Health Monitoring of Nonlinear Systems with Application to Gas Turbine Engines

Najmeh Daroogheh

A Thesis

in

The Department

of

Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of Degree of Doctor of Philosophy at

Concordia University

Montréal, Québec, Canada

November 2016

CONCORDIA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

This is to certify that the thesis prepared

By: Najmeh Daroogheh

Entitled: Health Monitoring of Nonlinear Systems with Application to Gas Turbine Engines

and submitted in partial fulfilment of the requirements for the degree of

Degree of Doctor of Philosophy

Complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

_____ Dr. Chair

_____ Dr. I. Sharf (External Examiner)

_____ Dr. Y. Zhang (MIE)

_____ Dr. S. Hashtrudi Zad (ECE)

_____ Dr. R. Soleymani (ECE)

_____ Dr. K. Khorasani, Thesis Co-Supervisor

_____ Dr. N. Meskin, Thesis Co-Supervisor

Approved by: _____

Dr. , Chair

Department of Electrical and Computer Engineering

_____ 2016 _____

Dr.

Dean, Faculty of Engineering and Computer Science

ABSTRACT

Health Monitoring of Nonlinear Systems with Application to Gas Turbine Engines

Najmeh Daroogheh, Ph.D.

Concordia University, 2016

Health monitoring and prognosis of nonlinear systems is mainly concerned with system health tracking and its evolution prediction to future time horizons. Estimation and prediction schemes constitute as principal components of any health monitoring framework. In this thesis, the main focus is on development of novel health monitoring techniques for nonlinear dynamical systems by utilizing model-based and hybrid prognosis and health monitoring approaches.

First, given the fact that particle filters (PF) are known as a powerful tool for performing state and parameter estimation of nonlinear dynamical systems, a novel dual estimation methodology is developed for both time-varying parameters and states of a nonlinear stochastic system based on the prediction error (PE) concept and the particle filtering scheme. Estimation of system parameters along with the states generate an updated model that can be used for a long-term prediction problem.

Next, an improved particle filtering-based methodology is developed to address the prediction step within the developed health monitoring framework. In this method, an observation forecasting scheme is developed to extend the system observation profiles (as time-series) to future time horizons. Particles are then propagated to future time instants according to a re-sampling algorithm in the prediction step. The uncertainty in the long-term prediction of the system states and parameters are managed by utilizing dynamic linear models (DLM) for development of an observation forecasting scheme. A novel hybrid architecture is then proposed

to develop prognosis and health monitoring methodologies for nonlinear systems by integration of model-based and computationally intelligent-based techniques. Our proposed hybrid health monitoring methodology is constructed based on a framework that is not dependent on the structure of the neural network model utilized in the implementation of the observation forecasting scheme. Moreover, changing the neural network model structure in this framework does not significantly affect the prediction accuracy of the entire health prediction algorithm.

Finally, a method for formulation of health monitoring problems of dynamical systems through a two-time scale decomposition is introduced. For this methodology the system dynamical equations as well as the affected damage model, are investigated in the two-time scale system health estimation and prediction steps. A two-time scale filtering approach is developed based on the ensemble Kalman filtering (EnKF) methodology by taking advantage of the model reduction concept. The performance of the proposed two-time scale ensemble Kalman filters is shown to be more accurate and less computationally intensive as compared to the well-known particle filtering approach for this class of nonlinear systems.

All of our developed methods have been applied for health monitoring and prognosis of a gas turbine engine when it is affected by various degradation damages. Extensive comparative studies are also conducted to validate and demonstrate the advantages and capabilities of our proposed frameworks and methodologies.

To my lovely parents Maryam and Hoseinali,

for their patience and tireless support throughout my life

ACKNOWLEDGEMENTS

The past six years have been a fantastic journey and certainly one of important periods of my life. But getting to the end is even sweeter because of all the challenges. I would like to express my special appreciation and thanks to my supervisors Prof.K Khorasani and Dr.N Meskin. I appreciate all your contributions of time, ideas, and funding to make my Ph.D. career productive. It has been an honor to be your student.

I would also like to thank my committee members, Prof.I Sharf, Dr.Y Zhang, Dr.Sh Hashtrudi Zad, and Dr.R Soleymani for serving as my committee and for your great comments and suggestions. I also want to thank you for letting my defense be an enjoyable moment.

My great colleagues in CAE. During the last two years of this career, I had chance to work with you in a progressive and friendly environment. I would like to express my special thanks and gratitude to my team lead Mr.D McIsaac, thank you for your support and motivation to start my career in the industry.

My fantastic friends Mehrnoosh Abbasfard, Zahra Gallehdari, Mahsa Karamzadeh, Shohreh Zehtabian, Azadeh Hayeri, and Yasaman Babaei. Now and always, thank you for being with me during this journey and your support. My wonderful friends and colleagues Amir Baniamerian, and Bahareh Pourbabaee, we had great times together in the same office. Thank you for your contributions and assistants during all of these years.

To my mom, dad, beautiful sister Hamideh and lovely brothers Mohammad Reza and Abbas. I feel blessed to have you. Thank you for your support. I am glad to celebrate this achievement with you and to see your pride. My spirit seeks challenges and I always aim for the best. I have these values because of you. Thank you for pushing me to do this, to make the best of myself.

You were the main inspiration that kept me going on this path.

TABLE OF CONTENTS

List of Figures

List of Tables

LIST OF ABBREVIATIONS AND SYMBOLS

PHM	prognosis and health management
PF	particle filter
EF	equivalent flop
DLM	dynamic linear model
RUL	remaining useful life
ARMA	autoregressive moving average
PRMSE	ercent of mean square error
VOM	variable order model
FOM	fixed order model
W/O	without resampling
RUL-API	RUL online accuracy-precision index
RUL-OPI	RUL online precision index
RUL-OSI	RUL online steadiness index
NN	Multi-Layer Perceptron (MLP) neural network
RNN	Recurrent neural network
WNN	Wavenet neural network
Gpdf	Gaussian probability density function
EnKF	Ensemble Kalman filter
NSP	Nonlinear singularly perturbed

Chapter 1

Introduction

1.1 Motivation

The maintenance cost reduction is one of the challenging goals in today's highly complex and interconnected engineering systems. Highly correlated to this objective is prediction of the future health of a system. This problem has recently become important and an active area of research for production and maintenance optimization goals. Research on reliable health monitoring techniques can potentially reduce the downtime and breakdowns of a system, and consequently enhance the cost savings and operational safety [6–13]. To achieve these goals, prognosis and health management (PHM) techniques have been pursued as principal active fields of research in various disciplines [13–18]. Moreover, research and development of new prognosis and health monitoring techniques for nonlinear complex engineering systems have

a potential to lead to significant improvements in their safety and reliability and reductions in their maintenance costs [8–12, 19–23].

The prognosis problem can be described as the prediction of the system's long-term behavior based on the evolution of its health indicators. In the domain of reliability engineering which is a common field of research among electrical, mechanical and material engineering, this problem is called "failure prognostics". It consists of two main steps: tracking anomalous behavior caused by a hidden damage from the system (noisy) observations, and predicting the remaining useful life (RUL) of affected components in the system. Two principal approaches for solving the prognostics problem exist: data-based and model-based methods. Whereas data-based methods are more efficient in cases where the experimental plant data is available, model-based methods are considered to be more useful for systems with available mathematical models.

In prognosis and health management (PHM) systems there are two principal components, namely: (i) the system health tracking that is to be achieved by analyzing the system behavior signatures or its health parameters (also known as estimation module), and (ii) the system health prediction that is to be achieved by analyzing the evolution of system signatures in the long term horizon (also known as prediction module) for predicting the RUL of the system. Prognostic methods attempt to predict the future health of a system for determining its RUL before failure occurs [6]. Performance of prognostic schemes are mainly influenced by the accuracy of the prediction method, that in turn can be affected by activities such as maintenance actions [24]. Therefore, uncertainty management is an important and challenging problem that should be considered in development of a prognostic framework [25].

In this thesis, we try to solve the health estimation and prediction problems for a class of nonlinear systems using nonlinear filtering methods. The linear stochastic methods rely on the linear and Gaussian model structures for performing the diagnosis and the prognosis parts in an on-line health monitoring scheme. These methods may have proper results for fault diagnosis of nonlinear systems, but for prognosis, in which we are particularly interested in the long-term prediction of system health, they cannot necessarily guarantee convergence to an accurate solution with acceptable bounded error margins. Therefore, the nonlinear filtering methods can be used as solutions to overcome this problem. The considered application for this thesis is the gas turbine engine system.

Generally, as far as the assumptions for the damage model are concerned, the health monitoring problem is solved in two different levels according to two separate approaches. In the first approach, no dynamics is considered for the damage and the system health tracking is done based on the system health parameters estimation. In the second approach, the dynamics of the hidden damage is augmented to the system state equations with slower dynamics which leads to a two-time scale system formulation. In this direction, the following general problems can be considered:

• How can the damage/degradation model be embedded in the system equations within a mathematical formulation?

• How can the problem of system health estimation and prediction be addressed in this framework?

1.2 Literature Review

In this section the literature related to the aforementioned problems and the approaches we would like to follow in order to solve them is presented. The main approaches for establishing the prognostics framework are known as data-based and model-based methods [26, 27]. There exists also another approach in the literature known as hybrid approach which is a combination of model-based and data-based prognostic approaches [24, 28]. However, our main focus in this thesis is on model-based health monitoring and prognosis approaches, the main part of the literature review is devoted to these methods. As an extension to our proposed model-based prognosis and health monitoring approach, a hybrid prognosis approach has also been developed in this thesis. Therefore, a brief literature review regarding data-driven based methods in prognosis has also been presented in this section.

When the mathematical model of a system and its affected damage mechanisms are known, model-based prognostic methods have been proposed in the literature [29,30]. The model-based approaches for solving the prognosis problem are useful when mathematical physical model of the system and of the damage that affects it, are available. It should be noted that in model-based prognosis methods the type of damage and the corresponding affected component must be known, and damage identification is not considered in this framework [25, 31–41].

There are two frameworks for addressing the prognosis problem in model-based approaches. The first approach was developed by electrical engineers first, proposed in [25] and further developed with other researchers in [38–41]. They consider the following two main parts for

prognostics: (i) a joint state-parameter estimation problem, in which by using the model, the health of a system or its components can be determined based on the observations; (ii) a prediction problem in which the state-parameter distribution is simulated forward in time to compute RUL and end of life (EOL) [25,38,39,42–46]. In this approach, the fault diagnosis and the failure prognosis are integrated on a single framework. The Particle Filtering (PF) methodology is used as a general solution to a joint state-parameter estimation problem and for prediction problem where the prediction is made using hypothesized future inputs of the system. It is assumed that these future inputs are known in advance.

The second model-based prognosis approach is mainly developed by mechanical engineers [31–35, 47, 48] and mainly utilized to address the crack damage in rotary mechanical systems bearings, and battery discharge process propagation. They have used the idea of time scale separation in which damage is considered as a hidden slow process and causes non-stationary behavior in the dynamics of the fast observable system. The two main parts of this algorithm are: (i) damage tracking to estimate the changes in the slow variables of the system using a tracking function based on the reference model short term prediction error and (ii) prediction of the remaining useful life (RUL) of the system components based on tracking metrics and mathematical damage evolution models. Recursive methods for RUL prediction are also proposed in [33], [36].

1.2.1 Model-based Health Monitoring Methods Based on Particle Filters

System state estimation, as one of the main steps in health monitoring approaches, is a fundamental problem in control, signal processing, and fault diagnosis fields [49]. Investigations on both linear and nonlinear state estimation and filtering in stochastic environments have been an active area of research during the past several decades. Linear state estimation methods use a simpler representation of an actual nonlinear system and can provide an acceptable performance only locally around an operating point and in the steady state operational condition of the system. Kalman filter based methods are used for performing prognosis in [50, 51], and a multiple model moving horizon estimation algorithm is developed for online prediction of the system health in [52]. However, as nonlinearities of the system dynamics become dominant, the performance of linear approaches deteriorates and linear algorithms will not necessarily converge to an accurate solution. Although an optimal state estimation solution for linear filtering methods exists, nonlinear filtering methods suffer from generating sub-optimal or near-optimal solutions. Consequently, investigation of nonlinear estimation and filtering problems remain a challenging research area.

As mentioned earlier, linear filtering methods such as Kalman filters can have a suitable result for the estimation scheme and might be useful for diagnosis, but as the prediction horizon extends, the linear methods fail in predicting the nonlinear system behavior. Therefore, numerous studies have appeared in the literature to solve and analyze standard nonlinear filtering problems [53–59]. As a general classification these methods are grouped into the following [58]:

• Linearization methods (Extended Kalman Filter): the nonlinear problem is linearized in small

time steps and then the linear Kalman filter is applied [53].

• Approximation using finite-dimensional nonlinear filters: the filtering problem is solved by using approximation with exact nonlinear filters [54].

• Particle methods: the conditional distribution is approximated by utilizing a set of particles for which a resampling algorithm is applied at each time step when a new observation is available [55, 60, 61].

• Classical Partial Differential Equation (PDE) method: the Zakai equation which is a stochastic PDE (SPDE) is solved [53, 56, 62–64].

• Wiener chaos expansions: the Zakai equation is solved by means of decomposition into Wiener integrals [57].

• Moment methods: approximation of conditional distribution is achieved by using its moments [58].

Particle filter (PF) is one of the most popular recursive nonlinear state estimation methods which solves the Bayesian recursive relations by using Sequential Monte Carlo (SMC) methods [55, 60][1]. These methods are the best known methods for numerically approximating the solution of the filtering problem [58]. The PF-based methods are flexible and easy to implement [65] and they provide a general solution for the problem of state estimation in the nonlinear state space system equations that are described by the Bayesian recursive methods. The main challenge in the implementation of such methods appears when the system has a high dimension (large number of states) and hence simulating and storing a large number of particles are necessary in order to have a proper estimation of the system states. On the other hand, these methods

[1] SMC methods are a set of simulation-based methods that provide an approach to compute the posterior distributions of the states, therefore the statistical estimates can be easily computed [55].

suffer from the curse of dimensionality problem which causes particle degeneracy in a sample. As the system dimension increases, the particle degeneracy effect grows exponentially [65].

One of the most important recent applications of nonlinear filtering methods is in the area of fault diagnosis of dynamical systems that can include fault detection, isolation, and identification (FDII) modules. Diagnosis methods that are based on linearization techniques suffer from poor detection and high rates of false alarms. Therefore, Monte Carlo filtering approach based on particle filters was first proposed in [66] to address the fault detection and isolation problem of nonlinear systems. In this work, the negative log-likelihood, which is calculated for a predefined time window, is considered as a measure for the fault detection. The fault isolation was achieved by using the augmentation of the fault parameters vector to the system states to perform the estimation task. However, the augmented state space model tends to increase the dimensionality of the model and as a result increases the number of required particles for achieving a sufficiently accurate result. For decreasing the computational burden of this method, the augmented model is used only after the fault detection stage and for only the fault isolation stage. An external covariance adjustment loop was added to this augmented model in [25] to enable the estimation algorithm to track changes in the system parameters in case of fault occurrences.

The combination of a particle filtering algorithm and the log-likelihood ratio (LLR) test in the multiple model environment, has led to the development of sensor/actuator FDI scheme in [67] for a general class of nonlinear non-Gaussian dynamical systems but with the assumption of full state measurements. The fault detection problem recently is addressed for a mobile

robot based on the combination of the negative LLR test and particle filtering approach in [68]. However, both methods in [67] and [68] suffer from the high computational burden for on-line implementation of the algorithms. Hence, the idea of parallelized particle filters for on-line fault diagnosis is introduced in [68] to improve the algorithm performance.

A PF-based robust navigation approach was proposed in [69] to address multiple and simultaneous faults occurrences in both actuators and sensors in an underwater robot where an anomaly is modeled by a switching-mode hidden Markov system. The component and actuator fault detection and isolation of a point mass satellite was tackled in [70] by introducing several particle filters that run in parallel and each rejects a different subset of the faults. A fault-tolerant control strategy based on particle filter has been developed in [71] for unmanned aerial systems where the prognostic information has been used in the reconfiguration mechanism of the controller to increase the system reliability.

Generally, the main issues with applying standard particle filters to the fault diagnosis problem can be stated as follows [72]: (i) false diagnosis decisions due to low probabilities of transitions to fault states when there are fewer samples of states, and (ii) the exponential growth of the required samples for accurately approximating the *a posteriori* distributions as dimensionality of the estimation problem increases. The risk-sensitive PF is introduced to address the first problem and the variable resolution PF is developed to overcome the second problem in [73]. Moreover, Gaussian PF (GPF) is also introduced in [74] as an efficient algorithm for performing fault diagnosis of hybrid systems faster than traditional methods that are based on PFs. Finally, the sample impoverishment problem in particle filters due to fault occurrence in a hybrid sys-

tem is addressed in [75]. The developed algorithm enables the PF method to be implemented by fewer number of particles even under faulty conditions.

The on-line estimation of the system time-varying parameters by using particle filters is another challenging and active area of research which can be indirectly related to the health monitoring problem where the changes in the system health parameters can affect the state estimation accuracy. There are two main classes of PF-based parameter estimation algorithms (for on-line as well as off-line implementations) [76] known as Bayesian and maximum likelihood (ML) approaches. In the Bayesian approach, *a priori* distribution is considered for the unknown parameters and the *a posteriori* distribution of the parameters is approximated given the observations [77,78], whereas in the ML approach the estimated parameter is the maximizing argument of the likelihood function given the observations [79–82]. In the ML framework for parameter estimation, the maximization of any cost function can be performed based on gradient-based search methods [79]. On the other hand, expectation maximization (EM) methods are only applicable for maximization of the likelihood functions [82]. However, EM methods are not suitable for on-line applications due to their high computational cost for implementation. The recursive maximum likelihood method (RML) is recognized as a promising method for on-line parameter estimation based on a stochastic gradient algorithm [80]. In order to avoid the direct computation of the likelihood function gradient, an alternative method is proposed in [3] that is known as the gradient-free ML parameter estimation. Despite the above, the on-line ML methods suffer from the practical point of view of slow convergence rates and requiring large number of particles to achieve accurate estimates [83].

In the Bayesian framework, on-line implementation of particle filter-based parameter estimation algorithms are computationally intensive [84]. A general method that is capable of *simultaneously* estimating the static (i.e., constant or fixed) parameters and time-varying states of a system is developed in [2]. The work is based on the sequential Monte Carlo (SMC) method in which an artificial dynamic evolution model is considered for unknown model parameters. In order to overcome the degeneracy concerns arising from the particle filtering, kernel smoothing technique as a method for smoothing the approximation of the parameters conditional density has been utilized in [77]. The estimation algorithm is further improved by re-interpretation of the artificial evolution algorithm according to the shrinkage scaling concept. However, the proposed method in [2, 77] is only applicable for estimating *fixed parameters* of the system and it uses the augmented state/parameter vector for the estimation task.

Among all the model-based approaches for prognosis, particle filtering (PF) Monte Carlo schemes are considered as representing the state-of-the-art in failure prognosis [25,85,86]. Their capability for taking into account and incorporating system parameters as augmented states in the estimation scheme enables them to be suitable for uncertainty management in failure prognosis through joint state and parameter estimation modules. This functionality performs model adaptation along with the state tracking, and thus produces an adjusted model that can be used for long term predictions.

A comprehensive review over around 50 papers dealing with the application of particle filters in prognosis can be found in [87], in which a variety range of applications in different systems such as rotary machines, Li-ion battery, water tank, pneumatic valve, wind turbine, etc.

are discussed. According to this study, particle filters have a lot of advantages in the context of prognosis when dealing with nonlinear non-stationary models and with non-necessary Gaussian noises. The application of particle filters for health monitoring of Li-ion batteries when they are affected by aging and degradation is also investigated in [88].

A statistical characterization of consumption profiles for Li-Ion batteries is addressed in [89] where a state space model is used to modify the observation equation that incorporates most of the non-linearities in the battery curves. This modification improves the convergence of the state estimate and provides suitable initial conditions for the prognosis stage. However, particle filters are subjected to exponential growth of computational complexity by increasing the state dimension. Several methods are developed to overcome the curse of dimensionality in particle filters [42–46]. Moreover, particle filters are not suitable for multi-parameter estimations in the case of multi-damage problems and modifications are supposed to be done to utilize them for this purpose. The model-based prognosis method based on PF is also used for addressing the problem of fatigue damage and crack propagation in a turbine blade [25, 44].

It is important to note that the predictive capabilities of particle filters are limited and can be used only for learning the current health of the system. Therefore, lots of issues regarding the choice of the particle filter and its adaptation to the requirements of a specific industrial system can be raised [87]. Hence, an uncertainty management system for a long-term prediction horizon using particle filters was developed in [25] by utilizing the invariant particle weights for future propagation of the particles whereas the regularization of particles is utilized to characterize the future uncertainties by modifying the position of the particles. The accuracy of

the prediction algorithm is shown to be improved by employing an outer correction loop to modify the hyper-parameters of the process noise used in the nonlinear dynamical model of the system specifically in the artificial evolution law utilized for estimating the system unknown parameters.

The outer feedback correction loops in particle filtering based prognosis approaches have been introduced in several works [38, 90, 91]. On the other hand, in another simpler prediction approach in terms of computational efforts in [25], the particle weights are considered fixed for future propagation since it was assumed that the error that can be generated by considering the invariant particle weights for future time instants is negligible as compared to the other sources of error that may affect the system in the practical application. A Bayesian approach has been developed in [92] to address the uncertainty management problem in online condition-based health monitoring based on the principle of subjective probability. Furthermore, for systems with large number of states (as well as parameters) particle filters are not the best choice for performing estimation.

1.2.2 Model-Based Health Monitoring Approaches Based on Two-Time Scale Systems

The literature review related to health monitoring based on two-time scale formulation of dynamical systems including damage mechanism [31–35, 47, 48] can be limited to the progresses that have been made so far in the development of estimation and prediction methods for such systems. Therefore, we have mainly reviewed the estimation methods developed for such sys-

tems which can also be applicable to fault diagnosis and failure prognosis schemes.

Singular Perturbation and Two-Time Scale Systems Overview

Two-time scale systems, also known as singularly perturbed systems are quantified by a discontinuous dependence of the system properties on a small perturbation parameter that is usually denoted by ϵ. Many physical systems, such as electrical power systems, electronic systems, mechanical systems, biological systems, economical systems and Quantum physics are examples of singularly perturbed systems. These systems do exhibit a two-time scale behavior known as the fast and slow dynamics. The two-time scale property makes the analysis and control of these systems more complicated than conventional regular systems [93, 94].

Study of systems with two-time scale separation is necessary for development of the next generation of health monitoring and condition based maintenance methods [31, 37]. For example, micro cracks in a spinning shaft, the misalignment of machinery parts during operation, corrosion process in the system components, and moisture accumulation in the composite materials of electrical circuits, etc. can be modeled as two-time scale systems [37].

Most of the significant work that have been conducted on singularly perturbed systems are related to more than a decade ago which were mainly concentrated on stability analysis and control of theses systems due to their important role in control system theory [94]. The filtering problem for singularly perturbed systems has been considered as a challenging issue that cannot be easily investigated through the deterministic observer based methods [93, 94]. Nevertheless, several works have been conducted in order to address the stochastic filtering problem in linear

singularly perturbed systems [95–98].

The problem of linear filtering in linear stochastic singularly perturbed systems was first considered in [95] in which the main estimation framework is developed for continuous-time systems with a composite type of filter. Exact decomposition of the fast and slow states in design of Kalman filters was proposed in [96, 97] as per the decomposition approach in [99]. The filtering methodology based on fast-slow decomposition of Kalman filter gains has also been addressed in [98].

Although, most of the work on singularly perturbed systems have been developed for continuous-time systems, discrete-time singularly perturbed systems have also been extensively studied in [100, 101]. Various filtering methods for linear discrete-time singularly perturbed systems have been proposed in [102, 103] based on the decomposition approach and in [104] based on outer and inner series for Kalman filter gain approximations in composite structures. Utilization of H_∞ concept in linear filtering theory has led to the development of H_∞-based filtering methods for state estimation in linear singularly perturbed systems as in [98, 105].

The nonlinear filtering problem of nonlinear singularly perturbed systems has been investigated in only a few works [106–109]. In [106] sufficient conditions for solvablity of the filtering problem in nonlinear singularly perturbed systems is obtained based on H_2/H_∞ approaches, nevertheless the approximations to the filter gains were not addressed. In [107, 108], the authors proposed a hybrid homogenized method based on the particle filter approach to approximate the nonlinear system states. This method is computationally very complex and its complexity grows exponentially as the number of states increases, and therefore it is not computationally

practical or efficient.

In [109], we have investigated the filtering problem in nonlinear singularly perturbed systems by using a hybrid robust extended Kalman filter approach. However, this method is not capable of achieving accurate prediction results in the framework of health monitoring problem as the prediction horizon time is extended.

Ensemble Kalman Filters (EnKF) Overview

The main problem associated with nonlinear filtering methods which rely on linearization as in extended Kalman filter (EKF) is that they characterize the distribution of the state only by its first and second moments (the same as in the linear case) and discard the higher order moments [110]. Although several methods have been proposed to address the estimation problem in nonlinear systems, the related results are either too narrow in applicability or are computationally expensive [54, 111, 112]. As a result, a numerous range of suboptimal methods have been developed for practical applications [110, 113, 114].

On the other hand by using Monte Carlo based nonlinear estimation methods, such as particle filters and ensemble Kalman filtering (EnKF) method, one can derive the Fokker-Planck partial differential equation for the time evolution of the probability density function which includes all the required information related to prediction error statistics [115, 116]. In other words, the EnKF can be considered as an extension of the classical Kalman filter to large scale nonlinear systems. It works by propagating an ensemble of N members which capture the mean and covariance of the current state estimate [117]. Our main motivation for choosing

EnKF method to develop a new two-time scale filter for addressing the health monitoring problem is related to the capability of EnKF approach in decreasing the dimensionality of the system dynamics as the number of the system states increases.

A comprehensive survey on the EnKF is conducted in [116], where the EnKF is introduced as a suboptimal solution for the general Bayesian problem of finding the *a posteriori* distribution of the states given the *a priori* state estimation and the observation densities (Gaussian densities). However, in [118] the convergence of the ensemble Kalman filter in the limit for large ensembles to the Kalman filter is shown. The main application of the EnKF is identified to be in atmospheric data assimilation, since one is dealing with high-dimensional states (large number of states). Unfortunately, EnKF estimation approach has not been studied extensively outside of this specific application domain and only a few works have been conducted outside of the weather forecasting and oceanography related applications [119, 120].

The ensemble Kalman filter has a large group of users and numerous research has been conducted on the application and theoretical aspects of this estimation and data assimilation method [115, 121–123]. The EnKF is related to the particle filter approach where a particle represents an ensemble member. The main difference between these two filters is on the assumption that all the probability distributions involved in the EnKF are Gaussian. In circumstances that this assumption is applicable, the EnKF method is more efficient than the particle filter [116].

The ensemble Kalman filter is a Monte Carlo approximation method for the Bayesian update problems. There are around one hundred different implementations for the EnKF [122]. In the original Kalman filter, it is assumed that all the probability distribution functions (pdfs) are

Gaussian and the change of the mean and covariance matrices are linear. However, storing the covariance matrix for advancing it in time is not computationally feasible for high-dimensional systems (high order systems). For this reason, EnKFs were developed [124].

In the EnKF method, the distribution of the system state is represented by selecting a collection of state vectors, that is designated as an ensemble, and by replacing the covariance matrix by the sample covariance which is computed from the ensembles. Consequently, advancing the probability density function in time can be achieved by simply advancing each ensemble member [125]. The main advantage of the EnKF approach over the classical Kalman filter as well as extended Kalman filter (EKF) methods is that it does not require any model linearization and can also be used to assimilate asynchronous observations. However, its main disadvantage is considered to be a possible dynamic imbalance and sub-optimality [126].

As stated above, the computational cost in implementing a Kalman filter for large scale systems is rather high. In order to overcome this challenge several methods have been proposed in the literature based on the idea of reduced estimation. There are two ways to obtain a reduced rank estimate of the *a priori* error covariance matrix [123, 127, 128]. There are those methods that are based on linearizatin of the model dynamics to reduce the rank of the *a priori* covariance matrix by projecting the model state on to a basis that has a much lower dimensionality than the full model space [129,130]. The main reason for using EnKF in data assimilation applications is due to the ease of its implementation and the low computational cost and storage requirements [120].

In the other set of approaches, a relatively small set of ensembles are used to estimate the *a*

priori error covariance [125, 131]. The ensembles are operated in such a manner that they are random samples, however the ensemble members are actually not independent and the EnKF will fuse them appropriately. The advantage of this method is that the advancing of the pdfs in time is achieved by simply advancing each ensemble member individually.

In the EnKF-based state estimation method, the *a priori* ensembles are generated by propagating the ensembles of initial conditions which are distributed according to the results of the previous analysis [123]. The generation of the *a posteriori* ensemble members can be achieved through different methods. One group is based on perturbed observations [116]. In this approach, *a posteriori* ensemble is obtained by assimilating a different set of observations to each *a priori* ensemble member. Different sets of observations are created by adding random noise to real observations, where this random noise is generated according to the observational error covariance matrix.

In another group of methods, such as Kalman square-root filters, the analysis for *a posteriori* state update is performed only once to obtain both the *a posteriori* state estimation mean and the error covariance matrix. Subsequently, the *a posteriori* ensemble perturbations (to the mean of the analysis) are generated by transforming the *a priori* ensemble perturbations to a set of vectors that can represent the *a posteriori* error covariance matrix. Therefore, the *a posteriori* analysis is rendered to the subspace of ensembles. Since there is an infinite set of *a posteriori* perturbations that can be used to represent the *a posteriori* error covariance matrix, numerous methods can be applied following the works in [124, 132–134]. An iterative extension to the ensemble Kalman filter has been developed in [135] to improve the estimation capabilities of the

filter in case that the relationship between the measurements and the system states is nonlinear.

One of the main goals in this thesis is to develop the EnKF estimation framework for two-time scale systems known as singularly perturbed systems. Inspired from the local EnKF method proposed in [123] in which the idea of covariance localization is proposed, we take advantage of this covariance definition in order to reduce the dimension of the covariance matrix in our estimation scheme to develop the Kalman filter in the dominant direction of the state space (slow time scale) which results in a reduced ensemble size as well. Then, a correction is made to the estimated slow states by taking into account the effects of fast states of the system whereas the remaining system states are also estimated.

1.2.3 A Brief Literature Review on Data-Based and Hybrid Health Monitoring Approaches

In this thesis, as mentioned earlier, the main focus is on developing new algorithms and/or methods to enhance the existing model-based health monitoring approaches. Therefore, the data-based health monitoring and prognosis approaches are not reviewed in detail. The main reason that we intend to state a brief literature review over such methods is that in part of this thesis our novel proposed hybrid health monitoring approach is introduced as a bridge between data-based and model-based methods.

The data-driven based prognosis and health management as well as condition-based monitoring approaches have been recently studied in detail in [136, 137] for different engineering

applications. Data-based prognosis approaches can be based on statistical methods or neural network schemes [138–141]. The bond graph (BG) modeling framework has been utilized in [142] where parametric uncertainty is modeled in the interval form. The system parameter is assumed to be affected by known *a priori* degradation model. Therefore, the prognostics problem is addressed as joint state-parameter estimation problem where the degradation detection is achieved based on a passive manner. In [143] various data-driven techniques have been compared for estimating the remaining useful life (RUL) of Li-ion batteries. The utilized data-driven methodologies include neural networks, neuro-fuzzy networks, group method of data handling, and random forests as an ensemble-based system. It is shown that the random forests and neuro-fuzzy techniques have superior performance in terms of the RUL prediction error and root mean square error.

Among prognosis methods that are concerned with avoiding time-based maintenance alternatives, the rotary machinery systems are known to be an emerging field of application [20–23, 144–146]. The PHM is also considered as the main step in the condition-based maintenance technology, known as the CBM for programming the maintenance policies in rotary machinery systems [147, 148]. Given the importance of rotary machinery systems, the main application in this thesis is considered to be a gas turbine engine where the compressor and the turbine components are considered to be affected by degradation damages.

In the prognosis framework, the future health of the system is predicted in order to determine the RUL of the system or its components before the failures occur [6]. However, as mentioned earlier, the performance of the prognosis algorithm is closely related to the accuracy

of the employed prediction method. This performance can be affected by activities such as maintenance actions [24]. In order to achieve a reliable health prediction for prognosis, several data-driven and soft computing or computationally intelligence methods have been developed in the literature [136, 137]. Data-driven and computational intelligence-based approaches tackle the PHM task by considering the prediction error (where the prediction is accomplished by a computational intelligent method) [149].

In other words, the discrepancy between the observation and its prediction is the only information that is used for the purpose of developing the PHM scheme. Methods which are based on neural networks and fuzzy logic are utilized in [150, 151]. In [150], uncertainty is represented through a confidence distribution and managed by a learning procedure for the prediction step to predict the system RUL.

In [151] the residual life of the system is predicted through projecting the fault estimate that affects the system RUL. Given that data-driven and computational intelligent-based methods do not require to consider the dynamical model of the system for developing a prediction scheme [152], it is therefore reasonable that one would require a large set of historical data. However, the main challenge in data-driven and computational intelligent-based prognostic methods is the necessity of assuming sufficient available system historical data. This limitation of intelligent based methods is one of the features to be addressed in the hybrid health monitoring structures by combining intelligent-based prognostic methods with model-based algorithms [28, 153].

On the other hand, in model-based prognostic methods it is assumed that the mathematical model of the system and its affected damage mechanisms are known and available [29]. How-

ever, the mathematical model of a complex industrial system can be very complicated such that in most of the approaches a simpler dynamics of the system which consists of fewer system dynamics as compared to the real system, is considered for further investigation. Therefore, ignoring some of the system dynamics without considering their effect on the other dynamic of the system can lead to erroneous estimation and prediction results in real applications. Therefore, the utilization of hybrid methods which are the combination of both model-based and intelligent based methods can potentially achieve more accurate results in the estimation and prediction of the system health indicators.

Neural networks are known as effective tools for designing fault-tolerant control schemes for MIMO discrete-time systems via online reinforcement learning algorithms [154]. By taking advantage of the strength and capabilities of both model-based and computational intelligent-based methods, a hybrid structure is expected to achieve a more robust health prediction result that could lead to a more reliable RUL estimate as compared to utilizing only one of the above methods alone. In [155] a machine condition prediction method based on adaptive neuro-fuzzy (ANFIS) and high order particle filters has been proposed in the framework of hybrid prognosis scheme. In this method the ANFIS constitutes a hidden Markov model to describe the fault propagation process, therefore development of the ANFIS requires a large amount of historical data to train the network for different fault scenarios. The high order particle filter in this method is only utilized for predicting the time evolution of the fault indicator in the long term time horizon. In [156], multilayer perceptron (MLP) and recurrent networks are used for performing filtering and smoothing purposes (to increase the accuracy of the PHM system), and it was shown that recurrent networks provide more promising results.

1.3 General Problem Statement and Thesis Outline

The main objective of this thesis is to develop solid frameworks for health monitoring and prognosis of nonlinear systems based on Monte Carlo nonlinear estimation methodologies. For this purpose, two modeling frameworks are introduced. The first one which is based on dual state/parameter estimation would not increase the computational cost of the estimation problem. On the other hand, the second approach which is based on modeling the damage mechanism with slower dynamics as compared to the system dynamics, results in a two-time scale system formulation with augmented states (damage model). Therefore, the problem of health tracking for such systems is converted to the problem of state estimation in singularly perturbed systems. The main concerns in the development of the proposed frameworks are as follows:

1. Development of a general model structure for the damage mechanism which is capable of representing a broad range of damages affecting the nonlinear system. The damage mechanism is considered to be known in health monitoring frameworks for the prognosis purpose. Hence, the methods that have been developed so far in the literature can only work for a specific damage model.

2. Establishment of the entire health monitoring framework in stochastic environment due to the fact that the damage model cannot be defined precisely in a deterministic mode.

3. Estimation of the time-varying health parameter of the system, which is an active research area in estimation domain, in order to develop an online fault diagnosis and failure prognosis framework for nonlinear systems.

4. The accurate prediction of the system health based on the available observations in order to overcome uncertainties that are originated from different sources such as model inaccuracy, estimation error, assumed Gaussian noise model for the process, etc.

5. The developed frameworks should be capable of addressing more than one damage mechanism, when the system is subjected to more than one type of damage (multi-damage mechanism).

In Chapters 2 to 6 of this thesis we try to address the aforementioned challenges related to health monitoring and prognosis and improve the present methods in the literature. The main focus is on health monitoring of rotary machinery systems (gas turbine application). These very important and expensive systems are subjected to degradation damage due to their continuous operation. Moreover, the main part of our research is devoted to system health tracking and prediction. In what follows, a summary of the thesis chapters is given.

In Chapter 2, the background information and models related to the main case study utilized in this thesis, i.e., gas turbine engine and its degradation mechanism formulation are reviewed.

In Chapter 3, a novel dual estimation methodology is developed for both time-varying parameters and states of a nonlinear stochastic system based on the recursive prediction error (RPE) concept and the particle filtering scheme. Estimation of the system parameters along with the states generate an updated model that can be used for a long-term prediction problem. The developed estimation methodology is utilized to address the component fault diagnosis problem in a nonlinear system when it is assumed to be affected by multiple faults.

In Chapter 4, an improved particle filtering-based methodology is developed to address the prediction step within the developed health monitoring framework. In this method an observation forecasting scheme is developed to extend the system observation profiles (as time-series) to future time horizons. Particles are then propagated to future time instants according to a resampling algorithm in the prediction step. The uncertainty in the long-term prediction of the system states and parameters are managed by utilizing dynamic linear models (DLM) for development of an observation forecasting scheme. The developed particle filtering-based methodology in this chapter has been quantified in terms of the algorithm computational cost (for the implementation) as compared to standard prediction methods based on particle filters.

In Chapter 5, which is a joint work with my colleague Dr.Baniamerian, as an extension to the prediction method developed in Chapter 4, a hybrid architecture is proposed to develop prognosis and health monitoring methodologies for nonlinear systems by integration of model-based and computationally intelligent-based techniques. Our proposed hybrid health monitoring methodology is constructed based on a framework that is not dependent on the structure of the neural network model utilized in the implementation of the observation forecasting scheme. Moreover, changing the neural network model structure in this framework does not significantly affect the prediction accuracy of the entire health prediction algorithm.

In Chapter 6, a method for formulation of health monitoring problem of dynamical systems through a two-time scale decomposition is introduced. For this methodology the system dynamical equations as well as the affected damage model are investigated in the two-time scale system health estimation and prediction steps. A two-time scale filtering approach is developed

based on the ensemble Kalman filtering (EnKF) methodology by taking advantage of the model reduction concept. The performance of the proposed two-time scale ensemble Kalman filters is shown to be more accurate and less computationally intensive as compared to the well-known particle filtering approach for this class of nonlinear systems.

Finally, all of our developed methods have been applied for health monitoring and prognosis of a gas turbine engine when it is affected by various degradation damages. Extensive comparative studies are also conducted to validate and demonstrate the advantages and capabilities of our proposed frameworks and methodologies using MATLAB as a powerful design engineering software.

1.4 Thesis Contributions

In this thesis, the health monitoring and prognosis problem of nonlinear systems is tackled. The main contributions of this thesis are as follows:

1. Development of a unified model-based framework for health monitoring, diagnosis, and prognosis of nonlinear systems based on particle filters which consists of the following principal steps:
   (a) Propose a general modeling strategy for damage mechanism that affects the system health parameters which are themselves a function of the system hidden states (non-measurable states).
   (b) Development of a dual state and parameter estimation methodology based on par-

ticle filters to address the nonlinear system health tracking step in the health monitoring and prognosis problem, when the system health parameters are affected by time-varying damages.

(c) Extend the developed dual estimation method to predict the future health of the affected nonlinear system. This methodology is developed by incorporating the dynamical linear models (DLM) for Bayesian forecasting of uni-variate time-series in an observation forecasting module which is enhanced to the particle filtering-based dual estimation method.

2. Development of a hybrid framework for health monitoring and prognosis methodology by extending the previously developed particle filtering-based prediction strategy and incorporating nonlinear time-series forecasting methods based on neural networks as opposed to linear time-series methodologies.

3. Develop a solid health monitoring and prognosis framework according to two-time scale formulation strategy using the ensemble Kalman filtering (EnKF) approach:

   (a) Introduce a new strategy to incorporate the hidden damage model in the nonlinear system dynamics by utilizing the singular perturbation theory.

   (b) Develop a two-time scale ensemble Kalman filter (EnKF) methodology to address the system health tracking and prediction steps in the health monitoring and prognosis problem.

Chapter 2

Background Information

As the main case study in this thesis, all of the developed health monitoring and prognosis approaches have been applied to a gas turbine engine model as a complex industrial system. The main reason to choose the gas turbine engine is related to the increasing demand on aerospace industry that has resulted in higher usage of aircraft engines. The growth rate in usage have caused a faster aircraft engine deterioration and considering the importance of safety, it is essential to predict the effects of the engine deterioration, which helps improve the engine utilization. Engine degradation is due to different damages that can change the specific thrust, fuel consumption, spool speed and turbine entry temperature. More serious effects of damage can cause a shorter engine life.

In the past recent years there has been an increasing interest in the field of prognosis and predicting the remaining useful life of jet engines components which can result in better safety

and less expenses by avoiding late or early replacement of the components. In the prognosis problem, in addition to the component faults, damages occurred to the components due to unbalanced operating conditions of the system and also aging effects of the components must be considered. The damage has a slower dynamics compared to fault and the aging effect can accelerate the damage propagation process. The damage itself is not a fault but if it is accumulated during time it can lead to failure of the entire system.

The main causes of degradation in a jet engine system can be categorized as erosion, corrosion, fouling and thermal distortion which can initiate and/or accelerate creep, low-cycle fatigue, high-cycle fatigue and thermal fatigue damages. These kinds of damages may lead to crack initiation and propagation in turbine blades. In a gas turbine engine many components are subject to deterioration but only a few of them have a significant impact on the engine life. These are rotating components which are subject to cyclic and steady-state stresses. The turbine blade is a very important part because it is under both highest rotating speed and gas temperature.

In this thesis, we study the effects of fouling and erosion phenomena as the main engine performance degradation causes and consequently their effects on the life consumption of the engine turbine component.

2.1 Model Overview

The mathematical model of a gas turbine used in this paper is a single spool jet engine as depicted in Figure 2.1 that was developed in [157, 158]. The four engine states are the combustion

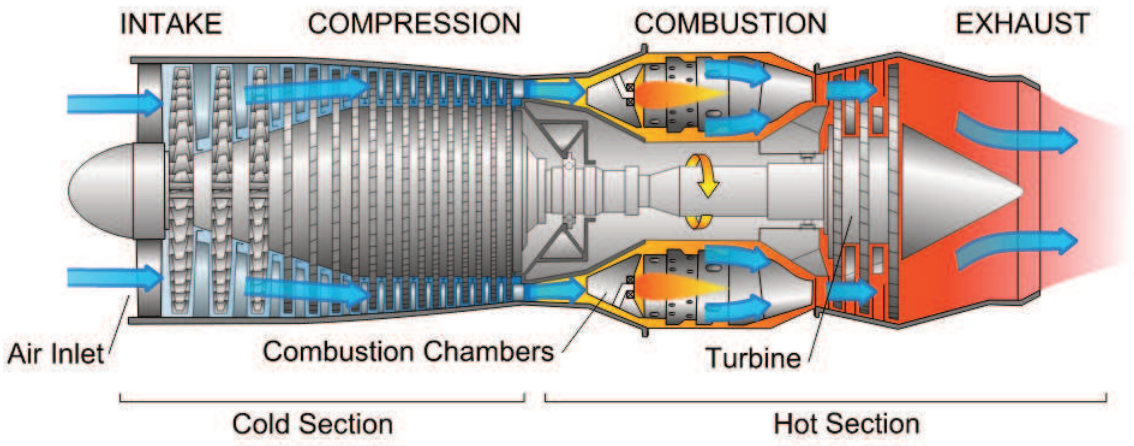


Figure 2.1: Diagram of a typical gas turbine jet engine (Photo credit: Wikipedia).

chamber pressure and temperature, $P_{\rm CC}$ and $T_{\rm CC}$, respectively, the spool speed S, and the nozzle outlet pressure $P_{\rm NLT}$. The continuous-time state space model of the system is given as follows,

$$\begin{aligned}
\dot{T}_{\rm CC} &= \frac{1}{c_{\rm v}\dot{m}_{\rm cc}}[(c_{\rm p}T_{\rm C}\dot{m}_{\rm C} + \eta_{\rm CC}H_{\rm u}\dot{m}_{\rm f} - c_{\rm p}T_{\rm CC}\dot{m}_{\rm T}) - c_{\rm v}T_{\rm CC}(\dot{m}_{\rm C} + \dot{m}_{\rm f} - \dot{m}_{\rm T})],\\
\dot{S} &= \frac{\eta_{\rm mech}\dot{m}_{\rm T}c_{\rm p}(T_{\rm CC} - T_{\rm T}) - \dot{m}_{\rm C}c_{\rm p}(T_{\rm C} - T_{\rm d})}{{\rm J}S(\frac{\pi}{30})^2},\\
\dot{P}_{\rm CC} &= \frac{P_{\rm CC}}{T_{\rm CC}}\frac{1}{c_{\rm v}\dot{m}_{\rm cc}}[(c_{\rm p}T_{\rm C}\dot{m}_{\rm C} + \eta_{\rm CC}H_{\rm u}\dot{m}_{\rm f} - c_{\rm p}T_{\rm CC}\dot{m}_{\rm T}) - c_{\rm v}T_{\rm CC}(\dot{m}_{\rm C} + \dot{m}_{\rm f}\\
&\quad - \dot{m}_{\rm T})] + \frac{\gamma{\rm R}T_{\rm CC}}{V_{\rm CC}}(\dot{m}_{\rm C} + \dot{m}_{\rm f} - \dot{m}_{\rm T}),\\
\dot{P}_{\rm NLT} &= \frac{T_{\rm M}}{V_{\rm M}}(\dot{m}_{\rm T} + \frac{\beta}{\beta+1}\dot{m}_{\rm C} - \dot{m}_{\rm Nozzle}),
\end{aligned} \tag{2.1}$$

where the physical significance of all the model parameters is provided in Table 2.1. The five gas turbine measured outputs are considered to be the compressor temperature (y_1), the combustion chamber pressure (y_2), the spool speed (y_3), the nozzle outlet pressure (y_4), and the turbine temperature (y_5), namely

$$\begin{aligned}
y_1 &= T_{\rm C} = T_{\rm diffuser}[1 + \frac{1}{\eta_{\rm C}}[(\frac{P_{\rm CC}}{P_{\rm diffuzer}})^{\frac{\gamma-1}{\gamma}} - 1]],\\
y_2 &= P_{\rm CC},\ y_3 = S,\ y_4 = P_{\rm NLT},\\
y_5 &= T_{\rm CC}[1 - \eta_{\rm T}(1 - (\frac{P_{\rm NLT}}{P_{\rm CC}})^{\frac{\gamma-1}{\gamma}}].
\end{aligned} \tag{2.2}$$

Table 2.1: Model Parameters Description

parameter	description	parameter	description
c_{v}	Specific heat at constant pressure, $\frac{\mathrm{J}}{\mathrm{kg.K}}$	T_{T}	Turbine temperature, K
c_{p}	Specific heat at constant volume, $\frac{\mathrm{J}}{\mathrm{kg.K}}$	T_{d}	Intake temperature, K
$\dot{m}_{\mathrm{cc}}$	Combustion chamber mass flow rate, $\mathrm{kg/s}$	J	Rotor moment of inertia, $\mathrm{kg.m^2}$
T_{C}	Compressor temperature, K	R	Gas constant, $\frac{\mathrm{J}}{\mathrm{kg.K}}$
H_{u}	Fuel specific heat, $\frac{\mathrm{J}}{\mathrm{kg}}$	γ	Heat capacity ratio
η_{CC}	Combustion chamber efficiency	V_{CC}	Combustion camber Volume, $\mathrm{m^3}$
$\dot{m}_{\mathrm{f}}$	Fuel flow, $\mathrm{kg/s}$	T_{M}	Mixer temperature, K
$\dot{m}_{\mathrm{T}}$	Turbine mass flow rate, $\mathrm{kg/s}$	V_{M}	Mixer volume, $\mathrm{m^3}$
η_{T}	Turbine efficiency	$\dot{m}_{\mathrm{nozzle}}$	Nozzle mass flow rate, $\mathrm{kg/s}$
$\dot{m}_{\mathrm{C}}$	Compressor mass flow rate, $\mathrm{kg/s}$	P_{diffuzer}	Diffuzer pressure, bar
η_{C}	Compressor efficiency	T_{diffuzer}	Diffuzer temperature, K
η_{mech}	mechanical efficiency	β	bypass ratio

2.1.1 Degradation Model Description

In this subsection degradation model that can be originated from different sources in the single spool jet engine system are discussed. The introduced models are utilized as a bench test for studying health monitoring and prognosis of the system components based on model based approaches when empirical data are not available. To validate the developed degradations caused by fouling and erosion, GSP software is used [159]. This software is a powerful tool to study the behavior of the jet engine system.

The main causes of degradation in gas turbine engine system can be generally classified as follows [160]:

1. Fouling:

Fouling is introduced as accumulation of unwanted particles on solid surfaces which cause the degradation of flow capacity and efficiency in jet engine. Fouling can be formed in different parts of the air path (stators, guide vanes and blades) and affects the aerodynamic behavior of the system and eventually reduce its flow rate. As a result, reduction of power, loss of efficiency and increase the fuel consumption, are the consequences of fouling in a jet engine system [161].

