Filtrage à incertitudes stochastiques et bornées : application au diagnostic actif en automobile

Quoc Hung Lu

Université Fédérale
Toulouse Midi-Pyrénées

THÈSE

En vue de l'obtention du
DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE
Délivré par l'Université Toulouse 3 - Paul Sabatier

Présentée et soutenue par

Quoc Hung LU

Le 15 mars 2022

FILTRAGE À INCERTITUDES STOCHASTIQUES ET BORNÉES - APPLICATION AU DIAGNOSTIC ACTIF EN AUTOMOBILE.

Ecole doctorale : **SYSTEMES**

Spécialité : **Automatique**

Unité de recherche :
LAAS - Laboratoire d'Analyse et d'Architecture des Systèmes

Thèse dirigée par
Carine JAUBERTHIE et Soheib FERGANI

Jury

M. Andreas RAUH, Rapporteur
M. Tarek RAISSI, Rapporteur
Mme Floriane COLLIN, Examinatrice
M. Alexandre CHAPOUTOT, Examinateur
Mme Louise TRAVE-MASSUYES, Examinatrice
M. Xavier MOREAU, Examinateur
Mme Carine JAUBERTHIE, Directrice de thèse
M. Soheib FERGANI, Co-directeur de thèse

Remerciements

C'est pour moi un plaisir d'avoir terminé ma thèse de doctorat sous la direction de mes directeur/trice de thèse au sein de l'équipe *DIagnostic, Supervision et COnduite* (DISCO) du *Laboratoire d'Analyse et d'Architecture des Systèmes* du CNRS (LAAS-CNRS) de Toulouse.

Mes premiers remerciements s'adressent certainement à Carine Jauberthie et Soheib Fergani, mes encadrants de thèse et mes chers compagnons dans les travaux de recherches qui m'accompagnaient à tous les pas de mon parcour scientifique. Je voudrais ensuite remercier Françoise Le Gall qui a contribué à mon premier article scientifique auprès de mes encadrants de thèse. Mes sincères remerciements sont encore destinés à Louise Travé-Massuyès ayant toujours de bons conseils le long de mon parcours de doctorat, étant un membre du comité de suivi scientifique de ma thèse, membre et Président du Jury de ma soutenance de thèse. Je suis également reconnaissant à Patrick Danès pour ses temps de discussions au sujet du filtrage de Kalman et du filtrage particulaire, ainsi que pour ses renseignements pertinents en tant que membre du comité de suivi scientifique de ma thèse. Je remercie infiniment mes collègues de l'équipe DISCO qui seront toujours dans mon coeur pour leur accueil, leur aide, leur amitié...

J'exprime ici ma reconnaissance à Andreas Rauh, Professeur de Carl Von Ossietzky Universität Oldenburg - Allemange, et Tarek Raïssi, Professeur du Conservatoire National des Arts et Métiers - France, ayant consacré leurs temps dans l'évaluation de mes travaux de thèse en me procurant des remarques très pertinentes et constructives en tant que rapporteurs. Je remercie également Xavier Moreau, Professeur de l'Université de Bordeaux - Laboratoire de l'Intégration du Matériau au Système, Floriane Collin, Maître de Conférence de l'Université de Lorraine - Polytech Nancy et Alexandre Chapoutot, Enseignant chercheur de l'École Nationale Supérieure de Techniques Avancées (ENSTA-Paris) qui m'ont fait l'honneur de participer à mon Jury de thèse.

Je garde enfin mes sentiments et remerciements particuliers à ma famille, mon père et ma mère mes premiers Professeurs de ma vie, ma conjointe

Anh Nguyen et mes deux petits enfants Kha Han et Chanh Minh, étant mes encouragements pour toujours. Je n'oublierai jamais les temps particuliers de l'époque en période de Covid durant mes années de thèse. Merci mes frères et soeurs de la communauté évangélique de Toulouse, merci mes amis vietnamiens à l'Instut de Mathémathiques de Toulouse et de l'Université Toulouse 1 Capitole, merci mon Dieu et merci de tous.

Toulouse, le 01 mars 2022,
Quoc Hung Lu

Contents

List of Figures

List of Tables

List of Algorithms

Notations

$\mathbb{R}^+$	The set of strict positive real scalars, page 24
$\mathbb{R}^{m \times n}$	The set of real $m \times n$ matrices., page 23
$A = (a_{ij})$	Matrix A with entries a_{ij}'s, page 23
A^T	Transpose of matrix A, page 23
A^+	Moore-Penrose pseudoinverse of matrix A, page 27
A^{-1}	Inverse of matrix A, page 27
I_n	Indentity matrix of order n, page 23
$\mathbb{I}(x)$	Indicator function, equals 1 if the condition x holds true and vanishces otherwise, in which x can be a vector of conditions, page 23
$\mathbf{1}$ (or $\mathbf{1}_n$)	The (n-)vector whose components all equal to 1, called the all one vector, page 23
e_i	The i-th standard unit vector which is the i-th column of the identity matrix (with dimension to be precised by the context), page 23
$\langle x, y \rangle$	Inner product of two real vectors x and y, page 25
δ_{ij}	The Kronecker delta, δ_{ij} equals 1 if $i = j$ and vanishes otherwise , page 24
$A \succ 0$	A is a positive definite matrix, page 24
$A \succeq 0$	A is a positive semidefinite matrix, page 24
$N \preceq M$	if and only if $M - N \succeq 0$, M is an upper bound of N, N a lower bound of M, page 29
$\lambda_i(A)$	The i-th eigenvalue of matrix A, page 23
$\sigma_i(A)$	The i-th singular value of matrix A, page 23

$\mathrm{Tr}(A)$	The trace of matrix A, page 23
$\mathrm{diag}(x)$	Operator that returns a diagonal matrix whose diagonal entries are components of the vector $x = (x_1, ..., x_n)^T$ or the n-tupple $x = (x_1, ..., x_n)$, page 24
$\mathrm{Diag}(A)$	Operator that returns a diagonal matrix having the same diagonal as the matrix A, page 24
$\mathrm{Diag}_v(A)$	Operator that returns the diagonal of matrix A as a vector, page 24
$S(n)$	The set of real symmetric matrices of order n, page 24
$S_+(n)$	The set of real symmetric positive semidefinite matrices of order n, page 24
$[x] = [\underline{x}, \overline{x}]$	Real interval, a closed connected subset of $\mathbb{R}$ containing all real numbers t such that $\underline{x} \leq t \leq \overline{x}$, where $\underline{x}$ and $\overline{x}$ are respectively the smallest and largest value of $[x]$, page 27
$[X] = ([x_{ij}])$	Real interval matrix, a matrix with interval entries $[x_{ij}]$'s, containing all real matrices M so that $\underline{X} \leq M \leq \overline{X}$ element-wise, where $\underline{X}$ and $\overline{X}$ are respectively the smallest and largest matrix of $[X]$, denote also $[X] = [\underline{X}, \overline{X}]$, page 27
$\inf([X]) = \underline{X}$	The smallest matrix of $[X]$, defined by $\inf([X]) \triangleq (\inf([x_{ij}]))$, page 27
$\sup([X]) = \overline{X}$	The largest matrix of $[X]$, defined by $\sup([X]) \triangleq (\sup([x_{ij}]))$, page 27
$S([X])$	The set of symmetric matrices belonging to $[X]$, page 30
$S_+([X])$	The set of symmetric positive semidefinite matrices belonging to $[X]$, page 30
$BS([X])$	The set of symmetric upper bounds of $S([X])$, page 30
$BS_+([X])$	The set of symmetric positive semidefinite upper bounds of $S_+([X])$, page 30

Abstract

The problem of filtering applied to automotive diagnostic is studied in this thesis, for linear or nonlinear, discrete-time dynamical systems, in a context of mixed uncertainties, i.e. uncertainties can be stochastic or bounded (in intervals). This context allows us to combine two well-known approaches of filtering: *stochastic* and *set-membership approach*. Through this thesis, we show that they complement rather than compete each other. Two models from the automotive industry are used in the applications along the thesis: *bicycle vehicle model* and *suspension model*.

Mixed filtering methods are first developed and presented in this work, namely *Optimal Upper Bound Interval Kalman Filter (OUBIKF)* and *Reinforced Likelihood Box Particle Filter (RLBPF)*, one is dedicated to linear systems and the other to nonlinear systems. The former is based on *interval Kalman filter* and enhances it by using developed properties and optimization strategy of upper bounds of all admissible covariance matrices belonging to a given interval matrix. The later proposes a general scheme of *box particle filter* and develops a reinforcement methodology to the likelihood computation, the crucial step of the scheme, to enhance the filter performance.

The second part of this thesis is dedicated to fault detection. The previous filters are used and combined with a χ^2-based hypothesis testing method with adaptive degrees of freedom, namely *Adaptive Degrees of Freedom χ^2-statistic* (ADFC), to deal with fault detection in linear or nonlinear systems. It is a *passive* fault detection method enhanced by the *adaptive threshold technique* in the *decision making stage*. This method allows the detection of single or multiple additive faults on the sensors.

In the last part of this work, a methodology of active diagnosis is developed, that is the *ADFC-based Active Fault Diagnosis* (AFD) using *auxiliary signals*. This methodology, a preliminary study to the active approach, is limited to single fault detection. However, its contributions are multiple: *isolation (localization)* and *identification (estimation)* of the fault, reduction of false alarms and improvement of the state estimation by returning the estimated fault as a feedback signal to the filter used. Our future researches focus specifically on this approach.

Résumé

Le problème du filtrage appliqué au diagnostic automobile est étudié dans ce travail de thèse, pour les systèmes dynamiques linéaires ou nonlinéaires, à temps discret, en contexte d'incertitudes mixtes, c'est-à-dire que les incertitudes peuvent être stochastiques ou bornées (dans des intervalles). Ce contexte permet de combiner deux approches bien connues du filtrage : les *approches stochastique* et *ensembliste*. Au travers de cette thèse, nous montrons qu'elles se complètent plutôt qu'elles se concurrencent. Deux modèles issus de l'automobile sont utilisés dans les applications tout-au-long de la thèse. Il s'agit des modèles *de véhicule à bicyclette* et *de suspension*.

Des méthodes mixtes de filtrage sont tout d'abord développées et présentées dans ce travail : *Optimal Upper Bound Interval Kalman Filter (OUBIKF)* et *Reinforced Likelihood Box Particle Filter (RLBPF)*, l'un est dédié aux systèmes linéaires et l'autre aux systèmes nonlinéaires. Le premier se base sur le *filtre de Kalman intervalle* et l'améliore en utilisant les propriétés développées et la stratégie d'optimisation des bornes supérieures de toutes les matrices de covariances admissibles appartenant à une matrice d'intervalle donnée. Le second propose un schéma général de *fitre particulaire ensembliste* et développe une méthodologie de renforcement du calcul de la vraisemblance, l'étape cruciale du schéma, pour améliorer la performance du filtre.

La deuxième partie de cette thèse est dédiée à la détection de défauts. Les filtres précédents sont utilisés et combinés à une méthode de test d'hypothèse basée χ^2 avec les degrés de liberté adaptatifs, à savoir *Adaptive Degrees of Freedom* χ^2*-statistic* (ADFC), pour traiter la détection de défauts dans les systèmes linéaires ou nonlinéaires. Il s'agit d'une méthode de détection de défaut *passive* renforcée par la technique de *seuil adaptatif* dans l'*étape de décision*. Cette méthode permet la détection de défauts additifs, simples ou multiples, sur les capteurs.

Dans la dernière partie de ce travail, une méthodologie de diagnostic actif est développée, à savoir *ADFC-based Active Fault Diagnosis* (AFD) utilisant des *signaux auxiliaires*. Cette méthodologie, étude préliminaire à l'approche active, se limite à la détection de défaut simple. Cependant, ses contributions

sont multiples : *isolement (localisation)* et *identification (estimation) du défaut*, réduction de fausses alarmes et amélioration de l'estimation de l'état en renvoyant le défaut estimé comme un signal de retour au filtre utilisé. Nos futures recherches se concentrent tout particulièrement sur cette approche.

Chapter 1

State of the art

With the growth of the industrial *automatization* and the fast development of intelligent systems applications, the necessity of efficient control strategies has risen to higher levels. Nerveless, the main problems to the synthesis of such solutions have been the cost and the feasibility. Indeed, all efficient control approaches are based on reliable information either from high precision sensors or high fidelity information reconstruction (estimators, observers). The former is considered as *hardware-based approach* which is usually expensive and not always easy to embed. The later is considered as *model-based approach*, since it bases entirely on a mathematical model. This approach is more flexible to control and embed with lower cost. Therefore, the model-based approach is widely used in many applications with numerous purposes, included *system control*, *state prediction/estimation*, *fault diagnosis*. In this chapter, an overview of the state estimation and the fault diagnosis, says the state of the art, is presented to introduce advanced developed contents in later chapters.

1.1 State estimation problem

In the model-based state estimation problem, a dynamical system is considered. It can be *linear* or *nonlinear*, *discrete* or *continuous time*. For the purpose of computer implementation, any continuous time system must be discretized. Therefore, through out this thesis, we focus on discrete time systems for both linear and nonlinear case. For each case, only the relevant contents of the literature involving our researches in next chapters are presented here: *Kalman filter* for linear system, *Bayesian filtering* and *Particle filter* for nonlinear system.

1.1.1 Standard Kalman Filter

State estimation is a topic of utmost importance when dealing with system control. Indeed, obtaining accurate estimations can lead to great improvements in the systems performances. One of the most significant ideas to emerge in the area of system and control theory is the Kalman Filter (Ian R. and Andrey V., 1999). The Kalman Filter was first introduced in (Kalman, 1960) and referred to as *Standard Kalman Filter* (SKF). In this method, the system under consideration is a *linear discrete time-varying* (LTV) system with additive centered Gaussian noises in state and measurement processes. The SKF provides optimal estimates for the real (actual) states and involves finite dimensional recursive computations which can be straightforwardly implemented on-line.

A precursor of the SKF, known as the Wiener Filter, was developed independently by (Wiener, 1949) and (Kolmogorov, 1941). Being also an optimal method of extracting a signal from noise (as well as the SKF), the Wiener Filter is however limited to *time-invariant* system with stationary noise processes and not computationally straightforward as the SKF.

Since SKF has released in 1960, many extensions have been investigated to enhance its performance, included:

— the *Extended Kalman Filter* (EKF) (Anderson and Moore, 1979) to deal with nonlinear system by linearization,

— numerous developments of the SKF to robust methods, says *Robust Kalman Filtering*, to deal with system uncertainty beside stochastic noises (Ian R. and Andrey V. (1999); Zhe and Zheng (2006); Mohamed and Nahavandi (2012a),...).

— *set-membership* methods (using intervals, zonotopes,...) also to deal with system uncertainty beside stochastic noises (Chen et al. (1997); Xiong et al. (2013); Tran et al. (2017); Lu et al. (2019); Combastel (2005, 2015)...).

The reason of these extensions is that the SKF has a good performance while relying on the following assumptions:

+ all parameter matrices A_k, B_k, C_k, D_k of the system are known and there is no other disturbance (than noises) affecting the system,
+ the noises must be Gaussian,

which are ideal and unrealistic in modeling. By here, we would distinguish robust methods from set-membership ones though some authors might consider the later being also robust methods. Both of them deal with system uncertainty beside stochastic noises, however, the former should be those methods providing *point estimates* while the later entirely produces *set-valued estimates* for the real states. These results represent the different objectives of

these methods. Estimation methods should be classified in priority by their estimate results (objectives) rather than the fact that they treat the same kind of uncertainty.

In the sequel, the SKF is presented in essential details. Consider the following linear discrete time dynamical system

$$\begin{cases} x_k = A_k x_{k-1} + B_k u_k + w_k \ , \\ y_k = C_k x_k + D_k u_k + v_k \ , \end{cases} \qquad k \in \mathbb{N}^*, \tag{1.1}$$

in which $x_k \in \mathbb{R}^{n_x}$ and $y_k \in \mathbb{R}^{n_y}$ represent state variables and measurements respectively, $u_k \in \mathbb{R}^{n_u}$ inputs, $w_k \in \mathbb{R}^{n_x}$ state noises and $v_k \in \mathbb{R}^{n_y}$ measurement noises.

Assumptions A0 (SKF Assumptions). Matrices A_k, B_k, C_k, D_k are assumed to be known. w_k, v_k are centered Gaussian vectors with known covariance matrices Q_k and R_k. The initial state x_0 is Gaussian with known mean and covariance matrix: μ_0 and P_0. In addition, x_0, $\{w_1, ..., w_k\}$ and $\{v_1, ..., v_k\}$ are assumed to be mutually independent (or uncorrelated). In terms of mathematical expression, that is

— $x_0 \sim \mathcal{N}(\mu_0, P_0)$, $w_k \sim \mathcal{N}(0, Q_k)$, $v_k \sim \mathcal{N}(0, R_k)$ for any $k \geq 1$,
— $\mathbb{E}[x_0 w_k^T] = \mathbb{E}[x_0 v_k^T] = \mathbb{E}[w_k v_l^T] = 0$ for any $k, l \geq 1$,
— $\mathbb{E}[w_k w_l^T] = Q_k \delta_{kl}$ and $\mathbb{E}[v_k v_l^T] = R_k \delta_{kl}$ for any $k, l \geq 1$,

where δ_{kl} is the Kronecker delta.

Aim. The problem aims to find an estimate $\hat{x}_k|y_{1:k}$ ($\equiv \hat{x}_{k|k}$ for short) given the observed values $\{y_1 : y_k\}$ of the real state x_k with which the expected loss $\mathbb{E}(\|x_k - \hat{x}_{k|k}\|^2)$ is minimized.

Remark 1. The notation $p : l : q$ is used for a range from p to q with step l provided that $p, l, q \in \mathbb{N}$, $p \leq q$ and l is a divisor of $(q - p)$. For $l = 1$, we write $p : q$. A sequence of variables can be noted interchangeably as $y_1, ..., y_k$ or $y_1 : y_k$ or $y_{1:k}$. □

Remark 2. Define $Y_k \triangleq y_{1:k}$ as the knowledge up to time $k \geq 1$ of the system (1.1). By convention, Y_0 is seen as the zero knowledge when no measurement is taken. in terms of σ-algebra, the known information corresponding to Y_0 is $\sigma(Y_0) = \{\emptyset, \Omega\}$, where Ω is the sample space on which the random vectors in consideration are defined. Therefore

$$\begin{aligned} \mathbb{E}[x_p|Y_0] &= \mathbb{E}[x_k|\sigma(Y_0)] = \mathbb{E}[x_p] \ , \quad \forall p \geq 1, \\ \mathbb{E}[x_k|Y_{k-1}] &= \mathbb{E}[x_k|\sigma(Y_{k-1})] \ , \quad \forall k \geq 1, \end{aligned}$$

where $\sigma(Y_{k-1})$ is the σ-algebra generated by Y_{k-1}, $k \geq 1$. Therefore, we write $\mathbb{E}[x_1|Y_0] \equiv \mathbb{E}[x_1|\sigma_0] = \mathbb{E}[x_1]$, where the first two terms are equivalent by notation convention. □

SKF principle. At time $k \geq 1$, the state x_k is first estimated, using a priori knowledge Y_{k-1} (use the convention of Remark 2 when $k = 1$), by $\hat{x}_{k|k-1} \triangleq \mathbb{E}[x_k|Y_{k-1}]$. This is the *prediction stage*. In this stage an approximate $\hat{y}_k$ of measurement y_k is also provided thanks to the second equation of (1.1) without noise v_k. The second stage, namely *correction stage*, will be implemented once y_k arrived. The *a priori estimate* $\hat{x}_{k|k-1}$ will be updated in terms of

$$\begin{aligned} \hat{x}_{k|k} &= \hat{x}_{k|k-1} + K_k r_k \ , \\ r_k &= y_k - \hat{y}_k \ , \\ \hat{y}_k &= C_k \hat{x}_{k|k-1} + D_k u_k = \mathbb{E}[y_k|Y_{k-1}] \ , \end{aligned} \tag{1.2}$$

where r_k is called the *residual* or *innovation* term and $K_k \in \mathbb{R}^{n_x \times n_y}$ is a *gain matrix*. The optimal estimate $\hat{x}_{k|k}$ is obtained by applying to (1.2) the choice of optimal gain $K_k = K_k^*$ with

$$\begin{aligned} K_k^* &= \text{argmin}_{K_k} \, \mathbb{E}(\|x_k - \hat{x}_{k|k}\|^2) \\ &= \text{argmin}_{K_k} \, \text{Tr}\{\mathbb{E}[(x_k - \hat{x}_{k|k})(x_k - \hat{x}_{k|k})^T]\}. \end{aligned} \tag{1.3}$$

Solution. Define $P_{k|k-i} \triangleq \mathbb{E}[(x_k - \hat{x}_{k|k-i})(x_k - \hat{x}_{k|k-i})^T]$, $i \in \{0, 1\}$, and call

— $P_{k|k-1}$ the *a priori estimation (or the prediction) error covariance*,

— $P_{k|k}$ the *(a posteriori) estimation error covariance*.

Using (1.1) and (1.2), the estimation error covariance can be expressed as:

$$\begin{aligned} P_{k|k} &= (I - K_k C_k) P_{k|k-1} (I - K_k C_k)^T + K_k R_k K_k^T, \\ &= P_{k|k-1} - K_k C_k P_{k|k-1} - P_{k|k-1} C_k^T K_k^T + K_k (C_k P_{k|k-1} C_k^T + R_k) K_k^T. \end{aligned} \tag{1.4}$$

The optimal gain K_k^*, if it exists, is solution to the equation $\frac{\partial \text{Tr}\{P_{k|k}\}}{\partial K_k} = 0$, that is

$$K_k^* = P_{k|k-1} C_k^T S_k^{-1} \quad , \quad S_k = C_k P_{k|k-1} C_k^T + R_k \ , \tag{1.5}$$

provided that S_k is nonsingular. It is clearly that S_k is positive semidefinite. Most of the cases S_k^{-1} exists, in particular when R_k is positive definite. A (Moore-Penrose) pseudoinverse S_k^+ is used alternatively in practice when S_k^{-1} does not exist.

The associated estimation error covariance has the form

$$P_{k|k}^* = (I - K_k^* C_k) P_{k|k-1}. \tag{1.6}$$

Finally, the optimal estimate is given by

$$\begin{aligned} \hat{x}_{k|k}^* &= \hat{x}_{k|k-1} + K_k^* (y_k - \hat{y}_k) \\ &= (I - K_k^* C_k) \hat{x}_{k|k-1} + K_k^* (y_k - D_k u_k) \end{aligned} \tag{1.7}$$

Remark 3. All above equations and results hold without an explicit determination of $\hat{x}_{k|k-1} = \mathbb{E}[x_k|Y_{k-1}]$. Lemma 1 provides a version of $\hat{x}_{k|k-1}$ (i.e. the two functions are equal with probability 1, or almost surely) and proves that the $\hat{x}_{k|k}$ expressed in (1.7) is actually a version of $\mathbb{E}[x_k|Y_k]$. □

Lemma 1. *Consider system (1.1) with assumptions A0. Assume that the filter is initialized at* $\hat{x}_{0|0} = \mu_0$. *For* $k \geq 1$, *let* $r_k = y_k - \hat{y}_k$, $\hat{y}_k = C_k\hat{x}_{k|k-1} + D_k u_k$, *define*

$$\hat{x}_{k|k-1} \stackrel{\triangle}{=} \mathbb{E}[x_k|Y_{k-1}] \quad \textit{and} \quad \hat{x}_{k|k} \stackrel{\triangle}{=} \hat{x}_{k|k-1} + K_k^* r_k,$$

where $K_k^* = \operatorname{argmin}_{K_k} \mathbb{E}[\|x_k - (\hat{x}_{k|k-1} + K_k r_k)\|^2] \in \mathbb{R}^{n_x \times n_y}$.
Then with probability 1:

$$\hat{x}_{k|k-1} \;=\; A_k\hat{x}_{k-1|k-1} + B_k u_k \quad \textit{and} \quad \hat{x}_{k|k} \;=\; \mathbb{E}[x_k|Y_k]\,. \tag{1.8}$$

Proof. By assumptions, it holds that :

$$\begin{aligned}
&\hat{x}_{0|0} = \mu_0 = \mathbb{E}(x_0) = \mathbb{E}[x_0|Y_0],\\
&\hat{x}_{1|0} = \mathbb{E}[x_1|Y_0] = A_1\mathbb{E}[x_0|Y_0] + B_1u_1 = A_1\hat{x}_{0|0} + B_1u_1.
\end{aligned}$$

Assume that, at time $k \geq 1$, the following quantities are given

$$\hat{x}_{k-1|k-1} = \mathbb{E}[x_{k-1}|Y_{k-1}] \quad \text{and} \quad \hat{x}_{k|k-1} = \mathbb{E}[x_k|Y_{k-1}].$$

Denote:

$$\begin{aligned}
\mathcal{Y}_p &= \{T_1y_1 + ... + T_py_p : T_1, ..., T_p \in \mathbb{R}^{n_x \times n_y}\}\ ,\ p \geq 1,\\
\mathcal{Y}_{p-1}^{\perp} &= \{u \in \mathcal{Y}_p : \mathbb{E}(uv) = 0, \forall v \in \mathcal{Y}_{p-1}\}\ ,\ p \geq 2.
\end{aligned}$$

It is straightforward to check :

— $\mathcal{Y}_p$'s are Hilbert spaces with inner product $\mathbb{E}(u^Tv)$, $\forall u, v \in \mathcal{Y}_p$,

— $\mathcal{Y}_{p-1}$ is closed subspace of $\mathcal{Y}_p$,

— $\mathcal{Y}_{p-1}^{\perp}$ is the orthogonal space of $\mathcal{Y}_{p-1}$ in $\mathcal{Y}_p$, says $\mathcal{Y}_p = \mathcal{Y}_{p-1} \oplus \mathcal{Y}_{p-1}^{\perp}$.

Then every $u \in \mathcal{Y}_p$ can be express as $u = P_{\mathcal{Y}_{p-1}}u + P_{\mathcal{Y}_{p-1}^{\perp}}u$, where $P_{\mathcal{X}}$ is the projection operator on the space $\mathcal{X}$, $P_{\mathcal{Y}_{p-1}}u \in \mathcal{Y}_{p-1}$ and $P_{\mathcal{Y}_{p-1}^{\perp}}u \in \mathcal{Y}_{p-1}^{\perp}$.

In addition, $\mathcal{Y}_{p-1}^{\perp}$ can be proved to have the form

$$\begin{aligned}
\mathcal{Z}_{p-1} &= \left\{Vy_p - P_{\mathcal{Y}_{p-1}}Vy_p : V \in \mathbb{R}^{n_x \times n_y}\right\}\\
&= \left\{V(y_p - \mathbb{E}[y_p|Y_{p-1}]) : V \in \mathbb{R}^{n_x \times n_y}\right\}.
\end{aligned}$$

Indeed, for every $V \in \mathbb{R}^{n_x \times n_y}$, $Vy_p \in \mathcal{Y}_p$ and hence $(Vy_p - P_{\mathcal{Y}_{p-1}}Vy_p) \in \mathcal{Y}_{p-1}^{\perp}$ thanks to the direct sum property. So, $\mathcal{Z}_{p-1} \subset \mathcal{Y}_{p-1}^{\perp}$.

Inversely, let $t \in \mathcal{Y}_{p-1}^{\perp}$, there exists $u \in \mathcal{Y}_{p-1}$ so that $u = \sum_{i=1}^{p} T_i y_i$ and $t = u - P_{\mathcal{Y}_{p-1}} u$. Thanks to the projection operator linearity,

$$\begin{aligned} t &= \sum_{i=1}^{p} T_i y_i - P_{\mathcal{Y}_{p-1}} \left(\sum_{i=1}^{p} T_i y_i \right) \\ &= T_p y_p + \sum_{i=1}^{p-1} T_i y_i - P_{\mathcal{Y}_{p-1}} \left(\sum_{i=1}^{p-1} T_i y_i \right) - P_{\mathcal{Y}_{p-1}} T_p y_p. \\ &= T_p y_p - P_{\mathcal{Y}_{p-1}} T_p y_p \ \in \ \mathcal{Z}_{p-1}, \end{aligned}$$

where the last equality holds by using the fact $P_{\mathcal{Y}_{p-1}} \left(\sum_{i=1}^{p-1} T_i y_i \right) = \sum_{i=1}^{p-1} T_i y_i$ since $\sum_{i=1}^{p-1} T_i y_i \in \mathcal{Y}_{p-1}$. Thus $\mathcal{Y}_{p-1}^{\perp} \subset \mathcal{Z}_{p-1}$.
The second form of $\mathcal{Z}_{p-1}$ is verified thanks to the following property:

$$x_k, y_k \text{ are Gaussian} \quad \Rightarrow \quad \begin{cases} \mathbb{E}[x_k|Y_p] &= P_{\mathcal{Y}_p} x_k \ , \\ \mathbb{E}[K_k y_k|Y_p] &= P_{\mathcal{Y}_p} K_k y_k \ , \end{cases}$$

for every $1 \leq p \leq k$, $K_k \in \mathbb{R}^{n_x \times n_y}$ and noting that $\mathbb{E}$ is also linear operator.

Therefore, there exists a $V \in \mathbb{R}^{n_x \times n_y}$ so that

$$\begin{aligned} \mathbb{E}[x_k|Y_k] &= P_{\mathcal{Y}_{k-1}} \mathbb{E}[x_k|Y_k] \ + \ P_{\mathcal{Y}_{k-1}^{\perp}} \mathbb{E}[x_k|Y_k] \\ &= \mathbb{E}[x_k|Y_{k-1}] \ + \ V \left(y_k - \mathbb{E}[y_k|Y_{k-1}] \right) \\ &= \hat{x}_{k|k-1} \ + \ V \left(y_k - \hat{y}_k \right) \ , \end{aligned}$$

noting that $\hat{y}_k = C_k \hat{x}_{k|k-1} + D_k u_k = \mathbb{E}[y_k|Y_{k-1}]$.

$\mathbb{E}[x_k|Y_k] = P_{\mathcal{Y}_k} x_k$ is the optimal approximate of x_k in the sense that

$$\begin{aligned} & \|x_k - \mathbb{E}[x_k|Y_k]\|_{\mathcal{Y}_k}^2 \leq \|x_k - z\|_{\mathcal{Y}_k}^2 \ , \quad \forall z : \sigma(Y_k)\text{-measurable}, \\ \Leftrightarrow \ & \|x_k - \mathbb{E}[x_k|Y_k]\|_{\mathcal{Y}_k}^2 \leq \|x_k - z\|_{\mathcal{Y}_k}^2 \ , \quad \forall z \in \mathcal{Y}_k, \\ \Leftrightarrow \ & \mathbb{E}\left[\|x_k - \mathbb{E}[x_k|Y_k]\|^2 \right] \leq \mathbb{E}\left[\|x_k - z\|^2 \right] \ , \quad \forall z \in \mathcal{Y}_k, \end{aligned} \tag{1.9}$$

This fact is achieved with $V \equiv K_k^*$, the optimal gain presented in (1.5), while the existence and uniqueness of K_k^* is ensured by the existence and uniqueness (almost surely or with probability 1) of $\mathbb{E}[x_k|Y_k]$.

So at time $k \geq 1$, the optimal estimate of x_k chosen as in (1.7) satisfies $\hat{x}_{k|k} \equiv \hat{x}_{k|k}^* = \mathbb{E}[x_k|Y_k]$. Then, $\hat{x}_{k+1|k} = A_{k+1} \hat{x}_{k|k} + B_{k+1} u_{k+1} = \mathbb{E}[x_{k+1}|Y_k]$. By induction, we conclude that, at any time $k \geq 1$,

$$\begin{aligned} \hat{x}_{k|k-1} &\triangleq \mathbb{E}[x_k|Y_{k-1}] \ \equiv \ A_k \hat{x}_{k-1|k-1} + B_k u_k, \\ \hat{x}_{k|k} &\triangleq \hat{x}_{k|k-1} + K_k^* r_k \ \equiv \ \mathbb{E}[x_k|Y_k] \end{aligned} \tag{1.10}$$

□

Algorithm 1 summarizes the SKF mechanism. The estimate $\hat{x}_{k|k}$ therein is actually the optimal estimate $\hat{x}^*_{k|k}$ mentioned above. The same is true for the estimation error covariance. The symbol " * " is just used for the gain matrix to emphasize optimality and is omitted in other terms for simplicity.

Algorithm 1 Standard Kalman Filter

Initialization: $\hat{x}_{0|0}, P_{0|0}, A_k, B_k, C_k, D_k, Q_k, R_k, u_k, y_k,\ k = 1, 2, 3, ..., N$
for $k = 1, 2, 3, ..., N$ **do**
  Prediction step:
    $\hat{x}_{k|k-1} = A_k \hat{x}_{k-1|k-1} + B_k u_k$
    $P_{k|k-1} = A_k P_{k-1|k-1} A_k^T + Q_k$
  Correction step:
    $S_k = C_k P_{k|k-1} C_k^T + R_k$
    $K_k^* = P_{k|k-1} C_k^T S_k^{-1}$
    $\hat{x}_{k|k} = (I - K_k^* C_k)\hat{x}_{k|k-1} + K_k^*(y_k - D_k u_k)$
    $P_{k|k} = (I - K_k^* C_k) P_{k|k-1}$
end for

For further discussion, stochastic properties of related terms appearing in the SKF are presented here. To this end, following notations are used:

$$A^{\otimes}_{k,s} = A_k A_{k-1} ... A_{s+1} A_s \text{ if } s \leq k \quad \text{and} \quad A^{\otimes}_{k,s} = I \text{ if } s > k,$$
$$\tilde{C}_k = I - K_k C_k, \quad \tilde{A}_k = \tilde{C}_k A_k \quad \text{and} \quad \tilde{B}_k = \tilde{C}_k B_k.$$

Properties:

P.1) For $k \geq 1$, the recursive formula of x_k is given by (1.1), whilst a general form of x_k can be written as

$$x_k = A^{\otimes}_{k,1} x_0 + \sum_{i=1}^{k} A^{\otimes}_{k,i+1} B_i u_i + \sum_{i=1}^{k} A^{\otimes}_{k,i+1} w_i \ . \tag{1.11}$$

So, x_k is a function of Gaussian vectors $\{x_0, w_1 : w_k\}$. Furthermore, $x_k \sim \mathcal{N}(\mu_k, P_k)$ where
— $\mu_k = A_k \mu_{k-1} + B_k u_k = A^{\otimes}_{k,1} \mu_0 + \sum_{i=1}^{k} A^{\otimes}_{k,i+1} B_i u_i$,
— $P_k = A_k P_{k-1} A_k^T + Q_k = (A^{\otimes}_{k,1}) P_0 (A^{\otimes}_{k,1})^T + \sum_{i=1}^{k} (A^{\otimes}_{k,i+1}) Q_i (A^{\otimes}_{k,i+1})^T$.

P.2) For y_k, no recursive formula is obtained, but a general form can be derived thanks to (1.11)

$$y_k = C_k A^{\otimes}_{k,1} x_0 + C_k \sum_{i=1}^{k} A^{\otimes}_{k,i+1} B_i u_i + D_k u_k + C_k \sum_{i=1}^{k} A^{\otimes}_{k,i+1} w_i + v_k \tag{1.12}$$

and therefore, y_k is a function of Gaussian vectors $\{x_0, w_{1:k}, v_k\}$.
Furthermore, $y_k \sim \mathcal{N}(\lambda_k, \Gamma_k)$ where
— $\lambda_k = C_k A_{k,1}^{\otimes} \mu_0 + C_k \sum_{i=1}^{k} A_{k,i+1}^{\otimes} B_i u_i + D_k u_k$,
— $\Gamma_k = C_k P_k C_k^T + R_k$
$= (C_k A_{k,1}^{\otimes}) P_0 (C_k A_{k,1}^{\otimes})^T + \sum_{i=1}^{k} (C_k A_{k,i+1}^{\otimes}) Q_i (C_k A_{k,i+1}^{\otimes})^T + R_k.$

P.3) Recursive and general formulas for $\hat{x}_{k|k}$ are deduced from (1.10). $\forall k \geq 1$:

$$\hat{x}_{k|k} = \tilde{A}_k \hat{x}_{k-1|k-1} + (\tilde{B}_k - K_k D_k) u_k + K_k y_k, \tag{1.13}$$

$$\hat{x}_{k|k} = \tilde{A}_{k,1}^{\otimes} \hat{x}_{0|0} + \sum_{i=1}^{k} \tilde{A}_{k,i+1}^{\otimes} (\tilde{B}_i - K_i D_i) u_i + \sum_{i=1}^{k} \tilde{A}_{k,i+1}^{\otimes} K_i y_i \tag{1.14}$$

The SKF is initialized at a chosen starting point $\hat{x}_{0|0}$ (so it is not random). Assuming that μ_0 is known, $\hat{x}_{0|0}$ can be chosen to be μ_0, or says $x_0 \sim \mathcal{N}(\hat{x}_{0|0}, P_0)$.
Furthermore, $\forall k \geq 1$, $\hat{x}_{k|k} \sim \mathcal{N}(\hat{\mu}_k, \hat{P}_k)$ where $\hat{\mu}_k = \mu_k$ and

$$\begin{aligned} \hat{P}_k &= \sum_{i=1}^{k} (\tilde{A}_{k,i+1}^{\otimes} K_i) \Gamma_i (\tilde{A}_{k,i+1}^{\otimes} K_i)^T , \\ &= \sum_{i=1}^{k} (\tilde{A}_{k,i+1}^{\otimes} K_i C_i) P_i (\tilde{A}_{k,i+1}^{\otimes} K_i C_i)^T + \sum_{i=1}^{k} (\tilde{A}_{k,i+1}^{\otimes} K_i) R_i (\tilde{A}_{k,i+1}^{\otimes} K_i)^T . \end{aligned}$$

By (1.14) and (1.12), $\hat{x}_{k|k}$ is a function of $\{y_{1:k}\}$ or of $\{x_0, w_{1:k}, v_k\}$ respectively.

Proof. Let prove $\hat{\mu}_k = \mu_k$ to emphasize this property. Other properties are obtained by direct computation. This property is verified since $\mathbb{E}(\hat{x}_{k|k}) = \mathbb{E}[\mathbb{E}(x_k | y_1 : y_k)] = \mathbb{E}(x_k) = \mu_k$. □

P.4) The estimation error is defined as $\epsilon_k = x_k - \hat{x}_{k|k}$ which can be expressed in the forms

$$\epsilon_k = \tilde{A}_k \epsilon_{k-1} + \tilde{C}_k w_k - K_k v_k , \tag{1.15}$$

$$\epsilon_k = \tilde{A}_{k,1}^{\otimes} \epsilon_0 + \sum_{i=1}^{k} \tilde{A}_{k,i+1}^{\otimes} \tilde{C}_i w_i - \sum_{i=1}^{k} K_i v_i . \tag{1.16}$$

For $k \geq 1$, ϵ_k is a function of $\{x_0, w_{1:k}, v_{1:k}\}$ and $\epsilon_k \sim \mathcal{N}(0, P_{k|k})$ since x_k and $\hat{x}_{k|k}$ are normally distributed with the same mean. The error covariance matrix $P_{k|k} = \mathbb{E}[\epsilon_k \epsilon_k^T]$ determined by (1.4) can also be expressed in recursive and general forms as:

$$P_{k|k} = \tilde{A}_k P_{k-1|k-1} \tilde{A}_k^T + \tilde{C}_k Q_k \tilde{C}_k^T + K_k R_k K_k^T, \quad (1.17)$$

$$P_{k|k} = (\tilde{A}_{k,1}^{\otimes}) P_{0|0} (\tilde{A}_{k,1}^{\otimes})^T + \sum_{i=1}^{k} (\tilde{A}_{k,i+1}^{\otimes} \tilde{C}_i) Q_i (\tilde{A}_{k,i+1}^{\otimes} \tilde{C}_i)^T + \sum_{i=1}^{k} (\tilde{A}_{k,i+1}^{\otimes} K_i) R_i (\tilde{A}_{k,i+1}^{\otimes} K_i)^T. \quad (1.18)$$

Under the assumption $x_0 \sim \mathcal{N}(\hat{x}_{0|0}, P_0)$, the covariance matrix of ϵ_0 is $P_{0|0} = \mathbb{E}[(x_0 - \hat{x}_{0|0})(x_0 - \hat{x}_{0|0})^T] = P_0$. In addition, $\forall k < s$, the errors ϵ_k are independent of w_s and v_s.

P.5) The residual term determined by $r_k = y_k - \hat{y}_k = C_k(x_k - \hat{x}_{k|k-1}) + v_k$ can also be expressed as

$$r_k = C_k A_k \epsilon_{k-1} + C_k w_k + v_k \ . \quad (1.19)$$

For $k \geq 1$, r_k is a function of $\{x_0, w_{1:k}, v_{1:k}\}$ and $r_k \sim \mathcal{N}(0, S_k)$ where

$$S_k = (C_k A_k) P_{k-1|k-1} (C_k A_k)^T + C_k Q_k C_k^T + R_k. \quad (1.20)$$

Furthermore, $\{r_k\}_{k \geq 1}$ is proved to be a sequence of independent innovation terms by its whiteness (null correlation) and Gaussianity properties (Mehra, 1970; Anderson and Moore, 1979).

1.1.2 Bayesian Filtering problem

Given the system (1.1) and assumption **A0**, the state process $\{x_k\}_{k \in \mathbb{N}}$ is Markovian, i.e. $p(x_k|x_{0:k-1}) = p(x_k|x_{k-1})$, and furthermore

$$\begin{aligned} p(y_k|x_{0:k}) &= p(y_k|x_k) \ , \\ p(y_k|x_{0:k}, y_{1:k-1}) &= p(y_k|x_k) \ , \\ p(y_k|x_k) &= \mathcal{N}(.; C_k x_k + D_k u_k, R_k) \ , \\ p(x_k|x_{k-1}) &= \mathcal{N}(.; A_k x_{k-1} + B_k u_k, Q_k) \ , \end{aligned}$$

where $\mathcal{N}(.; \mu, \Sigma)$ is the Gaussian density function with mean μ and covariance Σ. In terms of σ-algebra, above properties can be explained by $\sigma(x_{0:k}, y_{1:k}) = \sigma(x_k)$. This implies also that

$$\begin{aligned} p(y_{1:k}|x_{0:k}) &= p(y_k|x_{0:k}, y_{1:k-1}) p(y_{1:k-1}|x_{0:k}) \\ &= p(y_k|x_k) p(y_{1:k-1}|x_{0:k-1}) \end{aligned}$$

$$= \prod_{i=1}^{k} p(y_i|x_i),$$

or says $\{y_k|x_k\}_{k\in\mathbb{N}^*}$ are mutually independent. It is worth to note that, for the sake of simplicity, an abuse of notation is accepted in this section. That is random terms and their realizations are denoted by the same notations.

Using Bayes's theorem, one gets

$$\begin{aligned} p(x_k|y_{1:k}) &= \frac{p(y_k|x_k, y_{1:k-1})p(x_k|y_{1:k-1})p(y_{1:k-1})}{p(y_{1:k})} \\ &= p(x_k|y_{1:k-1})\frac{p(y_k|x_k)}{p(y_k|y_{1:k-1})} , \end{aligned} \tag{1.21}$$

where

$$\begin{aligned} p(x_k|y_{1:k-1}) &= \int p(x_k|x_{k-1})p(x_{k-1}|y_{1:k-1})dx_{k-1} , \\ p(y_k|y_{1:k-1}) &= \int p(y_k|x_k)p(x_k|y_{1:k-1})dx_k . \end{aligned} \tag{1.22}$$

Under the SKF assumptions, (1.21) and (1.22) are proved to be Gaussian density functions. Therefore, in order to obtain (reconstruct) these densities, one only needs to find the first and second moments related to them, says the means and the covariances. It is clear that $\hat{x}_{k|k}$ and $\hat{x}_{k|k-1}$ in the SKF algorithm (Algorithm 1) are respectively the first moments of $p(x_k|y_{1:k})$ and $p(x_k|y_{1:k-1})$. However, the SKF optimization is managed by the second moment of the error term $\epsilon_k = x_k - \hat{x}_{k|k}$ instead of using that of $p(x_k|y_{1:k})$.

In a more general framework, the Bayesian filtering is announced as follows. Given that $\{x_k\}_{k\in\mathbb{N}}$ is a Markovian process and $\{y_k\}_{k\in\mathbb{N}^*}$ so that $\sigma(x_{0:k}, y_{1:k}) = \sigma(x_k)$, then by using Bayes's theorem, one gets

$$\begin{aligned} p(x_{0:k}|y_{1:k}) &= \frac{p(y_{1:k-1}|x_{0:k}, y_k)p(y_k|x_{0:k})p(x_k|x_{0:k-1})p(x_{0:k-1})}{p(y_{1:k})} \\ &= \frac{p(y_k|x_k)p(x_k|x_{k-1})p(y_{1:k-1}|x_{0:k-1})p(x_{0:k-1})}{p(y_k|y_{1:k-1})p(y_{1:k-1})} \\ &= p(x_{0:k-1}|y_{1:k-1})\frac{p(y_k|x_k)p(x_k|x_{k-1})}{p(y_k|y_{1:k-1})} \\ &= p(x_{0:k-1}|y_{1:k-1})\frac{p(y_k|x_k)p(x_k|x_{k-1})}{\int p(y_k|x_k)p(x_k|x_{k-1})dx_k} \end{aligned} \tag{1.23}$$

By marginalizing (1.23) one recovers (1.21) which can be implemented together with (1.22) as recursive computations in a general Bayesian filter.

However, it is required suitable knowledge (assumptions) about the specificity of $p(y_k|x_k)$, $p(x_k|x_{k-1})$, for $k \geq 1$, and $p(x_0)$, e.g. the Gaussianity. If the processes $\{x_k\}$ and $\{y_k\}$ are given by dynamical equations with additive noises: $x_k = f_k(x_{k-1}, u_k) + w_k$ and $y_k = h_k(x_k, u_k) + v_k$, then (1.22) and (1.23) are rewritten as

$$\begin{aligned} p(x_k|y_{1:k-1}) &= \int p_{w_k}(x_k - f_k(x_{k-1}, u_k))p(x_{k-1}|y_{1:k-1})dx_{k-1} \ , \\ p(y_k|y_{1:k-1}) &= \int p_{v_k}(y_k - g_k(x_k, u_k))p(x_k|y_{1:k-1})dx_k \ , \end{aligned}$$

where $p_{w_k}(.)$ and $p_{v_k}(.)$ are density functions of w_k and v_k respectively.

In real-world data analysis, estimation problem consists in estimating unknown quantities from some given observations. In most of applications, prior knowledge about the phenomenon being modelled is available. This knowledge allows us to formulate Bayesian models, that is prior distributions for the unknown quantities and likelihood functions relating these quantities to the observations. Within this setting, all inference on the unknown quantities is based on the posterior distribution obtained from Bayes's theorem. Often, the observations arrive sequentially in time and one is interested in performing inference on-line. It is therefore necessary to update the posterior distribution as data become available (Doucet et al., 2001).

1.1.3 Particle Filter

The Bayesian filtering is shown to be successful in modeling a large class of applications as presented in the previous section. It provides however analytic solutions only in the case of linear system with additive Gaussian noises (SKF). Other cases require approximation methods, including the extended Kalman filter, *Gaussian sum approximations* and *grid-based filters*. The first two methods do not take into account all the relevant statistical features of the processes under consideration, leading quite often to poor results. Grid-based filters, based on deterministic numerical integration methods, can lead to accurate results, but are difficult to implement and too computationally expensive to be of any practical use in high dimensions (Doucet et al., 2001).

