We categorized such a problem according to the linear or nonlinear character of the equality constraint as well as to the dependence of the constraint on the operating mode.

For the mode-independent equality constraint case, we present the sufficient conditions on the initialization and dynamics for the classical interacting multiple models (IMM) algorithm to yield state estimates satisfying a linear equality constraint for hybrid linear time-invariant systems. Then, the IMM with projected initial condition (IMM_0) is presented as the approximate solution for this problem. We compare these algorithms by means of an examples of a water tank system in which the sum of the levels of the two tanks is constrained so that mass is conserved.

We investigate the scenario in which the mode-dependent time-varying linear equality constraints must be enforced along time for linear and nonlinear systems. In such cases, the equality constraints are different for each operating mode. For this reason, it is necessary to enforce these constraints along time. For such, we propose two algorithms as an approximate solution to the state estimation problem for the linear and nonlinear cases. For linear systems, the proposed CIMM_1 filter is a modified version of the IMM algorithm to yield state estimates satisfying a linear mode-dependent equality constraint. For nonlinear systems, we investigate the scenario in which the nonlinear equality constraint may vary with time and operational mode. For nonlinear systems, the proposed CIMM_2 filter to yields state estimates such that the nonlinear equality constraints are approximately satisfied. We illustrate the application of the proposed approaches by means of an example of tracking a ground vehicle in which the vehicle performs a constant velocity motion with a fixed heading or in which the vehicle moves with a constant speed.

Numerical results suggest that the proposed approaches provide improvement in the accuracy of analog and digital state estimates. The proposed approaches IMM_0, CIMM_1 and CIMM_2 have a smaller estimation error for the analog and digital state compared to the CIHE (special case of CIMM_1) [Mann, G., Hwang, I., 2013] and IMM filters. Moreover, the proposed approaches CIMM_1 and CIMM_2 have required only a slightly longer processing time. The IMM_0 filter has a same processing time than the IMM filter because the proper initialization of IMM_0 is done before the state estimation process.

Chapter 5

Conclusions and Future Work

5.1 Summary and Concluding Remarks

Hybrid systems are dynamic systems whose behavior is determined by the interaction between analog and digital states. The hybrid stochastic filtering problem consists in providing estimates for both analog and digital states from a sequence of noisy sample measurements and the knowledge of such hybrid model. The main hurdle for these systems is that both the number of possible sequences of modes and the number of possible analog trajectories grow exponentially over time, posing a practical challenge to solve this problem.

Approximate methods for state estimation in hybrid systems are based on multiple models (MM) and Monte Carlo approaches. The estimator of MMs assumes that the dynamic system can be characterized by a set of M models that capture the possible operating modes of the system. The estimate provided by the MM estimator is achieved by running M filters in parallel and combining their estimates. Alternatively, particle filtering (PF) methods approximate the joint *a posteriori* PDF of the hybrid system using sampled trajectories. The PF employs a set of particles with corresponding weights to provide estimates for both analog and digital states.

In this work, we addressed the problem of state estimation for discrete-time Markov jump systems. In such systems, the digital state transitions are governed by a Markov chain and transition probabilities depend only on the current mode and not on the analog state of the system.

5.1.1 M^{3}H with Gaussian Mixture Reduction

We considered the problem of state estimation for Markov jump systems. In the M^{3}H filter, the issue of the exponential growth of the number of possible trajectories is tackled by merging hypotheses with similar digital state trajectories. An alternative method for the merging step of the M^{3}H algorithm was discussed and investigated in this work. The proposed M^{3}HR filter leverages techniques from the theory of Gaussian mixture reduction to reduce the approximation error in the merging step. In this way, rather than using only the information from the mode sequence as in the M^{3}H filter, we also incorporate the information from the analog state estimate in performing the merging step.

Numerical results in a target tracking example suggest that the M^{3}HR provides improvement in the accuracy of analog and digital state estimates. In particular, most of the improvement is due to the first of three steps of the proposed algorithm. This motivated the study of abbreviated versions of the algorithm. In summary, when higher precision is demanded from the filter, this abbreviated M^{3}HR outperforms the original M^{3}H in both estimation error and processing time. As a consequence, M^{3}HR presents itself as a filtering strategy that offers the user the flexibility of operating efficiently with different constraints on processing time and precision. The same could not be said of M^{3}H with more stringent precision constraints.