Fouling phenomena mostly occurs in compressor part. Decreasing the mass flow area and efficiency will result in the engine performance reduction and also increase in either the rotational speed or turbine entry temperature (TET) in order to maintain the required thrust. In the present model providing the required thrust is achieved by an increase in the turbine entry temperature.

These factors together will cause a shorter remaining useful life time of the engine [160]. To represent the fouling effect on the engine performance, fouling index (FI) is used based on the work of Naeem [162]. This index is determined based on the reduction ratio of $1:2$ for compressor mass flow rate to compressor efficiency, and it is presented by FI. For example $\mathrm{FI} = 1\%$ means 0.5% reduction in mass flow rate while 1% reduction in the compressor efficiency. Therefore, by applying the effects of the fouling to the system dynamics (2.1) and output equation (2.2) result in manipulating the compressor efficiency and mass flow rate as follows

$$\eta_{\mathrm{C}} \to (1 - \mathrm{FI}(t))\eta_{\mathrm{C}}, \text{ and } \dot{m}_{\mathrm{C}} \to (1 - 0.5\mathrm{FI}(t))\dot{m}_{\mathrm{C}} \tag{2.3}$$

2. Erosion:

Erosion is defined as the removal of the material from the flow path components by hard particles that can cause aerodynamic changes in the behavior of the blades [161]. This phenomenon will result in increasing the pressure losses, performance degradation and even blade failure. Erosion can reduce up to 5% of performance in compressor or turbine and consequently the engine life [160]. Erosion Index (EI) is applied as a linear degradation per cycle.

To represent the erosion effect on the engine performance in a quantitative way, the erosion index is determined and is represented by EI. This index is determined based on the ratio of the reduction of turbine efficiency and increase in the turbine mass flow with the ratio of 1:2. For example if $\mathrm{EI} = 1\%$ means 0.5% increase in mass flow rate while 1% decrease in turbine efficiency [160]. Erosion Index is also applied as a linear degradation per cycle. Thus, by applying the effects of the erosion to the system dynamics (2.1) and output equation (2.2) result in manipulating the turbine efficiency and mass flow rate as follows

$$\eta_{\mathrm{T}} \to (1 - \mathrm{EI}(t))\eta_{\mathrm{T}}, \text{ and } \dot{m}_{\mathrm{T}} \to (1 + 0.5\mathrm{EI}(t))\dot{m}_{\mathrm{T}} \tag{2.4}$$

It must be mentioned that to maintain a constant maximum take-off thrust in the degraded engine during cycles of operation, fuel flow injection to the combustion chamber has to be increased to have higher temperature in the turbine inlet. So the amount of increase in the fuel flow for each cycle is approximated through a PID mechanism based on the error between the desired pressure ratio which generates the desired thrust and the calculated pressure ratio of the engine in different scenarios.

2.2 Summary

In this chapter, the background information related to the main case study in this thesis, namely, gas turbine engine are presented. The introduced models have been extensively used in other chapters of this thesis to validate the developed methodologies for the proposed health monitoring and prognosis frameworks. The fouling and erosion phenomena have also been introduced and formulated in this chapter to be utilized as the degradation damage that affects the gas turbine engine health condition which can lead to failure of the entire system.

Chapter 3

Particle Filter-Based Fault Diagnosis of Nonlinear Systems Using a Dual Particle Filters Scheme

In this chapter, a dual estimation methodology is developed for both time-varying parameters and states of a nonlinear stochastic system based on the Prediction Error (PE) concept and the Particle Filtering (PF) scheme. In this method we utilize nonlinear Bayesian and Sequential Monte Carlo (SMC) methods to develop, design, analyze, and implement a unified framework for both the state and parameter estimation as well as fault diagnosis problems of nonlinear systems. An on-line parameter estimation scheme is developed inspired from the recursive prediction error (RPE) method by using the particle filters (PF) approach. Specifically, by using

the prediction error to correct time-varying changes in the system parameters, a novel method is proposed for parameter estimation of nonlinear systems based on the PF. In the implementation of our proposed scheme, a dual structure for both state and parameter estimation is developed within the PF approach. In other words, the hidden states, and variations of the system parameters are estimated through two concurrent filters. Convergence and stability of our proposed dual estimation strategy are shown to be guaranteed formally under certain conditions.

The proposed dual estimation framework is then utilized for addressing the challenging problem of fault diagnosis of nonlinear systems. The performance capabilities of our proposed fault diagnosis methodology are demonstrated and evaluated by its application to a gas turbine engine through accomplishing state and parameter estimation under simultaneous and concurrent component fault scenarios. The health parameters of the system are considered to be slowly time-varying during the engine operation. Extensive simulation results are provided to substantiate and justify the superiority of our proposed fault diagnosis methodology when compared with another well-known alternative diagnostic technique that is available in the literature.

The main contributions of this chapter are now summarized as below:

1. Propose a general modeling strategy for damage mechanism that affects the system health parameters which are themselves a function of the system hidden states.

2. Development of a dual state and parameter estimation methodology based on particle filters to address the nonlinear system health tracking step in the health monitoring and prognosis problem, when the system health parameters are affected by time-varying dam-

ages.

The remainder of this chapter is organized as follows. In Section 3.1, the statement of the nonlinear filtering problem is presented. Our proposed dual state/parameter estimation scheme is developed in Section 3.2, in which state and parameter estimation methods are first developed concurrently and subsequently integrated together for simultaneously estimating the system states and parameters. The stability and convergence properties of the proposed schemes under certain conditions are also provided in Section 3.2. Our proposed fault diagnosis framework and formulation are also provided in Section 3.2. In Section 3.3, extensive simulation results and case studies are provided to demonstrate and justify the merits of our proposed method for fault diagnosis of a gas turbine engine under simultaneous (the same time in all components) and concurrent (one component after the other) component faults. Finally, the chapter is concluded in Section 3.4.

3.1 Problem Statement

The problem under consideration is to obtain an optimal estimate of states as well as time-varying parameters of a nonlinear system whose dynamics is governed by a discrete-time stochastic model,

$$x_{t+1} = f_t(x_t, \theta_t, \omega_t), \tag{3.1}$$

$$y_t = h_t(x_t, \theta_t) + \nu_t, \tag{3.2}$$

where $x_t \in \mathbb{R}^{n_x}$ is the system state, $t \in \mathbb{N}$, $f_t : \mathbb{R}^{n_x} \times \mathbb{R}^{n_\theta} \times \mathbb{R}^{n_\omega} \longrightarrow \mathbb{R}^{n_x}$ is a known nonlinear function, $\theta_t \in \mathbb{R}^{n_\theta}$ is an *unknown* and possibly *time-varying* parameter vector governed by

an *unknown dynamics*. The function $h_t : \mathbb{R}^{n_x} \times \mathbb{R}^{n_\theta} \longrightarrow \mathbb{R}^{n_y}$ is a known nonlinear function representing the map between the states, parameters and the system measurements, and ω_t and ν_t are uncorrelated stochastic process and measurement noise sequences with covariance matrices L_t and V_t, respectively. The following assumption is made regarding the dynamical system (3.1) and (3.2).

Assumption 3.1. The vector $\{x_t, \theta_t\}$ ranges over a compact set denoted by $D_{\mathcal{N}}$, for which the functions $f_t(x_t, \theta_t, \omega_t)$ and $h_t(x_t, \theta_t)$ are continuously differentiable with respect to the state x_t as well as the parameter θ_t.

The main objective of the dual state and parameter estimation problem is to approximate the following conditional expectations:

$$\mathbb{E}(\phi_1(x_t)|y_{1:t}, \theta_{t-1}) = \int \phi_1(x_t) p(x_t|y_{1:t}, \theta_{t-1}) \mathrm{d}x_t, \tag{3.3a}$$

$$\mathbb{E}(\phi_2(\theta_t)|y_{1:t}, x_t) = \int \phi_2(\theta_t) p(\theta_t|y_{1:t}, x_t) \mathrm{d}\theta_t, \tag{3.3b}$$

where $y_{1:t} = (y_1, y_2, ..., y_t)$ denotes the available observations up to time t, $\phi_1 : \mathbb{R}^{n_x} \to \mathbb{R}$ and $\phi_2 : \mathbb{R}^{n_\theta} \to \mathbb{R}$ are functions of states and parameters, respectively, that are to be estimated. The conditional probability functions $p(x_t|y_{1:t}, \theta_{t-1})\mathrm{d}x_t$ and $p(\theta_t|y_{1:t}, x_t)\mathrm{d}\theta_t$ are to be approximated by the designed particle filters (PFs) through determining the filtering distributions according to

$$\begin{aligned} \hat{p}_N(x_t|y_{1:t}, \theta_{t-1})\mathrm{d}x_t &= \sum_{i=1}^{N} w_{x_t}^{(i)} \delta_{x_t^{(i)}}(\mathrm{d}x_t), \\ \hat{p}_N(\theta_t|y_{1:t}, x_t)\mathrm{d}\theta_t &= \sum_{j=1}^{N} w_{\theta_t}^{(j)} \delta_{\theta_t^{(j)}}(\mathrm{d}\theta_t), \end{aligned} \tag{3.4}$$

where the subscript N in $\hat{p}_N(.)$ implies that the state/parameter conditional probability distributions are obtained from N particles. Each state particle $x_t^{(i)}$ has a weight $w_{x_t}^{(i)}$ and each parameter particle $\theta_t^{(j)}$ has a weight $w_{\theta_t}^{(j)}$, where $\delta(.)$ denotes the Dirac-delta function mass that is positioned at x_t or θ_t.

Based on the approximations used in equation (3.4), our goal is to address the convergence properties of the subsequently designed estimators to their true optimal estimates and also to develop and demonstrate under what conditions this convergence remains valid.

3.2 Proposed Dual State/Parameter Estimation and Fault Diagnosis Framework

In this section, the main theoretical framework for our proposed dual state/parameter filtering as well as the fault diagnosis methodology of the nonlinear system (3.1) and (3.2) are introduced and developed.

3.2.1 Dynamic Model in Presence of Time-Varying Parameters

Our first task is to represent the model (3.1) and (3.2) into another framework for our subsequent theoretical developments. Let $(\Omega, \mathcal{F}, P)$ denote the probability space on which the three real vector-valued stochastic processes $X = \{X_t, t = 1, 2, ...\}, \Theta = \{\Theta_t, t = 1, 2, ...\}$, and $Y = \{Y_t, t = 1, 2, ...\}$ are defined. The n_x-dimensional process X describes the evolution of the hidden states, the n_θ-dimensional process Θ describes the evolution of the hidden system parameters that are conditionally independent of the states, and the n_y-dimensional process Y denotes the observation process of the system.

The processes X and Θ are Markov processes with the associated initial state and parameter

X_0 and Θ_0, respectively. They are drawn from the initial distributions $\pi_{x_0}(\mathrm{d}x_0)$ and $\pi_{\theta_0}(\mathrm{d}\theta_0)$, respectively. The dynamic evolution of states and parameters are modeled by the Markov transition kernels $K_x(\mathrm{d}x_t|x_{t-1}, \theta_{t-1})$ and $K_\theta(\mathrm{d}\theta_t|\theta_{t-1}, x_t)$, that also admit densities with respect to the Lebesgue measure [1], such that

$$P(X_t \in A_1|X_{t-1} = x_{t-1}, \Theta_{t-1} = \theta_{t-1}) = \int_{A_1} K_x(x_t|x_{t-1}, \theta_{t-1})\mathrm{d}x_t, \tag{3.5}$$

$$P(\Theta_t \in A_2|\Theta_{t-1} = \theta_{t-1}, X_t = x_t) = \int_{A_2} K_\theta(\theta_t|\theta_{t-1}, x_t)\mathrm{d}\theta_t, \tag{3.6}$$

for all $A_1 \in \mathcal{B}(\mathbb{R}^{n_x})$ and $A_2 \in \mathcal{B}(\mathbb{R}^{n_\theta})$, where $\mathcal{B}(\mathbb{R}^{n_x})$ and $\mathcal{B}(\mathbb{R}^{n_\theta})$ denote the Borel σ-algebra on $\mathbb{R}^{n_x}$ and $\mathbb{R}^{n_\theta}$, respectively. The transition kernel $K_x(x_t|x_{t-1}, \theta_{t-1})$ is a probability distribution function (pdf) that follows the pdf of the stochastic process in process (3.1). The probability density function for approximating the parameter kernel transition $K_\theta(\theta_t|\theta_{t-1}, x_t)$ is to be provided in the subsequent subsections.

Given the states and parameters, the observations Y_t are conditionally independent and have the marginal distribution with a density with respect to the Lebesgue measure as given by,

$$P(Y_t \in B|X_t = x_t, \Theta_t = \theta_t) \quad = \int_B \rho(y_t|x_t, \theta_t)\mathrm{d}y_t, \tag{3.7}$$

where $\rho(y_t|x_t, \theta_t)$ is a probability density function that follows the probability density function of the stochastic process in equation (3.2).

In the dual state/parameter estimation framework, at first the state x_t is estimated (which

[1]The transition kernel $K(\mathrm{d}x_t|x_{t-1})$ admits density with respect to the Lebesgue measure if one can write $P(X_t \in \mathrm{d}x_t|X_{t-1} = x_{t-1}) = K(\mathrm{d}x_t|x_{t-1}) = K(x_t|x_{t-1})\mathrm{d}x_t$.

is denoted by $\hat{x}_{t|t}$). The estimated value at time t is then used to estimate the parameter θ_t at time t (which is denoted by $\hat{\theta}_{t|t}$). In the Bayesian framework for parameter estimation, the prior evolution of parameters are not specified, therefore it is necessary to consider a given evolution for the parameters in order to design an estimation filter. In our proposed dual structure for the state estimation filter, first the parameters are assumed to be constant at time $t-1$ at their estimated value $\hat{\theta}_{t-1|t-1}$, and then for the parameter estimation filter they are evolved to the next time instant by applying an update law that is inspired from the recursive prediction error method. The details regarding our proposed methodology are presented in the subsequent subsections in which the filtering of states and parameters are fully described and developed.

3.2.2 The Dual State/Parameter Estimation Framework

In our proposed dual state/parameter estimation framework, two filters are running concurrently. At every time step, the first PF-based state filter estimates the states by using the current available estimate of the parameters, $\hat{\theta}_{t-1|t-1}$, whereas the second PF-based parameter filter estimates the unknown parameters by using the current estimated states, $\hat{x}_{t|t}$. The developed schematic is shown in Figure 3.1.

In our dual estimation framework, the well-known maximum *a posteriori* (MAP) solution corresponding to the marginal estimation methods based on the decoupled approach is used for solving the dual estimation problem [163]. In this method, the joint state/parameter marginal density $p(x_t, \theta_t|y_{1:t})$ is expressed as

$$p(x_t, \theta_t|y_{1:t}) = p(x_t|\theta_t, y_{1:t})p(\theta_t|y_{1:t}), \tag{3.8}$$

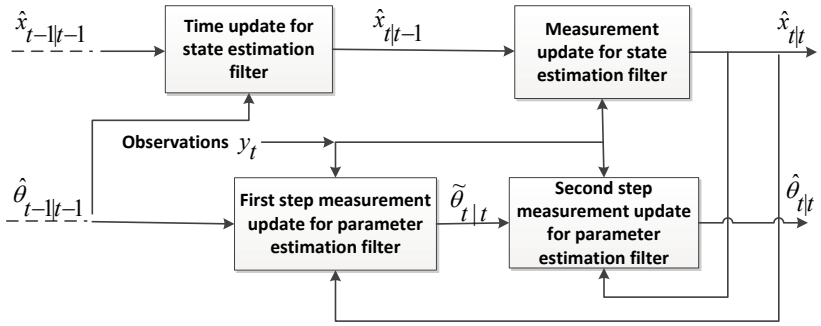


Figure 3.1: The schematic of the dual particle filter.

where $p(x_t|\theta_t, y_{1:t})$ and $p(\theta_t|y_{1:t})$ denote the state and parameter marginal densities, respectively. Assuming that the variations of parameters are slow when compared to the system state time variations, one can use the approximation $\theta_t \approx \theta_{t-1}$, so that the joint marginal density is approximated as

$$p(x_t, \theta_t|y_{1:t}) \approx p(x_t|\theta_{t-1}, y_{1:t})p(\theta_{t-1}|y_{1:t}). \tag{3.9}$$

Our ultimate goal is to maximize the two marginal distribution terms in expression (3.9) separately according to the decoupled approach in [163] as follows

$$\hat{x}_{t|t} = \text{argmax}_{x_t} p(x_t|\theta_{t-1}, y_{1:t}), \quad \hat{\theta}_{t|t} = \text{argmax}_{\theta_{t-1}} p(\theta_{t-1}|y_{1:t}). \tag{3.10}$$

In the above decoupled methodology, one attribute is optimized at a time by keeping the other attribute as fixed and then alternating them. Associated with optimization of both marginal distributions, different cost functions can be chosen [163]. For developing a dual extended Kalman filter, corresponding to specific cost functions of the parameter marginal density, various estimation methods have been proposed in the literature [163, 164]. For example, the maximum likelihood (ML) and prediction error approaches are selected for marginal estimations. The main motivation for choosing these two approaches is due to the fact that one considers to

maximize only the marginal density $p(\theta_{t-1}|y_{1:t})$ as opposed to the joint density $p(x_t, \theta_{t-1}|y_{1:t})$. However, in order to maximize the parameter marginal density, it is also necessary to generate state estimates that are produced by maximizing the state marginal density $p(x_t|\theta_{t-1}, y_{1:t})$.

It should be noted that in marginal estimation methods no explicit cost function is considered for maximization of the state marginal distribution, since the state estimation is only an implicit step in marginal approaches and the joint state/parameter cost is used that may have variety of forms in different filtering algorithms [163]. In our proposed dual particle filtering framework, $p(x_t|\theta_{t-1}, y_{1:t})$ is approximated by the state filtering distribution $\hat{p}_N(x_t|\theta_{t-1}, y_{1:t})$ from equation (3.4). Next, the prediction error cost function is chosen for maximization of the parameter marginal density, where this cost function is implemented in a recursive manner in order to attain a less computational cost [165].

In the subsequent subsections, specific details regarding the concurrent state and parameter estimation filters design and development are provided.

3.2.3 The State Estimation Problem

For designing the state and parameter estimation filters, our main objectives are to approximate the integrals in equations (3.5) and (3.6) by invoking the particle filter (PF) scheme as well as to approximate the estimate of the conditional state and parameter distributions. Consider $\pi_{x_{t|t-1}}(\mathrm{d}x_t) = \int_{\mathbb{R}^{n_x}} \pi_{x_{t-1|t-1}}(\mathrm{d}x_{t-1}) K_x(\mathrm{d}x_t|x_{t-1}, \theta_{t-1})$ to denote the *a priori* state estimation distribution before the observation at time t becomes available, and $\pi_{\theta_{t-1|t-1}}(\mathrm{d}\theta_{t-1})$ to denote

the marginal distribution of the parameter at time $t-1$. The *a posteriori* state distribution after the observation at the instant t becomes available is obtained according to the following rule,

$$\pi_{x_{t|t}}(\mathrm{d}x_t) \propto \rho(y_t|x_t, \theta_{t-1})\pi_{x_{t|t-1}}(\mathrm{d}x_t)\pi_{\theta_{t-1|t-1}}(\mathrm{d}\theta_{t-1}). \tag{3.11}$$

In the above it is assumed that $\hat{\theta}_{t-1|t-1}$ is known for this filter. Therefore, the last distribution in the right hand side of equation (3.11) is set to one.

The particle filter (PF) procedure for implementation of the state estimation and for determining $\pi_{x_{t|t}}(\mathrm{d}x_t)$ consists of two main steps, namely **(a)** the prediction step (time update step), and **(b)** the measurement update step. Consider one states in the N particles at time t. The prediction step utilizes the knowledge of the previous distribution of the states as well as the previous parameter estimate, these are denoted by $\{\hat{x}^{(i)}_{t-1|t-1},\ i = 1, ..., N\}$ (corresponding to N estimated state particles that follow the distribution $\pi_{x_{t-1|t-1}}(\mathrm{d}x_{t-1})$) and $\hat{\theta}_{t-1|t-1}$, respectively, as well as the process model given by equation (3.1). In other words, the prediction step is explicitly governed by the following equations for $i = 1, ..., N$, namely

$$\hat{x}^{(i)}_{t|t-1} = f_t(\hat{x}^{(i)}_{t-1|t-1}, \hat{\theta}_{t-1|t-1}, \omega^{(i)}_t), \tag{3.12a}$$

$$\hat{y}^{(i)}_{t|t-1} = h_t(\hat{x}^{(i)}_{t|t-1}, \hat{\theta}_{t-1|t-1}), \tag{3.12b}$$

$$\Sigma_{\hat{x}_{t|t-1}} = (\hat{x}^{(i)}_{t|t-1} - \frac{1}{N}\sum_{i=1}^{N}\hat{x}^{(i)}_{t|t-1})(\hat{x}^{(i)}_{t|t-1} - \frac{1}{N}\sum_{i=1}^{N}\hat{x}^{(i)}_{t|t-1})^{\mathrm{T}}, \tag{3.12c}$$

where $\omega^{(i)}_t$ denotes the process noise related to each particle $\hat{x}^{(i)}_{t|t-1}$ and is drawn from the noise distribution with the probability distribution function $p_{\omega_t}(.)$, and $\hat{x}^{(i)}_{t|t-1}$ denotes the independent samples generated from equation (3.12a) for $i = 1, ..., N$ particles. Moreover, $\hat{y}^{(i)}_{t|t-1}$ denotes the independent samples of the predicted outputs that are evaluated at $\hat{x}^{(i)}_{t|t-1}$ samples, and $\Sigma_{\hat{x}_{t|t-1}}$ denotes the *a priori* state estimation covariance matrix.

For the first step, the one-step ahead prediction distribution known as the *a priori* state estimation distribution is now given by,

$$\tilde{\pi}^{N}_{x_{t|t-1}}(\mathrm{d}x_t) \triangleq \frac{1}{N}\sum_{i=1}^{N}\delta_{\hat{x}^{(i)}_{t|t-1}}(\mathrm{d}x_t), \tag{3.13}$$

For the second step, the information on the present observation y_t is used. This results in approximating $\pi_{x_{t|t}}(\mathrm{d}x_t)$, where $\hat{\theta}_{t-1|t-1}$ is considered to be given from a parameter estimation filter and obtained from the distribution $\pi^{N}_{\hat{\theta}_{t-1|t-1}}(\mathrm{d}\theta_{t-1})$. Consequently, the particle weights $w^{(i)}_{x_t}$ are updated by the likelihood function (the importance function) according to $w^{(i)}_{x_t} \sim p_{\nu_t}(y_t - \hat{y}^{(i)}_{t|t-1}) = \rho(y_t|\hat{x}^{(i)}_{t|t-1}, \hat{\theta}_{t-1|t-1})$, where $p_{\nu_t}(.)$ denotes the probability distribution function of the additive noise of the output and is evaluated at $y_t - \hat{y}^{(i)}_{t|t-1}$.

In this thesis, since our ultimate goal is in developing a fault diagnosis algorithm that is practically stable, the structure of regularized particle filters (RPF) is chosen that has a better performance in cases that the sample impoverishment is severe, that is quite common and almost the case in all practical applications [166]. This characteristics of the RPFs are related to the fact that they are capable of transforming the discrete-time approximation of the *a posteriori* state estimation distribution $\pi^{N}_{x_{t|t}}(\mathrm{d}x_t)$ into a continuous-time one. Consequently, the resampling step is modified in such a manner that the new resampled particles are obtained from an absolutely continuous-time distribution with N different locations $\hat{x}^{(i)}_{t|t}$ from that of $\hat{x}^{(i)}_{t|t-1}$ [167]. Therefore, where the probability for taking the k-th particle is $P(\hat{x}^{(i)}_{t|t} = \hat{x}^{(k)}_{t|t-1}) = \tilde{w}^{(k)}_{x_t} \triangleq \frac{\rho(y_t|\hat{x}^{(k)}_{t|t-1}, \hat{\theta}_{t-1|t-1})}{\sum_{k=1}^{N}\rho(y_t|\hat{x}^{(k)}_{t|t-1}, \hat{\theta}_{t-1|t-1})}$, and $\tilde{w}^{(k)}_{x_t}$ for $k = 1, ..., N$ denotes the normalized particle weights. In other words, the particle selection in the resampling step is performed for particles that have higher probabilities of $\rho_{\nu_t}(y_t - \hat{y}^{(k)}_{t|t-1})$.

For the resampling step, two main choices can be considered that are known as **(i)** Bayesian bootstrap, and **(ii)** Sampling importance resampling (SIR) [55]. Although both approaches are applicable for this filter, the bootstrap method is chosen in this chapter. Therefore, the *a posteriori* state estimation distribution is approximated by $\tilde{\pi}^N_{x_{t|t}}(\mathrm{d}x_t)$ before one performs the resampling by using the RPF structure [167], and by $\pi^N_{x_{t|t}}(\mathrm{d}x_t)$ after one performs the resampling that is provided below,

$$\begin{aligned}
&\tilde{\pi}^N_{x_{t|t}}(\mathrm{d}x_t) \approx \sum_{l=1}^{N_{\mathrm{reg}}} \sum_{i=1}^{N} \tilde{w}^{(i)}_{x_t} \frac{|\mathbb{A}_t^{-1}|}{b^{n_x}} \mathcal{K}(\frac{1}{b}\mathbb{A}_t^{-1}(x_t^{\mathrm{reg}_l} - \hat{x}^{(i)}_{t|t-1})),\\
&\tilde{w}^{(i)}_{x_t} \triangleq \frac{\rho(y_t|\hat{x}^{(i)}_{t|t-1}, \hat{\theta}_{t-1|t-1})}{\sum_{i=1}^{N} \rho(y_t|\hat{x}^{(i)}_{t|t-1}, \hat{\theta}_{t-1|t-1})},\\
&\pi^N_{x_{t|t}}(\mathrm{d}x_t) = \frac{1}{N}\sum_{i=1}^{N} \delta_{\hat{x}^{(i)}_{t|t}}(\mathrm{d}x_t) \rightarrow \hat{x}_{t|t} = \frac{1}{N}\sum_{i=1}^{N} \hat{x}^{(i)}_{t|t},
\end{aligned} \tag{3.14}$$

where $x_t^{\mathrm{reg}_l},\ l = 1, .., N_{\mathrm{reg}}$ denotes the regularized state vector that is evaluated at N_{reg} points that are obtained from the absolutely continuous-time distribution of the particles as given by

$$\begin{aligned}
&X_{t|t-1} = [\hat{x}^{(1)}_{t|t-1}, ..., \hat{x}^{(N)}_{t|t-1}],\\
&x_t^{\mathrm{reg}_1} = \min(X_{t|t-1}) - \mathrm{std}(X_{t|t-1}),\ x_t^{\mathrm{reg}_{N_{\mathrm{reg}}}} = \max(X_{t|t-1}) + \mathrm{std}(X_{t|t-1}),\\
&\mathrm{d}x^{\mathrm{reg}} = (x_t^{\mathrm{reg}_{N_{\mathrm{reg}}}} - x_t^{\mathrm{reg}_1})/(N_{\mathrm{reg}} - 1),\\
&x_t^{\mathrm{reg}_l} = x_t^{\mathrm{reg}_{l-1}} + \mathrm{d}x^{\mathrm{reg}},\ l = 2, ..., N_{\mathrm{reg}},
\end{aligned} \tag{3.15}$$

where std denotes the first standard deviation of the particles from their mean. Hence, $\{\hat{x}^{(i)}_{t|t}\}_{i=1}^N$ is obtained from the continuous-time distribution through the regularization kernel $\mathcal{K}$ that is considered to be a symmetric density function on $\mathbb{R}^{n_x}$ [167]. The matrix $\mathbb{A}_t$ in equation (3.14) is chosen to yield a unit covariance value in the new $\hat{x}^{(i)}_{t|t}$ population and $\mathbb{A}_t\mathbb{A}_t^{\mathrm{T}} = \Sigma_{\hat{x}_{t|t-1}}$. The constant b denotes the bandwidth of the kernel, and $\hat{x}_{t|t}$ denotes the *a posteriori* state estimation at time t.

We are now in a position to introduce our overall particle filter (PF) scheme for implementing the state estimation filter. Our goal for proposing this algorithm is to ensure that an approximation to $\mathbb{E}(\phi(x_t)|y_{1:t}, \theta_{t-1})$ by $\phi(x_t) = x_t$ takes $\hat{x}_{t|t} \sim \pi^N_{x_{t|t}}(\mathrm{d}x_t) = \frac{1}{N}\sum_{i=1}^N \delta_{\hat{x}^{(i)}_{t|t}}(\mathrm{d}x_t)$, where $\pi^N_{x_{t|t}}(\mathrm{d}x_t)$ denotes the *a posteriori* distribution of $\{\hat{x}^{(i)}_{t|t}\}_{i=1}^N$ (after the resampling from $\{\hat{x}^{(i)}_{t|t-1}\}_{i=1}^N$), that is given by $\hat{x}_{t|t} = \frac{1}{N}\sum_{i=1}^N \hat{x}^{(i)}_{t|t}$. The estimated output from the state estimation filter is also given by $\hat{y}_t = h_t(\hat{x}_{t|t}, \hat{\theta}_{t-1|t-1})$.

The State Estimation Particle Filter Scheme

1. Initialize the PF scheme with N particles, $\{x_0^{(i)}\}_{i=1}^N \sim \pi_{x_0}(\mathrm{d}x_0)$ and the parameters θ_0 (the mean of the parameter initial distribution $\pi_{\theta_0}(\mathrm{d}\theta_0)$).

2. Draw $\omega_t^{(i)} \sim p_{\omega_t}(.)$, where $p_{\omega_t}(.)$ denotes a given distribution for the process noise in the filter, and then predict the state particles $\hat{x}^{(i)}_{t|t-1}$ according to equation (3.12a).

3. Compute $\hat{y}^{(i)}_{t|t-1}$ from equation (3.12b) to obtain the importance weights $\{w^{(i)}_{x_t}\}_{i=1}^N$ as $w^{(i)}_{x_t} = \rho(y_t|\hat{x}^{(i)}_{t|t-1}, \hat{\theta}_{t-1|t-1}),\ i = 1, ..., N$, and normalize them to $\tilde{w}^{(i)}_{x_t} = \frac{w^{(i)}_{x_t}}{\sum_{i=1}^N w^{(i)}_{x_t}}$.

4. Resampling: Draw N new particles with the replacement for each $i = 1, ..., N$, according to $P(\hat{x}^{(i)}_{t|t} = \hat{x}^{(k)}_{t|t-1}) = \tilde{w}^{(k)}_{x_t},\ k = 1, ..., N$, from the regularized kernel $\mathcal{K}$ where $\hat{x}^{(i)}_{t|t} \sim \tilde{\pi}^N_{x_{t|t}}(\mathrm{d}x_t)$ as given by equation (3.14).

5. Calculate $\hat{x}_{t|t}$ from the conditional distribution that is given by equation (3.14), $\pi^N_{x_{t|t}}(\mathrm{d}x_t) = \frac{1}{N}\sum_{i=1}^N \delta_{\hat{x}^{(i)}_{t|t}}(\mathrm{d}x_t)$ with equally weighted $\hat{x}^{(i)}_{t|t}$ as $\hat{x}_{t|t} = \frac{1}{N}\sum_{i=1}^N \hat{x}^{(i)}_{t|t}$.

6. Update the parameters from the parameter estimation filter (to be specified in the next

subsection).

7. Set $t := t + 1$ and go to Step 2.

Following the implementation of the above state estimation filter, the parameter estimation filter that is utilized for adjusting the parameters is now described in detail in the next subsection.

3.2.4 The Parameter Estimation Problem

One of the main contributions of this thesis is to develop a novel PF-based parameter estimation filter within our proposed dual state/parameter estimation framework by utilizing the prediction error (PE) concept. For this methodology it is assumed that the *a priori* distribution of the time-varying parameters is not known. Moreover, the estimated states that are generated by the state estimation filter provided in the previous subsection will be used. Therefore, it is imperative that one considers a dynamical model associated with the parameters evolution in order to estimate the density function $\pi_{\theta_{t|t}}(\mathrm{d}\theta_t)$.

The most common dynamical model that is considered for the parameter propagation (in case of the system with constant parameters) is the conventional artificial evolution law. In this representation small random disturbances are added to the state particles (parameters) between each consecutive time step [2]. However, in our work, the conventional update law for the parameters is modified to include the output prediction error as an extra term to the parameter evolution law to allow one to deal with time variations in the parameters that can affect the system output.

In order to derive the parameter update law, an algorithm based on the prediction error (PE) method is proposed by minimizing the expectation of a quadratic performance index $\bar{J}(\theta_{t-1})$ with respect to θ_{t-1}. This is due to the fact that our parameter estimation algorithm for obtaining the distribution of the *a posteriori* parameter estimate is based on the kernel smoothing that uses the shrinkage of the particle locations. This method attempts to force the particles towards their mean from the previous time step, i.e. the estimated value of θ_{t-1}, and is denoted by $\hat{\theta}_{t-1|t-1}$ (before adding noise to the particles). This is also used in the state estimation filter for approximating $\hat{x}_{t|t}$. Therefore, our goal is to investigate the convergence properties of $\hat{\theta}_{t-1|t-1}$ whose boundedness ensures the boundedness of $\hat{\theta}_{t|t}$. Towards this end, the performance index is now selected as $\mathbb{E}(\bar{J}(\theta_{t-1})|y_{1:t-1}, x_t) = \int \bar{J}(\theta_{t-1}) p(\theta_{t-1}|y_{1:t-1}, x_t) \mathrm{d}\theta_{t-1}$, where the integral is approximated in the PF by $\mathbb{E}(\bar{J}(\theta_{t-1})|y_{1:t-1}, x_t) \approx \frac{1}{N} \sum_{j=1}^{N} \bar{J}(\hat{\theta}^{(j)}_{t-1|t-1})$.

The term $\bar{J}(\hat{\theta}^{(j)}_{t-1|t-1})$ now represents a quadratic function of the output prediction error related to each particle j, $j = 1, ..., N$. The prediction error is now defined according to $\epsilon(t, \hat{\theta}^{(j)}_{t-1|t-1}) \triangleq \epsilon^{(j)}_t = y_t - h_t(\hat{x}_{t|t}, \hat{\theta}^{(j)}_{t-1|t-1})$, where $\hat{\theta}^{(j)}_{t-1|t-1}$ denotes the particle related to the estimated value of the parameter whose true value is denoted by $\theta^{\star}_{t-1}$ (this is clearly assumed to be unknown). Therefore, we define $\bar{J}(\hat{\theta}^{(j)}_{t-1|t-1}) = \frac{1}{\kappa} \sum_{\tau=t-\kappa}^{\tau=t} \mathbb{E}(Q(\epsilon(\tau, \hat{\theta}^{(j)}_{t-1|t-1})))$, in which the expectation is taken over the observation sequence of κ samples. Let us now select the quadratic criterion $Q(\epsilon(t, \hat{\theta}^{(j)}_{t-1|t-1}))$ as

$$Q(\epsilon(t, \hat{\theta}^{(j)}_{t-1|t-1})) = \frac{1}{2} \epsilon(t, \hat{\theta}^{(j)}_{t-1|t-1}) \epsilon^{\mathrm{T}}(t, \hat{\theta}^{(j)}_{t-1|t-1}). \tag{3.16}$$

The following modified artificial evolution law is now proposed for the parameter update in the particle filters for generating $j = 1, ..., N$ parameter particles that correspondingly de-

termine the distribution from which the *a priori* parameter estimate $\hat{\theta}^{(j)}_{t|t-1}$ is considered to be the same as $\hat{\theta}^{(j)}_{t-1|t-1}$, and the *a posteriori* parameter estimate is obtained in two steps that are denoted by $\tilde{\theta}^{(j)}_{t|t}$ and $\hat{\theta}^{(j)}_{t|t}$, respectively. In the first step one gets (the second step is described on the next page)

$$m^{(j)}_t = \hat{\theta}^{(j)}_{t-1|t-1} + \gamma_t R^{(j)}_t \psi^{(j)}_t \epsilon(t, \hat{\theta}^{(j)}_{t-1|t-1}), \tag{3.17a}$$

$$\tilde{\theta}^{(j)}_{t|t} = A m^{(j)}_t + (I - A)\bar{m}_{t-1} + \zeta^{(j)}_t,\ \bar{m}_{t-1} = \frac{1}{N}\sum_{j=1}^{N} \hat{\theta}^{(j)}_{t-1|t-1}, \tag{3.17b}$$

where $\psi_t = \frac{\partial \hat{y}_t}{\partial \hat{\theta}_{t-1|t-1}} = \frac{\partial h_t(\hat{x}_{t|t}, \hat{\theta}_{t-1|t-1})}{\partial \hat{\theta}_{t-1|t-1}}$, which when evaluated at $\hat{\theta}^{(j)}_{t-1|t-1}$ is denoted by $\psi^{(j)}_t$, γ_t denotes the step size design parameter, $\zeta^{(j)}_t \sim \mathcal{N}(0, (I - A^2)V_{\hat{\theta}_{t-1|t-1}})$ denotes the zero-mean normal increment particles to the parameter update law at each time step with the covariance matrix $(I - A^2)V_{\hat{\theta}_{t-1|t-1}}$ through the use of the kernel smoothing concept, A denotes the shrinkage matrix, and $V_{\hat{\theta}_{t-1|t-1}}$ denotes the covariance of the parameter estimates in the previous time step $t-1$. The kernel shrinkage algorithm attempts to force the distribution of the parameter particles towards the mean of their distribution in the previous time instant that was denoted by $\bar{m}_{t-1}$, by applying the shrinkage coefficient matrix A to the obtained $m^{(j)}_t$. The processes $\hat{\theta}^{(j)}_{t-1|t-1}$ and $\zeta^{(j)}_t$ are conditionally independent given observations up to time t. Moreover, $R^{(j)}_t = \sqrt{\mathrm{trace}(\mathcal{E}^{(j)}_t \mathcal{E}^{(j)^{\mathrm{T}}}_t)}$ where $\mathcal{E}^{(j)}_t = \epsilon_t(\hat{\theta}^{(j)}_{t-1|t-1}) - \frac{1}{n_y}\sum_{l=1}^{l=n_y} \epsilon^{(l)}_t(\hat{\theta}^{(j)}_{t-1|t-1})$ and $\epsilon^{(l)}_t(\hat{\theta}^{(j)}_{t-1|t-1})$ denotes the l-th element of the vector $\epsilon_t(\hat{\theta}^{(j)}_{t-1|t-1})$. The term $R^{(j)}_t$ denotes a time-varying coefficient that determines the updating direction and is a positive scalar to ensure that the criterion (3.16) can be minimized by changing $m^{(j)}_t$ in the steepest descent direction. Therefore, the first step estimate of the *a posteriori* parameter estimation particle is denoted by $\tilde{\theta}^{(j)}_{t|t}$. The convergence of the update law (3.17a)-(3.17b) will be shown in the Subsection 3.2.6.

The parameter update law according to (3.17a)-(3.17b) contains a term in addition to the

independent normal increment $\zeta_t^{(j)}$. The estimated parameter from this update law is invoked in the PF-based parameter estimation filter to represent the distribution from which the parameter particle population for the next time step is chosen. Therefore, the above proposed prediction error based modified artificial evolution law enables the PF-based estimation algorithm to handle and cope with the time-varying parameter scenarios. The time-varying term $\gamma_t R_t^{(j)}$ acts as an adaptive step size in equations (3.17a)-(3.17b), and therefore our algorithm can also be considered as an adaptive step size scheme.

In order to ensure that the obtained $\tilde{\theta}_{t|t}^{(j)}$ from the modified artificial evolution law given by equations (3.17a)-(3.17b) remains in $D_{\mathcal{N}}$ (refer to Assumption 3.1), the following projection algorithm is utilized that forces $\tilde{\theta}_{t|t}^{(j)}$ to remain inside $D_{\mathcal{N}}$ according to the following procedure [165],

1. Choose a factor $0 \leq \mu \leq 1$,
2. Compute $\breve{\theta}_{t|t}^{(j)} := \gamma_t R_t^{(j)} \psi_t^{(j)} \epsilon(t, \hat{\theta}_{t-1|t-1}^{(j)})$,
3. Construct $m_t^{(j)} := \hat{\theta}_{t-1|t-1}^{(j)} + \breve{\theta}_{t|t}^{(j)}$,
4. If $m_t^{(j)} \in D_{\mathcal{N}}$ go to Step 6, else go to Step 5,
5. Set $\breve{\theta}_{t|t}^{(j)} = \mu \breve{\theta}_{t|t}^{(j)}$, and go to Step 3,
6. Stop.

It should be noted that the main reason for considering the above mapping is related the fact that the actual dynamics of the parameters are not known, therefore such mapping ensures that

the assumed dynamics for the parameters based on modified artificial evolution model does not cause instability of the entire system.

Consequently, the *a priori* distribution of the parameter θ_t is assumed to have the same distribution as in the previous time step. On the other hand, as the present observation y_t becomes available in the measurement update step, the *a posteriori* distribution of the parameter is obtained through two steps that denoted by $\tilde{\pi}^N_{\tilde{\theta}_{t|t}}(\mathrm{d}\theta_t)$ and $\tilde{\pi}^N_{\theta_{t|t}}(\mathrm{d}\theta_t)$, respectively. In what follows, more details related to these distributions are presented.

Consider equations (3.17a)-(3.17b). The first step *a posteriori* distribution of the parameters calculated from the distribution of the parameter particles $\tilde{\theta}^{(j)}_{t|t}$ is given by,

$$\tilde{\pi}^N_{\tilde{\theta}_{t|t}}(\mathrm{d}\theta_t) \triangleq \frac{1}{N}\sum_{j=1}^{N}\delta_{\tilde{\theta}^{(j)}_{t|t}}(\mathrm{d}\theta_t), \tag{3.18}$$

and the measurement equation is expressed as,

$$\bar{y}^{(j)}_{t|t} = h_t(\hat{x}_{t|t}, \tilde{\theta}^{(j)}_{t|t}), \tag{3.19}$$

where $\bar{y}^{(j)}_{t|t}$ denotes the evaluated output that is obtained by the parameter estimation filter that is different from the one that is obtained by the state estimation filter, as provided in the Subsection 3.2.3.

Now, in the second step for estimating the *a posteriori* parameter estimate distribution, consider the present observation y_t, so that the particle weights $w^{(j)}_{\theta_t}$ are updated by the likelihood function according to $w^{(j)}_{\theta_t} \sim p_{\nu_t}(y_t - \bar{y}^{(j)}_{t|t}) = \rho(y_t|\hat{x}_{t|t}, \tilde{\theta}^{(j)}_{t|t})$. This can now be expressed by using the normalized weights $\tilde{w}^{(j)}_{\theta_t}$ as $\tilde{\pi}^N_{\theta_{t|t}}(\mathrm{d}\theta_t) = \sum_{j=1}^{N}\tilde{w}^{(j)}_{\theta_t}\delta_{\tilde{\theta}^{(j)}_{t|t}}(\mathrm{d}\theta_t)$, where $\tilde{w}^{(j)}_{\theta_t} \triangleq \frac{\rho(y_t|\hat{x}_{t|t}, \tilde{\theta}^{(j)}_{t|t})}{\sum_{j=1}^{N}\rho(y_t|\hat{x}_{t|t}, \tilde{\theta}^{(j)}_{t|t})}$. Following the resampling/selection step, an equally weighted particle distribution $\pi^N_{\theta_{t|t}}(\mathrm{d}\theta_t)$ is obtained as $\pi^N_{\theta_{t|t}}(\mathrm{d}\theta_t) = \frac{1}{N}\sum_{j=1}^{N}\delta_{\hat{\theta}^{(j)}_{t|t}}(\mathrm{d}\theta_t)$ for approximating $\pi_{\theta_{t|t}}(\mathrm{d}\theta_t)$, and the resampled (se-

lected) particles that are denoted by $\hat{\theta}_{t|t}^{(j)}$ follow the distribution $\tilde{\pi}_{\theta_{t|t}}^{N}(\mathrm{d}\theta_t)$. Therefore, the *a posteriori* parameter estimation distribution is approximated by a weighted sum of the Dirac-delta masses as $\tilde{\pi}_{\theta_{t|t}}^{N}(\mathrm{d}\theta_t)$ before one performs the resampling and with an equally weighted particle distribution approximation as $\pi_{\theta_{t|t}}^{N}(\mathrm{d}\theta_t)$ according to

$$\begin{aligned}
&\tilde{\pi}_{\theta_{t|t}}^{N}(\mathrm{d}\theta_t) \approx \sum_{j=1}^{N} \tilde{w}_{\theta_t}^{(j)} \delta_{\tilde{\theta}_{t|t}^{(j)}}(\mathrm{d}\theta_t), \\
&\tilde{w}_{\theta_t}^{(j)} \triangleq \frac{\rho(y_t|\hat{x}_{t|t}, \tilde{\theta}_{t|t}^{(j)})}{\sum_{j=1}^{N} \rho(y_t|\hat{x}_{t|t}, \tilde{\theta}_{t|t}^{(j)})}, \\
&\pi_{\theta_{t|t}}^{N}(\mathrm{d}\theta_t) = \frac{1}{N} \sum_{j=1}^{N} \delta_{\hat{\theta}_{t|t}^{(j)}}(\mathrm{d}\theta_t) \rightarrow \hat{\theta}_{t|t} = \frac{1}{N} \sum_{j=1}^{N} \hat{\theta}_{t|t}^{(j)},
\end{aligned} \tag{3.20}$$

where $\tilde{w}_{\theta_t}^{(j)}$ denotes the normalized parameter particle weight, $\{\hat{\theta}_{t|t}^{(j)}\}_{j=1}^{N}$ is obtained from the resampling/selection step of the scheme by duplicating the particles $\tilde{\theta}_{t|t}^{(j)}$ having large weights and discarding the ones with small values to emphasize the zones with higher *a posteriori* probabilities according to $P(\hat{\theta}_{t|t}^{(j)} = \tilde{\theta}_{t|t}^{(k)}) = \tilde{w}_{\theta_t}^{(k)},\ k = 1, ..., N$. In our proposed filter the residual resampling method is used to ensure that the variance reduction among the resampled particles is guaranteed [168].

Therefore, an approximation to $\mathbb{E}(\phi(\theta_t)|y_{1:t}, x_t)$ by $\phi(\theta_t) = \theta_t$ takes on the form $\hat{\theta}_{t|t} \sim \pi_{\theta_{t|t}}^{N}(\mathrm{d}\theta_t) = \frac{1}{N} \sum_{j=1}^{N} \delta_{\hat{\theta}_{t|t}^{(j)}}(\mathrm{d}\theta_t)$, where $\pi_{\theta_{t|t}}^{N}(\mathrm{d}\theta_t)$ denotes the *a posteriori* distribution of the parameter estimate (after performing the resampling from $\tilde{\theta}_{t|t}^{(j)}$). The resulting estimated output of this filter is obtained by $\hat{y}_t = h_t(\hat{x}_{t|t}, \hat{\theta}_{t|t})$. The explicit details for implementation of the parameter estimation filter are now provided below.

The Parameter Estimation Filter

The particle filter for implementation of the parameter estimation is described as follows:

1. Initialize the N particles for the parameters as $\{\theta_0^j\}_{j=1}^N \sim \pi_{\theta_0}(\mathrm{d}\theta_0)$, and use the initial values of the states as x_0 that represents the mean of the states initial distribution $\pi_{x_0}(\mathrm{d}x_0)$.

2. Draw $\zeta_t^{(j)} \sim \mathcal{N}(0, (I - A^2)V_{\hat{\theta}_{t-1|t-1}})$.

3. Predict $\tilde{\theta}_{t|t}^{(j)},\ j = 1, ..., N$ from equations (3.17a)-(3.17b) with the projection algorithm.

4. Compute the importance weights $\{w_{\theta_t}^{(j)}\}_{j=1}^N, w_{\theta_t}^{(j)} = \rho(y_t|\hat{x}_{t|t}, \tilde{\theta}_{t|t}^{(j)}), j = 1, ..., N$, and normalize them to $\tilde{w}_{\theta_t}^{(j)} = \frac{w_{\theta_t}^{(j)}}{\sum_{j=1}^N w_{\theta_t}^{(j)}}$.

5. Resampling: Draw N new particles with replacement for each $j = 1, ..., N$, $P(\hat{\theta}_{t|t}^{(j)} = \tilde{\theta}_{t|t}^{(k)}) = \tilde{w}_{\theta_t}^{(k)},\ k = 1, ..., N$, where $\tilde{\theta}_{t|t}^{(j)} \sim \tilde{\pi}_{\theta_{t|t}}^N(\mathrm{d}\theta_t) = \sum_{j=1}^N \tilde{w}_{\theta_t}^{(j)} \delta_{\tilde{\theta}_{t|t}^{(j)}}(\mathrm{d}\theta_t)$.

6. Construct $\hat{\theta}_{t|t}$ from the conditional distribution $\pi_{\theta_{t|t}}^N(\mathrm{d}\theta_t) = \frac{1}{N}\sum_{j=1}^N \delta_{\hat{\theta}_{t|t}^{(j)}}(\mathrm{d}\theta_t)$ with equally weighted $\hat{\theta}_{t|t}^{(j)}$ as $\hat{\theta}_{t|t} = \frac{1}{N}\sum_{j=1}^N \hat{\theta}_{t|t}^{(j)}$.

7. Set $t = t + 1$ and go to Step 2 of the state estimation filter as provided in the Subsection 3.2.3.

As stated earlier, the kernel from which the parameter particles i.e. $\tilde{\theta}_{t|t}^{(j)}$ for the next time step is chosen is a Gaussian kernel and its mean is obtained from $m_t^{(j)}$ and its variance is obtained based on the kernel smoothing consideration that is provided in the next Subsection 3.2.5. In the

subsections below, the required conditions for boundedness of the parameter transition kernel $K_\theta(.)$ are also investigated and developed.

3.2.5 Kernel Smoothing of the Parameters

In this subsection, the kernel smoothing approach [77] is utilized to ensure that the variance of the normal distribution which is obtained according to the modified artificial evolution law for the parameter estimates remains bounded.