Although the issue of the Bayesian approach is difficult to solve, the approach itself still attracts attention from researchers by its powerful mathematical fundamentals. Other numerical methods solving the Bayesian filtering problem developed since 1960's are named as *Sequential Monte Carlo* (SMC) methods. SMC forms a set of *simulation-based* methods providing an approach to compute the posterior distributions. Unlike grid-based methods, SMC methods are flexible, easy to implement, parallelisable and applicable

in general settings. Numerous closely related algorithms, under the names of *bootstrap filters*, *condensation*, *particle filters*, *interacting particle approximations* and *survival of the fittest*, have appeared in different research fields. As the computer powerful increases and since the key concept of *particle resampling (by bootstrapping)* was first introduced in (Gordon et al., 1993), the SMC methods have became powerful tools for many applications. This is also the reason that one calls interchangeably SMC filters and Particle Filter (PF).

The PF consists in approximating recursively the density $p(x_k|y_{1:k})$ as the cloud of N discrete particles with a probability mass, or weight, assigned to each of them. In other words, a continuous probability density function is approximated by a discrete one. Initially, all particles have equal weights attached to them. To progress to the next time instant, several steps are performed in sequence. First, at the prediction step, the state of every particle is updated according to the dynamic equation. Next, when the new measurements become available, this new information is used to adjust the particle weights. The weight corresponds to the likelihood of each particle state describing the true current state of the system. Finally, the sample states are redistributed to obtain uniform weighting for the following iteration by resampling them from the computed posterior probability distribution. Thus, at any time instant, certain characteristics (position, speed, etc.) can be directly computed, if desired, by using the particle set and weights as an approximation of the true probability density function.

Although being powerful especially when dealing with nonlinear system, the PF is computationally expensive due to the large number of particles being used. When the state dimension increases, the required number of particles also increases. With a high dimensional system, the PF provides only a poor performance. This is however the motivation for later researches, e.g. the branch of investigation related to *Box Particle Filter* (Abdallah et al., 2008), a set-membership (interval) approach to particle filters.

1.2 Fault diagnosis problem

Modern control systems are becoming more and more complex and control algorithms more and more sophisticated. Consequently, the issues of availability, cost efficiency, reliability, operating safety and environmental protection are of major importance. These issues are important to, not only normally accepted safety-critical systems such as nuclear reactors, chemical plants and aircraft, but also other advanced systems employed in cars, rapid transit trains, etc. For safety-critical systems, the consequences of faults can

be extremely serious in terms of human mortality, environmental impact and economic loss. Therefore, there is a growing need for on-line supervision and fault diagnosis to increase the reliability of such safety-critical systems. Early indications concerning which faults are developing can help avoid system breakdown, mission abortion and catastrophes. For systems which are not safety-critical, on-line fault diagnosis techniques can be used to improve plant efficiency, maintainability, availability and reliability (Chen and Patton, 1999).

The terminology used throughout this thesis is the one used in (Chen and Patton, 1999), a rigorous textbook of this research field. According to that, a *fault* is understood as an unexpected change of system function, although it may not represent physical failure or breakdown. Such a fault causes malfunctions or disturbances of the normal operation of an automatic system, which can lead to an unacceptable deterioration of the system performance or even a dangerous situation. The use of the term *failure* is not recommended since it may suggest to think about a catastrophe, a complete breakdown of a system component or function. In contrast, the term fault may be used to indicate that a malfunction may be tolerable at its present stage.

A fault must be diagnosed as early as possible even it is tolerable at its early stage, to prevent any serious consequences. The *fault diagnosis* consists of the following tasks:

- *Fault detection*: to make a decision whether a fault has occurred in the system or not.
- *Fault isolation*: to determine the location of the fault, e.g. which sensor or actuator has become faulty.
- *Fault identification*: to estimate the magnitude and type or nature of the fault.

These above three tasks may be called and classified differently using other terms as *Fault dectection and isolation* (FDI) and *Fault estimation* (FE). Fault diagnosis plays an important role in the fault-tolerant control, as before any control law reconfiguration is possible the fault must be reliably diagnosed and the information should be passed to a supervision mechanism to make proper decision.

There is two main approaches to fault diagnosis: the *hardware redundancy* and the *analytical redundancy* approaches. The former uses multiple sensors, actuators, computers and software to measure and/or control a particular variable. Then, a voting scheme is typically applied to take diagnosis decisions. The major problems encountered with this approach are the equipment and maintenance costs. In contrast, the later uses redundant analytical (or functional) relationships between various measured variables of the monitored process (e.g. inputs/outputs; outputs/outputs; inputs/inputs) to check

the consistency between fault-free (normal) behavior and faulty behavior and take diagnosis decisions. Therefore, this approach is more flexible to be designed, more reliable and powerful at the same cost level.

In analytical redundancy schemes, the resulting difference generated from the consistency checking is called as a *residual* signal. The residual should be zero-valued when the system is fault-free and diverge from zero when the system is faulty. The consistency checking is normally achieved through a comparison between a measured signal with its estimate. The estimate is generated by the mathematical model of the system being considered. In other words, a residual is a fault indicator reflecting the faulty situation of the monitored system.

A fault diagnosis scheme has in general two stages: the *residual generation* and the *residual evaluation or decision making* stages. In the first stage, the residuals are produced by a *residual generator* which can apply a deterministic (*norm-bounded*) or stochastic (*innovation-based*) approaches. There are also different methods for a residual generator, for instance (Ding, 2013):

— the *parity space* method,
— the *observer-based* method,
— the *parameter identification based* methods.

Another alternative approach in the literature is the use of the classical integrator disturbance models in an augmented state-space representation for sensor/actuator fault estimation. In the second stage, the residual or a function (transformation) of it is compared to a *threshold* predetermined *constantly* or determined *adaptively*.

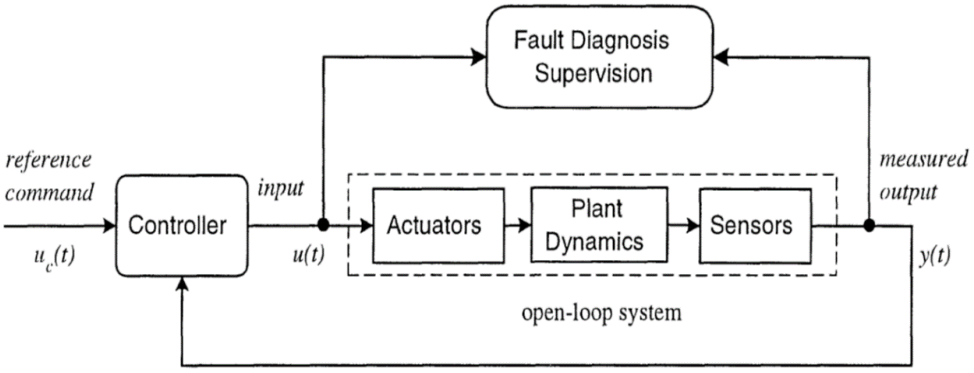

Figure 1.1 – Fault diagnosis and control loop

In the control view point, the system model required in model-based FDI is an open-loop system model although a closed-loop system is in consideration (Fig.1.1). Thus, it is not necessary to consider the controller in the

design of a fault diagnosis scheme. This is consistent with the separation principle in control theory because fault diagnosis can be broadly treated as an observation problem. Once the input to the actuators is available, the fault diagnosis problem is the same no matter how the system is working in open-loop or in the closed-loop (Chen and Patton, 1999).

The above statement is however nowadays limited to the *passive fault diagnosis* in comparison with the *active fault diagnosis*. The later consists in using designed *auxiliary signals* as supplement inputs injected to the monitored system to excite the fault if it exists. Thus, it concerns the system stability and hence the design of appropriate controller. This is a novel branch of research and is presented in more details in Chapter 5.

Moreover, in the field of *Fault-Tolerant Control* (FTC), *control reconfiguration* is an interesting method that uses the results of a fault diagnosis component to restructure the control loop and to adapt the controller to the faulty plant. This control aspect is not investigated in the thesis, however a detailed tutorial can be found in (Lunze and Richter, 2008).

Chapter 2

Optimal Upper Bound Interval Kalman Filter

2.1 Introduction

In both industry and academia, Kalman Filter introduced in (Kalman, 1960) has always been interested by its elegant form and result characteristics (optimal estimator, on-line implementation,...). This is a kind of *stochastic approach* for estimation and referred as *Standard Kalman Filter* (SKF). Since then, many extensions of the SKF have been presented to improve its applicability and performance when dealing additionally with bounded uncertainties, of which the two major derivations are *robust* and *interval Kalman filtering*. In the discussion below, the following extensions of both derivations deal with bounded uncertainty in parameter matrices only and do not concern bounded nonlinearities of the state equation derived from quasi-linear system models.

The robust Kalman filtering, (Xie et al., 1994; Sayed, 2001; Zhe and Zheng, 2006; Mohamed and Nahavandi, 2012a), provides essentially *point estimators* (of the real states) attempting to limit the disturbance effects to the filter performance. For instance, in (Zhe and Zheng, 2006) and (Mohamed and Nahavandi, 2012a), finite-horizon robust Kalman filters for discrete time-varying uncertain systems with additive uncertain covariance white noises are studied without and with missing measurements respectively. Both papers concern an minimization of the trace of a chosen upper bound of all admissible error estimation covariances with respect to (w.r.t.) some design scalar parameters selected (or tuned) adequately, says a *point-wise optimization approach*.

The interval Kalman filtering provides essentially intervals containing all admissible estimators (of the real states) consistent with considered uncertainties and usually being used as *interval estimators* for bounds of the real states. It may have a relation with the robust approach when using an element (usually the center) of the yielded interval as a robust estimator in some sense to be precised, however this is not the initial objective of the *set-membership (interval) approach*. The *Interval Kalman Filter* (IKF) was first introduced in (Chen et al., 1997) with an optimal solution and a suboptimal scheme for the purpose of real-time implementation. Then, authors have tried to further investigate this interesting research by its simplicity (although with *conservatism*) in computation thanks to interval computations (Section 2.2.1) and the similar structure of the SKF with two steps (prediction and correction) in which the later would improve the estimator obtained from the former via the stake of a gain matrix (Xiong et al., 2013; Tran et al., 2017; Lu et al., 2019; Tran et al., 2021).

Xiong et al. (2013) and Tran et al. (2017) study enhancing methods for IKF and Lu et al. (2019) proposes an optimal solution for the conservatism problem due to the choice of the IKF bounds. In (Xiong et al., 2013), the proposed method consists in adding some positivity constraints together with the *SIVIA algorithm* to obtain the interval matrix $[K_k]$ containing all potential optimal gains and hence yielding guaranteed estimation results (without missing some admissible estimates as in the suboptimal case proposed by (Chen et al., 1997)). In (Tran et al., 2017), the interval matrix $[K_k]$ of (Xiong et al., 2013) is replaced by a point matrix K_k minimizing the trace of an upper bound of the estimation error covariances, thanks to which the computation time is reduced and the resulted estimators are less conservative. In (Lu et al., 2019), an optimal upper bound of all symmetric positive semidefinite matrices belonging to a given interval is provided under the form $\alpha^* I$ with $\alpha^* \in \mathbb{R}_+$, thanks to which upper bound expressions are simplified and suitable for advanced optimizations and the computation time is further reduced. Then, considering a large class of upper bounds characterized by two real parameters and including the one used in (Tran et al., 2017), Lu et al. (2019) also proposes a point-wise optimization for each choice of these scalar parameters. More recently, (Tran et al., 2021) proposes an enhanced method of (Tran et al., 2017) with the same principle of the later, leading to less conservative interval estimates and requiring however larger computation time with respect to the later.

The present work is a development of (Lu et al., 2019). The first motivation drives our researches is to find an uniform optimized solution of the error estimation covariance upper bounds in terms of their characterized scalar parameters. Furthermore, in the interval approach, a major issue is

the conservatism of the resulted estimators due to the one of interval computations accumulated in algorithm iterations. In the worst case, the width of the resulted estimators may explode with a very high value. No study in the above papers addresses the conditions under which the provided algorithms can be controlled to perform with stability, i.e. without explosion in width of the resulted estimators. This is another motivation for our work.

The chapter presents analytical developments concerning the optimization of a concrete class of upper bounds of the one introduced in (Lu et al., 2019). Each upper bound is seen as a function of two arguments: a gain matrix and a (strict) positive parameter. This class also includes the upper bound used in (Tran et al., 2017). The optimization is presented with more concrete and consistency thanks to a system of proposed notations. The optimization is performed in two stages: firstly in terms of the gain matrix and secondly in terms of the depending scalar parameter. A connection with the well-known optimization result of SKF is pointed out in Theorem 6 of this chapter which is proved in a novel view and notations. Then, conditions under which the second stage optimization in terms of the scalar parameter can be performed are provided. Under these conditions, the optimal trace value is controlled and hence the algorithm in consideration is ensured to

— perform with *C-stability* to be clarified in Definition 6,
— obtain a smaller trace upper bound of the covariance matrices in the correction step than the one in the prediction step.

Thereby, the algorithm proposed in (Lu et al., 2019), namely *Optimal Upper Bound Interval Kalman Filter* (OUBIKF), is enhanced both theoretically and practically by the developments presented in this chapter.

2.2 Theoretical and mathematical background and tools

2.2.1 Essential of matrix and interval matrix

A real $m \times n$ matrix is denoted by $A = (a_{ij})$, $a_{ij} \in \mathbb{R}$, $i \in \{1, ..., m\}$, $j \in \{1, ..., n\}$. The set of real $m \times n$ matrices is denoted by $\mathbb{R}^{m \times n}$. A^T is the transpose matrix of A.

Let $A = (a_{ij}) \in \mathbb{R}^{n \times n}$ be a square matrix of order n, the notations $\sigma_i(A)$, $\lambda_i(A)$, $i \in \{1, ..., n\}$, are used to indicate respectively *singular values* and *eigenvalues* of A among which $\sigma_{\max}(A)$ and $\lambda_{\max}(A)$ are the corresponding maximum values. By definition, $\sigma_i(A) \triangleq \sqrt{\lambda_i(A^T A)}$, $i \in \{1, ..., n\}$. The *trace* of matrix A is defined by $\mathrm{Tr}(A) \triangleq \sum_{i=1}^{n} a_{ii}$. The *identity matrix* of

order n is denoted by I_n and its i-th column, denoted by e_i, is called the i-th *standard unit vector*. The notation $\mathbf{1}$ (or $\mathbf{1}_n$) denotes the (n-)vector whose components all equal to 1 and is called the *all one vector*. The *indicator function* $\mathbb{I}(x)$ is defined to be 1 if the condition x holds true and vanishes otherwise, in which x can be a vector of conditions.

Diagonal operators are defined as follows. Let x be a vector $(x_1, ..., x_n)^T$ or an n-tuple $(x_1, ..., x_n)$, define

$$\text{diag}(x) \quad \equiv \quad \text{diag}\{x_1, ..., x_n\} \quad \triangleq \quad (x_i \delta_{ij})_{i,j \in \{1,...,n\}}$$

being the diagonal matrix whose entries are of the form $x_i \delta_{ij}$ for $i, j \in \{1, ..., n\}$ and δ_{ij} is the Kronecker delta. Define also for any square matrix $A = (a_{ij})$ of order n:

$$\text{Diag}(A) \triangleq (a_{ii}\delta_{ij})_{i,j \in \{1,...,n\}} \qquad \text{and} \qquad \text{Diag}_v(A) \triangleq (a_{11}, ..., a_{nn})^T,$$

where $\text{Diag}(A)$ is the diagonal matrix having the same diagonal as the matrix A and $\text{Diag}_v(A)$ is the vector of diagonal entries of the matrix A. Thus,

$$\text{Diag}(A) \; = \; \text{diag}\{a_{11}, \ldots, a_{nn}\} \; = \; \text{diag}\{\text{Diag}_v(A)\}.$$

Definition 1 (**Positive semidefinite (definite) matrix**). A real square matrix A of order n is positive semidefinite (definite resp.), denoted by $A \succeq 0$ ($A \succ 0$ resp.), if and only if A satisfies $z^T A z \geq 0$, $\forall z \in \mathbb{R}^n$ ($z^T A z > 0$, $\forall z \in \mathbb{R}^n \setminus \{\mathbf{0}\}$ resp.).

Example 1 (**A positive semidefinite matrix is not necessarily symmetric**). Consider matrix $A = (a_{ij}) \in \mathbb{R}^{2\times 2}$ such that $a_{ii} = a > 0$, $i \in \{1, 2\}$. For any $z \in \mathbb{R}^2$, $z^T A z = a[z_1^2 + z_2^2 + \frac{a_{12}+a_{21}}{a} z_1 z_2]$. So if $a_{12} + a_{21} = 2a$ then A is actually positive semidefinite but not necessarily symmetric.

Denote:

a) $A \not\succeq 0$ ($\not\succ 0$ resp.), the matrix A being not positive semidefinite (definite resp.),

b) $S(n) \triangleq \left\{M \in \mathbb{R}^{n\times n} : M = M^T\right\}$, the set of real symmetric matrices of order n,

c) $S_+(n) \triangleq \{M \in S(n) : M \succeq 0\}$, the set of real symmetric positive semidefinite matrices of order n.

Remark 4. $S(n)$ is a vector subspace of $\mathbb{R}^{n\times n}$ and $S_+(n)$ is a convex cone, that is:

— $\alpha M + \beta N \in S(n),\ \forall M, N \in S(n),\ \forall \alpha, \beta \in \mathbb{R}$,
— $\alpha M + \beta N \in S_+(n),\ \forall M, N \in S_+(n),\ \forall \alpha, \beta \in \mathbb{R}^+$. □

Definition 2 (Matrix norm). Let $A = (a_{ij}) \in \mathbb{R}^{n\times n}$ and $x = (x_1, ..., x_n)^T$. Vector norm and matrix norms are defined as follow (Zhan, 2002):

a) The Euclidian vector norm: $\|x\|_2 \triangleq \sqrt{\sum_{i=1}^n x_i^2}$,

b) The nuclear norm: $\|A\|_* \triangleq \sum_{i=1}^n \sigma_i(A) = \sum_{i=1}^n \sqrt{\lambda_i(A^TA)}$,

c) The operator norm: $\|A\| \triangleq \sigma_{\max}(A) = \sqrt{\lambda_{\max}(A^TA)}$,

d) The Frobenius norm:
$\|A\|_F \triangleq \sqrt{\sum_{i=1}^n \sigma_i^2(A)} = \sqrt{\sum_{i=1}^n \lambda_i(A^TA)} = \sqrt{\mathrm{Tr}(A^TA)} = \sqrt{\sum_{i,j=1}^n |a_{ij}|^2}$.

Remark 5.

a) $\|A\| \le \|A\|_F \le \sqrt{n}\|A\|$ and $\|A\|_F \le \|A\|_* \le \sqrt{n}\|A\|_F$.

b) If $A \in S(n)$ then $\sigma_i(A) = |\lambda_i(A)|,\ \forall i = 1, ..., n$.
In particular, $\sigma_{\max}(A) = \max\{\lambda_{\max}(A), |\lambda_{\min}(A)|\}$.

c) If $A \in S_+(n)$ then $\sigma_i(A) = \lambda_i(A),\ \forall i = 1, ...n$, consequently
— $\|A\|_* = \sum_i \lambda_i(A) = \mathrm{Tr}(A)$,
— $\|A\| = \lambda_{\max}(A)$,
— $\|A\|_F = \sqrt{\sum_{i=1}^n \lambda_i^2(A)} = \sqrt{\mathrm{Tr}(A^2)} = \sqrt{\sum_{i,j=1}^n |a_{ij}|^2}$. □

Theorem 1. *Let $A \in \mathbb{R}^{n\times n}$ and λ be an eigenvalue of A. Then:*

i) $c\lambda$ is an eigenvalue of cA for any $c \in \mathbb{R}$.

ii) λ^k is an eigenvalue of A^k for any $k \in \mathbb{Z} \setminus \{0\}$.

iii) $p(\lambda)$ is an eigenvalue of $p(A)$ for any polynomial $p(.)$.

iv) $\lambda + \alpha$ is an eigenvalue of $A + \alpha I$ for any $\alpha \in \mathbb{R}$.

Proof. The first three statements are Theorems ESMM, EOMP and EPM in (Beezer, 2015) at pages 392-393 with detailed proofs therein. The second statement is proved by induction and the third one is proved using the first two others. The last statement is direct consequence of the third one.

It is noted that the third statement of the theorem essentially goes back to Cayley-Halminton theorem stated for 3×3 and smaller matrices in (Cayley, 1858) and the general case was first proved in (Frobenius, 1878). □

Theorem 2. *Let $A \in S(n)$. Following statements are verified:*

i) All eigenvalues of A are real: $\lambda_i(A) \in \mathbb{R},\ \forall i = 1, ..., n$.

ii) There exists an orthogonal matrix Q and diagonal matrix $D = diag\{\lambda_i(A)\}$, $i \in \{1, \ldots, n\}$, so that $A = QDQ^T$. It is called the spectral decomposition of matrix A.

The matrix Q has the following properties:
$Q \in \mathbb{R}^{n\times n}$, $Q^T = Q^{-1}$, $\|Q\| = 1$, $\langle Qx, Qy\rangle = \langle x, y\rangle$ for any $x, y \in \mathbb{R}^n$, columns q_i of Q are eigenvectors of A associated to $\lambda_i(A)$ and $\langle q_i, q_j\rangle = \delta_{ij}$ the Kronecker delta and $\langle ., .\rangle$ the inner product.

The matrix Q^T is also orthogonal and has similar properties of Q.

iii) The following inequalities hold:

$$\forall u \in \mathbb{R}^n \ : \quad \lambda_{\min}(A)\|u\|^2 \leq u^T Au \leq \lambda_{\max}(A)\|u\|^2. \tag{2.1}$$

Proof. The first statement is a direct consequence of Theorem HRME in (Beezer, 2015), page 400. The existence of matrices Q and D is confirmed by Theorem OD in (Beezer, 2015) at page 575 while properties of Q are those of a unitary matrix stated by Theorems at pages 212-213. The operator norm of Q is computed directly using the corresponding norm definition. The last statement is proved using the spectral decomposition of A and properties of Q^T as follow:

$$\begin{aligned} \forall u \in \mathbb{R}^n,\ \text{let } v = Q^T u \quad &\Rightarrow \quad u^T Au = (Q^T u)^T D(Q^T u) = \sum_{i=1}^{n} \lambda_i(A) v_i^2 \\ &\Rightarrow \quad \lambda_{\min}(A)\|v\|^2 \leq u^T Au \leq \lambda_{\max}(A)\|v\|^2, \end{aligned}$$

and noting that $\|v\|^2 = \langle v, v\rangle = \langle Q^T u, Q^T u\rangle = \langle u, u\rangle = \|u\|^2$. □

Theorem 3. *$A \in S_+(n)$ if and only if $A \in S(n)$ and $\lambda_{\min}(A) \geq 0$.*

Proof. Being symmetric, A can be decomposed as QDQ^T using Theorem 2.

($\Rightarrow$) For any $u \in \mathbb{R}^n$, $u^T Au \geq 0$. Choose $u = v_{\min}$, the eigenvector associated with $\lambda_{\min}(A)$, then $Av_{\min} = \lambda_{\min} v_{\min}$ and hence $\lambda_{\min}(A)\|v_{\min}\|^2 = v_{\min}^T Av_{\min} \geq 0$ which implies $\lambda_{\min}(A) \geq 0$.

($\Leftarrow$) By (2.1), $u^T Au \geq \lambda_{\min}(A)\|u\|^2 \geq 0, \forall u \in \mathbb{R}^n$ implying $A \in S_+(n)$. □

Lemma 2. *Let $A \in S(n)$ and $\alpha \in \mathbb{R}$. Then*

$$A + \alpha I \in S_+(n) \quad \Leftrightarrow \quad \alpha \geq -\lambda_{\min}(A).$$

Proof. Since A and αI are symmetric, $A+\alpha I$ is also symmetric. By Theorem 2, all eigenvalues of these matrices are real. Furthermore, by Theorem 1, $\lambda_i(A+\alpha\, I) = \lambda_i(A) + \alpha$, $\forall i = 1, ..., n$. Hence $\lambda_{\min}(A+\alpha I) = \lambda_{\min}(A) + \alpha$.

Then the lemma conclusion is straightforward using Theorem 3. □

Lemma 3. *The following statements hold true:*

a) $A^TA \succeq 0$ *and* $AA^T \succeq 0$ *for any* $A \in \mathbb{R}^{m\times n}$.

b) If $P \succeq 0$ *then* $MPM^T \succeq 0$ *and* $N^TPN \succeq 0$ *for all* M, N *with appropriate dimensions.*

Proof. For any $u \in \mathbb{R}^n$ and $v \in \mathbb{R}^m$, $u^TA^TAu = \|Au\|^2 \geq 0$ and $v^TAA^Tv = \|A^Tv\|^2 \geq 0$. This concludes the first statement of the lemma.

Let $P \in \mathbb{R}^{n\times n}$, $M \in \mathbb{R}^{m\times n}$. By definition, $z^TPz \geq 0$, $\forall z \in \mathbb{R}^n$. So, for any $u \in \mathbb{R}^m$, put $z = M^Tu$, then $z \in \mathbb{R}^n$ and hence

$$u^TMPM^Tu \;=\; (M^Tu)^TP(M^Tu) \;=\; z^TPz \;\geq\; 0.$$

It follows that $MPM^T \succeq 0$. $N^TPN \succeq 0$ is verified by putting $M = N^T$. □

Definition 3 (**Moore-Penrose pseudoinverse**)**.** Let $A \in \mathbb{R}^{m\times n}$. A matrix $A^+ \in \mathbb{R}^{n\times m}$ is said to be a Moore-Penrose pseudoinverse of A if it satisfied following conditions:

a) $AA^+A = A$,

b) $A^+AA^+ = A^+$,

c) $AA^+ = (AA^+)^T$ and $A^+A = (A^+A)^T$.

Proposition 1 (**Some specific Moore-Penrose pseudoinverse**)**.**

a) If $z \in \mathbb{C}$ *then* $z^+ = z^{-1}\mathbb{I}(z \neq 0)$.

b) If $D = diag\{d_1, ..., d_n\} \in \mathbb{C}^{n\times n}$ *then* $D^+ = diag\{d_1^+, ..., d_n^+\}$.

c) If $A \in S(n)$ *then it has a spectral decomposition* QDQ^T *and* $A^+ = QD^+Q^T$.

d) In general, for any $A \in \mathbb{R}^{m\times n}$, $A^+ = A^T(AA^T)^+ = \lim_{\beta\to 0} A^T(AA^T + \beta I)^{-1}$.

Proof. It is referred to (Barata and Hussein, 2012) for a tutorial of Moore-Penrose pseudoinverse. Proposition 1 gathers useful results in the reference. The proofs of these properties are based notably on Definition 3 and can be found in the reference. □

Interval analysis. A real interval, denoted by $[x]$, is a closed connected subset of $\mathbb{R}$. A real interval matrix $[X]$ of dimension $p \times q$ is a matrix with real interval components $[x_{ij}]$, $i \in \{1, ..., p\}$, $j \in \{1, ..., q\}$. Write $X \in [X]$ to indicate a point matrix $X = (x_{ij})$ belonging element-wise to $[X]$. Other element-wise operators used in the next are " $\inf, \sup, \leq (\geq)$ ". Define for all $i \in \{1, ..., p\}$, $j \in \{1, ..., q\}$:

- $\sup([X]) \triangleq (\sup([x_{ij}]))$,
- $\inf([X]) \triangleq (\inf([x_{ij}]))$,
- $\mathrm{mid}([X]) \triangleq (\sup([X]) + \inf([X]))/2 = (\mathrm{mid}([x_{ij}]))$,
- $\mathrm{rad}([X]) \triangleq (\sup([X]) - \inf([X]))/2 = (\mathrm{rad}([x_{ij}]))$,
- $\mathrm{width}([X]) \triangleq \sup([X]) - \inf([X]) = (\mathrm{width}([x_{ij}]))$.

The matrices $\mathrm{mid}([X])$, $\mathrm{rad}([X])$ and $\mathrm{width}([X])$ are called respectively the *midpoint matrix*, the *radius matrix* and the *width matrix* of $[X]$. Denote also

$$\overline{X} = \sup([X]) \ , \ \underline{X} = \inf([X]) \ , \ [X] = [\underline{X}, \overline{X}] = \mathrm{mid}([X]) \pm \mathrm{rad}([X]).$$

The matrices $\overline{X}$ and $\underline{X}$ will be called respectively the *largest* and *smallest matrix* of $[X]$ to distinguish with the notions of upper/lower bound matrices defined in the next section. From this, $[X]$ can be seen as a subset of $\mathbb{R}^{p \times q}$ determined by

$$[X] = \{X \in \mathbb{R}^{p \times q} : \underline{X} \leq X \leq \overline{X}\},$$

and therefore set operators $(\subset, \cap, \cup, \setminus, ...)$ can be applied as usual. Define the hull of a closed set $S \subset \mathbb{R}^{p \times q}$ and the hull of two interval matrices $[X_1]$, $[X_2]$ of the same dimension as follows

$$\begin{aligned} \mathrm{hull}\{S\} &\triangleq [\inf(S), \sup(S)] \,, \\ \mathrm{hull}\{[X_1], [X_2]\} &\triangleq \mathrm{hull}\{[X_1] \cup [X_2]\} = [\inf\{\underline{X}_1, \underline{X}_2\}, \sup\{\overline{X}_1, \overline{X}_2\}], \end{aligned}$$

and $\mathrm{hull}\{\emptyset\} = \emptyset$ by convention.

Basic interval computation. Let $[u] = [\underline{u}, \overline{u}]$ and $[v] = [\underline{v}, \overline{v}]$ be two real

intervals and $\alpha \in \mathbb{R}$. Define

- $[u] + [v] = [\underline{u} + \underline{v}\ ,\ \overline{u} + \overline{v}]$,
- $\alpha \times [u] = \begin{cases} [\alpha\underline{u}\ ,\ \alpha\overline{u}] & , \quad \alpha \geq 0 \\ [\alpha\overline{u}\ ,\ \alpha\underline{u}] & , \quad \alpha < 0 \end{cases}$
- $[u] - [v] = [\underline{u} - \overline{v}\ ,\ \overline{u} - \underline{v}]$,
- $[u] \times [v] = \text{hull}\{\underline{uv}, \underline{u}\overline{v}, \overline{u}\underline{v}, \overline{uv}\}$,
- $[u]^{-1} = \begin{cases} \emptyset & , \quad [u] \equiv 0\ , \\ [\overline{u}^{-1}, \underline{u}^{-1}] & , \quad [u] \not\ni 0\ , \\ [\overline{u}^{-1}\ ,\ \infty] & , \quad 0 = \underline{u} < \overline{u}, \\ [-\infty,\ \underline{u}^{-1}] & , \quad \underline{u} < \overline{u} = 0, \\ [-\infty\ ,\ \infty] & , \quad \underline{u} < 0 < \overline{u}. \end{cases}$
- $[u]/[v] = [u] \times [v]^{-1}$.

Since then, interval matrix computations are defined similarly to matrix computations using the basic operations above:

- $[M] \pm [N] = [P] = ([p_{ij}])$ such that $[p_{ij}] = [m_{ij}] \pm [n_{ij}]$,
- $\alpha \times [M] = [P] = ([p_{ij}])$ such that $[p_{ij}] = \alpha \times [m_{ij}]$,
- $[M] \times [N] = [P] = ([p_{ij}])$ such that $[p_{ij}] = \sum_k [m_{ik}] \times [n_{kj}]$,

for any $[M] = ([m_{ij}])$, $[N] = ([n_{ij}])$ of appropriate dimensions and $\alpha \in \mathbb{R}$. More general operators are constructed by means of *inclusion function* $[f]([x])$ (Jaulin et al., 2001). In practice, the package Intlab (Rump, 1999) developed for Matlab (also existing in Octave and C/C^{++}) is used for these computations.

A major issue of interval computation is the result *conservatism* after each operation (calculation). That is the resulted interval is always the superset of the one of all possible results yielded by the operator in consideration. Then, after a number of operations consecutive, the conservatism may be large.

2.2.2 Bounds of a non empty set of real square matrices

In this section, the notion of bounds (with respect to a partial order) of a non empty set of real square matrices is introduced.

Definition 4 (Partial order of real square matrices). Let M, N be two real square matrices of the same size. An order between M and N denoted by $N \preceq M$ is defined if $M - N \succeq 0$. M is called an *upper bound* of N and N a *lower bound* of M.

In the case of Hermitian matrices, this order is known as the *Loewner (partial) order* (ref. (Pukelsheim, 2006; Zhan, 2002)). Recall that a partial

order $\mathcal{R}$ satisfies the properties: *i)* $a\mathcal{R}a$ (*Reflexivity*); *ii)* If $a\mathcal{R}b$ and $b\mathcal{R}a$ then $a = b$ (*Anti-symmetry*); *iii)* If $a\mathcal{R}b$ and $b\mathcal{R}c$ then $a\mathcal{R}c$ (*Transitivity*).

The partial order in Definition 4 is extended to the notion of bounds for a non empty set Ω of real squared matrices as follows:

— U is an *upper bound* of Ω, denoted $\Omega \preceq U$, if $M \preceq U$, $\forall M \in \Omega$.

— L is a *lower bound* of Ω, denoted $L \preceq \Omega$, if $L \preceq M$, $M \in \Omega$.

— If P and Q are two upper (lower) bounds of Ω, then P is said *better* than Q if and only if the norm of P is smaller (greater) than or equal to the norm of Q depending on the choice of norms in Definition 2.

Definition 5. Let Ω be a non empty subset of $\mathbb{R}^{n\times n}$ and φ a function: $E \subset \mathbb{R}^{p\times q} \to \mathbb{R}^{n\times n}$ $(p, q, n \geq 1)$. Define:

a) $\inf \Omega$ is defined to be a matrix $L \in \mathbb{R}^{n\times n}$ s.t. following conditions hold:
- $L \preceq M$, $\forall M \in \Omega$,
- If $\tilde{L} \preceq M$, $\forall M \in \Omega$ then $\tilde{L} \preceq L$.

b) $\sup \Omega$ is defined to be a matrix $U \in \mathbb{R}^{n\times n}$ s.t. following conditions hold:
- $M \preceq U$, $\forall M \in \Omega$,
- If $M \preceq \tilde{U}$, $\forall M \in \Omega$ then $U \preceq \tilde{U}$.

c) $\inf_{\beta\in E}\{\varphi(\beta)\}$ is defined to be a matrix $L \in \mathbb{R}^{n\times n}$ s.t. following conditions hold:
- $L \preceq \varphi(\beta)$, $\forall \beta \in E$,
- If $\tilde{L} \preceq \varphi(\beta)$, $\forall \beta \in E$ then $\tilde{L} \preceq L$.

d) $\sup_{\beta\in E}\{\varphi(\beta)\}$ is defined to be a matrix $U \in \mathbb{R}^{n\times n}$ s.t. following conditions hold:
- $\varphi(\beta) \preceq U$, $\forall \beta \in E$,
- If $\varphi(\beta) \preceq \tilde{U}$, $\forall \beta \in E$ then $U \preceq \tilde{U}$.

It is worth to note that $\inf \Omega$ and $\sup \Omega$ are not necessarily included in Ω and in the last two definitions above, Ω can be considered as $\Omega = \{\varphi(\beta) \in \mathbb{R}^{n\times n}, \varphi \in E\}$.

Denote further that:

a) $S([X]) \triangleq \big\{X \in [X] : X = X^T\big\}$, the set of symmetric matrices belonging to $[X]$.

b) $S_+([X]) \triangleq \{X \in S([X]) : X \succeq 0\}$, the set of symmetric positive semidefinite matrices belonging to $[X]$.

c) $BS([X]) \triangleq \{U \in S(n) : S([X]) \preceq U\}$, the set of symmetric upper bounds of $S([X])$.

d) $BS_+([X]) \triangleq \{U \in S_+(n) : S_+([X]) \preceq U\}$, the set of symmetric positive semidefinite upper bounds of $S_+([X])$.

2.2.3 Optimal upper bound of the set of symmetric positive semidefinite matrices belonging to an interval matrix

In this section, we investigate the optimal upper bound of the set Ω of symmetric positive semidefinite matrices belonging to an interval matrix.

Proposition 2. *Let $A \in S(n)$. Then $A \preceq \alpha I$ if and only if $\alpha \geq \lambda_{\max}(A)$.*

Proof. The proposition is proved by using Lemma 2, that is

$$A \preceq \alpha I \;\Leftrightarrow\; (-A) + \alpha I \succeq 0 \;\Leftrightarrow\; \alpha \geq -\lambda_{\min}(-A),$$

noting that $\lambda_{\min}(-A) = \min_i\{\lambda_i(-A)\} = \min_i\{-\lambda_i(A)\} = -\lambda_{\max}(A)$. □

Proposition 3. *The following statements hold:*

a) If $A, B \in S(n)$ and $A \preceq B$ then : $\lambda_{\max}(A) \leq \lambda_{\max}(B)$ and $\mathrm{Tr}(A) \leq \mathrm{Tr}(B)$.

b) If $A, B \in S_+(n)$ and $A \preceq B$ then : $\|A\| \leq \|B\|$ and $\|A\|_* \leq \|B\|_*$.

Proof. By Proposition 2 and the transitivitive of the $\preceq$ order, $A \preceq B \preceq \lambda_{\max}(B)I$ and $\lambda_{\max}(A) \leq \lambda_{\max}(B)$. So the first inequality of 3a) is hold.

Since $u^T(B - A)u \geq 0, \forall u \in \mathbb{R}^n$ and by choosing consecutively u as i-th standard unit vectors e_i then diagonal entries of A and B are such that $b_{ii} \geq a_{ii}$, $i \in \{1, ..., n\}$, and hence the second inequality of 3a) is induced.

The part 3b) is obtained by using 3a) together with remark 5c). □

In following propositions and corollary of this section, let $[M] = ([m_{ij}])$ be a real interval symmetric matrix of order n, that is $[m_{ij}] = [m_{ji}]$, $i, j \in \{1, ..., n\}$ or $[M]^T = [M]$, and assume that $S_+([M])$ is non empty.

Proposition 4. *The following properties are verified:*

a) $S([M])$ is compact in the norm vector space $S(n)$.

b) $S_+([M])$ is a compact subset of $S([M])$.

c) $\Gamma \triangleq \{\gamma = \|M\| : M \in S([M])\}$ and $\Gamma_+ \triangleq \{\gamma = \|M\| : M \in S_+([M])\}$ are compact in $\mathbb{R}$.

d) Let $\alpha_+^* \triangleq \sup_{M \in S_+([M])} \{\lambda_{\max}(M)\}$ and $\alpha^* \triangleq \sup_{M \in S([M])} \{\lambda_{\max}(M)\}$. Then

$$\alpha_+^* \;\leq\; \alpha^* \;<\; \infty.$$

Proof. a) The upper triangular part of matrix $[M]$ has $m = (n^2 + n)/2$ interval elements $I_1, ..., I_m$. We can construct a continuous function f from

$I_1 \times ... \times I_m$ in $\mathbb{R}^{n\times n}$. Then, since $I_1 \times ... \times I_m$ is compact in $\mathbb{R}^m$, the image $f(I_1 \times ... \times I_m) = S([M])$ is also compact in $S(n)$. The construction of f is given by $f = \psi \circ \phi$ where ϕ and ψ are two continuous functions determined as follow

$$x = (x_1, ..., x_m) \mapsto \phi(x) = N = \begin{bmatrix} x_1 & x_2 & \cdots & x_n \\ 0 & x_{n+1} & \cdots & x_{2n-1} \\ \vdots & \cdots & \ddots & \cdots \\ 0 & 0 & \cdots & x_m \end{bmatrix}$$

and $N \mapsto \psi(N) = N + N^T - \text{Diag}(N)$.

b) It is only necessary to prove that $S_+([M])$ is closed in $S(n)$, and the result is concluded by the property: *If K is compact in a topological space X and if F is closed in X with $F \subseteq K$, then F is compact.*
Assume that $\{M_k\}_k$ is a sequence in $S_+([M])$ converging to $M_\infty \in S(n)$ and prove that $M_\infty \in S_+([M])$, i.e. $M_\infty \in [M]$ and $M_\infty \succeq 0$. Denote their corresponding entries as $m_{k,ij}$ and $m_{\infty,ij}$. By assumption,

$$\|M_k - M_\infty\|_F^2 = \sum_{i,j} (m_{k,ij} - m_{\infty,ij})^2 \xrightarrow{k\to\infty} 0$$

hence $(m_{k,ij} - m_{\infty,ij})^2 \xrightarrow{k\to\infty} 0$, $\forall i, j = 1, ..., n$.
Since each $m_{k,ij}$ belongs to a closed interval $[m_{ij}]$ of matrix $[M]$ then $m_{\infty,ij} \in [m_{ij}]$. So $M_\infty \in [M]$.
Next, we prove that $u^T M_k u \xrightarrow{k\to\infty} u^T M_\infty u, \forall u \in \mathbb{R}^n$. Indeed, since

$$\left|u^T M_k u - u^T M_\infty u\right| \;=\; \left|\sum_{i,j} u_i \left(m_{k,ij} - m_{\infty,ij}\right) u_j\right| \leq \|M_k - M_\infty\|_F \sum_{i,j} |u_i u_j|$$

and

$$\|M_k - M_\infty\|_F \xrightarrow{k\to\infty} 0 \,,$$

it is induced that $u^T M_k u \xrightarrow{k\to\infty} u^T M_\infty u$, $\forall u \in \mathbb{R}^n$.
Then, since $u^T M_k u \geq 0$, $\forall u \in \mathbb{R}^n$, so it is impossible that $u^T M_\infty u < 0$ for any $u \in \mathbb{R}^n$. Therefore $u^T M_\infty u \geq 0, \forall u \in \mathbb{R}^n$ or equivalently $M_\infty \succeq 0$.

c) Since the operator norm is a continuous function and $S([M])$, $S_+([M])$ are compact in $S(n)$, then Γ, Γ_+ are compact in $\mathbb{R}$.

d) The result is induced by extreme value theorem using the compactness of Γ, Γ_+ and the fact that $S_+([M]) \subseteq S([M])$ and hence:

$$\sup_{M\in S_+([M])} \{\lambda_{\max}(M)\} = \sup \Gamma_+ \leq \sup_{M\in S([M])} \{\lambda_{\max}(M)\} \leq \sup \Gamma < \infty.$$

□

Proposition 5. *The following statements hold:*

a) $S([M]) \preceq \alpha I$ *iff* $\alpha \geq \alpha^*$ *and* $S_+([M]) \preceq \alpha I$ *iff* $\alpha \geq \alpha_+^*$.

b) $\mathcal{E} \triangleq \left\{M \in S_+([M]) : Diag(M) = Diag\left(\overline{M}\right)\right\}$ *is the non empty set of maximal elements of* $S_+([M])$.

c) If $\mathcal{E}^c \triangleq S_+([M])\backslash\mathcal{E}$ *contains two elements* M, N *such that their entries* $m_{kl} \neq n_{kl}$ *for some tuple* $(k,l) : k \neq l,\ k,l \in \{1,...,n\}$, *then* $S_+([M])$ *has no greatest element.*

Proof. a) This statement is proved using Proposition 2:

$$S_+([M]) \preceq \alpha I \quad \Leftrightarrow \quad M \preceq \alpha I,\ \forall M \in S_+([M]) \quad \Leftrightarrow \quad \alpha \geq \alpha_+^*.$$

Similar argument is applied for $S([M])$.

b) Let $M \in S_+([M])$ (which is assumed to be non empty).
If $M \in \mathcal{E}$ then $\mathcal{E}$ is non empty. If $M \notin \mathcal{E}$, i.e. $\mathrm{Diag}(M) \neq \mathrm{Diag}(\overline{M})$, denote

$$\hat{M} = M + \Delta \quad , \qquad \Delta = -\mathrm{Diag}(M) + \mathrm{Diag}(\overline{M}).$$

Then $\hat{M} \in S_+([M])$ and satisfies $\mathrm{Diag}(\hat{M}) = \mathrm{Diag}(\overline{M})$. So $\mathcal{E}$ is non empty since $\hat{M} \in \mathcal{E}$. In addition, no matrix $M \notin \mathcal{E}$ is a maximal element of $S_+([M])$ since such a matrix M always has an upper bound $\hat{M}$ in $\mathcal{E}$. In other words, any maximal element of $S_+([M])$ (if it exists) must belong to $\mathcal{E}$.
Next, we prove that any element of $\mathcal{E}$ is a maximal element of $S_+([M])$. In fact, any matrix $M \notin \mathcal{E}$ is not an upper bound of an element $P \in \mathcal{E}$ since $\mathrm{Tr}(P) > \mathrm{Tr}(M)$ which contradicts the necessary condition of Proposition 3. Hence, we prove that any two elements of $\mathcal{E}$ are not an upper bound of each other. Let $P, Q \in \mathcal{E}$ such that $P \neq Q$ and $R = P - Q$. Then their entries satisfy $r_{ii} = 0$, $r_{ij} = p_{ij} - q_{ij}$ and $r_{ij} = r_{ji}$, $i,j \in \{1,...,n\}$. Assume that $Q \preceq P$ then

$$u^T R u = 2\sum_{i<j} u_i u_j r_{ij} \geq 0, \quad \forall u = (u_1,...,u_n) \neq 0.$$

Let $p, q \in \{1,...,n\}$, $p < q$, $\tilde{u} = e_p + e_q$ and $\hat{u} = e_p - e_q$ where e_p, e_q are standard unit vectors. Then:
$\tilde{u}^T R \tilde{u} = e_p^T R e_p + e_q^T R e_q + e_p^T R e_q + e_q^T R e_p = r_{pq} + r_{qp} = 2r_{pq} \geq 0,$
$\hat{u}^T L \hat{u} = e_p^T R e_p + e_q^T R e_q - e_p^T R e_q - e_q^T R e_p = -r_{pq} - r_{qp} = -2r_{pq} \geq 0,$
implying $r_{pq} = 0, \forall p, q \in \{1,...,n\}$ and $p < q$, which contradicts $P \neq Q$.

c) Let $M, N \in \mathcal{E}^c$ such that $m_{kl} \neq n_{kl}$ for some tuple $(k,l) : k \neq l,\ k,l \in \{1,...,n\}$. Let $P = M - \mathrm{Diag}(M) + \mathrm{Diag}(\overline{M})$ and $Q = N - \mathrm{Diag}(N) +$

Diag($\overline{M}$). Then P and Q belong to $\mathcal{E}$ and are two different maximal elements of $S_+([M])$. None of them is an upper bound of the other. Therefore $S_+([M])$ does not have the greatest element.

□

Corollary 1. *There exists a matrix $N^* \in \mathcal{E}$ such that $\lambda_{\max}(N^*) = \alpha_+^*$.*

Proof. By Proposition 4c) and the extreme value theorem, there exists a matrix $N \in S_+([M])$ such that $\lambda_{\max}(N) = \alpha_+^*$. If $N \notin \mathcal{E}$, then there exists a matrix $N^* \in \mathcal{E}$ such that $N \preceq N^*$. This implies that $\alpha_+^* = \lambda_{\max}(N) \leq \lambda_{\max}(N^*) \leq \alpha_+^*$ and hence $\lambda_{\max}(N^*) = \alpha_+^*$. □

Proposition 6. *The following statements hold:*

a) $\alpha_+^* I$ is the optimal upper bound of $S_+([M])$ in the set $BS_+([M])$ in the sense of operator norm minimization.

b) $\alpha_+^* I$ is the optimal upper bound of $S_+([M])$ in the set

$$\Omega = \left\{ K \in BS_+([M]) : \ n^{-1} \sum_{i=1}^{n} \lambda_i(K) \geq \alpha_+^* \right\}$$

in the sense of nuclear norm minimization.

Proof. a) Let $K \in BS_+([M])$, one gets:

$$\|\alpha_+^* I\| = \alpha_+^* \quad , \quad \|K\| = \lambda_{\max}(K) \quad , \quad S_+([M]) \preceq K \preceq \lambda_{\max}(K) I.$$

By Proposition 5a), since $S_+([M]) \preceq \lambda_{\max}(K) I$ then $\lambda_{\max}(K) \geq \alpha_+^*$. So $\|\alpha_+^* I\| \leq \|K\|$, $\forall K \in BS_+([M])$.

b) The result is straightforward by verifying $\|\alpha_+^* I\|_* \leq \|K\|_*$ for all $K \in BS_+([M])$ such that $\alpha_+^* \leq (\lambda_1(K) + ... + \lambda_n(K))/n$.