5.1.2 Constrained state estimation

We addressed the problem of state estimation for hybrid linear and nonlinear systems with state equality constraints. We divided such a problem into three groups according to the linear or nonlinear equality constraint as well as to the dependence of the constraint on the operating mode m_k. First, we addressed the mode-independent equality constraints case for linear systems. We presented sufficient conditions on the initialization and dynamics for the classical interacting multiple models (IMM) algorithm to yield state estimates satisfying a linear equality constraint for linear systems. Then, the IMM with projected initial condition (IMM_0) is presented as the approximate solution for this problem. We compared two algorithms by means of an example of a water tank system in which the sum of the levels of the two tanks is constrained so that mass is conserved. We verified that the IMM_0 yields state estimates satisfying the linear equality constraint.

Second, we addressed the mode-dependent equality constrained case for linear systems. In this case, we considered that the equality constraint vary with the mode m_k and for this reason the equality constraint must be enforced along time. Then, we presented a modified version of the IMM filter ($CIMM_3$) to enforce the linear mode-dependent equality constraint on the state estimates for linear systems. We illustrated the application of the proposed approach by means of an example of tracking a ground vehicle in which the vehicle performs a constant velocity motion with a fixed heading determined by the physical road the vehicle is on. We verified that the $CIMM_3$ yields state estimates satisfying the linear equality constraint.

Third, we investigated the mode-dependent equality constrained case for the nonlinear systems. In this case, we considered that the equality constraint may vary with time k and mode m_k. Then, we employed the $CIMM_3$ algorithm as approximate solution of this problem. We compared two algorithms by means of an example similar to the vehicle tracking problem in which the vehicle moves with a constant speed. We verified that the $CIMM_3$ provides state estimates satisfying a linear equality constraint, whereas the nonlinear equality constraint is approximately satisfied.

5.2 Future Work

Some issues for future investigation are presented from the results presented so far. We summarize these suggestions for future work as follows:

1. In Section 3.3 we investigate the use of Gaussian mixture reduction methods as an alternative for the merging step of the M^3H algorithm. Using the Gaussian mixture reduction by clustering approach, the proposed approach M^3HR reduces all hypotheses with the same current mode to a set of N_m hypotheses, where N_m is a tunning parameter, thus making up the maximum number of $N_m M$ hypotheses. We suggest the investigation of strategies to dynamically adapt the maximum number of components in order to further improve the processing time versus error tradeoff.

2. Suboptimal algorithms to adress the problem of state estimation for hybrid systems with state equality constraints are presented in Chapter 4. These algorithms consider that the mode transition probabilities depend only on the current mode and not on the analog state. However, the use of models whose mode transitions depend on the analog state is reported in [Mann, G., Hwang, I., 2013; Kwon, C., Hwang, I., 2016] for the hybrid constrained stochastic filtering problem. We suggest the use of guard conditions to estimate the mode transition probabilities in the proposed algorithms IMM_0 and $CIMM_3$.

3. The optimal estimation problem in networked control systems where the control signal is sent through a lossy channel is presented in [Lin, H., Su, H., Shu, Z., Wu, Z., Xu, Y., 2016]. The use of hybrid models to characterize the behavior of loss of information during the data transmission in communication networks can be investigated. For instance, when the controller does not receive the acknowledgment from the actuator, the controller does not know whether the sent signal has been applied or not, only knows a probability of this occurring. We suggest treating the networked control systems as a new example of application for hybrid stochastic filtering.

4. In this thesis, we address the problem of state estimation for discrete-time Markov jump systems. We suggest the investigation of the state estimation problem for a general class of continuous-time stochastic hybrid systems, which is a more general yet challenging problem than the discrete-time hybrid estimation problem.

In the literature, researches in state estimation for continuous-time stochastic hybrid systems are relatively limited [Liu, W., Hwang, I., 2014].

5. We address the problem of state estimation for hybrid systems with state equality constraints in Chapter 4. We suggest the use of both equality constraints and inequality constraints in hybrid constrained stochastic filtering methods [Cheng, Y., Singh, T., 2007], which are outside the scope of this thesis.

6. Finally, a suboptimal approach to the fixed-interval smoothing problem for hybrid systems with Markov jumps is presented in [Blom, H., Bar-Shalom, Y., 1990; Helmick, R. E., Blair, W. D., Hoffman, S. A., 1995]. This smoothing algorithm is based on Interacting multiple model filter. We suggest the use of our proposed approach M^3HR to address the same problem.