Consider the modified artificial evolution law (3.17a)-(3.17b) in which $\zeta_t^{(j)}$ is a normal zero-mean uncorrelated random increment to the parameter that is estimated at time $t-1$. If $A = I$, i.e. when there is no kernel shrinkage, as $t \to \infty$, the variance of the added evolution increases and can therefore yield $\tilde{\theta}_{t|t}^{(j)}$ in (3.17b) completely unreliable. This phenomenon is known as the loss of information that can also occur between two consequent sampling times [77]. On the other hand, since θ_t is time-varying, generally there will not exist an optimal value for the variance of the evolution noise $\zeta_t^{(j)}$ that remains suitable for all times.

Consequently, the idea of the kernel shrinkage has been proposed in [77] and later updated in [78]. In the kernel shrinkage approach [2], for the next time step one takes the mean of the estimated parameter distribution in the particle filter according to the following normal distribution

$$K_\theta(\mathrm{d}\theta_t|\theta_{t-1}^{(j)}, x_t) \sim \mathcal{N}(Am_t^{(j)} + (I-A)\bar{m}_{t-1}, (I-A^2)V_{\hat{\theta}_{t-1|t-1}}), \tag{3.21}$$

where $m_t^{(j)}$ for $j = 1, ..., N$, is obtained from (3.17a). By utilizing this kernel shrinkage rule,

the resulting normal distribution retains the mean $\bar{m}_{t-1}$ and has the appropriate variance for avoiding over-dispersion relative to the *a posteriori* sample. The kernel shrinkage forces the parameter samples towards their mean before the noise $\zeta_t^{(j)}$ is added. In our proposed approach the changes due to the parameter variations are considered in the mean of the parameter estimate distribution through the modified artificial evolution rule. Consequently, the mean of the distribution, i.e. $\bar{m}_{t-1}$, itself is time-varying and the kernel shrinkage ensures a smooth transition in the estimated parameters even when they are subjected to changes. To eliminate the information loss effect, by taking the variance from both sides of equation (3.17b) results in $V_{\hat{\theta}_{t|t-1}} = A^2 V_{\hat{\theta}_{t-1|t-1}} + (I - A^2) V_{\hat{\theta}_{t-1|t-1}} = V_{\hat{\theta}_{t-1|t-1}}$. This ensures that the variance of the added random evolution would not cause over-dispersion in the parameter estimation algorithm for all time.

The following proposition specifies an upper bound on the shrinkage factor. This upper bound is calculated in the worst case, that is when the parameter is considered to be constant but the modified evolution law (3.17a)-(3.17b) is used in the parameter estimation filter for estimating it. Utilization of this upper bound in the kernel shrinkage algorithm ensures the boundedness of the variance of the estimated parameters distribution that is obtained according to the PE-based artificial evolution update law and the kernel smoothing augmented with the shrinkage factor.

Proposition 3.1. Upper bound on the kernel shrinkage factor*: Given the parameter update law* (3.17a)-(3.17b)*, the estimated parameters conditional normal distribution based on the kernel smoothing as given by equation* (3.21)*, results in an upper bound for A that is obtained as* $A \leq I(1 - \sqrt{\frac{\sigma_{\min}(P_{\max}^2 \Psi V_y \Psi^{\mathrm{T}} V_{\hat{\theta}}^{-1})}{\sigma_{\max}(P_{\max}^2 \Psi V_y \Psi^{\mathrm{T}} V_{\hat{\theta}}^{-1})}})$*, where Ψ denotes the $\psi_t^{(j)}$ in equation* (3.17a) *but considered*

as a constant parameter between the time steps t and $t-1$. Moreover, $\sigma_{\min}$ and $\sigma_{\max}$ denote the minimum and the maximum eigenvalues of a matrix, respectively, V_y denotes the upper-bound on the variance of the measurement noise R_t, $V_{\hat{\theta}}$ denotes the variance of the parameters when they are constant that can be assumed to be the same as the initial covariance of the parameters, and $P_{\max} = \gamma_0\sqrt{\mathrm{trace}(\mathcal{E}_{\max}{\mathcal{E}_{\max}}^{\mathrm{T}})}$, where γ_0 denotes the initial value of the step size, and $\mathcal{E}_{\max}{\mathcal{E}_{\max}}^{\mathrm{T}}$ is a design parameter that denotes the maximum acceptable variance among the prediction error vector elements.

Proof: Let us consider the modified artificial evolution law by assuming that $A = I$ in equation (3.17b). Let us substitute $m_t^{(j)}$ from equation (3.17a) where the superscript (j) is omitted in order to define the modified artificial evolution law in a more general form that is also applicable to each single particle as

$$\tilde{\theta}_{t|t} = \hat{\theta}_{t-1|t-1} + P_t\psi_t\epsilon(t, \hat{\theta}_{t-1|t-1}) + \zeta_t, \tag{3.22}$$

where $P_t = \gamma_t R_t$. Now, let $V(.|y_{1:t})$ denote the variance of the stochastic process assuming that the observations up to time t are available, and $C(.,.|y_{1:t})$ denotes the covariance of the two stochastic processes by assuming that the observations up to time t are available. By taking into account the relationship between the variance of both sides of equation (3.22) when the covariance matrix is assumed to be non-singular and when the prediction error at time t is uncorrelated with the parameter estimate at time t, given that $\hat{\theta}_{t-1|t-1}$ is independent of $P_t\Psi\epsilon_t$, therefore we get $V(\tilde{\theta}_{t|t}|y_{1:t}) = V(\hat{\theta}_{t-1|t-1}|y_{1:t}) + P_t^2\Psi V_y\Psi^{\mathrm{T}} + W_t + 2C(\hat{\theta}_{t-1|t-1}, \zeta_t|y_{1:t}) + 2C(P_t\Psi\epsilon_t, \zeta_t|y_{1:t})$. Furthermore, since $P_t = \gamma_t R_t = \gamma_t\sqrt{\mathrm{trace}(\mathcal{E}_t\mathcal{E}_t^{\mathrm{T}})}$ is a scalar, one can write $V(P_t\Psi\epsilon_t|y_{1:t}) = (\mathbb{E}[P_t|y_{1:t}])^2V(\Psi\epsilon_t|y_{1:t}) + (\mathbb{E}[\Psi\epsilon_t|y_{1:t}])^2V(P_t|y_{1:t}) + V(P_t|y_{1:t})V(\Psi\epsilon_t|y_{1:t}) = P_t^2\Psi V_t\Psi^{\mathrm{T}}$.

In order to ensure that there is no information loss (particularly in the case that θ_t is constant), one must have, $V(\tilde{\theta}_{t|t}|y_{1:t}) = V(\hat{\theta}_{t-1|t-1}|y_{1:t}) = V_{\hat{\theta}_{t-1|t-1}}$, which implies that, $C(\hat{\theta}_{t-1}, \zeta_t|Y_t) + C(P_t\Psi\epsilon_t, \zeta_t|Y_t) = -\frac{1}{2}W_t - \frac{1}{2}P_t^2\Psi V_t\Psi^{\mathrm{T}}$. Therefore, negative correlations are needed to remove the effects of unwanted information loss. In case of approximate joint normality of the stochastic process $(\hat{\theta}_{t-1|t-1}, \zeta_t|Y_t)$ and $(P_t\Psi\epsilon_t, \zeta_t|Y_t)$, the conditional normal evolution is obtained as

$$p(\hat{\theta}_{t|t}|\hat{\theta}_{t-1|t-1}) \sim \mathcal{N}(\hat{\theta}_{t|t}|A_t\tilde{\theta}_{t|t} + (I - A_t)\hat{\theta}_{t-1|t-1}, (I - A_t^2)V_{\hat{\theta}_{t-1|t-1}}), \tag{3.23}$$

where the mean of this Gaussian distribution at each time step is found from equation (3.17a), when $\hat{\theta}_{t-1|t-1}$ is substituted by its modified version according to the shrinkage kernel. The shrinkage matrix A_t, is obtained as $A_t = I - [\frac{1}{2}(W_tV^{-1}_{\theta_{t-1|t-1}} + P_t^2\Psi V_y\Psi^{\mathrm{T}}V^{-1}_{\theta_{t-1|t-1}})]$. Assuming that in the kernel shrinkage method, the variance of the evolution noise is interpreted as $W_t = (I - A_t^2)V_{\hat{\theta}_{t-1|t-1}}$, therefore by replacing W_t in the equation that was obtained for A_t results in

$$A_t = I - \frac{1}{2}[(I - A_t^2) + P_t^2\Psi V_t\Psi^{\mathrm{T}}V^{-1}_{\theta_{t-1|t-1}}]. \tag{3.24}$$

Let us assume that our main goal is to obtain and determine the shrinkage matrix A_t as $A = aI$, therefore the matrix equation (3.24) can be written as

$$(a^2 - 2a + 1)I - P_t^2\Psi V_t\Psi^{\mathrm{T}}V^{-1}_{\theta_{t-1|t-1}} = 0. \tag{3.25}$$

We are interested in obtaining an upper bound for the shrinkage matrix that can be used for all time. Assuming that the last term in the right hand side of equation (3.24) has an upper bound given by $|P_t^2\Psi V_t\Psi^{\mathrm{T}}V^{-1}_{\theta_{t-1|t-1}}| \leq P^2_{\max}\Psi V_y\Psi^{\mathrm{T}}V^{-1}_{\hat{\theta}}$, where $P_{\max} = \gamma_0\sqrt{\mathrm{trace}(\mathcal{E}_{\max}\mathcal{E}^{\mathrm{T}}_{\max})}$ with γ_0 denoting the initial step size, therefore $\mathcal{E}_{\max}$ is considered as the maximum acceptable variance of the prediction error, V_y is an upper bound of the measurement noise covariance, and $V_{\hat{\theta}}$ is the minimum bound of the parameter estimation covariance that is considered to be similar to the initial covariance of the parameters (before adding the evolution noise in time). Therefore,

a bound on aI and consequently A can be obtained as $A = aI \leq I - \sqrt{\frac{\sigma_{\min}(P^2_{\max}\Psi V_y \Psi^{\mathrm{T}} V^{-1}_{\hat{\theta}})}{\sigma_{\max}(P^2_{\max}\Psi V_y \Psi^{\mathrm{T}} V^{-1}_{\hat{\theta}})}} I$, where the normalization of the eigenvalue is performed to ensure that the associated fraction remains less than 1. Let $a = 1 - \sqrt{\frac{\sigma_{\min}(P^2_{\max}\Psi V_y \Psi^{\mathrm{T}} V^{-1}_{\hat{\theta}})}{\sigma_{\max}(P^2_{\max}\Psi V_y \Psi^{\mathrm{T}} V^{-1}_{\hat{\theta}})}}$, therefore, the shrinkage matrix becomes $A = aI$. The smoothing matrix corresponding to the normal distribution variance is now obtained from the shrinkage factor as $(1 - a^2)I$. This guarantees that the distribution (3.23) has a finite variance as $t \to \infty$ for both constant and time-varying parameter estimation cases. This completes the proof of the proposition. ■

The convergence of the estimated parameter particles $\hat{\theta}^{(j)}_{t-1|t-1},\ j = 1, ..., N$ to the local minimum of $\mathbb{E}(\bar{J}(\hat{\theta}^{(j)}_{t-1|t-1})|y_{1:t-1}, \hat{x}_t)$ is now investigated in the following subsection. The developed convergence proof does not ensure the convergence of the PE-based parameter estimation method to the true parameter value, but only to a set of zeros of the gradient of the chosen performance index. The conditions under which the convergence of the estimated parameters to their optimal values can be guaranteed as $N \to \infty$ is stated in Remark 1.

3.2.6 Convergence of the PE-based Parameter Update Law

In this subsection, it will be shown that the update law (3.17a)-(3.17b) can guarantee the convergence of the parameter estimate particles $\hat{\theta}^{(j)}_{t-1|t-1},\ j = 1, ..., N$ (after the resampling step), to a local minimum of $\mathbb{E}(\bar{J}(\hat{\theta}^{(j)}_{t-1|t-1})|y_{1:t-1}, \hat{x}_t)$, that is located in a compact set of $\{x_t, \theta_t\}$, denoted by $D_{\mathcal{N}}$ as per Assumption 3.1.

In order to investigate the convergence of our proposed PE-based modified artificial evolu-

tion law for updating the parameter particles distribution and to achieve a local minimization of $\mathbb{E}(\bar{J}(\theta_{t-1})|y_{1:t-1}, x_t)$, consider equation (3.17a), where γ_t denotes a time-varying step size such that $\lim_{t\to\infty}\gamma_t = \mu_0 > 0$, where μ_0 is a small positive constant. The introduction of the step size γ_t is necessary to transform the discrete-time model (3.17a)-(3.17b) into a continuous-time representation as shown subsequently.

First, we need to state the following two assumptions 3.2-3.3 according to [165], to guarantee the convergence of our proposed algorithm as presented in our main result below in Theorem 1. Specifically, we have:

Assumption 3.2. The function $Q(\epsilon(t, \hat{\theta}_{t-1|t-1}))$ is sufficiently smooth and twice continuously differentiable w.r.t. ϵ, and $|Q_{\epsilon\epsilon}(\epsilon(t, \hat{\theta}_{t-1|t-1}))| \leq C$ for $\hat{\theta}_{t-1|t-1} \in D_{\mathcal{N}}$, where $Q_{\epsilon\epsilon}(\epsilon(t, \hat{\theta}_{t-1|t-1}))$ denotes the second derivative of $Q(\epsilon(t, \hat{\theta}_{t-1|t-1}))$ w.r.t. ϵ.

Assumption 3.3. The observation sequence y_t (generated from equation (3.2)), is such that $\bar{\mathbb{E}}(Q(\epsilon(t, \hat{\theta}_{t-1|t-1}))) = \bar{J}(\hat{\theta}_{t-1|t-1})$ and $\bar{\mathbb{E}}[\frac{\mathrm{d}}{\mathrm{d}\hat{\theta}_{t-1|t-1}} Q(\epsilon(t, \hat{\theta}_{t-1|t-1}))] = -g(\hat{\theta}_{t-1|t-1})$ exist for all $\hat{\theta}_{t-1|t-1} \in D_{\mathcal{N}}$, where $\bar{\mathbb{E}}(Q(\epsilon(t, \hat{\theta}_{t-1|t-1}))) = \frac{1}{\kappa}\sum_{\tau=t-\kappa}^{t} \mathbb{E}Q(\epsilon(\tau, \hat{\theta}_{t-1|t-1}))$.

It must be noted that the kernel shrinkage method, as stated earlier, attempts to retain the mean of the parameter estimation particles at time t near the estimated parameter in the previous time step $t-1$, i.e. $\hat{\theta}_{t-1|t-1}$. Therefore, in the following theorem the convergence properties of $\hat{\theta}_{t-1|t-1}$ is addressed. The main result of this section is stated below.

Theorem 3.1 *Consider the parameter estimation algorithm as specified by the equations* (3.17a)-(3.19). *Also consider the a posteriori parameter estimate as governed by equation* (3.20). *Let Assumptions 3.1 to 3.3 hold. It now follows that the particles* $\hat{\theta}^{(j)}_{t-1|t-1},\ j = 1, ..., N$, *and conse-*

quently the distribution of the estimated parameter particles approximated by the particle filter $\pi^N_{\theta_{t-1|t-1}}(\mathrm{d}\theta_{t-1})$*, w.p.1 converge either to the set* $D_{\mathcal{C}} = \{\hat{\theta}^{(j)}_{t-1|t-1} | \hat{\theta}^{(j)}_{t-1|t-1} \in D_{\mathcal{N}},\ \frac{\mathrm{d}}{\mathrm{d}\hat{\theta}^{(j)}_{t-1|t-1}} \bar{J}(\hat{\theta}^{(j)}_{t-1|t-1}) = 0,\ j = 1, ..., N\}$ *or to the boundary of* $D_{\mathcal{N}}$ *as* $t \to \infty$.

Proof: The existence of the projection algorithm in the parameter estimation scheme ensures that $\tilde{\theta}^{(j)}_{t|t}$ remains inside $D_{\mathcal{N}}$. According to equation (3.20), the *a posteriori* estimate of the parameter at time t is obtained from the resampled particles of the parameter estimate $\hat{\theta}^{(j)}_{t|t}$, as $\hat{\theta}_{t|t} = \frac{1}{N}\sum_{j=1}^{N} \hat{\theta}^{(j)}_{t|t}$, where $\hat{\theta}^{(j)}_{t|t}$ is selected from the N particles of $\tilde{\theta}^{(j)}_{t|t}$ for which $\rho_{\nu_t}(y_t - h(\hat{x}_{t|t}, \tilde{\theta}^{(j)}_{t|t}))$ yields higher probabilities. In order to avoid the discontinuity that is caused by resampling, in this procedure only the particles that are maintained from time $t-1$, i.e. $\hat{\theta}^{(j)}_{t-1|t-1}$ and are propagated to time t as $\hat{\theta}^{(j)}_{t|t}$, are considered. However, the rest of the particles that are to be discarded in the resampling process will be replaced by the kept particles. Therefore, the results can be generalized to all the particles.

Consider equation (3.17b) for generating $\tilde{\theta}^{(j)}_{t|t}$ and let us substitute $m^{(j)}_t$ from the PE-based update rule of (3.17a)-(3.17b) to obtain the following expression for the resampled particles $\hat{\theta}^{(j)}_{t|t}$, namely

$$\hat{\theta}^{(j)}_{t|t} = A\hat{\theta}^{(j)}_{t-1|t-1} + (I - A)\frac{1}{N}\sum_{i=1}^{N} \hat{\theta}^{(i)}_{t-1|t-1} + A\gamma_t R^{(j)}_t \psi^{(j)}_t \epsilon(t, \hat{\theta}^{(j)}_{t-1|t-1}) + \sqrt{I - A^2}\zeta^{(j)}_t, \quad (3.26)$$

where $\sqrt{I - A^2}\zeta^{(j)}_t$ denotes the evolution noise by taking into account the kernel smoothing

concept. By applying the sum operator to both sides of equation (3.26) to construct $\hat{\theta}_{t|t}$ yields,

$$\frac{1}{N}\sum_{j=1}^{N}\hat{\theta}_{t|t}^{(j)} = A\frac{1}{N}\sum_{j=1}^{N}\hat{\theta}_{t-1|t-1}^{(j)} + \frac{1}{N}\sum_{i=1}^{N}\frac{1}{N}\sum_{i=1}^{N}\hat{\theta}_{t-1|t-1}^{(j)} - A\frac{1}{N}\sum_{i=1}^{N}\frac{1}{N}\sum_{i=1}^{N}\hat{\theta}_{t-1|t-1}^{(j)}$$
$$+ A\frac{1}{N}\sum_{j=1}^{N}\gamma_t R_t^{(j)}\psi_t^{(j)}\epsilon(t,\hat{\theta}_{t-1|t-1}^{(j)}) + \sqrt{I-A^2}\frac{1}{N}\sum_{j=1}^{N}\zeta_t^{(j)} = \frac{1}{N}\sum_{j=1}^{N}\hat{\theta}_{t-1|t-1}^{(j)}$$
$$+ A\frac{1}{N}\sum_{j=1}^{N}\gamma_t R_t^{(j)}\psi_t^{(j)}\epsilon(t,\hat{\theta}_{t-1|t-1}^{(j)}),$$

which results in $\hat{\theta}_{t|t} = \hat{\theta}_{t-1|t-1} + A\frac{1}{N}\sum_{j=1}^{N}\gamma_t R_t^{(j)}\psi_t^{(j)}\epsilon(t,\hat{\theta}_{t-1|t-1}^{(j)})$. Assumptions 3.1 and 3.2 ensure that the regularity conditions are satisfied according to [165]. Consequently, a differential equation associated with (3.17a)-(3.17b) can be obtained by considering that $\Delta\tau$ is a sufficiently small number and $t, \check{t}$ are specified such that $\sum_{k=t}^{\check{t}}\gamma_k = \Delta\tau$. Through a change of time-scales as $t \to \tau$ and $\check{t} \to \tau + \Delta\tau$, for a sufficiently small $\Delta\tau$, and by assuming that $\hat{\theta}_{t-1|t-1} = \check{\theta}$, $R_t^{(j)} = \check{R}^{(j)}$, $A = aI$ is a constant matrix, the difference equation for $\hat{\theta}_{t|t}^{(j)}$ is now expressed as

$$\hat{\theta}_{\check{t}}^{(j)} \approx \check{\theta}^{(j)} + a\Delta\tau\check{R}^{(j)}g(\check{\theta}^{(j)}), \tag{3.27}$$

where $g(\check{\theta}^{(j)}) = \frac{1}{\Delta\tau}\sum_{k=t}^{t=\check{t}}\psi_t^{(j)}\epsilon(k,\hat{\theta}_{t-1|t-1}^{(j)})$. In the above derivation it is assumed that the $\hat{\theta}_{t|t}^{(j)}$ particle is kept after resampling (that is $\hat{\theta}_{t-1|t-1}^{(j)} \to \tilde{\theta}_{t|t}^{(j)} \to \hat{\theta}_{t|t}^{(j)}$). Consequently, considering Assumption 3.3, the differential equation associated with the evolution of each single particle is obtained as,

$$\frac{\mathrm{d}\theta_D^{(j)}}{\mathrm{d}\tau} = aR_D^{(j)}(\tau)g(\hat{\theta}_D^{(j)}(\tau)) = -aR_D^{(j)}(\tau)[\frac{\mathrm{d}}{\mathrm{d}\hat{\theta}_D^{(j)}}\bar{J}(\hat{\theta}_D^{(j)})]^{\mathrm{T}}, \tag{3.28}$$

where the subscript D is used to differentiate the solution of the differential equation (3.28) from the solution of the difference equation (3.27). Now, the required convergence analysis is reduced to investigating the properties of the deterministic continuous-time system (3.28).

Consider the positive definite function $L(\hat{\theta}_{t-1|t-1}^{(j)}) = \mathbb{E}(\bar{J}(\hat{\theta}_{t-1|t-1}^{(j)})) = \frac{1}{N}\sum_{j=1}^{N}\bar{J}(\hat{\theta}_{t-1|t-1}^{(j)})$

that represents the expectation of a positive definite function through N data points for $\hat{\theta}^{(j)}_{t-1|t-1}$. Our goal is to evaluate the derivative of this function along the trajectories of the system (3.28). According to Assumption 3.2, the second derivative of $Q(\epsilon(t, \hat{\theta}^{(j)}_{t-1|t-1}))$ is bounded, therefore the summation and derivative operations commute. According to Assumption 3.3 for $\hat{\theta}^{(j)}_{t-1|t-1} \in D_{\mathcal{N}}$, $\acute{\bar{J}}(\hat{\theta}^{(j)}_{D}(\tau)) = \frac{\mathrm{d}}{\mathrm{d}\hat{\theta}^{(\mathrm{j})}_{\mathrm{t-1|t-1}}} \bar{J}(\hat{\theta}^{(j)}_{t-1|t-1})|_{\hat{\theta}^{(j)}_{t-1|t-1}=\check{\theta}^{(j)}} = \bar{\mathbb{E}}(\frac{\mathrm{d}}{\mathrm{d}\hat{\theta}_{\mathrm{t-1|t-1}}} Q(\epsilon(t, \hat{\theta}^{(j)}_{t-1|t-1}))$, exists and is approximated by $-g(\check{\theta}^{(j)})$. Therefore, let us define $V(\hat{\theta}^{(j)}_{D}) = \mathbb{E}(\bar{J}(\hat{\theta}^{(j)}_{D}(\tau))$, and given that $a > 0$, and $R^{(j)}_{D}(\tau)$ is a positive scalar for $j = 1, ..., N$ (which represents the trace of a positive definite matrix at time τ), one gets

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}\tau} V(\hat{\theta}^{(j)}_{D}) &= \mathbb{E}(\frac{\mathrm{d}}{\mathrm{d}\tau} \bar{J}(\hat{\theta}^{(j)}_{D}(\tau)) = \frac{1}{N} \sum_{j=1}^{N} \acute{\bar{J}}(\hat{\theta}^{(j)}_{D}) \frac{\mathrm{d}}{\mathrm{d}\tau} \hat{\theta}^{(j)}_{D}(\tau) \\ &= \frac{-a}{N} \sum_{j=1}^{N} [g(\hat{\theta}^{(j)}_{D}(\tau))] R^{(j)}_{D}(\tau) [g(\hat{\theta}^{(j)}_{D})(\tau)]^{\mathrm{T}} \leq 0, \end{aligned} \tag{3.29}$$

where the equality is obtained only for $\hat{\theta}_D(\tau) \in D_{\mathcal{C}}$. Therefore, as $t \to \infty$ either $\hat{\theta}^{(j)}_{t-1|t-1}$ and consequently, $\pi^{N}_{\theta_{t-1|t-1}}$ w.p.1 tends to $D_{\mathcal{C}}$ or to the boundary of $D_{\mathcal{N}}$, where w.p.1 is with respect to the random variables related to the parameter estimate particles. It should be noted that for particles that have been replaced in the resampling this equality is valid since they are replaced by particles that have satisfied (3.29). This completes the proof of the theorem. ■

The main reason that our proposed dual state/parameter estimation method for its implementation does not necessarily need more particles than the one that we needed for only the state estimation scheme, is illustrated by the result that one can extract and obtain from Theorem 3.1. According to this theorem, PE-based modified artificial evolution law enables each single particle to tend to $D_{\mathcal{C}}$. Therefore, even increasing the number of particles would not affect the convergence properties of the filter but it can certainly result in a more accurate state/parameter

estimates. It was indicated earlier that the above theorem can only guarantee the boundedness of the estimated parameter distribution from the particle filters and not its convergence to the optimal distribution. However, in reality it is not possible to find an exact dynamical equation for the variations of the system health parameters since they can be affected by fault and/or damages during the normal operation of the system. As a result, the optimal values of the estimated parameters can not be guaranteed unless the Assumptions 3.1, 3.2, and 3.3 are not violated. Therefore, based on the results of Theorem 3.1 and Proposition 3.1 one can ensure that the probability density function and its related kernel $K_\theta(\mathrm{d}\theta_t|\theta_{t-1}, x_{t-1})$ (in the particle filter) do remain bounded. Then, the convergence of the dual state and parameter estimation algorithm can be investigated based on Theorem 3.1 in [169].

Remark 3.1 *Using the extended setting that is introduced in [169], and also by assuming that* $\rho(y_t|x_t, \hat{\theta}_t) < \infty$, *and* $K_x(x_t|x_{t-1}, \hat{\theta}_{t-1}) < \infty$, *the boundedness of the parameter estimation transition kernel* $K_\theta(\hat{\theta}_t|\hat{\theta}_{t-1}, \hat{x}_t)$ *is also ensured from the Proposition 3.1 and Theorem 3.1. Therefore, the convergence of the proposed dual state/parameter estimation filter to their optimal distributions, for* $\{x_t, \theta_t\} \in D_{\mathcal{N}}$ *as* $N \to \infty$ *can be investigated according to Theorem 3.1 that is provided in [169].*

3.2.7 Equivalent Flop Complexity Analysis of Dual State/Parameter Estimation Algorithm

In this section, the computational complexity of our proposed dual state and parameter estimation method is quantitatively obtained and analysed. The analysis is based on the number of

floating-point operations (flops) that are required by the selected algorithms. A flop is defined as one addition, subtraction, multiplication, or division of two floating-point numbers. However, there are certain algorithms where their complexity cannot be measured by using flops, for example for generating random numbers and for evaluating a nonlinear function.

On the other hand, the relationship between flop complexity and time complexity in execution of an algorithm depends on many other factors that are not necessarily reflected in the flop complexity measure. Therefore, in this chapter the equivalent flop (EF) complexity introduced in [170] for an operation will be utilized for conducting the complexity analysis. In the EF complexity, the number of flops that result in the same computational time as a given operation is evaluated through the so-called proportionality coefficients. Thereby, it follows how the computational time will increase as the problem size increases. The EF metric is mostly evaluated for those operations that depend on matrix and vector manipulations.

The dimensions and definitions of some of the entities that are used in the EF analysis of our proposed PF-based state and parameter estimation algorithm are provided in Table 3.1. The coefficients c_1, c_2, and c_3 are used to represent the complexity of random number generation, resampling, and reguralization, respectively, since their complexities cannot be measured by using flops. Therefore, their complexities have to be estimated by analyzing the actual computational time that is consumed by various segments of the algorithm [170]. It should be noted that Gaussian likelihood calculations are also included in the resampling step. The Schur decomposition is also used in the process of covariance matrix decomposition to generate the new particle populations in the filters. The complexity of the above operation increases cubic as the

Table 3.1: Definitions and Dimensions of the entities in the dual state and parameter estimation algorithm

Variable	Dimension	Definition
x_t	$\mathbb{R}^{n_x}$	state vector
θ_t	$\mathbb{R}^{n_\theta}$	parameter vector
Σ_x	$\mathbb{R}^{n_x \times n_x}$	state estimate covariance matrix
P_t	$\mathbb{Z}^+$	adaptive step size
A	$\mathbb{R}^{n_\theta \times n_\theta}$	shrinkage matrix
Σ_θ	$\mathbb{R}^{n_\theta \times n_\theta}$	parameter estimate covariance matrix
W_t	$\mathbb{R}^{n_x \times n_x}$	process noise variance
V_{θ_t}	$\mathbb{R}^{n_\theta \times n_\theta}$	parameter estimate distribution variance
y_t	$\mathbb{R}^{n_y}$	measurement output
$f_t(x_t, \theta_t^{\mathrm{T}}\lambda(x_t), \omega_t)$	$\mathbb{R}^{n_x \times 1}$	state dynamic function
$h_t(x_t, \theta_t^{\mathrm{T}}\lambda(x_t))$	$\mathbb{R}^{n_y \times 1}$	observation function

Table 3.2: Computational Complexity of some Common Matrix Operations

Operation	Matrix size	Multiplication	Addition
$A + A$	$A \in \mathbb{R}^{n \times m}$	$-$	nm
$A.B$	$A \in \mathbb{R}^{n \times m}, B \in \mathbb{R}^{m \times l}$	lmn	$(m-1)ln$
$B.C$	$B \in \mathbb{R}^{m \times n}, C \in \mathbb{R}^{n \times 1}$	nm	$(n-1)m$
D^{-1}	$D \in \mathbb{R}^{n \times n}$	n^3	$-$

dimension of the problem increases. The complexity related to initialization is ignored in the tables since initialization of state and parameter estimation filters are performed only once and in the start of the algorithm execution.

The details regarding to EF complexity analysis of our proposed dual algorithm, conventional Bayesian method for state and parameter estimation [2] (where Regularized particle filter structure is utilized to implement the filter), and recursive maximum likelihood method according to simultaneous perturbation stochastic approximation algorithm are presented in Tables 3.6, and 3.7. According to the summarized results in Tables 3.4, 3.5, 3.6, and 3.7, the EF complexity of the presented algorithms are presented in Table 3.3 where only dominant parts of $C(n_x, n_\theta, c_1, c_2, c_3, N)$ (that represents the EF complexity of our proposed dual algorithm), $\acute{C}(n_x, n_\theta, c_1, c_2, c_3, N)$ (that represents the EF complexity of the conventional Bayesian

method [2]), and $\acute{C}(n_x, n_\theta, c_1, c_2, c_3, N)$ (that represents the EF complexity of the recursive maximum likelihood method according to SPSA algorithm) are provided. This selection is justified by the fact that $N \gg 1$, therefore the dominant parts are the parts that are related to N.

To achieve the same complexity in the dual estimation algorithm, Bayesian method and RML parameter estimation algorithm based on particle filters, the number of required particles in our proposed dual estimation method can be determined based on the number of the particles in the other two methods denoted by $\acute{N}$, and $\acute{\acute{N}}$ as

$$N = \acute{N}(1 - \frac{2n_\theta^2 + 5n_\theta + 2n_\theta n_y + 6n_y + 2n_\theta - 6n_x n_\theta - c_3 n_\theta}{3n_x^2 + 5n_\theta^2 + 6n_\theta + 2n_\theta n_y + 7n_y + 3n_x + c_1(n_x + n_\theta) + c_2(n_x + n_\theta) + c_3 n_x)}) \tag{3.30}$$

$$N = \acute{\acute{N}}(1 - \frac{n_x^2 + 5n_\theta^2 + 2n_\theta + n_x + 2n_\theta n_y + 7n_y + c_2 n_\theta}{3n_x^2 + 5n_\theta^2 + 6n_\theta + 2n_\theta n_y + 7n_y + 3n_x + c_1(n_x + n_\theta) + c_2(n_x + n_\theta) + c_3 n_x)}) \tag{3.31}$$

Consider that in equation (3.30) $2n_\theta^2+5n_\theta+2n_\theta n_y+6n_y+2n_\theta-6n_x n_\theta-c_3 n_\theta > 3n_x^2+5n_\theta^2+6n_\theta+2n_\theta n_y+7n_y+3n_x+c_1(n_x+n_\theta)+c_2(n_x+n_\theta)+c_3 n_x)$, since the complexity corresponding to regularization (c_3) is assumed to be the dominant complexity term in the nominator of the coefficient of $\acute{N}$, therefore the coefficient of $\acute{N}$ is greater than one which indicates that in order to achieve the same complexity from dual estimation algorithm and conventional Bayesian method using regularized particle filter structure, the number of the required particles in the Bayesian method should be selected less than the number of particles in the dual estimation method. On the other hand, the similar analysis for the RML method according to (3.31) shows that since $n_x^2 + 5n_\theta^2 + 2n_\theta + n_x + 2n_\theta n_y + 7n_y + c_2 n_\theta < 3n_x^2 + 5n_\theta^2 + 6n_\theta + 2n_\theta n_y + 7n_y + 3n_x + c_1(n_x +$

Table 3.3: The Approximated Total Equivalent Complexity of the Filters

Prediction Method	Total Equivalent Complexity
Dual Estimation Algorithm	$C(n_x, n_\theta, c_1, c_2, c_3, N) \approx N(3n_x^2 + 5n_\theta^2 + 6n_\theta + 2n_\theta n_y + 7n_y + 3n_x + c_1(n_x + n_\theta) + c_2(n_x + n_\theta) + c_3 n_x)$
Standard Bayesian PF-Based estimation Method	$\acute{C}(n_x, n_\theta, c_1, c_2, c_3, N) \approx N(3n_x^2 + 3n_\theta^2 + 6n_x n_\theta + (1 + c_1 + c_2 + c_3)n_x + (1 + c_1 + c_2 + c_3)n_\theta + n_y)$
PF-based RML Parameter Estimation Method	$\acute{C}(n_x, n_\theta, c_1, c_2, c_3, N) \approx N(2n_x^2 + 4n_\theta + 2n_x + c_1(2n_x + n_\theta) + c_2 n_x + c_3 n_x)$

Table 3.4: The Equivalent Complexity for the state estimation step in Dual Structure

Instruction	Mult.	Add	Func. Eval.	Other
$[U_1, T_1] = \text{schur}(\Sigma_{\hat{x}_{t-1\|t}})$	–	–	–	$10n_x^3$
$R_1 = \text{randn}(n_x, N)$	–	–	–	$Nn_x c_1$
$\omega_t^{(i)} = (U_1\sqrt{T_1})R_1$	$n_x^3 + Nn_x^2$	$(n_x - 1)n_x^2 + N(n_x - 1)n_x$	–	n_x^2
$\hat{x}_{t\|t-1}^{(i)} = f_t(\hat{x}_{t-1\|t-1}^{(i)}, \hat{\theta}_{t-1\|t-1}^{\mathrm{T}}\lambda(\hat{x}_{t\|t-1}^{(i)}), \omega_t^{(i)})$	–	–	Nn_x	–
$\hat{y}_{t\|t-1}^{(i)} = h_t(\hat{x}_{t\|t-1}^{(i)}, \hat{\theta}_{t-1\|t-1}^{\mathrm{T}}\lambda(x_{t\|t-1}^{(i)}))$	–	–	Nn_y	–
$\Sigma_{\hat{x}_{t\|t-1}} = \frac{1}{N-1}\sum_{i=1}^{N}(\bar{x}_{t\|t-1}^{(i)} - \hat{x}_{t\|t-1})(\bar{x}_{t\|t-1}^{(i)} - \hat{x}_{t\|t-1})^{\mathrm{T}}$	Nn_x^2	$2Nn_x$	–	–
Regularization and resampling to find weights $w_{x_t}^{(i)}$ and $\hat{x}_{t\|t}^{(i)}$	–	–	–	$Nn_x c_2 + Nn_x c_3$
$\hat{x}_{t\|t} = \frac{1}{N}\sum_{i=1}^{N}\hat{x}_{t\|t}^{(i)}$	n_x	Nn_x	–	–
Total	$n_x^3 + 2Nn_x^2 + n_x$	$n_x^3 + (N-1)n_x^2 + 2Nn_x$	$N(n_x + n_y)$	$10n_x^3 + n_x^2$ $Nn_x(c_1 + c_2 + c_3)$

$n_\theta) + c_2(n_x + n_\theta) + c_3 n_x)$, therefore unlike the Bayesian method, in the RML method one needs more particles in order to achieve the same computational complexity with the dual estimation algorithm. Finally, we utilize, the EF complexity results to measure the time complexity of each algorithm considering the fact that EF complexity is proportional to the time complexity of the algorithm. In the following tables the EF complexity for three different methods i.e. our proposed dual algorithm, conventional Bayesian method for state and parameter estimation [2] (where Regularized particle filter structure is utilized to implement the filter), and recursive maximum likelihood method according to simultaneous perturbation stochastic approximation algorithm are presented, respectively.

Our proposed dual state and parameter estimation scheme is an effective methodology for the purpose of fault diagnosis of nonlinear systems, where without loss of any generality one initiates operating the system from the healthy mode of operation. During the healthy operation

Table 3.5: The Equivalent Complexity for the parameter estimation step in Dual Structure

Instruction	Mult.	Add	Func. Eval.	Other
$\bar{y}_{t\|t-1}^{(j)} = h_t(\hat{x}_{t\|t}, \theta_{t-1\|t-1}^{(j)^{\mathrm{T}}}\lambda(\hat{x}_{t\|t}))$	–	–	Nn_y	–
$\Sigma_\theta = (I - A^2)\Sigma_{\hat{\theta}_{t-1\|t-1}}$	n_θ^3	$(n_\theta - 1)n_\theta^2 + n_\theta^2$	–	–
$\epsilon_t^{(j)} = y_t - \bar{y}_{t\|t-1}^{(j)}$	–	Nn_y	–	–
$\psi_t^{(j)} = \frac{dh}{d\theta}\|_{\hat{x}_{t\|t}, \theta_{t-1\|t-1}^{(j)}}$	–	–	$n_y n_\theta$	–
$P_t^{(j)} = \gamma_t(\sqrt{\mathrm{trace}(\epsilon_t^{(j)}\epsilon_t^{(j)^{\mathrm{T}}})}$	$N + Nn_y$	$N(n_y - 1) + Nn_y$	–	–
$[U_2, L_2] = \mathrm{schur}(\Sigma_\theta)$	–	–	–	$10n_\theta^3$
$R_2 = \mathrm{randn}(n_\theta, N))$	–	–	–	$Nn_\theta c_1$
$\zeta_t^{(j)} = (U_2\sqrt{L_2})R_2$	$n_\theta^3 + Nn_\theta^2$	$(n_\theta - 1)n_\theta^2 + N(n_\theta - 1)n_\theta$	n_θ^2	–
$m_t^{(j)} = \hat{\theta}_{t-1\|t-1}^{(j)} + P_t^{(j)}\psi_t^{(j)}\epsilon_t^{(j)}$	$N(n_y n_\theta + n_\theta)$	$N(n_y - 1)n_\theta + Nn_\theta$	–	–
$\tilde{\theta}_{t\|t}^{(j)} = Am_t^{(j)} + (I - A)\frac{1}{N}\sum_{j=1}^N \hat{\theta}_{t-1\|t-1}^{(j)} + \zeta_t^{(j)}$	$Nn_\theta^2 + n_\theta$	$Nn_\theta^2 + 2Nn_\theta + Nn_\theta + n_\theta^2$	–	–
$\bar{y}_{t\|t} = h_t(\hat{x}_{t\|t}, \tilde{\theta}_{t\|t}^{(j)^{\mathrm{T}}}\lambda(\hat{x}_{t\|t}))$	–	–	Nn_y	–
Resampling to find weights, $w_{\theta_t}^{(j)}$, and $\hat{\theta}_{t\|t}^{(j)}$	–	–	–	$Nn_\theta c_2$
$\hat{\theta}_{t\|t} = \frac{1}{N}\sum_{j=1}^N \hat{\theta}_{t\|t}^{(j)}$	n_θ	Nn_θ	–	–
$\Sigma_{\hat{\theta}_{t\|t}} = \frac{1}{N-1}\sum_{i=1}^N (\hat{\theta}_{t\|t}^{(j)} - \hat{\theta}_{t\|t})(\hat{\theta}_{t\|t}^{(j)} - \hat{\theta}_{t\|t})^{\mathrm{T}}$	Nn_θ^2	$2Nn_\theta$	–	–
Total	$2n_\theta^3 + 3Nn_\theta^2$ $+(N+2)n_\theta + Nn_\theta n_y$ $+N(n_y + 1)$	$2n_\theta^3 + 2Nn_\theta^2$ $+5Nn_\theta + 3Nn_y - N$ $+Nn_\theta n_y$	$n_\theta^2 + 2Nn_y$ $+n_y n_\theta$	$10n_\theta^3 + Nn_\theta c_1 + Nn_\theta c_2$ – –

Table 3.6: The Equivalent Complexity for the augmented state and parameter estimation scheme [2]

Instruction	Mult.	Add	Func. Eval.	Other
$[U_1, T_1] = \mathrm{schur}(\Sigma_{x,\theta})$	–	–	–	$10(n_x + n_\theta)^3$
$R_1 = \mathrm{randn}(n_x + n_\theta, N)$	–	–	–	$N(n_x + n_\theta)c_1$
$\omega_t^{(i)} = (U_1\sqrt{T_1})R_1$	$(n_x + n_\theta)^3 + N(n_x + n_\theta)^2$	$(n_x + n_\theta - 1)(n_x + n_\theta)^2$ $+N(n_x + n_\theta - 1)(n_x + n_\theta)$	–	$(n_x + n_\theta)^2$
$\omega_{x_t}^{(i)} = \omega_t^{(i)}(1 : n_x)$	–	–	–	–
$\omega_{\theta_t} = (I - A^2)\omega_t^{(i)}(n_x + 1 : n_x + n_\theta)$	n_θ^3	$(n_\theta - 1)n_\theta^2 + n_\theta^2$	–	–
state/parameter augmentation: $[\hat{x}_{t\|t-1}^{(i)}; \hat{\theta}_{t\|t-1}^{(i)}] =$ $[f_t(\hat{x}_{t\|t-1}^{(i)}, \hat{\theta}_{t\|t-1}^{(i)^{\mathrm{T}}}\lambda(\hat{x}_{t\|t-1}^{(i)}), \omega_{x_t}^{(i)}); \hat{\theta}_{t\|t-1}^{(i)}]$	–	–	$N(n_x + n_\theta)$	–
$\hat{y}_{t\|t-1}^{(i)} = h_t(\hat{x}_{t\|t-1}^{(i)}, \hat{\theta}_{t\|t-1}^{(i)^{\mathrm{T}}}\lambda(\hat{x}_{t\|t-1}^{(i)}))$	–	–	Nn_y	–
$\Sigma_{x,\theta} = \frac{1}{N-1}\sum_{i=1}^N ([\bar{x}_{t\|t-1}^{(i)}; \bar{\theta}_{t\|t-1}^{(i)}] - [\hat{x}_{t\|t-1}; \hat{\theta}_{t\|t-1}])$ $\times([\bar{x}_{t\|t-1}^{(i)}; \bar{\theta}_{t\|t-1}^{(i)}] - [\hat{x}_{t\|t-1}; \hat{\theta}_{t\|t-1}])^{\mathrm{T}}$	$N(n_x + n_\theta)^2$	$2N(n_x + n_\theta)$	–	–
Regularization to find weights and resampling to find, $\hat{x}_{t\|t}^{(i)}$ and $\hat{\theta}_{t\|t}^{(i)}$	–	–	–	$N(n_x + n_\theta)(c_3 + c_2)$
$[\hat{x}_{t\|t}; \hat{\theta}_{t\|t}] = [\frac{1}{N}\sum_{i=1}^N \hat{x}_{t\|t}^{(i)}; \frac{1}{N}\sum_{i=1}^N \hat{\theta}_{t\|t}^{(i)}]$	$n_x + n_\theta$	$N(n_x + n_\theta)$	–	–
Total	$2n_\theta^3 + n_x^3 + n_x^2(2N + 3n_\theta)$ $+n_\theta^2(3n_x + 2N) + 4Nn_\theta n_x$ $+n_x + n_\theta$	$n_x^3 + 2n_\theta^3 + n_x^2(3n_\theta - 1 + N)$ $+n_\theta^2(N - 1 + 3n_x)$ $+n_x n_\theta(2N - 2)$	$N(n_x + n_\theta)$ $+Nn_y$	$10n_x^3 + 10n_\theta^3$ $n_x^2(30n_\theta + 1)$ $+n_\theta^2(30n_x + 1)$ $+n_x(Nc_1 + 2n_\theta + Nc_3)$ $+n_\theta(Nc_1 + Nc_3)$

Table 3.7: The Equivalent Complexity for the Recursive Maximum Likelihood Parameter scheme based on Particle Filters Using SPSA [3]

Instruction	Mult.	Add	Func. Eval.	Other
$[U_1, T_1] = \text{schur}(\Sigma_x)$	–	–	–	$10n_x^3$
$R_1 = \text{randn}(n_x, N)$	–	–	–	Nn_xc_1
$\omega_t^{(i)} = (U_1\sqrt{T_1})R_1$	$n_x^3 + Nn_x^2$	$(n_x-1)n_x^2 + N(n_x-1)n_x$	–	n_x^2
Generate random perturbation vector Δ_t	–	–	–	$n_\theta c_1$
for $i = 1, ..., N$ sample:	–	–	–	–
$\tilde{x}_{t,+}^{(i)} = f_t(\hat{x}_{t-1\|t-1}^{(i)}, (\hat{\theta}_{t-1\|t-1} + c_t\Delta_t)^{\mathrm{T}}\lambda(\hat{x}_{t-1\|t-1}^{(i)}), \omega_t^{(i)})$	–	–	–	Nn_x
$\tilde{x}_{t,-}^{(i)} = f_t(\hat{x}_{t-1\|t-1}^{(i)}, (\hat{\theta}_{t-1\|t-1} - c_t\Delta_t)^{\mathrm{T}}\lambda(\hat{x}_{t-1\|t-1}^{(i)}), \omega_t^{(i)})$	–	–	–	Nn_x
Evaluate:	–	–	–	$N(n_x + n_\theta)c_1$
$a_\theta(y_t, \tilde{x}_{t,+}^{(i)}, \hat{x}_{t-1\|t-1}^{(i)})$, and $a_\theta(y_t, \tilde{x}_{t,-}^{(i)}, \hat{x}_{t-1\|t-1}^{(i)})$	–	–	–	$2Nn_\theta$
Evaluate:	–	–	–	–
$\widehat{\nabla J}_{t,\mu}(\hat{\theta}_{t-1\|t-1}) = \frac{\hat{J}_t(\hat{\theta}_{t-1\|t-1}+c_t\Delta_t) - \hat{J}_t(\hat{\theta}_{t-1\|t-1}-c_t\Delta_t)}{2c_t\Delta_{t,\mu}}$	$n_\theta + 1$	$2n_\theta - 1$	–	–
where:	–	–	–	–
$\widehat{\nabla J}_{t,\mu}(\hat{\theta}_{t-1\|t-1} \pm c_t\Delta_t) = log\{\frac{1}{N}\sum_{i=1}^{N} a_\theta(y_t, \tilde{x}_{t,\pm}^{(i)}, \hat{x}_{t-1\|t-1}^{(i)})\}$	$2n_\theta$	$2Nn_\theta$	$2n_\theta$	–
Parameter Update:	–	–	–	–
$\hat{\theta}_{t\|t} = \hat{\theta}_{t-1\|t-1} + \gamma_t\widehat{\nabla J}_t(\hat{\theta}_{t-1\|t-1})$,	n_θ^2	$n_\theta + (n_\theta - 1)n_\theta$	–	–
$\widehat{\nabla J}_t(\hat{\theta}_{t-1\|t-1}) = [\widehat{\nabla J}_{t,1}(\hat{\theta}_{t-1\|t-1}), ..., \widehat{\nabla J}_{t,n_\theta}(\hat{\theta}_{t-1\|t-1})]$	–	–	–	–
for $i = 1, ..., N$ sample:	–	–	–	–
$\tilde{x}_t^{(i)} = f_t(\hat{x}_{t-1\|t-1}^{(i)}, \hat{\theta}_{t\|t}^{\mathrm{T}}\lambda(\hat{x}_{t-1\|t-1}^{(i)}), \omega_t^{(i)})$	–	–	Nn_x	–
Regularization to evaluate weights and resampling	–	–	–	$Nn_x(c_2 + c_3)$
Total	$n_x^3 + Nn_x^2 + n^2{}_\theta$ $+3n_\theta + 1$ $-$	$n_x^3 + (N-1)n_x^2 - Nn_x$ $+n_\theta^2 + (2N+2)n_\theta - 1$ $-$	$Nn_x + 2n_\theta$ $-$	$10n_x^3 + n_x^2$ $+n_x(2Nc_1 + 2N + Nc_2)$ $+n_\theta(c_1 + Nc_1 + 2N)$

of the system our proposed dual state and parameter estimation strategy can provide one with an accurate and reliable information on the health parameters of the system. This information can then be readily used to perform the tasks of fault detection, isolation and identification, following the presence or injection of faults in the components of the system. In the following subsection, the application of our proposed approach in previous subsections for addressing the fault diagnosis problem of nonlinear systems is investigated.