□

Proposition 7. *Let $Max([M]) = (max_{ij})$ be a matrix determined by*

$$max_{ij} = \begin{cases} \sup([m_{ij}]) & , \ \textit{if } \operatorname{mid}([m_{ij}]) \geq 0 \\ \inf([m_{ij}]) & , \ \textit{otherwise} \end{cases} \tag{2.2}$$

then

$$\alpha_+^* \ \leq \ \alpha^* \ \leq \ \sup\{\|M\|_F : M \in [M]\} \ \leq \ \|Max([M])\|_F.$$

In addition, if $Max([M]) \succeq 0$, then

$$\lambda_{\max}(Max([M])) \ \leq \ \alpha_+^* \ \leq \ \|Max([M])\|_F.$$

Proof. For any $M \in [M]$,

$$0 \leq |m_{ij}| \leq \max\{|\sup([m_{ij}])|, |\inf([m_{ij}])|\} = |max_{ij}|$$

then

$$\begin{aligned} & \sum_{i,j} |m_{ij}|^2 \leq \sum_{i,j} |max_{ij}|^2 \ , \ \forall M \in [M] \\ \Rightarrow \quad & \|M\|_F \leq \|Max([M])\|_F \ , \ \forall M \in [M]. \\ \Rightarrow \quad & \alpha_+^* = \lambda_{\max}(N^*) = \|N^*\| \leq \|N^*\|_F \leq \|Max([M])\|_F. \end{aligned}$$

where the matrix N^* is the one stated in Corollary 1.
The last conclusion of the proposition is straightforward. □

The following two theorems gather relevant properties of previous propositions and corollary. They provide the proof of the existence of an optimal upper bound of $S_+([M])$, the set of symmetric positive semidefinite matrices belonging to a given symmetric interval matrix $[M]$, and a simple way to localize this optimal upper bound. These theorems are useful to deal with covariances matrices belonging to a given symmetric interval matrix.

Theorem 4 (**Existence of Optimal upper bounds**). *The following properties hold:*

i) $\alpha_+^* \triangleq \sup_{M \in S_+([M])} \{\lambda_{\max}(M)\} < \infty$ *and* $S_+([M]) \preceq \alpha I$ *iff* $\alpha \geq \alpha_+^*$.

ii) $\alpha_+^* I$ *is the optimal upper bound of* $S_+([M])$ *in the set* $BS_+([M])$ *in the sense of operator norm minimization.*

iii) Let $\Omega = \left\{K \in BS_+([M]) : \ n^{-1} \sum_{i=1}^n \lambda_i(K) \geq \alpha_+^*\right\}$. $\alpha_+^* I$ *is the optimal upper bound of* $S_+([M])$ *in* Ω *in the sense of nuclear norm minimization.*

α_+^* *is said the optimal value of* $BS_+([M])$.

Theorem 5 (**Bounds of Optimal value** α_*). *The following properties hold:*

i) $\mathcal{E} \triangleq \{M \in S_+([M]) : Diag(M) = Diag(\sup([M]))\}$ *is the non empty set of maximal elements of* $S_+([M])$.

ii) There exists a matrix $N^* \in \mathcal{E}$ *such that* $\lambda_{\max}(N^*) = \alpha_+^*$.

iii) Let $Max([M]) = (max_{ij})$ *be a matrix determined by (2.2) then*

$$\alpha_+^* \leq \sup_{M \in [M]} \{\|M\|_F\} \leq \|Max([M])\|_F.$$

If $Max([M]) \succeq 0$ *then :* $\quad \lambda_{\max}(Max([M])) \leq \alpha_+^* \leq \|Max([M])\|_F.$

Remark 6. The optimal upper bound is not unique depending on the choice of criteria. It is not unique even in the norm minimization criterion since different matrices can have the same norm. However, $\alpha_+^* I$ might be the simplest one to use both in practice and theory.

Ideally, we have to find $N^* \in \mathcal{E}$ to determine α_+^* but this is quite intractable. An alternative way to localize the optimal value α_+^* is provided by Theorem 5 using the $\text{Max}([M])$ matrix. By definition, $\text{Max}([M]) \in S([M])$. If, in addition, $\text{Max}([M]) \succeq 0$, then $\text{Max}([M]) \in \mathcal{E}$, its nuclear norm and Frobenius norm are both maximum among $S_+([M])$. So, its operator norm, $\lambda_{\max}(\text{Max}([M]))$, in that case, might be very close to α_+^*. In many case, we can design or modify $[M]$ so that $\text{Max}([M]) = \sup([M]) \succeq 0$.

A more tighter upper bound of α_+^* is $\alpha^* = \sup\{\lambda_{\max}(M) : M \in S([M])\}$ since $\alpha_+^* \leq \alpha^* \leq \|\text{Max}([M])\|_F$. α^* is studied by many authors, for instance (Hertz, 1992) or (Hladik, 2013) and references therein, while the eigenvalue bounds of a square interval matrix can be referred to (Rohn, 1998) . We also refer to Gerschgorin circle, see e.g. (Meyer, 2000), for bounds of eigenvalues of square matrices. It may recommended to use intersection of all aforementioned bounds for more accurate choice of an approximate value of α_+^*. In the worst (and guaranteed) case, we can use $\alpha_+^* = \|\text{Max}([M])\|_F$ where this choice might be just a scale of the actual value of α_+^*. Any alternative choice of α_+^* between its bounds is meaningful especially when applying in interval computation. In many situations, a simple approximate choice of α_+^* might be more appreciated than using a complex algorithm to find its actual value. □

Proposition 8. *The following properties hold:*

a) (Proposition 1 in (Tran et al., 2017)) Let M, N be two real matrices of the same dimension, then

$$MN^T + NM^T \preceq t^{-1}MM^T + tNN^T, \ \forall t > 0. \tag{2.3}$$

b) If $\{M_u\}_{u=1:n}$ is a sequence of real matrices, then

$$\left(\sum_{u=1}^{n} M_u\right)\left(\sum_{u=1}^{n} M_u\right)^T \preceq \sum_{u=1}^{n}\left(1 + \sum_{v=1,\neq u}^{n} \sigma_{u,v}\right) M_u M_u^T \tag{2.4}$$

provided that $\sigma_{u,v} = \sigma_{v,u}^{-1} > 0$, $\forall u \in \{1:n\}$, $v \in \{1:n\} \setminus \{u\}$.

Proof. (2.3) holds thanks to $(M - tN)(M - tN)^T \succeq 0$, $\forall t > 0$.

Consider

$$\left(\sum_{u=1}^{n} M_u\right)\left(\sum_{u=1}^{n} M_u\right)^T = \sum_{u=1}^{n} M_u M_u^T + \sum_{u\neq v} M_u M_v^T$$

where $u, v \in \{1, ..., n\}$ and the last term of the right hand side is such that

$$\begin{aligned}
\sum_{u \neq v} M_u M_v^T &= \sum_{u<v} M_u M_v^T + \sum_{u>v} M_u M_v^T = \sum_{u<v} M_u M_v^T + \sum_{v>u} M_v M_u^T \\
&= \sum_{u<v} \left(M_u M_v^T + M_v M_u^T \right) \\
&\preceq \sum_{u<v} \left(\sigma_{u,v} M_u M_u^T + \sigma_{u,v}^{-1} M_v M_v^T \right), \ \forall \sigma_{u,v} > 0, u < v, \ \text{(using (2.3))}
\end{aligned}$$

Noting that there are $m = \frac{n^2-n}{2}$ real scalars $\sigma_{u,v} > 0$ such that $u < v$ and their m inverses $\sigma_{u,v}^{-1}$ in the above expression.
Putting $\sigma_{v,u} = \sigma_{u,v}^{-1}$ for all $v > u$ then

$$\begin{aligned}
& \sum_{u \neq v} M_u M_v^T \\
\preceq & \sum_{u<v} \sigma_{u,v} M_u M_u^T + \sum_{u<v} \sigma_{v,u} u M_v M_v^T, \ \forall \sigma_{u,v} > 0 : \sigma_{v,u} = \sigma_{u,v}^{-1}, \\
= & \sum_{t=1}^{n-1} \left(\sum_{s=t+1}^{n} \sigma_{t,s} \right) M_t M_t^T + \sum_{t=1}^{n-1} \sum_{s=t+1}^{n} \sigma_{s,t} M_s M_s^T \ , \ \forall \sigma_{t,s} > 0 : \sigma_{s,t} = \sigma_{t,s}^{-1}, \\
= & \sum_{t=1}^{n-1} \left(\sum_{s=t+1}^{n} \sigma_{t,s} \right) M_t M_t^T + \sum_{s=2}^{n} \sum_{t=1}^{s-1} \sigma_{s,t} M_s M_s^T \ , \ \forall \sigma_{t,s} > 0 : \sigma_{s,t} = \sigma_{t,s}^{-1}, \\
= & \left(\sum_{v=2}^{n} \sigma_{1,v} \right) M_1 M_1^T + \left(\sum_{v=1}^{n-1} \sigma_{n,v} \right) M_n M_n^T \\
& + \sum_{u=2}^{n-1} \left(\sum_{v=u+1}^{n} \sigma_{u,v} + \sum_{v=1}^{u-1} \sigma_{u,v} \right) M_u M_u^T \ , \ \forall \sigma_{u,v} > 0 : \sigma_{v,u} = \sigma_{u,v}^{-1}, \\
= & \sum_{u=1}^{n} \left(\sum_{v=1, \neq u}^{n} \sigma_{u,v} \right) M_u M_u^T \ , \forall \sigma_{u,v} > 0 : \sigma_{v,u} = \sigma_{u,v}^{-1}.
\end{aligned}$$

Consequently (2.4) is concluded. □

2.3 Optimal Upper Bound Interval Kalman Filter (OUBIKF)

In the previous section, it is pointed out that the matrix $\alpha_+^* I$ is the optimal upper bound of the set $S_+([M])$ among its other upper bounds according to the operator norm, for a given interval matrix $[M]$. This particular form of upper bound simplifies many computations and provides tractable forms of the obtained results subject to further optimization, without which the optimization problem might be unable to be solved. Although the exact value of α_+^* can not be found but its localization bounds are provided and help us to use alternative approximate values of α_+^*. Thanks to the provided theory background and the use of this kind of upper bound, the developed filter in the next is devoted to be named Optimal Upper Bound Interval Kalman Filter (OUBIKF). The optimization is not only due to the use of this kind of upper bound but also due to the fact that the obtained upper bound will be further minimized in terms of its trace.

2.3.1 Principle of the Filter

Consider the following linear discrete time dynamical system

$$\begin{cases} x_k = A_k x_{k-1} + B_k u_k + w_k \ , \\ y_k = C_k x_k + D_k u_k + v_k \ , \end{cases} \qquad k \in \mathbb{N}^*, \tag{2.5}$$

in which $x_k \in \mathbb{R}^{n_x}$ and $y_k \in \mathbb{R}^{n_y}$ represent state variables and measurements respectively, $u_k \in \mathbb{R}^{n_u}$ inputs, $w_k \in \mathbb{R}^{n_x}$ state noises and $v_k \in \mathbb{R}^{n_y}$ measurement noises.

Assumptions A1. Matrices A_k, B_k, C_k, D_k are unknown, deterministic and belonging to given interval matrices $[A], [B], [C], [D]$ respectively. w_k, v_k are centered Gaussian vectors with covariance matrices Q_k and R_k belonging respectively to given interval matrices $[Q]$ and $[R]$. The initial state x_0 is also Gaussian with mean μ_0 and covariance matrix P_0. In addition, x_0, $\{w_1, ..., w_k\}$ and $\{v_1, ..., v_k\}$ are assumed to be mutually independent.

Aim. The developed Filter is aimed to get estimate intervals $[\hat{x}_{k|k}]$ which contain all admissible estimates $\hat{x}_{k|k}$ of real states x_k induced by mixed uncertainties.

Principle. OUBIKF follows the same structure of the SKF. In the prediction step, thanks to interval computations, the *a priori* estimate

$$[\hat{x}_{k|k-1}] = [A][\hat{x}_{k-1|k-1}] + [B]u_k$$

is provided. It contains all admissible estimates $\hat{x}_{k|k-1} = A_k\hat{x}_{k-1|k-1} + B_k u_k$ for all values of $A_k \in [A]$, $B_k \in [B]$ and $\hat{x}_{k-1|k-1} \in [\hat{x}_{k-1|k-1}]$. In the correction step, an interval estimator

$$[\hat{x}_{k|k}] \;=\; [\hat{x}_{k|k-1}] + K_k(y_k - [\hat{y}_k]), \quad \text{with} \quad [\hat{y}_k] \;=\; [C][\hat{x}_{k|k-1}] + [D]u_k,$$

is provided, in which the gain K_k is a point matrix chosen by an optimization strategy. Concretely, the choice of K_k is proceed by a two stages optimization considering the class of upper bounds

$$\Gamma \;\triangleq\; \left\{\overline{\varphi}_k(K_k, \beta) : S_+([P_{k|k}]) \preceq \overline{\varphi}_k(K_k, \beta)\right\}$$

where $[P_{k|k}]$ is the interval matrix containing all admissible estimation error covariances $P_{k|k}$ and the form of $\overline{\varphi}_k(K_k, \beta)$ will be clarified in the next. Γ also includes the upper bound of $S_+([P_{k|k}])$ used in (Tran et al., 2017). Each upper bound in Γ is seen as a function of two arguments: gain matrix K_k and real parameter $\beta > 0$. The optimization is performed first in terms of K_k and then with respect to β in order to get the optimal bound $\overline{\varphi}_k^*$ of $S_+([P_{k|k}])$ among others in Γ. Finally, the Filter is developed and applied with the guaranteed conditions under which the model should be designed to obtain the filter stability in the sense that the $\text{Tr}\{\overline{\varphi}_k^*\}$ is non-asymptotically and asymptotically bounded. This means that the $\text{Tr}\{\overline{\varphi}_k^*\}$ is not exploded and hence the resulted estimator width is not exploded either. The Filter is then called *C-stable* as defined in Definition 6.

Again, we recall that the conservatism of interval computations is a major issue of all interval filters whose objective is to find interval estimates rather than point estimates for real states, so it is worthy to define

Definition 6. An interval filter is called *C-stable* if the widths of interval estimators for all time instant k are upper bounded by a common constant C.

2.3.2 First stage optimization of the Filter

In order to enter into the OUBIKF optimization, the following notations are necessary. They also provide a new way to prove that the gain matrix of SKF minimizes the trace of the estimation error covariance $P_{k|k}$.

For any $K_k \in \mathbb{R}^{n_x \times n_y}$, $k \geq 1$, define:

$$\varphi_k(K_k) \triangleq (I - K_k C_k)P_{k|k-1}(I - K_k C_k)^T + K_k R_k K_k^T, \tag{2.6}$$

then

$$\varphi_k(K_k) = P_{k|k} \quad , \quad \varphi_k(\mathbf{0}) = P_{k|k-1}, \tag{2.7}$$

where $\mathbf{0}$ is the zero matrix whose dimension is appropriate to the context, e.g. $\mathbf{0} \in \mathbb{R}^{n_x \times n_y}$ in this case, $P_{k|k-1}$ and $P_{k|k}$ are respectively prediction and estimation error covariances in SKF.
Using the SKF optimal gain $K_k^* = P_{k|k-1}C_k^T S_k^{-1}$ with $S_k = C_k P_{k|k-1} C_k^T + R_k$ and assuming S_k is nonsingular [1], one gets

$$\varphi_k(K_k^*) = (I - K_k^* C_k)\varphi_k(\mathbf{0}) = (I - K_k^* C_k)P_{k|k-1}.$$

The Theorem 6 in the following provides the optimal gain expression K_k^* and in the same time emphasizes that using K_k^*, $\mathrm{Tr}\{P_{k|k}\} = \mathbb{E}(\|x_k - \hat{x}_{k|k}\|^2)$ is minimized and hence the estimator $\hat{x}_{k|k}$ is better than $\hat{x}_{k|k-1}$ in the sense of mean square error minimization.

Theorem 6. *Consider system (1.1) with SKF assumptions. Then for any $k \geq 1$:*

$$0 \preceq \varphi_k(K_k^*) \preceq \varphi_k(K_k) \quad , \forall K_k \in \mathbb{R}^{n_x \times n_y}\ , \tag{2.8}$$
$$K_k^* = \mathrm{argmin}_{K_k} \mathrm{Tr}\{\varphi_k(K_k)\} = \mathrm{argmin}_{K_k} \mathrm{Tr}\{P_{k|k}\}. \tag{2.9}$$

Proof. Since any covariance matrix is positive semidefinite, then $\varphi_k(K_k) \succeq 0$, $\forall K_k$, and hence $\varphi_k(K_k^*) \succeq 0$. By assumptions, $S_k \in S_+(n_y)$ and is nonsingular, then

$$\begin{array}{rcll} 0 & \preceq & (K_k - P_{k|k-1}C_k^T S_k^{-1}) S_k (K_k - P_{k|k-1}C_k^T S_k^{-1})^T & \text{(Lemma 3)} \\ & = & K_k S_k K_k^T - K_k C_k P_{k|k-1} - P_{k|k-1}C_k^T K_k^T & \\ & + & P_{k|k-1}C_k^T S_k^{-1} C_k P_{k|k-1} & \\ & = & \varphi_k(K_k) - P_{k|k-1} + P_{k|k-1}C_k^T S_k^{-1} C_k P_{k|k-1} & \\ & = & \varphi_k(K_k) - \varphi_k(K_k^*), & \end{array}$$

which implies that $\varphi_k(K_k^*) \preceq \varphi_k(K_k)$, $\forall K_k \in \mathbb{R}^{n_x \times n_y}$.

Then, by Proposition 3, it implies $\mathrm{Tr}\{\varphi_k(K_k^*)\} \leq \mathrm{Tr}\{\varphi_k(K_k)\}$, $\forall K_k \in \mathbb{R}^{n_x \times n_y}$ and hence (2.9) is concluded. □

In the next, the class Γ of upper bounds to be optimized in terms of the gain K_k is presented by Theorem 7 which is the main contribution to the first stage optimization of OUBIKF.

1. The nonsingularity of S_k can be assured if R_k is assumed to be positive definite or more strictly measurement noises are assumed to be vectors of independent random components. In practice, the pseudo-inverse S_k^+ is used instead with notice that $S_k^+ = S_k^{-1}$ when the later exists.

Theorem 7. *Consider system (1.1) with Assumptions A1. Denote* $[C] = ([c_{ij}])$ *and* $M = mid([C])$. *Let* $R_{ij} = (r_{ij,uv})$ *be a matrix whose elements are zeros except its* ij*-th entry* $r_{ij,ij} = rad([c_{ij}])$ *and* n_0 *the number of non null radius of* $rad([C])$. *Denote also* $\Sigma = \sum_{i,j} R_{ij}R_{ij}^T = Diag\{rad([C])rad([C])^T\}$. *The following statements hold:*

1) $\forall k \geq 1$, $\forall A_k \in [A]$, $\forall C_k \in [C]$, $\forall Q_k \in [Q]$, $\forall R_k \in [R]$, $\forall \hat{x}_{k|k} \in [\hat{x}_{k|k}]$, $\forall \beta > 0$ *and* $\forall K_k \in \mathbb{R}^{n_x \times n_y}$*:*

$$\begin{aligned} P_{k|k} &\preceq (1+\beta^{-1}n_0)\,(I_{n_x} - K_k\,M)\,P_{k|k-1}\,(I_{n_x} - K_k\,M)^T \\ &+ K_k\left[(\beta+n_0)\sum_{i=1}^{n_y}\sum_{j=1}^{n_x} R_{ij}P_{k|k-1}R_{ij}^T + R_k\right]K_k^T, \end{aligned} \tag{2.10}$$

2) If $P_{k|k-1} \in [P_{k|k-1}]$, $S_+([P_{k|k-1}]) \preceq \alpha_k I$ *and* $S_+([R]) \preceq \gamma I$, *then* $\forall\, \beta > 0$ *and* $\forall\, K_k \in \mathbb{R}^{n_x \times n_y}$*:*

$$\begin{aligned} P_{k|k} &\preceq \alpha_k(1+\beta^{-1}n_0)\,(I_{n_x} - K_k\,M)\,(I_{n_x} - K_k\,M)^T \\ &+ K_k\left[\alpha_k\,(\beta+n_0)\,\Sigma + \gamma I_{n_y}\right]K_k^T, \end{aligned} \tag{2.11}$$

3) Denote the right hand side of (2.11) by $\overline{\varphi}_k(K_k,\beta)$.
Denote also $S_{k,\beta} = MM^T + \beta\Sigma + \frac{\gamma}{\alpha_k(1+n_0/\beta)}I_{n_y}$ *and* $\overline{K}_{k,\beta}^* = M^T S_{k,\beta}^{-1}$. *Then:*

$$\overline{\varphi}_k(\overline{K}_{k,\beta}^*,\beta) = \alpha_k(1+n_0\beta^{-1})(I_{n_x} - \overline{K}_{k,\beta}^* M), \tag{2.12}$$

$$0 \preceq \overline{\varphi}_k(\overline{K}_{k,\beta}^*,\beta) \preceq \overline{\varphi}_k(K_k,\beta)\,, \quad \forall K_k \in \mathbb{R}^{n_x \times n_y}, \forall \beta > 0, \tag{2.13}$$

$$\overline{K}_{k,\beta}^* = \text{argmin}_{K_k}\,\text{Tr}\{\overline{\varphi}_k(K_k,\beta)\}. \tag{2.14}$$

Proof. 1) Let $C_k \in [C]$. Using the decomposition $C_k = M + \Delta_k$ where $\Delta_k = \sum_{i=1}^{n_y}\sum_{j=1}^{n_x}\alpha_{ij}(k)R_{ij}$ for appropriate $\alpha_{ij}(k) \in [-1,1]$, $i \in \{1,...,n_y\}$, $j \in \{1,...,n_x\}$, one gets

$$\begin{aligned} P_{k|k} &= (I - K_kC_k)P_{k|k-1}(I - K_kC_k)^T + K_kR_kK_k^T \\ &= \Lambda_1 + \Lambda_2 + \Lambda_3 + K_kR_kK_k^T, \end{aligned}$$

where

$$\begin{aligned} \Lambda_1 &= (I - K_k\,M)\,P_{k|k-1}\,(I - K_k\,M)^T\,, \\ \Lambda_2 &= (K_k\Delta_k)\,P_{k|k-1}\,(K_k\Delta_k)^T\,, \\ \Lambda_3 &= -(I - K_k\,M)\,P_{k|k-1}\,(K_k\Delta_k)^T - (K_k\Delta_k)\,P_{k|k-1}\,(I - K_k\,M)^T\,. \end{aligned}$$

Since $P_{k|k-1}$ can be expressed as $P_{k|k-1} = P_{k|k-1}^{1/2}\left(P_{k|k-1}^{1/2}\right)^T$ and by applying (2.3) (Appendix), one gets

$$\Lambda_3 \preceq \sum_{i,j} T_{ij}\left\{\beta_{ij}^{-1}\Lambda_1 + \beta_{ij}K_k\,R_{ij}P_{k|k-1}\,(K_k\,R_{ij})^T\right\},$$

for any $\beta_{ij} > 0$, where $T_{ij} = 1$ when $\text{rad}([c_{ij}]) > 0$ and null otherwise. Applying (2.4) (Appendix), then

$$\Lambda_2 \quad \preceq \quad K_k \left[\sum_{i,j} T_{ij} \left(\sum_{u,v} T_{uv} \sigma_{i,j,u,v} \right) R_{ij} P_{k|k-1} R_{ij}^T \right] K_k^T,$$

for all $\sigma_{i,j,u,v} > 0$ such that $\sigma_{i,j,u,v} = \sigma_{u,v,i,j}^{-1}$. Therefore,

$$\Lambda_2 \quad \preceq \quad K_k \left[\inf \sup \left\{ \sigma_{i,j,u,v} > 0 : \sigma_{i,j,u,v} = \sigma_{u,v,i,j}^{-1} \right\} \sum_{i,j} T_{ij} \sum_{u,v} T_{uv} R_{ij}, P_{k|k-1} R_{ij}^T \right] K_k^T,$$

noting that $\inf \sup \left\{ \sigma_{i,j,u,v} > 0 : \sigma_{i,j,u,v} = \sigma_{u,v,i,j}^{-1} \right\} = 1$.

Choose $\beta_{ij} = \beta > 0$ for all i, j and get

$$\begin{aligned} P_{k|k} \quad &\preceq \quad \left(1 + \sum_{i,j} \beta^{-1} T_{ij} \right) \Lambda_1 \; + \; K_k R_k K_k^T \\ &+ \quad K_k \left[\sum_{i,j} T_{ij} \left(\beta + \sum_{u,v} T_{uv} \right) R_{ij} P_{k|k-1} R_{ij}^T \right] K_k^T, \end{aligned}$$

noting that $\sum_{i=1}^{n_y} \sum_{j=1}^{n_x} T_{ij} = n_0$ and $\sum_{i,j} T_{ij} R_{ij} P_{k|k-1} R_{ij}^T = \sum_{i,j} R_{ij} P_{k|k-1} R_{ij}^T$. Then, (2.10) holds.

2) This statement is directly implied by using the property (issued from Lemma 3):

$$A \preceq B \Rightarrow XAX^T \preceq XBX^T$$

for all X with appropriate dimension.

3) The proof of this statement can be derived in a similar way as the one of Theorem 6. □

Remark 7. The parameters $\beta_{ij}, \sigma_{i,j,u,v}, \beta, \sigma$ used in Theorem 7 and its proof depend actually on time instant k. □

Remark 8. Consider the proof of (2.10). When finding the upper bound of Λ_2, in (Tran et al., 2017), the choice $\beta_{ij} = \sigma_{i,j,u,v} = 1$, $\forall i, j, u, v$, is used while in (Lu et al., 2019), beside choosing $\beta_{ij} = \beta > 0$, $\forall i, j$, the choice $\sigma_{i,j,u,v} = \sigma > 0$, $\forall i, j, u, v$, is used regardless the condition $\sigma_{i,j,u,v} = \sigma_{u,v,i,j}^{-1}$. None of these studies provides a diligent investigation of an advanced optimization in terms of these real parameters (the optimization with respect to the gain matrix is always performed).

The present study considers a class of $P_{k|k}$'s upper bounds which are *optimal* with respect to the choice of

$$\sigma_{i,j,u,v} \equiv \sigma \equiv \sup\{\sigma_{i,j,u,v} > 0 : \sigma_{i,j,u,v} = \sigma^{-1}_{u,v,i,j}\} \equiv 1, \quad \forall i, j, u, v,$$

where $1 = \inf\sup\{\sigma_{i,j,u,v} > 0 : \sigma_{i,j,u,v} = \sigma^{-1}_{u,v,i,j}\}$. Any choice of $\sigma > 1$ will be called *superoptimal*. In addition, notice that $P_{k|k}$ is bounded above by a number of upper bounds which are not necessarily tight. Only the last sum of these upper bounds, the right hand side of (2.11), is further optimized thanks to its tractable form. Therefore, other choices of σ in $(0, 1)$ might yield, although it is not guaranteed, an upper bound of $P_{k|k}$. This choice of small σ (inferior to 1) is called *suboptimal*.

The emphasized terms can also be used for the choices of other parameters, e.g. β, α_k and γ, to provide corresponding $P_{k|k}$'s upper bounds. In general, a suboptimal choice of an upper bound might be compensated partially or totally by other superoptimal upper bounds in the sum. This results in the following Algorithm 2, a main contribution of (Lu et al., 2019), being explained in light of new viewpoint of the present study and named as the *OUBIKF Beta version*. It provides numerous choices of $P_{k|k}$'s upper bounds (via β and σ parameters) to obtain (more) reliable estimators in many situations where one of them is illustrated by Example 2. The optimal bound, corresponding to the choice of $\sigma = 1$, is further optimized with respect to β in the second stage optimization presented in the next section. □

Example 2 (**Academic example**). This example is issued from (Lu et al., 2019) in order to illustrate the algorithm working with small β, σ and compare its results to those of the proposed method UBIKF of (Tran et al., 2017). The system under consideration is described by equation (2.5) without input u_k, where:

$$[A] = \begin{pmatrix} [2.45, 2.72] & [-1.41, -1.28] & [0.26, 0.28] \\ [6.32, 6.98] & [-3.56, -3.22] & [2.45, 2.72] \\ [-0.79, -0.72] & [0.3, 0.34] & [0.1, 0.11] \end{pmatrix},$$

$$[C] = \begin{pmatrix} [-8.16, -7.84] & [-4.08, -3.92] & [1.96, 2.04] \\ [-2.04, -1.96] & [1.96, 2.04] & [5.88, 6.12] \\ [-0.41, -0.39] & [15.68, 16.32] & [6.86, 7.14] \end{pmatrix},$$

$$[Q] = [R] = \begin{pmatrix} [8, 12] & [-6, -4] & [3.2, 4.8] \\ [-6, -4] & [8, 12] & [1.6, 2.4] \\ [3.2, 4.8] & [1.6, 2.4] & [8, 12] \end{pmatrix}.$$

Algorithm 2 OUBIKF Beta version

Initialization:
  $[\hat{x}_{0|0}]$, $\mathcal{P}_{0|0}$, $[A]$, $[B]$, $[C]$, $[D]$, $[Q]$, $[R]$, u_k, y_k, $k = 1, 2, 3, ..., N$
  Find n_0 the number of non zero radius of $[C]$
  Find γ such that $S_+([R]) \preceq \gamma I$
for $k = 1, 2, 3, ...N$ **do**
  Prediction step:
    $[\hat{x}_{k|k-1}] = [A][\hat{x}_{k-1|k-1}] + [B]u_k$
    $[P_{k|k-1}] = [A]\mathcal{P}_{k-1|k-1}[A]^T + [Q]$
    Find α_k such that $S_+([P_{k|k-1}]) \preceq \alpha_k I$
  Correction step:
    Choose $\beta_k > 0$ and $\sigma_k > 0$
    $\tau_k = \dfrac{\beta_k + n_0\sigma_k}{1 + n_0/\beta_k}$; $v_k = \dfrac{\gamma_k}{\alpha_k(1 + n_0/\beta_k)}$
    $S_k = \text{mid}([C])\text{mid}([C])^T + \tau_k \text{Diag}\left\{\text{rad}([C])\text{rad}([C])^T\right\} + v_k I$
    $K_k = \text{mid}([C])^T S_k^{-1}$
    $[\hat{x}_{k|k}] = (I - K_k[C])[\hat{x}_{k|k-1}] + K_k(y_k - [D]u_k)$
    $\mathcal{P}_{k|k} = (I - K_k\text{mid}([C]))\alpha_k(1 + n_0/\beta_k)$
end for

The initial state is $x_0 = (5, -2, 6)^T$ and the algorithm starts at $[\hat{x}_0] = ([-2, 2], [-2, 2], [-2, 2])^T$. The initial error covariance bound is $\mathcal{P}_{0|0} = 10I$. The vector x_k has three components $x_{k,1}, x_{k,2}, x_{k,3}$ which are states in consideration of the system.

Firstly, state variables x_k, measures y_k and error covariance matrices $P_{k|k}$ corresponding to the SKF are simulated for $N = 10^4$ iteration steps. More precisely, at each time instant k, matrices A_k, C_k, Q_k, R_k are generated respectively from $[A]$, $[C]$, $[Q]$, $[R]$ such that Q_k and R_k are symmetric positive semidefinite. w_k, v_k are simulated such that $w_k \sim \mathcal{N}(0, Q_k)$ and $v_k \sim \mathcal{N}(0, R_k)$. Then, x_k, y_k and $P_{k|k}$ are computed straightforwardly using their corresponding expressions. Next, Algorithm 2 is run together with the one of (Tran et al., 2017), namely UBIKF, for N steps. The outputs of Algorithm 2 are $[\hat{x}_k^{opt}]$, $\mathcal{P}_{k|k}^{opt}$ and those of UBIKF are $[\hat{x}_k]$, $\mathcal{P}_{k|k}$, where $\mathcal{P}_{k|k}^{opt}$ and $\mathcal{P}_{k|k}$ are upper bounds of the set $S_+([P_{k|k}])$ yielded respectively by the two algorithms.

In the use of Algorithm 2, any upper bound αI of the corresponding set $S_+([.])$ is chosen with $\alpha = \|\text{Max}([.])\|_F$ where Max$([.])$ is defined by (2.2). Also, the choice $\beta_k = \frac{1}{2.n_0.10^3}$ and $\sigma_k = \frac{1}{n_0.10^3}$ are applied for the algorithm.

Simulation results. Using Algorithm 2, the computation time is reduced more than 40% with respect to the one of UBIKF (Table 2.1). The

traces of bounds $\mathcal{P}_{k|k}^{opt}$ decrease rapidly and have a convergence tendency while the traces of $\mathcal{P}_{k|k}$ increase (although bounded) (Fig. 2.1 and 2.2). In addition, $\mathrm{Tr}(P_{k|k}) \leq \mathrm{Tr}(\mathcal{P}_{k|k}^{opt}) \leq \mathrm{Tr}(\mathcal{P}_{k|k})$ for all $k \geq 1$ (Table 2.2). Besides, estimate intervals $[\hat{x}_k^{opt}]$ are contained in $[\hat{x}_k]$ for all $k \geq 1$.

	RMSE			Time
	$x_{k,1}$	$x_{k,2}$	$x_{k,3}$	
UBIKF	413.41	448.83	343.82	51.375 s
OUBIKF	416.95	451.48	346.51	28.719 s

Table 2.1 – Academic example - RMSE and computation times yielded by the OUBIKF Beta version and the UBIKF respectively for $N = 10^4$ iterations.

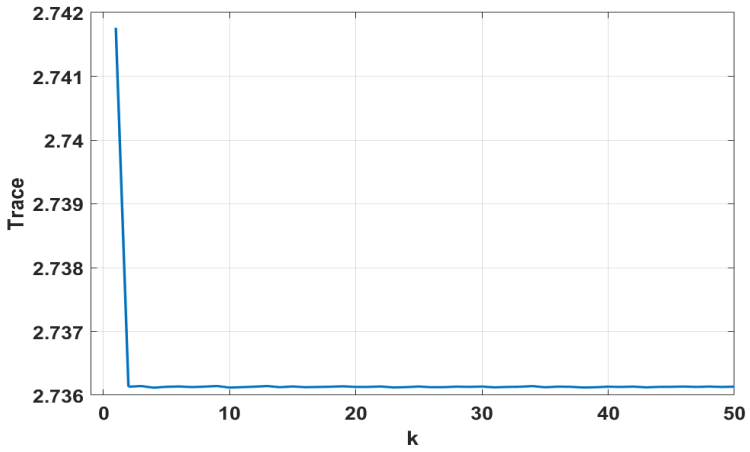

Figure 2.1 – Academic example - Behavior of the traces of error covariance upper bounds $\mathcal{P}_{k|k}^{opt}$ yielded by the OUBIKF Beta version.

Trace	Min	Mean	Max	Width*
$\mathrm{tr}(P_{k\|k})$	1.0592	1.3399	1.6418	0.5826
$\mathrm{tr}(\mathcal{P}_{k\|k}^{opt})$	2.7361	2.7361	2.7418	0.0057
$\mathrm{tr}(\mathcal{P}_{k\|k})$	15.353	132.72	133.47	118.117

*Width = Max - Min

Table 2.2 – Academic example - Traces of estimation error covariance $P_{k|k}$ and of their upper bounds according respectively to OUBIKF and UBIKF.

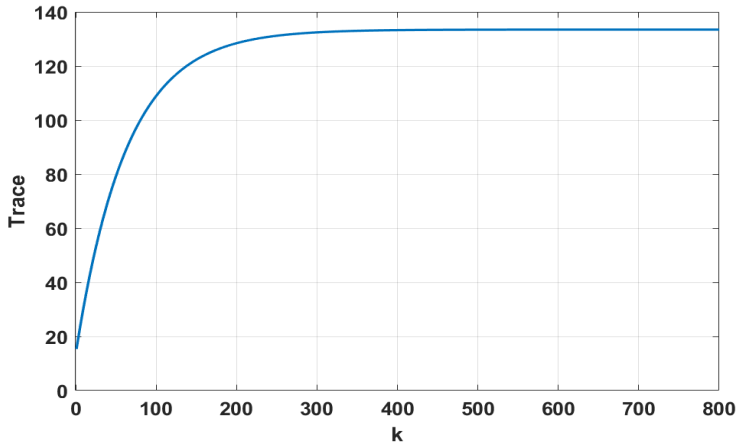

Figure 2.2 – Academic example - Behavior of the traces of error covariance upper bounds $\mathcal{P}_{k|k}$ yielded by the UBIKF.

The next result concerns the confidence intervals defined by

$$\begin{aligned} \mathrm{CI}_{k,r}^{opt} &= \left[\inf([\hat{x}_k^{opt}]) - r\sqrt{\mathrm{Diag}_v(\mathcal{P}_k^{opt})}\ ,\ \sup([\hat{x}_k^{opt}]) + r\sqrt{\mathrm{Diag}_v(\mathcal{P}_k^{opt})}\right], \\ \mathrm{CI}_{k,r} &= \left[\inf([\hat{x}_k]) - r\sqrt{\mathrm{Diag}_v(\mathcal{P}_k)}\ ,\ \sup([\hat{x}_k]) + r\sqrt{\mathrm{Diag}_v(\mathcal{P}_k)}\right] \end{aligned}$$

where $r = 1, 2, 3$ corresponding to $68\%, 95\%, 99.7\%$ confidence interval (the 3-sigma rule). According to the simulation, the 68% confidence intervals contain all corresponding state variables x_k and $\mathrm{CI}_{k,1}^{opt} \subseteq \mathrm{CI}_{k,1}$, for all $k \geq 1$ (Fig. 2.3). So the $O(\%)$, the percentage of confidence intervals containing corresponding state variables, are both 100% for two algorithms, however the $\mathrm{CI}_{k,1}^{opt}$'s are tighter.

We also deal with a criterion called *Root Mean Squared Error* (RMSE) to compare the performance of the two algorithms. The RMSE is defined in (Tran et al., 2017) by

$$RMSE = \sqrt{\frac{1}{N}\sum_{k=1}^{N}(x_k - \mathrm{mid}([\hat{x}_k]))^2}.$$

The result is that the RMSE of OUBIKF Beta version is slightly greater than the one of UBIKF (Table 2.1). This criterion is used here as it has been used before in (Tran et al., 2017) to compare different algorithms. But a critical point of view can be pointed out. The distance between the state x_k and the midpoint of the corresponding estimate interval is used. This fact dismisses the issue of the estimate interval width. Naturally, two estimate intervals

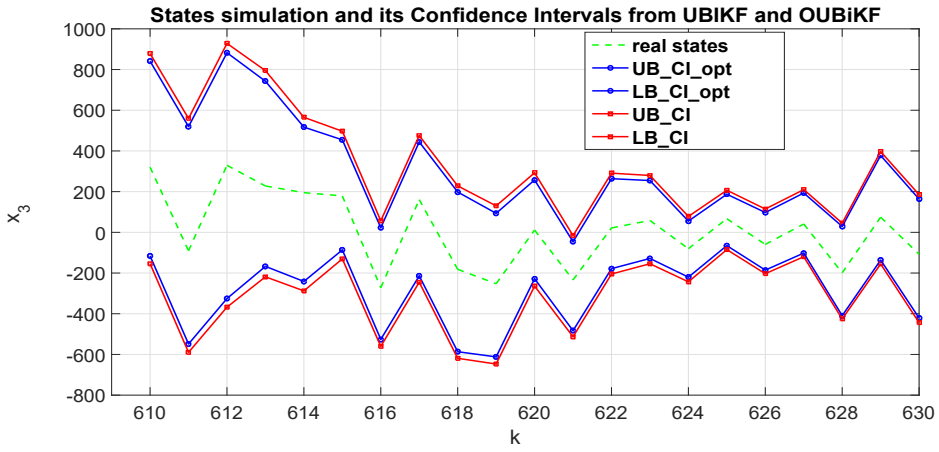

Figure 2.3 – Academic example - 68% Confidence Intervals yielded by the OUBIKF Beta version and the UBIKF with respect to the states $x_{k,3}$.

with a same midpoint have the same RMSE regardless their widths. In other words, this index just stands for the concentration tendency of states x_k with respect to the corresponding estimate interval midpoint. Another distance is proposed to improve the meaning of this criterion which is the Hausdorff distance determined by:

$$d_H(X,Y) = \max\left\{\sup_{x\in X}\inf_{y\in Y} d(x,y), \sup_{y\in Y}\inf_{x\in X} d(x,y)\right\}$$

for any two non empty sets X, Y in the metric space (Ω, d). In our case, we have

$$d_H(x_k, [\hat{x}_k]) = \max\left\{|x_k - \inf([\hat{x}_k])|, |x_k - \sup([\hat{x}_k])|\right\}$$

and the new RMSE is defined by

$$\widehat{RMSE} = \sqrt{\frac{1}{N}\sum_{k=1}^{N} d_H(x_k, [\hat{x}_k])^2}.$$

The result for $\widehat{RMSE}$ in Table 2.3 shows that the estimate intervals $[\hat{x}_k^{opt}]$ are more relevant by their tightness.

	$\widehat{RMSE}$		
	x_1	x_2	x_3
UBIKF	5047.2	4159.1	4213.6
OUBIKF	4708.5	3847.7	3934.2

Table 2.3 – The $\widehat{RMSE}$

□

2.3.3 Second stage optimization and guaranteed conditions of the Filter

In the next, in lieu of finding directly the optimal upper bound denoted by $\overline{\varphi}_k^* = \inf_{\beta>0} \overline{\varphi}_k(\overline{K}_{k,\beta}^*, \beta)$, the behavior of

$$\phi_k(\beta) \triangleq \mathrm{Tr}\{\overline{\varphi}_k(\overline{K}_{k,\beta}^*, \beta)\}, \quad \beta > 0,$$

is considered in the aim to find its minimum $\Phi_k^* \triangleq \inf_{\beta>0} \phi_k(\beta)$. The behavior of $\phi_k(\beta)$ is provided by Proposition 9 and illustrated in Fig. 2.4 and 2.5. The guaranteed conditions of the Filter is built afterward and the second stage optimization is contributed by Theorem 8.

The following notations are used:

- All notations defined in Theorems 6 and 7: φ_k, K_k^*, S_k, M, R_{ij}, Σ, n_0, α_k, γ, $\overline{\varphi}_k$, $\overline{K}_{k,\beta}^*$, $S_{k,\beta}$.
- $r = \mathrm{rank}(M)$,
- λ_i, $i = 1, ..., r$, are non null eigenvalues of MM^T,
- d_{ij}'s are entries of the diagonal matrix Σ,
- $d_{\min} = \min\{d_{ii} \neq 0, i = 1, ..., n_y\}$,
- $d_{\max} = \max\{d_{ii} \neq 0, i = 1, ..., n_y\}$.

Lemma 4. *Let $\alpha > 0, c > 0, \beta > 0$, $a(\beta) = \alpha(1 + n_0/\beta)$ and $\xi(\beta) = a(\beta)\left[n_x - \mathrm{Tr}\{M^T\left(MM^T + \beta c I_{n_y}\right)^{-1} M\}\right]$. Then*

$$\begin{aligned}
\lim_{\beta\to\infty} \xi(\beta) &= \alpha n_x = \alpha \mathrm{Tr}\{I_{n_x}\}, \\
\lim_{\beta\to 0} \xi(\beta) &= \begin{cases} \alpha c n_0 \mathrm{Tr}\{(MM^T)^+\} & , \quad n_x = r \\ \infty & , \quad n_x > r \end{cases} .
\end{aligned}$$

Furthermore:

a) If $n_x = r$ and $\lambda_i \geq n_0 c$, $\forall i \in \{1, ..., r\}$, then $\xi(\beta)$ is non-decreasing and

$$\inf_{\beta>0} \xi(\beta) = \lim_{\beta\to 0} \xi(\beta) = \alpha c n_0 \mathrm{Tr}\{(MM^T)^+\},$$
$$\sup_{\beta>0} \xi(\beta) = \lim_{\beta\to\infty} \xi(\beta) = \alpha n_x.$$

b) If $n_x > r$ and $\lambda_i \leq n_0 c$, $\forall i \in \{1, ..., r\}$, then $\xi(\beta)$ non-increasing and

$$\sup_{\beta>0} \xi(\beta) = \lim_{\beta\to 0} \xi(\beta) = \infty,$$
$$\inf_{\beta>0} \xi(\beta) = \lim_{\beta\to\infty} \xi(\beta) = \alpha n_x.$$

Proof. Since $MM^T \in S_+(n_y)$, it can be decomposed as $MM^T = Q\Lambda Q^T$, $QQ^T = I_{n_y}$ and $\Lambda = \mathrm{diag}\{\lambda_i(MM^T), i \in \{1, .., n_y\}\}$ with $r = \mathrm{rank}(M)$

non (null eigenvalues) positive λ_i of MM^T. This implies $(MM^T + sI)^{-1} = Q(\Lambda + sI)^{-1}Q^T$ for all $s \in \mathbb{R}$. Thus

$$\begin{aligned}\mathrm{Tr}\{M^T\left(MM^T + \beta c I_{n_y}\right)^{-1} M\} &= \mathrm{Tr}\{MM^T\left(MM^T + \beta c I_{n_y}\right)^{-1}\} \\ &= \mathrm{Tr}\{\Lambda\left(\Lambda + \beta c I_{n_y}\right)^{-1}\} = \sum_{i=1}^{r} \frac{\lambda_i}{\lambda_i + \beta c}\end{aligned}$$

and

$$\xi(\beta) = a(\beta)\left[n_x - \sum_{i=1}^{r} \frac{\lambda_i}{\lambda_i + \beta c}\right] = a(\beta)\sum_{i=1}^{r}\left[\frac{n_x}{r} - \frac{\lambda_i}{\lambda_i + \beta c}\right].$$

Since $0 < r = \mathrm{rank}(M) \leq \min\{n_x, n_y\}$ then $\dfrac{n_x}{r} = 1 + \delta$ for some $\delta \geq 0$, and

$$\begin{aligned}\xi(\beta) &= (1 + n_0/\beta)\alpha\delta r + \alpha c \sum_{i=1}^{r} \frac{\beta + n_0}{c\beta + \lambda_i}\,, \\ \frac{d\xi}{d\beta}(\beta) &= \frac{-\alpha\delta r}{\beta^2} + \alpha c \sum_{i=1}^{r} \frac{\lambda_i - n_0 c}{(c\beta + \lambda_i)^2}\,.\end{aligned}$$

The lemma conclusions are then straightforward. □

Proposition 9. *Let $k \geq 1$, $\epsilon > 0$, $c_{1,k} = d_{\min} + \frac{\gamma}{\alpha_k(\epsilon + n_0)}$ and $c_{2,k} = d_{\max} + \frac{\gamma}{\alpha_k n_0}$. Recall that $\phi_k(\beta) = \mathrm{Tr}\{\overline{\varphi}_k(\overline{K}^*_{k,\beta}, \beta)\}$, $\beta > 0$ and let*

$$\begin{aligned}h(\beta) &= \alpha_k(1 + n_0/\beta)\,[n_x - r]\,, \\ g(\beta) &= \alpha_k(1 + n_0/\beta)n_x, \\ \xi_{i,k}(\beta) &= \alpha_k(1 + n_0/\beta)\left[n_x - \mathrm{Tr}\{M^T\left(MM^T + \beta c_{i,k} I_{n_y}\right)^{-1} M\}\right],\ i \in \{1, 2\}.\end{aligned}$$

Then for all $0 < \beta \leq \epsilon$:

$$0 \leq h(\beta) \leq \xi_{1,k}(\beta) \leq \phi_k(\beta) \leq \xi_{2,k}(\beta) \leq g(\beta)\,, \tag{2.15}$$

a) If $n_x = r$ and $\lambda_i \geq n_0 d_{\max} + \frac{\gamma}{\alpha_k}$, $\forall i = 1, ..., r$, then

$$0 < \underline{c}_k.\mathrm{Tr}\{(MM^T)^+\} \leq \Phi^*_k \leq \lim_{\beta \to 0} \phi_k(\beta) \leq \overline{c}_k.\mathrm{Tr}\{(MM^T)^+\} < \alpha_k n_x$$

where $\underline{c}_k = \alpha_k n_0 d_{\min} + \gamma$, $\overline{c}_k = \alpha_k n_0 d_{\max} + \gamma$ and $\Phi^*_k = \inf_{\beta > 0} \phi_k(\beta)$.

b) If $n_x > r$ and $\lambda_i \leq n_0 d_{\min} + \frac{\gamma}{\alpha_k}$, $\forall i = 1, ..., r$, then

$$\infty = \lim_{\beta \to 0} \phi_k(\beta) \geq \xi_{2,k}(\beta) \geq \phi_k(\beta) \geq \xi_{1,k}(\beta) \geq \lim_{\beta \to \infty} \phi_k(\beta) = \alpha_k n_x = \Phi^*_k.$$

Proof. Using following facts

— $0 \preceq MM^T + \beta c_{1,k} I_{n_y} \preceq MM^T + \beta\Sigma + \frac{\gamma\beta}{\alpha_k(\beta+n_0)} I_{n_y} \preceq MM^T + \beta c_{2,k} I_{n_y}$,

— $A, B \in S_+(n)$ and $0 \preceq A \preceq B$ imply that

+ $0 \preceq B^+ \preceq A^+$ (note that $X^+ \equiv X^{-1}$ if X^{-1} exists),
+ $0 \preceq M^T A M \preceq M^T B M$, $\forall M \in \mathbb{R}^{n\times p}$,
+ $0 \preceq P + A \preceq P + B$, $\forall P \in \mathbb{R}^{n\times n}$,
+ $0 \preceq sA \preceq sB$, $\forall s > 0$ and $tB \preceq tA \preceq 0$, $\forall t < 0$,

and get for all $0 < \beta \leq \epsilon$ (note that $\xi_{1,k}(.)$ depends on ϵ):

$$0 \leq \xi_{1,k}(\beta) \leq \phi_k(\beta) \leq \xi_{2,k}(\beta) \leq \alpha_k(1 + n_0/\beta)n_x.$$

It remains to prove $0 \leq h_k(\beta) \leq \xi_{1,k}(\beta)$ for (2.15) to be true. It is obvious that $h_k(\beta) \geq 0$, $\forall\beta > 0$ since $r = \text{rank}(M) \leq \min\{n_x, n_y\}$. Then $h_k(\beta) \leq \xi_{1,k}(\beta)$ follows from the fact that

$$r \;\geq\; \sum_{i=1}^{r} \frac{\lambda_i}{\lambda_i + \beta c_{1,k}} = \text{Tr}\{M^T(M^T + \beta c_{1,k} I_{n_y})^{-1} M\} \;,\; \forall\beta > 0.$$

By (2.15) one gets:

$$\begin{array}{lccccll} & 0 & \leq & \inf\limits_{0<\beta\leq\epsilon} h(\beta) & \leq & \xi_{1,k}(\beta) \quad , & \forall\beta \in (0,\epsilon] \\ \Rightarrow & 0 & \leq & \inf\limits_{\epsilon>0}\inf\limits_{0<\beta\leq\epsilon} h(\beta) & \leq & \xi_{1,k}(\beta) \quad , & \forall\beta > 0, \forall\epsilon > 0 \\ \Rightarrow & 0 & \leq & \alpha_k(n_x - r) & \leq & \inf\limits_{\beta>0} \xi_{1,k}(\beta), & \forall\epsilon > 0. \end{array}$$

Then it is straightforward that

$$0 \leq \alpha_k(n_x - r) \leq \inf_{\beta>0} \xi_{1,k}(\beta) \leq \inf_{\beta>0} \phi_k(\beta) = \Phi_k^* \leq \inf_{\epsilon>0} \xi_{2,k}(\beta) \leq \alpha_k n_x, \forall\epsilon > 0$$

noting that $\xi_{1,k}(\beta)$ depends on ϵ. Therefore

$$0 \leq \alpha_k(n_x - r) \leq \sup_{\epsilon>0}\inf_{\beta>0} \xi_{1,k}(\beta) \leq \Phi_k^* \leq \inf_{\epsilon>0} \xi_{2,k}(\beta) \leq \alpha_k n_x.$$

Furthermore,

- $0 \leq \lim_{\beta\to 0} \xi_{1,k}(\beta) \leq \lim_{\beta\to 0} \phi_k(\beta) \leq \lim_{\beta\to 0} \xi_{2,k}(\beta)$,
- $\Phi_k^* = \inf\limits_{\beta>0} \phi_k(\beta) \leq \lim_{\beta\to 0} \phi_k(\beta)$.