Bibliography

Ackerson, G. A., Fu, K. S. (1970). On state estimation in switching environments. *IEEE Transactions Automatic Control*, AC-15:10–17.

Antsaklis, P. J. (2000). A Brief Introduction to the Theory and Application of Hybrid Systems. *Introductory Article for the Special Issue on Hybrid Systems: Theory and Applications, Proceedings of the IEEE*, 88(7):879–887.

Arulampalam, M. S., Maskell, S., Gordon, N., Clapp, T. (2002). A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2):174–188.

Bach R. (1991). *State estimation applications in aircraft flight data analysis*. Washington, nasa edition.

Bar-Shalom, Y., Challa, S.,Blom, H. A. (2005). IMM Estimator versus Optimal Estimator for Hybrid Systems. *IEEE Transactions on Aerospace and Electronic Systems*, 41(3):986–991.

Bar-Shalom Y., Li X. R., Kirubarajan T. (2001). *Estimation with Applications to Tracking and Navigation*. New York, wiley-interscience edition.

Benazera, E., Travé-Massuyès, L. (2009). Set-theoretic estimation of hybrid system configurations. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 39(5):1277–1291.

Blackman, S., Popoli, R. (1999). *Design and Analysis of Modern Tracking Systems*. Artech House, Norwood, MA.

Blackmore, L., Williams, B. (2008). Active Estimation for Jump Markov Linear Systems. *IEEE Transactions on Automatic Control*, 53(10):2223–2236.

Blom, H. A.P., Bar-Shalom, Y. (1988). The Interacting Multiple Model Algorithm for systems with Markovian Switching Coefficients. *IEEE Transactions on Automatic Control*, 33(8):780–783.

Blom, H., Bar-Shalom, Y. (1990). Time-reversion of a hybrid state stochastic difference system with a jump-linear smoothing application. *IEEE Transactions on Information Theory*, 36(4):836–847.

Boers, Y., Driessen, H (2000). Hybrid state estimation: a target tracking application. *Automatica*, 38:2153–2158.

Boers, Y., Driessen, H. (2003). Interacting multiple model particle filter. *Radar, Sonar and Navigation, IEE Proceedings*, 150(5):344–349.

Boers, Y., Driessen, H (2004). A multiple model multiple hypothesis filter for Markovian switching systems. *Automatica*, 41:709–716.

Candy, J.V. (2009). *Bayesian Signal Processing, Classical, Modern, and Particle Filtering Methods.* University of California Santa Barbara.

Carpenter, J., Clifford, P., Fearnhead, P. (1999). Improved particle filter for nonlinear problems. *Radar, Sonar and Navigation, IEE Proceedings*, 146(1):2–7.

Chaer W. S., Bishop R. H., Ghosh J. (1997). A mixture-of-experts framework for adaptive Kalman filtering. *IEEE Transactions on Systems, Man, and Cybernetics - Part B*, 27(3):452–464.

Chao-Yang, J., Yong-An, Z. (2013). Some results on linear equality constrained state filtering. *International Journal of Control*, 86(12):2115–2130.

Chen, Z. (2003). *Bayesian Filtering: from Kalman Filters to Particle Filters and Beyond.* The Institute of Statistical Mathematics.

Cheng, Y., Singh, T. (2007). Efficient particle filtering for road-constrained target tracking. *IEEE Transactions on Aerospace and Electronic Systems*, 43(4):1454–1469.

Choi, H. H. (2010). Robust Stabilization of Uncertain Fuzzy-Time-Delay Systems Using Sliding-Mode-Control Approach. *IEEE Transactions on Fuzzy Systems*, 18(5):979–984.

Costa, O. L., Guerra, S. (2002). Stationary Filter for Linear Minimum Mean Square Error Estimator of Discrete-Time Markovian Jump Systems. *IEEE Transactions on Automatic Control*, 47(8):1351–1355.

Costa, O.L.V., Fragoso, M.D., Marques, R.P. (2006). *Discrete-Time Markov Jump Linear Systems*. Springer London.

Crouse, D., Willett, P., Pattipati, K., Svensson, L. (2011). A look at Gaussian Mixture Reduction Algorithms. *Proceedings of the International Conference on Information Fusion (FUSION)*, pages 1–8.

Crouse, D., Willett, P., Svensson, L., Svensson, D., Guerriero, M. (2011). The set MHT. *Proceedings of the International Conference on Information Fusion (FUSION)*, pages 1–8.