3.2.8 The Fault Diagnosis Formulation

Determination and diagnosis of drifts in unmeasurable parameters of a system require an on-line parameter estimation scheme. In parametric modeling of a system anomaly or drift, generally it is assumed that the parameters are either constant or dependent on only the system states [171].

Hence, drifts in the parameters must be estimated through estimation techniques.

In [172], various parameter estimation techniques such as least squares, instrumental variables and estimation via discrete-time models have been surveyed. The main drawbacks and limitation with such methods arise due to the complexity and nonlinearity of the systems that we are considering in this thesis that render the parameter estimation here a nonlinear optimization problem that must be solved in real-time. In [173] a nonlinear least squares (NLS) optimization scheme is developed for only the fault identification of a hybrid system.

Parameter estimation techniques that are used for fault diagnosis of system components generate residuals by comparing the estimated parameters that are obtained by either the ordinary least squares (OLS) or the recursive least squares (RLS) algorithms with parameters that are estimated under the initial fault free operation of the system [174].

The fault diagnosis problem under consideration in this thesis deals with obtaining an optimal estimate of the states as well as the time-varying parameters of a nonlinear system whose dynamics is governed by the discrete-time stochastic model,

$$x_{t+1} = f_t(x_t, \theta_t^{\mathrm{T}} \lambda(x_t), \omega_t), \tag{3.32}$$

$$y_t = h_t(x_t, \theta_t^{\mathrm{T}} \lambda(x_t)) + \nu_t, \tag{3.33}$$

where $f_t : \mathbb{R}^{n_x} \times \mathbb{R}^{n_\theta} \times \mathbb{R}^{n_\omega} \longrightarrow \mathbb{R}^{n_x}$ is a known nonlinear function, $\theta_t \in \mathbb{R}^{n_\theta}$ is the unknown and time-varying parameter vector that for a healthy system is set to $\mathbf{1}$, $\lambda_t : \mathbb{R}^{n_x} \longrightarrow \mathbb{R}^{n_\theta}$ is a differentiable function that determines the relationship between the system states and the health parameters. The function $h_t : \mathbb{R}^{n_x} \times \mathbb{R}^{n_\theta} \longrightarrow \mathbb{R}^{n_y}$ is a known nonlinear function, ω_t and ν_t are uncorrelated noise sequences with covariance matrices L_t and V_t, respectively. According to the

formulation used in equations (3.32) and (3.33), the parameter θ_t is a multiplicative fault vector whose value is considered to be set equal to $\mathbf{1}$ under the healthy mode of the system operation.

The model (3.32) and (3.33) is now used to investigate the problem of fault diagnosis (FD), which in this thesis is defined as the problem of fault detection, isolation, and identification (FDII) when the system health parameters are considered to be affected by an unknown and potentially time-varying multiplicative fault vector θ_t.

The system health parameters are known functions of the system states, $\lambda(x_t)$, and the multiplicative fault vector θ_t is to be estimated. In other words, the *a posteriori* estimated parameter $\hat{\theta}_{t|t}$ will be used to generate residual signals for accomplishing the fault diagnosis goal and objective. It is worth noting that based on our proposed formulation in (3.32) and (3.33) for capturing the variations in the system health parameters, the system health parameter itself is considered as a function of the system states whereas its variations are captured by introducing the fault vector. Therefore, the changes due to variations in the system states are not considered as faults and determination of the thresholds for fault diagnosis scheme is always based on the health system in which the fault vector is supposed to be equal to one.

The required residuals are obtained as the difference between the estimated parameters under the healthy operational mode that is denoted by $\hat{\theta}_0$, and the estimated parameters under the faulty operational mode of the system that is denoted by $\hat{\theta}_{t|t}$ as follows

$$r_t = \hat{\theta}_0 - \hat{\theta}_{t|t}. \tag{3.34}$$

It should be pointed out that the true value of the parameter is denoted by $\theta_t^\star$, which is assumed to be unknown.

For the implementation of our proposed fault diagnosis strategy that is constructed based on the previously developed state/parameter estimation framework, the parameter estimates will be considered as the main indicators for detecting, isolating, and identifying the faults in the system components. The residuals are generated from the parameter estimates under the healthy and faulty operational modes of the system according to equation (3.34). The estimation of the parameters under the healthy operational mode is determined according to,

$$\hat{\theta}_0 = \text{argmax}(-\log(\hat{p}(\theta_0|y_{1:T})), \tag{3.35}$$

where $\hat{p}(\theta_0|y_{1:T})$ denotes the probability density (conditioned on the observations up to time T associated with the healthy data), that is obtained from the collected estimates and fitted to a normal distribution. The time window T is chosen according to the convergence time of the parameter estimation algorithm. The thresholds to indicate the confidence intervals for each parameter are obtained through Monte Carlo analysis that is performed under different single-fault and multi-fault scenarios. The estimated parameters $\hat{\theta}_{t|t}$ are generated by following the procedure that was developed and proposed in previous subsections.

3.3 Fault Diagnosis of a Gas Turbine Engine

In this section, the utility of our proposed dual estimation framework when applied to the problem of fault diagnosis of a nonlinear model of a gas turbine engine is demonstrated and investigated. The performance of our proposed state/parameter estimation scheme will be evaluated and investigated when the gas turbine is subjected to deficiencies in its health parameters due to injection of simultaneous faults.

3.3.1 Model Overview

The mathematical model of a gas turbine as described in Chapter 2, is utilized in this chapter to evaluate the performance of the proposed dual state/parameter estimation method based on particle filtering approach to address the fault diagnosis problem in the single spool gas turbine engine. The continuous-time state space model of the gas turbine considering the formulation presented in equations (3.32), and (3.33), is given as follows,

$$\begin{aligned}
\dot{T}_{\rm CC} &= \frac{1}{c_{\rm v}\dot{m}_{\rm cc}}[(c_{\rm p}T_{\rm C}\theta_{m_{\rm C}}\dot{m}_{\rm C} + \eta_{\rm CC}H_{\rm u}\dot{m}_{\rm f} - c_{\rm p}T_{\rm CC}\theta_{m_{\rm T}}\dot{m}_{\rm T}) - c_{\rm v}T_{\rm CC}(\theta_{m_{\rm C}}\dot{m}_{\rm C} + \dot{m}_{\rm f} - \theta_{m_{\rm T}}\dot{m}_{\rm T})],\\
\dot{S} &= \frac{\eta_{\rm mech}\theta_{m_{\rm T}}\dot{m}_{\rm T}c_{\rm p}(T_{\rm CC} - T_{\rm T}) - \theta_{m_{\rm C}}\dot{m}_{\rm C}c_{\rm p}(T_{\rm C} - T_{\rm d})}{JS(\frac{\pi}{30})^2},\\
\dot{P}_{\rm CC} &= \frac{P_{\rm CC}}{T_{\rm CC}}\frac{1}{c_{\rm v}\dot{m}_{\rm cc}}[(c_{\rm p}T_{\rm C}\theta_{m_{\rm C}}\dot{m}_{\rm C} + \eta_{\rm CC}H_{\rm u}\dot{m}_{\rm f} - c_{\rm p}T_{\rm CC}\theta_{m_{\rm T}}\dot{m}_{\rm T}) - c_{\rm v}T_{\rm CC}(\theta_{m_{\rm C}}\dot{m}_{\rm C} + \dot{m}_{\rm f} - \theta_{m_{\rm T}}\dot{m}_{\rm T})]\\
&\quad + \frac{\gamma {\rm R}T_{\rm CC}}{V_{\rm CC}}(\theta_{m_{\rm C}}\dot{m}_{\rm C} + \dot{m}_{\rm f} - \theta_{m_{\rm T}}\dot{m}_{\rm T}),\\
\dot{P}_{\rm NLT} &= \frac{T_{\rm M}}{V_{\rm M}}(\theta_{m_{\rm T}}\dot{m}_{\rm T} + \frac{\beta}{\beta+1}\theta_{m_{\rm C}}\dot{m}_{\rm C} - \dot{m}_{\rm Nozzle}).
\end{aligned} \tag{3.36}$$

For the physical significance of the model parameters and details refer to Chapter 2. The five gas turbine measured outputs are also presented as

$$\begin{aligned}
y_1 &= T_{\rm C} = T_{\rm diffuser}[1 + \frac{1}{\theta_{\eta_{\rm C}}\eta_{\rm C}}[(\frac{P_{\rm CC}}{P_{\rm diffuzer}})^{\frac{\gamma-1}{\gamma}} - 1]],\\
y_2 &= P_{\rm CC},\ y_3 = S,\ y_4 = P_{\rm NLT},\\
y_5 &= T_{\rm CC}[1 - \theta_{\eta_{\rm T}}\eta_{\rm T}(1 - (\frac{P_{\rm NLT}}{P_{\rm CC}})^{\frac{\gamma-1}{\gamma}}].
\end{aligned} \tag{3.37}$$

In order to discretize the above model for implementation of our proposed dual state/parameter estimation particle filters, a simple Euler Backward method is applied with the sampling period of $T_s = 10$ msec.

The system health parameters are represented by the compressor and the turbine efficiency, η_{C} and η_{T}, respectively, and the compressor and turbine mass flow capacities, $\dot{m}_{\mathrm{C}}$ and $\dot{m}_{\mathrm{T}}$, respectively. A fault vector is incorporated in the above model to manifest the effects of system health parameters that are denoted by $\theta = [\theta_{\eta_{\mathrm{C}}}, \theta_{m_{\mathrm{C}}}, \theta_{\eta_{\mathrm{T}}}, \theta_{m_{\mathrm{T}}}]^{\mathrm{T}}$. By introducing a new parameter as $\acute{\theta}_{\eta_{\mathrm{C}}} = \frac{1}{\theta_{\eta_{\mathrm{C}}}}$, the measurement equations (3.37) can be represented as smooth functions with respect to the fault parameters. Each parameter variation can be a manifestation of changes in the fault vector that is considered as a multiplicative fault type. All the simulations that are conducted in this section corresponds to the cruise flight condition mode.

In order to demonstrate the effectiveness and capabilities of our proposed algorithms, we have also conducted simulation results corresponding to the conventional Bayesian method [2], and well-known recursive maximum likelihood (RML) parameter estimation method based on PF [3, 80, 81]. It should be noted that the number of particles in each algorithm is chosen based on the execution time of the algorithm such that approximately the same execution time is achieved for each algorithm. Moreover, the gradient free PF-based RML method [3] could not also yield an acceptable performance in this application given the large number of tuning parameters that are associated with each parameter in this method. Therefore, the RML based on the direct gradient method is utilized for the purpose of performance comparison.

In our schemes, the adaptive step size ($P_t^{(j)} = \gamma_t R_t^{(j)}$) is defined as the product of the constant $\gamma_t (\gamma_t = 0.9)$ with $R_t^{(j)}$, which is evaluated on-line from the trace of the prediction error covariance matrix that is estimated from the maximum likelihood method. On the other hand, the step size in the RML method was chosen as $\gamma_t = 0.05 = \mathrm{const.}$ that is obtained by trial and

error. The residuals corresponding to the parameter estimates are also obtained. Based on the percentage of the maximum absolute error criterion, a convergence time of 2 seconds is obtained in simulations for estimating both the states and parameters corresponding to 25 Monte Carlo runs of simultaneous faults with severities ranging from 1% to 10% loss of effectiveness of the healthy condition magnitudes.

To choose the number of particles for implementation of the state and parameter estimation filters, a quantitative study is conducted. Specifically, based on the mean absolute error (MAE$\%$) that was obtained at the steady state estimation process and by taking into account the algorithm's computational time, the number of particles is chosen as $N = 50$ for both the state and parameter estimation filters in this application. On the other hand, considering the average execution time of $18sec$ for one iteration of the dual estimation algorithm, the equivalent execution time is achieved for Bayesian algorithm with $\acute{N} = 45$ and in RML algorithm with $\acute{\acute{N}} = 150$ number of particles.

Subsequently, it was confirmed that acceptable performance and convergence times are obtained. The shrinkage matrix is also selected as $0.93I$. The initial distributions (i.e., the mean and covariance matrices) of the states and parameters are selected to correspond to the cruise flight operational condition as provided in [158]. In what follows, the two main simulation scenarios for conducting the fault diagnosis investigation of the gas turbine engine are presented.

Scenario I: Concurrent Faults in the Compressor and Turbine Health Parameters.

In this scenario, the input fuel flow rate to the engine is changed by decreasing it by 2% from its nominal value one second after reaching the steady state condition. Next, the effects of con-

current faults in both the compressor and the turbine are studied by injecting sequential fault patterns affecting the system components. Specifically, at time $t = 4$ sec the compressor efficiency is reduced by 5% (this represents the level of the fault severity), followed by at $t = 9$ sec the same fault type affecting the compressor mass flow capacity, and at $t = 14$ sec the same fault type affecting the turbine efficiency, and finally at $t = 19$ sec the same fault type is applied to the turbine mass flow capacity.

The results corresponding to changes in the fault parameters are depicted in Figure 3.2. The dotted lines depict the confidence bounds for residuals that are determined based on 50 Monte Carlo simulation runs under various concurrent and simultaneous single and/or multiple fault scenarios using the PE-based method. By analyzing the residuals, the detection time of a fault in each component and its severity can be determined and identified. It follows from this figure that the constructed residuals corresponding to the dual estimation method according to PE-based method and the RML method almost do not exceed their confidence bounds subject to changes in the engine input (applied at $t = 1$ sec). On the other hand, the Bayesian method shows false alarm for the residuals corresponding to the turbine heath parameters, and also this method is not able to track the changes in the fault vector in the selected time window in all the residual signals after fault occurrence.

In order to obtain a quantitative measure on the precision of our proposed estimation algorithm the results related to the 5% fault severity in terms of the mean absolute error (MAE$\%$) of estimates corresponding to the last 2 sec of simulations (following the algorithm convergence) after each change are provided and summarized in Table 3.8. The state/parameter estimation

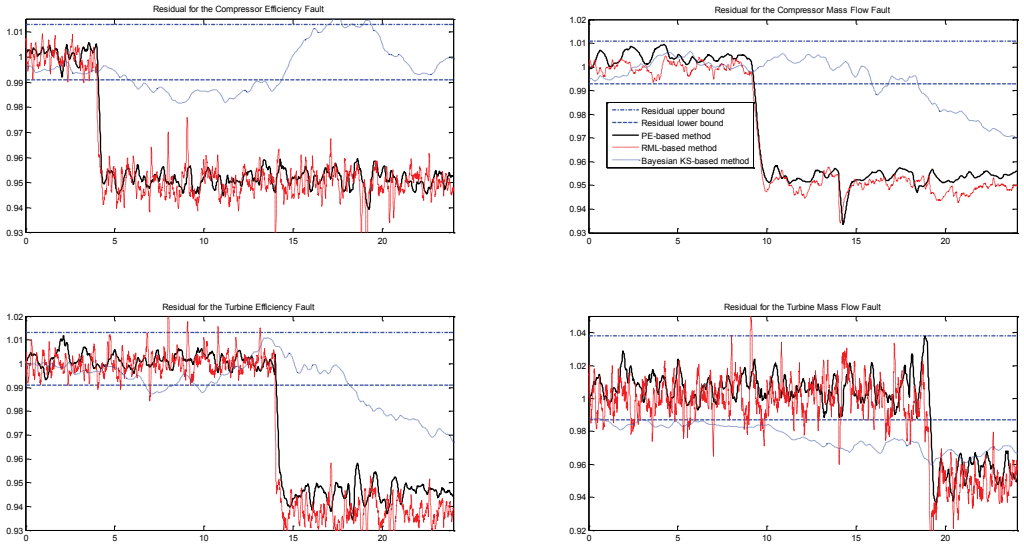


Figure 3.2: Residuals corresponding to the concurrent fault scenarios in the turbine and the compressor parameters.

MAE$\%$ for our proposed dual estimation algorithm according to PE-based method with $N = 50$, the conventional Bayesian method based kernel smoothing (KS-based) with $\acute{N} = 45$, and the RML method with $\acute{\acute{N}} = 150$ are presented. In this table, the i-th fault for $i = 1, ..., 4$ denotes the last 2 sec of simulations after the i-th fault occurrence, and the first column refers to the healthy system before the fault occurrence.

The results shown in Table 3.8 demonstrate that for the PE-based method the maximum MAE$\%$ for the states is between $0.03\% - 1.06\%$ of their nominal values. In case of the health parameters, for η_{C} and $\dot{m}_{\mathrm{C}}$ the maximum MAE$\%$ is around 0.91% and for η_{T} and $\dot{m}_{\mathrm{T}}$ it is around 0.98% of their nominal values. On the other hand, according to results presented in Table 3.8 (b), the maximum MAE$\%$ for the states corresponding to the RML method is between $0.1\% - 1.16\%$ of their nominal values. In case of the health parameters, the maximum MAE$\%$ ranges

between $0.8\% - 2.8\%$ of their nominal values, where both mass flow rates are estimated with higher MAE$\%$. The results corresponding to Table 3.8 (c) indicate that the maximum MAE$\%$ for the state estimation results in Bayesian KS-based method ranges between $0.2\% - 18.3\%$, for compressor health parameters between $0.53\% - 19.3\%$, and for turbine health parameters between $0.15\% - 3.0\%$

The MAE$\%$ for the estimated measurements (outputs) of the engine are also provided in Table 3.9 for the PE-based ,RML , and Bayesian KS-based methods. From the results presented in Table 3.9 (a) one can conclude that the maximum MAE$\%$ for the temperatures (of the turbine and the compressor) corresponding to our proposed PE-based method is less than 0.3%, and for the spool speed is less than 0.16%, and for the compressor pressure is less than 1.4%, and for the turbine pressure is less than 2.5%. On the other hand, the results presented in Table 3.9 (b) for the RML method show that the maximum MAE$\%$ for the compressor and turbine temperatures is less than 0.4% and 0.6%, respectively. For the spool speed the MAE$\%$ is less than 0.2%, and for the compressor pressure it is less than 1.5% and for the turbine pressure it is less than 2.5%. In the Bayesian KS-based method, instead of compressor temperature and spool speed outputs, the maximum MAE$\%$ exceeds 13% of the nominal values. Consequently, the results presented in these two tables confirm that the Bayesian KS-based method does not have acceptable estimation accuracy as compared to other two alternative methods. On the other hand, PE-based method outperforms the RML method significantly. The accuracy in the measurement estimation is an important aspect and factor given that from practical considerations the system states and parameters are unknown. Therefore, it is generally necessary to judge the estimation accuracy based on the output estimation error performance and behaviour.

In order to demonstrate and illustrate the precision of our proposed fault detection algorithm based on the dual state/parameter estimation scheme, at the end of this section a quantitative study is conducted by performing a confusion matrix analysis [175] in presence of various fault scenarios having different fault severities and in presence of the same level of process and measurement noise that are stated in [158] for the PE-based method, the RML method, and the Bayesian KS-based combined state and parameter estimation algorithm.

Scenario II: Simultaneous Faults in the Compressor and the Turbine Health Parameters.

In the second scenario, a simultaneous fault in all the 4 health parameters of the engine is applied at $t = 9$ sec. The compressor and the turbine efficiencies faults follow the pattern of a drift fault that starts at $t = 9$ sec and causes a 5% loss of effectiveness in the compressor efficiency by the end of the simulation time (i.e. at $t = 19$ sec), and a 3% loss of effectiveness in the turbine efficiency by the end of the simulation time. Simultaneously, the mass flow rate capacities of both the compressor and the turbine are affected by a fault that causes a 5% loss of effectiveness.

The residuals corresponding to the three previous estimation methods are provided in Figure 3.3. The simulations show that in case of changes in the engine input (applied at $t = 1$ sec) the RML method residuals has high false alarm rates as compared to dual estimation method according to PE-based algorithm, similar to the first scenario for the concurrent faults. More quantitative analysis on the performance of the RML method that is compared to the PE-based method is provided in the subsequent subsection. The presented results admit that the Bayesian KS-based method is not able to track the variations in the fault vectors in the case of simultaneous fault scenario.

Table 3.8: State/Parameter MAE$\%$ in case of concurrent fault scenarios for (a) Dual estimation algorithm according to PE-based method with $N = 50$ and (b) RML method with $N = 150$ (c) Bayesian KS-based method with $N = 45$.

(a)

State	No Fault	$1nd$ Fault	$2rd$ Fault	$3th$ Fault	$4th$ Fault
P_{CC}	0.3529	0.2097	0.3614	0.4336	0.2374
N	0.1473	0.0761	0.1087	0.1624	0.0296
T_{CC}	0.2683	0.1674	0.1678	0.3838	0.1155
P_{NLT}	0.8575	0.5325	0.3978	1.0614	0.3213
η_{C}	0.2702	0.1785	0.2749	0.3879	0.2107
$\dot{m}_{\mathrm{C}}$	0.6621	0.4229	0.3682	0.9132	0.2236
η_{T}	0.2865	0.1648	0.1743	0.4885	0.1873
$\dot{m}_{\mathrm{T}}$	0.4744	0.4557	0.4889	0.9757	0.5037

(b)

State	No Fault	$1nd$ Fault	$2rd$ Fault	$3th$ Fault	$4th$ Fault
P_{CC}	0.5352	0.6342	0.3921	0.8934	0.5882
N	0.0995	0.0912	0.1018	0.2060	0.1383
T_{CC}	0.2064	0.2443	0.2574	0.5174	0.3374
P_{NLT}	0.7181	0.8112	0.7771	1.1603	0.5666
η_{C}	0.9268	1.9195	2.1698	1.4508	1.4651
$\dot{m}_{\mathrm{C}}$	1.6338	1.8037	1.0761	2.6062	2.2717
η_{T}	0.9252	0.7876	0.8714	1.6517	1.1411
$\dot{m}_{\mathrm{T}}$	1.3719	1.7858	1.6162	1.9666	2.7653

(c)

State	No Fault	$1nd$ Fault	$2rd$ Fault	$3th$ Fault	$4th$ Fault
P_{CC}	1.8961	2.6032	6.1590	18.2816	7.9636
N	0.2127	0.5032	0.4490	4.7275	2.5564
T_{CC}	0.4789	1.0029	1.6025	8.6930	6.5029
P_{NLT}	0.6841	1.3838	3.6558	14.9288	8.9250
η_{C}	0.7248	3.3660	3.0788	6.3141	4.9584
$\dot{m}_{\mathrm{C}}$	0.5306	1.3399	4.3086	19.3026	11.8476
η_{T}	0.1445	0.7394	0.9198	1.0082	1.8379
$\dot{m}_{\mathrm{T}}$	1.4943	1.6979	2.6633	3.0198	1.7125

Table 3.9: Output estimation MAE$\%$ in case of concurrent fault scenarios for (a) Dual estimation algorithm according to PE-based method with $N = 50$ and (b) RML method with $N = 150$ (c) Bayesian KS-based method with $N = 45$.

(a)

Output	No Fault	$1nd$ Fault	$2rd$ Fault	$3th$ Fault	$4th$ Fault
T_{C}	0.2893	0.2319	0.2749	0.2805	0.2357
P_{C}	1.3548	1.2332	1.2507	1.4070	1.1813
N	0.1473	0.0761	0.1087	0.1624	0.0296
T_{T}	0.2034	0.1857	0.1911	0.2804	0.1322
P_{T}	2.2231	2.1696	2.1839	2.4577	2.0783

(b)

Output	No Fault	$1nd$ Fault	$2rd$ Fault	$3th$ Fault	$4th$ Fault
T_{C}	0.2902	0.3240	0.2956	0.3985	0.2991
P_{C}	1.4012	1.3755	1.3030	1.4779	1.2902
N	0.0995	0.0912	0.1018	0.2060	0.1383
T_{T}	0.2181	0.2461	0.2122	0.5786	0.5206
P_{T}	2.3446	2.3994	2.1356	2.5220	2.2474

(c)

Output	No Fault	$1nd$ Fault	$2rd$ Fault	$3th$ Fault	$4th$ Fault
T_{C}	0.9416	0.7569	0.6033	3.4434	0.3789
P_{C}	1.9784	2.6211	6.5484	18.0905	7.8644
N	0.2127	0.5032	0.4490	4.7275	2.5564
T_{T}	0.3428	0.7125	2.6660	13.3328	8.6219
P_{T}	2.2715	2.7023	3.9467	14.9107	8.7964

The results in Table 3.10 (a) show that for our proposed PE-based method, the maximum MAE$\%$ for both state and parameter estimates are between $0.1\% - 0.5\%$ of their nominal values. However, in the worst case the post fault estimated MAE$\%$ of the $\dot{m}_{\mathrm{T}}$ is 0.47% of its nominal value. Moreover, in the results shown for the RML method in Table 3.10 (b) it follows clearly that the state estimation MAE$\%$ can be achieved within $0.1\% - 0.8\%$ of the nominal values, whereas the parameter estimation MAE$\%$ is achieved within $0.7\% - 3\%$ of the nominal values with higher error rates after the fault occurrence, specially in the compressor and turbine mass flow rate capacities. However, for the Bayesian KS-based method in Table 3.10 (c) the maximum MAE$\%$ is achieved within $0.19\% - 8.4\%$ of the nominal values for the estimated states and within $0.25\% - 7.2\%$ of the nominal values for the estimated parameters.

The MAE$\%$ measurement (output) estimate error given in Table 3.11 (a) for the PE-based method shows that after simultaneous fault occurrences the error increases when compared to their values before the fault occurrences. This is caused due to accumulation of parameter estimation errors while all the four parameters are affected by a fault. On the other hand, the results corresponding to the output estimation as given in Tables 3.11(a)-(c) show that with the exception of the turbine pressure, our PE-based method outperforms the RML method for estimating the other four measurement outputs. However, the maximum MAE$\%$ for the outputs from Bayesian KS-based method performs high level of errors after fault occurrence in all measurement outputs as compared to the other two estimation methods.

To summarize, our proposed PE-based fault diagnosis algorithm is capable of detecting, isolating and estimating the component faults of a gas turbine engine with an average accuracy of

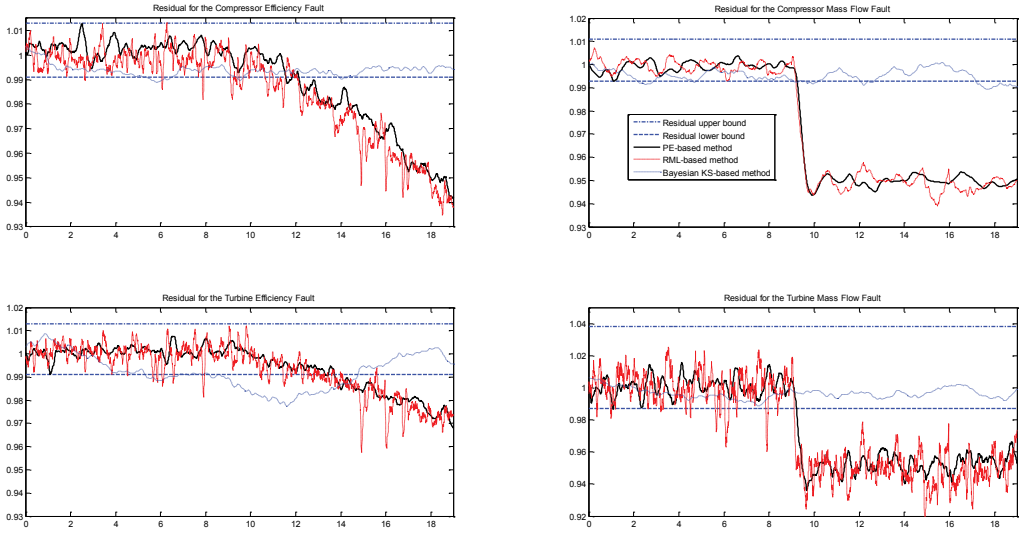


Figure 3.3: Residuals corresponding to the simultaneous fault scenarios.

0.3% for the compressor and 0.5% for the turbine faults. In contrast the RML algorithm is capable of achieving the performance of an average 3% for the compressor and 1.6% for the turbine faults. The Bayesian KS-based method does not have acceptable accuracy for simultaneous fault diagnosis application.

Fault Diagnosis Confusion Matrix Analysis

Finally, in this subsection a quantitative study is performed by utilizing the confusion matrix analysis [175] to evaluate the increase in the false alarms and/or misclassification rates of the faults in our considered application when the fault diagnosis algorithm is implemented by our proposed PE-based method with $N = 50$ particles, the RML method with $N = 150$, and the Bayesian KS-based method with $N = 45$ particles. The thresholds corresponding to each algorithm are determined from 25 Monte Carlo runs on simultaneous fault scenarios that are not

Table 3.10: State/Parameter MAE% in case of simultaneous fault scenarios for (a) PE-based method with $N = 50$ and (b) RML method with $N = 150$ (c) Bayesian KS-based method with $N = 45$.

(a)

State	Before Fault	After Fault
P_{CC}	0.2217	0.2372
N	0.0535	0.1061
T_{CC}	0.2086	0.1928
P_{NLT}	0.3970	0.4291
η_{C}	0.1735	0.1821
$\dot{m}_{\mathrm{C}}$	0.2811	0.3293
η_{T}	0.1016	0.1485
$\dot{m}_{\mathrm{T}}$	0.4589	0.4744

(b)

State	Before Fault	After Fault
P_{CC}	0.4865	0.4383
N	0.1025	0.1053
T_{CC}	0.2247	0.1888
P_{NLT}	0.7540	0.5264
η_{C}	0.9295	1.7956
$\dot{m}_{\mathrm{C}}$	1.7291	2.9729
η_{T}	0.7306	1.0040
$\dot{m}_{\mathrm{T}}$	1.4290	1.5923

(c)

State	Before Fault	After Fault
P_{CC}	0.5186	8.3590
N	0.1906	2.7197
T_{CC}	0.2834	4.5219
P_{NLT}	0.6506	5.0157
η_{C}	0.5992	4.8358
$\dot{m}_{\mathrm{C}}$	0.5775	7.2215
η_{T}	0.2463	4.8958
$\dot{m}_{\mathrm{T}}$	0.2861	4.3080

Table 3.11: Output estimation MAE% in case of simultaneous fault scenarios for (a) PE-based method with $N = 50$ and (b) RML method with $N = 150$ (c) Bayesian KS-based method with $N = 45$.

(a)

Output	Before Fault	After Fault
T_{C}	0.2207	0.2852
P_{C}	1.2926	1.3729
N	0.0535	0.1027
T_{T}	0.1565	0.1650
P_{T}	2.1210	2.2348

(b)

Output	Before Fault	After Fault
T_{C}	0.2482	0.2832
P_{C}	1.4051	1.4369
N	0.1025	0.1053
T_{T}	0.2058	0.1774
P_{T}	2.1413	2.1580

(c)

Output	Before Fault	After Fault
T_{C}	0.3250	3.5756
P_{C}	1.3621	8.1037
N	0.1906	2.7197
T_{T}	0.4126	4.9311
P_{T}	2.2252	5.0404

necessarily the same for the three algorithms. The confusion matrix data is obtained by performing simulations for another 35 Monte Carlo simultaneous fault scenarios having different fault severities and in presence of the same process and measurement noise covariances corresponding to 50% of the nominal values of the process and measurement noise covariances (according to [158]). In these scenarios, at each time more than one of the system health parameters are affected by component faults.

The results are shown in Tables 3.12(c)-3.12(a) corresponding to PE-based method with $N = 50$ particles, the RML method with $N = 150$, the RML method with $N = 150$, and the Bayesian KS-based method with $N = 45$ particles, respectively. In these tables the rows depict the actual number of fault categories that are applied and the columns represent the number of estimated fault categories. The diagonal elements represent the true positive rate (TP) for each

Table 3.12: Confusion matrix for (a) PE-based method with $N = 50$ and (b) RML method with $N = 150$ (c) Bayesian KS-based method with $N = 45$.

(a)

Fault	$\dot{\eta}_{\mathrm{C}}$	$\dot{m}_{\mathrm{C}}$	$\dot{\eta}_{\mathrm{T}}$	$\dot{m}_{\mathrm{T}}$	No Fault
η_{C}	**31**	0	2	2	0
$\dot{m}_{\mathrm{C}}$	0	**30**	2	3	0
η_{T}	1	1	**28**	4	1
$\dot{m}_{\mathrm{T}}$	1	1	3	**29**	1
No Fault	0	0	1	1	**33**

(b)

Fault	$\dot{\eta}_{\mathrm{C}}$	$\dot{m}_{\mathrm{C}}$	$\dot{\eta}_{\mathrm{T}}$	$\dot{m}_{\mathrm{T}}$	No Fault
η_{C}	**28**	2	3	2	0
$\dot{m}_{\mathrm{C}}$	1	**27**	1	4	2
η_{T}	2	3	**26**	3	1
$\dot{m}_{\mathrm{T}}$	1	3	4	**26**	1
No Fault	0	2	1	1	**31**

(c)

Fault	$\dot{\eta}_{\mathrm{C}}$	$\dot{m}_{\mathrm{C}}$	$\dot{\eta}_{\mathrm{T}}$	$\dot{m}_{\mathrm{T}}$	No Fault
η_{C}	**10**	5	6	4	**10**
$\dot{m}_{\mathrm{C}}$	9	**13**	8	6	9
η_{T}	6	6	**9**	7	7
$\dot{m}_{\mathrm{T}}$	5	7	8	**11**	4
No Fault	**10**	9	7	4	5

fault occurrence. The accuracy (AC), precision (P), and the false positive rate (FP) of the three algorithms are also evaluated from the confusion matrix results according to the following formulae [175],

$$AC = \frac{\sum_{j=1}^{5} c_{jj}}{\sum_{i=1}^{5}\sum_{j=1}^{5} c_{ij}},\ P_j = \frac{c_{jj}}{\sum_{i=1}^{5} c_{ij}},\ FP = \frac{\sum_{j=1}^{4} c_{5j}}{\sum_{j=1}^{5} c_{5j}},$$

where $c_{ij},\ i, j = 1, ..., 5$ denote the elements of the confusion matrix. In Table 3.13, the confusion matrix results according to the above metrics for the Tables 3.12(a)-3.12(c) are provided. The results demonstrate that the accuracy of the fault diagnosis for the dual PE-based estimation algorithm outperforms RML method with 7.43% and the false positive rate of 5.71% less than RML method. The precision of the algorithm for all the system four health parameters

is more than the ones from RML method. However, the Bayesian KS-based method indicates poor accuracy and high false alarm rate for the fault diagnosis of the system. Consequently, the PE-based method with $N = 50$ outperforms the other two methods significantly in terms of higher accuracy, lower false positive rate, and higher precision for all the four health parameters of the gas turbine engine.

Table 3.13: Confusion matrix Analysis results.

Noise Level	$AC\%$	$FP\%$	$P_{\eta_{\mathrm{C}}}\%$	$P_{\dot{m}_{\mathrm{C}}}\%$	$P_{\eta_{\mathrm{T}}}\%$	$P_{\dot{m}_{\mathrm{T}}}\%$
PE-based Method with 50 Particles	86.29	5.71	93.94	93.75	77.78	74.36
RML Method with 150 Particles	78.86	11.43	87.50	72.97	74.29	72.22
Baysian KS-based Method with 45 Particles	25.95	85.71	25.00	32.50	23.68	34.38

3.4 Conclusion

In this chapter, a novel dual estimation filtering scheme is proposed and developed based on particle filters (PF) to estimate a nonlinear stochastic system states and time variations in its parameters. The dual structure is based on the extension of the Bayesian parameter estimation framework. A dual structure is proposed for achieving simultaneous state and parameter estimation objectives. Performance results of the application of our method to a gas turbine engine under healthy and faulty scenarios are provided to demonstrate and illustrate the superior capability and performance of our scheme for a challenging fault diagnostic application as compared to the well-known recursive maximum likelihood (RML) method based on particle filters and conventional Bayesian method for combined state and parameter estimation based on particle filters while the computational complexity of all the algorithms remains the same. On the other

hand, the false alarm rate of our proposed dual algorithm is significantly lower than the RML and conventional Bayesian methods. These two main characteristics justify and substantiate the observation that our proposed algorithm is more suitable for the purpose of fault diagnosis of critical nonlinear systems that require lower fault detection times and false alarm rates. Moreover, the estimation results accuracy in terms of the fault identification are also provided. The obtained results are demonstrated and validated by performing a confusion matrix analysis.

Chapter 4

An Improved Particle Filtering Based Approach for Health Prediction and Prognosis of Nonlinear Systems

In this chapter the previously developed dual state and parameter estimation algorithm based on particle filters, as presented in Chapter 3, is extended for long-term prediction of nonlinear systems states and health parameters. In our proposed approach, an observation forecasting scheme is developed to extend the system observation profiles (as time-series) to future. Particles are then propagated to future time instants according to a resampling algorithm instead of considering constant weights for the particles propagation in the prediction step. The uncertainty in the long-term prediction of the system states and parameters are managed by utilizing

dynamic linear models for development of an observation forecasting scheme. This task is addressed through an outer adjustment loop for adaptively changing the sliding observation injection window based on the Mahalanobis distance criterion. Our proposed approach is then applied to predict the health condition of a gas turbine engine that is affected by degradations in the system health parameters for demonstrating and illustrating the capabilities and performance characteristics of developed schemes.

In our proposed method, it is shown that the previously developed dual state and parameter estimation algorithm in Chapter 3 along with the newly developed DLM-based prediction method where the particles resampling is maintained for future time instants can yield improved long-term prediction performance and achieve more accurate RUL estimation of the system. These constitute as the main goals of the second component of any PHM strategy [38], [176–179]. The above results are obtained by evaluating the percent of root mean square error criterion and its effect on the accuracy of the RUL prediction for the prognosis problem [180]. The online performance of the developed prognosis approach is evaluated using prognosis metrics as introduced in [181, 182] in comparison with other well-known methods in the literature [25]. Furthermore, we have conducted an extensive study on the computational complexity of our proposed method in terms of the number of flop operations. Moreover, the conditions under which the equivalent time complexity of our proposed method including the observation prediction and the resampling scheme for the prediction horizon can be comparable to the previously developed method in [25] with augmented state and parameter vector and constant particle weight propagation, are obtained.

Dynamic linear models (DLM) represent flexible approaches for modeling a variety of fixed, time-varying, univariate or multivariate systems using Bayesian analysis [183]. The key feature in working with the DLM models is that their estimation and prediction schemes can be obtained recursively by using Bayesian approaches. One can generate useful models for forecasting non-stationary observations where the implementation and analysis are not as complicated as nonlinear time-series analytic methods used for forecasting [77], [183]. In the present work, DLM models and particle filters are integrated into a module for managing uncertainty in the long-term prediction of the system health condition. This is achieved by introducing fixed-lag DLM models that are updated according to an adaptive scheme. This adaptive updating scheme is developed based on the Mahalanobis distance metric that enables the prediction algorithm manage uncertainties originated from the non-Gaussian process noise. Mahalanobis distance is an important metric that has been used for fault detection of dynamical systems [184].

Finally, our proposed prediction strategy is applied to a gas turbine engine application to predict the system health parameters variations when it is subjected to soft degradation damages. Based on the predicted health parameters, the remaining useful life of the engine is determined. In this thesis, we concentrate and mainly investigate the effects of the fouling and the erosion phenomena as the main causes of the engine performance degradation. The probabilities of an engine failure due to these degradation phenomena are approximated based on the developed health prediction scheme.

The main contribution of this chapter is now summarized as below:

1. Extend the developed dual estimation method to predict the future health of the affected

nonlinear system. This methodology is developed by incorporating the dynamical linear models (DLM) for Bayesian forecasting of uni-variate time-series in an observation forecasting module which is enhanced to the particle filtering-based dual estimation method.

The remainder of this chapter is organized as follows. In Section 4.1, the nonlinear filtering problem is formulated to include changes in the health parameters of the system through a multiplicative fault vector. A brief background information related to our previously developed dual state and parameter nonlinear filtering schemes based on the particle filter method is also presented in this section. Our proposed framework for predicting the future propagation of the nonlinear system states and parameters is developed in Section 4.2. Section 4.3 provides the complexity analysis of our proposed prediction scheme based on the equivalent flop complexity analysis. The utilization and implementation of our proposed prediction method in evaluating the system remaining useful life (RUL) is presented in Section 4.4. Finally, simulation results corresponding to the application of our proposed method for failure prognosis of a gas turbine engine that is affected by degradations due to compressor fouling and turbine erosion are provided in Section 4.5.

4.1 Problem Statement and Backgroung Information

In model-based prognosis and health monitoring approaches, the first module in the problem statement is characterizing the damage model. Consider the following nonlinear dynamical

system,

$$x_{t+1} = f_t(x_t, \theta_t^{\mathrm{T}} \lambda(x_t), \omega_t), \tag{4.1}$$

$$y_t = h_t(x_t, \theta_t^{\mathrm{T}} \lambda(x_t)) + \nu_t, \tag{4.2}$$

where $x_t \in \mathbb{R}^{n_x}$ is the system state, $\lambda(.) : \mathbb{R}^{n_x} \longrightarrow \mathbb{R}^{n_\theta}$ is a known differentiable function that determines the relationship between the health parameters and the system states, $\theta_t \in \mathbb{R}^{n_\theta}$ is an unknown and possibly time-varying multiplicative fault vector that represents the damage effect on the system health parameter where for a healthy system, θ_t is set to $\mathbf{1}$, $y_t \in \mathbb{R}^{n_y}$ is the output measurement, ω_t and ν_t are uncorrelated noise sequences with covariance matrices W_t and V_t, respectively, $f_t : \mathbb{R}^{n_x} \times \mathbb{R}^{n_\theta} \times \mathbb{R}^{n_\omega} \longrightarrow \mathbb{R}^{n_x}$, and $h_t : \mathbb{R}^{n_x} \times \mathbb{R}^{n_\theta} \longrightarrow \mathbb{R}^{n_y}$ are known nonlinear functions representing the relationship between the states, parameters and the output measurements (observations). For example, the degradation phenomenon in mechanical systems can be identified from the changes it causes on the efficiency of the system, where the efficiency is designated as the health parameter that can be analytically obtained from the states and measurements of the system. The process noise is not considered as an additive noise since corresponding to our main focused application (that is, mechanical systems application) the additive process noise assumption is not necessarily valid.

The main objective of this thesis is to develop a novel framework for performing system failure prognosis according to the following two principle modules [25], namely: (a) joint state and parameter estimation (health tracking), and (b) prediction of the state and parameter distribution (health prediction). The first module, that is the joint state and parameter estimation, has already been developed in our work presented in Chapter 3 [4], where the particle filtering (PF) method was used to develop a novel dual state and parameter estimation scheme that can be utilized for

health tracking problems. In this chapter, which represents an extension to our work in [185], additional theoretical and simulation results are developed and provided corresponding to the second module of the proposed prognosis approach. The developed method can be utilized for predicting the propagation of the system states and changes in the system health parameters in the long-term horizon and its effects on the accuracy of the system remaining useful life prediction. In the subsequent sections, more details are explained regarding the development of our proposed methodology.

In Table 4.1, the summary corresponding to our previously developed dual state and parameter estimation filter for the nonlinear system (4.1)-(4.2) is provided. More details regarding this algorithm can be found in Chapter 3.

We are now in the position to present our proposed prediction framework based on particle filters by utilizing the dynamic linear models (DLM) as local models for obtaining and developing observation profiles forecasting.

Table 4.1: Summary of our developed dual state and parameter estimation algorithm [4, 5]

- Initialization of the states and parameters: $x_0 \sim \pi_{x_0}$ and $\theta_0 \sim \pi_{\theta_0}$, where π_{x_0} and π_{θ_0} are the initial distributions of states and parameters, respectively,

$[\hat{x}_{t|t}, \{\hat{x}_{t|t}^{(i)}\}_{i=1}^{N}, \hat{\theta}_{t|t}, \{\hat{\theta}_{t|t}^{(j)}\}_{j=1}^{N}, \Sigma_{\hat{\theta}_{t|t}}] = \text{DualPF}(\{\hat{x}_{t-1|t-1}^{(i)}\}_{i=1}^{N}, \{\hat{\theta}_{t-1|t-1}^{(j)}\}_{j=1}^{N}, \Sigma_{\hat{\theta}_{t-1|t-1}}, y_t)$

1. Estimation of the *a priori* state distribution:
   (a) Generate the *a priori* state particles: $\hat{x}_{t|t-1}^{(i)} = f_t(\hat{x}_{t-1|t-1}^{(i)}, \hat{\theta}_{t-1|t-1}^{\mathrm{T}}\lambda(\hat{x}_{t-1|t-1}^{(i)}), \omega_t^{(i)}),\ i = 1, ..., N$
   (b) *A priori* state covariance approximation: $\Sigma_{\hat{x}_{t|t-1}} = \frac{1}{N-1}\sum_{i=1}^{N}(\hat{x}_{t|t-1}^{(i)} - \frac{1}{N}\sum_{j=1}^{N}\hat{x}_{t|t-1}^{(j)})(\hat{x}_{t|t-1}^{(i)} - \frac{1}{N}\sum_{j=1}^{N}\hat{x}_{t|t-1}^{(j)})^{\mathrm{T}}$.
2. Estimation of the *a posteriori* state distribution:
   (a) Calculate the state particles weights: $\tilde{w}_{x_t}^{(i)} \triangleq \frac{\rho(y_t|\hat{x}_{t|t-1}^{(i)}, \hat{\theta}_{t-1|t-1})}{\sum_{i=1}^{N}\rho(y_t|\hat{x}_{t|t-1}^{(i)}, \hat{\theta}_{t-1|t-1})},\ i = 1, ..., N$ where $\rho(y_t|.)$ is the conditional probability density function of y_t,
   (b) Regularization of the *a priori* state distribution: $\tilde{\pi}_{x_{t|t}}^{N}(\mathrm{d}x_t) \approx \sum_{l=1}^{N_{\text{reg}}}\sum_{i=1}^{N}\tilde{w}_{x_t}^{(i)}\frac{|\mathbb{A}_t^{-1}|}{b^{n_x}}\mathcal{K}(\frac{1}{b}\mathbb{A}_t^{-1}(x_t^{\text{reg}_l} - \hat{x}_{t|t-1}^{(i)})),\ \mathbb{A}_t\mathbb{A}_t^{\mathrm{T}} = \Sigma_{\hat{x}_{t|t-1}}$,
   (c) Resampling to approximate *a posteriori* state estimate distribution: $\pi_{x_{t|t}}^{N}(\mathrm{d}x_t) = \frac{1}{N}\sum_{i=1}^{N}\delta_{\hat{x}_{t|t}^{(i)}}(\mathrm{d}x_t)$,
   (d) Estimate the *a posteriori* state: $\hat{x}_{t|t} = \frac{1}{N}\sum_{i=1}^{N}\hat{x}_{t|t}^{(i)}$.
3. Estimation of the *a priori* parameter distribution:
   (a) Calculate the prediction error as: $\epsilon_t(\hat{\theta}_{t-1|t-1}^{(j)}) = y_t - h(\hat{x}_{t|t}, \hat{\theta}_{t-1|t-1}^{(j)^{\mathrm{T}}}\lambda(\hat{x}_{t|t})),\ j = 1, ..., N$,
   (b) Calculate $m_t^{(j)}$ from the PE-based modified artificial law: $m_t^{(j)} = \hat{\theta}_{t-1|t-1}^{(j)} + \gamma_t R_t^{(j)}\psi_t^{(j)}\epsilon_t(\hat{\theta}_{t-1|t-1}^{(j)}),\ j = 1, ..., N$,
   (c) Apply the kernel smoothing concept through the shrinkage matrix A to obtain the first step *a posteriori* parameter estimation distribution: $\tilde{\theta}_{t|t}^{(j)} = Am_t^{(j)} + (I - A)\bar{m}_{t-1} + \zeta_t^{(j)},\ \bar{m}_{t-1} = \frac{1}{N}\sum_{j=1}^{N}\hat{\theta}_{t-1|t-1}^{(j)},\ \zeta_t^{(j)} \sim \mathcal{N}(0, (I - A^2)\Sigma_{\hat{\theta}_{t-1|t-1}})$,
   (d) Calculate the predicted output from the parameter estimation filter: $\bar{y}_{t|t-1}^{(j)} = h_t(\hat{x}_{t|t}, \hat{\theta}_{t|t-1}^{(j)^{\mathrm{T}}}\lambda(\hat{x}_{t|t})),\ j = 1, ..., N$.
4. Estimation of the *a posteriori* parameter distribution:
   (a) Calculate the parameter particles weights: $\tilde{w}_{\theta_t}^{(j)} \triangleq \frac{\rho(y_t|\hat{x}_{t|t}, \tilde{\theta}_{t|t}^{(j)})}{\sum_{j=1}^{N}\rho(y_t|\hat{x}_{t|t}, \tilde{\theta}_{t|t}^{(j)})},\ j = 1, ..., N$,
   (b) Calculate the first step *a posteriori* parameter estimation distribution: $\tilde{\pi}_{\theta_{t|t}}^{N}(\mathrm{d}\theta_t) \approx \sum_{j=1}^{N}\tilde{w}_{\theta_t}^{(j)}\delta_{\tilde{\theta}_{t|t}^{(j)}}(\mathrm{d}\theta_t)$,
   (c) Resampling to approximate *a posteriori* parameter estimate distribution: $\pi_{\theta_{t|t}}^{N}(\mathrm{d}\theta_t) = \frac{1}{N}\sum_{j=1}^{N}\delta_{\tilde{\theta}_{t|t}^{(j)}}(\mathrm{d}\theta_t)$,
   (d) Obtain the *a posteriori* parameter estimate: $\hat{\theta}_{t|t} = \frac{1}{N}\sum_{j=1}^{N}\hat{\theta}_{t|t}^{(j)}$,
   (e) Calculate the *a posteriori* parameter estimate covariance: $\Sigma_{\hat{\theta}_{t|t}} = \frac{1}{N-1}\sum_{j=1}^{N}(\hat{\theta}_{t|t}^{(j)} - \hat{\theta}_{t|t})(\hat{\theta}_{t|t}^{(j)} - \hat{\theta}_{t|t})^{\mathrm{T}}$.