Following results are based on Lemma 4.

a) For $n_x = r$ and $\lambda_i \geq n_0 c_{2,k} \geq n_0 c_{1,k}$, $\forall i \in \{1, ..., r\}$, one gets:

$$\begin{array}{lcccl} \inf\limits_{\beta>0} \xi_{i,k}(\beta) & = & \lim\limits_{\beta\to 0} \xi_{i,k}(\beta) & = & \alpha_k n_0 c_{i,k} \text{Tr}\{(MM^T)^+\}, \quad i \in \{1, 2\}, \\ \sup\limits_{\epsilon>0}\inf\limits_{\beta>0} \xi_{1,k}(\beta) & = & \sup\limits_{\epsilon>0} \alpha_k n_0 c_{1,k} \text{Tr}\{(MM^T)^+\} & = & \alpha_k n_0 d_{\min} + \gamma \end{array}$$

where $\alpha_k n_0 c_{1,k} = \alpha_k n_0 d_{\min} + \frac{\gamma n_0}{\epsilon + n_0}$, $\epsilon > 0$ and $\alpha_k n_0 c_{2,k} = \alpha_k n_0 d_{\max} + \gamma$. Substituting above results, the conclusion holds.

b) For $n_x > r$ and $\lambda_i \leq n_0 c_{1,k} \leq n_0 c_{2,k}$, $\forall i = 1, ..., r$, the functions $\xi_{j,k}$, $j \in \{1, 2\}$, are non-increasing and

$$\sup_{\beta>0} \xi_{j,k}(\beta) = \lim_{\beta \to 0} \xi_{j,k}(\beta) = \infty \quad , \quad \inf_{\beta>0} \xi_{j,k}(\beta) = \lim_{\beta \to \infty} \xi_{j,k}(\beta) = \alpha_k n_x.$$

Then the conclusion are verified. □

Figures 2.4 and 2.5 illustrate Proposition 9 in which (2.15) is highlighted. Lemma 4 is technically needed for Proposition 9, while the later provides the bounds of $\phi_k(\beta)$ together with its infimum value Φ_k^* in two accessible cases.

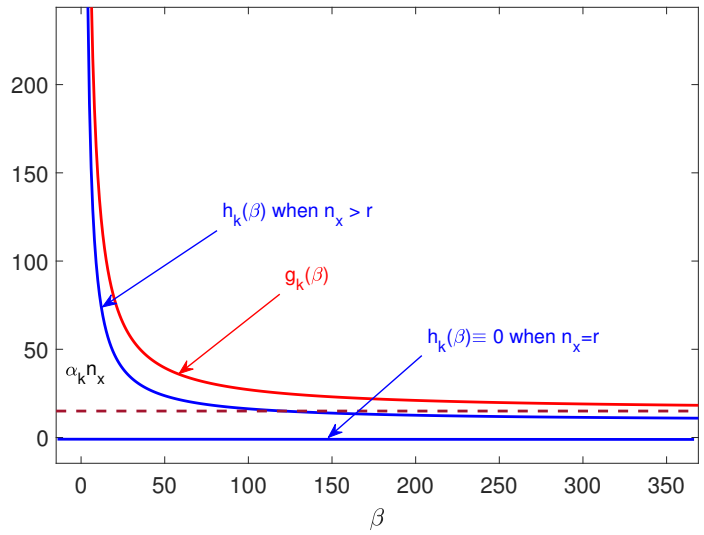

Figure 2.4 – The smallest bound $h(\beta)$ and greatest bound $g(\beta)$ of $\phi_k(\beta) = \mathrm{Tr}\{\overline{\varphi}_k(\overline{K}^*_{k,\beta}, \beta)\}$ at a fixed time $k \geq 1$.

Guaranteed conditions

It is difficult to get the exact behavior of $\phi_k(\beta)$, unless its bounds and limits, in particular, conditioning:

$$\mathbf{C1}: \quad \begin{cases} n_x = r \\ \lambda_i \geq n_0 d_{\max} + \gamma/\alpha_k, \quad \forall i = 1, ..., r \end{cases}$$

It is interesting to know $\Phi_k^* = \inf_{\beta>0} \phi_k(\beta)$ but we are just able to determined from Proposition 9 that in conditions C1,

$$0 \leq |\lim_{\beta \to 0} \phi_k(\beta) - \Phi_k^*| \leq \alpha_k n_0 (d_{\max} - d_{\min}) \mathrm{Tr}\{(MM^T)^+\},$$

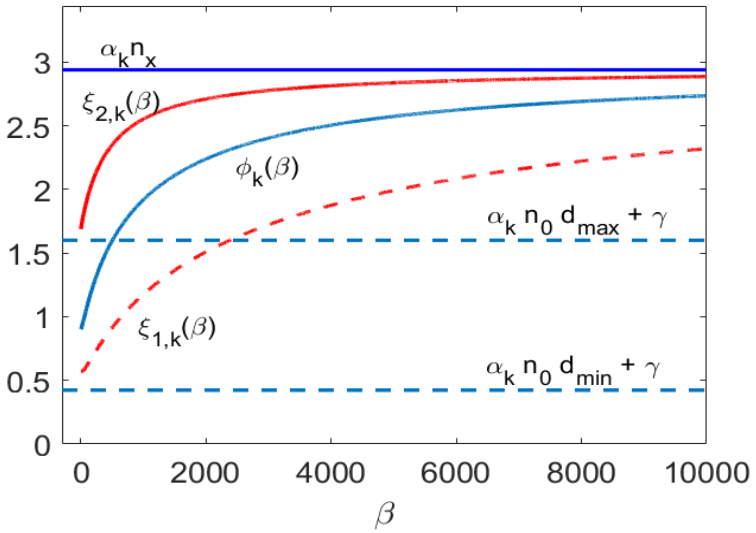

Figure 2.5 – An example of $\phi_k(\beta) = \mathrm{Tr}\{\overline{\varphi}_k(\overline{K}^*_{k,\beta}, \beta)\}$ for the case $n_x = r$ and $\lambda_i \geq n_0 d_{\max} + \gamma/\alpha_k$, $\forall i \in \{1, ..., r\}$, at a fixed time $k \geq 1$.

in which $d_{\max}, d_{\min}$ are controllable. So, the minimization of $\phi_k(\beta)$ (exactly or approximately) consists in how to design the system under consideration to reach conditions **C1** and control $d_{\max}, d_{\min}$ in an appropriate way. For instance, when $d_{\max} = d_{\min} = d$ for some $d > 0$, i.e. $\Sigma = dI_{n_y}$, then Φ^*_k is determined.

Design of conditions C1

Condition 1: $n_x = r$.

Since $M \in \mathbb{R}^{n_y \times n_x}$, $r \leq \min\{n_x, n_y\}$, to get $r = n_x$, it requires two things: $n_x \leq n_y$ and M has n_x linearly independent columns. Note that the number of output measurements n_y is not necessary the number of physical sensors n_s. Output measurements are basically designed regarding the application system and are functions of states: $y_k(i) = f_i(x_k(1), ..., x_k(n_x))$, $i = 1, ..., n_y$. In view of the system design, when $n_y < n_x$ we can take more $y_k(j) = f_j(x_k(1), ..., x_k(n_x))$, $n_y < j \leq n_x$ for appropriate f_j with notice that a function (e.g. a combination) of $y_k(1), ..., y_k(n_y)$ is also a function of $x_k(1), ..., x_k(n_x)$. Thus, the matrix $[C]$ and hence $M = \mathrm{mid}([C])$ can be obtained with $n_x \leq n_y$ before any $y_k(i)$ is measured by sensor. The missing output measurements $y_k(j)$ could be estimated by several ways, e.g. by an observer, which are considered as virtual sensors. In that case, an implicit *observability* assumption is required and the *robust sensitivity* must be taken into account. This necessitates further research in the future for a systematic implementation. The second requirement is simple to regularize numerically in particular for the interval context. A regularization $[C] \leftarrow [C] + [\underline{\epsilon}, \overline{\epsilon}]$ is

suitable so that $M = \text{mid}([C])$ has n_x linearly independent columns.

Condition 2: $\lambda_i \geq n_0 d_{\max} + \gamma/\alpha_k, \quad \forall i \in \{1, ..., r\}$.

This condition is equivalent to $\lambda_{\min} \geq n_0 d_{\max} + \gamma/\alpha_k$, where $\lambda_{\min} = \min\{\lambda_i, i \in \{1, ..., r\}\}$. This condition is achievable thanks to the Lemma 5 below.

Lemma 5. *If for some $s \in (0, 1)$, the following two expressions hold*

$$\circ \quad 0 \quad \leq \quad \max\left\{ rad([c_{ij}]),\ i \in \{1, ..., n_y\},\ j \in \{1, ..., n_x\} \right\} \quad \leq \quad \sqrt{s\,\frac{\lambda_{\min}}{n_0 n_x}},$$

$$\circ \quad \alpha_k \quad \geq \quad \max\left\{ \frac{\gamma}{(1-s)\lambda_{\min}}\ ,\ \sup\{\lambda_{\max}(P), P \in S_+([P_{k|k-1}])\} \right\}$$

then the condition $\lambda_{\min} \geq n_0 d_{\max} + \gamma/\alpha_k$ is verified.

Proof. From assumptions of the lemma we get

$$d_{\max} \leq n_x \left(s\,\frac{\lambda_{\min}}{n_0 n_x} \right) \quad \text{and} \quad \frac{1}{\alpha_k} \leq \frac{(1-s)\lambda_{\min}}{\gamma}\ ,$$

which imply that $n_0 d_{\max} + \frac{\gamma}{\alpha_k} \leq \lambda_{\min}$. □

For a more precise context, in Lemma 6, denote $\mathbf{0}_{p \times q}$ as a $p \times q$ zeros matrix. This lemma is needed for Theorem 8 computations. Only its third statement requires the first condition of **C1**.

Lemma 6. *The following statements hold:*

a) $\overline{K}^* \triangleq M^+ = M^T(MM^T)^+ = (M^TM)^+M^T \in \mathbb{R}^{n_x \times n_y}$.

b) $\lim_{\beta \to 0} \overline{K}^*_{k,\beta} = \overline{K}^*$ *and* $\lim_{\beta \to 0} \varphi_k(\overline{K}^*_{k,\beta}) = \varphi_k(\overline{K}^*)$.

c) If rank$(M) = n_x$, i.e. M has full column rank, then $I_{n_x} - \overline{K}^* M = \mathbf{0}_{n_x \times n_x}$.

Proof. a) The first expression is just a denotation for $\overline{K}^*$ with the two equalities of M^+ from Proposition 3.2 of (Barata and Hussein, 2012).

b) Since $A^+ = A^{-1}$ when the later exists and applying the Tikhonov's regularization from Theorem 4.3 of (Barata and Hussein, 2012), we get

$$\begin{aligned}
\lim_{\beta \to 0} \overline{K}^*_{k,\beta} &= \lim_{\beta \to 0} M^T \left(MM^T + \beta\Sigma + \frac{\gamma\beta}{\alpha_k(\beta + n_0)} I \right)^+ \\
&= \lim_{\beta \to 0} \lim_{\eta \to 0} M^T \left(MM^T + \beta\Sigma + \frac{\gamma\beta}{\alpha_k(\beta + n_0)} I \right)^T \times \\
&\qquad \times \left[\left(MM^T + \beta\Sigma + \frac{\gamma\beta}{\alpha_k(\beta + n_0)} I \right)^2 + \eta I \right]^{-1} \\
&= \lim_{\eta \to 0} M^T \left(MM^T \right) \left[\left(MM^T \right)^2 + \eta I \right]^{-1} \\
&= M^T(MM^T)^+ = \overline{K}^*.
\end{aligned}$$

Since $\lim_{\beta\to 0}\overline{K}^*_{k,\beta}=\overline{K}^*$ and noting that

$$\begin{aligned}\varphi_k(\overline{K}^*_{k,\beta})-\varphi_k(\overline{K}^*) &= (\overline{K}^*_{k,\beta}-\overline{K}^*)(S_k\overline{K}^{*T}-C_kP_{k|k-1})\\ &+ (\overline{K}^*_{k,\beta}S_k-P_{k|k-1}C_k^T)(\overline{K}^*_{k,\beta}-\overline{K}^*)^T,\end{aligned}$$

then $\lim_{\beta\to 0}\varphi_k(\overline{K}^*_{k,\beta})=\varphi_k(\overline{K}^*)$.

c) By definition of Moore-Penrose pseudoinverse, we get $MM^+M=M$ and hence $M(I_{n_x}-M^+M)=\mathbf{0}_{n_y\times n_x}$.
Let $X=I_{n_x}-M^+M=[X_1...X_{n_x}]\in\mathbb{R}^{n_x\times n_x}$ where X_i is i-th columns of X, $i\in\{1,...,n_x\}$. So

$$\begin{aligned}MX=\mathbf{0}_{n_y\times n_x} &\Leftrightarrow MX_i=\mathbf{0}_{n_y\times 1}\ , &\forall i\in\{1,...,n_x\}\\ &\Leftrightarrow X_i\in\mathcal{N}(M)\ , &\forall i\in\{1,...,n_x\}\end{aligned}$$

where $\mathcal{N}(M)$ is the null space of M.
Using the assumption $\text{rank}(M)=n_x$ then $\mathcal{N}(M)=\{\mathbf{0}_{n_x\times 1}\}$.
It follows that $X_i=\mathbf{0}_{n_x\times 1},\forall i\in\{1,...,n_x\}$ and hence

$$X=I_{n_x}-M^+M=I_{n_x}-\overline{K}^*M=\mathbf{0}_{n_x\times n_x}.$$

□

Theorem 8. *Assume that rank$(M)=n_x$ and assumptions of Lemma 5 are verified. Let* $\delta=\max\Big\{rad([c_{ij}]),\ i\in\{1,...,n_y\},\ j\in\{1,...,n_x\}\Big\}$. *Then*

$$\varphi_k(\overline{K}^*)\preceq\lim_{\beta\to 0}\overline{\varphi}_k(\overline{K}^*_{k,\beta},\beta)\preceq\lim_{\beta\to 0}\overline{\varphi}_k(\overline{K}^*,\beta)\preceq(\alpha_k n_0\delta^2 n_x+\gamma)(M^TM)^+\preceq\alpha_k I_{n_x} \tag{2.16}$$

where $\varphi_k(\overline{K}^*)$ is the estimation error covariance associated with the use of $K_k=\overline{K}^*$ and will be denoted by $P^{\overline{K}^*}_{k|k}$.

Assume further that: $S_+([Q])\preceq\lambda_1 I_{n_x}$, $S_+([A][A]^T)\preceq\lambda_2 I_{n_x}$, $S_+([P_{0|0}])\preceq\overline{\alpha}_0 I_{n_x}$, $\frac{\lambda_2 n_0 n_x\delta^2}{\lambda_{\min}}\leq L<1$, and let

$$\Psi_L=\frac{\lambda_1 n_0 n_x\delta^2+\gamma}{\lambda_{\min}(1-L)}\quad,\quad\Psi_L(k)=\Psi_L+(\overline{\alpha}_0-\Psi_L)L^k,$$

then

$$P^{\overline{K}^*}_{k|k}\preceq\Psi_L(k)I_{n_x}\ ,\qquad MSE^{\overline{K}^*}=\text{Tr}\{P^{\overline{K}^*}_{k|k}\}\leq\Psi_L(k)n_x\ , \tag{2.17}$$

$$P^{\overline{K}^*}_{k+1|k}\preceq(\lambda_2\Psi_L(k)+\lambda_1)\,I_{n_x}. \tag{2.18}$$

Remark 9. Being a function of L, Ψ_L is non decreasing on $(0,1)$ and $\gamma/\lambda_{\min} = \lim_{L\to 0} \Psi_L < \Psi_L < \lim_{L\to 1} \Psi_L = \infty$. The fact $L \to 0$ implies that $\delta \to 0$, equivalently, $\text{rad}([C]) = 0$ or $[C]$ reduces to the point matrix $\text{mid}([C])$, and then the limit $\gamma/\lambda_{\min} I_{n_x}$ is the minimum upper bound of $P_{k|k}^{\overline{K}^*}$ according to the change of $\text{rad}([C])$. If furthermore, $L \leq \lambda_2 < 1$, then $\Psi_L \leq \lim_{L\to\lambda_2} = \frac{\lambda_1 n_0 n_x \delta^2 + \gamma}{\lambda_{\min}(1-\lambda_2)}$ which is a finite value. The condition $\lambda_2 < 1$ relates to a requirement for the stability of the considered system.

Remark 10. The assumption $\frac{\lambda_2 n_0 n_x \delta^2}{\lambda_{\min}} \leq L < 1$ is equivalent to $\delta^2 \leq \frac{L}{\lambda_2}\frac{\lambda_{\min}}{n_0 n_x}$. Together with the assumption $\delta^2 \leq s\frac{\lambda_{\min}}{n_0 n_x}$ for some $s \in (0,1)$, we can get different choices to achieve these assumptions. First, we can think that $\lambda_2 = \sup\{\lambda_{\max}(A_k A_k^T), \forall A_k \in [A]\}$. If L and λ_2 are given and satisfy $0 < \frac{L}{\lambda_2} < 1$ then we may choose $s = \frac{L}{\lambda_2}$ and $\text{rad}([C])$ is controlled by $\delta \leq \sqrt{s\frac{\lambda_{\min}}{n_0 n_x}}$. For instance, in many applications, a reasonable value for δ is $5\% - 10\%$. Another possible setting is that we choose $L, s \in (0,1)$ so that $\lambda_2 \leq \frac{L}{s}$ then the assumption $\frac{\lambda_2 n_0 n_x \delta^2}{\lambda_{\min}} \leq L$ holds. The smaller λ_2 is, the greater s can be chosen in $(0,1)$ and hence the more uncertainty of $[C]$ can be covered via its radii. Besides, L can be seen as convergence rate of $\Psi_L(k)$ to Ψ_L as k tends to ∞.

Remark 11. Since $\Psi_L(k) = \Psi_L + (\overline{\alpha}_0 - \Psi_L)L^k$, then $\Psi_L(k) \downarrow \Psi_L$ if $\overline{\alpha}_0 \geq \Psi_L$ and $\Psi_L(k) \uparrow \Psi_L$ if $\overline{\alpha}_0 < \Psi_L$. In the later case, $P_{k|k}^{\overline{K}^*} \preceq \Psi_L I_{n_x}$, $\forall k \geq 1$. Furthermore, Ψ_L can be precomputed and controlled before the algorithm starts. For instance, it can be controlled the choice of L, s, δ so that $\Psi_L \leq \Psi$ with a given constant $\Psi > 0$. Concretely, the constraint $\frac{L}{\lambda_2} \geq s \geq \frac{n_0 n_x \delta^2}{\lambda_{\min}}$ can be reduce to $s = L/\lambda_2 = n_0 n_x \delta^2/\lambda_{\min}$ which implies $\Psi_L = \frac{\lambda_1 L}{\lambda_2(1-L)} + \frac{\gamma}{\lambda_{\min}(1-L)}$. Let $\Psi_L = \Psi$ and get

$$L = \frac{\Psi - \gamma/\lambda_{\min}}{\Psi + \lambda_1/\lambda_2}\ , \quad s = L/\lambda_2\ , \quad \delta^2 = s\lambda_{\min}/(n_0 n_x),$$

provided that $\gamma/\lambda_{\min} < \Psi$.

Proof. By assumptions of the theorem, the conditions **C1** holds. Then, it follows from Lemma 6, Proposition 9 and Theorems 6-7 that :

$$0 \preceq \varphi_k(K_k^*) \preceq \lim_{\beta\to 0} \varphi_k(\overline{K}_{k,\beta}^*) = \varphi_k(\overline{K}^*) \preceq \lim_{\beta\to 0} \overline{\varphi}_k(\overline{K}_{k,\beta}^*, \beta) \preceq \lim_{\beta\to 0} \overline{\varphi}_k(\overline{K}^*, \beta),$$

and

$$\lim_{\beta\to 0} \overline{\varphi}_k(\overline{K}^*, \beta) = \overline{K}^* \left[\alpha_k n_0 \Sigma + \gamma I_{n_y}\right] \overline{K}^{*T} \preceq (\alpha_k n_0 \delta^2 n_x + \gamma)(M^T M)^+,$$

in which $\Sigma \preceq \delta^2 n_x I_{n_y}$ and $\overline{K}^* \overline{K}^{*T} = (M^T M)^+$.

Since MM^T and $M^T M$ have the common non null eigenvalues then they have the same $\lambda_{\min}$ (non null). So we get $\overline{K}^* \overline{K}^{*T} = (M^T M)^+ \preceq \frac{1}{\lambda_{\min}} I_{n_x}$ and hence

$$\lim_{\beta \to 0} \overline{\varphi}_k(\overline{K}^*, \beta) \preceq \left(\alpha_k n_0 \delta^2 n_x + \gamma\right) \frac{1}{\lambda_{\min}} I_{n_x} \preceq \alpha_k I_{n_x},$$

where the last inequality holds thanks to $\lambda_{\min} \geq n_0 \delta^2 n_x + \frac{\gamma}{\alpha_k}$.

By recursion, one gets

$$\begin{aligned} P_{k|k}^{\overline{K}^*} &= (\tilde{A}_{k,1}^{\otimes}) P_{0|0} (\tilde{A}_{k,1}^{\otimes})^T + \sum_{i=1}^{k} (\tilde{A}_{k,i+1}^{\otimes} \tilde{C}_i) Q_i (\tilde{A}_{k,i+1}^{\otimes} \tilde{C}_i)^T \\ &+ \sum_{i=1}^{k} (\tilde{A}_{k,i+1}^{\otimes} \overline{K}^*) R_i (\tilde{A}_{k,i+1}^{\otimes} \overline{K}^*)^T, \end{aligned} \tag{2.19}$$

where $\tilde{A}_{k,s}^{\otimes} = \tilde{A}_k \tilde{A}_{k-1} ... \tilde{A}_{s+1} \tilde{A}_s$ if $s \leq k$ and $\tilde{A}_{k,s}^{\otimes} = I$ if $s > k$, $\tilde{C}_k = I - \overline{K}^* C_k$, $\tilde{A}_k = \tilde{C}_k A_k$.

For any $p \geq 1$, $C_p \in [C]$ is decomposed as $C_p = M + \Delta_p$, $\Delta_p = \sum_{i,j} \alpha_{ij}(p) R_{ij}$, $\alpha_{ij}(p) \in [-1, 1]$ and hence, using Lemma 6 and (2.4), one gets

$$\begin{aligned} \left(I - \overline{K}^* C_p\right) \left(I - \overline{K}^* C_p\right)^T &= \overline{K}^* \Delta_p \Delta_p^T \overline{K}^{*T} \\ &\preceq \overline{K}^* (n_0 \Sigma) \overline{K}^{*T} \\ &\preceq n_0 \delta^2 n_x \overline{K}^* \overline{K}^{*T} \\ &\preceq n_0 \delta^2 n_x \frac{1}{\lambda_{\min}} I_{n_x}, \end{aligned}$$

implying that $\tilde{A}_p \tilde{A}_p^T \preceq \lambda_2 n_0 \delta^2 n_x \frac{1}{\lambda_{\min}} I_{n_x}$.

Substituting these results into (2.19), it follows that

$$P_{k|k}^{\overline{K}^*} \preceq \overline{\alpha}_0 \left(\frac{\lambda_2 n_0 n_x \delta^2}{\lambda_{\min}}\right)^k I_{n_x} + \frac{\lambda_1 n_0 n_x \delta^2 + \gamma}{\lambda_{\min}} \sum_{i=1}^{k} \left(\frac{\lambda_2 n_0 n_x \delta^2}{\lambda_{\min}}\right)^{k-i} I_{n_x}$$

and the conclusion holds noting that $\frac{\lambda_2 n_0 n_x \delta^2}{\lambda_{\min}} \leq L < 1$ and $\sum_{i=0}^{k-1} L^i = \frac{1-L^k}{1-L}$. In addition, $\lim_{k \to \infty} \Psi_L(k) = \Psi_L$. □

2.3.4 OUBIKF Algorithm

Applying Theorem 8 to the OUBIKF Beta version (Algorithm 2) with the choice of $\sigma = 1$, $\beta \to \infty$ and under conditions **C1**, theoretically, one obtains

$$\begin{aligned}
[\hat{x}_{k|k}] &= \lim_{\beta\to 0}\{(I_{n_x} - \overline{K}^*_{k,\beta}[C])[\hat{x}_{k|k-1}] + \overline{K}^*_{k,\beta}(y_k - [D]u_k)\},\\
&= (I_{n_x} - \overline{K}^*[C])[\hat{x}_{k|k-1}] + \overline{K}^*(y_k - [D]u_k),\\
&= \overline{K}^*\Big([-1,1]\cdot \mathrm{rad}([C])\Big)[\hat{x}_{k|k-1}] + \overline{K}^*(y_k - [D]u_k),\\
\mathcal{P}_{k|k} &\triangleq \lim_{\beta\to 0}\overline{\varphi}_k(\overline{K}^*_{k,\beta},\beta) = \lim_{\beta\to 0}(I - \overline{K}^*_{k,\beta}M)\alpha_k\,(1+n_0/\beta)\\
&= \lim_{\beta\to 0}(I - \overline{K}^*_{k,\beta}M)\alpha_k n_0/\beta,\\
\mathrm{Tr}\{\mathcal{P}_{k|k}\} &= \lim_{\beta\to 0}\phi_k(\beta) \approx \Phi^*_k,
\end{aligned}$$

and, numerically, for small $0 < \beta \ll$, one obtains:

$$\begin{aligned}
\mathcal{P}_{k|k} &\overset{\beta\ll}{\approx} (I - \overline{K}^*_{k,\beta}M)\alpha_k n_0/\beta \;\approx\; \overline{K}^*\left[\alpha_k n_0\Sigma + \gamma I_{n_y}\right]\overline{K}^{*T},\\
\mathrm{Tr}\{\mathcal{P}_{k|k}\} &\overset{\beta\ll}{\approx} \phi_k(\beta) \;\approx\; \mathrm{Tr}\{\overline{K}^*\left[\alpha_k n_0\Sigma + \gamma I_{n_y}\right]\overline{K}^{*T}\} \approx\; \Phi^*_k.
\end{aligned}$$

Above results constitute the optimal version of the OUBIKF Algorithm which is simply named as OUBIKF (Algorithm 3) in the following.

Remark 12. The corresponding confidence intervals are determined by

$$\mathrm{CI}^i_k = \left[\inf([\hat{x}^i_{k|k}]) - h\sqrt{\mathcal{P}^{ii}_{k|k}}, \sup([\hat{x}^i_{k|k}])] + h\sqrt{\mathcal{P}^{ii}_{k|k}}\right],$$

for $i = 1, ..., n_x$ and $h = 1, 2, 3$, which contain the states x_k with probabilities at least $68\%, 95\%, 99.7\%$ according to h.

2.4 Application

In this section, the OUBIKF Algorithm is applied in simulation to a model taken from automotive domain (Fergani, 2014). This model is a nonlinear continuous-time model which has been discretized/linearized and thus given under the form (1.1).

Algorithm 3 OUBIKF

Initialization:
  $[\hat{x}_{0|0}]$, $\mathcal{P}_{0|0}$, $[A]$, $[B]$, $[C]$, $[D]$, $[Q]$, $[R]$, s, $\lambda_{\min}$, u_k, y_k, $k = 1, 2, 3, ..., N$
  Find n_0 the number of non zero radius of $[C]$
  Find γ such that $S_+([R]) \preceq \gamma I$ using Theorem 5
  $\overline{K}^* = \text{mid}([C])^+$;
  $\Sigma \ = \text{Diag}\left\{\text{rad}([C])\text{rad}([C])^T\right\}$;
for $k = 1, 2, 3, ...N$ **do**
  Prediction step:
    $[\hat{x}_{k|k-1}] = [A][\hat{x}_{k-1|k-1}] + [B]u_k$
    $[P_{k|k-1}] = [A]\mathcal{P}_{k-1|k-1}[A]^T + [Q]$
    Find α_k such that $S_+([P_{k|k-1}]) \preceq \alpha_k I$ using Theorem 5
    $\alpha_k = \max\left\{\gamma/[(1-s)\lambda_{\min}]\ ,\ \alpha_k\right\}$
  Correction step:
    $[\hat{x}_{k|k}] = \overline{K}^*\Big([-1, 1] \cdot \text{rad}([C])\Big)[\hat{x}_{k|k-1}] + \overline{K}^*(y_k - [D]u_k)$
    $\mathcal{P}_{k|k} = \overline{K}^*\left[\alpha_k n_0 \Sigma + \gamma I_{n_y}\right]\overline{K}^{*T}$
end for
$^{(*)}$ s and $\lambda_{\min}$ satisfy conditions **C1**.

2.4.1 Bicycle vehicle model

The model parameters

The vehicle model parameters obtained by an identification process on the Renault Mégane Coupé are presented. Throughout the paper, indexes $i = \{f, r\}$ and $j = \{l, r\}$ are used to identify vehicle front, rear and left, right positions, respectively. The full vehicle model with all the nonlinear equations describing its dynamical behaviour can be found in (Fergani, 2014).

The linear bicycle model

Since the full model is highly non linear, a linear bicycle model as illustrated by Fig. 2.6 reproducing the lateral behaviour of the car is used for this study by linearizing the former. Reference to Fig. 2.6, $\beta(t)$ is the sideslip angle and $\psi(t)$ is the vehicle yaw which form the model state variables. $F_{ty_f}(t)$ represents lateral front tire forces, $F_{ty_r}(t)$ represents lateral rear tire forces and $F_{tx_f}(t)$ represents the longitudinal front tire forces, v is the vehicle speed, $\Delta F_{tx_r(t)}$ is the differential rear braking force (obtained based on the braking torques $T_{b_{rj}}$), δ is the steering angle and M_{dz} is the yaw moment disturbance.

Symbol	Value	Unit	Signification
m	1535	kg	vehicle mass
I_z	2149	kg.m^2	vehicle yaw inertia
C_f	20000	N/degree	lateral tire front stiffness
C_r	20000	N/degree	lateral tire rear stiffness
S_r	12720	N	longitudinal tire rear stiffness
l_f	1.4	m	distance COG - front axle
l_r	1	m	distance COG - rear axle
t_r	1.4	m	rear axle length
R	0.3	m	tire radius
μ	$[2/5; 1]$	$-$	tire/road contact friction coefficient
v	$[50; 130]$	km/h	vehicle velocity coefficient

Table 2.4 – Renault Mégane Coupé parameters.

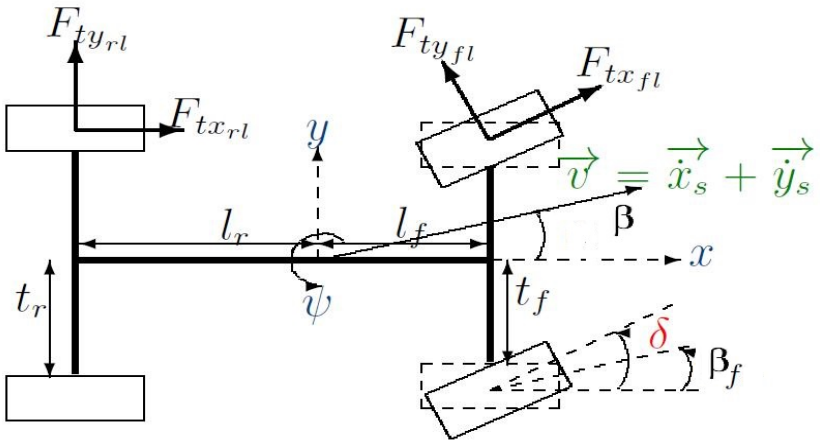

Figure 2.6 – View of the bicycle model reproducing the lateral behaviour of the car.

The model is obtained considering the following:

a) Low sideslip angles: $|\beta| < 7$ degrees,

b) Low longitudinal slip ratio: < 0.1,

c) Low steering angles: $\cos(\delta) \simeq 1$.

The linearized lateral tire forces are:

$$F_{ty_f}(t) = C_f \beta_f(t), \quad F_{ty_r}(t) = C_r \beta_r(t), \tag{2.20}$$

with β_f(t) and β_r(t) denoting the front and rear sideslip angles,

$$\beta_f(t) = \delta(t) - \beta(t) - \frac{l_f \dot{\psi}(t)}{v}, \quad \beta_r(t) = \beta(t) + \frac{l_f \dot{\psi}(t)}{v}. \tag{2.21}$$

This leads to the following state space representation (2.22):

$$\begin{bmatrix} \dot{\beta}(t) \\ \ddot{\psi}(t) \end{bmatrix} = \begin{bmatrix} \frac{-C_f - C_r}{mv} & 1 + \mu\frac{-l_r C_r - l_f C_f}{mv^2} \\ \frac{-l_r C_r - l_f C_f}{I_z} & \frac{-l_f^2 C_f - l_r^2 C_r}{I_z v} \end{bmatrix} \begin{bmatrix} \beta(t) \\ \dot{\psi}(t) \end{bmatrix} + \begin{bmatrix} \frac{C_f}{mv} & 0 & 0 & 0 \\ \frac{l_f C_f}{I_z} & \frac{1}{I_z} & \frac{S_r R t_r}{2I_z} & -\frac{S_r R t_r}{2I_z} \end{bmatrix} \begin{bmatrix} \delta \\ M_{dz} \\ T_{b_{rl}} \\ T_{b_{rr}} \end{bmatrix} . \quad (2.22)$$

Remark 13. It is worth noting that the sideslip dynamics are highly nonlinear and cannot be measured via a conventional sensor.

Remark 14. $\mu \in [0; 1]$ is the tire/road adhesion coefficient. Its value depends on the road conditions (dry, wet, icy,...) and highly influences the lateral dynamics of the vehicle.

2.4.2 Simulation

A discretization phase with a sampling time $T = 0.05s$ is applied to the considered continuous model to get matrices A_d, B_d, C_d, D_d (non interval and independent of time instant k) according to equations in (1.1). Then, interval matrices $[A]$, $[B]$, $[D]$ are generated as follow: for $F \in \{A_d, B_d, D_d\}$, let $F = \text{mid}([F])$ and choose the radii $\text{rad}([F])$ at random in $[0\,, \max\_\text{rad}]$ with $\max\_\text{rad} = 0.5$. The covariance matrices $[Q]$ and $[R]$ are generated in the same way, their diagonal elements being intervals of positive real numbers.

Choose $M = \text{mid}([C]) = C_d$. With this choice, $\text{rank}(M) = n_x$, so the first part of conditions **C1** is satisfied. The second part of conditions **C1** is reached using Remark 11 to compute L, s, δ with the choices $\Psi = 10\gamma/\lambda_{\min}$ and $n_0 = n_x n_y$. Then $[C]$ is generated in the same way of $[A]$ where $\max\_\text{rad} = \delta$.

Ψ_L	$\lambda_{\min}$	λ_1	λ_2	γ	s	L	δ
3.84	3.49	0.91	2.30	1.34	0.24	0.82	0.23

Table 2.5 – Parameter computation results.

Inputs u_k are simulated according to a dynamic change for $N = 864$ time instances (Fig. 2.7), that is the vehicle is assumed to be driven at 15m/s (54 km/h) on a dry road ($\mu = 1$) and a double line change maneuver is performed from $t = 0.5$s to $t = 1.5$s by the driver. The initial state is chosen at $x_0 = (0, 0)^T$. At each time instant k, generate A_k, B_k, C_k, D_k, Q_k, R_k according to uniform distribution in corresponding interval matrices and so that $Q_k \in S_+(n_x)$ and $R_k \in S_+(n_y)$. Then w_k and v_k are simulated.

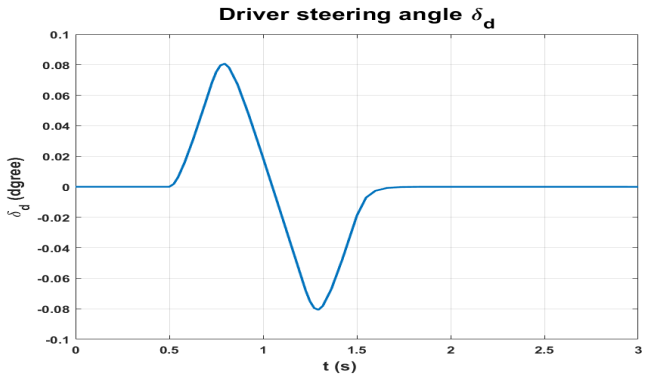

Figure 2.7 – Bicycle model - Input u_k simulation

Finally, $\{x_k, y_k\}_{k\in 1:N}$ are computed according to system (1.1) where $x_k = (\beta(k), \dot{\psi}(k))^T \equiv (x_1(k), x_2(k))^T$.

For state estimation, Algorithm 3 is used to obtain $[\hat{x}_{k|k}]$ and corresponding confidence intervals CI_k. The Algorithm is initialized at $[\hat{x}_0] = ([-0.5, 0.5], [-0.5, 0.5])^T$ and $\mathcal{P}_{0|0} = \max\{\mathrm{Diag}(\sup([Q]))\}I$.

The 95% confidence intervals CI_k contain all real states x_k as shown in Figure 2.8. The computation time using the OUBIKF with the new setting of the present work is improved against the OUBIKF Beta version with the setting proposed in (Lu et al., 2019) (Table 2.6), while the last one has been shown by simulation to be more efficient in computation time against its precursor (Tran et al., 2017).

	OUBIKF	OUBIKF Beta version
Computation time (s)	2.33	3.02

Table 2.6 – Computation times of OUBIKF and OUBIKF Beta version with two settings for $N = 864$ iterations.

Remark 15. It is worth to note that Algorithm 3 is not applicable for Example 2 since the widths of the given matrices are too large so that $\frac{\lambda_2 n_0 n_x \delta^2}{\lambda_{\min}} > 1$ and thus there is no suitable L can be chosen.

2.5 Conclusion and perspective

The OUBIKF Algorithm proposed in (Lu et al., 2019) (Beta version) is enhanced theoretically and practically by the two stages optimization and the guaranteed conditions **C1**.

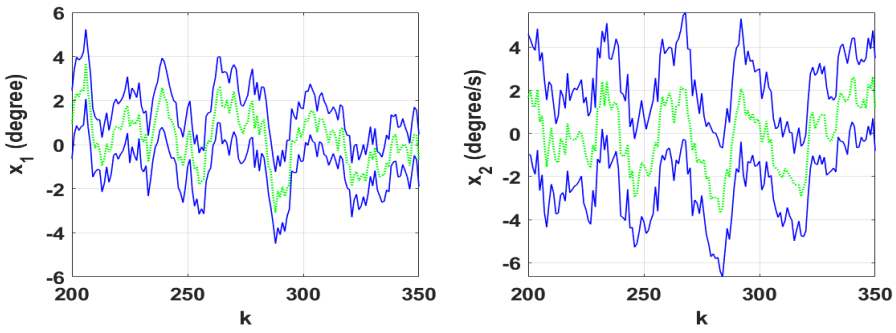

Figure 2.8 – Estimation results. For $i = 1, 2$, *the center green line*: real states x_k^i, *the solid blue lines*: 95% confidence intervals CI_k^i.

Under these conditions, the optimal upper bound is reached approximately by the chosen $\mathcal{P}_{k|k}$ whose trace is highly close to the optimal value Φ_k^*. Also, the considered Algorithm is ensured to perform with stability in the sense that the trace of $\mathcal{P}_{k|k}$ is non-asymptotically and asymptotically bounded and can be controlled, implying that there is no width explosion of the resulted estimators. In addition, the trace of $\mathcal{P}_{k|k}$ is smaller than the one of the upper bound $\alpha_k I_{n_x}$ of $S_+([P_{k|k-1}])$ in the prediction step.

Thanks to deep analysis in limit results, expressions of the correction step are simplified and many factors of them can be computed off-line. It reduces the algorithm computation time in comparison Algorithm 3 with others used in (Chen et al., 1997), (Xiong et al., 2013), (Tran et al., 2017), (Lu et al., 2019) depending on the complexity of the gain expressions and of the corresponding method of finding the gain.

The present work concerns however interval Kalman filtering in which field a number of issues are not investigated systematically, especially those related to filter convergence, system/filter stability, controller/observer design using interval estimates. For instance, using the OUBIKF, the interval filter C-stable notion (Definition 6) might have connections to the filter convergence (which notion must be properly defined). In another view, investigate the robust control aspect of the OUBIKF is also an interesting research.

Chapter 3

Reinforced Likelihood Box Particle Filter

3.1 Introduction

In *State Estimation* or *Filtering* problems, when dealing with a linear Gaussian state-space model, analytical expressions computing the state estimates according to posterior distributions can be derived by the well known and widespread Standard Kalman Filter (SKF) (Kalman, 1960). Many extension of SKF are then provided by various researches in different contexts (Mohamed and Nahavandi, 2012b; Combastel, 2015; Chen et al., 1997; Lu et al., 2019). For nonlinear model without Gaussian measurement assumption, *Particle Filters* (PF) have been applied successfully to a variety of state estimation problems (Gordon et al., 1993; Doucet et al., 2001). The PF efficiency and accuracy depend mostly on the number of particles used in the estimation which may require a large computation time.

One of the famous extensions of PF to set membership approach is the *Box Particle Filter* (BPF) (Abdallah et al., 2008). BPF handles box (interval vector of) states and bounded errors by using *interval analysis* and constraint satisfaction techniques. This method has been shown to control quite efficiently the number of required particles, hence reducing the computational cost and providing good results in several experiments.

Since then, numerous variants of BPF have been developed (Nassreddine et al., 2010; Blesa et al., 2015; Tran et al., 2018) to deal with measurement bounded uncertainty, measurement stochastic uncertainty or measurement mixed uncertainty. Various techniques and theories are proposed to address the diversity of requirements in these contexts, e.g. weight updating using Bayesian filtering technique extending to box particle case (Blesa et al., 2015)

or belief function theory with different methods (Nassreddine et al., 2010; Tran et al., 2018).

In the present work, regarding the large variety of BPF, a scheme is proposed to give a generalized description that highlights the specificity of this class of filters. An analysis of the likelihood computation (the crucial step in the scheme) methodology is investigated, thanks to which a novel filter, namely *Reinforced Likelihood Box Particle Filter* (RLBPF), is produced. This filter benefits the advantages from various existing BPFs via the use of a number of reinforcement techniques (score function, reduction percentage, exponential weighting, backward estimate,...) to enhance the estimation performance. An overview on BPFs and discussions about from assumptions used in the literature to the filters performance evaluation approach are presented. Also, an academic illustration example and an application to the suspension (quarter vehicle) model are provided to highlight the efficiency of the proposed estimation strategy.

The chapter is organized as follows. The problem formulation is presented in Section 3.2 with discussions about assumptions used in the literature. Section 3.3 presents the general scheme of BPF and the likelihood computation methodology. Section 3.4 deals with the main disadvantage of *Likelihood Computation Methods* (LCMs) and provides necessary requirements of a novel BPF method. Section 3.5 presents the RLBPF method with its essential algorithm version, a filter performance evaluation approach and an academic illustration example. Section 3.6 provides an application of RLBPF to the suspension model and its full algorithm version which helps to deal with more complexes models like that used in the application. Section 3.7 presents the conclusion of the chapter with discussions and perspectives.