Daum F. (2005). Nonlinear filters: Beyond the Kalman filter. *IEEE Aerospace and Electronics Systems Magazine*, 20(8):57–69.

Doucet, A., de Freitas, J.F., Gordon, N. (2001). *An introduction to sequential Monte Carlo methods.* New York: Springer-Verlag.

Doucet, A., Gordon, N., Krishnamurthy, V. (2001). Particle filters for state estimation of jump Markov linear systems. *IEEE Transactions on Signal Processing*, 49(3):613–624.

Doucet, A., Johansen, A. M. (2008). *A tutorial on particle filtering and smoothing: fifteen years later.* The Institute of Statistical Mathematics.

Doucet, A., Ristic, B. (2002). Recursive state estimation for multiple switching models with unknown transition probabilities. *IEEE Aerospace and Electronic Systems Magazine*, 38(3):1098–1104.

Driessen, H., Boers, Y. (2001). A multiple model multiple hypothesis filter for tracking maneuvering targets. *Signal and Data Processing of Small Targets 2001, Proceedings of SPIE*, 4473:279–288.

Ebinger, B., Bouaynaya, N., Polikar, R., Shterenberg, R. (2015). Constrained state estimation in particle filters. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4050 – 4054.

Eras-Herrera, W. Y. (2012). Detecção do potencial relacionado à imaginação do movimento usando a filtragem de Kalman. Master's thesis, Universidade Federal de Minas Gerais, Belo Horizonte, Brasil.

Eras-Herrera, W. Y., Mesquita, A. R., Teixeira, B. O. S. (2017). Multiple-model multiple-hypothesis filter with Gaussian mixture reduction. *International Journal of Adaptive Control and Signal Processing*, 32(2):286–300.

Eras-Herrera, W. Y., Mesquita, A. R., Teixeira, B. O. S. (2018). Equality-constrained state estimation for hybrid systems, to be submitted.

Eras, W. Y., Erazo-Costa, F. J., Tierra-Criollo, C. J., Teixeira, B. O. (2012). Detecção do potencial relacionado à imaginação do movimento usando a filtragem de Kalman. *XXIII Congresso Brasileiro em Engenharia Biomédica*, 8(1):312–317.

Fortmann, T., Bar-Shalom, Y., Scheffe, M. (1983). Sonar Tracking of Multiple Targets Using Joint Probabilistic Data Association. *IEEE Journal of Oceanic Engineering*, OE-8(3):173–184.

Goebel, R., Sanfelice, R. G., Teel, A. R. (2009). Hybrid Dynamical Systems. Robust stability and control for systems that combine continuous-time and discrete-time dynamics. *IEEE Control system magazine*, pages 28–93.

Gustafsson, F. (2010). Particle Filter Theory and Practice with Positioning Applications. *IEEE Aerospace and Electronic Systems Magazine*, 25(7):53–81.

Hallouzi, R., Verhaegen, M., Kanev, S. (2009). Multiple model estimation: A convex model formulation. *International Journal of Adaptive Control and Signal Processing*, 23:217–240.

Hanlon, P., Maybeck, P. (2000). Multiple-model adaptive estimation using a residual correlation Kalman filter bank. *IEEE Transaction Aerospace Electronic System*, 36:393–406.

He, X., Wang, Z., Liu. Y., Zhou, D. H. (2013). Least-Squares Fault Detection and Diagnosis for Networked Sensing Systems Using A Direct State Estimation Approach. *IEEE Transactions on Industrial Informatics*, 9(3):1670–1979.

Helmick, R. E., Blair, W. D., Hoffman, S. A. (1995). Fixed-Interval Smoothing for Markovian Switching Systems. *IEEE Transactions on Information Theory*, 41(6):1845–1855.

Hendeby, G., Karlsson, R., Gustafsson, F. (2010). The Rao-Blackwellized Particle Filter: a Filter Bank Implementation. *EURASIP Journal on Advances in Signal Processing*, pages 1–10.

Hide C., Moore T., S. M. (2004). Adaptive Kalman filtering algorithms for integrating GPS and low cost INS. *IEEE Position Location and Navigation Symposium*, pages 227–233.

Ho, Y.C., Lee, R.C.K. (1964). A Bayesian approach to problems in stochastic estimation and control. *IEEE Transactions on Automatic Control*, 9:333–339.