The definition of the notations in the above algorithm is as follows:

- $\omega_t^{(i)}$ denotes the process noise added to each particle for $i = 1, ..., N$,
- $\mathcal{K}(.)$: The regularization kernel,
- $\mathbb{A}_t$ is chosen such that $\mathbb{A}_t\mathbb{A}_t^{\mathrm{T}} = \Sigma_{\hat{x}_{t|t-1}}$,
- $x_t^{\text{reg}_l}$: The regularized points around which $\mathcal{K}(.)$ is evaluated for $l = 1, ..., N_{\text{reg}}$(number of regularized steps),
- $\gamma_t R_t^{(j)}$: The adaptive step size in the parameter estimation where $R_t^{(j)} = \sqrt{\text{trace}(\mathcal{E}_t^{(j)}\mathcal{E}_t^{(j)^{\mathrm{T}}})}$, $\mathcal{E}_t^{(j)} = \epsilon_t(\hat{\theta}_{t-1|t-1}^{(j)}) - \frac{1}{n_y}\sum_{l=1}^{l=n_y}\epsilon_t^{(l)}(\hat{\theta}_{t-1|t-1}^{(j)})$, where $\epsilon_t^{(l)}(\hat{\theta}_{t-1|t-1}^{(j)})$ denotes the l-th element of the $\epsilon_t(\hat{\theta}_{t-1|t-1}^{(j)})$ vector, and γ_t is a constant or decreasing step size,
- $\psi_t^{(j)}$: The Jacobian of $h(\hat{x}_{t|t}, \hat{\theta}_{t-1|t-1}^{(j)})$ with respect to $\hat{\theta}_{t-1|t-1}^{(j)}$ evaluated for j-th particle,
- $\zeta_t^{(j)}$: The evolution noise added to each parameter particle.
- A: The shrinkage matrix and chosen as, $A \leq I(1 - \sqrt{\frac{\sigma_{\min}(P_{\max}^2\Psi V_y\Psi^{\mathrm{T}}V_{\hat{\theta}}^{-1})}{\sigma_{\max}(P_{\max}^2\Psi V_y\Psi^{\mathrm{T}}V_{\hat{\theta}}^{-1})}})$, where $\psi_t^{(j)}$ is considered as constant between the time steps t and $t-1$ and is denoted by Ψ, W denotes the upper-bound on the variance of the added noise W_t, V_y denotes the upper-bound on the variance of the measurement noise R_t, $V_{\hat{\theta}}$ denotes the variance of the parameters when they are constants that can be assumed the same as the initial covariance of the parameters, and $P_{\max} = \gamma_0\sqrt{\text{trace}(\mathcal{E}_{\max}\mathcal{E}_{\max}{}^{\mathrm{T}})}$, with γ_0 denotes the initial value of the step size, and $\mathcal{E}_{\max}\mathcal{E}_{\max}{}^{\mathrm{T}}$ is a design parameter denotes the maximum acceptable variance among the prediction error vector elements.

4.2 Prediction Framework

In this section, the second module in our proposed model-based prognosis approach (system health prediction), as described in Section 4.1, is developed and presented. This is accomplished by extending our previously developed PF-based dual state and parameter estimation scheme to the future time instants, where the weight update in the long-term prediction with particle filters cannot be easily implemented in the absence of future observations.

Our proposed strategy is to first forecast the system observations from the available historical data for a predefined time horizon where the observation forecasting algorithm is adaptively adjusted whenever a new observation batch becomes available. However, the nonlinearity of the system and non-normality of the measurement noise coupled with the degradations effect lead to a non-stationary behavior in the system observations [186]. Hence, to model the observations of a dynamical system as a time-series, instead of nonlinear models, dynamical linear models known as the DLM are utilized that are constructed based on the assumptions of local normality and linearity of the time-series in each short-term time interval in which the observation time-series manifest a stationary behavior [186].

The DLM model essentially represents a special class of state space linear and Gaussian models, in which the time-series is considered as the output of a dynamical system that is perturbed by random disturbances. In our proposed fixed-lag DLM model, in each time window the available observation history that originates from a stochastic non-stationary process is approximated by a stationary process based on a linear regression method, and the observation

forecasting is performed for a specific time horizon window.

By augmenting and integrating the observation forecasting module with the PF-based estimation scheme, the PF algorithm can be extended to future time steps by utilizing the same weight update rule (through performing a resampling). This is accomplished in the same manner as the estimation module to predict the system state and parameters for the long-term horizon according to the previously described estimation algorithm (Section 3.2).

The DLM structure for observation forecasting is constructed according to the well-known autoregressive moving average (ARMA) models. In the next subsection, a brief overview to forecasting with ARMA models within the DLM formulation is provided.

4.2.1 Forecasting with ARMA Models

ARMA models are well-known as suitable modeling strategies for forecasting or predicting the value of a stationary zero-mean stochastic process. Although, the observation process y_t in (4.2) is non-stationary, one can still approximate it as an ARMA process using locally dynamic linear models (DLM) in a short-term time window. It is assumed that the variations in the observation time-series are not very fast during this time window, therefore the assumption of the stationarity remains valid. This assumption is indeed not going to be restrictive for degradation forecasting in dynamical systems, since the degradations affect the system dynamical behavior quite slowly in time. Next, we analyze ARMA models in our developed fixed-lag DLM framework.

Towards this goal, let us denote $y_{j,t}$, $j = 1, ..., n_y$ as the j-th element of the output mea-

surement vector y_t (implying a univariate time-series). Since y_t and consequently $y_{j,t}$ are non-stationary process, to model each $y_{j,t}$ as a stationary time-series one has to consider time intervals during which the process behavior can be approximated as a stationary process. It is assumed that $y_{j,t}$ is a stationary process between the time instants t_l and t_{l+1}, where $l \in \mathbb{N}$ and the τ number of available data in the time window $(t_l - \tau,\ t_l]$ are used for constructing the l-th DLM model related to each output in the interval $[t_l,\ t_{l+1})$. Later, it will be shown how the time instants $t_l,\ l \in \mathbb{N}$ are specified based on the Mahalanobis distance criterion [187].

The DLM models in the observation forecasting module are constructed according to the ARMA process model. Consequently, corresponding to each output j individually in the time interval $(t_l - \tau,\ t_l]$, the ARMA(p_j, q_j) structure is stated as follows,

$$(1 - \phi_{j,l,1}L - \phi_{j,l,2}L^2 - ... - \phi_{j,l,p_j}L^{p_j})(y_{j,t} - \mu_{j,l}) = (1 + \theta_{j,l,1}L + \theta_{j,l,2}L^2 + ... + \theta_{j,l,q_j}L^{q_j})\varepsilon_{j,t}, \tag{4.3}$$

where $\phi_{j,l,i},\ \ i = 1, ..., p_j$, and $\theta_{j,l,i},\ \ i = 1, ..., q_j$ are the coefficients of the autoregressive and moving average parts, respectively, the p_j and q_j denote the delay orders corresponding to the autoregressive and moving average parts in the ARMA model formulation, respectively, L denotes the delay operator, $\mu_{j,l}$ denotes the mean of the j-th historical observation $y_{j,t}$ in the time window $(t_l - \tau,\ t_l]$, and the sequence $\{\varepsilon_{j,t}\}$ is a white noise error process with zero mean and variance σ_{ε_j}.

Remark 4.1 *The order of ARMA(p_j, q_j) can be different for each l-th DLM model related to each observation time-series $y_{j,t}$. Therefore, the DLM model can be implemented based of fixed*

order or variable order ARMA process. In the case of variable order ARMA model, the well-known Akaike information criterion [188] can be applied for finding the order of the ARMA process.

Another representation for ARMA process in (4.3) can be obtained as

$$y_{j,t} - \mu_{j,l} = \frac{(1 + \theta_{j,l,1}L + \theta_{j,l,2}L^2 + ... + \theta_{j,l,q_j}L^{q_j})}{(1 - \phi_{j,l,1}L - \phi_{j,l,2}L^2 - ... - \phi_{j,l,p_j}L^{p_j})}\varepsilon_{j,t}, \tag{4.4}$$

which yields,

$$y_{j,t} - \mu_{j,l} = \psi_{j,l,1}\varepsilon_{j,t} + \psi_{j,l,2}\varepsilon_{j,t-1} + \cdots, \tag{4.5}$$

where $\psi_{j,l,i},\ i = 1, 2, ...$ denote the coefficients corresponding to the quotient of the term in the right hand side of (4.4), and refer to the fact that the stationary process $y_{j,t} - \mu_{j,l}$ can be written as an infinite autoregressive problem [186]. Therefore, the realisation of (4.5) at $t + 1$ based on the information set $\{\varepsilon_{j,t+1}, \varepsilon_{j,t}, \varepsilon_{j,t-1}, \cdots\}$ is obtained as $y_{j,t+1} - \mu_{j,l} = \psi_{j,l,1}\varepsilon_{j,t+1} + \psi_{j,l,2}\varepsilon_{j,t} + \cdots$. Now, let us define a forecasting function based on the information set up to time t, i.e. $\{\varepsilon_{j,t}, \varepsilon_{j,t-1}, \cdots\}$ as $\hat{Y}_{j,t+1|t} - \mu_{j,l} = \rho_{j,l,1}\varepsilon_{j,t} + \rho_{j,l,2}\varepsilon_{j,t-1} + \cdots$, it was clearly shown in the literature [186, 189, 190] that the mean-square error which minimizes $\mathbb{E}\{(y_{j,t+1} - \hat{Y}_{j,t+1|t})^2\}$ is achieved by setting $\rho_{j,l,i} = \psi_{j,l,i}$. This can also be derived from (4.3) which generates the true value of $y_{j,t+1}$ by setting the unobserved noise $\varepsilon_{j,t+1}$ to zero. Therefore, the one-step ahead forecast at time t using ARMA process (4.3) model is now obtained as,

$$\begin{aligned}\hat{Y}_{j,t+1|t} - \mu_{j,l} = \phi_{j,l,1}(y_{j,t} - \mu_{j,l}) + \phi_{j,l,2}(y_{j,t-1} - \mu_{j,l}) + ... + \phi_{j,l,p_j}(y_{j,t-p_j+1} - \mu_{j,l}) \\ + \theta_{j,l,1}\varepsilon_{j,t} + \theta_{j,l,2}\varepsilon_{j,t-1} + ... + \theta_{j,l,q_j+1}\varepsilon_{j,t-q_j+1}.\end{aligned} \tag{4.6}$$

The ARMA model parameters can be estimated by applying any recursive parameter estimation

method, such as the least-mean squares (LMS), the recursive-least squares (RLS) or the Kalman filters [188] such that the desired cost function $V_\tau(\phi_{j,l,n}, \theta_{j,l,m}) = \frac{1}{\tau}\sum_{t=t_l-\tau+1}^{t_l}(y_{j,t} - \hat{Y}_{j,t|t-1})^2$ is minimized for the available data in time window $t \in (t_l - \tau,\ t_l]$, where $n = 1, ..., p_j$ and $m = 1, ..., q_j$.

We utilize this stationary process for constructing the locally dynamic linear model represented by ARMA process. Afterwards, the observation forecast is performed based on the obtained DLM model. The window size τ refers to the fixed number of available data points used to construct the DLM model and as the new observations become available, the DLM model is updated according to the Mahalanobis distance [187] through an external adjustment loop. In general, the observation forecasting task is accomplished recursively for the window time interval $[t+1,\ t+k]$, denoted by $\hat{Y}_{t+i|t} = (\hat{Y}_{1,t+i|t}, ..., \hat{Y}_{n_y,t+i|t})^{\mathrm{T}},\ i = 1, ..., k$. More details regarding the DLM update is investigated as follows.

The DLM Update Law

As stated before, at $t = t_l$ a new DLM model is developed based on the available data $(t_l-\tau,\ t_l]$. The next step is to check when the developed DLM should be updated. It is assumed that the batch of new observation data are received with the size s, where the minimum value of s is 1 and its maximum value depends on the size of the observation batch data that becomes available at each step. For example, for an aircraft engine the number of recorded data at each flight cycle might be more than one data point for each measurement. Therefore, at $t = t_l + s$ it is required to check the validity of the stationarity assumption.

The validation data set is the available observations in the time interval $[t_l+s-\acute{s}+1,\ t_l+s]$, where $\acute{s}$ is the size of validation data. To check the changes in the observation data, a sliding window of size $\acute{q}$ is moved over the validation data and the Mahalanobis distance metric [187] is used to detect the change as follows. Let us define two $\acute{q}$-tuples which determine the two sliding windows in the calculation of Mahalanobis distance metric for the observation vector as $Q_{i,1} = (y_{t_l+s-\acute{s}+i}, ..., y_{t_l+s-\acute{s}+\acute{q}+i-1})$, and $Q_{i,2} = (y_{t_l+s-\acute{s}+i+1}, ..., y_{t_l+s-\acute{s}+\acute{q}+i})$, for $i = 1, ..., \acute{s}-\acute{q}$. Next, the mean of the data for these two vectors are calculated as

$$\mu_{Q_{i,1}} = \frac{1}{\acute{q}} \sum_{j=0}^{\acute{q}-1} y_{t_l+s-\acute{s}+i+j}, \quad \mu_{Q_{i,2}} = \frac{1}{\acute{q}} \sum_{j=0}^{\acute{q}-1} y_{t_l+s-\acute{s}+i+1+j}, \tag{4.7}$$

and the corresponding covariance matrices are calculated as

$$\Sigma_{Q_{i,1}} = \tfrac{1}{\acute{q}-1} \textstyle\sum_{j=0}^{\acute{q}-1} (y_{t_l+s-\acute{s}+i+j} - \mu_{Q_{i,1}})(y_{t_l+s-\acute{s}+i+j} - \mu_{Q_{i,1}})^{\mathrm{T}}, \tag{4.8}$$

$$\Sigma_{Q_{i,2}} = \tfrac{1}{\acute{q}-1} \textstyle\sum_{j=0}^{\acute{q}-1} (y_{t_l+s-\acute{s}+i+1+j} - \mu_{Q_{i,2}})(y_{t_l+s-\acute{s}+i+1+j} - \mu_{Q_{i,2}})^{\mathrm{T}}. \tag{4.9}$$

The Mahalanobis distance is calculated for the data points in $Q_{i,1}$, and $Q_{i,2}$ as follows,

$$\begin{aligned} D_{\mathcal{M}}(Q_{i,1}) &= \sqrt{(Q_{i,1} - \mu_{Q_{i,1}})^{\mathrm{T}} \Sigma_{Q_{i,1}}^{-1} (Q_{i,1} - \mu_{Q_{i,1}})}, \\ D_{\mathcal{M}}(Q_{i,2}) &= \sqrt{(Q_{i,2} - \mu_{Q_{i,2}})^{\mathrm{T}} \Sigma_{Q_{i,2}}^{-1} (Q_{i,2} - \mu_{Q_{i,2}})}, \end{aligned} \tag{4.10}$$

where, $D_{\mathcal{M}}(.)$ denotes the Mahalanobis distance for the observation vector. The change in the observation data at time $t_l + i$ is detected if the following condition is satisfied according to [187, 191], as

$$\exists i \in \{2, ..., \acute{s} - \acute{q}\} \text{ such that } |D_{\mathcal{M}}(Q_{i,1}) - D_{\mathcal{M}}(Q_{i,2})| > \delta |D_{\mathcal{M}}(Q_{i-1,1}) - D_{\mathcal{M}}(Q_{i-1,2})|, \tag{4.11}$$

where $\delta \geq 1$ is a positive constant which along with $|D_{\mathcal{M}}(Q_{i-1,1}) - D_{\mathcal{M}}(Q_{i-1,2})|$ determines the threshold for the change detection and its determination is application specific and the window length $\acute{s}$, and $\acute{q}$ are chosen such that $\acute{s} - \acute{q} > 2$. Once the change is detected in the observations at $t_l + i$, then $t_{l+1} = t_l + i$ and the new DLM model is obtained based on the observations in the time interval $(t_{l+1} - \tau, t_{l+1}]$, otherwise the DLM model is not updated. However, the validation

algorithm is performed whenever a new s observation data become available, therefore it is executed at time instant $t_l + \acute{k}s$, where $\acute{k}$ denotes the number of data batches received after the last DLM update. The schematic of our proposed DLM update algorithm based on the calculation of Mahalanobis distance is provided in Figure 4.1 on a sample trajectory of $y_{j,t}$.

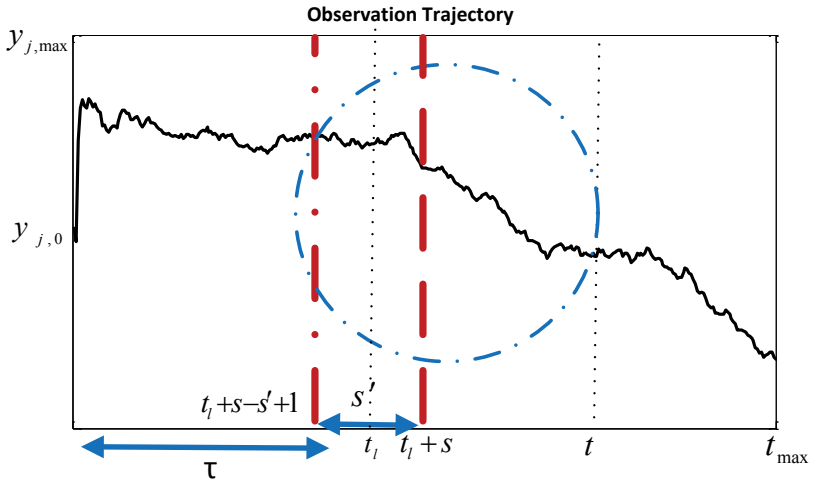
Remark 4.2 *The Mahalanobis distance metric for determining when the DLM model should be updated can be also defined based on the error between the predicted observations from DLM model and the real observations which are available in the time window* $(t_l, t_l + s]$ *as in [185]. In such case, more data is required for evaluating the error based on this criterion.*

4.2.2 Fixed-Lag DLM Model Error Analysis for Observation Forecasting

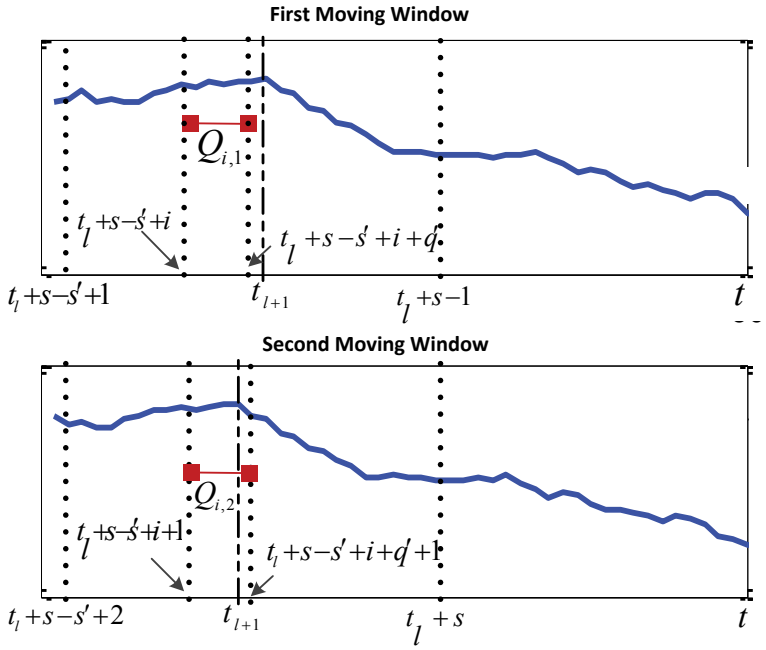
The error analysis corresponding to the observation forecasting scheme is now presented in this subsection. Based on the ARMA(p_j, q_j) model in (4.3), the l-th general univariate DLM (for the j-th output of the dynamical system) can be written by the following state space representation according to [183],

$$\begin{aligned} \mathcal{Y}_{j,t} &= G_{j,l}\mathcal{Y}_{j,t-1} + F_{j,l}\varepsilon_{j,t}, \\ y_{j,t} - \mu_{j,l} &= H\mathcal{Y}_{j,t}, \end{aligned} \tag{4.12}$$

where $y_{j,t}$ for $t \in (t_l - \tau,\ t_l]$ refers to the available observations in the time interval of size τ, $d_j = \max(p_j, q_j + 1)$, $F_{j,l} = (1\ \ \theta_{j,l,1}\ \ ...\ \ \theta_{j,l,d_j-1})^{\mathrm{T}}$, $H = (1\ \ 0\ \ ...\ \ 0)$, $\mathcal{Y}_{j,t} = (y_{j,t} - \mu_{j,l}, y_{j,t-1} - \mu_{j,l}, ..., y_{j,t-d_j+1} - \mu_{j,l})^{\mathrm{T}} \in \mathbb{R}^{d_j}$, $\varepsilon_{j,t}$ denotes the sequence of white noise error,



(a) Observation trajectory



(b) Magnified change in the observation trajectory under two moving windows

Figure 4.1: Change detection in a sample observation trajectory for DLM update.

$$G_{j,l} = \begin{pmatrix} \phi_{j,l,1} & 1 & 0 & \cdots & 0 \\ \phi_{j,l,2} & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \phi_{j,l,d_j-1} & 0 & 0 & \cdots & 1 \\ \phi_{j,l,d_j} & 0 & 0 & \cdots & 0 \end{pmatrix},$$

$\phi_{j,l,i},\ i = 1, ..., d_j$ and $\theta_{j,l,i},\ i = 1, ..., d_j - 1$ denote constant ARMA model coefficients of the l-th DLM of the j-th output. Corresponding to the resulting stationary model, the back substitution of the state equation from (4.12) yields,

$$\mathcal{Y}_{j,t} = G_{j,l}\mathcal{Y}_{j,t-1} + F_{j,l}\varepsilon_{j,t} = \sum_{i=0}^{\infty}(G_{j,l})^i F_{j,l}\varepsilon_{j,t-i}. \tag{4.13}$$

It should be pointed out that the stationarity condition requires that all the eigenvalues of $G_{j,l}$ are located inside the unit circle and, moreover $(G_{j,l})^i$ decreases in (4.13) by increasing i.

Definition 4.1. The forecast of the nonlinear non-stationary observation at time instant k, i.e. $y_{j,t+k}$ made at time $t + k - 1$ using ARMA model based on its linear projection on the last available p_j observations $\{y_{j,t+k-1}, ..., y_{j,t+k-p_j}\}$, is denoted by $y^{\text{lin}}_{j,t+k}$ which is a linear approximation of the actual nonlinear observation process. By applying the one-step ahead prediction algorithm (4.12), $y^{\text{lin}}_{j,t+k}$ is obtained as

$$y^{\text{lin}}_{j,t+k} - \mu_{j,l} = H(G_{j,l}\mathcal{Y}_{j,t+k-1} + F_{j,l}\varepsilon_{j,t+k}) = H(\sum_{i=0}^{\infty}(G_{j,l})^i F_{j,l}\varepsilon_{j,t+k-i}). \tag{4.14}$$

We denote the k-step ahead forecast of $y_{j,t+k}$ made at time t based on the linear approximation according to ARMA model (4.12) as $\hat{Y}_{j,t+k|t}$. The most commonly used criterion for evaluating the performance of the predictor $\hat{Y}_{j,t+k|t}$ is considered to be its mean-square error (MSE) from $y^{\text{lin}}_{j,t+k}$ that is defined by $\mathbb{E}\{(y^{\text{lin}}_{j,t+k} - \hat{Y}_{j,t+k|t})^2\}$. To this aim, $\hat{Y}_{j,t+k|t}$ is obtained by

back substitution of (4.12) into (4.14) when the noise error terms related to the future time are set to zero (i.e., $\varepsilon_{j,t+i} = 0$ for $i > 0$) and applying (4.13), as follows

$$\hat{Y}_{j,t+k|t} - \mu_{j,l} = H((G_{j,l})^{k+1}\mathcal{Y}_{j,t-1} + (G_{j,l})^{k}F_{j,l}\varepsilon_{j,t}) = H((G_{j,l})^{k}\sum_{i=0}^{\infty}(G_{j,l})^{i}F_{j,l}\varepsilon_{j,t-i}). \quad (4.15)$$

The above implies that once the matrix $G_{j,l}$ is determined from the available data within the specified time window, the linear forecast and prediction of the observation can be obtained from (4.15) for a specific prediction horizon by assuming that the process remains stationary.

Below, in Theorem 4.1, an upper bound on the mean square error (MSE) of the k-step ahead forecasting using the ARMA process with the DLM model formulation is obtained. It should be noted that the total forecast error has to be computed from the deviation between the predicted values of the observations, $\hat{Y}_{j,t+k|t}$ and their values from the nonlinear non-stationary observation process, $y_{j,t+k}$ as governed by (4.2). In Theorem 4.2, this total forecast error will be stated as a function of the prediction horizon (k).

First, the following lemma is stated following the results in [192] which is necessary for the proof of Theorem 4.1.

Lemma 4.1. *Let A represent a real square matrix with all its eigenvalues located inside the unit circle, and B represent any real matrix having proper dimension. Then,*

$$\|\sum_{i=0}^{k-1} A^{i}B(A^{i})^{\mathrm{T}}\| \leq \|B\|\frac{\kappa(1-r^{k})}{1-r}, \quad (4.16)$$

where $\|B\|$ denotes the matrix norm and is defined as the spectral norm $\rho^{1/2}(B^{\mathrm{T}}B)$, κ and r are positive constants such that $\rho(A) < r < 1$, and $\|A^{i}\|^{2} \leq \kappa r^{i}$ where $\rho(X)$ denotes the maximum eigenvalue of the matrix X.

Proof: Let us define $S = \sum_{i=0}^{k-1} A^i B (A^i)^{\mathrm{T}}$. Given that $\|A^i\| = \|(A^i)^{\mathrm{T}}\|$, one gets

$$\|S\| \leq \|B\| \sum_{i=1}^{k} \|A^i\|^2. \tag{4.17}$$

Assuming that $\|A^i\|^2 \leq \kappa r^i$, the summation in (4.17) represents a geometric series with common ratio that is less than one ($r < 1$). Therefore, it follows that $\sum_{i=1}^{k} \|A^i\|^2 \leq \sum_{i=0}^{k-1} \kappa r^i = \frac{\kappa(1-r^k)}{1-r}$. Consequently, equation (4.17) can be re-written as

$$\|S\| \leq \|B\| \frac{\kappa(1-r^k)}{1-r}, \tag{4.18}$$

where $\|B\| \frac{\kappa(1-r^k)}{1-r}$ is an upper bound for $\| \sum_{i=0}^{k-1} A^i B (A^i)^{\mathrm{T}} \|$ which is the function of k. This completes the proof of the lemma. ■

The result in Lemma 4.1 is now used to find an upper bound on the MSE observation forecasting error as a function of the prediction horizon (k).

Theorem 4.1 *Consider the ARMA model with the DLM formulation as given by* (4.12)*. Using this model, the k-step ahead forecast of the observation at time t, denoted by $\hat{Y}_{j,t+k|t}$ has a bounded mean square error of $\mathbb{E}\{(y^{\mathrm{lin}}_{j,t+k} - \hat{Y}_{j,t+k|t})^2\} \leq \sigma^2_{\varepsilon_j} \|F_{j,l} F^{\mathrm{T}}_{j,l}\| (\frac{\kappa_j(1-r_j^k)}{1-r_j})$, where $y^{\mathrm{lin}}_{j,t+k}$ is defined in Definition 4.1, $\sigma^2_{\varepsilon_j} = \mathbb{E}\{\varepsilon_{j,t}\varepsilon_{j,t}\}$ denotes the variance of the noise error to $\varepsilon_{j,t}$, and $F_{j,l}$ denotes the vector related to the moving average part of the ARMA model in the DLM formulation, κ_j and r_j denote positive constants such that $\rho(G_{j,l}) < r_j < 1$, and for any integer $i > 0$, $\|(G_{j,l})^i\|^2 \leq \kappa_j r_j^i$.*

Proof: Consider the state space representation of the ARMA model corresponding to each output that is modeled as a univariate time-series with the DLM structure. The MSE related to the k-step ahead forecast of the j-th output after substituting $y^{\mathrm{lin}}_{j,t+k}$ and $\hat{Y}_{j,t+k|t}$ from (4.14) and

(4.15), respectively, can be obtained as follows. First, we have

$$\mathbb{E}\{(y^{\text{lin}}_{j,t+k} - \hat{Y}_{j,t+k|t})^2\} = \mathbb{E}\{(H(\sum_{i=0}^{\infty}(G_{j,l})^i F_{j,\tau}\varepsilon_{j,t+k-i} - (G_{j,l})^k \sum_{i=0}^{\infty}(G_{j,l})^i F_{j,l}\varepsilon_{j,t-i}))^2\}. \quad (4.19)$$

Expanding the term inside the expectation in the right hand side of (4.19) results in

$$\begin{aligned} y^{\text{lin}}_{j,t+k} - \hat{Y}_{j,t+k|t} = H(F_{j,l}\varepsilon_{j,t+k} + G_{j,l}F_{j,l}\varepsilon_{j,t+k-1} + ... + (G_{j,l})^{k-1}F_{j,l}\varepsilon_{j,t+1} + (G_{j,l})^k F_{j,l}\varepsilon_{j,t} \\ + ... - (G_{j,l})^k F_{j,l}\varepsilon_{j,t} - (G_{j,l})^{k+1}F_{j,l}\varepsilon_{j,t-1} - ...) = H(\sum_{i=0}^{k-1}(G_{j,l})^i F_{j,l}\varepsilon_{j,t+k-i}). \end{aligned} \quad (4.20)$$

Hence, the mean-square error is written as

$$\begin{aligned} \mathbb{E}\{(y^{\text{lin}}_{j,t+k} - \hat{Y}_{j,t+k|t})^2\} = \mathbb{E}\{H(F_{j,l}\varepsilon_{j,t+k} + G_{j,l}F_{j,l}\varepsilon_{j,t+k-1} + ... + (G_{j,l})^{k-1}F_{j,l}\varepsilon_{j,t+1}) \\ \times (F_{j,l}\varepsilon_{j,t+k} + G_{j,l}F_{j,l}\varepsilon_{j,t+k-1} + ... + (G_{j,l})^{k-1}F_{j,l}\varepsilon_{j,t+1})^{\text{T}} H^{\text{T}}\}. \end{aligned}$$

Consequently, by considering that $\mathbb{E}\{\varepsilon_{j,t-n}\varepsilon_{j,t-m}\} = \begin{cases} \sigma^2_{\varepsilon_j} & , \text{ if } \quad n = m \\ 0 & , \text{ if } \quad n \neq m \end{cases}$, we have

$$\mathbb{E}\{(y^{\text{lin}}_{j,t+k} - \hat{Y}_{j,t+k|t})^2\} = \sigma^2_{\varepsilon_j} H(\sum_{i=0}^{k-1}(G_{j,l})^i F_{j,l}F^{\text{T}}_{j,l}(G_{j,l})^{i^{\text{T}}})H^{\text{T}}. \quad (4.21)$$

By applying the result from Lemma 4.1 to the right hand side of (4.21), the following upper bound on the forecast error is resulted,

$$\mathbb{E}\{(y^{\text{lin}}_{j,t+k} - \hat{Y}_{j,t+k|t})^2\} \leq \sigma^2_{\varepsilon_j}\|F_{j,l}F^{\text{T}}_{j,l}\|(\frac{\kappa_j(1-r_j^k)}{1-r_j}). \quad (4.22)$$

Consequently, the upper bound on the MSE as $\mathbb{E}\{(y^{\text{lin}}_{j,t+k} - \hat{Y}_{j,t+k|t})^2\}$ is dependent on the prediction horizon k. This completes the proof of Theorem 4.1. ■

It can now be concluded that a threshold for the bound on the forecast error can be chosen by considering it as a percent of the mean $\mu_{j,l}$ within the window. Hence, one can find the maximum acceptable value for k in the observation forecasting algorithm for each observation vector (within the time window $(t,\ t+k]$), by using the l-th DLM model, such that k satisfies the condition $\sigma^2_{\varepsilon_j} \|F_{j,l} F^{\mathrm{T}}_{j,l}\| (\frac{\kappa_j(1-r_j^k)}{1-r_j}) \leq \iota \mu_{j,l}$, where ι denotes the desired percentage of the error in the k-step ahead prediction as a percentage of the mean within the window interval $(t_l-\tau,\ t_l]$. It is pointed out that based on this criterion for finding k, it can be different for each DLM model. Therefore, for ease of notation k is considered as the minimum step ahead prediction horizon that satisfies the mentioned condition for all DLM models.

Our proposed observation forecasting scheme based on the DLM models is only capable of forecasting the observations as univariate time-series. In health prediction strategies one is generally more interested in predicting the system hidden states (that are not necessarily measurable) as well as the system health parameters to evaluate the dynamical system health condition. Our ultimate goal in the proposed health monitoring strategy is to utilize the above developed observation forecasting scheme to predict the propagation of the system states and parameters. Using the k-step ahead forecast of observations, an online prediction method is now developed based on the particle filter to predict the evolution of the system states and parameters. In the following subsections the observation forecasting scheme is integrated with particle filter for enhancements in the prediction performance and capabilities.

4.2.3 Enhancement of Particle Filters for State and Parameter Prediction

Now, the observation forecasting scheme is integrated with the previously developed PF-based dual state and parameter estimation filter, as discussed in Chapter 3. This will allow us to construct a dual estimation algorithm for forecasting and predicting the system states as well as the system slowly time-varying hidden parameters for a k-step ahead horizon. This prediction task is performed by replacing the real observation matrix y_{t+k}, where $y_{t+k} = (y_{1,t+k}, y_{2,t+k}, ..., y_{n_y,t+k})^{\mathrm{T}}$ and is not available after the time instant t, by the forecaseted observation matrix $\hat{Y}_{t+k|t}$, where $\hat{Y}_{t+k|t} = (\hat{Y}_{1,t+k|t}, \hat{Y}_{2,t+k|t}, ..., \hat{Y}_{n_y,t+k|t})^{\mathrm{T}}$. Therefore, the resampling algorithm for both state and parameter estimation filters is performed by utilizing the predicted measurements. It should be noted that to differentiate the forecasted output that is obtained from the fixed-lag DLM forecasting algorithm and the one that is estimated from the particle filters, we use $\hat{Y}_{j,t}$ to designate the forecasted output obtained from the DLM-based algorithm, and $\hat{y}_{j,t}$ to designate the estimated output obtained from the particle filter algorithm.

Consider the dynamical system (4.1) and the associated dual state and parameter estimation filtering expressions given in Table 4.1 for estimation of the *a priori* and *a posteriori* state and parameter distributions. Applying the forecasted measurements $\hat{Y}_{t+k|t}$, the *a priori* and *a posteriori* distributions of the system states and parameters for k-step ahead prediction horizon, are approximated accordingly. The flowchart of our proposed methodology for system state and health parameters prediction is provided in Figure 4.2.

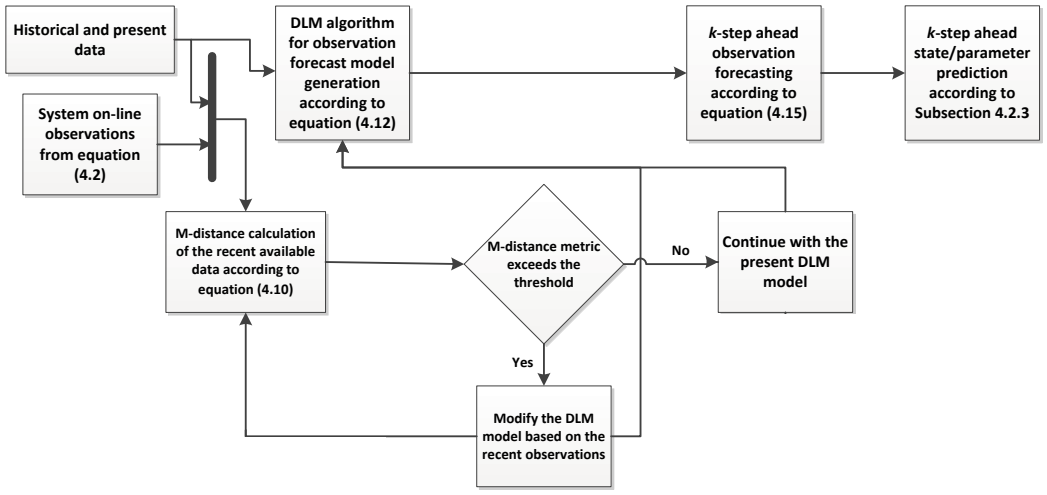


Figure 4.2: Flowchart of the health prediction methodology based on the DLM framework.

k-step Ahead State and Parameter Prediction Algorithm

The details corresponding to states and parameters predictions procedures are provided and summarized below where $r = 1, ..., k$, namely,

1. The r-step ahead *a priori* state prediction:

   (a) Predict the *a priori* state particles as: $\hat{x}^{(i)}_{t+r|t} = f_t(\hat{x}^{(i)}_{t+r-1|t}, \hat{\theta}^{\mathrm{T}}_{t+r-1|t}\lambda(\hat{x}^{(i)}_{t+r-1|t}), \omega^{(i)}_{t+r})$, where $\omega^{(i)}_{t+r}$ is the noise particle added to the state particle at $t + r$ and is generated from the same distribution as $\omega^{(i)}_t$,

   (b) Approximate *a priori* state covariance matrix as:

   $\Sigma_{\hat{x}_{t+r|t}} = \frac{1}{N-1}\sum_{i=1}^{N}(\hat{x}^{(i)}_{t+r|t} - \frac{1}{N}\sum_{j=1}^{N}\hat{x}^{(j)}_{t+r|t})(\hat{x}^{(i)}_{t+r|t} - \frac{1}{N}\sum_{j=1}^{N}\hat{x}^{(j)}_{t+r|t})^{\mathrm{T}}$.

2. The r-step ahead *a posteriori* state prediction:

(a) Calculate the state estimation filter weights for resampling as:

$\tilde{w}_{x_{t+r}}^{(i)} \triangleq \frac{\rho(\hat{Y}_{t+r|t}|\hat{x}_{t+r|t}^{(i)},\hat{\theta}_{t+r|t})}{\sum_{i=1}^{N}\rho(\hat{Y}_{t+r|t}|\hat{x}_{t+r|t}^{(i)},\hat{\theta}_{t+r-1|t})}$,

(b) Approximate the *a posteriori* state distribution from:

$\tilde{\pi}_{x_{t+r|t}}^{N}(\mathrm{d}x_{t+r}) \approx \sum_{l=1}^{N_{\mathrm{reg}}}\sum_{i=1}^{N}\tilde{w}_{x_{t+r}}^{(i)}\frac{|\mathbb{A}_t^{-1}|}{b^{n_x}}\mathcal{K}(\frac{1}{b}\mathbb{A}_t^{-1}(x_{t+r}^{\mathrm{reg}_l}-\hat{x}_{t+r|t}^{(i)})),\ \mathbb{A}_t\mathbb{A}_t^{\mathrm{T}}=\Sigma_{\hat{x}_{t+r|t}}$,

where the kernel density $\mathcal{K}$ is not the optimal kernel i.e. Epanechnikov kernel [193] but it is a simple Gaussian kernel,

(c) Resample the state estimate particles according to the filter weights $\tilde{w}_{x_{t+r}}^{(i)}$, denoted by $\bar{x}_{t+r|t}^{(i)}$ and approximate the *a posteriori* state estimation distribution after resampling according to: $\pi_{x_{t+r|t}}^{N}(\mathrm{d}x_{t+r}) = \frac{1}{N}\sum_{i=1}^{N}\delta_{\bar{x}_{t+r|t}^{(i)}}(\mathrm{d}x_{t+r})$,

(d) Approximate the *a posterior* state: $\hat{x}_{t+r|t} = \frac{1}{N}\sum_{i=1}^{N}\bar{x}_{t|t-1}^{(i)}$.

3. The first step r-step ahead *a posteriori* parameter prediction:

(a) Calculate the corrected output prediction error as:

$\epsilon_{t+r}(\hat{\theta}_{t+r-1|t}^{(j)}) = \hat{Y}_{t+r|t} - h_t(\hat{x}_{t+r|t},\hat{\theta}_{t+r-1|t}^{(j)^{\mathrm{T}}}\lambda(\hat{x}_{t+r|t}))$,

(b) Calculate $m_{t+r}^{(j)}$ as: $m_{t+r}^{(j)} = \hat{\theta}_{t+r-1|t}^{(j)} + \gamma_{t+r}R_{t+r}^{(j)}\psi_{t+r}^{(j)}\epsilon_{t+r}(\hat{\theta}_{t+r-1|t}^{(j)})$,

(c) Calculate the first step *a posteriori* parameter prediction by applying the kernel shrinkage algorithm as: $\tilde{\theta}_{t+r|t}^{(j)} = Am_{t+r}^{(j)} + (I-A)\bar{m}_{t+r-1} + \zeta_{t+r}^{(j)},\ \bar{m}_{t+r-1} = \frac{1}{N}\sum_{j=1}^{N}\hat{\theta}_{t+r-1|t}^{(j)},\ \zeta_{t+r}^{(j)} \sim \mathcal{N}(0,(I-A^2)\Sigma_{\hat{\theta}_{t+r-1|t}})$,

4. The second step r-step ahead *a posteriori* parameter prediction:

(a) Calculate the weights in the parameter estimation filter for performing resampling as:

$\tilde{w}_{\theta_{t+r}}^{(j)} \triangleq \frac{\rho(\hat{Y}_{t+r|t}|\hat{x}_{t+r|t},\tilde{\theta}_{t+r|t}^{(j)})}{\sum_{j=1}^{M}\rho(\hat{Y}_{t+r|t}|\hat{x}_{t+r|t},\tilde{\theta}_{t+r|t}^{(j)})}$,

(b) Approximate the first step *a posteriori* parameter distribution according to:

$\tilde{\pi}^N_{\theta_{t+r|t}}(\mathrm{d}\theta_{t+r}) \approx \sum_{j=1}^N \tilde{w}^{(j)}_{\theta_{t+r}} \delta_{\bar{\theta}^{(j)}_{t+r|t}}(\mathrm{d}\theta_{t+r})$,

(c) Perform resampling to approximate the *a posteriori* parameter prediction distribution based on the resampled parameter predicted particles $\bar{\theta}^{(j)}_{t+r|t}$ as:

$\pi^N_{\theta_{t+r|t}}(\mathrm{d}\theta_{t+r}) = \frac{1}{N} \sum_{j=1}^N \delta_{\bar{\theta}^{(j)}_{t+r|t}}(\mathrm{d}\theta_{t+r})$,

(d) Obtain the *a posteriori* parameter prediction: $\hat{\theta}_{t+r|t} = \frac{1}{N} \sum_{j=1}^N \bar{\theta}^{(j)}_{t+r|t}$,

(e) Calculate the *a posteriori* parameter prediction covariance:

$\Sigma_{\hat{\theta}_{t+r|t}} = \frac{1}{N-1} \sum_{i=1}^N (\bar{\theta}^{(j)}_{t+r|t} - \hat{\theta}_{t+r|t})(\bar{\theta}^{(j)}_{t+r|t} - \hat{\theta}_{t+r|t})^{\mathrm{T}}$.

Consequently, the state and parameter probability density functions can be generated for the k-step ahead prediction time instants by utilizing the predicted observations and by maintaining resampling for the future time instants.

Remark 4.3 *Consider the set $D_{\mathcal{N}}$ as a set for which the functions $f_t(x_t, \theta_t, \omega_t)$, and $h_t(x_t, \theta_t)$ are sufficiently smooth. Using a projection as stated in Chapter 3, ensures that $\hat{\theta}^{(j)}_{t|t-1},\ j = 1, ..., N$ and consequently $\hat{\theta}^{(j)}_{t|t}$ in the estimation algorithm as stated in Table 4.1 will remain inside the subset $\bar{D}$ of $D_{\mathcal{N}}$ ($\bar{D} \subset D_{\mathcal{N}}$). On the other hand, for any dynamical system $\forall (x_t, \theta_t) \in \mathcal{D}_{\mathcal{R}}$, where $\mathcal{D}_{\mathcal{R}}$ denotes the stability region of the dynamical system, the function $h_t(x_t, \theta_t^{\mathrm{T}} \lambda(x_t))$ is bounded, where $D_{\mathcal{N}} \subset D_{\mathcal{R}}$. Therefore, the existence of the mentioned mapping in the estimation and also r-step ahead prediction algorithm for system states and parameters guarantees that for $r > 1$, $h_t(\hat{x}_{t+r|t+r}, \hat{\theta}^{(j)^{\mathrm{T}}}_{t+r-1|t+r-1} \lambda(\hat{x}_{t+r|t+r}))$, and $h_t(\hat{x}_{t+r|t}, \hat{\theta}^{(j)^{\mathrm{T}}}_{t+r-1|t} \lambda(\hat{x}_{t+r|t}))$ remains bounded where these bounds are denoted by $\mathcal{C}_1$, and $\mathcal{C}_2$, respectively.*

It is pointed out that in the parameter estimation filter, the prediction error $\epsilon_{t+r}(\hat{\theta}^{(j)}_{t+r-1|t})$ is used to generate a propagation law for the unknown parameters of the system. Due to lack of observations after time instant t, the forecasted observations utilizing the fixed-lag DLM models are used for this purpose. Therefore, as the error increases in the observation forecasting scheme, it can lead to incorrect predicted parameters as well as states. Consider the prediction error regarding to j-th parameter particle calculated at time instant $t+r$ assuming that observations up to $t+r$ become available as: $\epsilon^{(j)}_{t+r} = y_{t+r} - h_t(\hat{x}_{t+r|t+r}, \hat{\theta}^{(j)^{\mathrm{T}}}_{t+r-1|t+r-1}\lambda(\hat{x}_{t+r|t+r}))$, and the r-step ahead prediction of $\epsilon^{(j)}_{t+r|t}$ assuming that observations up to time instant t are available, as: $\epsilon^{(j)}_{t+r|t} = \hat{Y}_{t+r|t} - h_t(\hat{x}_{t+r|t}, \hat{\theta}^{(j)^{\mathrm{T}}}_{t+r-1|t}\lambda(\hat{x}_{t+r|t}))$. Therefore, the discrepancy between these two quantities will affect the accuracy of the state and parameter prediction algorithm, which is calculated as,

$$\begin{aligned}
|\epsilon^{(j)}_{t+r} - \epsilon^{(j)}_{t+r|t}| &= |y_{t+r} - \hat{Y}_{t+r|t} + h_t(\hat{x}_{t+r|t}, \hat{\theta}^{(j)}_{t+r-1|t}) - h_t(\hat{x}_{t+r|t+r}, \hat{\theta}^{(j)}_{t+r-1|t+r-1})| \\
&\leq |y_{t+r} - \hat{Y}_{t+r|t}| + |h_t(\hat{x}_{t+r|t}, \hat{\theta}^{(j)}_{t+r-1|t}) - h_t(\hat{x}_{t+r|t+r}, \hat{\theta}^{(j)}_{t+r-1|t+r-1})| \\
&\leq |y_{t+r} - \hat{Y}_{t+r|t}| + |\mathcal{C}_1| + |\mathcal{C}_2|,
\end{aligned} \tag{4.23}$$

where the last inequality is obtained based on the results summarized in Remark 4.3. Now, the main goal is to investigate the boundedness of $|y_{t+r} - \hat{Y}_{t+r|t}|$ to ensure the boundedness of the prediction algorithm. To this aim, an element-wise approach based on the error generated for each forecaseted observation as compared to the real nonlinear non-stationary process (4.2), for the long-term horizon prediction is developed in the following theorem.

Theorem 4.2 *Consider the stochastic nonlinear system as described by* (4.1) *and* (4.2)*, where the observation noise to $y_{j,t}$ is considered to be generated from a non-stationary stochastic process with bounded variance $\sigma^2_{y_{j,t}}$. The k-step ahead observation prediction based on DLM model using ARMA process followed by particle filter, results in the following bound on the*

mean square error,

$$\mathbb{E}\{(y_{j,t+k} - \hat{Y}_{j,t+k|t})^2\} \leq \sigma^2_{y_{j,t}} + \mathcal{C}_{j,l} + \mathcal{D}_{j,l} + \mathcal{B}_{j,l} + 2\sqrt{(\sigma^2_{y_{j,t}} + \mathcal{C}_{j,l} + \mathcal{D}_{j,l})\mathcal{B}_{j,l}},$$

where $\mathcal{B}_{j,l} = \sigma^2_{\varepsilon_j}\|F_{j,l}F^{\mathrm{T}}_{j,l}\|(\frac{\kappa_1(1-r_j^k)}{1-r_j})$*, assuming* κ_j *and* $r_j,\ j = 1, ..., n_y$ *denote positive constants that satisfy* $\|G^i_{j,l}\| \leq \kappa_j r^i_j$*.*

Moreover, $\mathcal{C}_{j,l} = h^2_{j,t+k}(x_{t+k}, \theta^{\mathrm{T}}_{t+k}\lambda(x_{t+k})) + \mu^2_{j,l} - 2\mu_{j,l}h_{j,t+k}(x_{t+k}, \theta^{\mathrm{T}}_{t+k}\lambda(x_{t+k}))$*, and* $\mathcal{D}_{j,l} = \sigma^2_{\varepsilon_j}\|F_{j,l}F^{\mathrm{T}}_{j,l}\|(\frac{\kappa_j}{1-r_j})$.