3.2 Problem formulation

In this section, we present the assumptions used in the literature by a number of researches. The first assumption is a common assumption used by all related researches while the other three assumptions are used differently by each of them. Then, some further discussions are also provided.

Consider the following dynamical system:

$$(\Sigma): \quad \begin{cases} x_k &= f(x_{k-1}, u_k, w_k), \\ y_k &= h(x_k, u_k, v_k)\ , \end{cases} \quad k \in \mathbb{N}^*, \tag{3.1}$$

where $x_k \in \mathbb{R}^{n_x}$ and $y_k \in \mathbb{R}^{n_y}$ are respectively state and measurement output, $u_k \in \mathbb{R}^{n_u}$ input, $w_k \in \mathbb{R}^{n_w}$ state dynamic disturbance and $v_k \in \mathbb{R}^{n_y}$ measurement noise.

***Assumption (A)* : State Process Uncertainty**

u_k, w_k are unknown and belong respectively to known intervals $[u_k]$ and $[w_k]$.

***Assumption (B)* : Measurement Bounded Uncertainty**

(B1) v_k is unknown and belongs to known interval $[v_k]$.

(B2) The observed measurements are intervals $[y_k]$.

(B3) The measurements are assumed to be accurate in the sense that $[y_k] \ni h(x_k, u_k) \equiv h(x_k, u_k, 0)$ (the zero noise case), where x_k is the real state.

***Assumption (C)* : Measurement Stochastic Uncertainty**

(C1) v_k are additive noises with known density p_v.

(C2) The observed measurements are point values y_k.

***Assumption (D)* : Measurement Mixed Uncertainty**

(D1) v_k are additive Gaussian noises with unknown mean $\mu_k \in \mathbb{R}^{n_y}$ and covariance $\Sigma_k \in \mathbb{R}^{n_y \times n_y}$.

(D2) $\mu_k \in [\mu_k]$, $\Sigma_k \in [\Sigma_k]$ with known intervals $[\mu_k]$, $[\Sigma_k]$.

(D3) The observed measurements are point values y_k.

Assumption (A) is used in (Abdallah et al., 2008; Nassreddine et al., 2010; Blesa et al., 2015; Tran et al., 2018).

Assumptions (B) are under study in (Abdallah et al., 2008; Nassreddine et al., 2010). In (Abdallah et al., 2008), the BPF is introduced and becomes standard for many extensions or variants with essential steps: *initialization, propagation, contraction, likelihood (weight) computation, state estimation and resampling*. In (Nassreddine et al., 2010), the Belief State Estimation algorithm is developed using the belief function theory. It may require some techniques for the construction and computation of masses, but after being normalized, these masses become likelihoods in the probability sense. Therefore, we also call likelihood computation as an essential step of this method.

Assumptions (C) are used in (Blesa et al., 2015). The method proposed therein includes a different approach to weight the box particles as well as a resampling procedure based on repartitioning the box enclosing the propagated states. There is no contraction step in this method.

Assumptions (D) are used in (Tran et al., 2018), in which (D1) is a special case of (C1) with a slight relaxation by adding bounded uncertainties to Gaussian parameters μ_k and Σ_k. In (Tran et al., 2018), the belief function theory is used with continuous mass functions to represent these kinds of uncertainties and to compute box particle likelihoods. The proposed approach therein leads to the so-called Evidential Box Particle Filter (EBPF) including all essential steps of the standard BPF.

Remark 16. (B3) is the implicit assumption deriving the consistency between the predicted measurement boxes $[h]([x_k^i],[u_k])$, $i \in \{1,...,M\}$ (M the number of partitioned boxes), and the real measurement box $[y_k]$. This consistency is used in the contraction step and the likelihood computation by penalizing all particle boxes with which the intersections $[h]([x_k^i],[u_k]) \cap [y_k]$ are empty. □

Remark 17. Assumptions (D3) and (C2) are coincided. They can be transformed into (B2) with a slight relaxation of (B3). That is, knowing the density of v_k, we deduce its confidence intervals $[v_k]$ with some significant level α and define $[y_k] \triangleq y_k - [v_k]$. Then (B3) is relaxed in the sense that the observed measurements $[y_k]$ do not contain $h(x_k,u_k)$ with certainty but with only a high probability $(1-\alpha)$. □

3.3 General scheme of Box Particle Filter

3.3.1 Scheme

In general, although applying different background theories, the proposed methods in (Abdallah et al., 2008; Nassreddine et al., 2010; Blesa et al., 2015; Tran et al., 2018) study State Estimation in a framework of stochastic uncertainties and/or bounded uncertainties with two main objectives :

- **Objective 1:** Reduce as much as possible the width of box particles to penalize the conservatism due to interval computations.
- **Objective 2:** Quantify (compute) box particle likelihoods as well as possible to enhance the accuracy of the estimates.

The methods used in these references can be considered as variants of BPF and be summarized by Scheme 4 which is applied in a mostly similar manner across them.

Remark 18. In this scheme, for a general presentation, the observed measurements are denoted as intervals since the point values are considered as special cases of intervals.

N_{k_0} in the initialization step takes value in $\{1,...,M\}$ and is the number of box particles obtained at the end of the likelihood computation step at the previous time instant (k_0-1). For $k_0=0$, the initialization concerns only the partition of $[x_0]$ and not the resampling. Condition C in the while loop is different from method to method. □

Scheme 4 General Scheme of Box Particle Filtering

STEP 1 : Initialization.
At a time step $k_0 \geq 0$, (re)partition the interval $[x_{k_0}]$ or resample the set $\{[x^j_{k_0}], w_j\}_{j=1,...,N_{k_0}}$ into M disjoints equal-volume sub-boxes with the same weights: $\{[x^i_{k_0}], w_i = 1/M\}_{i=1,...,M}$.

while $\{[x^i_{k_0}], w_i\}_{i=1,...,M}$ still satisfies a predetermined Condition C **do**

STEP 2 : Propagation.
Get a new set of box particles $\{[x^i_{k_0+1}] = [f]([x^i_{k_0}], [u_{k_0}])\}_{i=1,...,M}$ estimating the box containing the real state $x_{k_0+1} = f(x_{k_0}, u_{k_0})$ with or without a contraction step.

STEP 3 : Likelihood computation

- Compute (and normalize) the likelihoods of box particles $\{[x^i_{k_0+1}]\}_{i=1,...,M}$ being the box containing the real state x_{k_0+1}. This computation bases on the consistency between the estimated measurement $[h]([x^i_{k_0+1}], [u_{k_0}])$'s and the obtained measurement $[y_{k_0+1}]$ using different criteria and methods.
  By this step, the following set of box particles with updated weights is obtained : $\{[x^i_{k_0+1}], w_i\}_{i=1,...,M}$.
- Some techniques can be applied at this step to get a more "efficient" set of box particles, e.g. discarding the boxes with small weights (smaller than some predetermined threshold) and with or without replicating the box associated with the greatest weight,... From this, the set of box particles becomes $\{[x^i_{k_0+1}], w_i\}_{i=1,...,N_{k_0+1}}$, $1 \leq N_{k_0+1} \leq M$.

STEP 4 : Estimation

$$\text{Interval estimate} \quad : \quad [x_{k_0+1}] = \sum_{i=1}^{M} w_i \cdot [x^i_{k_0+1}] \tag{3.2}$$

$$\text{Point estimate} \quad : \quad x_{k_0+1} = \sum_{i=1}^{M} w_i \cdot \text{mid}([x^i_{k_0+1}]) \tag{3.3}$$

STEP 5 : $k_0 = k_0 + 1$

end while

STEP 6 : Restarting at STEP 1.

Remark 19. The interval estimate obtained from (3.2) has the same midpoint values as the one computed by (3.3) and its width equal the mean of propagated boxes ($[x^i_{k_0+1}]$'s) widths, i.e.:

$$\text{mid}([x_{k_0+1}]) = \sum_{i=1}^{M} w_i.\text{mid}([x^i_{k_0+1}]) \quad , \quad \text{rad}([x_{k_0+1}]) = \sum_{i=1}^{M} w_i.\text{rad}([x^i_{k_0+1}]) \; .$$

Some methods considered above propose to use alternately a kind of confidence interval determined by:

$$CI = \text{ Point estimate (3.3) } \pm \; h\sqrt{\text{Diag}_v(\text{Covariance matrix})} \; ,$$

where $h > 0$ and the Covariance matrix is computed in several ways (Abdallah et al., 2008; Tran, 2017). Therefore, a more general interval estimate of the real state can be obtained by

$$\begin{aligned} [x_{k_0+1}] &= \sum_{i=1}^{M} w_i.\text{mid}([x^i_{k_0+1}]) \pm \mathbf{ScF}.\sum_{i=1}^{M} w_i.\text{rad}([x^i_{k_0+1}]), \qquad (3.4) \\ \mathbf{ScF} &= \text{diag}\{\alpha_1, ..., \alpha_{n_x}\} \; , \quad \alpha_i > 0, \; i = 1, ..., n_x, \end{aligned}$$

where **ScF** is called scaling factors. Indeed, with appropriate values of **ScF**, (3.2) and (3.3) can be achieved from (3.4). Furthermore, **ScF** can be fixed or time variant. □

3.3.2 Likelihood computation methodology

In the next, the diagram in Fig.3.1 is used to discuss the methodology of LCMs related to Scheme 4.

First of all, that is the assumptions of the system under consideration supply the information needed to build the likelihood. For instance, the information may be:

- **Information (a)**: The intersection between $[y_k]$ and the box $[h]([x^i_k], [u_k])$ containing the real value y_k must be non empty,
- **Information (b)**: The distribution of v_k and hence the distribution of $r_k = y_k - h(x_k, u_k)$ is Gaussian (for additive measurement noise),

  (or more other piece of information)...

The information can be directly an assumption or a deduction of the later. In bounded-error context, only **Information (a)** is treated (Abdallah et al., 2008) while in the mixed uncertainty case, both **Information (a)** and **Information (b)** are taken into account (Tran et al., 2018).

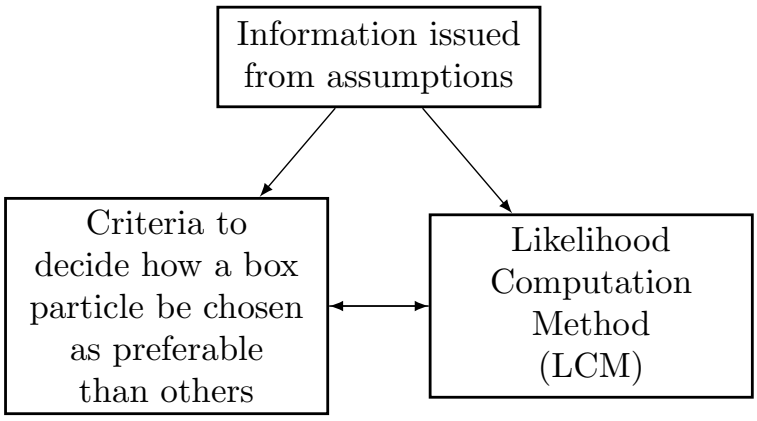

Figure 3.1 – Likelihood computation methodology schema

Criteria and methods are then chosen to exploit the information. On the one hand, once a criterion is chosen, different methods can be used to calculate the likelihood. On the other hand, a calculation method may correspond to one or many criteria. There are also calculation methods that exploit better the supplied information than others. These ideas are illustrated by Example 3.

Example 3 (**Illustration example**). To exploit **Information (a)**, following criteria can be used:

- **Criterion 1**: The particle $[x_k^i]$ giving a "bigger" intersection determined by $[\hat{z}_k^i] \triangleq [h]([x_k^i], [u_k]) \cap [y_k]$ must be preferable,
- **Criterion 2**: The particle $[x_k^i]$ making $[\hat{y}_k^i] \triangleq [h]([x_k^i], [u_k])$ "closer" to $[y_k]$ must be preferable.

How to represent "bigger" (size) or "closer" (closeness) notions and how to calculate the corresponding likelihoods depend on the choice of LCMs.

Criterion 1 is used in (Abdallah et al., 2008). The associated LCM uses the volume Vol(.) to represent the box size and computes the likelihoods as $L_1 = (L_1^1, ..., L_1^M)$:

$$L_1^i = \frac{\text{Vol}([z_k^i])}{\text{Vol}([\hat{y}_k^i])} \ , \ i \in \{1, ..., M\} \ ,$$

with $\text{Vol}([x]) \triangleq \prod_{j=1}^{n_x} \text{width}([x_j])$ and $[x]$ an interval vector. However, other methods can also be used, perhaps with some advantages or disadvantages,

to calculate (with a normalization) the likelihoods such as $L_j = (L_j^1, ..., L_j^M)$, $j \in \{2, 3, 4\}$:

$$L_2^i = \text{Vol}([z_k^i]),\ L_3^i = \left\|\text{width}([z_k^i])\right\|_\infty,\ L_4^i = \left\|\text{width}([z_k^i])\right\|_2, ...$$

in which L_3 and L_4 are LCMs using distances between the two bounds of the intersection to represent its size. Which method will be chosen regarding its convenience and performance is not an obvious question.

Criterion 2 can be applied with different LCMs using a distance (e.g. Hausdorff) between $[\hat{y}_k^i]$ and $[y_k]$ to represent their closeness. **Criterion 2** can also be used to exploit **Information (b)** as in (Tran et al., 2018) via the central tendency of the Gaussian vector $[r_k^i] = y_k - [\hat{y}_k^i]$ along with the belief function theory in the sense that: *the more* $[\hat{y}_k^i]$ *is close to* y_k *(equivalently,* $[r_k^i]$ *is close to the mean* $[\mu_k]$*) the greater belief and plausibility* $[\hat{y}_k^i]$ *attains.*

A more detailed analysis of the method used in (Tran et al., 2018) is found in section 3.4.1 and a LCM that meets all these criteria is developed in section 3.5. □

It is worth to note that, in some cases, it is difficult to distinguish clearly between criterion and LCM as illustrated by Fig.3.1, e.g. in (Blesa et al., 2015) with Interval Bayes filtering approach or in (Nassreddine et al., 2010) and (Tran et al., 2018) with the belief theory. The reason is that the criteria are implied under complexes theories.

3.4 Toward a novel method for Box Particle Filtering

3.4.1 Indistinguishability of likelihood computation methods

In order to deal with **Objective 1**, in the literature, contractors are usually applied based on the Constraint Satisfaction Problem technique. However, this is not the most crucial step of BPFs using Scheme 4, e.g. this step is skipped in (Blesa et al., 2015). Furthermore, partition a box into M disjoint equal-volume sub-boxes and then compute the expected interval by (3.2) also help to reduce the conservatism due to interval computations. The most crucial step that differs one method to another in this class of BPFs is the Likelihood computation focusing on **Objective 2**. This is thus the main discussion of this section.

In the next, two representative groups of criteria and LCMs used in the literature will be analyzed to show their major disadvantage which is the *indistinguishability*. In general, the box likelihoods are computed at every time step k. The more they can represent the ability of a box containing the real value, the better estimate is obtained by Estimation step. Indistinguishability means that most of the computed likelihoods are quasi equal and hence not useful for distinguishing between box particles.

Group I . Apply **Criterion 1** with LCMs using the box volume for the box size representation.

This criterion is used implicitly in the contraction step of all BPF algorithms including it and applied in (Abdallah et al., 2008) with the LCM L_1. Since (Abdallah et al., 2008) is the pioneering paper to BPF, most of other related papers with bounded (or mixed) uncertainty are affected by its proposed method. Therefore L_1 is investigated as a representative method of this group.

To show the indistinguishability of this kind of methods, consider real interval vector $[y_k] = [\underline{y}_k, \overline{y}_k]$ and real point vector δ such that $0 \leq \delta \leq \overline{y}_k$ (element-wise), $\delta \in \mathbb{R}^p$. Let $T = \text{diag}\{t_1, ..., t_p\}$ be a diagonal matrix where its diagonal entries are $t_r \geq 0$, $r \in \{1, ..., p\}$.
Then, all boxes $[\hat{y}_k^i]$, $i \in \{1, ..., M\}$, having the form :

$$\left[\underline{y}_k - T\delta, \underline{y}_k + \delta\right] \quad \text{or} \quad \left[\overline{y}_k - \delta, \overline{y}_k + T\delta\right]$$

give the same likelihoods $L_1^i = \frac{1}{\prod_{r=1}^{p}(1+t_r)}$.
There are many other cases in which likelihoods are quasi equal and thus making the corresponding boxes $[\hat{y}_k^i]$ indistinguishable. For instance, M boxes $[\hat{y}_k^i]$'s may have likelihoods $L^i = 1/M \pm \epsilon_i$ with an appropriate small $\epsilon_i \geq 0$ so that $\sum_{i=1}^{M} L^i = 1$. In this case, the benefit of the likelihood computation step could be insignificant.

Group II. Use **Criterion 2** to exploit **Information (b)**.

The LCM used in (Tran et al., 2018) is investigated as the representative method of this group to deal with stochastic or mixed uncertainties and with additive Gaussian measurement noises.

In this method, the innovation term $r_k = y_k - h(x_k, u_k)$ is Gaussian with $\mu_k \in [\mu_k] = [\underline{\mu}_k, \overline{\mu}_k]$ and $\Sigma_k \in [\Sigma_k] = [\underline{\Sigma}_k, \overline{\Sigma}_k]$. It belongs to some of intervals $[r_k^i] = y_k - [\hat{y}_k^i]$, $i \in \{1, ..., M\}$. A mass function $m(.; [\mu_k], [\Sigma_k])$ is defined with focal elements

$$[HV_\alpha] = \left[\underline{\mu}_k - \sqrt{\alpha \text{Diag}_v(\overline{\Sigma}_k)}, \overline{\mu}_k + \sqrt{\alpha \text{Diag}_v(\overline{\Sigma}_k)}\right] , \ \alpha \geq 0,$$

where $\sqrt{(.)}$ is an element-wise operator and $\text{Diag}_v(X)$ returns the diagonal of matrix X as a vector (see Chapter 2, Section 2.2.1).

Then, the belief $bel(.)$ and plausibility $pl(.)$ of $[r_k^i]$ are computed and considered as lower and upper bound of the probability of $[r_k^i]$ containing the real value r_k. At this stage, **Criterion 2** is applied based on the central tendency of the Gaussian vector r_k: the more $[\hat{y}_k^i]$ is close to y_k (equivalently $[r_k^i]$ is close to $[\mu_k]$), the greater belief and plausibility the box particles $[x_k^i]$ (that yield $[\hat{y}_k^i]$ via the function h) attains. The computation rule is that:

$$\begin{aligned} bel([x]) &= F_{n+2}(\alpha_{bel}) \quad , \quad \alpha_{bel} = \max\{\alpha : [HV_\alpha] \subseteq [x]\} \ , \\ pl([x]) &= 1 - F_{n+2}(\alpha_{pl}) \quad , \quad \alpha_{pl} = \min\{\alpha : [HV_\alpha] \cap [x] \neq \emptyset\} \ , \end{aligned}$$

where n is the dimension of considered boxes, F_{n+2} is the cumulative distribution function of the χ^2 distribution with $n+2$ degrees of freedom. Finally, the likelihood of each particle $[x_k^i]$ is computed thanks to the Generalized Bayes theorem (GBT) and Pignistic transformation.

Example 4 (Belief and plausibility computation in EBPF method). Consider example 3 in (Tran et al., 2018). One compute the belief and plausibility of 3 boxes $[x_1], [x_2], [x_3]$ where the result is shown in Fig. 3.2 considering that $[x_i] = y_k - [\hat{y}_k^i]$, $i = 1, 2, 3$. □

The indistinguishability of the method is shown via the following two critical points.

Firstly, all boxes that do not contain $[\mu_k] \equiv [HV_0]$ have a positive plausibility and a null belief. This fact gives us a very poor information in terms of probabilities. The probability of a box containing the real value in this case belong to $[bel, pl] = [0, pl]$ with $0 \leq pl \leq 1$. The more a box is close to $[\mu_k]$, the more its plausibility is close to 1, and hence the weaker information is provided.

Secondly, all boxes intersecting $[\mu_k]$ have the plausibility 1. So, these boxes are not distinctive regarding their plausibilities. They are distinguish only by their beliefs, in which:

- For the boxes that intersect $[\mu_k]$ but do not contain it, their beliefs are 0 and $[bel, pl] = [0, 1]$. A zero information can be issued about these boxes in this case.
- For the ones containing $[\mu_k]$, their beliefs are characterized by the greatest focal element $[HV_{\alpha_{bel}}]$ they contain. The greater $[HV_{\alpha_{bel}}]$ a box can contain, the more belief it gets.
  It is quite similar to apply the rule: "*the more* $[r_k^i]$ *is centralized (having a bigger intersection with* $[\mu_k]$*) and has a bigger volume, the greater likelihood it gets*". Different LCMs can be applied using that rule with a lightened calculation strategy and background theory.

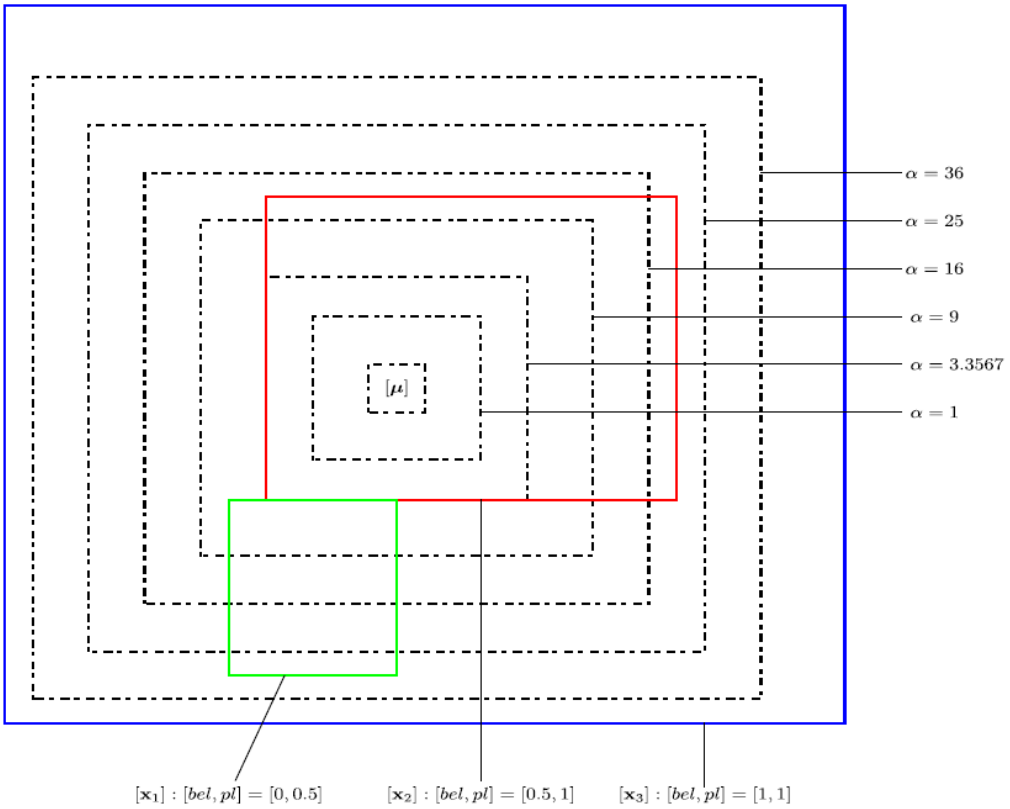

Figure 3.2 – Belief and plausibility computation example for EBPF method

Due to the above two critical points, the likelihoods computed in the next step using GBT and Pignistic transformation are almost indistinguishable. The following example illustrates this fact. The computation formulae are as follows (Tran et al., 2018):

$$m(A|y_k) = \eta \prod_{[x_k^i] \in A} pl\left(y_k - [h]\left([x_k^i], [u_k]\right)\right) \prod_{[x_k^j] \notin A} \left[1 - pl\left(y_k - [h]\left([x_k^j], [u_k]\right)\right)\right],$$

$$L_k^i = \sum_{A \subset \Omega, A \neq \emptyset} \frac{m(A|y_k)}{|A|} \cdot \mathbb{I}([x_k^i] \in A) \quad , \quad \forall [x_K^i] \in \Omega,$$

where $\Omega = \{[x_k^i], i = 1, ..., M\}$, A a subset of Ω, $|A|$ the cardinality of A and $\eta = 1 - \prod_{[x_k^i] \in \Omega} [1 - pl\left(y_k - [h]\left([x_k^i], [u_k]\right)\right)]$.

Example 5 (Likelihood computation using GBT and Pignistic transformation in EBPF method). Let $\Omega = \{[x_k^1], [x_k^2], [x_k^3]\}$ where $[x_k^i] \in \mathbb{IR}^2$, $i = 1, 2, 3$. For short, put $[r_k^i] = y_k - [h]([x_k^i], [u_k])$, $i = 1, 2, 3$. Using the assumption $v_k \sim \mathcal{N}(\mu_k, \Sigma_k)$, $\mu_k \in [\mu_k]$, $\Sigma_k \in [\Sigma_k]$, in this example, we assume further that $[r_k^i] \cap [\mu_k] \neq \emptyset$, $i = 1, 3$, and $[r_k^2] \cap [\mu_k] = \emptyset$. Then, we get $pl([r_k^i]) = 1$ for $i = 1, 3$ and $0 < pl([r_k^2]) < 1$. Therefore $\eta = 1$ and

$$\begin{aligned} L_k^1 &= \frac{m([x_k^1]|y_k)}{|\{[x_k^1]\}|} + \frac{m(\{[x_k^1], [x_k^2]\}|y_k)}{|\{[x_k^1], [x_k^2]\}|} + \frac{m(\{[x_k^1], [x_k^3]\}|y_k)}{|\{[x_k^1], [x_k^3]\}|} + \frac{m(\{[x_k^1], [x_k^2], [x_k^3]\}|y_k)}{|\{[x_k^1], [x_k^2], [x_k^3]\}|}, \\ &= \frac{1 - pl([r_k^2])}{2} + \frac{pl([r_k^2])}{3}, \end{aligned}$$

in which

$$\begin{aligned} m([x_k^1]|y_k) &= pl([r_k^1]) \left(1 - pl([r_k^2])\right) \left(1 - pl([r_k^3])\right) &= 0, \\ m(\{[x_k^1], [x_k^2]\}|y_k) &= pl([r_k^1]) pl([r_k^2]) \left(1 - pl([r_k^3])\right) &= 0, \\ m(\{[x_k^1], [x_k^3]\}|y_k) &= pl([r_k^1]) pl([r_k^3]) \left(1 - pl([r_k^2])\right) &= 1 - pl([r_k^2]), \\ m(\{[x_k^1], [x_k^2], [x_k^3]\}|y_k) &= pl([r_k^1]) pl([r_k^2]) pl([r_k^3]) &= pl([r_k^2]). \end{aligned}$$

Similarly, we get

$$L_k^3 = \frac{1 - pl([r_k^2])}{2} + \frac{pl([r_k^2])}{3} \quad , \quad L_k^2 = \frac{pl([r_k^2])}{3}.$$

Having the same likelihood, $[x_k^1]$ and $[x_k^3]$ are thus indistinguishable.
If furthermore $pl([x_k^2])$ is close to 1 then all the three likelihoods are quasi equal. □

Remark 20. In (Blesa et al., 2015), a more general framework is applied for stochastic uncertainty context. The measurement additive noise v_k can be non Gaussian and the weights (likelihoods) w_k^i are updated thanks to Bayesian Filtering strategy:

$$w_k^i \propto \text{prior distribution } \times \int_{x_k \in [x_k^i]} p_v(y_k - h(x_k, u_k)) dx_k$$

where the integral term approximates to $\int_{t \in [r_k^i]} p_v(t) dt$. The **Criterion 2** is then interpreted as: the more $[r_k^i]$ is close to the high density region of v_k, this integral term and hence w_k^i gets greater value. □

3.4.2 Requirements of a novel Box Particle Filter method

We aim to find a novel method benefiting almost advantages of existing criteria and LCMs and also attaining a gain in computation time. More precisely, the novel one

— must exploit as much as possible the supplied information needed to build the likelihood,
— must provide a LCM reducing the indistinguishability,
— can combine many simple methods rather than use only a complex one in order to get a gain in computation time while the algorithm performance is at least not weakened (or weakened in an acceptable margin).

A combination of several methods in parallel can be used to benefit all their advantages but with a large requirement of resources and with no gain even high computation time cost. Therefore, such a method is not in the scope of our intention.

3.5 Reinforced Likelihood Box Particle Filter (RLBPF)

3.5.1 Assumptions

Consider system (Σ) under **Assumptions (A), (B2), (D1)** and **(D2)**. The measurements are generally intervals $[y_k]$ due to sensor errors.

Remark 21. The above measurement assumptions concern sensor errors and model (stochastic) uncertainties. Regarding to Remark 17, it is necessary to treat $[y_k] = [y_k] - [v_k]$ where $[v_k]$ is a confidence interval of v_k chosen practically as proposed in (Tran et al., 2018) by

$$[v_k] = \left[\underline{\mu}_k - r\sqrt{\mathrm{Diag}_v(\overline{\Sigma}_k)}, \overline{\mu}_k + r\sqrt{\mathrm{Diag}_v(\overline{\Sigma}_k)}\right], \tag{3.5}$$

where $r = 1, 2, 3$. This treatment generalizes the one in Remark 17 when the measurements are given by point values. □

3.5.2 Method and Algorithm (Essential version)

In this section, for the sake of simplicity, we introduce the core of the method also the essential version of the RLBPF Algorithm. The full version Algorithm will be introduced via the Quarter vehicle model simulation at the

next section with all optional techniques that are able to make the RLBPF more efficient and suitable for numerous applications.

The essential of the proposed method RLBPF is to build a score function $J_k = (J_k^1, ..., J_k^M)$ satisfying many criteria. The particles having small scores J_k^i are preferable. After computing J_k^i's, these scores are then normalized and transformed into likelihoods W_k^i's . The smaller J_k^i corresponds to the greater W_k^i.

The proposed score function J_k is:

$$J_k^i = \left(d_{k,1}^i + d_{k,2}^i\right) V_k^i, \quad i \in \{1, ..., M\}, \tag{3.6}$$

where

$$\begin{aligned} d_{k,1}^i &= d_H([y_k], [\hat{y}_k^i]) \quad \text{(Hausdorff distance)}, \\ d_{k,2}^i &= \|\text{mid}([y_k]) - \text{mid}([\hat{y}_k^i])\|_2, \\ V_k^i &= \frac{Vol([\hat{y}_k^i] \setminus [y_k])}{Vol([\hat{y}_k^i])} = 1 - \frac{Vol([\hat{y}_k^i] \cap [y_k])}{Vol([\hat{y}_k^i])}. \end{aligned} \tag{3.7}$$

Thereby, J_k measures the closeness between $[\hat{y}_k^i]$'s and $[y_k]$ via both a kind of maximum distance $d_{k,1}^i$ and a kind of concentric tendency measure $d_{k,2}^i$. J_k also takes into account the size of intersections $[\hat{y}_k^i] \cap [y_k]$ via the volume proportions V_k^i's. Consequently, J_k exploits at the same time **Information (a)** and **Information (b)** and meets both **Criterion 1** and **Criterion 2**.

Then J_k is sorted in ascending direction and a reduction percentage $R\%$ is applied, i.e. $N_{hold} = \lfloor (100 - R)\% M \rfloor$ particles corresponding to N_{hold} first scores $\{J_k^i\}_{i=1,...,N_{hold}}$ of the sorted J_k are retained. This stage is optional with $R\%$ can be 0. It is however recommended using $0 < R\% \leq 0.3$ to penalize directly unlikelihood particles and to reduce the conservatism of interval computations as well as the computation time.

There are several ways to compute likelihoods from the score function J_k such as:

$$W_k^i = \frac{1 - J_k^i / \text{mean}(J_k)}{N_{hold} - 1}, \quad \text{mean}(J_k) = \sum_{p=1}^{N_{hold}} J_k^p, \tag{3.8}$$

where $(N_{hold} - 1)^{-1} = \left(\sum_{i=1}^{N_{hold}} (1 - J^i / \text{mean}(J_k))\right)^{-1}$ is the normalization constant, or :

$$W_k^i = \frac{\exp\{-J_k^i\}}{\sum_{p=1}^{N_{hold}} \exp\{-J_k^i\}} = \frac{\exp\{-J_k^i + c\}}{\sum_{p=1}^{N_{hold}} \exp\{-J_k^i + c\}}, \tag{3.9}$$

where c is any chosen positive constant to avoid the case that $\exp\{-J_k^i\}$'s are too small and represented numerically as 0 (e.g. the choice $c = \text{mean}(J)$ is recommended). In this developed method, in order to reinforce one more time (beside the use of the score function J_k and after the reduction stage) the distinguishability between N_{hold} remained particles, the later computing method (3.9) is chosen.

After computing estimate $[x_{k+1}]$ according to (3.2), a backward estimate is added as follows:

$$[x_k] = \text{hull}\{[x_k^i]\}, \tag{3.10}$$

for those $[x_k^i]$'s correspond to W_k^i's just computed. This backward estimation does not used in (Abdallah et al., 2008; Nassreddine et al., 2010; Blesa et al., 2015; Tran et al., 2018).

The proposed method is summarized in Algorithm 5.

Algorithm 5 Reinforced Likelihood Box Particle Filter (Essential version)

Initialization:
$[x_0] \equiv [\hat{x}_0]$, $R\%$, M, $[u_k]$, $[w_k]$, $[y_k]$, $[\mu_k]$, $[\Sigma_k]$, $k = 1, ..., N$.
Compute $N_{hold} = \lfloor (100 - R)\%M \rfloor$.
for $k = 1, 2, 3, ...N$ **do**
  Partition $[\hat{x}_{k-1}]$ into M disjoints sub-boxes $\{[\hat{x}_{k-1}^i]\}_{i=1,...,M}$
  Propagation:
  $[\hat{x}_k^i] = [f]([\hat{x}_{k-1}^i], [u_k], [w_k])$, $i = 1, ..., M$
  Likelihood computation:
  $[\hat{y}_k^i] = [h]([\hat{x}_k^i], [u_k])$, $i = 1, ..., M$
  Compute $J_k = (J_k^1, ..., J_k^M)$ using equation (3.6)
  Sort J_k in ascending direction and hold N_{hold} first values:
    $[J_k, \text{index}] = \text{sort}(J_k)$;
    $J_k = J_k(1 : N_{hold})$;
    $\text{index} = \text{index}(1 : N_{hold})$;
  Compute $W_k^i, i = 1, ..., M$ using equation (3.9)
  Estimation:
    $[\hat{x}_k] = \sum_{i \in \text{index}} W_k^i . [\hat{x}_k^i]$
    $[\hat{x}_{k-1}] = \text{hull}\{[\hat{x}_k^i], i \in \text{index}\}$
end for

Remark 22. BPFs often use a non large (small) number of particles to gain computation time and reduce the loss of a guaranteed estimation. Consequently, the resampling or repartition step happens almost always, at every

or only after a few iterations. In some sense, the fact that we hold previous weights and update them afterward has no significant effect while this effect might not be quantified straightforwardly. Furthermore, conditions under which the resampling or repartition is implemented base usually on some heuristic choice of a threshold. This is also an issue of discussion but out of the scope of the present work. Therefore, the proposed method uses a reasonable (small) number of particles, performs the repartition at each iteration and strengthens the likelihood computation and the estimation with more efficient strategies. □

3.5.3 Performance evaluation of Box Particle Filters sharing the general Scheme

In order to evaluate how the computed likelihoods bring efficiency to the estimation, it must compare the result of a BPF with that of the basic scenarios of Scheme 4:

- **Scenario 1**: Using the contraction step without partition (1 box particle);
- **Scenario 2**: Using equi-likelihood $1/M$ and without contraction step (M box particles);
- **Scenario 3**: Using equi-likelihood $1/M$ and with contraction step (M box particles).

The reason is that, in some applications, using solely the contraction step, the algorithm performance has been rather good and the efficiency brought by the computed likelihoods might be insignificant. The same manner might happen for the other scenarios.

The following indexes, proposed in (Tran et al., 2018), will be used for performance evaluations:

$$\begin{aligned}
\overline{RMSE}_j &= \sup \sqrt{\sum_{k=1}^{N}(x_{k,j} - [\hat{x}_{k,j}])^2/N},\ j \in \{1, ..., n_x\} \\
E &= \sum_{k=1}^{N} \text{width}([\hat{x}_k])/N = (E_1, ..., E_{n_x})^T, \\
O &= \sum_{k=1}^{N} \mathbb{I}(x_k \in [\hat{x}_k])/N = (O_1, ..., O_{n_x})^T,
\end{aligned}$$

where $\overline{RMSE}$ is the root mean squared error upper bound, $\mathbb{I}(.)$ is the indicator function.

3.5.4 Academic simulation example

Consider the following nonlinear system which was used as an illustration example in (Tran et al., 2018). It will be relaunched in the present work to compare the proposed method (RLBPF) with the one (EBPF) in the reference.

$$\begin{aligned} x_{k+1} &= \begin{pmatrix} \alpha_{k,1} & 1 \\ 1-\alpha_{k,1} & \alpha_{k,2} \end{pmatrix} x_k + \begin{pmatrix} \beta_{k,1} & 0 \\ 0 & \beta_{k,2} \end{pmatrix} u_k + \begin{pmatrix} 20 & 0 \\ 0 & 10 \end{pmatrix} w_k, \\ y_k &= x_{k,2} x_k/10 + v_k, \end{aligned}$$

with $z_k = (z_{k,1}, z_{k,2})^T$, $z_k \in \{x_k, u_k, w_k\}$,

$\alpha_{k,i} = (0.2 + e_i^T \delta_k/20)(100 + e_i^T x_k)/200$,

$\beta_{k,i} = (200 + e_i^T x_k)/400$, $i \in \{1, 2\}$,

$e_1 = [-1, 1]^T$, $e_2 = [1, 2]^T$,

$\delta_k \in [\delta_k] = ([-0.1, 0.1], [-0.1, 0.1])^T$,

$u_k \in [u_k] = ([75, 85], [-35, -25])^T$,

$w_k \in [w_k] = ([-0.01, 0.01], [-0.01, 0.01])^T$.

The initial state is $x_0 = [90, 80]^T$ with $[x_0] = ([85, 103], [75, 91])^T$, the number of iteration $N = 10000$ and $v_k \sim \mathcal{N}(\mu_k, \Sigma_k)$ where $\mu_k \in [\mu_k]$, $\Sigma_k \in [\Sigma_k]$, $[\mu_k] = ([-1, 1], [-1, 1])^T$ and $[\Sigma_k] = \text{diag}\{[90, 200], [90, 200]\}$.

Since, in (Tran et al., 2018), the point value measurements y_k are considered, in this simulation we also use such an assumption and get $[y_k] = y_k - [v_k]$, where $[v_k]$ is chosen as in (3.5) with $r = 3$. The reduction percentage $R = 20\%$ is applied throughout the simulation for RLBPF and the particle number $M = 9$ is applied for both methods.

Let's consider the three basic scenarios (see Section 3.5.3) of the simulation (Table 3.1) and the comparison between RLBPF and EBPF (Table 3.2 and Fig 3.3).

	Scenario 1		Scenario 2		Scenario 3	
	$j = 1$	$j = 2$	$j = 1$	$j = 2$	$j = 1$	$j = 2$
$\overline{RMSE}_j$	11.04	19.22	4.72	8.06	4.70	7.85
O_j (%)	100	100	99.80	99.92	99.80	99.80
E_j	18.86	31.27	6.88	11.92	6.86	11.68
Time (s)	46.93		49.12		81.98	

Table 3.1 – Academic example - The three basic scenarios of Box Particle Filters

Table 3.1 shows that using only the contraction step gives no good performance results in terms of $\overline{RMSE}$ and E indexes (Scenario 1). Comparing Scenarios 2 and 3, it is shown that the contraction step just brings a poor efficiency to the use of equi-likelihood. Table 3.2 shows the better performance of RLBPF versus EBPF in terms of $\overline{RMSE}$, E indexes and the computation time (with a reduction of more than 60%). Also, in this simulation example, EBPF performance is not better than those of the two basic scenarios 2 and 3.

	RLBPF		EBPF	
	$j=1$	$j=2$	$j=1$	$j=2$
$\overline{RMSE}_j$	4.67	7.43	5.62	8.90
O_j (%)	99.76	99.97	99.99	99.92
E_j	6.85	11.81	8.72	14.41
Time (s)	67.49		190.06	

Table 3.2 – Academic example - RLBPF versus EBPF

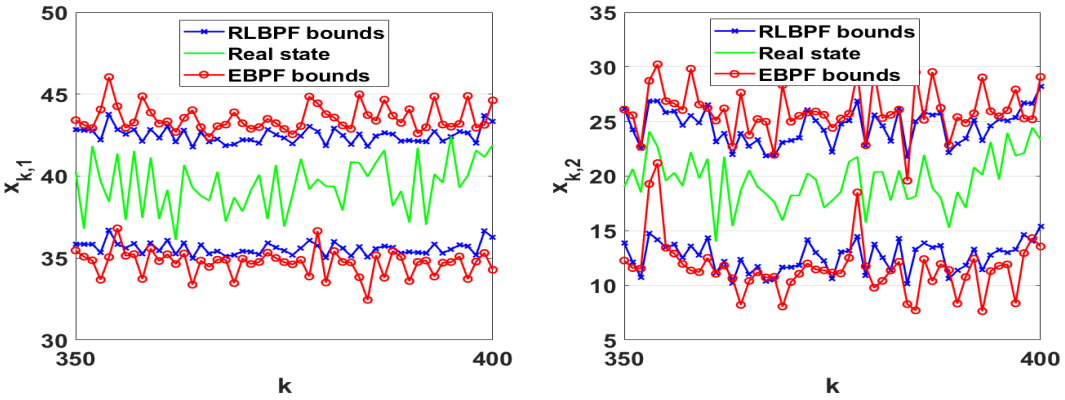

Figure 3.3 – Academic example - RLBPF versus EBPF

3.6 Application - The RLBPF full version Algorithm

3.6.1 Quarter vehicle model

The vertical quarter car model is often used to study the vertical behavior of a vehicle according to the suspension characteristic (passive or controlled) (Fig. 3.4). When controlled suspension is considered, the passive damper F_c

is removed and replaced by an actuator that provides a force u either active or semi-active depending on the chosen actuator (Fig. 3.5). In figures, the sprung mass m_s and unsprung mass m_{us} represent respectively the vehicle chassis and the vehicle wheel. z_s and z_{us} are respectively the relative vertical displacement of the vehicle chassis and the vehicle wheel with respect to the road. z_r is considered as the road disturbance.

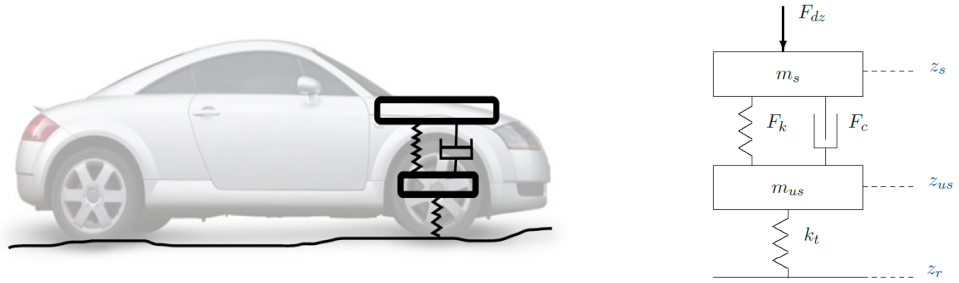

Figure 3.4 – Quarter vehicle model

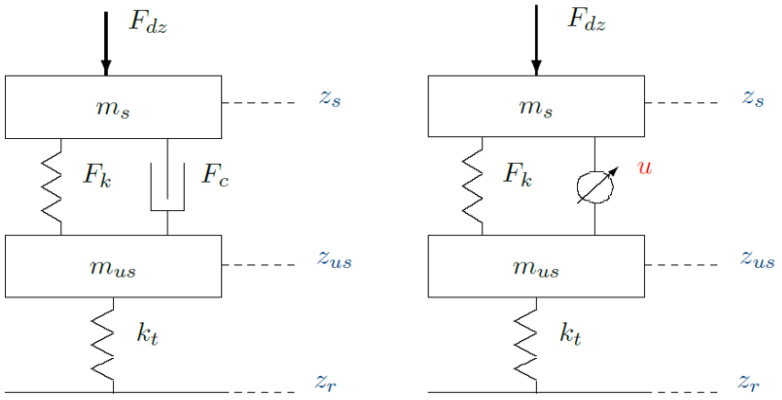

Figure 3.5 – Quarter vehical model - Passive (left) and Active control (right) modes

Vertical efforts generated by the suspension and tire elements are nonlinear. Let recall that:

$$\begin{cases} F_{tz} = k_t\left(z_{us} - z_r\right) + c_t\left(\dot{z}_{us} - \dot{z}_r\right) & \\ F_{sz} = F_k\left(z_s - z_{us}\right) + F_c\left(\dot{z}_s - \dot{z}_{us}\right) & \text{(passive suspension)} \\ F_{sz} = F_k\left(z_s - z_{us}\right) + u & \text{(controlled suspension)} \end{cases} \tag{3.11}$$

where k_t and c_t are the linear tire stiffness and damping factors, F_{tz} the tire force usually assumed to be linear and F_{sz} the suspension force.

The vertical quarter car model is given by the following dynamical equations,

$$\begin{cases} m_s \ddot{z}_s &= -(F_{sz} + F_{dz}) \\ m_{us} \ddot{z}_{us} &= F_{sz} - F_{tz} \end{cases} \tag{3.12}$$

where

- $z_{def} = (z_s - z_{us})$ is the suspension deflection,
- z_s and z_{us} are the chassis and unsprung masses bounce,
- m_s and m_{us} are sprung and unsprung masses,
- $F_k(.)$ is a nonlinear function of z_{def},
- $F_c(.)$ is a nonlinear function of $\dot{z}_{def}$,
- F_{dz} describes a vertical disturbance force (that can be caused by a load transfer, e.g. steering situation).

Then, according to the suspension model chosen, different kinds of quarter car models may be obtained:

— If $u = F_c(\dot{z}_{def})$, the suspension is passive.
— If $u = F_c(\dot{z}_{def}, \Omega)$, the suspension is semi-active, where Ω is input parameter of the controlled damper that modifies the damping factor.
— If u is an independent function, the quarter car is said to be active.

Remark 23. In the vertical quarter vehicle model, the nonlinear phenomena come from the force description of the suspension elements and not from the equation structure. Therefore, the model can be set as a LPV system.

The unsprung mass m_{us} corresponds to the set of elements that compose the wheel, the suspension system and multiple links from the chassis to the "road". Without loss of generality, it is often referred to as the wheel since z_{us} is the center of the wheel. □

3.6.2 Simulation

Consider the following nonlinear system modeling the MR (Magneto-Rheological) damper:

$$\begin{cases} m_s \ddot{z}_s &= -k_s z_{def} - F_{damper} \\ m_{us} \ddot{z}_{us} &= k_s z_{def} + F_{damper} - k_t(z_{us} - z_r)\,, \end{cases} \tag{3.13}$$

$$F_{damper} = c_0 \dot{z}_{def} + k_0 z_{def} + f_I \tanh\left(c_1 \dot{z}_{def} + k_1 z_{def}\right),$$

where c_0, k_0, c_1, k_1 are constant chosen according to (Nino-Juarez et al., 2008) such that

$$c_0 = 1500\ (\mathrm{Nsm}^{-1}),\quad c_1 = 129\ (\mathrm{sm}^{-1}),\quad k_0 = 989\ (\mathrm{Nm}^{-1}),\quad k_1 = 85\ (\mathrm{m}^{-1}),$$

and f_I is a controllable force depending on the input current I and satisfying the dissipativity constraint

$$0 \ < \ f_{\min} \ \leq \ f_I \ \leq \ f_{\max} \, .$$

In this simulation, we consider $f_{\min} = 1000$ N/m and $f_{\max} = 1500$ N/m. Other parameter values used in the simulation are presented in Table 3.3 issued from (Fergani, 2014).