Hofbaur, M. W. (2005). *Hybrid estimation of complex systems*. Graz University of Technology.

Hofbaur, M. W., Williams, B. C. (2004). Hybrid Estimation of Complex Systems. *IEEE Transactions on systems, man, and Cybernetics Part B: Cybernetics*, 34(5):2178–2191.

Hwang, I. , Balakrishnan H., Tomlin, C. (2006). State estimation for hybrid systems: applications to aircraft tracking. *IEE Proceeding - Control Theory and Applications*, 153(5):233–241.

Jazwinski, A. H. (1970). *Stochastic Processes and Filtering Theory*. New York, academic press, inc. edition.

Jilkov, V., Li, X. (2004). Online Bayesian Estimation of Transition Probabilities for Markovian Jump Systems. *IEEE Transactions on Signal Processing*, 52(6):1620–1630.

Julier, S. J., Uhlmann, J. K., Durrant-Whyte, H. F. (2000). A New Method for the Nonlinear Transformation of Means and Covariances in Filters and Estimators. *IEEE Transactions on Automatic Control*, 45(3):477–482.

Julier, S.J., Uhlmann, J.K. (2004). Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92:401–422.

Juloski, A. Lj., Weiland, S., Heemels, W. P. M. H. (2005). A Bayesian Approach to Identification of Hybrid Systems. *IEEE Transactions on Automatic Control*, 50(10):1520–1533.

Kalman, R. E. (1960). A New Approach to Linear Filtering and Prediction Problems. *Transactions of the ASME – Journal of Basic Engineering*, 82(Series D):35–45.

Kawamoto, K. (2010). A Particle Filter with Optimal Discrete Density for Hybrid State Estimation. *International Symposium on Communications and Information Technologies*, pages 296–299.

Kitagawa, G. (1996). Monte Carlo filter and smoother for non-Gaussian nonlinear state space models. *Journal of Computational and Graphical Statistics*, 5(1):1–25.

Ko, S., Bitmead, R.R. (2007). State estimation for linear systems with state equality constraints. *Automatica*, 43(8):1363–1368.

Kottakki, K. K., Bhushan, M., Bhartiya, S. (2014). Interval constrained state estimation of nonlinear dynamical systems using unscented Gaussian sum filter. *Australian Control Conference (AUCC)*, pages 297–302.

Koutsoukos, L., Williams, B. (2003). Estimation of Hybrid Systems Using Discrete Sensors. *42th IEEE Conference on Decision and Control*, 1:155–160.

Kravaritis, G.., Mulgrew, B. (2008). Variable-mass particle filter for road-constrained vehicle tracking. *EURASIP Journal on Advances in Signal Processing*, pages 1–13.

Kwon, C., Hwang, I. (2016). Constrained stochastic hybrid system modeling to road map - GPS integration for vehicle positioning. *Conference on Decision and Control, Las Vegas, USA*, pages 314–319.

Lefebvre, T., Bruyninckx, H., De Schutter, J. (2002). Comment on "A new method for the nonlinear transforrmation of means and covariances in filter and estimators". *IEEE Transactions on Automatic Control*, 47(8):1406–1408.

Li, X. R. (2000). Multiple-model estimation with variable structure Part II: Model-set adaptation,. *IEEE Transaction Aerospace Electronic System*, 45:2047–2060.

Li, X. R., Bar-Shalom, Y. (1996). Multiple-model estimation with variable structure. *IEEE Transactions Automatic Control*, 41:478–493.

Li, X. R., Zhi, X., Zhang, Y. (1999). Multiple-model estimation with variable structure Part III: Model-group switching algorithm. *IEEE Transaction Aerospace Electronic System*, 35:225–241.

Liberzon, D. (2003). *Switching in systems and control*. Birkhauser, Boston, MA.

Lin, H., Su, H., Shu, Z., Wu, Z., Xu, Y. (2016). Optimal estimation in UDP-like networked control systems with intermittent inputs: stability analysis and suboptimal filter design. *IEEE Transactions on Automatic Control*, 61(7):1794–1809.

Liu, J.S., Chen, R. (1998). Sequential Monte Carlo methods for dynamical systems. *Journal of the American Statistical Association*, 93(443):1032–1044.

Liu, K., Yao, Y., Sun, D., Balakrishnan, V. (2012). Improved State Feedback Controller Synthesis For Piecewise-Linear Systems. *International Journal of Innovative Computing, Information and Control*, 8(9):1–11.