Proof: Let us construct a vector of k-step ahead predicted observations that are obtained through n_y univariate time-series through the fixed-lag DLM models as:
$\hat{Y}_{t+k|t} = (\hat{Y}_{1,t+k|t}, \hat{Y}_{2,t+k|t}, ..., \hat{Y}_{n_y,t+k|t})^{\mathrm{T}}$. At the time instant t the forecasted observations (from the linear regression model according to ARMA process), $\hat{Y}_{j,t+k|t}$ are independent from one another since each observation is constructed only from its own historical data, i.e.,
$\mathrm{cov}(\hat{Y}_{n,t+k|t}\hat{Y}_{m,t+k|t}) = 0$, for $n \neq m$. Consequently, the covariance matrix of $\hat{Y}_{t+k|t}$ reduces to the variance matrix with the diagonal entities as $\mathrm{var}(\hat{Y}_{j,t+k|t}),\ j = 1, ..., n_y$.

The k-step ahead prediction error of the j-th observation, when it is obtained based on the fixed-lag DLM model using ARMA process, is approximated along the lines described in Theorem 4.1. In the referred theorem, the nonlinear observation $y_{j,t}$ was approximated with its linear projection according to ARMA process introduced in (4.3) and defined in Definition 4.1, i.e. at time t, $y_{j,t} \approx y^{\mathrm{lin}}_{j,t}$. Therefore, the error due to mismatch between the actual nonlinear observation and its corresponding linear projection (4.14) in k-step ahead prediction horizon is

calculated as,

$$\begin{aligned}\mathbb{E}\{(y_{j,t+k}-y_{j,t+k}^{\text{lin}})^2\} &= \mathbb{E}\{(y_{j,t+k}-\mu_{j,l}-H\sum_{i=0}^{\infty}(G_{j,l})^iF_{j,l}\varepsilon_{j,t+k-i})(y_{j,t+k}-\mu_{j,l}\\ &\quad - H\sum_{i=0}^{\infty}(G_{j,l})^iF_{j,l}\varepsilon_{j,t+k-i})^{\mathrm{T}}\}\\ &= \mathbb{E}\{y_{j,t+k}^2\}-2\mathbb{E}\{y_{j,t+k}\mu_{j,l}\}-2\mathbb{E}\{y_{j,t+k}(\sum_{i=0}^{\infty}(G_{j,l})^iF_{j,l}\varepsilon_{j,t+k-i})^{\mathrm{T}}H^{\mathrm{T}}\}\\ &\quad +\mathbb{E}\{\mu_{j,l}^2\}+2\mathbb{E}\{\mu_{j,l}(\sum_{i=0}^{\infty}(G_{j,l})^iF_{j,l}\varepsilon_{j,t+k-i})^{\mathrm{T}}H^{\mathrm{T}}\}\\ &\quad +\mathbb{E}\{H(\sum_{i=0}^{\infty}(G_{j,l})^iF_{j,l}\varepsilon_{j,t+k-i})(\sum_{i=0}^{\infty}(G_{j,l})^iF_{j,l}\varepsilon_{j,t+k-i})^{\mathrm{T}}H^{\mathrm{T}}\}.\end{aligned} \tag{4.24}$$

Assume that the white noise process $\varepsilon_{j,t}$ and the measurement noise process ν_t in (4.2) are zero-mean process with covariance matrices $\sigma^2_{\varepsilon_{j,t}}$, and $V_t = \text{diag}(\sigma^2_{\nu_{1,t}},\ \cdots,\ \sigma^2_{\nu_{n_y,t}})$, respectively. Moreover, v_t and $\varepsilon_{j,t}$ are two independent process which yields that $\mathbb{E}\{y_{j,t}\epsilon_{j,t}\} = 0,\ \forall t$, and $\mathbb{E}\{y_{j,t}\} = \mathbb{E}\{h_t(x_t,\theta_t^{\mathrm{T}}\lambda(x_t))+\nu_{j,t}\} = h_t(x_t,\theta_t^{\mathrm{T}}\lambda(x_t))$. Furthermore, considering that $y_{j,t+k}$ is conditionally independent from $\epsilon_{j,t+k}$ (since $\epsilon_{j,t+k}$ is used to construct $y_{j,t+k}^{\text{lin}}$ not $y_{j,t+k}$) then (4.24) reduces as,

$$\begin{aligned}\mathbb{E}\{(y_{j,t+k}-y_{j,t+k}^{\text{lin}})^2\} &= \sigma^2_{\nu_{j,t+k}}+h^2_{j,t+k}(x_{t+k},\theta_{t+k}^{\mathrm{T}}\lambda(x_{t+k}))-2\mu_{j,l}h_{j,t+k}(x_{t+k},\theta_{t+k}^{\mathrm{T}}\lambda(x_{t+k}))-0\\ &\quad +\mathbb{E}\{\mu_{j,l}^2\}+\sigma^2_{\varepsilon_j}H(\sum_{i=0}^{\infty}(G_{j,l})^iF_{j,l}F_{j,l}^{\mathrm{T}}(G_{j,l})^{i^{\mathrm{T}}})H^{\mathrm{T}}.\end{aligned} \tag{4.25}$$

Now, one requires to calculate the error due to the k-step ahead forecast of the observation vector according to ARMA process, denoted by $\hat{Y}_{j,t+k|t}$ with the original nonlinear observations $y_{j,t+k}$, where $\hat{Y}_{j,t+k|t}$ is considered to be obtained from (4.15) in the selected time window as long as the l-th DLM is valid. Hence, considering the linear projection $y_{j,t+k}^{\text{lin}}$ for k-step ahead

prediction horizon, the expectation of the error between $y_{j,t+k}$, and $\hat{Y}_{j,t+k|t}$ can be stated as,

$$\mathbb{E}\{|y_{j,t+k} - \hat{Y}_{j,t+k|t}|\} = \mathbb{E}\{|y_{j,t+k} - y^{\text{lin}}_{j,t+k} + y^{\text{lin}}_{j,t+k} - \hat{Y}_{j,t+k|t}|\}. \tag{4.26}$$

Consider the MSE representation $E\{|y_{j,t+k} - y^{\text{lin}}_{j,t+k} + y^{\text{lin}}_{j,t+k} - \hat{Y}_{j,t+k|t}|^2\}$, Minkowski's inequality [194] is utilized for obtaining the following inequality

$$\begin{aligned}\mathbb{E}\{|y_{j,t+k} - y^{\text{lin}}_{j,t+k} + y^{\text{lin}}_{j,t+k} - \hat{Y}_{j,t+k|t}|^2\} \leq \mathbb{E}\{(y_{j,t+k} - y^{\text{lin}}_{j,t+k})^2\} + \mathbb{E}\{(y^{\text{lin}}_{j,t+k} - \hat{Y}_{j,t+k|t})^2\} \\ + 2\sqrt{\mathbb{E}\{y_{j,t+k} - y^{\text{lin}}_{j,t+k}\}^2 \mathbb{E}\{y^{\text{lin}}_{j,t+k} - \hat{Y}_{j,t+k|t}\}^2}.\end{aligned} \tag{4.27}$$

Finally, the upper bound on the error is obtained as

$$\begin{aligned}E\{|y_{j,t+k} - \hat{Y}_{j,t+k|t}|^2\} &\leq \mathbb{E}\{(y_{j,t+k} - y^{\text{lin}}_{j,t+k})^2\} + \mathbb{E}\{(y^{\text{lin}}_{j,t+k} - \hat{Y}_{j,t+k|t})^2\} \\ &\quad + 2\sqrt{\mathbb{E}\{y_{j,t+k} - y^{\text{lin}}_{j,t+k}\}^2 \mathbb{E}\{y^{\text{lin}}_{j,t+k} - \hat{Y}_{j,t+k|t}\}^2}, \\ &\leq \sigma^2_{y_{j,t}} + \mathcal{C}_{j,l} + \mathcal{D}_{j,l} + \mathcal{B}_{j,l} + 2\sqrt{(\sigma^2_{y_{j,t}} + \mathcal{C}_{j,l} + \mathcal{D}_{j,l})\mathcal{B}_{j,l}},\end{aligned} \tag{4.28}$$

where $\mathcal{D}_{j,l}$ is obtained according to Lemma 4.1 as an upper bound for the last term in the right hand side of (4.25). Moreover, $\mathcal{B}_{j,l}$ is an upper bound on the observation forecast error from the DLM model according to Theorem 4.1. Consequently, considering that for any dynamical system $\forall(x_t, \theta_t) \in \mathcal{D}_{\mathcal{R}}$, the function $h_t(x_t, \theta_t^{\mathrm{T}}\lambda(x_t))$ and as a result $\mathcal{C}_{j,l}$ is bounded, hence the k-step ahead prediction error due to observations remains bounded. This therefore completes the proof of the theorem. ■

In the following section computational complexity of our developed algorithm for the dual state and parameter estimation and their propagation prediction is evaluated and studied. The complexity results are compared with the ones corresponding to a conventional augmented parameter and state estimation method [25, 195] based on the particle filters with the fixed weight

equally weighted propagation law for the particles for the long-term prediction horizon.

4.3 Complexity Analysis

In this section, the computational complexity of our proposed prediction algorithm based on our previously developed dual state and parameter estimation method integrated with the fixed-lag DLM model observation forecasting scheme for the prediction of the long-term behavior of the system states and parameters is quantitatively obtained and analyzed. The analysis is based on the number of floating-point operations (flops) that are required by the selected algorithms as explained in Chapter 3. The dimensions of the entities in the dual state and parameter estimation algorithm is considered according to Table 3.1. The coefficient c_4 in this chapter is used to represent the complexity of the ARMA model.

The complexity of our proposed dual state and parameter estimation and its propagation predictions can be compared with the complexity of a conventional algorithm for state and constant parameter estimating using particle filters when the particles are propagated with fixed weights to the future time instants [25, 195] as stated in Table 4.4. In the standard and conventional algorithm the parameters are augmented to the state vector, therefore the dimension of the augmented state and parameter system becomes $n_x + n_\theta$ and the complexity associated with the resampling step, i.e., c_2 was removed from the EF complexity evaluation.

In Table 4.5, the EF complexity of the two methods are summarized. To compare the EF complexity results, only the dominant parts of $C(n_x, n_\theta, c_1, c_2, c_3, c_4, N)$ (that represents the EF

Table 4.2: The Equivalent Complexity for the state estimation/prediction step

Instruction	Mult.	Add	Func. Eval.	Other
$[U_1, T_1] = \text{schur}(\Sigma_{\hat{x}_{t+r-1\|t}})$	–	–	–	$10n_x^3$
$R_1 = \text{randn}(n_x, N)$	–	–	–	Nn_xc_1
$\omega_{t+r}^{(i)} = (U_1\sqrt{T_1})R_1$	$n_x^3 + Nn_x^2$	$(n_x-1)n_x^2 + N(n_x-1)n_x$	–	n_x^2
$\hat{x}_{t+r\|t}^{(i)} = f_t(\hat{x}_{t+r-1\|t}^{(i)}, \hat{\theta}_{t+r-1\|t}^{\mathrm{T}}\lambda(\hat{x}_{t+r-1\|t}^{(i)}), \omega_{t+r}^{(i)})$	–	–	Nn_x	–
$\hat{y}_{t+r\|t}^{(i)} = h_t(\hat{x}_{t+r\|t}^{(i)}, \hat{\theta}_{t+r-1\|t}^{\mathrm{T}}\lambda(x_{t+r\|t}^{(i)}))$	–	–	Nn_y	–
$\Sigma_{\hat{x}_{t+r\|t}} = \frac{1}{N-1}\sum_{i=1}^{N}(\bar{x}_{t+r\|t}^{(i)} - \hat{x}_{t+r\|t})(\bar{x}_{t+r\|t}^{(i)} - \hat{x}_{t+r\|t})^{\mathrm{T}}$	Nn_x^2	$2Nn_x$	–	–
Regularization and resampling to find weights $w_{x_{t+r}}^{(i)}$ and $\bar{x}_{t+r\|t}^{(i)}$	–	–	–	$Nn_xc_2 + Nn_xc_3$
$\hat{x}_{t+r\|t} = \frac{1}{N}\sum_{i=1}^{N}\bar{x}_{t+r\|t}^{(i)}$	n_x	Nn_x	–	–
Total	$n_x^3 + 2Nn_x^2 + n_x$	$n_x^3 + (N-1)n_x^2 + 2Nn_x$	$N(n_x+n_y)$	$10n_x^3 + n_x^2$ $Nn_x(c_1+c_2+c_3)$

Table 4.3: The Equivalent Complexity for the parameter estimation/prediction step using the Observation prediction scheme

Instruction	Mult.	Add	Func. Eval.	Other
$\bar{y}_{t+r\|t}^{(j)} = h_t(\hat{x}_{t+r\|t}, \theta_{t+r-1\|t}^{(j)^{\mathrm{T}}}\lambda(\hat{x}_{t+r\|t}))$	–	–	Nn_y	–
$\Sigma_\theta = (I - A^2)\Sigma_{\hat{\theta}_{t+r-1\|t}}$	n_θ^3	$(n_\theta-1)n_\theta^2 + n_\theta^2$	–	–
$\epsilon_{t+r}^{(j)} = \hat{Y}_{t+r} - \bar{y}_{t+r\|t}^{(j)}$	–	Nn_y	–	–
$\psi_{t+r}^{(j)} = \frac{dh}{d\theta}\|_{\hat{x}_{t+r\|t}, \theta_{t+r-1\|t}^{(j)}}$	–	–	n_yn_θ	–
$P_{t+r}^{(j)} = \gamma_{t+r}(\sqrt{\text{trace}(\epsilon_{t+r}^{(j)}\epsilon_{t+r}^{(j)^{\mathrm{T}}})}$	$N + Nn_y$	$N(n_y-1) + Nn_y$	–	–
$[U_2, L_2] = \text{schur}(\Sigma_\theta)$	–	–	–	$10n_\theta^3$
$R_2 = \text{randn}(n_\theta, N))$	–	–	–	$Nn_\theta c_1$
$\zeta_{t+r}^{(j)} = (U_2\sqrt{L_2})R_2$	$n_\theta^3 + Nn_\theta^2$	$(n_\theta-1)n_\theta^2 + N(n_\theta-1)n_\theta$	n_θ^2	–
$m_{t+r}^{(j)} = \tilde{\theta}_{t+r-1\|t}^{(j)} + P_{t+r}^{(j)}\psi_{t+r}^{(j)}\epsilon_{t+r}^{(j)}$	$N(n_yn_\theta + n_\theta)$	$N(n_y-1)n_\theta + Nn_\theta$	–	–
$\tilde{\theta}_{t+r\|t}^{(j)} = Am_{t+r}^{(j)} + (I-A)\frac{1}{N}\sum_{j=1}^{N}\tilde{\theta}_{t+r-1\|t}^{(j)} + \zeta_{t+r}^{(j)}$	$Nn_\theta^2 + n_\theta$	$Nn_\theta^2 + 2Nn_\theta + Nn_\theta + n_\theta^2$	–	–
$\hat{y}_{t+r\|t} = h_t(\hat{x}_{t+r\|t}, \hat{\theta}_{t+r\|t}^{(j)^{\mathrm{T}}}\lambda(\hat{x}_{t+r\|t}))$	–	–	Nn_y	–
Resampling to find weights, $w_{\theta_{t+r}}^{(j)}$, and $\bar{\theta}_{t+r\|t}^{(j)}$	–	–	–	$Nn_\theta c_2$
$\hat{\theta}_{t+r\|t} = \frac{1}{N}\sum_{j=1}^{N}\bar{\theta}_{t+r\|t}^{(j)}$	n_θ	Nn_θ	–	–
$\Sigma_{\hat{\theta}_{t+r\|t}} = \frac{1}{N-1}\sum_{i=1}^{N}(\bar{\theta}_{t+r\|t}^{(j)} - \hat{\theta}_{t+r\|t})(\bar{\theta}_{t+r\|t}^{(j)} - \hat{\theta}_{t+r\|t})^{\mathrm{T}}$	Nn_θ^2	$2Nn_\theta$	–	–
M-Distance calculation for $\acute{s}$ observations	–	–	–	$\acute{s}(q^{-1})n_y$
ARMA evaluation after τ observations	–	–	–	$\tau c_4 n_y$
Total	$2n_\theta^3 + 3Nn_\theta^2$ $+(N+2)n_\theta + Nn_\theta n_y$ $+N(n_y+1)$	$2n_\theta^3 + 2Nn_\theta^2$ $+5Nn_\theta + 3Nn_y - N$ $+Nn_\theta n_y$	$n_\theta^2 + 2Nn_y$ $+n_yn_\theta$	$10n_\theta^3 + Nn_\theta c_1 + Nn_\theta c_2$ $+\acute{s}(\acute{q})^{-1}n_y$ $+\tau c_4 n_y$

Table 4.4: The Equivalent Complexity for the augmented state and parameter estimation/prediction scheme

Instruction	Mult.	Add	Func. Eval.	Other
$[U_1, T_1] = \text{schur}(\Sigma_{x,\theta})$	–	–	–	$10(n_x + n_\theta)^3$
$R_1 = \text{randn}(n_x + n_\theta, N)$	–	–	–	$N(n_x + n_\theta)c_1$
$\omega_{t+r}^{(i)} = (U_1\sqrt{T_1})R_1$	$(n_x + n_\theta)^3 + N(n_x + n_\theta)^2$	$(n_x + n_\theta - 1)(n_x + n_\theta)^2$ $+N(n_x + n_\theta - 1)(n_x + n_\theta)$	–	$(n_x + n_\theta)^2$
$\omega_{x_{t+r}}^{(i)} = \omega_{t+r}^{(i)}(1 : n_x)$	–	–	–	–
$\omega_{\theta_{t+r}} = (I - A^2)\omega_{t+r}^{(i)}(n_x + 1 : n_x + n_\theta)$	n_θ^3	$(n_\theta - 1)n_\theta^2 + n_\theta^2$	–	–
state/parameter augmentation: $[\hat{x}_{t+r\|t}^{(i)}; \hat{\theta}_{t+r\|t}^{(i)}] =$ $[f_t(\hat{x}_{t+r-1\|t}^{(i)}, \hat{\theta}_{t+r-1\|t}^{(i)^{\mathrm{T}}}\lambda(\hat{x}_{t+r-1\|t}^{(i)}), \omega_{x_{t+r}}^{(i)}); \hat{\theta}_{t+r-1\|t}^{(i)}]$	–	–	$N(n_x + n_\theta)$	–
$\hat{y}_{t+r\|t}^{(i)} = h_t(\hat{x}_{t+r\|t}^{(i)}, \hat{\theta}_{t+r\|t}^{(i)^{\mathrm{T}}}\lambda(\hat{x}_{t+r\|t}^{(i)}))$	–	–	Nn_y	–
$\Sigma_{x,\theta} = \frac{1}{N-1}\sum_{i=1}^{N}([\bar{x}_{t+r\|t}^{(i)}; \bar{\theta}_{t+r\|t}^{(i)}] - [\hat{x}_{t+r\|t}; \hat{\theta}_{t+r\|t}])$ $\times([\bar{x}_{t+r\|t}^{(i)}; \bar{\theta}_{t+r\|t}^{(i)}] - [\hat{x}_{t+r\|t}; \hat{\theta}_{t+r\|t}])^{\mathrm{T}}$	$N(n_x + n_\theta)^2$	$2N(n_x + n_\theta)$	–	–
Regularization without resampling to find, $\bar{x}_{t+r\|t}^{(i)}$ and $\bar{\theta}_{t+r\|t}^{(i)}$	–	–	–	$N(n_x + n_\theta)c_3$
$[\hat{x}_{t+r\|t}; \hat{\theta}_{t+r\|t}] = [\frac{1}{N}\sum_{i=1}^{N}\bar{x}_{t+r\|t}^{(i)}; \frac{1}{N}\sum_{i=1}^{N}\bar{\theta}_{t+r\|t}^{(i)}]$	$n_x + n_\theta$	$N(n_x + n_\theta)$	–	–
Total	$2n_\theta^3 + n_x^3 + n_x^2(2N + 3n_\theta)$ $+n_\theta^2(3n_x + 2N) + 4Nn_\theta n_x$ $+n_x + n_\theta$	$n_x^3 + 2n_\theta^3 + n_x^2(3n_\theta - 1 + N)$ $+n_\theta^2(N - 1 + 3n_x)$ $+n_x n_\theta(2N - 2)$	$N(n_x + n_\theta)$ $+Nn_y$	$10n_x^3 + 10n_\theta^3$ $n_x^2(30n_\theta + 1)$ $+n_\theta^2(30n_x + 1)$ $+n_x(Nc_1 + 2n_\theta + Nc_3)$ $+n_\theta(Nc_1 + Nc_3)$

Table 4.5: The Total Equivalent Complexity of the Filters

Prediction Method	Total Equivalent Complexity
DLM-Based Prediction Method	$C(n_x, n_\theta, c_1, c_2, c_3, c_4, N)$= $12n_x^3 + 14n_\theta^3 + N(3n_x^2 + 5n_\theta^2 + 6n_\theta$ $+2n_\theta n_y + 7n_y + 3n_x + c_1(n_x + n_\theta) + c_2(n_x + n_\theta) + c_3 n_x) + c_1(n_x + n_\theta)$ $+c_2(n_x + n_\theta) + c_3 n_x) + n_x + n_\theta^2 + 2n_\theta + n_y n_\theta + n_y(\tau c_4 + (\acute{q})^{-1}\acute{s})$
Standard PF-Based Prediction Method	$\acute{C}(n_x, n_\theta, c_1, c_3, N)$ = $12n_x^3 + 14n_\theta^3 + N(3n_x^2 + 3n_\theta^2 + 6n_x n_\theta$ $+(1 + c_1 + c_3)n_x + (1 + c_1 + c_3)n_\theta + n_y) + 36n_\theta^2 n_x + 36n_x^2 n_\theta + n_x + n_\theta$

complexity of our proposed method) and $\acute{C}(n_x, n_\theta, c_1, c_3, N)$ (that represents the conventional state and parameter prediction method based on particle filters in the literature [25, 195]) are provided. This selection is justified by the fact that $N \gg 1$, therefore the dominant parts are the parts that are related to N.

Let us assume that the time interval $\tau = \beta N$, and $\beta < 1$ is a constant. To quantitatively evaluate the EF complexity, two cases are considered now. In the first case, it is assumed that the measurement dimension, n_y as well as the parameter dimension ($n_\theta \leq n_y$) is much smaller than the state dimension. Consequently, the components related to only n_x are considered, that is

$$\begin{aligned} C(n_x, n_\theta, c_1, c_2, c_3, c_4, N) &\approx N(3n_x^2 + 3n_x + c_1 n_x + c_2 n_x + c_3 n_x), \\ \acute{C}(n_x, n_\theta, c_1, c_2, c_3, N_0) &\approx N_0(3n_x^2 + 6n_x n_\theta + n_x + c_1 n_x + c_3 n_x), \end{aligned} \tag{4.29}$$

where N_0 denotes the number of particles that are required for implementation of the conventional method.

It follows from (4.29) that for achieving the same EF complexity in the two methods, the number of particles that are required in our proposed method can be determined based on the number of the particles that are required in the conventional method as follows,

$$N = N_0\left(1 - \frac{2n_x + c_2 n_x - 6n_x n_\theta}{3n_x^2 + 3n_x + c_1 n_x + c_2 n_x + c_3 n_x}\right), \tag{4.30}$$

It should be pointed out that resampling algorithm deals with ordering, therefore it is assumed that in the worst case its computational complexity is much greater than other operations in the algorithm. Hence, it follows clearly that for a given complexity, since $c_2 + 2 > 6n_\theta$ (this results in $N < N_0$), one should use fewer particles in our proposed method as compared to the conventional method. Therefore, the conventional method can be implemented with more particles

as compared to our proposed method to achieve the same complexity, whereas increasing the particles in our proposed methods results in more computational complexity and consequently more implementation cost. To decrease the cost of implementing our proposed algorithm, the coefficient of N_0 in (4.30) has to be as much as possible near to 1. As a result, one should get

$$3n_x^2 + 3n_x + c_1 n_x + c_2 n_x + c_3 n_x \gg 2n_x + c_2 n_x - 6n_x n_\theta. \tag{4.31}$$

The above inequality is always satisfied since the left-hand side term in (4.31) is much more greater than the right-hand side term due to the existence of the terms corresponding to the EF complexity of the random number generation (c_1), and regularization (c_3). The most interesting result in this case is that the choice of resampling algorithm does not affect the inequality stated in (4.31).

In the second case, it is assumed that the measurement dimension n_y and the state dimension n_x are larger than the dimension of the parameters n_θ, i.e. $n_x,\ n_y \gg n_\theta$. Hence, to evaluate the EF complexity of the two methods (according to Table 4.5), the dominant terms are selected as the ones that are functions of N, n_x and/or n_y, while the terms that are dependent on only N and n_θ are ignored. Therefore, the EF complexity evaluations in this case become,

$$\begin{aligned} C(n_x, n_\theta, c_1, c_2, c_3, c_4, N) \approx N(&3n_x^2 + 2n_\theta n_y + 7n_y + 3n_x + c_1 n_x + c_2 n_x + c_3 n_x \\ &+ n_y \beta c_4), \end{aligned} \tag{4.32}$$

$$\acute{C}(n_x, n_\theta, c_1, c_2, c_3, N_0) \approx N_0(3n_x^2 + 6n_x n_\theta + (1 + c_1 + c_3)n_x + n_y).$$

Finally, from (4.32) to achieve the same EF complexity for two methods, the number of particles that are required in our proposed method is determined based on the number of particles that are required in the conventional method (the number of particles is set to N_0) and is set to

$$N = N_0\left(1 - \frac{c_2 n_x + 6n_y + 2n_x + 2n_\theta n_y + n_y \beta c_4 - 6n_\theta n_x}{3n_x^2 + 2n_\theta n_y + (3 + c_1 + c_2 + c_3)n_x + 7n_y + n_y \beta c_4}\right). \tag{4.33}$$

Applying the same analysis as the first case, to achieve less cost in the implementation of our proposed algorithm as compared to the conventional method, one must have

$$3n_x^2 + 2n_\theta n_y + (3 + c_1 + c_2 + c_3)n_x + 7n_y + n_y\beta c_4 \gg c_2 n_x + 6n_y + 2n_x + 2n_\theta n_y + n_y\beta c_4 - 6n_\theta n_x.$$

Assume that $n_x,\ n_y \gg n_\theta$, the above inequality can be simplified as follows

$$3n_x^2 + n_x + c_1 n_x + c_3 n_x + 7n_y \gg 6n_y - 6n_x n_\theta,$$

where this condition is always satisfied due to the high EF complexity of c_1 and c_3 and the evaluation of the implementation cost of our proposed algorithm is independent from choice of c_2 and c_4 (the ARMA model calculation). In simulation results that are presented in Section 4.5, it will be shown that under this circumstance where one employs the same number of particles in the two methods, the computational time (which is equivalent to the EF complexity) of our method when the ARMA structure with variable order is used (for implementing the observation forecasting scheme), would be comparable and in some cases significantly less than the conventional method with invariant particle weights.

4.4 Remaining Useful Life (RUL) Evaluation

The system model that is defined in (4.1) and (4.2) is suitable for model-based prognosis specifically in the case that the system health parameters (denoted by $\lambda(x_t)$) are affected by degradation damage through the fault vector θ_t. Performing a system health tracking and its evolution prediction to the future, the remaining useful life (RUL) of the system can be evaluated by taking into account the probability of failure distribution. This is accomplished according to a known

criterion on the maximum acceptable changes (critical bounds) of the system health parameters. In the implementation of the health tracking step, the filtering method presented in Chapter 3 is used. The developed and proposed prediction algorithm presented in Section 4.2 is also utilized for implementing the health prediction step.

It follows from Theorem 4.2 that as the prediction horizon is extended, the parameters prediction errors do increase accordingly. For evaluating the performance of our prediction scheme in terms of changes in the system health parameters, given that the true values of the parameters are assumed to be unknown, the percentage root mean square error (PRMSE) criterion is used for the estimated outputs instead of the parameters. This is computed according to [180]: $\text{PRMSE}_{y_j}(i) = 100\sqrt{\frac{1}{M}\sum_{j=t_{l_1}}^{j=t_{l_M}}(\frac{\hat{y}_{j,m+i}-y_{j,m+i}}{y_{j,m+i}})^2}$, where $\hat{y}_{j,m+i}$ denotes the predicted value of the j-th system output from the particle filters at time $m+i$ when m denotes the time instants at which the DLM model is updated and $m \in \{t_{l_1}, ..., t_{l_M}\}$ and M refers to the total number of DLM models in the entire observation trajectory, $y_{j,m+i}$ denotes the actual measured output (observation) at time $m+i$, and the mean is taken about all $m+i,\ m = t_{l_1}, ..., t_{l_M}$, time instants in the prediction horizon and $i = 1, ..., k$. The number of steps ahead prediction for the parameters is chosen from the PRMSE_{y_j} results of the outputs based on the considered acceptable threshold for each output. It is emphasized that the prediction error ϵ_{t+i}, which is used in the state and parameter prediction algorithm based on particle filters utilizing the forecasted observations obtained from the DLM models, is calculated as the difference between the forecasted observation and the predicted observation from the particle filter after estimating state and parameters as $\epsilon_{t+i} = \hat{Y}_{t+i|t} - \hat{y}_{t+i}$. Therefore, to evaluate the performance of the prediction scheme for states and parameters, $\hat{y}_{t+i} = h(\hat{x}_{t+i|t}, \hat{\theta}^{\mathrm{T}}_{t+i-1|t}\lambda(\hat{x}_{t+i|t}))$ is utilized which also includes the error due

to state and parameter prediction algorithm.

To calculate the RUL of the system when multiple health parameters are estimated we first denote and select $\theta_{\rm cr}^{(s)}, s = 1, ..., n_\theta$, as the critical value for the s-th health parameter. The following rule is now utilized for evaluating the RUL at time t [196], namely: RUL$(t) = t_j - t$, where t_j is $t_j = \min_{t_j}\{\hat{\theta}_{t_j}^{(s)} - \theta_{\rm cr}^{(s)} \geq 0,\ s = 1, ..., n_\theta\}$, i.e. the time at which the first parameter associated with the specific degradation damage reaches its critical value. Once reaching the critical value in one of the parameters, maintenance must be performed. Choosing the suitable value for $\theta_{\rm cr}^{(s)}$ is application specific and is generally determined based on the system performance and operator experience. In most applications and problems defining an exact value for the RUL is not possible. Therefore, an acceptable bound is considered as a confidence interval for the RUL prediction. The above procedure is estimating the RUL at most k-steps before the occurrence of the failure. As one gets closer to the failure time the RUL is approximated more accurately due to readjustments in the prediction scheme based on more recent observations.

4.5 Failure Prognosis of a Gas Turbine Engine

The application of our proposed PF-based prediction method for health monitoring and prognosis of a gas turbine engine is presented in this section. The approach is used for failure prognosis of the engine, when the system is assumed to be affected by health degradations phenomena. Our proposed and developed prediction scheme is demonstrated and illustrated to be capable of handling cases when non-Gaussian process noise is applied to the system. Moreover, the performance of our proposed state and parameter prediction scheme is evaluated and investi-

gated under general scenarios of degradations in both turbine and compressor components due to erosion and fouling phenomena.

4.5.1 Simulation Scenarios

In Chapter 3, the capabilities of the developed state and parameter estimation algorithm were shown in case of abrupt degradation damages that are modeled as multiplicative faults vector. In the present scenarios the engine is assumed to be subjected to degradation damages that are due to the compressor fouling and turbine erosion that cause gradual drifts in the system health parameters. A slowly changing linear degradation model is applied to the compressor health parameters during 1000 cycles of operation that cause a 3% drop in the compressor efficiency and 1.5% drop in its mass flow capacity, followed by a recovery through washing after 1000 cycles. A cycle refers to a single ground-air-ground (GAG) flight cycle [197], where the recorded cruise data related to each flight cycle is used for prognosis of the engine due to gradual degradations. The erosion degradation in the turbine is propagated through a quadratic evolution during the entire 1500 cycles of simulation that causes a 6% drop in the turbine efficiency and a 3% increase in its mass flow capacity. It should be noted that fouling and erosion degradation phenomena follow linear propagation pattern under low degradation index values, however as the degradation index increases they would not necessarily follow a linear profile [198].

A moving window of 150 observation data is used with $N = 150$ particles, where the window is moved according to the previously described Mahalanobis distance (M-distance) criterion after each batch of 10 observations become available ($s = 10$). We use the available

150 observation data points ($\tau = 150$) [1] in the observation forecasting scheme for developing the fixed-lag DLM model and the last 35 data points for validation of the DLM model ($\acute{s} = 35$). The size of sliding window in M-distance algorithm is chosen as $\acute{q} = 10$, and δ in the change detection algorithm (4.11) is set to $\delta = 1.5$. The parameters of the ARMA models related to each observation time window are adjusted based on the RLS method. The order of the time-varying ARMA structure is considered to be variable. For the AR part of the ARMA model the number of delays varies from 1 to 4, while the order of the MA part is fixed and set to 1. The criterion for choosing the order of ARMA model in each time window for generating DLM model, is applied based on Akaike information criterion or AIC [188]. To compare the results of the variable order time-varying ARMA model for forecasting the system observations with the fixed order ARMA model, an ARMA structure with the order of 4, i.e., AR(4) and MA(1) is also considered. The summary of the proposed prediction algorithm is shown in Table 4.6.

To show the effectiveness of our proposed prediction algorithm compared to the developed prediction algorithm with constant weights in particle filtering estimation method in the literature [25, 195], the $\text{PRMSE}_{y_j}(i)$ results for $i = 1, ..., 60$, and $j = 1, ..., 5$ corresponding to three methods, namely (a) the DLM-based particle filtering prediction with the constant model order for the observation forecasting (b) the variable DLM model order, and finally (c) particle filtering prediction with equally weighted particles without resampling ([25, 195]) are presented in Table 4.7 and in Figure 4.3 depicting the predicted observations.

From the results obtained it can be seen that our developed DLM-based particle filtering pre-

[1] The minimum number of data points needed for the convergence of the RLS algorithm for the estimation of the DLM model parameters, for this application is 150 data points.

Table 4.6: Summary of our proposed prediction algorithm

1. Collect 150 available observations.
2. For $j = 1 : n_y$
   (a) Construct the input-output database for the ARMA model,
   (b) Use the recent 150 data points for DLM models construction,
   (c) Find the best DLM structure according to ARMA model with the maximum order ARMA(4,1) that fits y_j, by using the AIC (Akaike Information Criterion) criterion [188]. In case of a constant model order, a fixed order model structure ARMA(4,1) is selected in this step,
   (d) Apply the RLS to obtain the coefficients of the ARMA model (DLM model parameters) recursively for the selected data set,
   (e) Predict the behavior of y_j for the next 60-steps ahead $(\hat{Y}_{j,t+k|t},\ k = 1, ..., 60)$ using the approximated DLM model.
3. Run the PF algorithm by applying the forecasted observation vector $\hat{Y}_{t+k|t}$, to obtain the k-step ahead prediction of states and parameters.
4. Move the observation window for the minimum $s \geq 10$ of the recently observed data points.
   (a) Calculate the M-distance for the recently received s observations and the last available 35 data points in the time window of 150 data points (which were used for DLM model construction),
   (b) If the M-distance exceeds the threshold go to Step 2 and re-calculate the DLM parameters,
   (c) If the M-distance does not exceed the threshold continue with the previously constructed DLM model for the 60-steps ahead prediction of the observations and go to Step 3.

diction algorithm (fixed order model (FOM) and variable order model (VOM)) for prediction of system states and health parameters outperforms the conventional PF-based prediction method in the sense of PRMSE_{y_j} values on the predicted observations. This is clearly shown when the prediction horizon is extended to 60-steps ahead. Note that the PRMSE_{y_j} results corresponding to the compressor and the turbine pressures are almost the same for the three methods.

The maximum step-ahead prediction horizon is chosen based on the mean PRMSE_{y_j} (percentage of the mean square error for the measurement outputs) as stated in Table 4.7, whereas beyond the chosen horizon $k = 60$ the error becomes unacceptable due to the deviation of $\mathbb{E}\{(y_{j,t+k} - \hat{Y}_{j,t+k|t})^2\}$ from $\iota\mu_{j,l},\ \iota = 0.01$ (according to Result 1). The PRMSE analysis is done based on the 48 different DLM models which are generated throughout the whole scenarios.

In the next subsection, the obtained parameter estimates are used to evaluate the remaining useful life (RUL) of the gas turbine engine in both fouling and erosion scenarios.

Table 4.7: PRMSE Results for the predicted outputs using Variable Order Model (VOM), Fixed Order Model (FOM) and the method without resampling (W/O).

Output	Max PRMSE			Mean PRMSE		
	VOM	FOM	W/O	VOM	FOM	W/O
T_{C}	1.5729	1.6043	1.6653	1.5160	1.5387	1.5946
P_{CC}	4.7397	4.8860	5.0903	4.0564	4.1217	4.3287
N	1.7625	1.7782	1.8481	1.7487	1.7577	1.8147
P_{NLT}	5.0001	5.2708	5.1541	2.9853	3.0272	3.0720
T_{T}	4.1064	4.1173	4.1971	4.0757	4.0872	4.1488

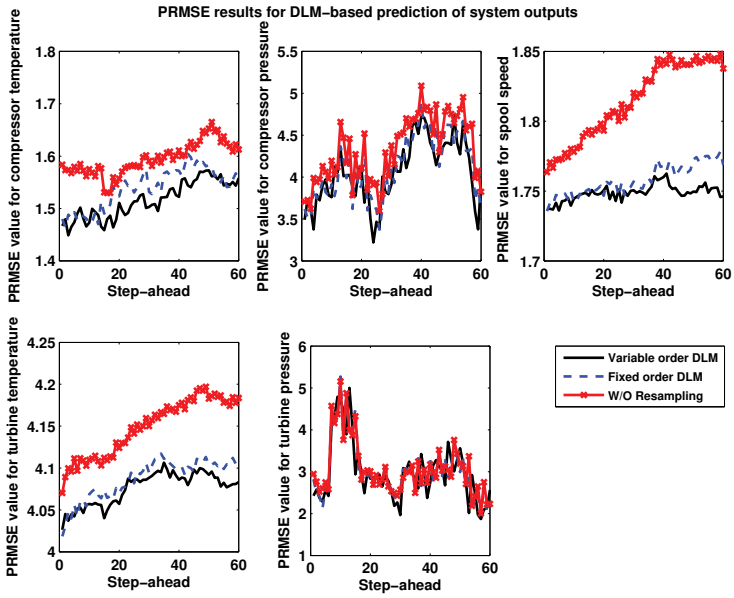


Figure 4.3: PRMSE values of the predicted outputs using Fixed Order Model (FOM), Variable Order Model (VOM) and the model without resampling (W/O).

4.5.2 RUL Prediction

For evaluating the RUL of the gas turbine engine, as per the criteria stated in Section 4.3, all the four parameters are considered for health evaluation and estimating of the system RUL in a prediction horizon of 60-steps ahead. The RUL is evaluated within the prediction windows starting from two subsequent flight cycles 854 and 877 (these two windows are corresponding to two updated DLM models) that include the ground truth failure cycle in their 60-steps ahead prediction horizon. The ground truth failure cycle for compressor health evaluation due to fouling phenomenon is located at the cycle 900. For turbine health evaluation, the windows starting from the cycles 1114 and 1138 are considered (this is due to the same reason as stated for the compressor failure time windows) while the ground truth failure cycle due to erosion is located at the cycle 1162. The predicted health parameters and their fault parameters along with

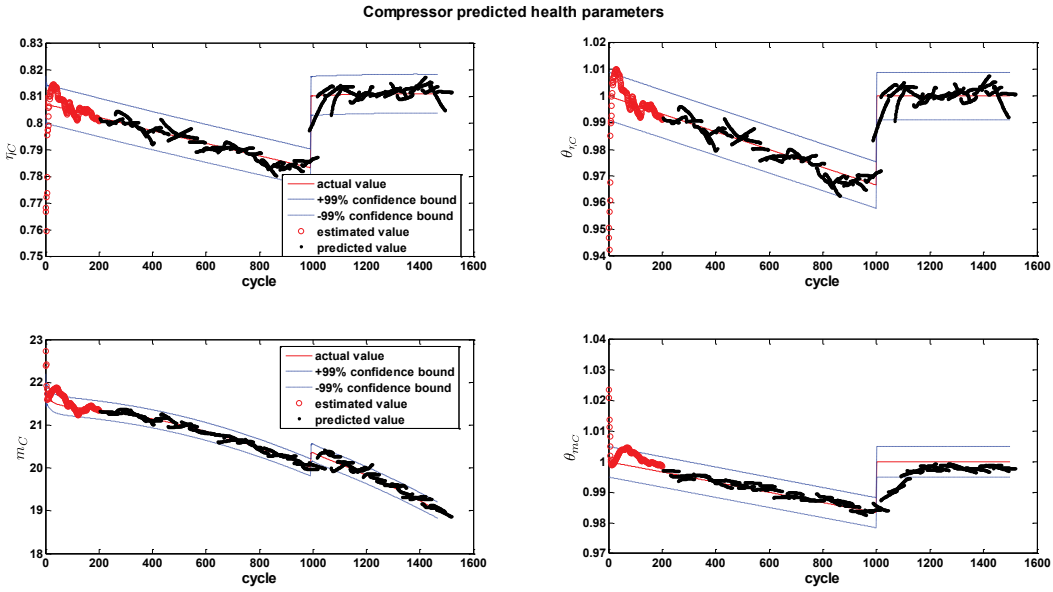


Figure 4.4: The predicted compressor health parameters and their related fault vectors.

the actual degradations are depicted in Figures 4.4 and 4.5 for the compressor and the turbine, respectively.

The $\pm 1\%$ confidence intervals for the compressor efficiency and the mass flow capacity indicate that the two predicted parameters, are located within the confidence intervals for most of the time in the simulations. For the turbine parameters these confidence intervals are found to be $\pm 1\%$ for the turbine efficiency and $\pm 2\%$ for the turbine mass flow capacity. It should be noted that the first 150 data points in these plots are corresponding to the estimated health parameters (one-step ahead prediction) which are calculated in the presence of the system observations. As it was mentioned earlier, as the new observations become available the k-step ahead prediction of the system states and parameters is performed based on the recent received observations. Therefore, regarding to each prediction window for the health parameters in Figures 4.4, and 4.5

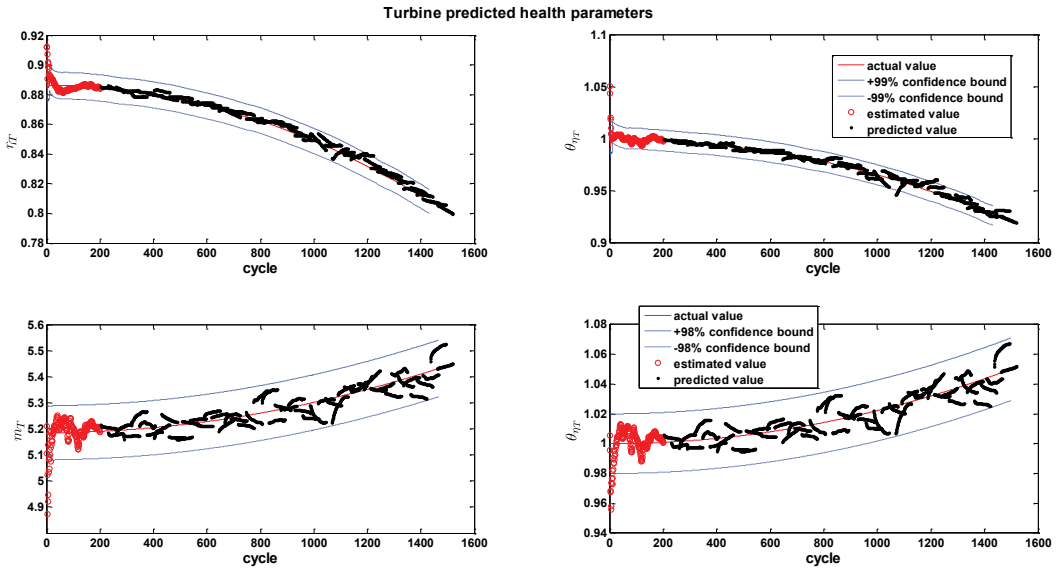


Figure 4.5: The predicted turbine health parameters and their related fault vectors.

there are 150 data points of their estimated values which are only shown for the first prediction window and they are removed from the figures for the rest of prediction windows to have more clear figures for distinguishing the predicted parameters from the estimated ones.

The critical values for the parameter degradations are considered to be a 3% decrease in the compressor efficiency [198] along with a 1.5% decrease in the mass flow capacity due to the fouling phenomenon. This implies that after achieving this level of deficiency in any of these parameters, the compressor must be taken for a wash up. On the other hand, the critical values for detecting erosion in the gas turbine are considered to be a 6% decrease in the turbine efficiency and a 3% increase in the mass flow capacity.

Fouling Scenario

Corresponding to the obtained prediction results for the fault parameters of the compressor health parameters, two subsequent time windows are considered. These time windows are starting from flight cycles 854 and 877, respectively. For determination of the flight cycle at which the maximum probability of failure is occurred, a probabilistic analysis method is proposed. In this method the distribution of the predicted data related to the system health parameters (i.e. in compressor mass flow capacity and efficiency in fouling scenario and turbine mass flow capacity and efficiency in erosion scenario) in each time window of 60-step ahead horizon, is fitted to a Gaussian distribution. Hence, the amount of changes in the mean value of this distribution through consequence prediction windows, determine the changes in its related health parameter. Consider the health indicator vector in fouling scenario as $\mathrm{HI}_{t+1:t+k} = \begin{pmatrix} \hat{\theta}_{\eta_{\mathrm{C}t+1|t}} & \cdots & \hat{\theta}_{\eta_{\mathrm{C}t+k|t}} \\ \hat{\theta}_{m_{\mathrm{C}t+1|t}} & \cdots & \hat{\theta}_{m_{\mathrm{C}t+k|t}} \end{pmatrix}$, where its mean value is presented as $\mathrm{mean}(\mathrm{HI_{t+1:t+k}}) = (\mu_{\eta_{\mathrm{C}}},\ \mu_{m_{\mathrm{C}}})^{\mathrm{T}}$, and its variance in denoted by $\mathrm{var}(\mathrm{HI_{t+1:t+k}}) = \begin{pmatrix} \delta^2_{\theta_{\eta_{\mathrm{C}}}} & 0 \\ 0 & \delta^2_{\theta_{m_{\mathrm{C}}}} \end{pmatrix}$. Hence, the Gaussian probability function value corresponding to each health indicator (compressor health parameters) is calculated as

$$\mathrm{Gpdf}(\hat{\theta}_{\eta_{\mathrm{C}t+i|t}}|\mu_{\eta_{\mathrm{C}}}, \delta^2_{\eta_{\mathrm{C}}}) = \frac{1}{\delta_{\eta_{\mathrm{C}}}\sqrt{2\pi}} e^{-\frac{(\hat{\theta}_{\eta_{\mathrm{C}t+i|t}} - \mu_{\eta_{\mathrm{C}}})^2}{2\delta^2_{\theta_{\eta_{\mathrm{C}}}}}},$$

$$\mathrm{Gpdf}(\hat{\theta}_{m_{\mathrm{C}t+i|t}}|\mu_{m_{\mathrm{C}}}, \delta^2_{m_{\mathrm{C}}}) = \frac{1}{\delta_{m_{\mathrm{C}}}\sqrt{2\pi}} e^{-\frac{(\hat{\theta}_{m_{\mathrm{C}t+i|t}} - \mu_{m_{\mathrm{C}}})^2}{2\delta^2_{\theta_{m_{\mathrm{C}}}}}}.$$

Moreover, the failure cycle corresponding to each health indicator, is calculated as the cycle at which the predicted health parameter exceeds the 99.5% confidence bound around the critical value of that health parameter. Therefore, considering that the fouling phenomenon is identified

by its decreasing effect on both compressor health parameters, the upper bound on the critical values of the compressor health parameters is considered as the criteria for determination of the system failure cycle due to fouling. Because, the failure cycle according to this rule can be more than one cycle, the one corresponding to the maximum probability density function is considered as the failure cycle due to the assumed health parameter which is calculated as follows

$$\text{FailureCycle}_{\eta_{\text{C}}} = t + j, \text{ such that } \hat{\theta}_{\eta_{\text{C}}t+j|t} \leq (1.05)\theta^{\text{cr}}_{\eta_{\text{C}}} \text{ and } t + j = \text{argmax}_{j=1}^{k}\text{Gpdf}(\hat{\theta}_{\eta_{\text{C}}t+j|t}),$$

$$\text{FailureCycle}_{m_{\text{C}}} = t + l, \text{ such that } \hat{\theta}_{m_{\text{C}}t+l|t} \leq (1.05)\theta^{\text{cr}}_{m_{\text{C}}} \text{ and } t + l = \text{argmax}_{l=1}^{k}\text{Gpdf}(\hat{\theta}_{m_{\text{C}}t+l|t}),$$

where $\theta^{\text{cr}}_{\eta_{\text{C}}}$ and $\theta^{\text{cr}}_{m_{\text{C}}}$ are the critical values of the compressor efficiency and mass flow capacity, respectively. Finally, the failure cycle of the system is considered as the minimum of $\text{FailureCycle}_{\eta_{\text{C}}}$, and $\text{FailureCycle}_{m_{\text{C}}}$, as $\text{FailureCycle} = \min(\text{FailureCycle}_{\eta_{\text{C}}}, \text{FailureCycle}_{m_{\text{C}}})$.

Performing the probabilistic study on the distribution of the predicted data in the considered two time windows, the results summarized in Table 4.8 and Figures 4.6, 4.7 are obtained for the prediction performed from all three methods as stated before. Assume that the ground truth failure cycle, which is the flight cycle 900, corresponds to $\theta^{\text{cr}}_{\eta_{\text{C}}} = 0.97$, and $\theta^{\text{cr}}_{m_{\text{C}}} = 0.985$.