Symbol	Value	Unit	Signification
m_s	315	kg	sprung mass
m_{us}	37.5	kg	unsprung mass
k_s	29500	N/m	suspension linearized stiffness
k_t	208000	N/m	tire stiffness
z_{def}	$[-0.09; 0.05]$	m	suspension bound (stroke limit)

Table 3.3 – Linearized Renault Mégane Coupé parameters of the quarter vertical model (front suspension).

Comparing to the general system (3.12), in the MR damper model (3.13), it is assumed that $F_{dz} = 0$ and $F_{tz} = k_t(z_{us} - z_r)$.

Putting

- $x = [z_s, \dot{z}_s, z_{us}, \dot{z}_{us}]^T$ as state variable under consideration and $x(i)$, $i \in \{1, ..., 4\}$, are its components,
- $u = f_I$ as controllable input,
- $w = z_r$,

then x, u, w are functions of time t and the state-space representation of (3.13) is expressed in the form

$$\dot{x}_t \quad = \quad f(t, x_t) \quad = \quad \begin{bmatrix} f_1(t, x_t) \\ f_2(t, x_t) \\ f_3(t, x_t) \\ f_4(t, x_t) \end{bmatrix} \, , \tag{3.14}$$

where

$$\begin{aligned} f_1(t, x_t) \quad &= \quad e_2^T x_t \quad , \\ f_3(t, x_t) \quad &= \quad e_4^T x_t \quad , \\ f_2(t, x_t) \quad &= \quad \left(a^T x_t - u_t \tanh(b^T x_t)\right) / m_s \quad , \\ f_4(t, x_t) \quad &= \quad \left(c^T x_t + u_t \tanh(b^T x_t) + k_t w_t\right) / m_{us} \quad , \end{aligned}$$

with e_i's are i-th standard unit vectors and

$$a = \begin{bmatrix} -k_s - k_0 \\ -c_0 \\ k_s + k_0 \\ c_0 \end{bmatrix}, \qquad b = \begin{bmatrix} k_1 \\ c_1 \\ -k_1 \\ -c_1 \end{bmatrix}, \qquad c = \begin{bmatrix} k_s + k_0 \\ c_0 \\ -k_s - k_0 - k_t \\ -c_0 \end{bmatrix}.$$

The system (3.14) will be discretized using the Fourth order Runge-Kutta method (Kincaid and Cheney, 1991) with a chosen sampling time $T = 10^{-4}$(s). The resulted discrete time state dynamical system is denoted by:

$$x_k \;=\; \tilde{f}(x_{k-1}, u_k, w_k) \;+\; \eta_k, \qquad k \in \mathbb{N}^*, \tag{3.15}$$

where η_k is assumed to be Gaussian noise with zero mean and covariances $10^{-8} I_{n_x}$. The corresponding observed measurements are assumed to be z_{def} at every time step, thus the measurement dynamical equation can be expressed in the form

$$y_k \;=\; h(x_k) \;+\; v_k \;=\; C x_k \;+\; v_k \quad , \quad C = [1, 0, -1, 0]\,, \tag{3.16}$$

where v_k is assumed to be Gaussian with mean $\mu_k \in [\mu_k] = [-0.005, 0.005]$ and variance $\sigma_k^2 \in [\sigma_k^2] = [1, 4] * 10^{-6}$. The precision of the sensors is assumed to be ± 0.005 (m).

State and measurement simulation: Assume that the initial state is $x_0 = (0, 0, 0, 0)^T$, the control force input is set to get its maximum value constantly ($u = 1500$) for all time instants and the road disturbance is set as $w = 0.05 \max\{0, \sin(\pi t)\}$. $\{x_k, y_k\}_{k=1:N}$ are then generated using (3.15) and (3.16) for $N = 4.10^4$ steps. The measurements obtained will be intervals $[y_k] = y_k \pm 0.005$ because of sensor errors.

The three basic scenarios of BPFs sharing Scheme 4: The filters start at $[x_0] = [-0.06, 0.06] \times \mathbf{1}_{n_x}$.

— Scenario 1 is the simplest one without partition and contraction. It has a divergent result with $\overline{RMSE} \propto 10^{97}$ and the mean of widths $E \propto 10^{97}$, where $x \propto 10^p$ means that $x = c.10^p$ with $0 < c < 10$. The computation time of this scenario is 490s.

— For scenarios 2 and 3, the questions arisen are that how many particles will be made and at which (all or some and which ones) components of the box the partition will be implemented. The state variable has 4 components. We tried with an intermediate solution: the partition is a bisection at the two components having greater widths among them, so the number of particles is 4. Scenarios 2 and 3 have very similar resulted estimates as shown in Table 3.4 and Figure 3.6. The

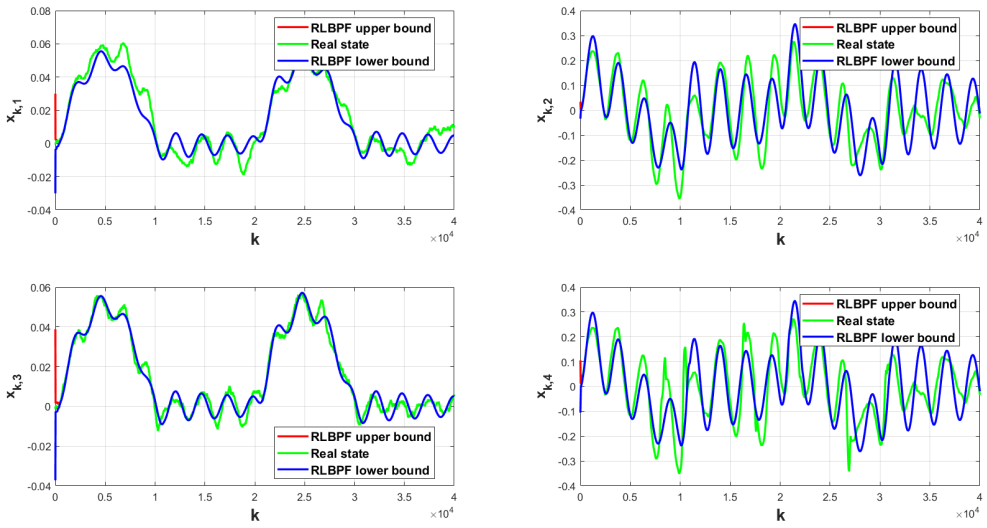

Figure 3.6 – Quarter vehicle model - Scenario 2 and Scenario 3 with 4 particles. (Both scenarios have very similar (but not coincident) resulted estimates. The figure shows only the ones of Scenario 3.)

	$\overline{RMSE}_j$		$O_j(\%)$		E_j	
Scenario	2	3	2	3	2	3
$j=1$	0.007	0.006	1.14	0.71	$\propto 10^{-4}$	
$j=2$	0.081	0.080	0.47	0.40	$\propto 10^{-4}$	
$j=3$	0.004	0.003	1.58	1.07	$\propto 10^{-4}$	
$j=4$	0.091	0.090	0.57	0.61	$\propto 10^{-4}$	

Table 3.4 – Quarter vehicle model - Scenarios 2 and Scenario 3 with 4 particles.

resulted estimates in these scenarios are nearly point estimates with rather good $\overline{RMSE}$ indexes (Fig. 3.6). The computation times are respectively 519s (scenario 2) and 590s (scenario 3).

— Since the more particles are partitioned, the interval estimates have smaller widths, then we also tried to reduced the number of particles to 2 and the partition is effectuated at the maximum width component of the box. Regarding Table 3.5 and Figure 3.7, the resulted estimates of both scenarios are divergent (for N becomes more and more greater).

	$\overline{RMSE}_j$		$O_j(\%)$		E_j	
Scenario	2	3	2	3	2	3
$j = 1$	1.81	0.92	100	100	2.35	1.21
$j = 2$	2.17	0.10	90.79	80.68	2.73	1.36
$j = 3$	1.88	0.92	100	100	2.47	1.22
$j = 4$	2.86	1.44	96.72	86.86	3.69	1.83

Table 3.5 – Quarter vehicle model - Scenarios 2 and 3 with 2 particles.

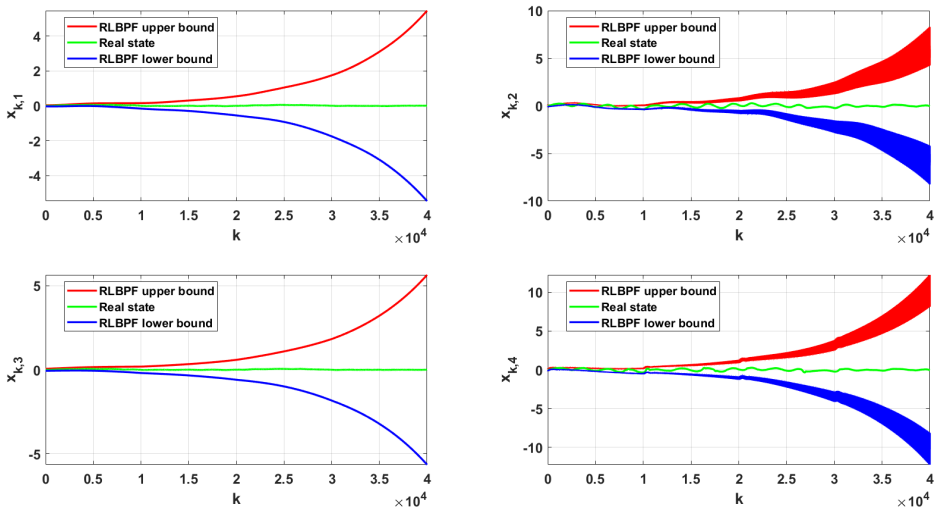

Figure 3.7 – Quarter vehicle model - Scenario 2 and Scenario 3 with 2 particles. (Both scenarios have very similar (but not coincident) resulted estimates. The figure shows only the ones of Scenario 2.)

Estimation using RLBPF: The measurements are treated according to Remark 21. The number of particles is 4 obtained by bisection at two components with greater widths of the box to be partitioned. Since the number of particles is small, the reduction percentage $R\% = 0\%$ is used. It encounters that the essential version of RLBPF failed to provide good estimates because the volume V in (3.7) gets 0 or 1 at all of its components for many iteration steps. $V = 0$ means that all estimated measurements $[\hat{y}_k^i]$'s are contained in $[y_k]$, so the partition of the box $[\hat{x}_{k-1}]$ is not necessary hence lines $18 - 21$ in the Algorithm 6 are added. Furthermore, to avoid partially this situation, a condition at line 5 of the Algorithm is added and

controlled by a chosen constant $\mathbf{c_1}$. In contrast, when $V = 1$, all estimated measurements $[\hat{y}_k^i]$'s have empty intersection with $[y_k]$, so a regularization controlled by a constant $\mathbf{c_2}$ is provided by lines $12-17$ of the Algorithm. This regularization bases on 2 conditions: the previous estimates is good enough and the state dynamic is smooth. Finally, a smoothing factor **SmF** and a scaling factor **ScF** are applied additionally to get more reliable estimates depending on the application in consideration. The scaling factor is used in the same manner as Remark 19 has discussed. So, that is the reason of the full version of RLBPF (Algorithm 6) with additional controlling factors making the filter more suitable for numerous applications. When $\mathbf{c_1}$ is small enough, $\mathbf{c_2} < 1$, $\mathbf{SmF} = 1$, $\mathbf{ScF} = I_{n_x}$ and the lines $18 - 21$ are inactivated then the essential version of RLBPF is recovered. Thus, the methodology does not change and the full version provides more freedom to the filter.

	$\overline{RMSE}_j$		$O_j(\%)$		E_j	
Case	I	II	I	II	I	II
$j = 1$	0.040	0.125	100	100	0.065	0.228
$j = 2$	0.367	0.216	100	83.46	0.590	0.263
$j = 3$	0.014	0.116	100	100	0.022	0.223
$j = 4$	0.327	0.364	99.1	98.73	0.503	0.527

Table 3.6 – Quarter vehicle model using the RLBPF full version. Case I: $\mathbf{SmF} = 0.2$, $\mathbf{ScF} = \text{diag}\{0.9965, 2, 0.99, 1.4\}$. Case II: $\mathbf{SmF} = 1$, $\mathbf{ScF} = I_{n_x}$.

The estimation results using the full version of RLBPF with and without **SmF** and **ScF** are presented by Figures 3.8, 3.9 and Table 3.6, i.e. $\mathbf{SmF} = 0.2$, $\mathbf{ScF} = \text{diag}\{0.9965, 2, 0.99, 1.4\}$ in the first case and $\mathbf{SmF} = 1$, $\mathbf{ScF} = I_{n_x}$ in the second. In both cases, $\mathbf{c_1} = \mathbf{c_2} = 5$, $\mathbf{c_3} = 0.9$ and $\epsilon = 0.001$.

It is worth to note that, concerning the partition process, it would be natural to consider physical conservation properties as further (virtual) measurements. The investigation of this subject is found in (Rauh et al., 2011).

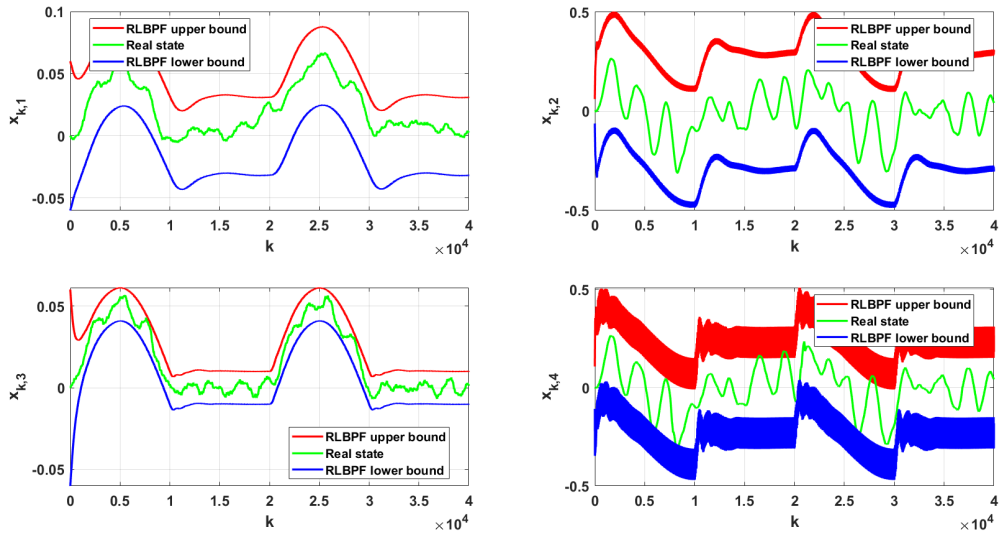

Figure 3.8 – Quarter vehicle model - RLBPF full version with smoothing and scaling factor (4 particles).

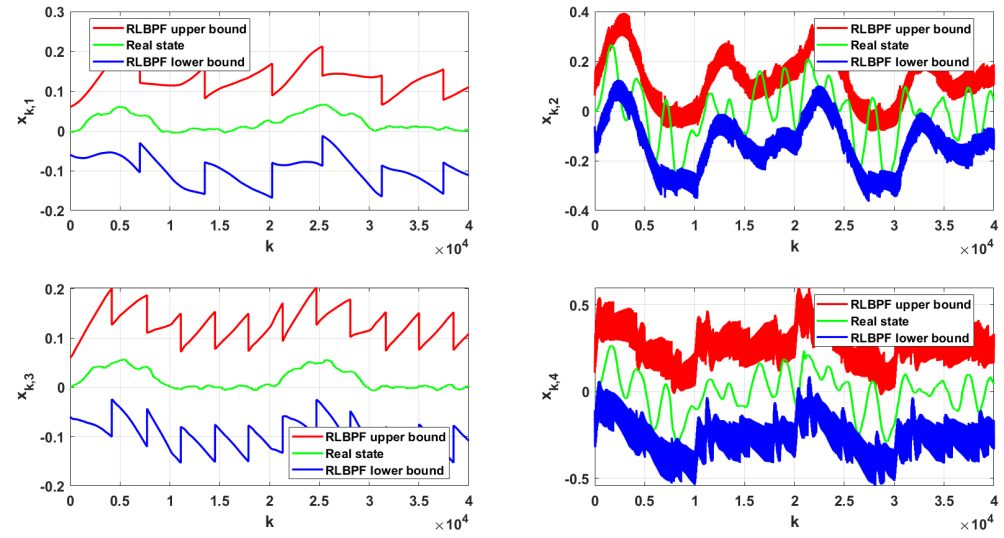

Figure 3.9 – Quarter vehicle model - RLBPF full version without smoothing and scaling factor (4 particles).

3.7 Conclusion and perspective

A general scheme is provided to generalize the specificity of BPFs. The likelihood computation methodology is investigated. This analysis point out the disadvantages of existing filters and opens a way to improve the computed

Algorithm 6 Reinforced Likelihood Box Particle Filter (Full version)

1: **Initialization:** $[x_0] \equiv [\hat{x}_0]$, $[u_k]$, $[w_k]$, $[y_k]$, $[\mu_k]$, $[\Sigma_k]$, $k = 1, ..., N$.
2: Choose: $\mathbf{c_1}$, $\mathbf{c_2}$, $\mathbf{c_3}$, ϵ, **ScF**, **SmF**, $R\%$, M.
3: Compute $N_{hold} = \lfloor (100 - R)\%M \rfloor$.
4: **for** $k = 1, 2, 3, ...N$ **do**
5: **if** $\max\{\text{width}([\hat{x}_{k-1}])\} < \mathbf{c_1}.\max\{\text{width}([\hat{x}_0])\}$ **then**
6: $[\hat{x}_k] = [f]([\hat{x}_{k-1}], [u_k], [w_k])$;
7: **else**
8: Partition $[\hat{x}_{k-1}]$ into M disjoints sub-boxes $\{[\hat{x}^i_{k-1}]\}_{i=1,...,M}$;
9: $[\hat{x}^i_k] = [f]([\hat{x}^i_{k-1}], [u_k], [w_k])$, $i = 1, ..., M$; % *Propagation*
10: $[\hat{y}^i_k] = [h]([\hat{x}^i_k], [u_k])$, $i = 1, ..., M$; %*Likelihood computation*
11: Compute $V = (V^1, ..., V^M)$ using (3.7);
12: $count = 1$;
13: **while** $V^i = 1, \forall i = 1, ..., M$ and $count < \mathbf{c_2}$ **do**
14: $[\hat{x}_{k-1}] = \mathbf{c_3}.[\hat{x}_{k-1}] + (1 - \mathbf{c_3}).[\hat{x}_{k-2}]$; %$k > 2$ *by choosing* $[\hat{x}_0]$
15: Redo lines $8 - 11$; %$\mathbf{c_3} \in [0, 1]$
16: $count \mathrel{+}= 1$;
17: **end while**
18: **if** $V^i = 0$, $\forall i = 1, ..., M$ **then**
19: $[\hat{x}_k] = [f]([\hat{x}_{k-1}], [u_k], [w_k])$;
20: **Continue** %*Skip all remaining commands in the for loop*
21: **end if**
22: $V(V == 0) \leftarrow \epsilon$; %$0 < \epsilon < \min\{V(V > 0)\}$
23: Compute $J_k = (J^1_k, ..., J^M_k)$ using (3.6) and (3.7);
24: Sort J_k in ascending direction and hold N_{hold} first values:
25: $[J_k, \text{index}] = \text{sort}(J_k)$;
26: $J_k = J_k(1 : N_{hold})$;
27: $\text{index} = \text{index}(1 : N_{hold})$;
28: Compute $W^i_k, i = 1, ..., M$ using (3.9);
29: $\text{mid} = \sum_{i \in \text{index}} W^i_k.\text{mid}([\hat{x}^i_k])$; %*Estimation*
30: $\text{rad} = \mathbf{SmF}.\left(\mathbf{ScF}.\sum_{i \in \text{index}} W^i_k.\text{rad}([\hat{x}^i_k])\right) + (1 - \mathbf{SmF}).\text{rad}([\hat{x}_{k-1}])$;
31: $[\hat{x}_k] = \text{mid} \pm \text{rad}$; %***ScF**: Scaling factors (see Remark 19)*
32: $[\hat{x}_{k-1}] = \text{hull}\{[\hat{x}^i_k], i \in \text{index}\}$; %$\mathbf{SmF} \in [0, 1]$*: Smoothing factor*
33: **end if**
34: **end for**

likelihoods by making them more reliable using a reinforcement method. Although a proper definition of the indistinguishability of likelihood computation methods is not provided as well as the degree of this indistinguishability, a strategy is proposed to evaluate the performance of this class of filters using the three basic scenarios (Section 3.5.3). The simulation highlights the efficiency of the RLBPF in gain of computation time and evaluation indexes.

In principle, the RLBPF can be implemented with any state and measurement continuous dynamical functions, unless conditions under which the filter provide a good performance or guaranteed results, e.g. with C-stability, are not pointed out. The control factors $\mathbf{c_1}, \mathbf{c_2}, \mathbf{c_3}, \epsilon$, **SmF, ScF**, on the one hand, make the filter be more efficient, flexible and suitable for numerous applications, on the other hand, they are subject to future studies about optimal choices and/or automatic adaptive choices of them either by analytical or machine learning method. Investigation of RLBPF on some concrete classes of state and measurement dynamical functions (e.g. L-Lipschitz, L_2,...) is also a potential research perspective. The score function J as well as the method weighting it may be improved and the number of particles will be applied in the partition step is an issue of the filter.

Chapter 4

Adaptive Degrees of Freedom χ^2-statistic Method to sensor fault detection

4.1 Introduction

Within the control theory and its field of applications, *fault detection* is extremely important for all system engineering problems. It is a crucial component of any system diagnosis scheme and has received a lot of attention in both academia and industry. Reliable fault detection and isolation is a first class requirement in many fields. Indeed, efficient (early and accurate) fault detection can help avoid dangerous scenarios (accidents, explosions,...) or improve productivity (reducing process activity loss such as leakage...). In 2013, the World Health Organization (WHO) has registered more than 1.24 million deaths and over than 50 million injuries worldwide on roads (globally the eighth leading cause of death) most of them caused by abnormal vehicle behavior (Prevention, 2013). In the petrochemical industries loss has been estimated to over than 20 Billion dollars every year caused by the non efficiency of the AEM (Anormal Event Management).

Many methods and techniques have been developed to meet these abundant requirements. The *model-based approaches* are proven to provide good results and acceptable tradeoff between fault sensitivity and computational cost especially those based on residual generation (see Patton et al. (2013) and references within). Several methods for fault detection in dynamic systems are mentioned in (Willsky, 1976), including the *innovation-based method* in which a χ^2-statistic hypothesis testing was used. This method is applied appropriately with the standard Kalman filter (Kalman, 1960) to process

the linear dynamic system with (known) deterministic coefficient matrices. In (Sainz et al., 2002), an approach to generate envelopes based on interval techniques of the modal interval analysis is proposed. In (Puig, 2010), the use of *set-membership methods* in *fault diagnosis* and *fault tolerant control* is reviewed. These methods aim at checking the consistency between observed and predicted behaviors by using simple sets (intervals, zonotopes,...) to approximate the exact set of possible behaviors. Also, the design of stable *interval observers* for linear systems with additive time-varying zonotopic input bounds is proposed in (Raka and Combastel, 2013). Interval observers provide an estimate on the set of admissible values of the state vector at each time instant. Ideally, the size of the evaluated set is proportional to the model uncertainty, thus interval observers generate the state estimates with estimation error bounds, similarly to Kalman filters, but in the deterministic framework. Main tools and techniques for design of interval observers are reviewed in (Efimov and Raïssi, 2016) for continuous-time, discrete-time and time-delayed systems.

The efficiency of these strategies has attracted the attention of the industrial community, especially, the automotive industry. Thus, many academical studies have tried to provide solutions within this field based on set membership *fault detection and isolation* (FDI). In (Meseguer et al., 2010), a fault diagnosis approach is proposed. It has been motivated by the problem of detecting and isolating faults of the Barcelona's urban sewer system limnimeters (level meter sensors). It is based on interval observers improving the integration of FDI tasks. (Ifqir et al., 2018) reviews the problem of robust state estimation and unknown input interval reconstruction for uncertain switched linear systems. A design method for obtaining interval observers that provide guaranteed lower and upper bounds of the state and unknown inputs is applied to vehicle lateral dynamic estimation to show the effectiveness of the algorithms. Also, in (Chen et al., 2020), an extended set-membership filter applied to the vehicle's longitudinal velocity, lateral velocity, and sideslip angle provides not only higher accuracy, but also can provide a 100% hard boundary which contains the real values of the vehicle states (compared to the *Unscented Kalman Filter* UKF-based approaches).

Recently, (Tran, 2017) proposes an approach combining the χ^2-statistics hypothesis test with the Upper Bound Interval Kalman Filter (UBIKF, Tran et al. (2017)) in order to solve detection problems dealing with interval Kalman filter. The contribution to the fault detection in (Tran, 2017) is the use of an upper bound for all positive semi-definite matrices belonging to an interval matrix. This upper bound aims to overcome the singularity of the inverse of interval matrices. Based on this concept, an adaptive hypothesis test method is developed in (Lu et al., 2021) to detect sensor faults

applied to a linear discrete time dynamic system with assumptions requiring the use of interval computations. The proposed method combines, on the one hand, OUBIKF the optimal version of UBIKF presented in Chapter 2, with, on the other hand, a χ^2 hypothesis test whose degrees of freedom (d.f.) are adaptively chosen thanks to amplifier coefficients. This is the content of the present chapter together with a development of this method applied to the nonlinear dynamical system.

The chapter is organized as follows. Section 4.2 presents the state-space representation of a dynamical system (linear or nonlinear) with additive sensor faults and performance indicators for fault detection methods. Section 4.3 provides the main contributions of the chapter, included principles and algorithms of the novel adaptive testing method for sensor fault detection to linear and nonlinear dynamical systems and two corresponding application simulations (bicycle vehicle model and suspension model). The chapter conclusion and perspectives are presented in Section 4.4 .

4.2 State-space representation with sensor faults and performance indicators for fault detection methods

4.2.1 State-space representation with sensor faults

The state-space representation with sensor faults can be expressed in the form

$$\begin{cases} x_k &= A_k x_{k-1} + B_k u_k + w_k \ , \\ y_k &= C_k x_k + D_k u_k + v_k + f_k^s \ , \end{cases} \quad k \in \mathbb{N}^*, \tag{4.1}$$

or more general

$$\begin{cases} x_k &= f_k(x_{k-1}, u_k, w_k) \ , \\ y_k &= h_k(x_k, u_k, v_k) + f_k^s \ , \end{cases} \quad k \in \mathbb{N}^*, \tag{4.2}$$

where $x_k \in \mathbb{R}^{n_x}$ and $y_k \in \mathbb{R}^{n_y}$ represent state variables and measures respectively, $u_k \in \mathbb{R}^{n_u}$ inputs, $w_k \in \mathbb{R}^{n_x}$ state noises, $v_k \in \mathbb{R}^{n_y}$ measurement noises, $f_k^s \in \mathbb{R}^{n_y}$ additive sensor fault vectors.

Sensor faults occur when an affecting value (fault) f_k^s comes into a measurement. Each of its components corresponds to a sensor fault. Thus the fault vector f_k^s can be of the multiple or single error type. In the first type, some (or all) sensors cause errors which affect the y_k value for the corresponding components. In the second type, only one sensor causes an error and just the corresponding y_k component is affected.

4.2.2 Performance indicators for fault detection methods

To evaluate the fault detection performance, some indicators are introduced. Assume that the dynamical system, (4.1) or (4.2), implements in N iterations among which faults occur in a region $\mathcal{R}$ with length l ($0 \leq l \leq N$). The region $\mathcal{R}$ may be a range or union of ranges. For simplicity, hereafter we call $\mathcal{R}$ an error range. Knowing that the detection signal has value 1 or 0, we call *right detected signal* the 1-value detection signal situated inside the error range and *false detected signal* the 1-value detection signal situated outside the error range. Furthermore,

- *Detection Rate (DR)* is determined by the number of right detected signals over the length l of error range.
- *No Detection Rate (NDR)* is determined by $\text{NDR} = 1 - \text{DR}$.
- *False Alarm Rate (FAR)* is determined by the number of false detected signals over $N - l$, the cardinal of the region outside the error range.
- The *Efficiency (EFF)* of the detection is determined by $\text{EFF} = \text{DR} - \text{FAR}$.

More details on indicators can be found in (Chen and Patton, 1999) with a slight difference.

4.3 Adaptive Degrees of Freedom χ^2-statistics (ADFC) method for sensor fault detection

4.3.1 Fault detection based on ADFC and OUBIKF for linear system

The ADFC method introduced in this section is the main contribution of (Lu et al., 2021) using Algorithm 2 (Lu et al., 2019) in the residual generation. The method works obviously with Algorithm 3 as well as with the RLBPF (Algorithm 6) as we will see in Section 4.3.3.

Consider the dynamical system (4.1) with the same assumptions **A1** using for OUBIKF. System (4.1) with assumptions **(A1)** is a quite general model adapted to a wide range of applications. In this system, parameter matrices are time varying, the uncertainty may result from different sources (system disturbances, measurement noises) and may be of different kinds (stochastic and bounded uncertainties).

The Algorithm 2 is used to generate residual intervals $[r_k] = y_k - [\hat{y}_k]$, $[\hat{y}_k] = [C][\hat{x}_{k|k-1}] + [D]u_k$, and the fault detection procedure of the proposed method is based on a statistical hypothesis testing. Therefore, it is vital to

investigate carefully the stochastic property of related terms of the system under standard conditions (SKF, Section 1.1.1), otherwise the fault detection test for the interval case cannot be derived. Summarizing these properties, in the fault free case, state x_k, measure output y_k, estimator $\hat{x}_{k|k}$, estimation error ϵ_k and residual r_k are all Gaussian vectors, in which $r_k \sim \mathcal{N}(0, S_k)$. Furthermore, the following property plays a key role for the development in the next.

Key property (K). Assuming S_k is non singular and let $\eta_k = S_k^{-1/2} r_k = (\eta_{k,1}, ..., \eta_{k,n_y})$. Then $\eta_k \sim \mathcal{N}(0, I)$, that is $\eta_{k,i}$'s are $\mathcal{N}(0,1)$-distributed and independent each other.

Innovation-based fault detection method. In the literature, using the χ^2-statistics test for sensor fault detection is a kind of *Innovation-based approach* mentioned in (Mehra and Peschon, 1971). In (Willsky et al., 1974), (Willsky et al., 1975), this method is applied for fault detection problems in which the following statistic is used

$$\nu_k = \sum_{i=k-W+1}^{k} \eta_i^T \eta_i = \sum_{i=k-W+1}^{k} r_i^T S_i^{-1} r_i \ , \tag{4.3}$$

where W is a window size ($W \leq k$) and r_i's are residual terms obtained by the SKF. The statistic ν_k is considered as a χ^2 -distributed random variable with Wn_y degrees of freedom. A rule for the fault detection test is established as: (H_0) $\nu_k \leq \delta$, no error occurred; (H_1) $\nu_k > \delta$, an error occurred, where δ is the threshold determined by $\mathbb{P}(\chi^2(Wn_y) > \delta) = \alpha$ with α a chosen significance level (or the probability of Type I error). The window size W and the threshold δ are to be chosen to provide an acceptable trade-off between the probability of declaring $(H1)$ when actually $(H0)$ and the probability declaring $(H0)$ when actually $(H1)$ (Willsky et al., 1975).

For the next development, it is worth to note that a statistic T can follow exactly a distribution F or be approximated by another statistic $\tilde{T}$ with distribution F. Any statistic can be used as estimator for a quantity of interest with or without consistency and with different accuracies.

Principles of the method. By system (4.1) and assumptions (**A1**), measures y_k and interval matrices $[A]$, $[B]$, $[C]$, $[D]$, $[Q]$, $[R]$ are known, and we obtain by computation measure estimate intervals $[\hat{y}_k] = [C][\hat{x}_{k|k-1}] + [D]u_k$, residual intervals $[r_k] = y_k - [\hat{y}_k]$ and the interval matrix $[S_k] = ([C][A])[P_{k-1|k-1}]([C][A])^T + [C][Q][C]^T + [R]$ which contains all accessible residual covariances S_k.

In the literature, to use the χ^2-statistics test method, a standard normal distribution form ($\eta_k = S_k^{-1/2} r_k \sim \mathcal{N}(0, I)$) is needed. A similar form but for the interval vector $[r_k]$ is meant to match our goals, and thus the singularity

problem of $[S_k]$ is an impediment. To overcome this impact, it is proposed in (Tran, 2017) to use the upper bound of $S_+([S_k])$ instead of $[S_k]$ and a better choice of this upper bound is applied thanks to properties developed in (Lu et al., 2019).

The following strategy is proposed in the present work:

- Find Σ_k such that $S_+([S_k]) \preceq \Sigma_k$. This upper bound matrix is of the form $\Sigma_k = a_k I$ ($a_k \in \mathbb{R}^+$) using Theorem 5.
- Compute:
  $[\tilde{\eta}_k] = \Sigma_k^{-1/2}[r_k] = [r_k]/\sqrt{a_k}$,
  $[\xi_k] = [\tilde{\eta}_k]^T[\tilde{\eta}_k] = [r_k]^T\Sigma_k^{-1}[r_k] = [r_k]^T[r_k]/a_k$.
- Apply the absolute operator for intervals $[\xi_k]$ since $\xi_k = \tilde{\eta}_k^T\tilde{\eta}_k$ is non negative for all $\xi_k \in [\xi_k]$ whilst during interval computations, most of the time $\inf([\xi_k]) < 0 < \sup([\xi_k])$.
  The absolute operator for intervals is defined by

$$\text{abs}([a\,,b]) = \begin{cases} [\min(|a|,|b|)\,,\max(|a|,|b|)] & ,\ 0 \notin [a\,,b] \\ [0\,,\max(|a|,|b|)] & ,\ 0 \in [a\,,b] \end{cases}.$$

- Let $U_k = \sup(\text{abs}([\xi_k]))$. The statistic U_k will be used in hypothesis testing for which it is approximated by a $\chi^2(\kappa_k n_y)$ random variable (explication in the next paragraph). κ_k is called an *adaptive amplifier coefficient*.

Some remarks can be made immediately as follows: $\forall k \geq 1$,

— $0 \leq \xi_k \leq \eta_k^T\eta_k \sim \chi^2(n_y)$ since $S_k \preceq \Sigma_k, \forall S_k \in S_+([S_k])$,
— $0 \leq \xi_k \leq U_k$,
— $\mathbb{E}[\chi^2(n_y)] = n_y \ll U_k$ almost of times.

It is reasonable to consider ξ_k as a χ^2-distributed random variable with a d.f. smaller than n_y, but this statistic is actually unknown. What we have in hand is the statistic U_k obtained by computation. Based on above remarks, it is proposed to approximate this statistic U_k by a χ^2-distributed random variable with an adaptive d.f. $\kappa_k n_y$ ($\kappa_k > 1$) where κ_k is an adaptive amplifier coefficient (a.a.c.). Thanks to this a.a.c., *adaptive thresholds* are built and help to detect faults.

The rule for the fault detection test is that: (H_0) $U_k \leq \delta_k$, no error occurred; (H_1) $U_k > \delta_k$, an error occurred, where δ_k is the adaptive threshold determined by $\mathbb{P}(\chi^2(\kappa_k n_y) > \delta_k) = \alpha$ with α is a chosen significance level.

After test running, an adjustment procedure is proposed to obtain detection signals more accurately. That is, in a window of size w, if the number of consecutive error occurrences is smaller than w, we consider that these errors (if exist) don't cause serious effects and will be dismissed. Furthermore, since

error is often detected with a delay, all detection signals will be shifted to the left $\lfloor w/2 \rfloor$ steps ($\lfloor . \rfloor$ is the floor function).

Choice of the a.a.c. A χ^2 distributed random variable has the cumulative distribution function with d.f. k:

$$F(x,k) = \mathbb{P}(\chi_k^2 \leq x) = \frac{\int_0^{x/2} t^{\frac{k}{2}-1} e^{-t} dt}{\int_0^{\infty} t^{\frac{k}{2}-1} e^{-t} dt} \,. \tag{4.4}$$

In the literature, k is a positive integer. However, from the analysis point of view, $F(x,k)$ is a continuous function of k ($k > 0$) at any positive value of x (since the Gamma function $\Gamma(z) = \int_0^{\infty} t^{z-1} e^{-t} dt$ is continuous for all $z > 0$). Consequently a positive real d.f. $\kappa_k n_y$ can be used.

For an accurate choice of a.a.c κ_k, some conditions are required:

— Firstly, it must be sensitive to the fault occurred.

— Secondly, it must be large enough to get a small FAR (e.g. $\leq 5\%$) in the fault free case.

— In addition, being a distribution parameter of statistic U_k, it is highly recommended that the chosen κ_k is related to the U_k's construction.

Concretely, by writing residual intervals in the form $[r_k] = \text{mid}([r_k]) + [-\frac{1}{2}, \frac{1}{2}] * \text{width}([r_k])$, the statistic U_k is expressed as

$$U_k = \frac{\|\text{width}([r_k]) + 2 \cdot \text{abs}(\text{mid}([r_k]))\|^2}{4a_k} \,, \tag{4.5}$$

a function of $\text{mid}([r_k])$ and $\text{width}([r_k])$ where the later is more sensitive to the fault than the former. The residual width is a major factor influencing the U_k's computation and, furthermore, reflects the performance of the model and algorithm. Consequently it is reasonable to chose κ_k as a function of residual width.

Remark 24. An example is shown in Fig.4.1 illustrating the sensitivity to the fault of the residual width. It is simulated from Bicycle vehicle model presented in section 4.3.2 and according to the result shown in Fig.4.5. Indeed, Fig.4.1 shows that residual widths become very large inside the error range (between the two vertical lines) whilst residual midpoints are stable around 0 outside the error range and do not change too much inside it. □

Which function of residual width will be chosen is a hard problem due to many impacts, for instance:

— no further information about $\text{width}([r_k])$ and $\text{mid}([r_k])$ is available,

— κ_k and corresponding threshold δ_k are both unknown; it exists only a relation represented via the quite complex function (4.4) so that $F(\delta_k, \kappa_k n_y) \geq 1 - \alpha$,

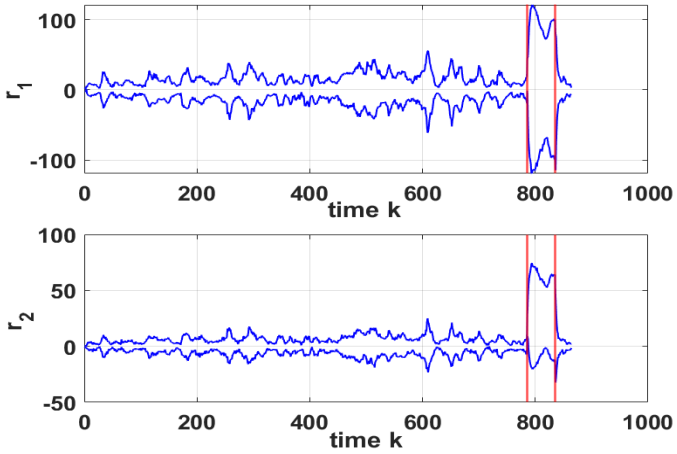

Figure 4.1 – Behavior of the first (top) and the second (bottom) residual components with fault value $b = 20$.

— the yielded δ_k (by κ_k) must satisfy the fault detection constraint: $\delta_k \geq U_k$ when no error occurs and $\delta_k < U_k$ otherwise.

Therefore, an additional requirement for the chosen κ_k is that it must (while being sufficiently large in the fault free case as aforementioned) not increase as fast as U_k when an error occurs and affects on the width($[r_k]$).

Combining all constraints and noticing that, in general, identify analytically degrees of freedom for a test problem is always not evident, the first step, the following a.a.c. is proposed:

$$\kappa_k = \frac{1}{n_y} \sum_{i=1}^{n_y} \left(\sup([r_k])_i - \inf([r_k])_i \right). \tag{4.6}$$

Simulation results in section 4.3.2 favored this choice by showing that it provides a small FAR and also satisfies all other requirements aforementioned.

Being not unique, the a.a.c. can be chosen differently by a scale of (4.6) which will be discuss in section 4.3.2.

Algorithms. The OUBIKF algorithm is originally developed for estimation with outputs $[\hat{x}_{k|k}]$ and $\mathcal{P}_{k|k}$. For fault detection purpose, this algorithm is used within Algorithm 7 so that $[\hat{x}_{k|k-1}]$ and $[P_{k|k-1}]$ are yielded as outputs of the former.

Algorithm 7 ADFC method to linear system

Initialization:
  $[\hat{x}_{0|0}], \mathcal{P}_{0|0}$, $[A], [B], [C], [D], [Q], [R]$, α, u_k, y_k, $k = 1, 2, ..., N$.
for $k = 1, 2, 3, ...N$ **do**
  Use OUBIKF (Algorithm 2 or 3) to get: $[\hat{x}_{k|k-1}]$, $[P_{k|k-1}]$.
  $[r_k] = y_k - [C][\hat{x}_{k|k-1}] - [D]u_k$
  $[S_k] = [C][P_{k|k-1}][C]^T + [R]$
  Find a_k using Theorem 5 s.t. : $S_+([S_k]) \preceq a_k I$.
  $U_k = \sup\{\text{abs}([r_k]^T[r_k]/a_k)\}$
  $\kappa_k = \text{mean}\{\sup([r_k]) - \inf([r_k])\}$
  Find δ_k s.t.: $\mathbb{P}(\chi^2(\kappa_k n_y) > \delta_k) = \alpha$.
  Detection signal : $\pi_k = \mathbb{I}(U_k > \delta_k)$.
end for

(*): $\mathbb{I}(x)$ is the indicator function which equal to 1 if the conditions x are true and vanishes otherwise.

4.3.2 Application

Consider again the Bicycle vehicle model (2.22) presented in Chapter 2. Recall that the simulation presented in this section is a contribution of (Lu et al., 2021) which had been developed before the optimal version of OUBIKF (Algorithm 3) was investigated. Therefore, the Beta version of OUBIKF (Algorithm 2) was applied to the ADFC method for fault detection in this section. This fact does not change the methodology of the ADFC method and it is interesting to see that, in the next section, ADFC method can be applied with the RLBPF to deal with nonlinear system.

Simulation procedure

A discretization with a sampling time $T = 0.05s$ is applied to the Bicycle vehicle model (2.22) to get (non interval and independent of time instant k) matrices A, B, C, D according to equations of the dynamical system. Then, interval matrices $[A]$, $[B]$, $[C]$, $[D]$ are generated in such a way that $M = \text{mid}([M])$ and the radii $\text{rad}([M])$ are chosen at random in $[0\,, \max\_\text{rad}]$ for $M = A, B, C, D$ and $\max\_\text{rad} = 0.5$. The covariance matrices $[Q]$ and $[R]$ are generated in the same way, their diagonal elements being intervals of positive real numbers.

Variable simulation. Inputs u_k's are simulated according to a dynamic change for $N = 864$ iterations (Fig.2.7). The initial state is chosen at $x_0 = (0, 0)^T$. At each step $k = 1 : N$, generate A_k, B_k, C_k, D_k, Q_k, R_k according to uniform distribution in corresponding interval matrices and so that Q_k

and R_k are symmetric positive semi-definite. Then $w_k \sim \mathcal{N}(0, Q_k)$ and $v_k \sim \mathcal{N}(0, R_k)$ are simulated. Finally, variable sequences $\{x_k\}$, $\{y_k\}$ are computed according to the dynamical system.

Fault generation: Sensor faults are generated in terms of bias vector $b_k \in \mathbb{R}^{n_y}$ added to y_k. Let $b, b' \in \mathbb{R}$ be constant fault values. Following types of error can be treated:

- Type 1: $b_k = b.\mathbf{1}$ where $\mathbf{1}$ is the all-ones vector in $\mathbb{R}^{n_y}$.
- Type 2: $b_k = b.\mathbf{e_j}$ where $\mathbf{e_j}$ is the j-th standard unit vector for some $j \in \{1, ..., n_y\}$.
- Type 3: $b_k = b.\mathbf{e_j} + b'.\mathbf{e_{j'}}$, with $j, j' \in \{1, ..., n_y\}$, $j \neq j'$.

The error terms are added to y_k for all k in a range $\mathcal{R}$ with length l, i.e. $k \in \mathcal{R} = r : r + l - 1$ for some r in $1 : N - l + 1$. Each sequence of y_k's components, e.g. $\{y_{1i}, y_{2i}, ..., y_{Ni}\}$ for some $i = 1, ..., n_y$, is called a *chain*. So, the errors occurred on multiple chains of y_k (and in the range $\mathcal{R}$) in type 1 and type 3 and only on single chain j in type 2. Moreover, in type 3, two errors with different values occur on two distinct chains.

Fault detection. Apply Algorithm 7 for N steps. The following choices are applied inside the algorithm: starting point $[\hat{x}_0] = ([-0.5, 0.5], [-0.5, 0.5])^T$, initial error covariance bound $\mathcal{P}_{0|0} = \max\{\text{diag}(\sup([Q]))\}I = 0.4412\, I$, $p = 3$, upper bounds $\omega_k I$ of any set $S_+([M])$ identified by $\omega_k = \|\text{Max}\|_F$ (Frobenius norm) where the Max matrix is defined in (2.2).

Adjusted fault detection. Use a window size $w = 5$.

Simulation results

Comparison. In this part, comparisons between the ADFC method and two others proposed respectively in (Tran, 2017) (method A) and (Raka and Combastel, 2013) (method B) are provided with concrete cases. The error range is between the two vertical black lines.

The method A uses the statistic $T_k = \inf([r_k]^T \mathcal{S}_k^{-1} [r_k])$, $S_+([S_k]) \preceq \mathcal{S}_k$ with the decision rule: a fault is detected if $T_k > \delta$ where the threshold δ defined by $\mathbb{P}(\chi^2(W n_y) > \delta) = \alpha$. The first disadvantage of this method is that interval computation can let T_k be negative, consequently no fault is detected as illustrated in Fig.4.2 according to the Bicycle vehicle model simulation. The second disadvantage is that a windows size W is arbitrarily chosen. An example (**E**) can be built to illustrate this method works quite well (Fig.4.3) in which T_k is non negative, but then the second disadvantage is still critical: another choice of W leads to another detection result. The ADFC method in this case still provides an accurate fault detection (Fig.4.4).

Consider again the Bicycle vehicle model. A result of the detection for type 1 of error with fault value $b = 20$ is shown in Fig. 4.5 and 4.6 using the

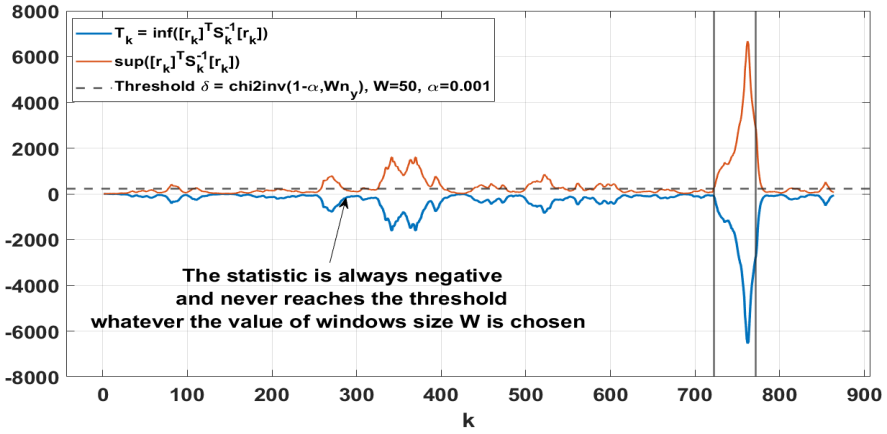

Figure 4.2 – Method A - Fault detection to Bicycle vehicle model.