Liu, W., Hwang, I. (2012). A stochastic approximation based state estimation algorithm for stochastic hybrid systems. *American Control Conference*, pages 312–317.

Liu, W., Hwang, I. (2014). On hybrid state estimation for stochastic hybrid systems. *IEEE Transactions on Automatic Control*, 59(10):2615–2628.

Lymperopoulos, I., Lygeros, J. (2009). Improved ground trajectory prediction by multi-aircraft track fusion for air traffic control. *AIAA Guidance, Navigation and Control Conference and Exhibit*, pages 1–22.

Mann, G., Hwang, I. (2012). Four-dimensional aircraft taxiway conformance monitoring with constrained stochastic linear hybrid systems. *Journal of Guidance, Control and Dynamics*, 35(5):1593–1604.

Mann, G., Hwang, I. (2013). State estimation and fault detection and identification for constrained stochastic linear hybrid systems. *The Institution of Engineering and Technology IET Control Theory and Applications*, 7(1):1–15.

Margaliot, M. (2006). Stability analysis of switched systems using variational principles: an introduction,. *Automatica*, 42(12):2059–2077.

Maybeck, P. S. (1979). *Stochastic Models, Estimation and Control*. San Diego, academic press, inc. edition.

Mazor, E., Averbuch, A., Bar-Shalom, Y., Dayan, J. (1998). Interacting multiple model methods in target tracking: a survey. *IEEE Transactions on Aerospace and Electronic Systems*, 34(1):103–123.

Mehra R.K. (1972). Approaches to Adaptive Filtering. *IEEE Transactions on Automatic Control*, 17:693–698.

Menegaz, H.M.T., Ishihara, J.Y., Borges, G.A., Vargas, A.N. (2015). A systematization of the unscented Kalman filter theory. *IEEE Transactions on Automatic Control*, 60(10):2583–2598.

Nordlund, T., Gustafsson, F. (2001). Sequential Monte Carlo filtering techniques applied to integral navigation systems. *Proceedings of the American Control Conference*, 6:4375–4380.

Pannetier, B., Benameur, K., Nimier, V., Rombaut, M. (2005). VS-IMM using road map information for a ground target tracking. *8th International Conference on Information Fusion*, 1:1–8.

Rao, C. V., Rawlings, J. B., Lee, J. H. (2001). Constrained Linear State Estimation - A Moving Horizon Approach. *Automatica*, 37(10):1619–1628.

Rao, C. V., Rawlings, J. B., Mayne, D. Q. (2003). Constrained state estimation for nonlinear discrete-time systems: Stability and moving horizon approximations. *IEEE Transactions on Automatic Control*, 48(2):246–258.

Ristic, B., Arulampalam, S., Gordon, N. (2004). *Beyond the Kalman Filter: Particle filters for tracking applications.* Artech House, Boston, MA, USA.

Rong Li, X. (1996). *Hybrid Estimation Techniques*. Lakefront, New Orleans, LA, 76 edition.

Rong Li, X. (2016). Compatitbility and modeling of constrained dynamic systems. *International Conference on Information Fusion*, pages 5–8.

Rong Li, X., Jilkov, V. P. (2003). Survey of maneuvering target tracking. Part I: Dynamic models. *IEEE Transactions on Aerospace and Electronic Systems*, 39(4):1333–1364.

Runnalls, A. R. (2007). Kullback-Leibler approach to Gaussian mixture reduction. *IEEE Transactions on Aerospace and Electronic Systems*, 43(3):989–999.

Salmond, D.J (1990). Mixture reduction algorithms for target tracking in clutter. *Signal and Data Processing of Small Targets*, 1305:434–445.

Santana, P. H. R. Q. A., Borges G. A., Ishihara, J. Y. (2010). Hybrid Data Fusion for 3D Localization Under Heavy Disturbances. *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, pages 2425–2430.

Santana, P. H. R. Q. A., Menegaz, H. M., Borges G. A., Ishihara, J. Y. (2010). Multiple Hypotheses Mixing Filter for Hybrid Markovian Switching Systems. *49th IEEE Conference on Decision and Control*, pages 5080–5085.

Schieferdecker, D., Huber, M. F. (2009). Gaussian mixture reduction via clustering. *Proceedings of the 11th International Conference on Information Fusion (FUSION)*, pages 1536–1543.