In Table 4.8, the mean and standard deviation of the distribution for the 60-step ahead predicted data (when they are fitted to a Gaussian distribution), are shown which are based on the results depicted in Figures 4.6, and 4.7. In these figures the changes in the distributions of the compressor fault parameters for the two considered time windows which are close to the failure cycle, are presented for the three prediction methods. In addition to the distributions related to the changes in the fault vector of the mentioned health parameter, the cycles at which the cal-

culated probabilities are achieved, are also plotted in Figure 4.6 for the compressor efficiency and Figure 4.7 for the compressor mass flow capacity. In the cases that these distributions are reliable Gaussian distributions, the failure cycle can be predicted from their results as the cycle at which the probability distribution reaches its maximum value.

The presented results in Table 4.8 guarantee that as one gets closer to the failure cycle, the means of the distributions related to the prediction results for both compressor health parameters faults are located within the 99.5% confidence bound around the actual critical values. However, the results corresponding to the conventional method (W/O) do not follow the correct direction for the degradation propagation. While the mean of the distribution for the predicted values is located around 0.9674 in the first time window, as the window moves towards closer points around the failure cycle, the mean of the distribution increases to 0.9705 which is not correct because the fouling phenomenon causes gradual degradation in the compressor health parameters which has decreasing effect on the compressor health parameters (not decreasing in one window and increasing in the consequent one). Therefore, the RUL prediction for the compressor fouling scenario from this method cannot give us a reliable result in terms of the failure cycle. As one can see from the results presented in Figure 4.6, the maximum probability of failure is achieved at several cycles (instead of one cycle). Considering the probability distributions around the related flight cycles presented in Figure 4.6, the failure cycle can be predicted from both windows for VOM and FOM, whereas for W/O only from the first time window one can obtain a reliable prediction for the failure cycle.

Similar to the results of the compressor efficiency fault vector, the corresponding results for

Table 4.8: Predicted values distributions due to compressor fouling

Parameter	Prediction from cycle 854			Prediction from cycle 877		within $\pm 99.5\%$ of critical value
	method	mean	$\pm$ std	mean	$\pm$ std	
θ_{η_C}	VOM	0.967	0.0003	0.967	0.0009	yes
	FOM	0.966	0.0006	0.967	0.0017	yes
	W/O	0.967	0.0028	0.970	0.0009	yes
θ_{m_C}	VOM	0.986	0.0004	0.985	0.0001	yes
	FOM	0.985	0.0004	0.985	0.0001	yes
	W/O	0.984	0.0018	0.982	0.0002	yes

the compressor mass flow capacity are also located in the 99.5% confidence bound around the actual critical value. Moreover, for predicting the failure cycle, from the results presented in Figure 4.7 one can use the distributions related to VOM from both windows for the purpose of failure cycle prediction. However, W/O is not suitable for failure cycle prediction from the first window due to the unreliable probability distribution in this time window which causes several cycles with maximum probability of failure values, and FOM is not suitable for failure cycle prediction from the second window for the same reason.

Finally, the predicted RUL from the starting point of the two considered time windows is calculated as the difference between the start cycle in the considered time window and the predicted failure cycle in that window for each health parameter separately as shown in Table 4.9. The predicted failure cycle is calculated as the cycle at which the maximum probability distribution value in the related time window is reached (for both fault vectors of compressor and turbine). Assume that in the first time window starting at 854 the actual RUL is 46 ($900 - 854 = 46$), and in the second time window starting at 877 the actual RUL is 23 ($900 - 877 = 23$). The RUL error is also indicated in Table 4.9 which is calculated as the difference between the actual RUL and the predicted RUL from different methods.

From the presented results in this table, it is concluded that VOM is able to predict the

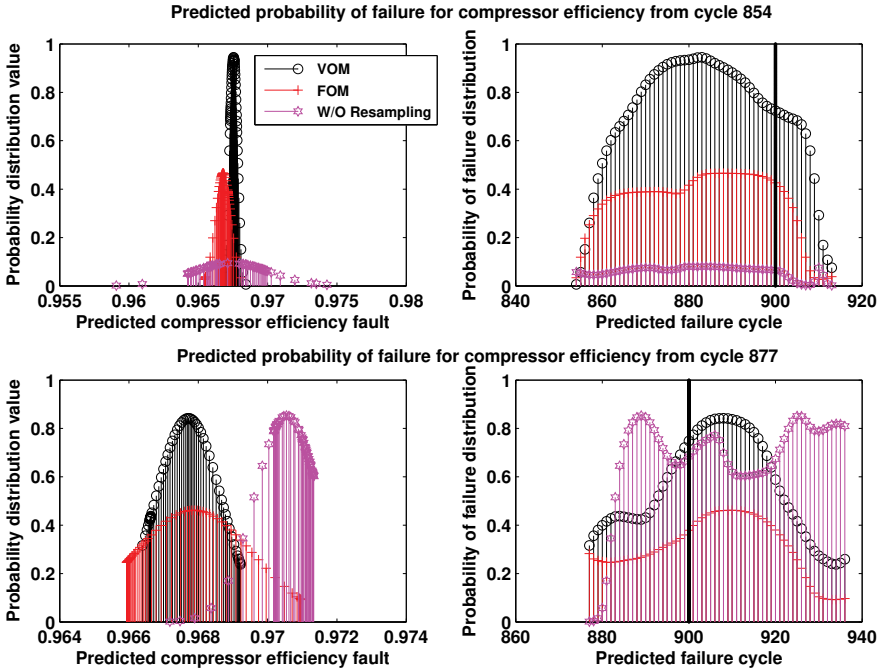


Figure 4.6: Predicted probability of the failure for the compressor efficiency from two prediction windows.

RUL from two time windows for both compressor health parameters and as the time window moves towards the actual failure cycle, the RUL prediction becomes more accurate such that in the second time window the RUL can be predicted within ± 5 cycles around the actual RUL. Moreover, FOM method can determine the RUL based on both health parameter in the first time window within ± 14 cycles around the actual RUL, but in the second time window only one of the parameters can be used for RUL prediction. However, the prediction method based on W/O does not have enough accuracy for RUL prediction based on both health parameters of the system from the two time windows. The criterion for evaluating the failure cycle based on multiple system health parameters is according to parameter that predicts an earlier failure cycle (according to the discussion in Section 4.4).

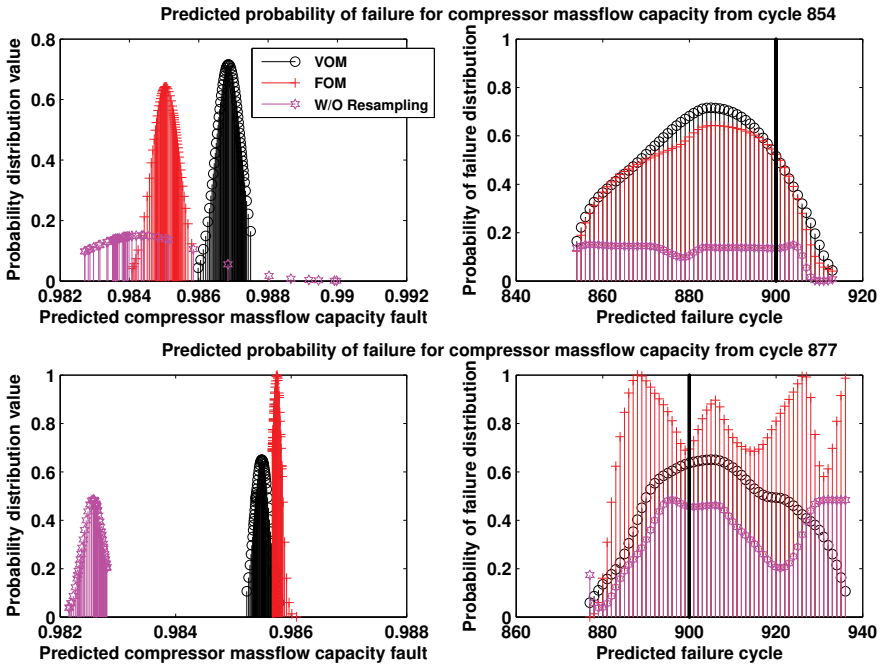


Figure 4.7: Predicted probability of the failure for the compressor mass flow capacity from two prediction windows.

Table 4.9: RUL prediction in compressor fouling scenario

Parameter	Prediction from cycle 854				Prediction from cycle 877		
	method	failure cycle	predicted RUL	RUL error	failure cycle	predicted RUL	RUL error
θ_{η_C}	VOM	883	29	+17	908	31	-8
	FOM	888	34	+12	910	33	-10
	W/O	880	26	+20	-	-	-
θ_{m_C}	VOM	885	31	+15	905	28	-5
	FOM	886	32	+14	-	-	-
	W/O	-	-	-	896	19	+4

Erosion scenario

For the erosion scenario, the accomplished prediction results for the fault parameters of the turbine health parameters from two subsequent time windows close to the failure cycle are considered. These time windows are starting from flight cycles 1114 and 1138, respectively. The health indicator vector in this scenario consists of the turbine efficiency and mass flow capacity $\mathrm{HI}_{t+1:t+k} = \begin{pmatrix} \hat{\theta}_{\eta_{\mathrm{T}}t+1|t} & \cdots & \hat{\theta}_{\eta_{\mathrm{T}}t+k|t} \\ \hat{\theta}_{m_{\mathrm{T}}t+1|t} & \cdots & \hat{\theta}_{m_{\mathrm{T}}t+k|t} \end{pmatrix}$. The probability analysis similar to the fouling scenario is also performed for this scenario assuming the turbine health parameters as the system health indicators. However, considering that the erosion phenomenon is identified by its decreasing effect on the turbine efficiency and increasing effect on its mass flow capacity, the upper bound on the critical value of the turbine efficiency and the lower bound on the critical value of the turbine mass flow capacity are considered as the criteria for determination of the system failure cycle due to erosion. Hence, in the calculation of failure cycle in this scenario the following criteria is considered

$$\mathrm{FailureCycle}_{\eta_{\mathrm{T}}} = t + j, \text{ such that } \hat{\theta}_{\eta_{\mathrm{T}}t+j|t} \leq (1.05)\theta^{\mathrm{cr}}_{\eta_{\mathrm{T}}} \text{ and } t + j = \mathrm{argmax}_{j=1}^{j=k} \mathrm{Gpdf}(\hat{\theta}_{\eta_{\mathrm{T}}t+j|t}),$$

$$\mathrm{FailureCycle}_{m_{\mathrm{T}}} = t + l, \text{ such that } \hat{\theta}_{m_{\mathrm{T}}t+l|t} \geq (0.995)\theta^{\mathrm{cr}}_{m_{\mathrm{T}}} \text{ and } t + l = \mathrm{argmax}_{l=1}^{l=k} \mathrm{Gpdf}(\hat{\theta}_{m_{\mathrm{T}}t+l|t}),$$

where $\theta^{\mathrm{cr}}_{\eta_{\mathrm{T}}}$ and $\theta^{\mathrm{cr}}_{\eta_{\mathrm{T}}}$ are the critical values of the turbine efficiency and mass flow capacity, respectively. The results summarized in Table 4.10 and Figures 4.8, 4.9 are obtained based on the probabilistic analysis of the prediction results. Assume that the ground truth failure cycle, which is the flight cycle 1162, corresponds to $\theta^{\mathrm{cr}}_{\eta_{\mathrm{T}}} = 0.94$, and $\theta^{\mathrm{cr}}_{m_{\mathrm{T}}} = 1.03$. The presented results in Table 4.10, show that from both time windows the means of the distributions related to the turbine faults parameters are located within the 99.5% confidence bound around the critical value

Table 4.10: Predicted values distributions due to turbine erosion

Parameter	Prediction from cycle 1114			Prediction from cycle 1138		within $\pm 99.5\%$ of critical value
	method	mean	$\pm$ std	mean	$\pm$ std	
θ_{η_T}	VOM	0.957	0.0011	0.957	0.0017	yes
	FOM	0.955	0.0002	0.961	0.0028	yes
	W/O	0.956	0.0015	0.959	0.0012	yes
θ_{m_T}	VOM	1.033	$9.013e^{-4}$	1.033	0.0015	yes
	FOM	1.029	$9.486e^{-4}$	1.045	0.0049	no
	W/O	1.029	$9.033e^{-4}$	1.040	0.0037	no

for the turbine efficiency fault parameter. However, FOM and W/O do not result in prediction results distributions within 99.5% confidence bounds. According to the results indicated in Figures 4.8 and 4.9, the VOM method can be used for failure cycle prediction from both windows for both turbine health parameters, whereas FOM cannot be used for this purpose from the first time window for the turbine efficiency and W/O can not be used from the first window for the turbine mass flow capacity.

Consequently, the predicted RUL from the starting point of the two considered time windows in erosion scenario, is calculated and presented in Table 4.11. Since the actual failure cycle is located at 1162, the actual RUL in the first time window starting at 1114 is 48 ($1162 - 1114 = 48$), and in the second time window starting at 1138 the actual RUL is 24 ($1162 - 1138 = 24$). The RUL error is also indicated in this table. From the results presented in Table 4.11 one can find that RUL prediction based on FOM and W/O according to the variations in both health parameters of the turbine is possible only from the second window, i.e. as one gets closer to the failure cycle, whereas VOM is able to predict RUL based on both turbine health parameters from both time windows. However, as the time window moves towards the actual failure cycle, the RUL prediction based on VOM becomes more accurate such that in the second time window RUL can be predicted within ± 5 cycles from the actual failure cycle. Moreover, FOM and W/O

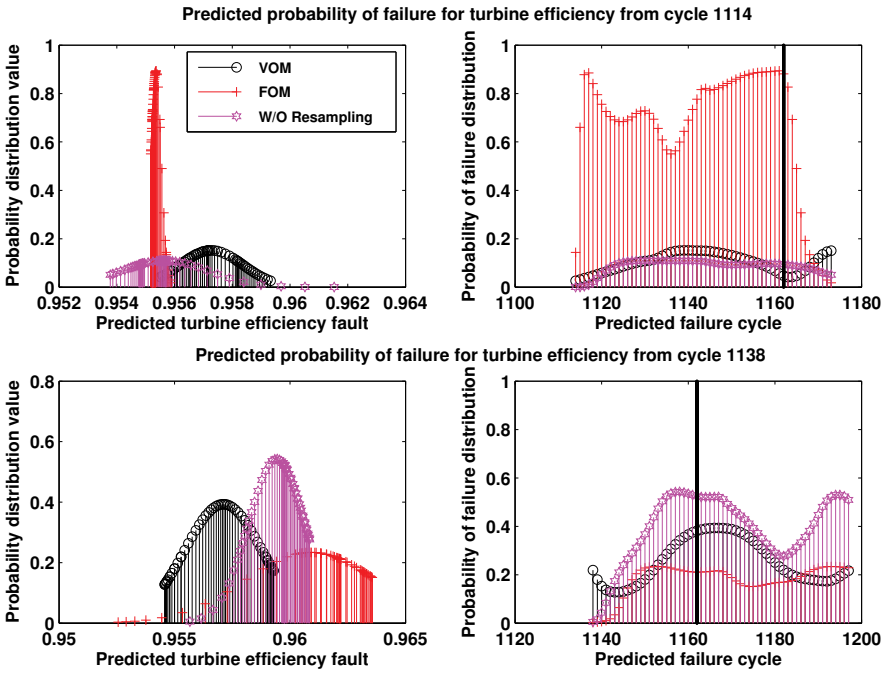


Figure 4.8: Predicted probability of the failure for the turbine efficiency from two prediction windows.

can determine the RUL based on both turbine health parameters only from the second window within ± 9, and ± 4 cycles from the actual failure cycle, respectively.

Table 4.11: RUL prediction in turbine erosion scenario

Parameter	Prediction from cycle 1114				Prediction from cycle 1138		
	method	failure cycle	predicted RUL	RUL error	failure cycle	predicted RUL	RUL error
θ_{η_T}	VOM	1139	25	+23	1167	29	-5
	FOM	-	-	-	1153	15	+9
	W/O	1134	20	+28	1158	20	+4
θ_{m_T}	VOM	1170	56	-8	1170	32	-8
	FOM	1147	33	+15	1156	18	+6
	W/O	-	-	-	1162	24	0

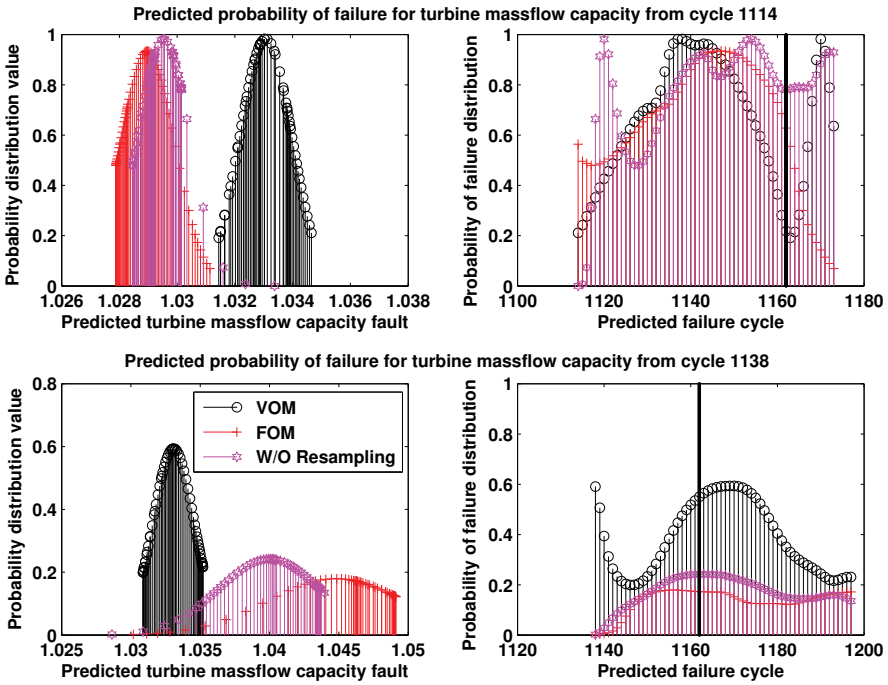


Figure 4.9: Predicted probability of the failure for the turbine mass flow capacity from two prediction windows.

4.5.3 RUL Performance Analysis with Different DLM Parameters

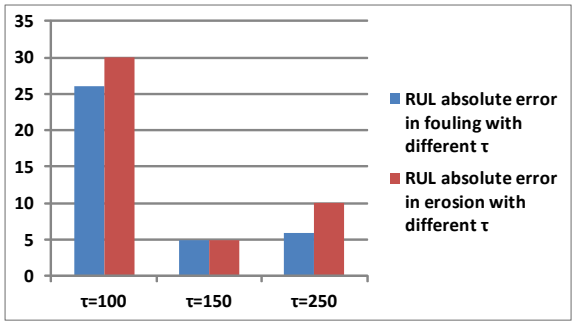
In this subsection, the performance of RUL prediction is evaluated when different parameters in the DLM are varying around their designed values for the prognosis of the gas turbine engine in both fouling and erosion scenarios. It is pointed out that the designed parameters are selected as: the DLM fixed lag window size $\tau = 150$ where 35 data points are used in the validation step ($\acute{s} = 35$), the sliding window size in M-distance algorithm is $\acute{q} = 10$, the number of recent available observation at each time instant is selected as $s = 10$, and the threshold in the M-distance for change detection is $\delta = 1.5$. To evaluate the performance of prognosis in terms of RUL prediction error, several scenarios are performed where in each scenario only one of the above mentioned parameters has been changed while the rest of the DLM tuning parameters are considered to be fixed at their designed values. Moreover, the parameter $\iota = 0.01$ is selected fixed for all scenarios for obtaining the maximum k-step ahead prediction horizon that satisfy the desired threshold on the PRSME$_{y_j}$ value (based on ι) in all the considered scenarios which results in a maximum 30-step ahead prediction horizon for all scenarios. Therefore, all the comparisons among different conditions are done in a prediction window 30-step before the failure occurrence.

In Figures 4.10 (a)-(d) the effects of changes of the DLM parameters in the absolute error of RUL prediction are plotted for two more values around the designed value of the selected parameter. The summarized results are obtained from a window located in the 30-step before reaching the ground truth failure cycle in each scenario. The presented results in Figure 4.10 (a) shows that decreasing the number of data points used in constructing the DLM model (τ)

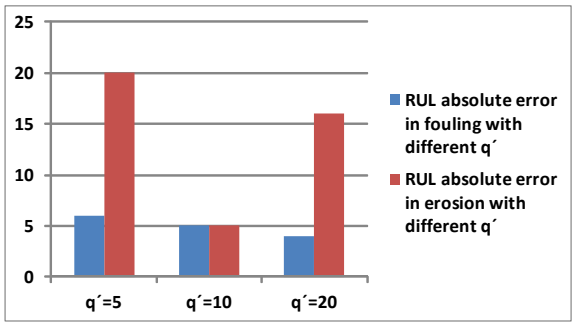
results in higher error in the RUL prediction in both fouling and erosion scenarios. From Figure 4.10 (b) one can find out that whereas changes in the sliding window $\acute{q}$ do not affect the RUL error in fouling scenario significantly, the error due to erosion scenario is rather high when $\acute{q}$ is selected as a small or high value compared to the number of recent available data in each time instant, s. From the presented graphs in Figure 4.10 (c) one can also conclude that over increasing/decreasing the length of s can affect the RUL error significantly in both scenarios. Finally, the results related to changes in the threshold which is utilized in the M-distance algorithm for change detection as presented in Figure 4.10 (d) show that selecting smaller values for δ can lead into erroneous RUL prediction results.

4.5.4 Time Complexity Analysis of the Prediction Scheme

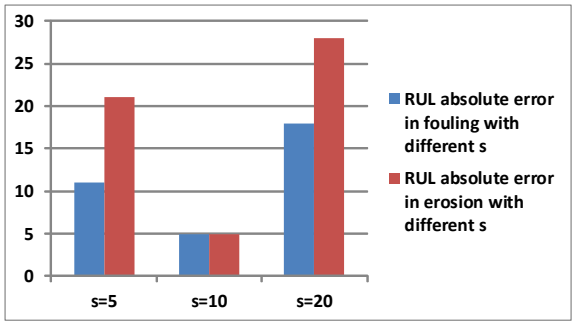
To compare the efficiency of our developed prediction schemes based on the variable order DLM (VOM) and the fixed order DLM (FOM) models, with a conventional method based on augmented state and parameter estimation algorithm according to reguralized particle filters and without performing resampling [25], the execution times of these schemes are obtained for each iteration of prediction step. Assuming that the computational complexity of the prediction step is proportional to the EF complexity of the algorithms [170]. These metrics are estimated by using an Intel Xeon CPU E31230, 3.2GHz processor with 16GB memory. Therefore, the time complexity as a measure of time (in seconds) that are required to execute the algorithms for the best scenario (that is the minimum execution time), the average scenario (that is the average execution time), and the worst scenario (that is the maximum execution time) are obtained for the VOM, the FOM and the W/O schemes in Table 4.12. In these methods the number of



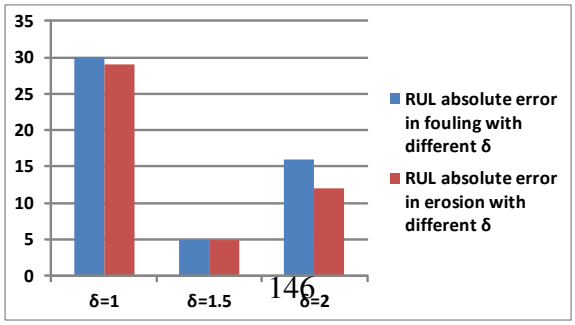
(a) RUL error with different τ values.



(b) RUL error with different $\acute{q}$ values.



(c) RUL error with different s values.



(d) RUL error with different δ values.

Figure 4.10: RUL performance using different DLM parameters.

particles are considered to be the same and is set to $N_0 = N = 150$.

From the shown results it can be concluded that the time required for execution of the VOM is less than that of the FOM and W/O. It is further concluded that even with equal number of particles when the dimensions of the states and measurements are comparable with the dimension of the parameters ($n_\theta = n_x = 4,\ n_y = 5$), the resampling algorithm that we have used (namely, the residual resampling) did not increase the time complexity (which is equivalent to the EF complexity) of our developed VOM algorithm compared to a conventional method without performing resampling.

Table 4.12: Computational Time Corresponding to the VOM, FOM and W/O Algorithms in seconds.

Model	Best Scenario	Average Scenario	Worst Scenario
VOM	88	136.0208	190
FOM	94.0146	138.9729	194.5736
W/O	92.4846	138.2435	193.9752

4.5.5 Prognosis Online Performance Assessment

In this subsection, the performance of our developed particle filtering based prediction scheme for prognosis of a gas turbine engine is evaluated according to the introduced online prognosis assessment metrics in [181, 182]. The obtained results are also compared with the other two presented prognosis approaches in [25] with the same performance metric measures. These metrics are RUL online precision index (RUL-OPI), RUL accuracy precision index (RUL-API), and RUL online steadiness index (RUL-OSI) as elaborated below.

RUL-OPI: Considers the 95% confidence interval computed at time t denoted by CI_t,when compared to the RUL. According to this metric the more data the algorithm processes, the more precise the prognosis results should be and in the best case it should be close to one. The RUL-OSI is denoted by $I_1(t)$ for all $t \in [1, \mathrm{E}_t\{TOF\}],\ t \in \mathbb{N}$ as $I_1(t) = e^{-(\frac{\sup(CI)_t - \inf(CI)_t}{\mathrm{E}_t\{RUL\}})}$, where $\mathrm{E}_t\{RUL\}$ is the estimate of the expectation of the system RUL at time t, and where $\mathrm{E}_t\{TOF\}$ is the estimate of the expectation of the system time of failure at time t.

RUL-API: Represents the error of time of failure estimates relative to the width of the corresponding 95% confidence interval, CI_t and denoted by $I_2(t)$. This metric penalizes whenever the $\mathrm{E}_t\{TOF\}$ is greater than the ground truth failure cycle, i.e. whenever actual failure happens before the expected time and is introduced as $I_2(t) = e^{-(\frac{\mathrm{GroundTruth}\{TOF\} - \mathrm{E}_t\{TOF\}}{\sup(CI)_t - \inf(CI)_t})}$. The accurate prognosis results correspond to the values of $I_2(t)$ such that $0 \leq 1 - I_2(t) \leq \varsigma$, where ς is a small positive constant.

RUL-OSI: Considers the current TOF expectation which is calculated given the measurements at time t. According to this metric, the more data the algorithm processes, the steadier the prognostic result will be. It is denoted by $I_3(t)$ and calculated as
$I_3(t) = \sqrt{\mathrm{Var}(\mathrm{E}_t\{TOF\})},\ I_3(t) \geq 0, \forall t \in \mathbb{N}$.

Next, utilizing the aforementioned prognosis indices, the performance of our developed prediction method with variable order model structure (VOM) is compared with other two methods from the literature [25], i.e. the prediction method based on particle filters with invariant weights for future propagation of the particles with the standard structure for particle filter implementation (PF method), and the particle filtering-based prediction method using the regularization

(with optimal kernel density) of particles with invariant weights and applying outer feedback loop for online adjustment of hyper parameters (RPF method). The summarized prognosis results according to the above mentioned indices are presented in Figures 4.11, and 4.12 for fouling and erosion scenarios, respectively assuming the compressor and turbine efficiency as the health indicators (according to the mentioned criterion in Section 4.4).

For the fouling scenario, as presented in Figure 4.11, the analysis results are plotted from the cycle 854 to cycle 900 which is the failure cycle due to fouling. In this interval, the VOM algorithm is updated two times, therefore the steps in the curves related to our developed VOM method are because of this update in the DLM models of the observation forecasting part of the algorithm. From the presented results in Figure 4.11, VOM can maintain RUL-OPI more than 0.9 for all the analysis in the considered time window. The RPF method with outer feedback loop follows very close results to VOM instead of some spikes. On the other hand, the standard PF method shows the least precise results by decreasing to near 0.8. The RUL-API results for VOM and RBF even in the most conservative case, lie below 2, whereas PF has continuously increasing RUL-API curve. Finally, the RUL-OSI index results for all three methods are located in the same range. It should be noted that the fluctuations in the RPF methods are because of updates in the outer feedback loop which enables the RPF algorithm to cope with the parameter changes. The RUL-OSI for PF and RPF methods is calculated by considering a moving window of size 20 of the predicted results.

In the erosion scenario, the presented results in Figure 4.12 show that all three performance curves which are calculated based on the results of PF and RPF method follow similar patterns

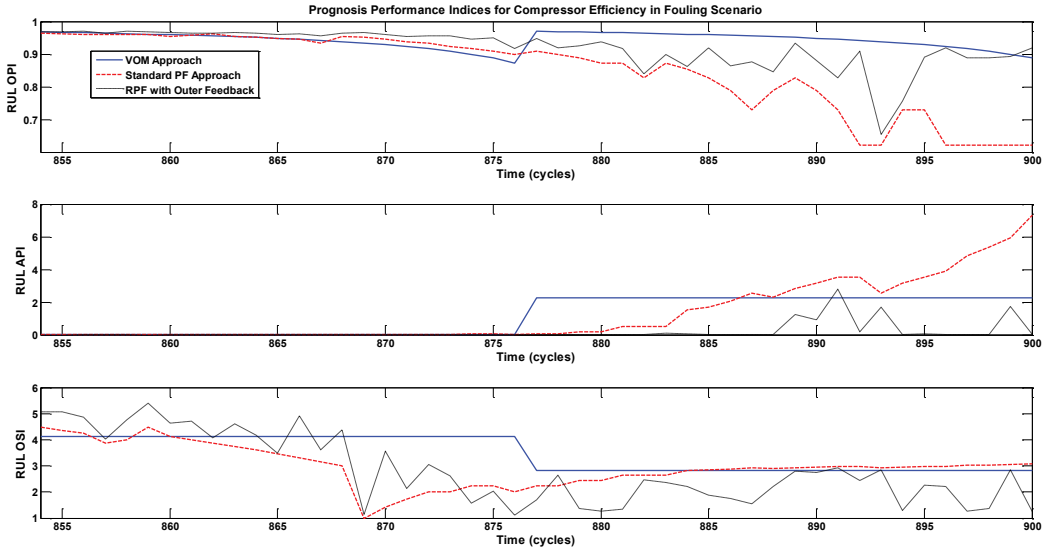


Figure 4.11: Prognosis performance indices for compressor efficiency in fouling scenario.

and even the outer feedback loop is not able to cope with the changes in the turbine efficiency more effectively. The main reason can be related to the fact that outer feedback adjustment loop can compensate the changes in the parameter for a limited range of parameter variations and if the parameter changes exceed this range the outer loop is not able to compensate the changes perfectly. However, our developed VOM method shows compromising results from all three prognosis performance metrics for erosion scenario as well as fouling scenario.

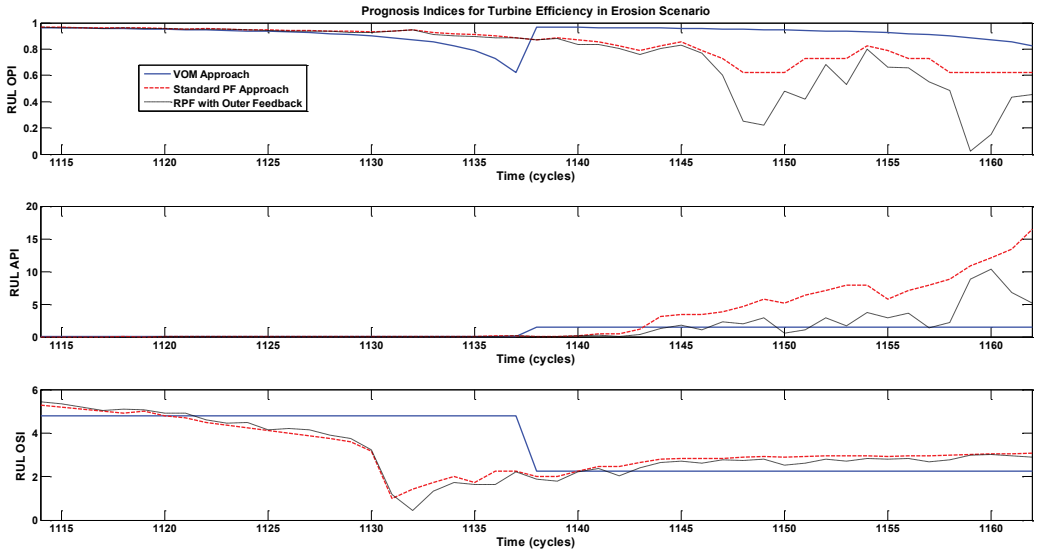


Figure 4.12: Prognosis performance indices for turbine efficiency in erosion scenario.

4.6 Conclusions

In this chapter, conventional particle filtering schemes are extended to predict the future behavior of a nonlinear dynamical system states and parameters by utilizing the observation forecasting concept and using time-series methods. This observation forecasting scheme is developed based on fixed/variable order DLM models which are adjusting online according to an adaptive external adjustment loop. It is shown that despite its improved performance, our proposed scheme does not impose additional computational complexity when compared to the other available methods in the literature. The developed model is applied for the purpose of failure prognosis in a gas turbine engine. The results for the remaining useful life (RUL) prediction demonstrate and illustrate the acceptable performance of our developed scheme.

Chapter 5

Prognosis and Health Monitoring of Gas Turbine Engines using a Hybrid Scheme through Integration of Particle Filters and Neural Networks

In this chapter, *which is investigated in collaboration with Dr. Baniamerian*, a hybrid architecture is proposed to develop prognosis and health monitoring methodologies for nonlinear systems by integration of model-based and computationally intelligent-based techniques. In our proposed prognosis algorithm in this chapter the propagation of the health indicator is estimated and predicted through particle filters where the intelligent based methods are utilized

as add-ons to enhance the accuracy of the overall particle filtering based method by taking into account the effect of hidden damages which are not modeled in the system dynamics based on their effect on the system on line observations. Hence, significantly less historical data in this method is required without need for pre-training the system for different fault scenarios. Moreover, due to explicitly utilizing a mathematical model in our proposed hybrid methodology, the required data for training purposes would be lower than those required for only a computational intelligent-based method [199].

In our proposed approach, the neural networks are continuously adjusted according to the most recent available observations through a sliding window process. This ensures that the networks are adapted in cases when maintenance actions are performed on the system and the process should then be considered as starting its operation from a new healthy condition. As demonstrated subsequently through extensive simulation case studies, our proposed hybrid methodology remains robust when implemented with different neural networks that are utilized in the observation forecasting scheme.

In this chapter, we utilize three neural networks to predict the observation profile of the system for a future time horizon. These predictions are then utilized in our particle filter-based prognosis method for performing the health monitoring task. Consequently, the proposed hybrid approach enables one to select the appropriate signatures critical for determining the remaining useful life (RUL) of the system and its components based on the system hidden states/parameters that are made possible with integration of model-based and computationally intelligent-based neural networks. Finally, our proposed methodology is utilized in prognosis of a gas turbine en-

gine that is affected by degradation damages due to compressor fouling and turbine erosion. The robustness and performance capabilities of our proposed methodology are investigated when the RUL of the system is estimated under three cases, namely when the observation forecasting scheme is implemented by utilizing three different types of neural networks. It has been shown that our hybrid framework is capable of dealing with scenarios when maintenance actions have also been performed on the gas turbine engine.

The main contribution of this chapter is now summarized as below:

1. Development of a hybrid framework for health monitoring and prognosis methodology by extending the previously developed particle filtering-based prediction strategy and incorporating nonlinear time-series forecasting methods based on neural networks as opposed to linear time-series methodologies.

The remainder of this chapter is organized as follows. The problem statement is provided in Section 5.1. The proposed hybrid prediction strategy that is developed in Section 5.2 is based on a nonlinear observation forecasting scheme that integrates neural networks with particle filters. The details related to three selected neural networks are also provided. In Section 5.4, the application of our proposed prediction method is verified by utilizing it in the RUL prediction of a gas turbine engine that is affected by degradations due to the compressor fouling and turbine erosion. Finally, the chapter is concluded in Section 5.5.

5.1 Problem Statement

Considering that the damage affects θ_t on the system health parameters $\lambda(.)$ (as a multiplicative faults vector), the system equations are then governed by:

$$\begin{aligned} x_t &= f_t(x_{t-1}, \theta_{t-1}^{\mathrm{T}} \lambda(x_{t-1}), \omega_t), \\ y_t &= h_t(x_t, \theta_t^{\mathrm{T}} \lambda(x_t), \nu_t), \end{aligned} \tag{5.1}$$

where $f_t : \mathbb{R}^{n_x} \times \mathbb{R} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_w} \longrightarrow \mathbb{R}^{n_x}$ and $h_t : \mathbb{R}^{n_x} \times \mathbb{R} \times \mathbb{R}^{n_v} \longrightarrow \mathbb{R}^{n_y}$ denote the nonlinear functions defining and representing the state at the next time step t ($t \in \mathbb{N}$) and the relationship between the state, parameters and measurements at time t, respectively. Also, $\theta_t \in \mathbb{R}^{n_\theta}$ denotes the unknown fault parameter vector at time t, where for a healthy system it is set equal to 1, $\lambda : \mathbb{R}^{n_x} \longrightarrow \mathbb{R}^{n_\theta}$ is a differentiable function in terms of system states that determines the health parameters. Moreover, ω_t, ν_t denote the uncorrelated white noise sequences with zero-mean and covariance matrices Q_t and R_t, respectively.

Our main objective in this chapter is to develop a hybrid framework for accomplishing failure prognosis by employing the previously stated two principle steps in health monitoring and prognosis, namely the joint state/parameter estimation, and their propagation prediction. The first step has been previously developed in Chapter 3 through design of a dual state/parameter estimation filter based on particle filters (PF). In this chapter, the second step of our proposed hybrid prognosis approach is implemented by predicting the long-term propagation of the system states and variations in the system health parameters. For this purpose, the previously developed PF-based prediction algorithm for the system states as well as health parameters variations (due to the fault vector θ_t) are utilized. The proposed prediction method is developed based

on extending the particle filters to future time horizons by utilizing an observation forecasting scheme. This scheme is developed by utilizing a neural network approach as a nonlinear time-series forecasting tool vs the linear time-series forecasting approach as discussed in Chapter 4. The developed hybrid framework for the system failure prognosis is shown to be robust against the choice of the neural network that is employed in the observation forecasting module. Moreover, it is shown that our hybrid methodology outperforms the approach that only utilizes the particle filters for achieving prediction. This justifies and substantiates development of the strategy that is introduced in this chapter.

The required background associated with the dual state/parameter estimation algorithm that has already been developed in Chapter 3 and briefly presented in Table 4.1 in Chapter 4. We are now in a position to present our proposed prediction framework by utilizing the neural network structures that are used as local models for obtaining and developing the observation profiles prediction.

5.2 Neural Network-based Prediction Framework

The challenging step in prognosis and health monitoring involves the system health prediction to future time horizons. In this section, the second step in our proposed hybrid prognosis approach (namely, the system health prediction) is developed. Our main goal is to extend and enhance the performance of conventional prediction frameworks that are based on particle filters as they rely on constant particle weights for their propagation to future time horizons (refer to [200]). This is achieved by invoking the concept of nonlinear univariate time-series approach [1] that is now

based on neural networks, where dual state/parameter estimation algorithm is extended to future time horizons according to the forecasted observations. In our proposed method, unlike the works in [200] and [1], the assumption of observation stationarity in each window is removed, since the neural networks will be trained adaptively based on the newly received data when the deviations between the forecasted observation from the neural network and the real observation increase from one test data set to another test data set.

By enhancing the observation forecasting component for the PF-based estimation algorithm, the PF scheme can now be extended to future time horizons by utilizing the same weight update rule as in the estimation step. The details corresponding to the observation forecasting module are now presented in the following subsections.

5.2.1 Neural Networks for Observation Prediction

As stated earlier, in this chapter we develop and implement three different types of neural networks for the purpose of observation forecasting. Specifically, we utilize MLP, wavenet, and recurrent neural networks, where the first two networks are feed forward and static, whereas the last one is equipped with delayed feedback that is more suitable for dynamical system representation and modeling.

In the remainder of this subsection, we briefly review the basics on neural networks. In the next subsection, we propose our hybrid approach by using neural networks for the purpose of prediction. Figure 5.1 depicts the structure of the recurrent neural networks (with one hidden

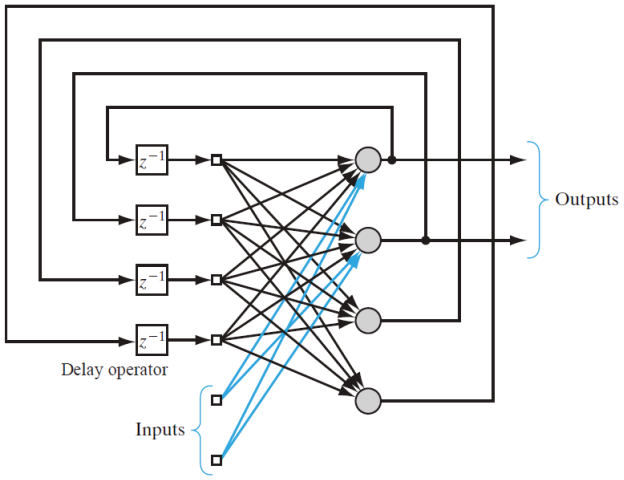


Figure 5.1: Structure of a recurrent neural network [1].

layer) that is utilized in this thesis. The schematic of the MLP and wavenet are standard based on the well known methods presented in the literature [1, 201]

Let the measurement vector at time t be denoted by $y_t = [y_{1,t}, ..., y_{n_y,t}]^{\mathrm{T}}$, where $y_{j,t}$ for $j = 1, ..., n_y$ refers to the j-th measurement output in the system (5.1). The output of a feed forward neural network (that is for both the MLP and the wavenet) can be represented by

$$y_{j,t} = \sum_{k=1}^{n_n} w_{j,k}^2 net_k(t) + b_j^2,$$

where the index j refers to the j-th output, n_n denotes the number of neurons in the hidden layer, $net_k(t)$ denotes the output of the k-th neuron of the hidden layer, and $w_{j,k}^2$ and b_j^2 denote the weights matrix and bias vector related to all MLPs used in the feed forward network structure. The main difference between the MLP and the wavenet lies on how one computes $net_k(t)$. In the MLP network, we have

$$net_k(t) = s_a(w_k z(t) + b_k^1),$$

where s_a denotes an activation function (in this thesis we use $s_a(r) = \frac{1}{1+e^{-r}}$), and $z(t)$ is the input vector to the network that is selected from the measurement outputs y_t of the system (5.1), w_k is a row weight vector with appropriate dimension, and b_k^1 a scalar denotes the bias parameter of the network. The structure of a wavenet can be expressed as

$$net_k(t) = w_a(||D_k(z(t) - t_k)||),$$

where D_k denotes a diagonal matrix, t_k denotes the translation vector and w_a denotes a wavelet function (in this thesis we use the Mexican-hat wavelet, namely $w_a(r) = (1 - r^2)e^{r^2}$) [201]. The main reason for this selection has its roots in the fact that as compared to the Haar wavelet, the Mexican-hat is differentiable, and hence the training processes (that involves the derivative of the activation function) is well-defined.

To represent and model a dynamical system with a feed forward neural network, it is necessary that delayed system outputs are also used in the set of inputs to the network. In other words, one can obtain an approximate representation or map as follows

$$y_{j,t} = \mathcal{F}(y_{j,t-1}, y_{j,t-2}, \cdots, y_{j,t-n_d})$$

where $\mathcal{F}$ denotes the function that the network is realizing. In this chapter, one can invoke the Akaike information criterion (AIC) [188] to determine n_d (the number of delayed outputs). Therefore, for the MLP and the wavenet networks the input vector is expressed as $z(t) = [y_{j,t-1}, y_{j,t-2}, \cdots, y_{j,t-n_d}]^{\mathrm{T}}$. However, when a recurrent neural network is employed one only needs the output $y_{j,t-1}$ as an input to the network. Consequently, one has $z(t) = y_{j,t-1}$.

For the recurrent neural network we have

$$net_k(t) = s_a(w_k z(t) + b_k^1 + w_k^d net(t-1) + b_k^1),$$

where w_k^d denotes the weight matrix related to delayed outputs, and

$$net(t-1) = \Big[net_1(t-1), net_2(t-1), \cdots, net_k(t-1)\Big]^{\mathrm{T}}.$$

Eventually, it is worth nothing that all the parameters in the feedforward network, i.e. $w_{j,k}^2$, and b_j^2, as well as D_k, and t_k in the wavenet network, and also w_j, and b_k^1, in recurrent network are tuned and adopted by using the Levenberg-Marquardt (LM) algorithm [202] in all these three networks. The main reason for this selection is related to the stability and fast convergence rate of the LM algorithm for solving the nonlinear least square problems as discussed in [1].

5.2.2 Neural Network Updating

For accomplishing the prediction task we consider each output as a time-series. A neural network such as the MLP, the wavenet or the recurrent network is applied for this purpose. When new observations are made available we measure performance of the network by using the j-th mean square error of the output vector y_t and its estimate vector $\hat{y}_t$ as given by

$$E_j = \frac{1}{n_{ts}} \sum_{t=1}^{n_{ts}} (\hat{y}_{j,t} - y_{j,t})^2,\ j = 1, ..., n_y \tag{5.2}$$

where $\hat{y}_{j,t}$ denotes the estimation of $y_{j,t}$ obtained from the neural network with fixed parameters as calculated in the previous training step of the algorithm, and n_{ts} denotes the number of observations that are used to validate the neural network. We consider two windows with lengths n_{tr} and n_{ts} (where n_{tr} denotes the number of observations that are used to train the neural network and n_{ts} is used to test the performance of the neural network). If E_j is larger than a predefined threshold, one needs to retrain the network by using the last n_{tr} observations.

5.2.3 Observation Forecasting Module

The observation forecasting module is constructed with the use of neural networks. We are interested in first constructing $y_{j,t}$ from the previous available observations $y_{j,t-p_i}$ as a univariate time-series, where p_j denotes the number of delayed outputs that are utilized. The proposed algorithm that utilizes neural networks is now summarized in Table 5.1. Note that the threshold th is selected according to a desirable accuracy that one expects from the overall prognosis scheme. To accomplish a more accurate prognosis one needs to specify a smaller th.

Table 5.1: Algorithm to retrain the neural networks that are used for observation forecasting that is integrated with the particle filters.

1. For each output, $y_{j,t}$, $j = 1, ..., n_y$, implement the following steps:
   (a) If the network is feedforward, apply the AIC algorithm [188] to determine the number of delayed outputs $y_{j,t-p_i}$, where p_j is the order of delay for the j-th output, that are used as inputs to the network, i.e. $y_{j,t} = \mathcal{F}(y_{j,t-1}, ..., y_{j,t-p_i})$.
   (b) By using the last n_{tr} observations, that is $\{y_{j,t}, ..., y_{j,t-n_{tr}+1}\}$, train and validate the neural network.
   (c) When n_{ts} new observations are available (we set $n_{ts} < h$), test the performance of the network by utilizing the equation (5.2).
   (d) If the network error (that is, E_j as defined in equation (5.2)) is larger than the threshold th, go to Step a). Otherwise, go to Step c).

5.2.4 Hybrid Prediction Methodology

The algorithm corresponding to our proposed hybrid prediction methodology that is achieved by integration of the neural networks with the particle filters (PF) is provided in Table 5.2. The neural networks are now used for generating the h-step ahead prediction of the measurement, i.e. $\hat{y}_{j,t+h}$. Finally, the h-step ahead prediction of the system hidden states/parameters is achieved by using the generated $\hat{y}_{j,t+h}$ from the particle filter scheme.

In other words, the h-step ahead predicted observations are denoted by $\hat{y}_{j,t+h}$ for $j = 1, ..., n_y$ and are computed from

$$\hat{y}_{j,t+h} = \sum_{k=1}^{n_n} w_{j,k}^2 net_k(t+h) + b_j^2,$$

where $net_k(t+h)$ is obtained from the predicted observations in the previous time step as

$$net_k(t+h) = s_a(w_k x + b_k^1 + w_k^d net(t+h-1) + b_k^1),$$

with

$$net(t+h-1) = [net_1(t+h-1), \cdots, net_k(t+h-1)]^{\mathrm{T}}$$

Moreover, integration of the neural networks with particle filters (PF) is established through the DualPF function ($\texttt{DualPF}(\hat{x}_{t+h-1|t}, \{\hat{x}_{t+h-1|t}^{(i)}\}_{i=1}^N, \hat{\theta}_{t+h-1|t}, \{\hat{\theta}_{t+h-1|t}^{(j)}\}_{j=1}^N, \Sigma_{\hat{\theta}_{t+h-1|t}}, \hat{y}_{t+h})$), that accepts the predicted observations $\hat{y}_{t+h} = [\hat{y}_{1,t+h}, ..., \hat{y}_{n_y,t+h}]^{\mathrm{T}}$ along with

1. the predicted state from the previous time step $\hat{x}_{t+h-1|t}$,
2. the corresponding state particles $\{\hat{x}_{t+h-1|t}^{(i)}\}_{i=1}^N$ (N denotes the number of particles),
3. the predicted parameters in the previous time step $\hat{\theta}_{t+h-1|t}$,

Table 5.2: Algorithm of the proposed hybrid prediction methodology.