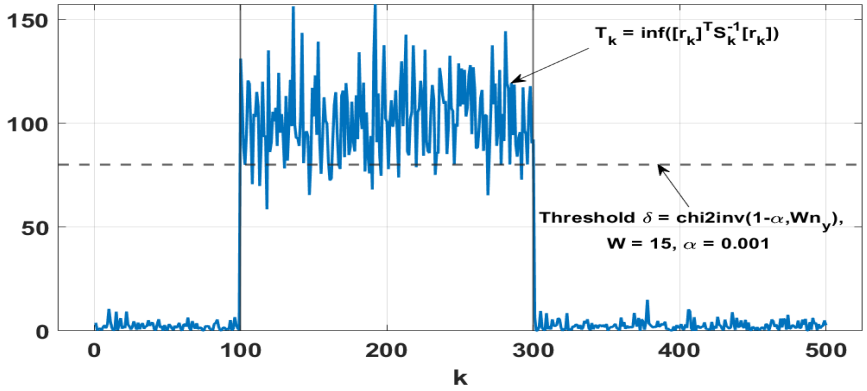

Figure 4.3 – Method A - Example (**E**) with $b = 10$ for type 1 of error.

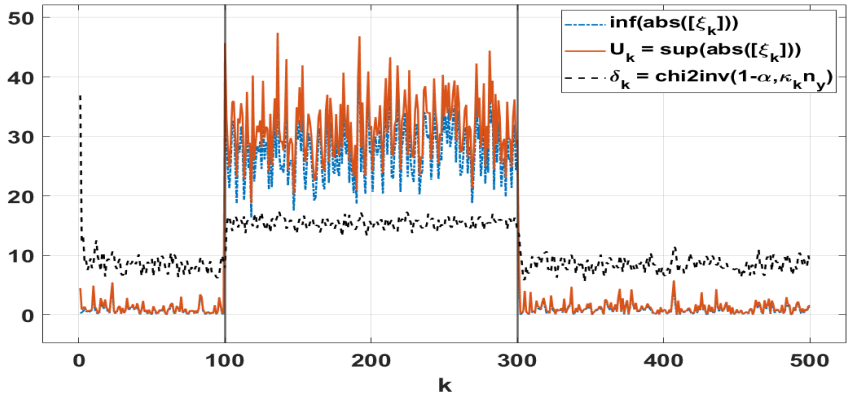

Figure 4.4 – ADFC method - Example (**E**) with $b = 10$ for type 1 of error.

ADFC method. The detection signals are very well determined.

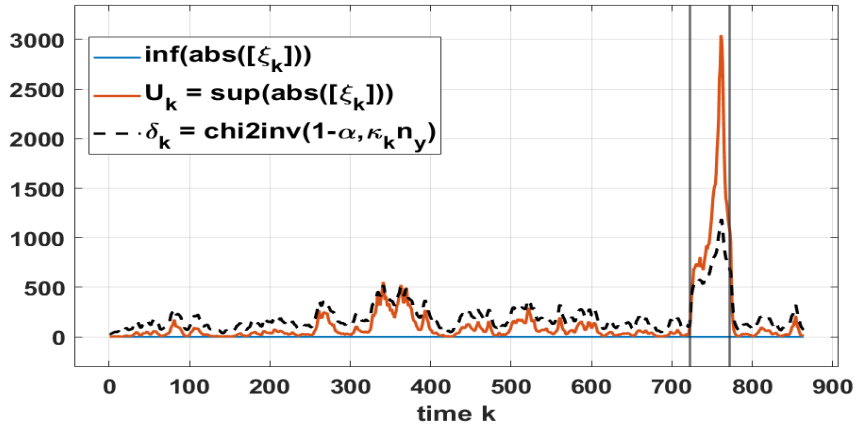

Figure 4.5 – ADFC method - Fault detection to Bicycle vehicle model.

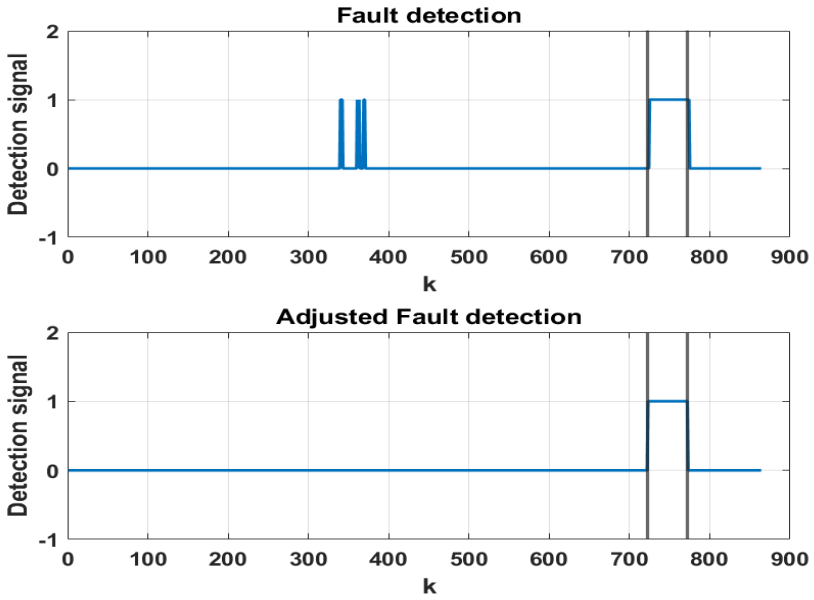

Figure 4.6 – ADFC method - Detection signals for Bicycle vehicle model

The method B is also an adaptive method. It consists in applying interval observer for a linear continuous time dynamic system with additive and multiplicative disturbances to compute adaptively upper bounds (ub_t) and lower bounds (lb_t) of residuals r_t, and the fault detection rule is that a fault is detected if $0 \notin [\mathrm{lb}_t, \mathrm{ub}_t]$. Fig.4.7-4.8 present the simulation of this method applying to Bicycle vehicle model with an as similar as possible setting with that used for ADFC method resulting in Fig.4.5-4.6. The setting is that: 1-dimension multiplicative and additive disturbances ($q = 1, \delta_t =$

$d_t = \sin(2\pi t))$ are used, a bias sensor fault with value $b = 20$ is added to all chains of measurements y_t in a time range [te1, te2]. This error range is determined as follow: total time of simulation is $[t_0, t_f]$ which corresponds to N discrete time steps in Fig.4.5, error range [te1, te2] corresponds to discrete range $[722 : 772]$ by calculating: te1 $= 722.t_f/N$, te2 $= 772.tf/N$, $t_0 = 0$.

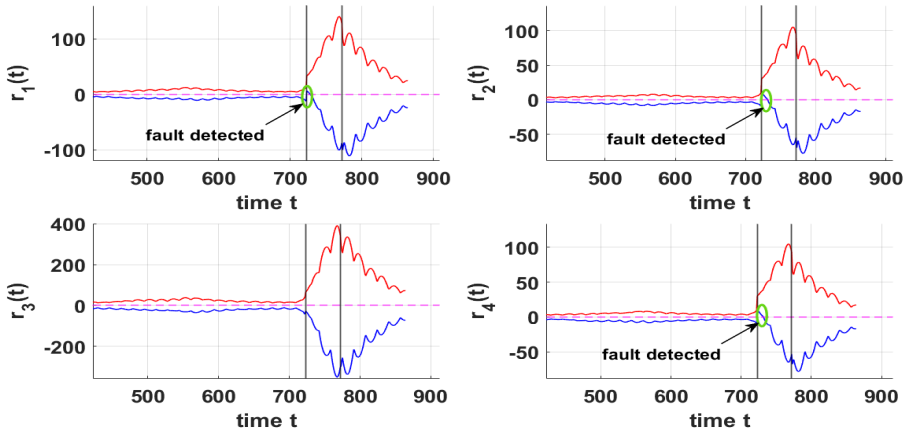

Figure 4.7 – Method B - Fault detection to Bicycle vehicle model.

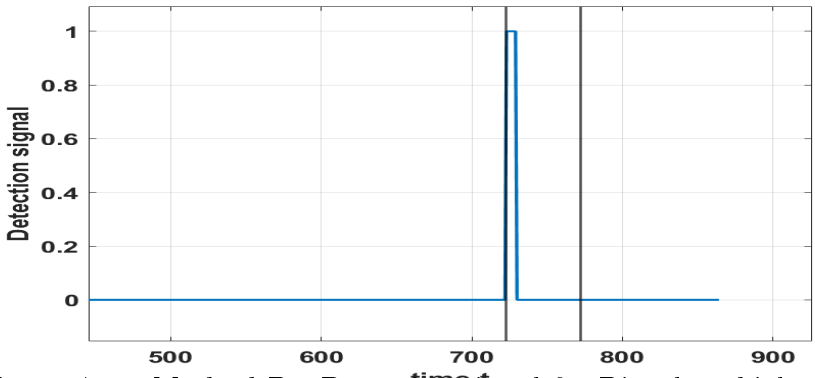

Figure 4.8 – Method B - Detection signal for Bicycle vehicle model.

Some remarks can be pointed out:

- method B takes the fault detection chain by chain;
- in the error region of 50 time steps, method B detects almost all starts of faults (e.g. about the first 10 time steps as shown in Fig.4.7), except on chain 3, and after that the detection is degenerated and no more accurate;
- to compare with the ADFC method, detection signals of method B in all chains are combined in one in such a way that new detection

signal is 1 if there is any detection signal in any chain getting value 1 (Fig.4.8).

Then, we redo the simulation in 100 times with error range $\mathcal{R}$ chosen randomly as described by the general implementation in the next and obtain the comparison results shown in Table 4.1.

	DR%	NDR%	FAR%	EFF%
Method B	5.90	94.10	4.31	1.59
ADFC method	98.56	1.44	5.67	92.89

Table 4.1 – ADFC method versus Method B.

Further investigation simulation. Now, in order to survey, using ADFC method, how well the detection is when influencing factors are changed (e.g. fault value b, error range $\mathcal{R}$ and simulated variable y_k), three simulation scenarios are implemented using indicators introduced in Section 4.2.2. Scenarios 1 and 2 will be treated with the type 1 of error, while the scenario 3 will be implemented with all three types of error.

General implementation. For different types of error of bias vector b_k, let b and b' take values respectively in discrete sets E and E'. The error range $\mathcal{R}$ has length $l = 50$. According to each scenario, type of error and value of (b, b'), $L = 100$ times of fault detections are implemented. Indicators are computed for each of L simulation times and their means are yielded afterward as representative values that will be shown in result tables.

Remark 25. Let $\tau_k = \max\{b_k\}/\text{Max\_width}$ where $\max\{b_k\}$ is the maximum among the b_k's components and Max_width is the maximum width of the diagonal elements of $[Q]$ and $[R]$. This quantity gives an idea of how large is the maximum of the actual fault value with respect to some known quantity causing the fault and propagating according to the dynamic system, that is the maximum covariance of noises. □

Remark 26. The comments in next parts hold for (b, b') values belonging to the considered sets outside of which related comments might be solely intuitive deductions. □

Scenario 1. Fix variable simulation $\{y_k\}_{k=1:N}$, for each fault value b in $\{0 : 5 : 30\}$, choose randomly error range $\mathcal{R}$ and do L times error generations. This scenario helps us to consider the method performance in terms of fault values b and the positions at which errors occur (in $\mathcal{R}$) w.r.t. a given measurement sample $\{y_k\}_{k=1:N}$.

Table 4.2 shows that DR has ascending trend as well as b increases while FAR is rather stable in $[1.0\,,\,1.5](\%)$ with mean 1.25%. This means that the

b	τ	DR%	NDR%	FAR%	EFF%
0	0	1.26	98.74	1.03	0.23
5	13.8	5.34	94.66	1.04	4.30
10	27.5	21.12	78.88	1.04	20.08
15	**41.3**	**67.28**	**32.72**	**1.17**	**66.11**
20	55.0	94.94	5.06	1.26	93.68
25	68.8	98.64	1.36	1.36	97.28
30	82.5	99.84	0.16	1.42	98.42

Table 4.2 – Fault detection for scenario 1 and type 1 error.

larger the fault value b, the better the fault detection procedure is performed and, conversely, the b change hardly affects the false alarm rate FAR. This also means that the current choice of a.a.c. κ_k is appropriate for a fault detection eliminating well false alarms and dismissing almost all non clear signs of error existence (a prudent fault detection). For different purposes of fault detection, κ_k can be adjusted (see discussions in next part).

Seeing more, EFF represents the effectiveness of the fault detection procedure taking into account both DR and FAR. It has also ascending trend according to b. Starting at $b = 15$ ($\approx 41\times$ Max_width) EFF begins to achieve remarkable value (66.11%).

b	τ	DR%	NDR%	FAR%	EFF%
0	0	0	100	0	0
5	13.8	3.30	96.70	0.01	3.29
10	27.5	17.70	82.30	0.02	17.68
15	**41.3**	**63.54**	**36.46**	**0.05**	**63.49**
20	55.0	96.36	3.64	0.06	96.30
25	68.8	99.96	0.04	0.15	99.81
30	82.5	100	0	0.38	99.62

Table 4.3 – Adjusted fault detection for scenario 1 and type 1 error.

Table 4.3 shows that the adjustment procedure eliminates almost all FAR indexes (at least 73% comparing to those in Table 4.2). Additionally, this procedure yields a positive effect with large fault value ($b > 15$) and a negative effect otherwise for EFF indexes. In application, if we know that measurement fault value often reaches a threshold (which depends on used sensors), e.g. 15 units in the Bicycle application, then adjustment procedure is recommended and vice versa. Thus, the adjustment procedure and the choice of a.a.c. κ_k are two tunning factors to make the method suitable.

Scenario 2. Fix error range $\mathcal{R}$. For each fault value b in $\{0:5:30\}$, do L times variable simulations to get measurements y_k. This scenario aims to show the effects of different measurement samples $\{y_k\}_s$ $(k = 1:N, s = 1:L)$ on the fault detection procedure for a given error range $\mathcal{R}$. Specifically, these effects come from random noises existing inside of y_k since the later is a function of $\{x_0, w_1 : w_k, v_k\}$.

b	τ	DR%	NDR%	FAR%	EFF%
0	0	3.34	96.66	1.90	1.44
5	13.8	3.08	96.92	2.36	0.72
10	27.5	19.24	80.76	2.41	16.83
15	**41.3**	**82.48**	**17.52**	**2.18**	**80.30**
20	55.0	94.74	5.26	2.48	92.26
25	68.8	98.66	1.34	2.28	96.38
30	82.5	99.88	0.12	2.37	97.51

Table 4.4 – Fault detection for scenario 2 and type 1 error.

b	τ	DR%	NDR%	FAR%	EFF%
0	0	2.8	97.20	1.44	1.36
5	13.8	2.16	97.84	1.84	0.32
10	27.5	14.06	85.94	1.97	12.09
15	**41.3**	**82.42**	**17.58**	**1.62**	**80.80**
20	55.0	96.96	3.04	1.82	95.14
25	68.8	99.8	0.20	1.71	98.10
30	82.5	100	0	1.89	98.11

Table 4.5 – Adjusted fault detection for scenario 2 and type 1 error.

In Table 4.4, DR and EFF indexes are not necessarily increasing functions w.r.t. b but their main trends are always ascending. The FAR index is also stable in $[1.9\,, 2.5](\%)$ with mean 2.2%. In addition, in comparison with the one in Table 4.2 (FAR $\in$ [1.0, 1.5](%)), we see that FAR is rather greater in scenario 2 than in scenario 1. This means that FAR is more affected by random noises than by the position of error range. The adjustment procedure eliminates more than 18% of FAR indexes comparing Table 4.5 and Table 4.4. It has also positive effect or negative effect for EFF index according to the fault value b being greater or smaller than 15.

Scenario 3. For each value of b, choose randomly error range $\mathcal{R}$ and do L times of variable simulations. This scenario combines the two previous scenarios and will be implemented with three types of error.

(1) Type 1 error: The sensor faults come to all chains of $\{y_k\}$, $b_k = b.\mathbf{1}$ and $\tau_k = \tau = b/\text{Max\_width}$ with $b \in \{0 : 5 : 30\}$.

b	τ	DR%	NDR%	FAR%	EFF%
0	0	1.88	98.12	2.02	-0.14
5	13.8	5.1	94.9	2.02	3.08
10	27.5	17.36	82.64	2.16	15.20
15	**41.3**	**74.06**	**25.94**	**2.53**	**71.53**
20	55.0	94.08	5.92	2.36	91.72
25	68.8	98.26	1.74	2.84	95.42
30	82.5	99.82	0.18	2.50	97.32

Table 4.6 – Fault detection for scenario 3 and type 1 error.

b	τ	DR%	NDR%	FAR%	EFF%
0	0	1.48	98.52	1.53	-0.05
5	13.8	4.54	95.46	1.59	2.95
10	27.5	13.88	86.12	1.69	12.19
15	**41.3**	**72.38**	**27.62**	**1.97**	**70.41**
20	55.0	95.38	4.62	1.75	93.63
25	68.8	99.62	0.38	2.13	97.49
30	82.5	99.96	0.04	1.96	98.00

Table 4.7 – Adjusted fault detection for scenario 3 and type 1 error.

It can be pointed out similar comments as those of the two previous scenarios with this type 1 error although values in result tables must be different. In addition, the negative value for EFF at $b = 0$ can be explained by the fact that, in this case, the fault detection procedure not only provides no efficiency gains, but rather a loss.

Another notice is that the FAR does not vanish. Therefore EFF never reaches 100% although the fault value b can be more higher (than 30) and DR can reach 100%.

The adjustment procedure eliminates at least 21% of FAR indexes comparing Table 4.7 and Table 4.6.

(2) Type 2 error: In this simulation, the sensor faults only occur in one chain of $\{y_k\}$, $b_k = b.\mathbf{e_j}$ and $\tau_k = \tau = b/\text{Max\_width}$ with $b \in \{0 : 5 : 60\}$. The general implementation for scenario 3 is always respected noticing that the chain j on which the faults occur is chosen randomly as well at each of L times of variable simulations. This situation corresponds to a single faulty sensor and normally the case where all sensors

are damaged at the same time is less frequent. This situation is also necessary for fault isolation in a further phase.

b	τ	DR%	NDR%	FAR%	EFF%
0	0	4.08	95.92	1.96	2.12
5	13.8	4.54	95.46	2.48	2.06
10	27.5	4.90	95.10	1.82	3.08
15	41.3	11.42	88.58	2.12	9.30
20	55.0	27.92	72.08	2.24	25.7
25	68.8	42.90	57.10	2.40	40.50
30	82.5	51.96	48.04	2.36	49.60
35	**96.3**	**74.72**	**25.28**	**2.25**	**72.50**
40	110.0	77.70	22.30	2.10	75.60
45	123.8	84.68	15.32	2.28	82.40
50	137.5	90.24	9.76	3.49	86.75
55	151.3	96.58	3.42	2.71	93.87
60	165.0	98.86	1.14	2.50	96.36

Table 4.8 – Fault detection for scenario 3 and type 2 error.

b	τ	DR%	NDR%	FAR%	EFF%
0	0	3.28	96.72	1.46	1.82
5	13.8	3.74	96.26	2.01	1.73
10	27.5	3.58	96.42	1.44	2.14
15	41.3	10.26	89.74	1.66	8.60
20	55.0	26.50	73.50	1.74	24.76
25	68.8	42.26	57.74	1.88	40.38
30	82.5	50.40	49.60	1.80	48.60
35	**96.3**	**75.58**	**24.42**	**1.64**	**73.94**
40	110.0	78.08	21.92	1.52	76.56
45	123.8	85.22	14.78	1.72	83.50
50	137.5	89.88	10.12	2.91	86.98
55	151.3	96.24	3.76	2.10	94.14
60	165.0	98.18	1.82	2.05	96.13

Table 4.9 – Adjusted fault detection for scenario 3 and type 2 error.

Consider Table 4.8. Since the faults occur only on one chain of $\{y_k\}$, it is obvious that, for each corresponding value b, the EFF indexes of Table 4.8 are lower than those of Table 4.6. Until $b = 35$ ($\approx 96\times$Max_width) EFF reaches a remarkable value 72.50%. This is also the threshold

beyond which the adjustment procedure has a positive effect on EFF. FAR remains stable in a modest range ($[1.8\,, 3.5](\%)$). In addition, from $b = 60$ the EFF almost reaches its maximum value (96.36%) (the greatest value of EFF will be around 97% due to the existence of FAR).

(3) Type 3 error. Let $b = 10.m$ and $b' = 10.(m+1)$ for $m = 1, 2, 3$, then $\tau_k = \tau = \max\{b, b'\}/\text{Max\_width}$. The chains j and j' at which the faults occur are chosen randomly as well as the error range $\mathcal{R}$ at each of L times of variable simulations. This setting, while still being of the multiple error type, can represent an intermediate situation between the settings of type 1 and type 2 error previously presented.

(b, b')	τ	DR%	NDR%	FAR%	EFF%
(10,20)	55.0	44.72	55.28	6.53	38.19
(20,30)	82.5	81.34	18.66	6.57	74.77
(30,40)	110.0	99.46	0.54	6.82	92.64

Table 4.10 – Fault detection for type 3 error.

(b, b')	τ	DR%	NDR%	FAR%	EFF%
(10,20)	55.0	42.84	57.16	5.29	37.55
(20,30)	82.5	79.38	20.62	5.33	74.05
(30,40)	110.0	98.82	1.18	5.59	93.23

Table 4.11 – Adjusted fault detection for type 3 error.

The following remarks are valid for all cases already simulated above.

Remark 27. FAR does not vanish even in the fault free case ($b = 0$). This fact implies that there are other reasons (than fault) causing FAR. Actually, in this case, the error range degenerates to length 0, all 1-value detection signals are false detected signals, DR and NDR are not defined and FAR must be recomputed, e.g. according the first row of Table 4.6: $\text{FAR} = [2.02 \times (N - 50) + 1.88 \times 50]/N \approx 2.01$. However, we can think that b has a very small (non zero) value and thus results remain unchanged. □

Remark 28. The factors causing FAR are multiple. Two of these factors that differ from one simulation to another are random noises and random error ranges, which can therefore be called specific factors. Some other factors that exist for all the simulations, and which can therefore be called general factors, are: the model performance (how well the model describes the dynamics of

the vehicle), the conservatism of interval computations, the lack of knowledge on the exact coefficient matrices A_k, B_k, C_k,... and the performance of the χ^2-statistic test (with α significance level). □

Remark 29. Let define the *magnitude of fault* (MF) be the maximum of absolute values of b_k's components. Then, DR and EFF have ascending trends according to the MF while FAR is rather stable in some range with positive values. □

Remark 30. There is a threshold for good/bad result of EFF and for positive/negative effect of the adjustment procedure to the EFF, e.g.: $b \geq 15$ for type 1, $b \geq 35$ in type 2, $\max\{b, b'\} > 30$ in type 3. □

Remark 31. The adjustment procedure and the choice of a.a.c. κ_k are two tuning factors for an appropriate fault detection. □

Discussion

For a further discussion, other a.a.c κ_k can be chosen to improve EFF index for small values b, e.g. $b < 35$ and according to the type 2 error framework. To this end, and since the choice (4.6) of κ_k provides rather good results, it is proposed to use some scales of κ_k. The a.a.c. now becomes $\tilde{\kappa}_k = \lambda_k \kappa_k$, where $\lambda_k > 0$ is a scale parameter.

For simple experiments, the λ_k's are chosen identically in $\{0.7, 0.3\}$ for all $k \geq 1$, briefly named λ. The simulation results are shown in Tables 4.12-4.13. Only the type 2 error with the adjustment procedure is simulated. Both cases of λ are applied for common data samples when error range is randomly changed and L times of variable simulations are executed.

b	τ	DR%	NDR%	FAR%	EFF%
0	0	5.84	94.16	4.39	1.45
5	13.8	5.88	94.12	4.62	1.26
10	27.5	13.30	86.70	4.58	8.72
15	41.3	29.52	70.48	4.42	25.10
20	55.0	53.56	46.44	5.01	48.55
25	68.8	68.10	31.90	3.99	64.11
30	82.5	77.94	22.06	3.84	74.10
35	96.3	85.08	14.92	4.43	80.65
40	110.0	94.12	5.88	4.71	89.41

Table 4.12 – Adjusted fault detection for type 2 error using a.a.c. κ_k with scale parameter $\lambda = 0.7$

.

From Tables 4.12 and 4.13, it is noticed that FAR increases when the parameter scale λ decreases. EFF rises up for small values b (< 35) depending on the decrease of λ and inversely for EFF with larger values b.

When $\lambda = 0.7$ (Table 4.12), the values of the FAR index disperse in $[3.8\,, 5.1](\%)$, and thus EFF $=$ DR $-$ FAR does not overpass 96.2% even if DR reaches its maximum value (100%). In addition, compared to Table 4.9 ($\lambda = 1$), EFF in Table 4.12 increases considerably for many fault values $b < 35$ (starting at 10) and begins to achieve a remarkable rate (64.11%) starting at $b = 25$. However, the EFF does not increase in the case $b = 5$; this is due to the fact that Tables 4.9 and 4.12 display the simulation results of the different samples. The case $b = 0$ is not comparable (see Remark 27).

b	τ	DR%	NDR%	FAR%	EFF%
0	0	24.64	75.36	25.14	-0.50
5	13.8	37.12	62.88	24.52	12.60
10	27.5	61.84	38.16	25.12	36.72
15	41.6	84.50	15.50	24.08	60.42
20	55.0	95.46	4.54	24.72	70.74
25	68.8	99.42	0.58	24.83	74.59
30	82.5	99.82	0.18	24.06	75.76
35	96.3	99.84	0.16	23.96	75.88
40	110.0	99.92	0.08	25.25	74.67

Table 4.13 – Adjusted fault detection for type 2 error using a.a.c. κ_k with scale parameter $\lambda = 0.3$

When $\lambda = 0.3$ (Table 4.13), the FAR range is $[23.9\,, 25.3](\%)$, EFF is never beyond 76.1% and increases again for all $b < 35$ (unless $b = 0$) w.r.t Table 4.12.

In summary, depending on the applications requiring a low FAR or a high EFF for a fine fault detection (detecting error with small fault value), different a.a.c. $\tilde{\kappa}_k$'s are chosen suitably thanks to the scale parameter $\lambda_k > 0$. We also notice that a non constant (adaptive) scale parameter λ_k can also be applied. Whether or not λ_k could be chosen in an optimal way, under some criteria, can be issues in a future work.

Furthermore, the modified EFF index proposed by EFF $= c_1 \times$ DR $-$ $c_2 \times$ FAR with c_1, c_2 two constants in $[0, 1]$ can be applied to control the importance of the two indexes DR and FAR so that a compromise between them is achieved.

4.3.3 Fault detection based on ADFC method and RLBPF for nonlinear system

In this section, the ADFC method is extended to a more general framework, including the nonlinear dynamical system. It can be applied with any interval filter, e.g. RLBPF, requiring only that the measurement dynamic is noisy with additive Gaussian noises.

Consider the system (4.2) with additive measurement noises :

$$(\boldsymbol{\Sigma}) \ : \ \begin{cases} x_k = f_k(x_{k-1}, u_k, w_k) \ , \\ y_k = h_k(x_k, u_k) + v_k + f_k^s \ , \end{cases} \qquad k \in \mathbb{N}^*,$$

where $x_k \in \mathbb{R}^{n_x}$ and $y_k \in \mathbb{R}^{n_y}$ represent state variables and measures respectively, $u_k \in \mathbb{R}^{n_u}$ inputs, $w_k \in \mathbb{R}^{n_x}$ state disturbances/noises, $v_k \in \mathbb{R}^{n_y}$ measurement noises, $f_k^s \in \mathbb{R}^{n_y}$ additive sensor fault vectors.

Assumptions (A2):

- Given $[u_k], [w_k], [y_k], [\mu_k], [\Sigma_k], f_k, h_k, k = 1, 2, 3, ...$ and $[x_0] \ni x_0$. Note that the measurements are given as intervals $[y_k]$ due to the sensor precision and thus assume further that $[y_k] \ni y_k$ with probability 1.
- $u_k \in [u_k]$, $w_k \in [w_k]$, $y_k \in [y_k]$.
- $v_k \sim \mathcal{N}(\mu_k, \Sigma_k)$ where $\mu_k \in [\mu_k]$, $\Sigma_k \in [\Sigma_k]$.

Now, we consider the following straightforward but useful property for the development in the next. Its proof is straightforward and hence is omitted.

Lemma 7. *Consider system* $(\boldsymbol{\Sigma})$ *with assumptions (**A2**) and* $f_k^s = 0$ *for all* $k \in \mathbb{N}^*$. *Let* $r_k = y_k - h_k(x_k, u_k) - \mu_k$ *be the residual. Then, for all* $k \geq 1$*:*

(i) $y_k \sim \mathcal{N}(m_k, \Sigma_k)$ *where* $m_k = h_k(x_k, u_k) + \mu_k$.
(ii) $r_k \sim \mathcal{N}(0, \Sigma_k)$ *where* $\Sigma_k \in [\Sigma_k]$.

Then, the extension of the ADFC method is based on the following lemma.

Lemma 8. *Consider system* $(\boldsymbol{\Sigma})$ *with assumptions (**A2**). For any interval filter providing interval estimates* $[\hat{x}_{k|k}]$, *assuming that* $[\hat{x}_{k|k}] \ni x_k$ *at every* $k \geq 1$, *then:*

(i) $[\hat{x}_{k+1|k}] \triangleq [f_{k+1}]\Big([\hat{x}_{k|k}], [u_{k+1}], [w_{k+1}]\Big)$ *contains the real state* x_{k+1},
(ii) $[\hat{r}_{k+1}] \triangleq [y_{k+1}] - [h_{k+1}]\Big([\hat{x}_{k+1|k}], [u_{k+1}]\Big) - [\mu_{k+1}]$ *contains the residual* $r_{k+1} \sim \mathcal{N}(0, \Sigma_{k+1})$ *with* $\Sigma_{k+1} \in [\Sigma_{k+1}]$ *with probability 1.*

Proof. The proof of the lemma bases on the inclusion function property in interval analysis, that is: $f(x) \in [f]([x]), \forall x \in [x]$. By assumption, $[\hat{x}_{k+1|k}] \ni f_{k+1}(x_k, u_{k+1}, w_{k+1}) = x_{k+1}$, $[h_{k+1}]([\hat{x}_{k+1|k}], [u_{k+1}]) \ni h_{k+1}(x_{k+1}, u_{k+1})$ and

$[\mu_{k+1}] \ni \mu_{k+1}$. Thus, $[\hat{r}_{k+1}] \ni r_{k+1}$ with certainty ensured by the inclusion function property. By assumptions, r_{k+1} is random and $[y_k] \ni y_k$ with probability 1, therefore $[\hat{r}_{k+1}]$ can be seen in general as random and we conclude probabilistically that the fact $[\hat{r}_{k+1}] \ni r_{k+1}$ holds true with probability 1. □

Remark 32. The notation $[\hat{x}_{k|k}]$, $k \geq 1$, denotes the interval estimate provided by the corresponding interval filter at the end of the time instant k, while $[\hat{x}_{k+1|k}]$ denotes the propagation box at the next iteration being apt to further estimation techniques to obtain $[\hat{x}_{k+1|k+1}]$. Note that $[\hat{x}_{k+1|k}]$ and $[\hat{x}_{k+1|k+1}]$ can be coincident depending on the filter used. □

Remark 33. The assumption $[\hat{x}_{k|k}] \ni x_k$ at every $k \geq 1$ is strong and related to the performance and convergence of the filter. Practically, an interval filter that has the $O(\%)$ measuring the percentage of $x_k \in [\hat{x}_{k|k}]$ greater than some level (e.g. 80% or 90%) can be suited to the fault detection procedure of the ADFC method, unless missed conditions may cause certain false alarms. □

In contrast, in previous sections, the residual term is determined by $r_k = y_k - h_k(\hat{x}_k, u_k) - \mu_k$ where $\hat{x}_k$ is an estimate of unknown real state x_k and h_k is a linear function. Thus, r_k can be computed explicitly at each time step as well as its distribution under the standard (SKF) assumptions. Consequently, with additional bounded uncertainties, the covariance of $[r_k]$ is also well determined as some calculable matrix $[S_k]$.

In a more general framework, estimates $\hat{x}_k$ might be unknown (or unused) and h_k might be complex (so, for instance, even if $\hat{x}_k$ is Gaussian then the distribution of $h_k(\hat{x}_k)$ might be not determined). Then, by considering $r_k = y_k - h_k(x_k, u_k) - \mu_k$, although it is unknown, we can compute $[\hat{r}_k]$ such that $[\hat{r}_k] \ni r_k \sim \mathcal{N}(0, \Sigma_k)$, $\Sigma_k \in [\Sigma_k]$ with probability 1 (Lemma 8). This implies that $[\hat{r}_k] \supset [\Sigma_k]$ and $[\hat{r}_k]$ must be greater considerably than $[\Sigma_k]$ in order to contain r_k with such a certainty. Thus, if we want to use the statistic $U_k = \sup\{\text{abs}([\hat{r}_k]^T[\hat{r}_k]/a_k)\}$ with some a_k such that $S_+([\Sigma_k]) \preceq a_k I$, then a_k must be a compromise between $[\hat{r}_k]$ and $[\Sigma_k]$, says a function of them $a_k = \phi_k([\hat{r}_k], [\Sigma_k])$. In the development of this section, a simple proposition for a_k is that

$$a_k = \lambda_1 * \text{mean}\{\text{width}([\hat{r}_k])\} \quad \text{so that} \quad S_+([\Sigma_k]) \preceq a_k I, \tag{4.7}$$

where $\lambda_1 \in (0, 1)$ is a scaling factor changing from application to application and the function mean$\{x\}$ provides the mean value of the vector x over its components.

Then, the ADFC method proposed in the previous section can be applied to the novel framework as shown in the following algorithm using scaling factors λ_1, λ_2 where the last one is discussed in the previous section.

Algorithm 8 ADFC method to nonlinear system

Initialization:
    α, λ_1, λ_2, $[\hat{x}_{0|0}] \equiv [x_0]$, f_k, h_k, $[u_k]$, $[w_k]$, $[y_k]$, $[\mu_k]$, $[\Sigma_k]$, $k = 1, 2, ..., N$.
for $k = 1, 2, 3, ...N$ **do**
    Use RLBPF (Algorithm 6) to get: $[\hat{x}_{k|k-1}]$
    $[\hat{r}_k] = [y_k] - [h_k]([\hat{x}_{k|k-1}], [u_k]) - [\mu_k]$
    $a_k = \lambda_1 * \text{mean}\{\text{width}([\hat{r}_k])\}$
    $U_k = \sup\{\text{abs}([r_k]^T[r_k]/a_k)\}$
    $\kappa_k = \text{mean}\{\sup([r_k]) - \inf([r_k])\}$
    Find δ_k s.t.: $\mathbb{P}(\chi^2(\lambda_2\kappa_k n_y) > \delta_k) = \alpha$.
    Detection signal : $\pi_k = \mathbb{I}(U_k > \delta_k)$.
end for

4.3.4 Application

Consider the Quarter vehicle model presented in Section 3.6. In this part, the simulation will perform in $N = 5000$ iteration steps while all other settings remain unchanged comparing to those of Section 3.6. For the fault detection purpose, a fault $b = 0.02$ is added to the simulated observed measurements in a range $\mathcal{R}$ with length $l = 200$ and the following choices are used: $\lambda_1 = 0.02$, $\lambda_2 = 10$, $\alpha = 0.03$.

A simulation result is figured out in Figures 4.9 - 4.11. In the error range, the residual deviate from 0 (Fig. 4.9) and most of the statistics U_k (blue line) passe over the adaptive thresholds δ_k (red line) (Fig. 4.10).

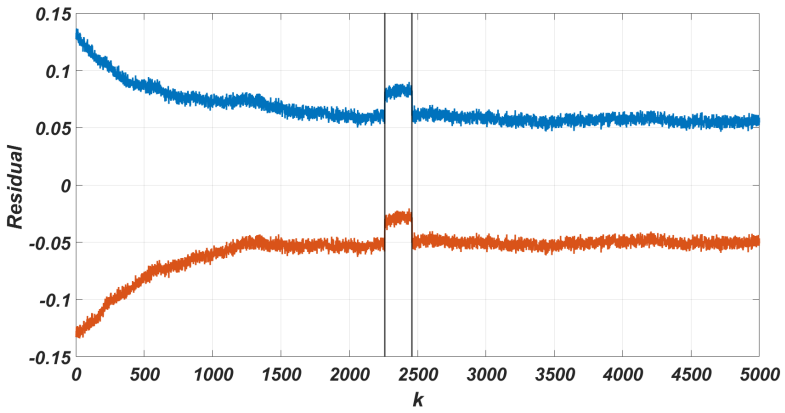

Figure 4.9 – ADFC method - Residual $[\hat{r}_k]$ for the Quarter vehicle model with sensor fault.

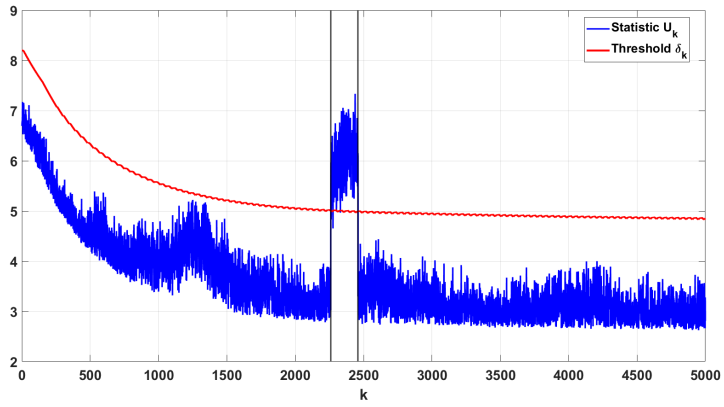

Figure 4.10 – ADFC method - Fault detection to the Quarter vehicle model.

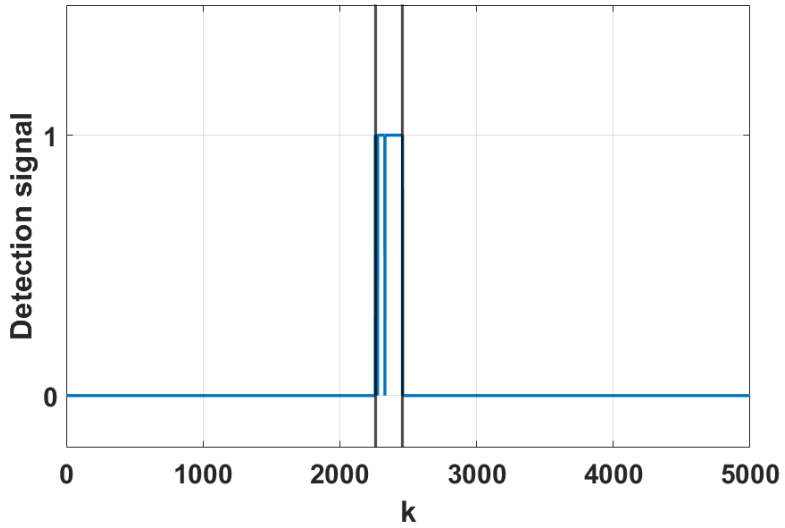

Figure 4.11 – ADFC method - Detection signal for Quarter vehicle model.

Then the fault detection procedure is replicated for $L = 100$ times where the error range $\mathcal{R}$ is chosen randomly and indicators DR, NDR, FAR, EFF are yielded as their corresponding means after L times of simulations (Table 4.14). For a fault value $b = 0.02$, the efficiency index (EFF) is about 66%.

b	DR(%)	NDR(%)	FAR(%)	EFF (%)
0.02	67.715	32.285	1.2762	66.439

Table 4.14 – ADFC method - Fault detection to Quarter vehicle model.

4.4 Conclusion and perspective

An adaptive approach to sensor fault detection applied to linear or nonlinear discrete time dynamical system is proposed. The approach combines OUBIKF or RLBPF with a new hypothesis testing method using χ^2-statistics with adaptive degrees of freedom. Theoretical framework is developed. Numerous tuning techniques are also presented, in particular the choice of a_k with a scaling factor (in nonlinear case), the choice of a.a.c. κ_k with parameter scale λ_k, the adjustment procedure for fault detection and the modified EFF index. A great flexibility of adjusting these factors makes the approach highly fitted to multiple applications.

In the chapter, simulation applications are presented based on the Bicycle vehicle model (linear case) and the Suspension model (nonlinear case). The simulation results show that the ADFC method has worked quite well in either cases and its performance depends on fault magnitudes and scenarios under consideration.

The ADFC method is developed however in the framework of (additive) sensor fault systems. Extend this method to deal with other kinds of fault (e.g. actuator faults) and with fault identification is a perspective of our future research.

Chapter 5

Active Fault Diagnosis based on Adaptive Degrees of Freedom χ^2-statistic method

5.1 Introduction

Fault diagnosis is becoming nowadays an increasingly indispensable functionality for modern systems. Including the fault detection as a non negligible task, fault diagnosis is characterized by its ability of fault isolation and identification with which the system can have adaptive actions in time.

In the field of model-based fault diagnosis, two approaches are known: *passive* and *active fault diagnosis* (AFD). The passive approach has been introduced very earlier in 1970's from (Beard, 1971; Jones, 1973; Mehra and Peschon, 1971) and investigated by a massive researches since then. It is referred to (Chen and Patton, 1999) and (Ding, 2013) for a panorama of the passive approach. In contrast, the active approach has been developed more recently and remains a new and dynamic research branch. (Chen and Patton, 1999), a well-known textbook on model-based fault diagnosis for dynamical system, does not yet mention the notion of active fault diagnosis. (Campbell and Nikoukhah, 2004), a textbook specified in auxiliary signal design for failure detection, does use active failure detection as key terminology. H. Niemann et al. in their papers since 2005 ((Niemann, 2005), (Niemann and Poulsen, 2005), (Stoustrup and Niemann, 2010), (Niemann and Poulsen, 2014),...) use the AFD terminology. To the best of our knowledge until now, (Punčochář and Škach, 2018) is a rather complete and recent survey about the active approach devoting to use this terminology.

A key idea of AFD is to use *auxiliary input signals* that are injected into

the monitored system in order to improve the quality of decision making stage of the fault diagnosis. This technique was investigated in several researches around 1990, well before the publication of the AFD terminology (Punčochář and Škach, 2018). Also, according to this survey, AFD approach can be classified into groups based on different relevant features, e.g.: deterministic (norm bounded)/probabilistic (based on uncertainty description), fixed/variable finite time interval or infinite time interval (based on the length of time interval in which auxiliary signals are designed). A list up to date and non exhaustive of related researches can be found in (Punčochář and Škach, 2018) and (Tan et al., 2021).

The relevant issue of AFD approach in the literature is that the injection of auxiliary signals into the monitored system disturbs the outputs to be controlled both in the fault-free case as well as in the faulty case (Niemann and Poulsen, 2014). It should be ensured that auxiliary signals do not drive the monitored system out of desired control performance specifications (Punčochář and Škach, 2018). Therefore, there is a trade-off between the fault diagnosis quality improvement and a minimal disturbance of the controlled outputs.

In this chapter, we present a novel AFD method based on the Adaptive degrees of freedom χ^2-statistic (ADFC) method already developed in Chapter 4. The novel one also uses designed auxiliary signals to enhance fault detection performance as well as provide the ability of localization and estimation of the detected fault. It uses the ADFC method as its main fault detector, called *ADFC detector*, and therefore deals with a dynamical system with mixed uncertainties. More precisely, in this development, a linear discrete time system with mixed uncertainties is concerned and sensor fault context is treated. The most relevant difference is that, in the novel method, auxiliary signals are not injected into the monitored system but provided only to the diagnoser whenever a fault is detected. Thanks to characteristics of ADFC detector, using these auxiliary signals, the diagnoser can decide whether the detected signal is a false alarm or not. In the case of positive confirmation, the diagnoser localizes and estimates the detected fault. Then, the estimated fault is returned backwardly to the diagnoser to compensate for the actual fault effect to the next iteration. Beside the fault identification and fault estimation functionalities, the diagnoser enhances the fault detection by:

(1) reducing false alarms,

(2) compensating for the fault effect to the next diagnosis,

(3) tuning the ADFC detector (by its tuning parametes) to increase reasonably its detection rate.

Furthermore, since auxiliary signals are not injected to the monitored sys-

tems, the undesired system disturbance due to auxiliary signals is avoided. Once an actual fault is correctly diagnosed, what action will be made regarding the diagnostic information is a matter of a separate scheme.

The chapter is organized as follows. In Section 5.2, the considered problem is formulated. Section 5.3 presents the main contribution which is the proposed AFD method. Section 5.4 applies the method to the Bicycle vehicle application. Finally, Section 5.5 provides the chapter conclusion with some discussions and perspectives.

5.2 Problem formulation

In this section, a description of the problem formulation for the fault diagnosis purpose is addressed. Based on that, objectives, scope and methodology of a resolution can be determined. In general, a standard fault diagnosis resolution addresses both the fault detection and isolation (FDI) and the fault estimation (FE).

The system under consideration is the same of (4.1) presented in Chapter 4. The assumptions **(A1)** are also considered in this development. Furthermore, additional assumptions are also required in the following.

Assumption F1. The fault vector is of the form of single fault, that is $f_k^b = b.\mathbf{e_j} \equiv f_{k,j}^b$ where $b \in \mathbb{R}$ and $\mathbf{e_j}$ is the j-th standard unit vector for some $j \in \{1, ..., n_y\}$.

Assumption F2. For each chain $j \in \{1, ..., n_y\}$, there exists a value $b_j^* > 0$ so that the detection signal π_k is such that

$$\mathbb{P}\left(\pi_k = 1 \,\middle|\, |b| \geq b_j^*\right) \geq p_j^* \quad \text{and} \quad \mathbb{P}\left(\pi_k = 0 \,\middle|\, |b| < b_j^*\right) \geq \tilde{p}_j^* ,$$

where p_j^*, $\tilde{p}_j^*$ are acceptable probabilities (e.g. $p_j^* = \tilde{p}_j^* = 0.95$).

Remark 34. The assumption **F1** is considered as it is the simplest but indispensable case for a FDI and FE problem. The notation $f_{k,j}^b$ used in this chapter is dedicated to designate a vector depending on k, b and j and not the j-th element of the vector f_k^b. □

Remark 35. The assumption **F2** raises naturally for any fault detection method. As the fault value b has a large magnitude, the fault is detected easily ($\pi_k = 1$) and vice versa. Thus, such thresholds b_j^*'s always exist. The values b_j^*, p_j^* and $\tilde{p}_j^*$ are chosen based on the application and the considered scenario. □

Problem 1. Given a linear discrete time-variant dynamical system with assumptions **(A1)**, **(F1)** and **(F2)**, for a detected fault, determine (isolate) the chain on which it occurs and estimate its value. ■

The Problem 1 focuses on the diagnosis (isolation and estimation) of large faults ($|b| \geq b_j^*$) which are detected almost always (with high probabilities) and some moderate ones ($|b| < b_j^*$) which are detected occasionally (with small probabilities). The detected signals ($\pi_k = 1$) include also false alarms. Beside the ability of isolation and estimation of the real faults, a good fault diagnosis scheme might help to reduce false alarms.

Problem 2. Given a linear discrete time-variant dynamical system with assumptions **(A1)**, **(F1)** and **(F2)**, detect (diagnosis if possible) incipient faults, i.e. the ones with small fault values ($|b| < b_j^*$), in the early stage. ■

Incipient faults are naturally harder to detect. The Problem 2 can be seen as solved (or partially solved) by a scheme which has the ability to:

1. Reduce thresholds b_j^*'s to smaller values (as small as possible value or a reasonable one),
2. Solve the Problem 1 using reduced thresholds b_j^*'s with good/acceptable performance,
3. Reduce false alarms.