Scott, D. W. (1999). *Parameter modeling by minimum L2 error.* PhD thesis, Department of Statistics, Rice University.

Seah, C. E., Hwang, I. (2009). 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.

Shao, X., Huang, B., Lee, J. M. (2010). Constrained Bayesian state estimation - A comparative study and a new particle filter based approach. *International Journal of Control*, 20(5):143:157.

Sigalov, D., Leiter, N., Kalish, N., Oshman, Y. (2012). State estimation in hybrid systems with a bounded number of mode transitions in the Presence of Spurious Measurements. *Information Fusion (FUSION) 15th International Conference*, pages 1955 – 1962.

Simon, D. (2010). Kalman filtering with state constraints: a survey of linear and nonlinear algorithms. *Institution of Engineering and Technology (IET) Control Theory and Applications*, 4(8):1303–1318.

Simon, D., Chia, T. (2002). Kalman filtering with state equality constraints. *IEEE Transactions on Aerospace and Electronic Systems*, 38(1):128–136.

Simon, D. J. (2006). *Optimal State Estimation*. Cleveland State University, wiley-interscience, edition.

Suzdaleva, E., Nagy, I. (2011). Recursive state estimation for hybrid systems. *Applied Mathematical Modelling*, 36:1347–1358.

Tahk, M., Speyer, J. L. (1990). Target tracking problems subject to kinematic constraints. *IEEE Transactions on Automatic Control*, 35(3):324–326.

Teixeira, B. O. S., Aguirre, L. A., Tôrres, L. A. B. (2010). Filtragem de kalman com restrições para sistemas não-lineares: revisão e novos resultados. *Sociedade Brasileira de Automatica - Controle e Automação*, 21:127–146.

Teixeira, B. O. S., Chandrasekar, J., Palanthandalam-Madapusi, H. J., Tôrres, L. A. B., Aguirre, L. A., Bernstein, D. S. (2008). Gain-constrained Kalman filtering for linear and nonlinear systems. *IEEE Transactions on Signal Processing*, 56(9):4113–4123.

Teixeira, B. O. S., Chandrasekar, J., Tôrres, L. A. B., Aguirre, L. A., Bernstein, D. S. (2009). State estimation for linear and non-linear equality constrained systems. *International Journal of Control*, 82(5):918–936.

Teixeira, B. O. S., Tôrres, L. A. B., Aguirre, L. A., Bernstein, D. S. (2010). On unscented Kalman filtering with state interval constraints. *Journal of Process Control*, 20:45–57.

Teixeira, B.O. S., Barbosa, B. H. G., Gomes, L. P., Teixeira, A. F., Aguirre, L. A. (2012). UKF-Based Data-Driven Soft Sensing : A Case Study of a Gas-Lifted Oil Well. (1):918–923.

Tian, Y., Floquet, T., Belkoura, L.,Perruquetti, W. (2011). Algebraic switching time identification for a class of linear hybrid systems. *Nonlinear Analysis: Hybrid Systems*, 5:233–241.

Tugnait, J. (1982). Detection and estimation for abruptly changing systems. *Automatica*, 18:607–615.

Vachhani, P., Narasimhan, S., Rengaswamy, R. (2006). Robust and reliable estimation via Unscented Recursive Nonlinear Dynamic Data Reconciliation. *Journal of Process Control*, 16:1075–1086.

Wang, X., Xu, M., Wang, H., Wu, Y., Shi, H. (2012). Combination of Interacting Multiple Models with the Particle Filter for Three-Dimensional Target Tracking in Underwater Wireless Sensor Networks. *EURASIP Journal on Advances in Signal Processing*, pages 1–16.

Williams, J. L. (2003). *Gaussian mixture reduction for tracking multiple maneuvering targets in clutter.* PhD thesis, Air Force Institute of Technology.

Xu, L., Rong Li, X., Liang, Y., Duan, Z. (2017). Constrained dynamic systems: generalized modeling and state estimation. *IEEE Transactions on Aerospace and Electronic Systems*, 53(5):2594–2609.

Zhang, L., Pan, Q., Chen, T. (2010). State smoothing in Markov jump systems with lagged mode observation. *International Journal of Adaptive Control and Signal Processing*, 24:1005–1020.

Zhang, M., Knedlik, S., Loffeld, O. (2008). An adaptive road-constrained IMM estimator for ground target tracking in GSM networks. *11th International Conference on Information Fusion*, 1:1–8.