A perfect solution to Problem 2 which can detect all incipient faults might be unrealistic.

5.3 Active Fault Diagnosis Scheme for Adaptive degrees of freedom χ^2-statistic method

In this section, an AFD scheme to a discrete dynamical system based on the ADFC method dealing with sensor additive single fault is developed. This scenario corresponds to the type 2 of error in Chapter 4. The proposed scheme focuses to handle the Problem 1 and solves partially the Problem 2.

5.3.1 Motivation

ADFC method is a fault alarm (detection) method that can detect multiple and single faults with magnitudes beyond a threshold. The fault values can be positive or negative. Consequently, it is compatible to the system (4.1) with assumptions **(A1), (F1)** and **(F2)**. These properties are illustrated thanks to the following example.

Example 6. Return to the Bicycle vehicle model simulation in Chapter 4 with scenario 3 and type 2 error. All settings are unchanged unless the ADFC method (Algorithm 7) will be applied with the use of the OUBIKF (Algorithm 3) and the scale parameter $\lambda_k = a_k^{-1/2}$ as mentioned in the related

-40	100	100	100	98.24	
-35	100	100	100	97.92	
-30	100	100	100	97.18	
-25	100	99.56	100	96.76	
-20	**99.70**	**98.64**	**100**	**96.52**	
-15	92.86	89.62	64.98	35.16	
-10	48.34	38.76	22.22	16.22	
-5	15.46	10.92	10.10	6.84	
0	6.9	3.88	9.36	7.52	(FAR(%))
5	14.44	10.94	14.50	11.32	
10	49.12	40.84	15.04	8.52	
15	94.30	91.50	67.22	38.34	
20	**99.56**	**98.42**	**100**	**96.42**	
25	100	99.76	100	97.02	
30	100	99.98	100	97.32	
35	100	100	100	97.92	
40	100	99.98	100	98	

Table 5.1 – Detection rate of ADFC detector applied for Bicycle vehicle model.

discussion, where a_k is such that $S_+([S_k]) \preceq a_k I$ (Algorithm 7). The results after $L = 100$ times of simulations are shown in the following table, where DR(%) stands for detection rate.

It is shown in Table 5.1 that the ADFC method functions with either positive or negative fault values, the thresholds mentioned in assumptions **(F2)** are determined as $b_1^* = b_2^* = b_3^* = b_4^* = 20$ with probabilities beyond 0.96. The values of these thresholds are reduced remarkably thanks to the use of scale parameter λ_k (remaining only 20 from 55 of Table 4.8).

Note that in the case of $b = 0$, there is in fact no fault, thus the corresponding detection rates shown in the table are actually false alarm rate (FAR(%)) which notation is noted right next to them. □

In the framework of system (4.1) and assumptions **(A1), (F1), (F2)**, ADFC method is applied together with OUBIKF where the later is used as a residual generator providing $[r_k] = y_k - [\hat{y}_k]$. The whole fault detection process of this method is summarized in Algorithm 7 and called in brief the *ADFC detector* as mentioned so far.

Let $k \geq 1$, $y_k = y_k^0 + f_k^s$ where y_k^0 is the fault-free measurement. In the faulty case, $f_k^s = b.\mathbf{e_{j_0}}$ with $b \neq 0$ and $j_0 \in \{1, ..., n_y\}$. In the same manner, one gets $[r_k] = [r_k^0] + f_k^s$ and $[\hat{x}_{k|k}] = [\hat{x}_{k|k}^0] + K_k f_k^s$. The developed AFD

scheme is motivated by following questions:

(Q1) Given that the ADFC detector has detected the existence of the sensor fault $f_k^s = b.\mathbf{e_{j_0}}$, what happens if one can add (by chance!) a quantity $\tilde{f}_k = -b.\mathbf{e_{j_0}}$ to the measurement y_k to obtain $\tilde{y}_k$ then rerun the ADFC detector with $\tilde{y}_k$?

(Q2) In the same manner, what happens if one adds each of following quantities:
— $\tilde{f}_k^1 = b_1\mathbf{e_{j_0}}$ where $b_1.b > 0$,
— $\tilde{f}_k^2 = b_2\mathbf{e_{j_0}}$ where $b_2.b < 0$,
— $\tilde{f}_k^3 = b_3\mathbf{e_j}$ where $j \neq j_0$?

Assuming that the ADFC detector has a good enough performance, for **(Q1)**, detection signal π_k equals 0 with the use of $\tilde{y}_k$. Then consider **(Q2)**. In the first case, the fault value has been augmented its magnitude by an additive term of the same sign, so the ADFC detector gives $\pi_k = 1$ with high probability. In the second case, regarding assumption **(F2)**, one gets with high probability that $\pi_k = 0$ if $|b_2 + b| < b_{j_0}^*$ and $\pi_k = 1$ if $|b_2 + b| \geq b_{j_0}^*$. For the last case, it is worthy to note that:
— add a term b_3 to the j-th element of y_k is equivalent to add b_3 to the j-th element of $[r_k]$,
— the statistic used in ADFC detector is $U_k = \sup\{\text{abs}([r_k]^T[r_k]/a_k)\}$.
Since $j \neq j_0$, by assumption, the j-th element of $[r_k]$, says $[r_{k,j}]$, is fault-free. Therefore, $[r_{k,j}]$ is centered nearby 0. With the additive term b_3, whatever it is negative or positive, $[r_{k,j}]$ deviates more from 0 while its width remains unchanged provided that $|b_3|$ is not too small. This implies that U_k takes a greater value while the threshold δ_k determined by $\mathbb{P}[\chi^2(\kappa_k n_y) > \delta_k] = \alpha$ is unchanged since κ_k, n_y and α are unchanged (Algorithm 7). Thus, $\pi_k = 1$ also in this case of **(Q2)**.

5.3.2 Methodology and Scheme

In the considered framework, a sensor fault is characterized by a fault value b and the chain j on which it occurs. Denote a detection signal obtained by applying a sensor fault $f_{k,j}^b$ to the ADFC detector as $\pi_k(f_j^b)$. In addition, since the ADFC detector performs also with the multi faults case, in the next we also use the notation $\pi_k(f_j^b + f_{j'}^{b'})$, $j \neq j'$ to designate the detection signal associated to faults occurring on two different chains.

Let $\Delta \in \left[0\,,\, \min_{j=1:n_y}\{b_j^*\}\right]$ and $M \in \mathbb{N}^*$ so that $M.\Delta$ is superior the maximum fault magnitude we want to estimate (and obviously superior all thresholds b_j^*'s of assumption **(F2)**). Generate auxiliary signals, says fictive faults, as follows:

$$\begin{array}{rclcrcl}
\tilde{f}_{k,1}^{-M} & = & -M.\Delta.\mathbf{e_1}\ , & \ldots & ,\tilde{f}_{k,1}^{M} & = & M.\Delta.\mathbf{e_1}\ , \\
\tilde{f}_{k,2}^{-M} & = & -M.\Delta.\mathbf{e_2}\ , & \ldots & ,\tilde{f}_{k,2}^{M} & = & M.\Delta.\mathbf{e_2}\ , \\
\ldots & & & & & & \\
\tilde{f}_{k,n_y}^{-M} & = & -M.\Delta.\mathbf{e_{n_y}}, & \ldots & ,\tilde{f}_{k,n_y}^{M} & = & M.\Delta.\mathbf{e_{n_y}}\ ,
\end{array} \tag{5.1}$$

or briefly,

$$\tilde{f}_{k,j}^{q} = q.\Delta.\mathbf{e_j}\ , \quad q = -M : M\ , \quad j = 1 : n_y. \tag{5.2}$$

If a fault occurs on a chain $j_0 \in \{1, ..., n_y\}$, says $f_{k,j_0}^{b} = b.\mathbf{e_{j_0}} \neq \mathbf{0}$ and assuming that $0 \leq |b| \leq M.\Delta$, there exists a $q^* \in \{-M, ..., M\}$ so that

$$q^*\Delta \quad < \quad b \quad \leq \quad (q^*+1)\Delta\ .$$

Therefore:

$$0 \ < \ |b + (-q^*\Delta)| \ \leq \ \Delta \qquad \text{and} \qquad 0 \ \leq \ |b + [-(q^*+1)\Delta]| \ \leq \ \Delta,$$

which imply that

$$\pi_k\left(f_{j_0}^{b} + \tilde{f}_{j_0}^{b_i}\right) = 0\ ,\ b_i = -(q^*+i)\Delta\ ,\ i \in \left\{-\left\lfloor x_{j_0}^*/\Delta \right\rfloor + 1 : \left\lfloor x_{j_0}^*/\Delta \right\rfloor\right\} \tag{5.3}$$

with high probability regarding discussions of **(Q1)** and **(Q2)** in the previous section. Also, we have with high probability

$$\begin{array}{rcl}
\pi_k\left(f_{j_0}^{b} + \tilde{f}_{j_0}^{b_i}\right) & = & 1\ ,\quad b_i = -(q^*+i)\Delta\ ,\quad i \notin \left\{-\left\lfloor \frac{x_{j_0}^*}{\Delta} \right\rfloor + 1 : \left\lfloor \frac{x_{j_0}^*}{\Delta} \right\rfloor\right\} \\
\pi_k\left(f_{j_0}^{b} + \tilde{f}_{j}^{q\Delta}\right) & = & 1\ ,\quad q \in \{-M : M\}\ ,\quad j \neq j_0\ .
\end{array}$$

The developed AFD scheme for ADFC method is explanatory via the diagram in Fig.5.1 and Algorithm 9. Regarding Fig.5.1, the ADFC detector includes OUBIKF as its residual generator and the fault detection part of AFD block without auxiliary signals and fault diagnosis part. It provides only detection signal π_k at each iteration. From the figure, we note that when a fault occurs at a time instant k, it follows the feedback $[\hat{x}_{k|k}]$ and affects the residual at next iterations. Thus π_{k+1} reflects not only the existence of f_{k+1}^{s} but also the effect of previous fault f_k^s. The AFD scheme consists in using *fictive faults* as auxiliary inputs and providing them only to the AFD block and not to the monitored system (plant). Thanks to these auxiliary inputs, the ADFC detector produces a signature matrix

$$\mathcal{S}_k = \left[\pi_k\left(f_{j_0}^{s} + \tilde{f}_{j}^{q\Delta}\right)_{j,q}\right] \equiv \left[\pi_{k,j}^{q}\right], \quad j = 1 : n_y, \quad q = -M : M,$$

where j, q are row and column indexes respectively. Then the AFD block is equipped with a fault diagnosis part decoding $\mathcal{S}_k$ to decide whether π_k equals 0 or 1, f_k^s is 0 or takes which estimated value. After that, the estimated fault $\hat{f}_k^s$ is sent backwardly to OUBIKF in order to compensate for the fault effect to $[\hat{x}_{k|k}]$ as an interval estimate for the real state x_k and to the fault diagnosis at the next iteration. Without this fault feedback process, the fault diagnosis has a poor performance.

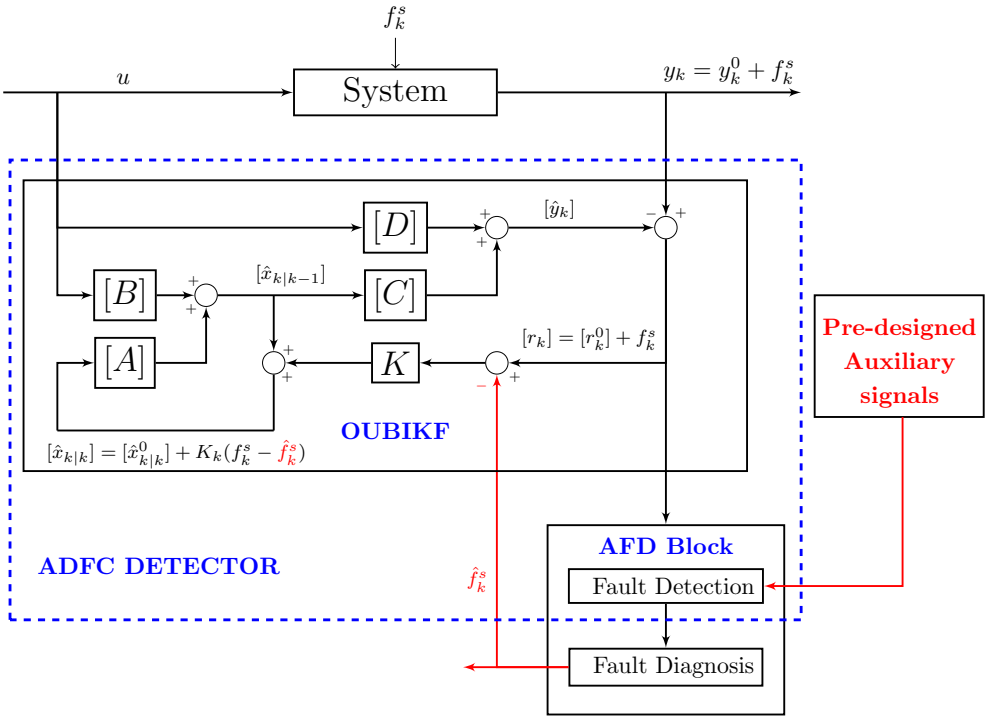

Figure 5.1 – Active Fault Diagnosis diagram using ADFC detector.

The function of the fault diagnosis part of the AFD block as a signature matrix decoder is specified in Algorithm 9. When $\mathcal{S}_k$ has no zero element, all yielded detection signals are 1, in particular,

$$\pi_k\left(f_{j_0}^b + \tilde{f}_{j_0}^{b_i}\right) = 1\ , \quad b_i = -(q^* + i)\Delta\ ,\ i \in \left\{-\left\lfloor x_{j_0}^*/\Delta \right\rfloor + 1 : \left\lfloor x_{j_0}^*/\Delta \right\rfloor\right\}.$$

which contradicts (5.3). In other words, it is almost the case that $\pi_k(f_{j_0}^b) = 1$ for all b nearby and including 0. Therefore, the detected signal $\pi_k = 1$ (line 3 of Algorithm 9) initially dispatched is a false alarm with high probability. When $\mathcal{S}_k$ has at least one zero element, the error chain is estimated as the

Algorithm 9 AFD scheme for ADFC method

Initialization:
Δ, M, $B^* = \{b_j^*, j = 1 : n_y\}$, λ,
$[\hat{x}_{0|0}], \mathcal{P}_{0|0}$, $[A], [B], [C], [D], [Q], [R]$, α, u_k, y_k, $k = 1, 2, ..., N$.
for $k = 1, 2, 3, ...N$ **do**
  Use ADFC detector (Algorithm 7) to get detection signal: π_k
  if $\pi_k = 1$ **then**
    $q = -M : M$; $j = 1 : n_y$; $\tilde{y}_{k,j}^q = y_k + q.\Delta.\mathbf{e_j}$;
    Rerun ADFC detector with $\{\tilde{y}_{k,j}^q\}$ to get $\mathcal{S}_k = \left[\pi_{k,j}^q\right]$,
    if $\mathcal{S}_k$ has no zero element **then**
      $\pi_k = 0$
    else
      $\hat{\mathbf{j_0}} = \text{argmax}_{j=1:n_y} \sum_{q=-M}^{M} \pi_{k,j}^q$;
      Find $I_0 \subset \{-M : M\}$ so that $\pi_{k,\hat{\mathbf{j_0}}}^q = 0$ for all $q \in I_0$
      $\hat{b} = -\text{mean}\{I_0\}.\Delta$; $\hat{f}_k^s = \hat{b}.\mathbf{e_{\hat{j}_0}}$;
      if $|\hat{b}| \leq \lambda.\min\{B^*\}$ **then**
        $\pi_k = 0$
      else
        $\hat{f}_k^s = \text{sign}(\hat{b}).\max\{|\hat{b}|, \text{mean}(B^*)\}.\mathbf{e_{\hat{j}_0}}$
        $[\hat{x}_{k|k}] = [\hat{x}_{k|k}] - K.\hat{f}_k^s$
      end if
    end if
  end if.
end for

Note: The parameters in the second line of the initialization are required only for ADFC detector (Algorithm 7).

one on which $\mathcal{S}_k$ has the maximum number of zero elements, denoted by $\hat{j}_0$. That is because a single fault at a chain j_0 may affect the behavior of residual $[r_k]$ on another chain $j \neq j_0$, however its effects on the chain j_0 is the stronger. Then, the fault is estimated as the additive inverse of the mean of all fictive faults $\tilde{f}_{k,\hat{j}_0}^{q\Delta}$ with which $\pi_k \left(f_{j_0}^b + \tilde{f}_{\hat{j}_0}^{q\Delta} \right) = 0$.

Denote the estimated fault as $\hat{f}_k^s = \hat{b}.\mathbf{e_{\hat{j}_0}}$ and the diagnosed detection signal as $\hat{\pi}_k$. The diagnosis is called *r-accurate* if

$$\hat{j}_0 \equiv j_0 \qquad \text{and} \qquad |\hat{b} - b| \leq r \,, \tag{5.4}$$

where $r > 0$ is a predetermined radius. This condition is also called the *r-accuracy*.

Next, in order to eliminate more false alarms and reinforce the estimation accuracy, a regularization is performed. Regarding the assumption **(F2)**, ideally a fault value $b \geq b^*_{j_0}$ is detected and hence $\hat{b}$ must be at least close to $\min\{b^*_j, j = 1 : n_y\}$. So, if the estimated value $\hat{b}$ is such that

$$|\hat{b}| \leq \lambda . \min\{b^*_j, j = 1 : n_y\}, \quad \lambda \in [0, 1],$$

we consider that it is not consistent with assumption **(F2)** and hence $\hat{b}$ is replaced by 0 and $\hat{\pi}_k = 0$ is dispatched. In the case that

$$\lambda . \min\{b^*_j, j = 1 : n_y\} \ < \ |\hat{b}| \ \leq \ \text{mean}\{b^*_j, j = 1 : n_y\},$$

we consider that there is something intervening and lessening the estimated value $\hat{b}$. So $\hat{b}$ is replaced by $\text{mean}\{b^*_j, j = 1 : n_y\}$ and $\hat{\pi}_k = 1$ is dispatched. The value $\text{mean}\{b^*_j, j = 1 : n_y\}$ is chosen as the replacing value for $\hat{b}$ because the actual faulty chain j_0 is not known, otherwise $b^*_{j_0}$ could be used. Finally, the estimated fault is fed backwardly to ADFC detector by subtracting the amount $K\hat{f}^s_k$ to $[\hat{x}_{k|k}]$.

5.4 Application

In this section, consider again the Bicycle vehicle model which is applied in the Chapter 4 using the ADFC method. All parameters related to the model remain unchanged and the OUBIKF (Algorithm 3) is used inside the ADFC detector (Algorithm 7). Here, the case of single sensor faults is applied to diagnosis. The threshold values b^*_j's mentioned in assumption **(F2)** are taken from Example 6: $b^*_j = 20$, $\forall j = 1 : n_y$ with probabilities beyond 0.96. So, the ADFC detector has been tuned by the use of the scale parameter $\lambda_k = a_k^{-1/2}$ to match the goals of increasing of detection rates and decreasing of thresholds b^*_j's. Other parameters chosen for AFD scheme 9 are: $\Delta = 5$, $M = 12$, $\lambda = 0.5$ and significance level $\alpha = 0.03$.

Recall that the AFD scheme is dedicated to diagnose faults beyond thresholds b^*_j's as the **Problem 1** has been formulated. Thus, in order to illustrate the function of the scheme as well as its performance, a fault value $b = -25$ is fixed. Then, the four different faults $f^s_{k,j} = -25\mathbf{e_j}$ are tested corresponding to the four chains $j = 1 : 4$ of the measurements y_k. In each case, an error range of length 50 (time instants) are randomly chosen in which the fault occurs. Then the ADFC detector (Algorithm 7) is performed without and with AFD scheme. Thanks to that we can answer to several questions:

(1) Does the AFD scheme enhance the fault detection of ADFC detector ?

(2) Does the AFD scheme help to reduce false alarms ?

(3) How well is the fault diagnosis provided by the AFD scheme ?

The first two questions are answered positively thanks to Fig.5.2 and Tables 5.2-5.3 using evaluation indexes (DR, NDR, FAR, EFF) defined in Section 4.2.2. By these, the faults are totally detected in the error range (DR = 100%) by the ADFC detector. In addition, applying the AFD scheme, it reduces the false alarm rate (FAR) from about 8.6% to about 1% and hence the efficient (EFF) indexes are augmented to 99%.

Chain	DR (%)	NDR (%)	FAR (%)	EFF (%)
1	100	0	8.6	91.4
2	100	0	8.6	91.4
3	100	0	8.7	91.3
4	100	0	8.7	91.3

Table 5.2 – Detection performance without AFD technique

Chain	DR (%)	NDR (%)	FAR (%)	EFF (%)
1	100	0	0.7	99.3
2	100	0	1.0	99.0
3	100	0	1.0	99.0
4	100	0	1.0	99.0

Table 5.3 – Detection performance with AFD technique

To deal with the question (3) above, we measure the fault diagnosis performance by the accuracy rate A_r defined by

$$A_r = \sum_{k \in \mathcal{R}_j} \frac{\mathbb{I}(|\hat{b}(k) - b(k)| \leq r)\mathbb{I}(\hat{j} = j)}{|\mathcal{R}_j|} \times 100\% \,, \tag{5.5}$$

where $r > 0$ is a predetermined radius, $\mathbb{I}(.)$ is the indicator function, $\mathcal{R}_j$ is the error range corresponding to the fault chain j, $|\mathcal{R}_j|$ is the length of $\mathcal{R}_j$, $\hat{b}(k)$ and $\hat{j}$ are estimates of actual values $b(k)$ and j, k is the time instant. So, A_r is the percentage of fault estimates in $\mathcal{R}_j$ satisfying (5.4).

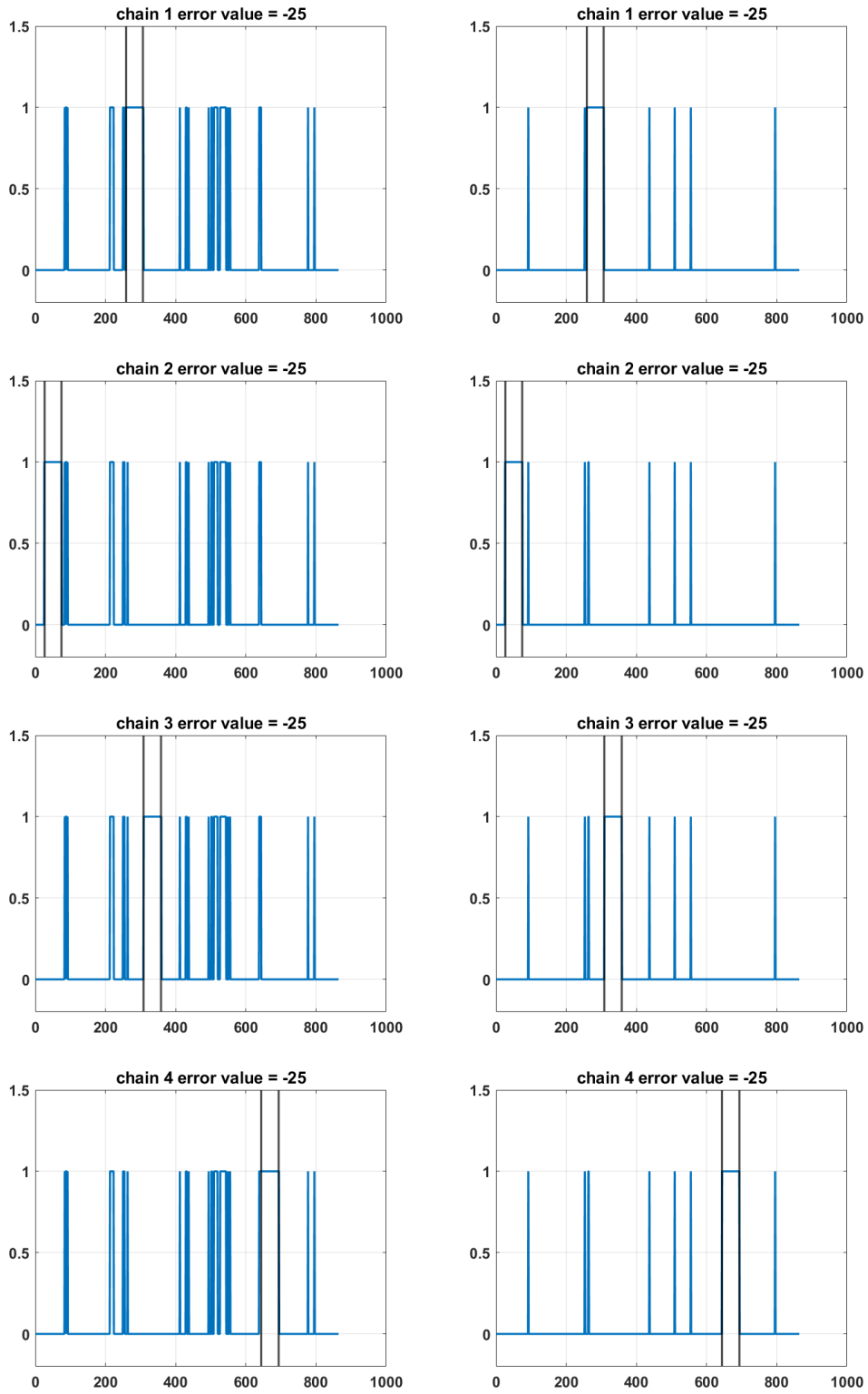

Figure 5.2 – Detection signals without (*left*) and with (*right*) AFD technique

The simulation results related to the question (3) are given by Table 5.4. From the last row of the table, it is shown that all estimated chains are correct and all estimated fault values $\hat{b}(k)$ are away from the actual faults $b(k)$ at most a radius of $r = \Delta = 5$. The second row of the Table provides the accuracy percentage corresponding to the radius $r = \Delta/2$.

Chain		1	2	3	4
A_r (%)	$r = 5/2$	86	100	78	98
	$r = 5$	100	100	100	100

Table 5.4 – The $A_r(\%)$ accuracy rate

Apart from the three questions discussed above, the fact that AFD scheme using estimated fault to feed backwardly into $[\hat{x}_{k|k}]$ increases the estimation performance of the OUBIKF as the results shown in Fig.5.3-5.4. In the first figure, the estimate intervals between two vertical black lines (the error range) deviate from the real states, even no longer contain these states and, in addition, the widths of these estimate intervals increase. In contrast, in the later figure, the estimate intervals still track the real states well with reasonable widths.

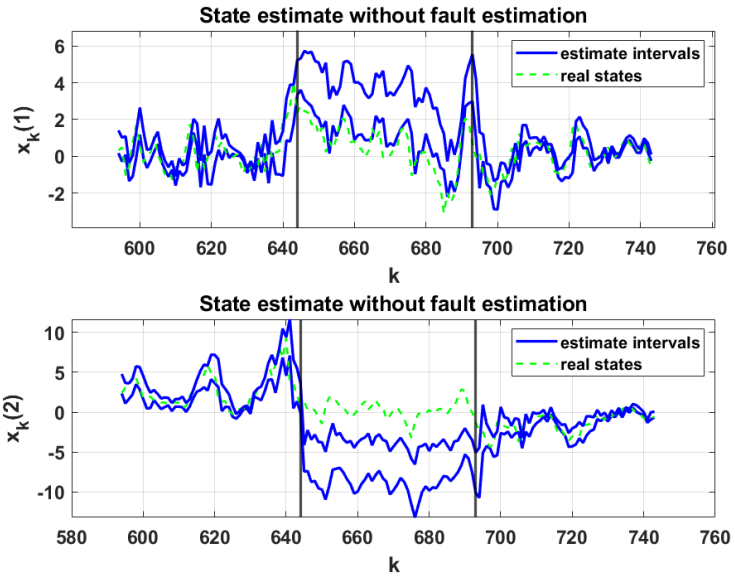

Figure 5.3 – Active fault diagnosis - State estimates without fault estimation.

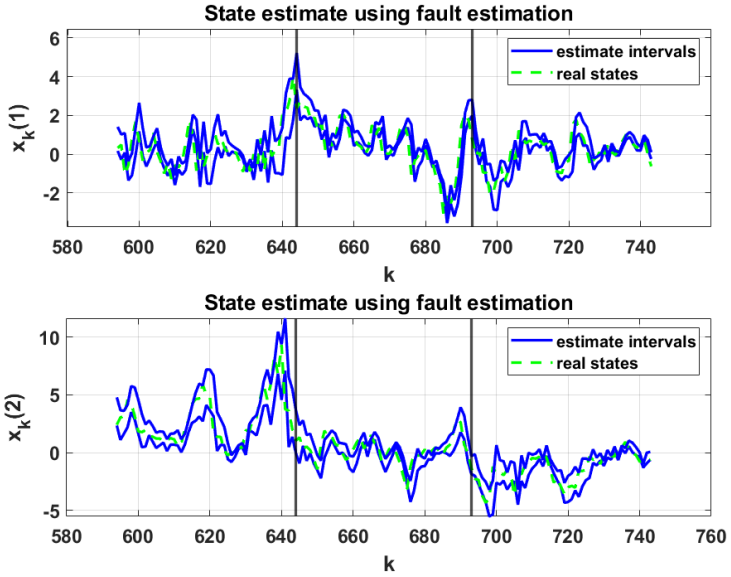

Figure 5.4 – Active fault diagnosis - State estimates using fault estimation.

5.5 Conclusion and perspective

In this chapter, an AFD scheme is developed for ADFC method. It can be considered as an extension and improvement of the ADFC detector to make it become an advanced detector or says a diagnoser using auxiliary signals. All standard functionalities of a fault diagnoser are concerned, included FDI and FE.

The most relevant characteristic of the scheme is that auxiliary signals are not injected into the monitored system but provided only to the diagnoser. This is also the key difference of the scheme with other AFD methods. This helps to avoid additional disturbances due to auxiliary signals on the monitored system. In addition, auxiliary signals are designed off-line and only injected into the diagnoser once a detected signal ($\pi_k = 1$) is dispatched at a time instant k. Then, the generated signature matrix is analyzed to provide decisions about the fault candidate without delay of any finite time (instant) interval in which the diagnoser waits reactions of the monitored system being injected. This implies that the developed scheme can provide an on-line fault diagnosis with no delay in time instant and with computation time depending only on the computer performance.

Another important characteristic of the scheme is the compensation for the actual fault effect to the diagnosis at the next iteration by using the estimated fault as a feedback to the diagnoser. It is important for the developed

scheme because without it, the diagnosis performance of the scheme degrades severely. It may be also a good additional strategy for several existing AFD methods.

As an initial development proper for the ADFC method, the scheme has its limits needed to be solved in future researches. In the scheme, only the case of single fault is treated. Thus, multiple fault diagnosis is an important extension of the scheme. Secondly, there is a number of faults which are neglected by the initial detection of the ADFC detector. An additional scheme aiming to deal with these neglected faults is then a potential improvement of the one studied in this chapter. Finally, a control feedback design for the case of a fault already diagnosed (detected, localized and estimated) might be an interesting and significant issue to be investigated.

Chapter 6

Conclusion

The thesis is structured with two main parts contributing in two major subjects after a preamble chapter, namely State of the art (Chapter 1), and followed by a global conclusion of the thesis (Chapter 6). In the first part, the former subject is concerned the state estimation or filtering problem in a framework of mixed uncertainties, while in the second part, the later subject deals with fault diagnosis based on results developed in the first one.

Throughout the thesis, discrete time dynamical systems are investigated. The linear case is studied in Chapter 2 which results in the Optimal Upper Bound Interval Kalman Filter (OUBIKF). The nonlinear case is under consideration in Chapter 3 and this study produces the so-called Reinforced Likelihood Box Particle Filter (RLBPF). Then, these two filters are used in development of fault diagnosis methods in a unified framework corresponding to the passive approach in Chapter 4 and the active approach in Chapter 5. The unified framework is based on the Adaptive Degrees of Freedom χ^2-statistic (ADFC) method applied to either linear or nonlinear system as a passive stochastic adaptive approach and extended as the main part, called ADFC detector, of the Active Fault Diagnosis (AFD) scheme applied to linear system. Furthermore, a global unified framework applied to the whole study is the mixed uncertainty context to which the dynamical systems under consideration either linear or nonlinear and either in filtering or diagnosis problems are concerned. In this context, for linear case, bounded-error uncertainty is considered for system matrices (A_k, B_k, C_k, D_k) with known inputs u_k's and additive Gaussian noises with bounded-error uncertainty covariance matrices are taken into account in state and measurement dynamics. For nonlinear case, similar additive Gaussian noise assumptions are considered for the measurement dynamic, while the bounded-error uncertainty is applied to system inputs u_k and state dynamic disturbances w_k contained in known intervals $[u_k], [w_k]$, given that dynamic functions f, h are already

known. Thus, the thesis contribution in this context is a generalization of classical (set-membership and stochastic) approaches using interval analysis in the viewpoint that point values of variables/parameter-matrices are special cases of corresponding interval values.

A literature background is introduced in the first chapter of the thesis. This review provides basic notions used in the followed chapters involving the standard Kalman Filter, the Bayesian filtering approaches, the particle filtering and the fault diagnosis. This introduction matches completely the development of the OUBIKF and RLBPF for the filtering part and of the ADFC method and AFD scheme for the fault diagnosis part. Also, a mathematical background is presented in the beginning of Chapter 2 which provides consistent notations and definitions applied in the whole study as well as necessary properties recalled or developed concisely in a theorem-proof structure. Furthermore, the proposed methods are all summarized in corresponding Algorithms which favor the comprehension and re-implementation of them. The method efficiency and performance are illustrated numerically by automotive benchmark models (Bicycle vehicle model for linear case and Suspension model for nonlinear case) throughout the thesis. Some academic examples are also provided. The thesis also proposes indicator indexes and/or scenarios in order to evaluate developed methods in comparison with others.

The essential advantages of the developed filters pointed out via simulations can be summarized as follows: the computation cost and the resulted interval estimate widths are remarkably reduced while guaranteed estimates are preserved (i.e. interval estimates contain the real states with high percentage O%). In the passive approach, the ADFC method is shown to be efficient in fault detection of either single or multiple additive faults and either positive or negative fault values thanks to adaptive threshold technique and tuning factors. In the active approach, the ADFC method is enhanced and embedded in the AFD scheme using predesigned auxiliary signals together with the fault estimate feedback to reduce the false alarms, increase the detection rate, localize the fault and estimate its value.

Some other advantages of the proposed methods can be rediscovered at the end of corresponding chapters with more detailed discussions. Besides strong points, the thesis contributions have several limitations leading to a number of potential future researches presented below.

In the filtering part, the OUBIKF concerns the interval Kalman filtering in which field a number of issues are not investigated systematically, especially those related to filter convergence, system/filter stability, controller/observer design using interval estimates. In another view, investigate the robust control aspect of the OUBIKF is also an interesting research. The RLBPF, in principle, can be implemented with any state and measurement

continuous dynamical functions, unless conditions under which the filter provide a good performance or guaranteed results, e.g. with C-stability, are not pointed out. The control factors $\mathbf{c_1}, \mathbf{c_2}, \mathbf{c_3}, \epsilon$, **SmF, ScF**, on the one hand, make the filter be more efficient, flexible and suitable for numerous applications, on the other hand, they are subject to future studies about optimal choices and/or automatic adaptive choices of them either by analytical or machine learning method. Investigation of RLBPF on some concrete classes of state and measurement dynamical functions (e.g. L-Lipschitz, L_2,...) is also a potential research perspective. The score function J as well as the method weighting it may be improved and the number of particles applied in the partition step is an issue of the filter.

About the fault diagnosis part, the ADFC method is developed however in the framework of (additive) sensor fault systems. Extend this method to deal with other kinds of fault (e.g. actuator faults) and with fault identification is a perspective of our future research. In the AFD scheme, only the case of single fault is treated. Thus, multiple fault diagnosis is an important extension of the scheme. Secondly, following the scheme, there is a number of faults which are neglected by the initial detection of the ADFC detector. An additional scheme aiming to deal with these neglected faults is then a potential improvement of the present study. Finally, a control feedback design for the case of a fault already diagnosed (detected, localized and estimated) might be an interesting and significant issue to be investigated.

Bibliography

Abdallah, F., Gning, A., and Bonnifait, P. (2008). Box particle filtering for nonlinear state estimation using interval analysis. *Automatica*, 44(3):807–815.

Anderson, B. D. and Moore, J. B. (1979). *Optimal Filtering*. Prentice-Hall, Inc., Englewood Cliffs, N.J. 07632.

Barata, J. and Hussein, M. (2012). The Moore – Penrose pseudoinverse: A tutorial review of the theory. *Braz J Phys 42*, page 146–165.

Beard, R. V. (1971). *Failure accomodation in linear system through self-reorganization*. PhD thesis, Massachusetts Institute of Technology Dept. of Aeronautics and Astronautics.

Beezer, R. A. (2015). *A First Course in Linear Algebra*. Robert A. Beezer, Congruent Press, Gig Harbor, Washington, USA.

Blesa, J., Le Gall, F., Jauberthie, C., and Travé-Massuyès, L. (2015). State estimation and fault detection using box particle filtering with stochastic measurements. In *26th International Workshop on Principles of Diagnosis (DX-15)*, pages 67–73, Paris, France.

Campbell, S. L. and Nikoukhah, R. (2004). *Auxiliary Signal design for Failure Detection*. Princeton Univeristy Press, 41 William Street, Princeton, New Jersey, 08540.

Cayley, A. (1858). A memoir on the theory of matrices. *Philos. Trans. 148*.

Chen, G., Wang, J., and Shieh, S. (1997). Interval Kalman filtering. *IEEE Transactions on Aerospace and Electronic Systems*, 33(1):250–259.

Chen, J., Guo, C., Hu, S., Sun, J., Langari, R., and Tang, C. (2020). Robust estimation of vehicle motion states utilizing an extended set-membership filter. *Applied Sciences*, 10(4):1343.

Chen, J. and Patton, R. J. (1999). *Robust Model-Based Fault Diagnosis for Dynamic Systems*. The Kluwer International Series on Asian studies in Computer and Information Science. Kluwer Academic Publishers, Springer science+business media New York edition.

Combastel, C. (2005). A state bounding observer for uncertain non-linear continuous-time systems based on zonotopes. In *Proceedings of the 44th IEEE Conference on Decision and Control and European Control Conference*, pages 7228–7234. Seville, Spain.

Combastel, C. (2015). Merging Kalman filtering and zonotopic state bounding for robust fault detection under noisy environment. In *Proceedings of the 9th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*. Paris, France.

Ding, S. X. (2013). *Model-Based Fault Diagnosis Techniques*. Springer.

Doucet, A., de Freitas, N., and Gordon, N., editors (2001). *Sequential Monte Carlo Methods in Practice*. Springer.

Efimov, D. and Raïssi, T. (2016). Design of interval observers for uncertain dynamical systems. *Automation and Remote Control / Avtomatika i Telemekhanika*, 77(2):191–225.

Fergani, S. (2014). *Robust multivariable control for vehicle dynamics*. PhD thesis, Grenoble INP, GIPSA-lab, Control System dpt., Grenoble, France.

Frobenius, F. G. (1878). Ueber lineare substutionen und bilineare formen. *J. Reine Angew. Math. 1878 (84): 1–63.*

Gordon, N. J., Salmond, D. J., and Smith, A. F. M. (1993). Novel approach to nonlinear/non-gaussian bayesian state estimation. *IEE Proceedings F - Radar and Signal Processing*, 140(2):107–113.

Hertz, D. (1992). The extreme eigenvalues and stability of real symmetric interval matrices. *IEEE Transactions on Automatic Control*, 37(4):532–536.

Hladik, M. (2013). Bounds on eigenvalues of real and complex interval matrices. *Applied Mathematics and Computation*.

Ian R., P. and Andrey V., S. (1999). *Robust Kalman Filtering for signals and systems with large uncertainties*. Birkhauser Boston.

Ifqir, S., Ichalal, D., Oufroukh, N. A., and Mammar, S. (2018). Robust interval observer for switched systems with unknown inputs: Application to vehicle dynamics estimation. *European Journal of Control*, 44:3–14.

Jaulin, L., Kieffer, M., Didrit, O., and Walter, E. (2001). *Applied Interval Analysis, with Examples in Parameter and State Estimation, Robust Control and Robotics*. Springer-Verlag, London.

Jones, H. L. (1973). *Failure accomodation in linear system through self-reorganization*. PhD thesis, Massachusetts Institute of Technology Dept. of Aeronautics and Astronautics.

Kalman, R. (1960). A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering*, 82(Series D):35–45.

Kincaid, D. and Cheney, W. (1991). *Numerical Analysis*. Brooks/Cole Publishing Company, Wadsworth, Inc.

Kolmogorov, A. (1941). Interpolation and extrapolation of stationary sequences. *Bull. de l'Académie des Sciences de USSR*, pages 3–14.

Lu, Q. H., Fergani, S., and Jauberthie, C. (2021). A new scheme for fault detection based on optimal upper bounded interval kalman filter. *IFAC-PapersOnLine*, 54(7):292–297. 19th IFAC Symposium on System Identification SYSID 2021.

Lu, Q. H., Fergani, S., Jauberthie, C., and Le Gall, F. (2019). Optimally bounded interval kalman filter. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 379–384.

Lunze, J. and Richter, J. (2008). Reconfigurable fault-tolerant control: A tutorial introduction. *European Journal of Control*, 14(5):359–386.

Mehra, R. (1970). On the identification of variances and adaptive kalman filtering. *IEEE Transactions on Automatic Control*, 15(2):175–184.

Mehra, R. and Peschon, J. (1971). An innovations approach to fault detection and diagnosis in dynamic systems. *Automatica*, 7:637–640.

Meseguer, J., Puig, V., and Escobet, T. (2010). Fault diagnosis using a timed discrete-event approach based on interval observers: Application to sewer networks. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 40(5):900–916.

Meyer, C. D. (2000). *Matrix Analysis and Applied Linear Algebra*. SIAM.

Mohamed, S. and Nahavandi, S. (2012a). Robust finite-horizon Kalman filtering for uncertain discrete-time systems. *IEEE Transactions on Automatic Control*, 57(6):1548–1552.

Mohamed, S. M. K. and Nahavandi, S. (2012b). Robust finite-horizon kalman filtering for uncertain discrete-time systems. *IEEE Transactions on Automatic Control*, 57(6):1548–1552.

Nassreddine, G., Abdallah, F., and Denœux, T. (2010). State estimation using interval analysis and belief-function theory: Application to dynamic vehicle localization. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 40(5):1205–1218.

Niemann, H. (2005). Fault tolerant control based on active fault diagnosis. In *Proceedings of the 2005, American Control Conference, 2005.*, pages 2224–2229 vol. 3.

Niemann, H. and Poulsen, N. K. (2005). Active fault diagnosis in closed-loop systems. *IFAC Proceedings Volumes*, 38(1):448–453. 16th IFAC World Congress.

Niemann, H. and Poulsen, N. K. (2014). Active fault detection in MIMO systems. In *2014 American Control Conference*, pages 1975–1980.

Nino-Juarez, E., Ramirez-Mendoza, R., Morales-Menendez, R., Sename, O., and Dugard, L. (2008). Minimizing the frequency effect in a black box model of a magneto-rheological damper. In *Mini conference; 11th, Vehicle system dynamics, identification and anomalies*, pages 733–742. Technical University of Budapest.

Patton, R. J., Frank, P. M., and Clark, R. N. (2013). *Issues of fault diagnosis for dynamic systems*. Springer Science & Business Media.

Prevention, W. H. O. V. I. (2013). *Global status report on road safety 2013: supporting a decade of action*. World Health Organization.

Puig, V. (2010). Fault diagnosis and fault tolerant control using set-membership approaches: Application to real case studies. *International Journal of Applied Mathematics and Computer Science*, 20(4):619–635.

Pukelsheim, F. (2006). *Optimal design of Expiriments*. SIAM's Classics in Applied Mathematics. M. Dekker, New York.

Punčochář, I. and Škach, J. (2018). A survey of active fault diagnosis methods. *IFAC-PapersOnLine*, 51(24):1091–1098. 10th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes SAFEPROCESS 2018.

Raka, S.-A. and Combastel, C. (2013). Fault detection based on robust adaptive thresholds: A dynamic interval approach. *Annual Reviews in Control*, 37(1):119–128.

Rauh, A., Krasnochtanova, I., and Aschemann, H. (2011). Quantification of overestimation in interval simulations of uncertain systems. In *2011 16th International Conference on Methods Models in Automation Robotics*, pages 116–121.

Rohn, J. (1998). Bounds on Eigenvalues of Interval Matrices. *Zamm J. Appl. Math. Mech.*, 78:1049–1050.

Rump, S. (1999). INTLAB - INTerval LABoratory. In Csendes, T., editor, *Developments in Reliable Computing*, pages 77–104. Kluwer Academic Publishers, Dordrecht. http://www.tuhh.de/ti3/rump/.

Sainz, M. Á., Armengol, J., and Vehí, J. (2002). Fault detection and isolation of the three-tank system using the modal interval analysis. *Journal of process control*, 12(2):325–338.

Sayed, A. (2001). A framework for state-space estimation with uncertain models. *IEEE Transactions on Automatic Control*, 46(7):998–1013.

Stoustrup, J. and Niemann, H. (2010). Active fault diagnosis by controller modification. *International Journal of Systems Science*, 41.

Tan, J., Olaru, S., Seron, M. M., and Xu, F. (2021). Set-based guaranteed active fault diagnosis for lpv systems with unknown bounded uncertainties. *Automatica*, 128:109602.

Tran, T. (2017). *Cadre unifié pour la modélisation des incertitudes statistiques et bornées - Application à la détection et isolation de défauts dans les systèmes dynamiques incertains par estimation.* Phd thesis, EDSYS, Université Toulouse III Paul Sabatier, LAAS-lab, Toulouse, France.

Tran, T. A., Jauberthie, C., Le Gall, F., and Travé-Massuyès, L. (2018). Evidential box particle filter using belief function theory. *International Journal of Approximate Reasoning*, 93:40–58.

Tran, T. A., Jauberthie, C., Le Gall, F., and Travé-Massuyès, L. (2017). Interval kalman filter enhanced by positive definite upper bounds. *Proceedings of 20th IFAC World Congres 2017*, 50(1):1595–1600.

Tran, T. A., Jauberthie, C., Trave-Massuyés, L., and Lu, Q. H. (2021). An interval kalman filter enhanced by lowering the covariance matrix upper bound. *International Journal of Applied Mathematics and Computer Science*, 31(2):259–269.

Wiener, N. (1949). *Extrapolation, Interpolation and Smoothing of stationary time series: With engineering applications*. The MIT Press.

Willsky, A. S. (1976). A survey of design methods for failure detection in dynamic systems. *Automatica*, 12:601–611.

Willsky, A. S., Deyst, J. J., and Crawford, B. S. (1974). Adaptive filtering and self-test methods for failure detection and compensation. *Proc. of the 1974 JACC*, pages 434–437.

Willsky, A. S., Deyst, J. J., and Crawford, B. S. (1975). Two self-test methods applied to an inertial system problem. *J.Spacecraft*, 12(7):434–437.

Xie, L., Soh, Y. C., and de Souza, C. (1994). Robust kalman filtering for uncertain discrete-time systems. *IEEE Transactions on Automatic Control*, 39(6):1310–1314.

Xiong, J., Jauberthie, C., Travé-Massuyès, L., and Le Gall, F. (2013). Fault detection using interval Kalman filtering enhanced by constraint propagation. In *Proceedings of the 52nd IEEE Annual Conference on Decision and Control (CDC)*, pages 490–495, Florence, Italy.

Zhan, X. (2002). *Matrix Inequalities*. Springer-Verlag Berlin Heidelberg.

Zhe, D. and Zheng, Y. (2006). Finite-horizon robust Kalman filtering for uncertain discrete time-varying systems with uncertain-covariance white noises. *IEEE Signal Processing Letters*, 13(8):493–496.