1. Select the first M available observations.
2. For $j = 1 : n_y$
   (a) Generate the input-output data set for the neural networks training,
   (b) Run the code that is described in Table 5.1 for observation prediction that is based on the neural networks,
3. Run the PF algorithm with the predicted measurement $\hat{y}_{j,t+h}$ to obtain the h-steps ahead prediction of the states and parameters (e.g., h is set to 40 in Section 5.2). Utilize the neural networks to predict the observations for the h-steps ahead according to $\hat{y}_{j,t+h} = \sum_{k=1}^{n_n} w_{j,k}^2 net_k(t+h) + b_j^2$. Implement the dual particle filter algorithm for achieving the h-steps ahead prediction of the states/parameters by applying the predicted observations $\hat{y}_{t+h} = [\hat{y}_{1,t+h}, ..., \hat{y}_{n_y,t+h}]$, as: $[\hat{x}_{t+h|t}, \{\hat{x}_{t+h|t}^{(i)}\}_{i=1}^N, \hat{\theta}_{t+h|t}, \{\hat{\theta}_{t+h|t}^{(j)}\}_{j=1}^N, \Sigma_{\hat{\theta}_{t+h|t}}] = \texttt{DualPF}(\hat{x}_{t+h-1|t}, \{\hat{x}_{t+h-1|t}^{(i)}\}_{i=1}^N, \hat{\theta}_{t+h-1|t}, \{\hat{\theta}_{t+h-1|t}^{(j)}\}_{j=1}^N, \Sigma_{\hat{\theta}_{t+h-1|t}}, \hat{y}_{t+h})$ as specified in Table 4.1.
4. Move the observation window for the minimum of $n_{ts} = 10$ recently observed data points.
   (a) If the calculated MSE exceeds the threshold th go to Step 2 and re-train the neural networks based on the most recent M data points,
   (b) If the MSE does not exceed the threshold th continue with the observation prediction with the previously trained neural networks and go to Step 2.
5. If the available observation data set is exhausted, exit the algorithm, otherwise go to Step 1 with the most recent M data points.

4. the corresponding parameter particles $\{\hat{\theta}_{t+h-1|t}^{(j)}\}_{j=1}^N$, and

5. the parameter estimation covariance matrix that is denoted by $\Sigma_{\hat{\theta}_{t+h-1|t}}$,

all as input arguments. The function DualPF produces the predicted states/parameters and the predicted parameters covariance matrix for the next time step (refer to Step 3 in Table 5.2).

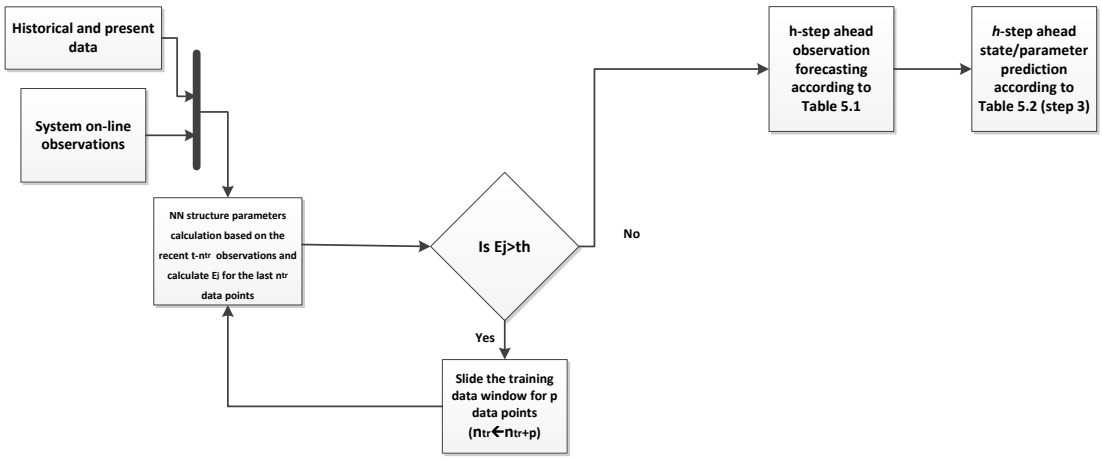


Figure 5.2: The Hybrid structure block diagram for state and parameter prediction.

A simple block diagram of our proposed hybrid structure for state and parameter prediction scheme based on the algorithms summarized in Tables 4.1, 5.1 and 5.2 is presented in Figure 5.2.

5.3 Remaining Useful Life (RUL) Evaluation

Our proposed hybrid scheme is conceived based on integration of model-based and neural network-based prognostic methods. It enables the user to define system health signatures for the purpose of obtaining and estimating the remaining useful life (RUL) of the system or its components. This is achieved not only through monitoring the changes in system observations, but also based on the changes in the internal system states as well as health parameters. In order to evaluate the health condition of the system, it is necessary to obtain information on the dynamics of the system.

In neural network-based prognosis methods, due to absence of a system mathematical model one requires to employ a large volume of historical data to capture and represent the dynamical behavior of the system [203]. On the other hand, without incorporating an observation forecasting module to a model-based prognosis approach, the prediction of system health parameters variations (through the fault vectors) will not be readily feasible. Moreover, performing the RUL prediction that is only based on the dynamical model of the system and its components can generally lead to erroneous results.

Hence, by properly taking advantage of the strength and capabilities of neural network-based and model-based approaches in our proposed hybrid methodology, the health evaluation of the system can be performed more robustly and effectively. Specifically, we will now determine the health signatures that are corresponding to the system health parameters (as well as observations) without explicitly requiring to have a large amount of historical data at our disposal.

Once the system health tracking and its evolution prediction to the future are accomplished, the RUL of the system can be evaluated by taking into account the probability of the failure distribution. This is accomplished according to a known criterion on the maximum acceptable change (critical bounds) of the system health parameters. In this approach, the system health evaluation for the RUL prediction will be performed based on changes in the system predicted health parameters. The observations are then utilized to evaluate only the performance of the prediction scheme in terms of changes in the system health parameters.

It is assumed clearly that the true values of the parameters are unknown and unmeasurable.

The percentage root mean square error (PRMSE) criterion for the estimated outputs is then used instead of the parameter estimation errors (that are practically unknown and unmeasurable) for evaluating the performance of the prediction algorithm. Specifically, for the output y_j at the time instant i the PRMSE is defined according to [180]:

$$\text{PRMSE}_{y_j}(i) = 100 \sqrt{\frac{1}{M} \sum_{m=t_{l_1}}^{m=t_{l_M}} (\frac{\hat{y}_{j,m+i} - y_{j,m+i}}{y_{j,m+i}})^2},$$

where $\hat{y}_{j,m+i}$ denotes the predicted j-th system output obtained from the particle filter at the time instant $m+i$, where m denotes the time instant at which the neural network is updated. In other words, $m = t_{l_1}, ..., t_{l_M}$ denote the time instants in the prediction horizon, $i = 1, ..., h$, and M refers to the total number of updated neural networks corresponding to the entire observation trajectory. Moreover, $y_{j,m+i}$ denotes the actual measured output (observation) at the time instant $m+i$, and the mean is taken over all $m+i$. The number of steps-ahead prediction is selected from the above PRMSE results based on an acceptable user pre-specified threshold th for each observation output (refer to Table 5.2).

The system RUL when multiple health parameters are estimated is now determined by first denoting and selecting $\theta_{\text{cr}}^{(l)}, l = 1, ..., n_\theta$, as the critical value for the l-th health parameter. The following rule is now utilized for evaluating the RUL at the time instant t [196] according to:

$$RUL(t) = t_j - t,$$

where t_j is defined as

$$t_j = \min_{t_j} \{\hat{\theta}_{t_j}^{(l)} - \theta_{\text{cr}}^{(l)} \geq 0,\ l = 1, ..., n_\theta\},$$

In other words, as described in Chapter 4, the RUL is determined to be the time at which the first parameter associated with a specific degradation damage reaches its associated criti-

cal threshold value. Once the critical threshold value in one of the parameters is reached, the maintenance action must be performed on the system or its components. Choosing a suitable value for the threshold $\theta_{\rm cr}^{(l)}$ is application specific and is generally determined based on the system performance and the operator or maintenance engineers experience. For most applications and problems defining an exact or a specific value for the RUL is not meaningful and feasible. Therefore, in general an acceptable bound is considered as a confidence interval or range for the RUL prediction. The above procedure is now used in the next section for estimating the RUL for an h-steps ahead horizon before the occurrence of a failure.

5.4 Gas Turbine Engine Failure Prognosis

The application of our proposed hybrid particle filter (PF)-based and neural network-based prediction method for health monitoring and prognosis of a gas turbine engine is presented in this section. Our interest is in investigating and determining failure prognosis of the engine, when the system components are assumed to be affected by health degradation phenomena. The proposed and developed hybrid prediction scheme is demonstrated and shown to be capable of prognosis the gas turbine engine when non-Gaussian process noise are applied to the system. Moreover, performance of our proposed state/parameter prediction scheme is evaluated and investigated under quite general degradation scenarios corresponding to both the turbine and the compressor components due to erosion and fouling phenomena, respectively.

In order to illustrate the effects of the choice on various neural networks that can be used for implementation of the neural network-based module of the hybrid methodology, simulation

scenarios under three different choices of neural networks are conducted. Specifically, we consider the following neural networks (i) MLP networks, (ii) recurrent neural networks, and (i) wavelet neural networks. The abbreviations "NN", "RNN" and "WNN" are used subsequently for the MLP, the recurrent and the wavenet neural networks, respectively.

Moreover, performance of the parameter prediction scheme will also be compared with the case when the observation forecasting module is not utilized. In other words, we will compare our proposed hybrid methodology with the prognosis scheme that is implemented by only the particle filtering scheme (this is denoted by "PF" in the simulation results). Specifically, in the latter approach, the state/parameter particles are propagated to the future time instants with constant weights (that is, without performing resampling) since the predicted observations are not available.

In order to discretize the gas turbine engine continuous-time model (as stated in Chapter 2) for implementation of our hybrid prognosis approach, an Euler Backward method is applied with a sampling period of $T_s = 10 \text{ msec}$.

The gas turbine engine health parameters are again represented by the compressor and the turbine efficiency, η_{C} and η_{T}, respectively, and the compressor and the turbine mass flow capacities, $\dot{m}_{\text{C}}$ and $\dot{m}_{\text{T}}$, respectively. A fault vector is therefore incorporated in the model to represent the effects of the system health parameters that are denoted by $\theta = [\theta_{\eta_{\text{C}}}, \theta_{m_{\text{C}}}, \theta_{\eta_{\text{T}}}, \theta_{m_{\text{T}}}]^{\text{T}}$. Each parameter variation is a manifestation of changes in the fault vector and is considered as a multiplicative fault type injected to the gas turbine engine.

5.4.1 Simulation Scenarios

The capability of our proposed hybrid methodology is now verified through its application to a gas turbine engine as given by the dynamical model (2.1) in Chapter 2. Two scenarios are considered by incorporating the effects of degradation damages to the engine compressor and turbine that are modeled as multiplicative faults. The main assumption that is made is that the only damage affecting the engine during the entire 300 simulation cycles (or flights) of operation will be due to the above two degradations.

It should be pointed out that for network training, after 48 number of Monte-Carlo simulation runs for the system degradation scenario, the fixed number of 200 data point has been selected for implementation of the algorithm. This window size is selected based on the average number of data which is required to achieve a training error less than $3e^{-5}$ and at the same time greater than $1e^{-5}$ to eliminate the risk of network overtraining.

The maximum number of step ahead prediction horizon for presenting the prognosis results has been selected according to a quantitative analysis on the predicted measurement outputs obtained from the particle filter corresponding to 48 Monte-Carlo simulation runs utilizing the powerful PRMSE metric. Based on the obtained results a 40-step ahead prediction horizon is chosen which shows an acceptable PRMSE error for all system measurements which were calculated in hybrid structure when the neural network part of the algorithm was implemented from all three mentioned neural network structures. This 40-step ahead prediction horizon is used to evaluate the probability of failure for accomplishing the prognosis task from the hybrid

Table 5.3: Number of neurons in the hidden layer of neural networks that are used for each output.

Network Output	y_1	y_2	y_3	y_4	y_5
MLP (NN)	8	10	7	15	8
Recurrent (RNN)	5	3	8	10	3
Wavenet (WNN)	4	5	4	4	4

framework.

A fixed number of $M = 200$ data points are used (as stated in Table 5.2). As new observations become available, if the deviation between the output of the neural network estimate and the actual observation according to equation (5.2) exceeds a pre-defined threshold criterion, the network is retrained by including the latest and recent observed data points.

The number of hidden layer neurons in the three neural networks that are used for simulations is provided in Table 5.3. The simulation scenarios are described in detail below. The corresponding results are presented in Figure 5.3. According to the results shown in this figure, it can be observed that as the number of steps-ahead prediction horizon exceeds beyond the 40 steps, the PRMSE value for the compressor pressure (P_C) and the turbine temperature (T_T) become over 4.2% of their nominal values. Consequently, the prediction horizon is set to 40 for all the subsequent simulation scenarios.

5.4.2 Compressor fouling scenario

For the compressor fouling scenario, a slowly changing linear degradation model is applied to the compressor health parameters during 300 cycles (or flights) of operation that cause a

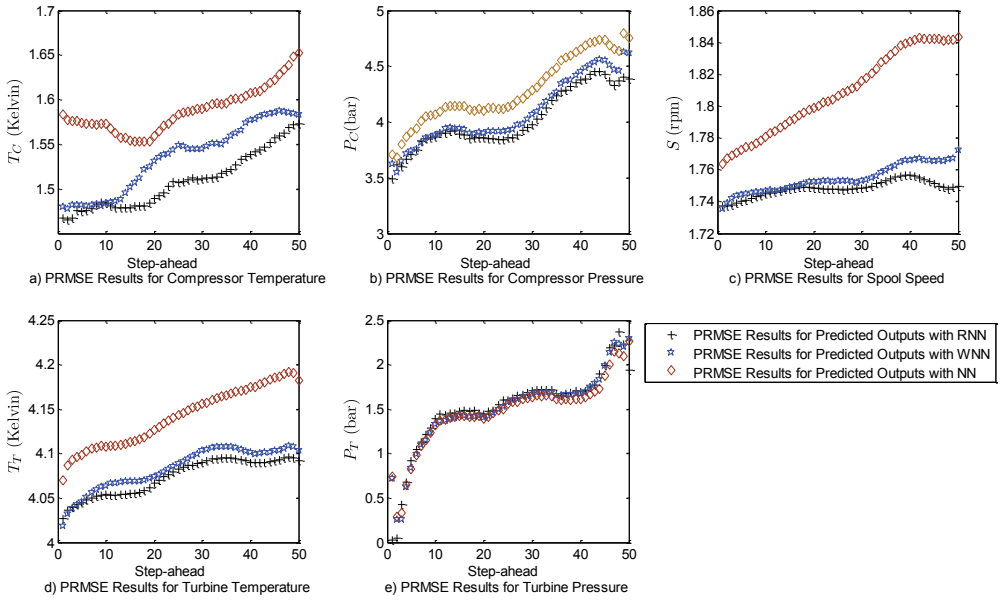


Figure 5.3: The PRMSE results for the gas turbine engine predicted outputs.

3% drop in the compressor efficiency and 1.5% drop in its mass flow capacity. In order to determine the compressor failure time based on the prediction results, variations in the system health parameters that are modeled by the fault vector are utilized. Assuming that the gas turbine engine starts operating from its healthy condition, the initial fault parameters are set to 1 (in other words, engine is 100% healthy). Therefore, if the fault parameter of the compressor efficiency (θ_{η_C}) and the mass flow capacity (θ_{m_C}) are set to 0.97 and 0.985, respectively, this implies that the compressor reaches a 3% fouling degradation that may cause failure of the gas turbine engine if a maintenance action was not performed.

For the remaining useful life (RUL) determination one requires to specify a threshold associated with the system health parameters. In view of the fault parameters critical values (as explained above), the related critical value for the compressor health parameter is obtained as

$\theta_{cr} = 0.7832$. This corresponds to $\theta_{\eta_C} = 0.97$ and $\theta_{m_C} = 0.985$.

Note that the percent of change in each health parameter is calculated with respect to the parameter under the healthy mode of the engine under the cruise condition of the flight. Fouling and erosion degradation phenomena follow linear propagation patterns under the low degradation index (equivalent to the fault severity level) of 1% to 3%, however as the degradation index increases (beyond 3%) they do not necessarily follow a linear profile [198].

To evaluate the performance of our proposed hybrid prediction strategy for the compressor failure prognosis, a prediction window of 40-steps ahead is now considered around the failure cycle. By assuming that the required amount of data for training the neural networks are 200 cycles ($M = 200$), the last 40 data points as shown in Figure 5.4 do correspond to the predicted health parameters of the compressor. Following the flight cycle 240 the compressor fouling degradation will go through a maintenance action and will be recovered by washing. Our interest here is in determining an acceptable approximation to the compressor failure time subject to the fouling degradation before its occurrence.

The results shown in Figure 5.4 demonstrate and illustrate that the PF method without its integration with the neural network-based observation prediction module, is incapable of tracking the compressor efficiency changes correctly, and has indeed over-estimated the health parameter in such a manner that it cannot be located within the $\pm 99\%$ confidence interval around the true value. Although as far as the mass flow capacity of the compressor is concerned the PF method has for some times estimated the health parameter within the confidence interval. However, the direction of changes corresponding to the predicted mass flow capacity is not correct. Therefore,

the use of the PF method alone will be excluded for further investigation on the system health evaluation for the fouling scenario as considered in the following studies. In contrast associated with both compressor health parameters (namely, the efficiency and the mass flow capacity), all the three neural network-based strategies are capable of predicting these parameters within the $\pm 99\%$ of the confidence bound around the true values (these are depicted by dash lines in Figure 5.4).

For determining the failure cycle as the result of the compressor fouling, a probabilistic analytical method is now proposed and implemented below. In our proposed method the distribution of the predicted data associated with the compressor system health parameters in each time window of 40-steps ahead horizon is fitted to a Gaussian distribution. Consequently, the changes in the mean of this distribution through subsequent prediction windows determine the changes in its related health parameter.

Let the health indicator vector under the fouling scenario be denoted by $\mathrm{HI}_{t+1:t+k} = \begin{pmatrix} \hat{\theta}_{\eta_{\mathrm{C}t+1|t}} & \cdots & \hat{\theta}_{\eta_{\mathrm{C}t+k|t}} \\ \hat{\theta}_{m_{\mathrm{C}t+1|t}} & \cdots & \hat{\theta}_{m_{\mathrm{C}t+k|t}} \end{pmatrix}$, where the mean is expressed as $\mathrm{mean}(\mathrm{HI_{t+1:t+k}}) = (\mu_{\eta_{\mathrm{C}}},\ \mu_{m_{\mathrm{C}}})^{\mathrm{T}}$, and the variance is denoted by $\mathrm{var}(\mathrm{HI_{t+1:t+k}}) = \begin{pmatrix} \delta^2_{\theta_{\eta_{\mathrm{C}}}} & 0 \\ 0 & \delta^2_{\theta_{m_{\mathrm{C}}}} \end{pmatrix}$. Therefore, the Gaussian probability density function corresponding to each health indicator (related to the compressor health parameters) is computed as

$$\mathrm{Gpdf}(\hat{\theta}_{\eta_{\mathrm{C}t+i|t}}|\mu_{\eta_{\mathrm{C}}}, \delta^2_{\eta_{\mathrm{C}}}) = \frac{1}{\delta_{\eta_{\mathrm{C}}}\sqrt{2\pi}} e^{-\frac{(\hat{\theta}_{\eta_{\mathrm{C}t+i|t}} - \mu_{\eta_{\mathrm{C}}})^2}{2\delta^2_{\theta_{\eta_{\mathrm{C}}}}}},$$

$$\mathrm{Gpdf}(\hat{\theta}_{m_{\mathrm{C}t+i|t}}|\mu_{m_{\mathrm{C}}}, \delta^2_{m_{\mathrm{C}}}) = \frac{1}{\delta_{m_{\mathrm{C}}}\sqrt{2\pi}} e^{-\frac{(\hat{\theta}_{m_{\mathrm{C}t+i|t}} - \mu_{m_{\mathrm{C}}})^2}{2\delta^2_{\theta_{m_{\mathrm{C}}}}}}.$$

$$\begin{aligned}
&\mathrm{FailureCycle}_{\eta_{\mathrm{C}}} = k^{\star},\ \text{such that } \hat{\theta}_{\eta_{\mathrm{C}}k^{\star}|t} \leq (1.05)\theta^{\mathrm{cr}}_{\eta_{\mathrm{C}}} \text{ and } k^{\star} = \mathrm{argmax}_{k=t+1}^{t+h}\mathrm{Gpdf}(\hat{\theta}_{\eta_{\mathrm{C}}k|t}),\\
&\mathrm{FailureCycle}_{m_{\mathrm{C}}} = l^{\star},\ \text{such that } \hat{\theta}_{m_{\mathrm{C}}l^{\star}|t} \leq (1.05)\theta^{\mathrm{cr}}_{m_{\mathrm{C}}} \text{ and } l^{\star} = \mathrm{argmax}_{l=t+1}^{t+h}\mathrm{Gpdf}(\hat{\theta}_{m_{\mathrm{C}}l|t}),
\end{aligned} \tag{5.3}$$

Moreover, the failure cycle corresponding to each health indicator is determined and declared as the cycle at which the predicted health parameter exceeds the 99.5% confidence interval around the critical value of that health parameter. Therefore, given that the fouling phenomenon is identified by its decreasing effects on both compressor health parameters, the upper bound on the critical values of the compressor health parameters is considered as the criterion for declaring the system failure cycle. However, since the failure cycle according to this guideline and rule can yield more than one cycle, the one corresponding to the maximum probability density function is considered as the appropriate failure cycle. This is formally obtained from equation (5.3) below, where $\theta^{\mathrm{cr}}_{\eta_{\mathrm{C}}}$ and $\theta^{\mathrm{cr}}_{m_{\mathrm{C}}}$ denote the compressor efficiency and the mass flow capacity critical values, respectively. Finally, given the two failure cycles that are obtained from equation (5.3) the selected system failure cycle is taken as the minimum of $\mathrm{FailureCycle}_{\eta_{\mathrm{C}}}$ and $\mathrm{FailureCycle}_{m_{\mathrm{C}}}$, that is $\mathrm{FailureCycle} = \min(\mathrm{FailureCycle}_{\eta_{\mathrm{C}}}, \mathrm{FailureCycle}_{m_{\mathrm{C}}})$.

For determining an approximation to the RUL for this scenario, the health parameters distributions that are predicted from the three considered neural networks along with the cycle at which the maximum probability of the failure has occurred, are shown in Figure 5.5. The ground truth failure cycle due to the critical values considered for the compressor health parameters is expected to be at the cycle 230. It is assumed that the predicted data distribution for the considered prediction window of size 40 (starting from the cycle 200) has a Gaussian distribution with its maximum that is located at the mean of the distribution. Therefore, the maximum

probability of failure is considered as the probability that is associated with the cycle at which the mean of the distribution is determined in all the three neural networks.

In Table 5.4, the fouling scenario results are summarized for comparing the predicted failure cycle. Towards this end, the predicted compressor health parameters (the compressor efficiency and the mass flow capacity) are selected as the health signatures for evaluating the system RUL. In this table, the predicted compressor health parameters along with the cycles at which these critical values were obtained are presented. It should be pointed out that the recorded values are related to the cycles at which the maximum probability of failure (based on the critical bounds for the fouling scenario) is reached. Moreover, the RUL is calculated as the difference between the predicted failure cycle and the ground truth failure cycle (namely the cycle 230) within the prediction window (starting at the cycle 200).

It can be concluded from Table 5.4 that for the hybrid prediction algorithms that are implemented with the RNN, WNN and NN neural networks, the predicted critical values that are considered as the values at which the maximum probability of the failure are obtained (refer to Figure 5.5) are located within the $\pm 99\%$ confidence interval around the actual critical value. Therefore, the predicted cycles at which these confidence intervals have exceeded are considered as the failure cycle due to the fouling. From the summary of the results and based on the criteria that are elaborated above, the failure cycle corresponding to the first health parameter that exceeds the critical interval should be considered as the failure cycle.

Corresponding to the mass flow rate capacity health parameter the predicted failure cycle corresponding to the RNN is at the cycle 215, and for the WNN is at the cycle 220, and finally

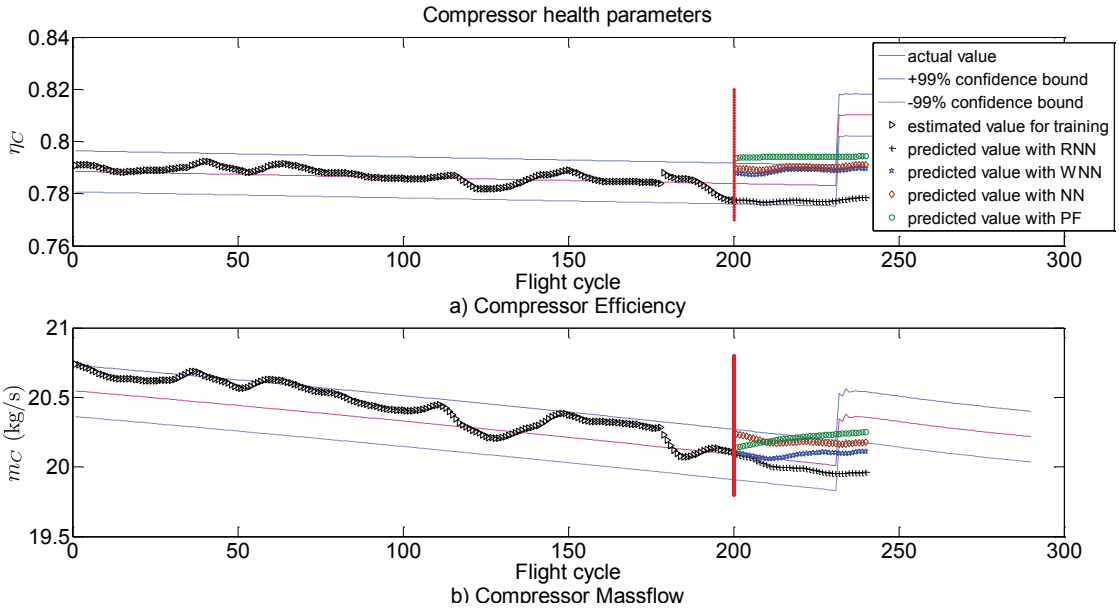


Figure 5.4: The predicted compressor health parameters where the prediction window starts at the cycle 200.

Table 5.4: RUL estimates due to the compressor fouling.

Par.	Prediction from the cycle 200 (ground truth cycle is 230)				within $\pm 99\%$ of the critical value
	network	predicted value	failure cycle	RUL error	
η_C	RNN	0.777	221	+9	yes
	WNN	0.788	231	-1	yes
	NN	0.79	226	+4	yes
m_C	RNN	19.995	215	+15	yes
	WNN	20.09	220	+10	yes
	NN	20.181	222	+8	yes

for the NN is at the cycle 222. This demonstrates that our proposed hybrid prediction algorithms are capable of predicting the failure cycle within 8 to 15 cycles before the gas turbine engine failure due to the fouling degradation, which is practically acceptable as it provides sufficient time to the ground personnel to perform the required maintenance tasks.

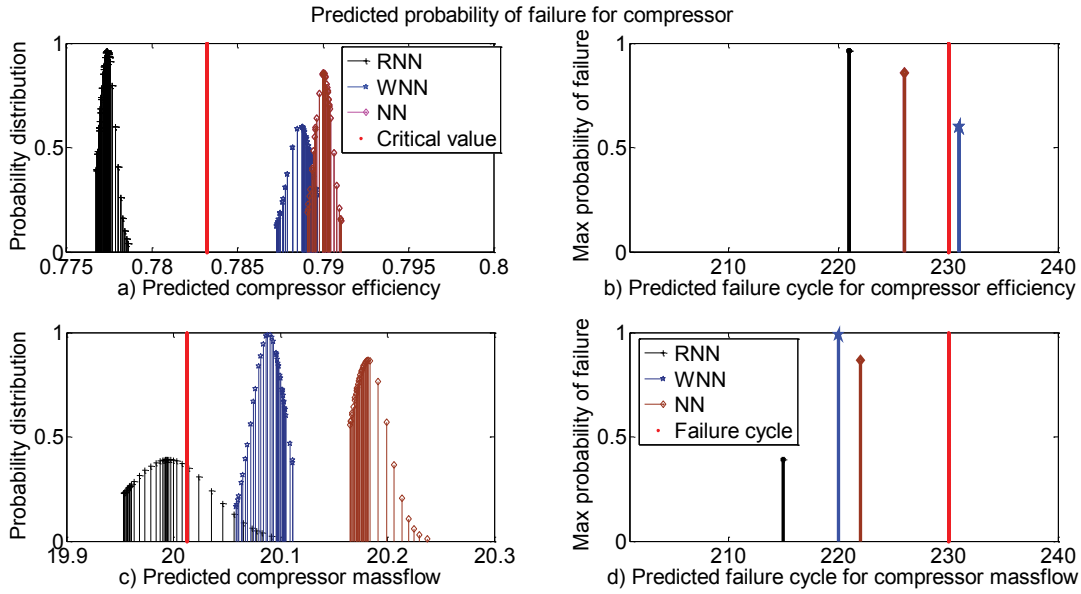


Figure 5.5: Maximum probability of the failure for the compressor efficiency and the mass flow capacity.

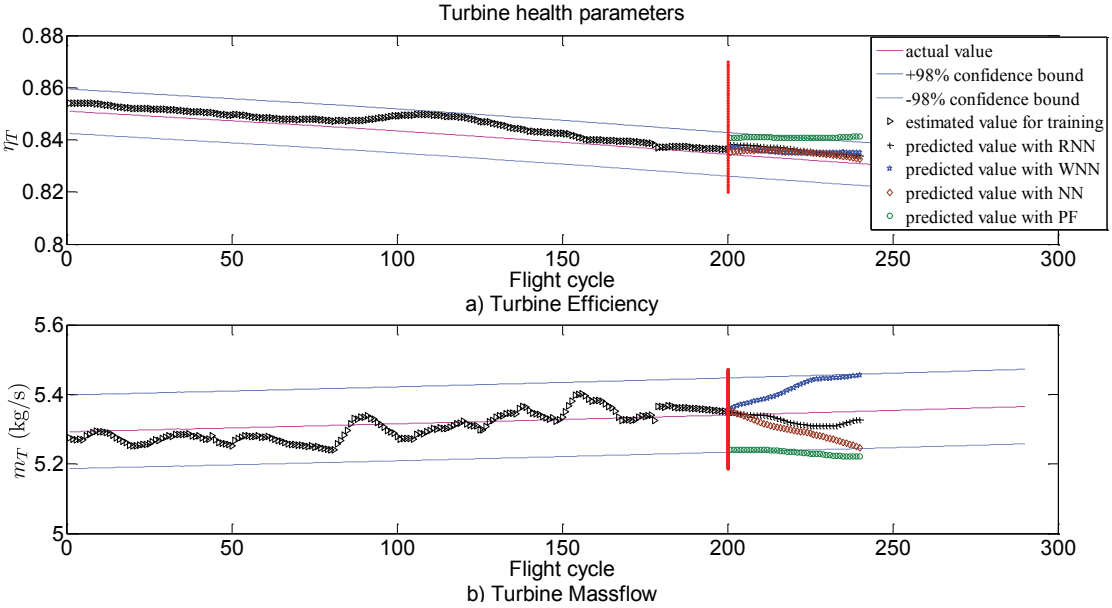


Figure 5.6: The predicted turbine health parameters where the prediction window starts at the cycle 200.

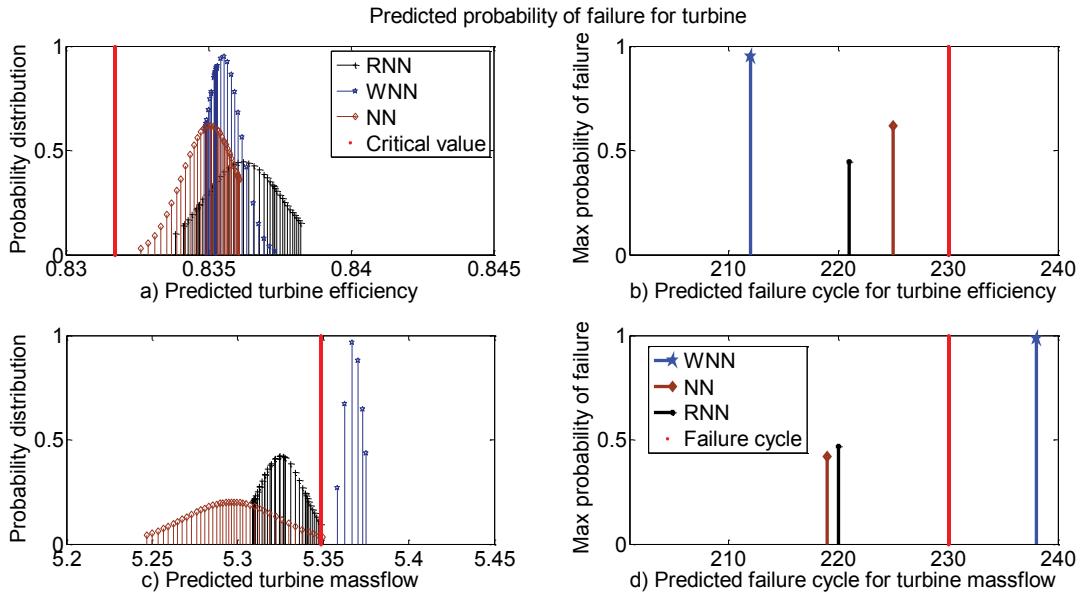


Figure 5.7: Maximum probability of the failure for the turbine efficiency and the mass flow capacity.

5.4.3 Turbine erosion scenario

In this scenario, the turbine erosion degradation is propagated through a quadratic evolution during the entire 300 flight cycles of the simulations that cause a 6% drop in the turbine efficiency and a 3% increase in its mass flow capacity. In order to determine the turbine failure time by utilizing our proposed hybrid prediction schemes, the same variations as in the first scenario in the health parameters are considered. Therefore, if the fault parameters of the turbine efficiency (θ_{η_T}) and the mass flow capacity (m_{m_T}) reach 0.94 and 0.97, respectively, this implies that the turbine has reached a 6% erosion that can cause failure in the gas turbine engine.

The health indicator vector in this scenario consists of the turbine efficiency and mass flow

$$\begin{aligned}
&\text{FailureCycle}_{\eta_{\mathrm{C}}} = k^{\star},\ \text{such that } \hat{\theta}_{\eta_{\mathrm{T}}k^{\star}|t} \leq (1.05)\theta^{\mathrm{cr}}_{\eta_{\mathrm{T}}} \text{ and } k^{\star} = \text{argmax}_{k=t+1}^{t+h}\text{Gpdf}(\hat{\theta}_{\eta_{\mathrm{T}}k|t}),\\
&\text{FailureCycle}_{m_{\mathrm{T}}} = l^{\star},\ \text{such that } \hat{\theta}_{m_{\mathrm{T}}l^{\star}|t} \geq (0.995)\theta^{\mathrm{cr}}_{m_{\mathrm{T}}} \text{ and } l^{\star} = \text{argmax}_{l=t+1}^{t+h}\text{Gpdf}(\hat{\theta}_{m_{\mathrm{T}}l|t}),
\end{aligned} \tag{5.4}$$

capacity and is denoted by $\mathrm{HI}_{t+1:t+k} = \begin{pmatrix} \hat{\theta}_{\eta_{\mathrm{T}}t+1|t} & \cdots & \hat{\theta}_{\eta_{\mathrm{T}}t+k|t} \\ \hat{\theta}_{m_{\mathrm{T}}t+1|t} & \cdots & \hat{\theta}_{m_{\mathrm{T}}t+k|t} \end{pmatrix}$. A probability analysis similar to the fouling scenario is also performed here. However, given that the erosion phenomenon is identified by its decreasing effects on the turbine efficiency and increasing effects on its mass flow capacity, exceeding the upper bound on the turbine efficiency critical value and exceeding the lower bound on the turbine mass flow capacity critical value are considered as indicators for determining the gas turbine engine failure cycle. Consequently, in deciding the failure cycle for this scenario the criterion of equation (5.4) given below will be considered, where $\theta^{\mathrm{cr}}_{\eta_{\mathrm{T}}}$ and $\theta^{\mathrm{cr}}_{\eta_{\mathrm{T}}}$ denote the turbine efficiency and the mass flow capacity critical values, respectively.

As in the previous scenario, for defining the system RUL, the related turbine health parameters critical values are obtained as $\theta_{cr} = 0.8317$, that corresponds to $\theta_{\eta_T} = 0.94$ and $\theta_{m_T} = 0.97$ (with respect to the gas turbine engine health parameters under the cruise condition).

Similar to the fouling scenario, a prediction window of 40-steps ahead is considered around the failure cycle. Assuming that one requires 200 data points for training the neural networks ($M = 200$), the last 40 data points in Figure 5.6 corresponds to the predicted health parameters of the turbine subject to the erosion. Note that the results shown in both Figures 5.4 and 5.6 show that the PF method is not capable of providing an accurate predictions based on the turbine health parameters within the acceptable confidence bounds, therefore it is not a suitable

strategy for performing the turbine health evaluation. This is due to the fact that the PF method results within the selected time window (around the ground truth failure) are located outside the confidence interval for almost all the prediction horizons. This implies that for the PF method the failure cycle has already been reached before the selected time window (which is clearly not correct). On the other hand, for both turbine health parameters, the three neural network hybrid schemes are capable of predicting the turbine health parameters within the $\pm 98\%$ of the confidence interval around the true values.

It should be pointed out that the main reason for the discrepancy between the prediction accuracy of the compressor and the turbine health parameters is related to simplifications that have been applied in modeling of the compressor dynamics in equation (2.1). Specifically, the turbine subsystem contains dynamics whereas the compressor subsystem is simpler and contains no dynamics. This does lead to a more accurate compressor prediction.

The RUL approximation results that are obtained from the predicted turbine health parameters distributions (by using the three proposed neural networks) along with the cycle where the maximum probability of failure has occurred are shown in Figure 5.7. The ground truth failure cycle given the critical values of the turbine health parameters is expected to be at the cycle 230. The comparative results are summarized in Table 5.5. These results imply that the hybrid prediction schemes implemented with the three neural network methods yield predicted critical values that correspond to when the maximum probability of the failure is achieved (refer to Figure 5.7) and do not exceed the $\pm 98\%$ confidence interval around the actual critical values. Consequently, the failure cycle corresponding to the RNN neural network is at the cycle 221,

for the WNN neural network is at the cycle 212, and finally for the NN neural network is at the cycle 220. This implies that the hybrid prediction schemes are capable of predicting the failure cycle due to erosion degradation within the 8 to 18 cycles before the gas turbine engine failure occurs.

Table 5.5: RUL estimates due to the turbine erosion.

Par.	Prediction from cycle 200 (ground truth cycle is 230)				within $\pm 99\%$ of the critical value
	method	predicted value	failure cycle	RUL error	
η_T	RNN	0.836	221	+9	yes
	WNN	0.835	212	+18	no
	NN	0.833	225	+5	yes
m_T	RNN	5.324	238	-8	yes
	WNN	5.366	219	+11	yes
	NN	5.296	220	+10	yes

5.5 Conclusion

In this chapter, a particle filtering scheme is integrated with neural network paradigms for predicting the future behavior (prognosis) of a nonlinear dynamical system states and parameters. The main advantages of our proposed hybrid prognosis framework can be summarized as follows:

1. Achieving more accurate prediction results as compared to model based particle filtering based method.

2. The combination of model based and intelligent based algorithms proposes a prediction structure which is not closely dependent on the structure of the network selected in the

intelligent based part, unlike pure intelligent based methods for time series forecasting where the structure of the neural network is not easily exchangeable.

3. The hybrid structure enables the prediction algorithm to predict not only the nonlinear system observations (which is the case in most of the intelligent based prediction algorithms) but also the system hidden states as well as health parameters which are not measurable in reality.

4. In the framework of prognosis, our proposed hybrid structure enables the users to consider more comprehensive set of indicators to track the health of the system. Actually, in this method the health indicators can be considered as the combination of the indicators one can consider in model based as well as intelligent based schemes.

5. For the neural network part, it is not needed to consider a complicated multi layer neural network to address the nonlinear system observations dynamics since the model based part is utilized as a add-on to neural network to compensate for achieving more accurate prediction results.

6. Finally, the application of the hybrid prognosis approach for health monitoring and failure prognosis of a gas turbine engine is firstly applied in this thesis.

   It is noted that all of the codings related to neural networks in this chapter have been done by my colleague Dr. Baniamerian.

Chapter 6

Ensemble Kalman Filters for State Estimation and Prediction of Two-time Scale Nonlinear Systems

An alternative method for formulation of the health monitoring problem in dynamical systems, suggests to model the dynamic of the damage mechanism as a slow state augmented to the system fast dynamical equations. This augmentation results in a two-time scale system to be investigated in the system health estimation and prediction steps in the health monitoring framework. In this chapter, a two-time scale filtering approach is developed for this purpose based on ensemble Kalman filtering method by taking advantages of model reduction concept. The performance of the proposed two-time scale ensemble Kalman filter is shown to be more accurate

and less expensive in terms of equivalent flop complexity, as compared to to the well-known particle filtering approach. Utilizing the augmentation of state equations and damage mechanism, our developed two-time scale ensemble Kalman filter is applied for health monitoring of a gas turbine engine when it is assumed to be affected by degradation phenomenon, i.e. erosion of the turbine, as the damage mechanism.

The main contributions of this chapter are now summarized as below:

1. Develop a solid health monitoring and prognosis framework according to two-time scale formulation strategy using the ensemble Kalman filtering (EnKF) approach:
   (a) Introduce a new strategy to incorporate the hidden damage model in the nonlinear system dynamics by utilizing the singular perturbation theory.
   (b) Develop a two-time scale ensemble Kalman filter (EnKF) methodology to address the system health tracking and prediction steps in the health monitoring and prognosis problem.

The remainder of this chapter is organized as follows. In Section 6.1, the statement of the nonlinear singularly perturbed problem is presented. The necessary background information regarding the nonlinear singularly perturbed systems and ensemble Kalman filtering approach is presented in Section 6.2. In Section 6.3, our main methodology for addressing the state estimation problem in nonlinear two-time scale systems is developed. In Section 6.4 our proposed method for state propagation prediction of nonlinear two-time scale systems in developed. Extensive simulation results and case studies are presented in Section 6.5. Finally, the chapter is

concluded in Section 6.6.

6.1 Problem Statement

Consider the time-invariant nonlinear singularly perturbed (NSP) system Σ_ϵ given by,

$$\Sigma_\epsilon : \begin{cases} \dot{x}_1(t) &= f_1(x_1(t), x_2(t), \epsilon) + g_1(x_1(t), x_2(t), \epsilon)\omega_1(t),\ x_1(t_0) = x_1(0), \\ \epsilon\dot{x}_2(t) &= f_2(x_1(t), x_2(t), \epsilon) + \sqrt{\epsilon}g_2(x_1(t), x_2(t), \epsilon)\omega_2(t),\ x_2(t_0) = x_2(0), \\ y(t) &= h(x_1(t), x_2(t), \epsilon) + \nu(t), \end{cases} \tag{6.1}$$

where $x_1(t) \in \mathbb{R}^{n_s}$ and $x_2(t) \in \mathbb{R}^{n_f}$ denote the slow and fast state vectors, respectively. The output $y(t) \in \mathbb{R}^{n_y}$ denotes the vector of system measurements, and the parameter $0 < \epsilon \ll 1$ is a small parameter that determines the two-time scale separation of the system as $\epsilon \to 0^+$. For some $\epsilon^\star > 0$, the functions $f_1(.), g_1(.) : \mathbb{R}^{n_s} \times \mathbb{R}^{n_f} \times [0, \epsilon^\star) \to \mathbb{R}^{n_s},\ f_2(.), g_2(.) : \mathbb{R}^{n_s} \times \mathbb{R}^{n_f} \times [0, \epsilon^\star) \to \mathbb{R}^{n_f}$, and $h(.) : \mathbb{R}^{n_s} \times \mathbb{R}^{n_f} \times [0, \epsilon^\star) \to \mathbb{R}^{n_y}$ are nonlinear continuous functions. The initial conditions $x_1(0)$, and $x_2(0)$ are assumed to be deterministic [204] and the noise inputs $\omega_1(t)$, $\omega_2(t)$, and $\nu(t)$ are zero-mean uncorrelated noise processes with variances $Q_1(t)$, $Q_2(t)$, and $R(t)$, respectively.

The dynamical system Σ_ϵ is utilized to characterize the two-time scale property in physical systems. One of the recent interesting applications of such modeling strategy is in damage modeling of mechanical systems as suggested in [31]. The main reason for using the singular perturbation strategy for representing the damage mechanism in physical systems is motivated by the slow dynamics (i.e., slowly changing) of the damage mechanism ($x_1(t)$ in (6.1)) as com-

pared to the other main physical component dynamics that are changing fast ($x_2(t)$ in (6.1)). Therefore, we utilize the model Σ_ϵ to represent the effects of the degradation damage on the health parameters of the dynamical system.

The system formulation Σ_ϵ can be utilized to develop a unified framework for health monitoring of the nonlinear systems which are assumed to be affected by degradation damages. Towards this aim, the slowly time-varying health parameters of the system (which are affected by degradation phenomenon) are augmented to the system states (fast states) as system slow states. More details regarding this formulation is presented in Subsection 6.5.3.

In the following section, the necessary background regarding the stochastic singular perturbation theory and the sufficient conditions that are required for its exponential stability are presented according to [204].

6.2 Background Information

Consider the system model Σ_ϵ where the following assumptions are held according to [204]:

Assumption 6.1. For each $\epsilon \geq 0$, $f_1(0,0,\epsilon) = 0$, $f_2(0,0,\epsilon) = 0$, $g_1(0,0,\epsilon) = 0$, and $g_2(0,0,\epsilon) = 0$.

Assumption 6.2. For each $x_1(t) \in \mathbb{R}^{n_s}$, $t \geq 0$, the equation $0 = f_2(x_1(t), x_2(t), 0)$ has a unique solution for $x_2(t)$ denoted by $x_2^\star(t) = \psi(x_1(t), 0)$, where $\psi(.)$ is continuously twice differentiable.

The second assumption leads to the reduced order model (slow dynamics) corresponding to Σ_ϵ by setting $\epsilon = 0$ and $x_2(t) = \psi(x_1(t), 0)$ in (6.1) as follows

$$\dot{x}_1(t) = f_1(x_1(t), \psi(x_1(t), 0), 0) + g_1(x_1(t), \psi(x_1(t), 0), 0)\omega_1(t). \tag{6.2}$$

Let us now define a new time variable $\tau = \frac{(t-t_0)}{\epsilon}$, as the fast time scale or the stretched time [93] for any $t_0 > 0$, so that the other new state variables $x_{1_f}(\tau) \triangleq x_1(t_0 + \epsilon\tau) = x_1(t)$ and $x_{2_f}(\tau) \triangleq x_2(t_0 + \epsilon\tau) = x_2(t)$, and the noise processes $w_1(\tau) = \sqrt{\epsilon}\omega_1(t_0 + \epsilon\tau)$ and $w_2(\tau) = \sqrt{\epsilon}\omega_2(t_0 + \epsilon\tau)$ are obtained. Therefore, the state space representation of Σ_ϵ in terms of these new variables takes the form

$$\begin{aligned} \frac{\mathrm{d}x_{1_f}(\tau)}{\mathrm{d}\tau} &= \epsilon f_1(x_{1_f}(\tau), x_{2_f}(\tau), \epsilon) + \sqrt{\epsilon} g_1(x_{1_f}(\tau), x_{2_f}(\tau), \epsilon) w_1(\tau), \\ \frac{\mathrm{d}x_{2_f}(\tau)}{\mathrm{d}\tau} &= f_2(x_{1_f}(\tau), x_{2_f}(\tau), \epsilon) + g_2(x_{1_f}(\tau), x_{2_f}(\tau), \epsilon) w_2(\tau). \end{aligned} \tag{6.3}$$

By setting $\epsilon = 0$, equation (6.3) becomes $\frac{\mathrm{d}x_{1_f}(\tau)}{\mathrm{d}\tau} = 0$, which results in $x_{1_f}(\tau) = constant = x_{1_f}(0) = x_1(t_0)$. Therefore, the so-called boundary-layer system dynamics is described by

$$\frac{\mathrm{d}x_{2_f}(\tau)}{\mathrm{d}\tau} = f_2(x_1(t_0), x_{2_f}(\tau), 0) + g_2(x_1(t_0), x_{2_f}(\tau), 0) w_2(\tau), \tag{6.4}$$

where $x_1(t_0)$ is considered as a constant parameter.

We now introduce the boundary-layer or the fast state as $\eta(t) = x_2(t) - \psi(x_1(t), 0)$. In the new coordinate system the singularly perturbed system Σ_ϵ can be represented as

$$\begin{aligned} \dot{x}_1(t) &= F_1(x_1(t), \eta(t), \epsilon) + G_{11}(x_1(t), \eta(t), \epsilon)\omega_1(t), \ x_1(t_0) = x_1(0), \\ \epsilon\dot{\eta}(t) &= F_2(x_1(t), \eta(t), \epsilon) + G_{21}(x_1(t), \eta(t), \epsilon)\omega_1(t) + G_{22}(x_1(t), \eta(t), \epsilon)\omega_2(t), \\ \eta(t_0) &= x_2(t_0) - \psi(x_1(t_0), 0), \end{aligned} \tag{6.5}$$

where the i-th and the l-th components of $F_1,\ F_2,\ G_{11},\ G_{21}$, and G_{22}, for $i, j, k = 1, ..., n_s,\ l =$

$1, ..., n_f$ are specified according to [204] as

$$
\begin{aligned}
F_{1_i}(x_1(t), \eta(t), \epsilon) &= f_{1_i}(x_1(t), \eta(t) + \psi(x_1(t), 0), \epsilon), \\
F_{2_l}(x_1(t), \eta(t), \epsilon) &= f_{2_l}(x_1(t), \eta(t) + \psi(x_1(t), 0), \epsilon) \\
&\quad - \epsilon[\sum_{j=1}^{n_s} \frac{\partial \psi_l(x_1(t), 0)}{\partial x_{1_j}} f_{1_j}(x_1(t), \eta(t) + \psi(x_1(t), 0), \epsilon)] \\
&\quad + \frac{1}{2} \sum_{j=1}^{n_s} \sum_{k=1}^{n_s} \frac{\partial^2 \psi_l(x_1(t), 0)}{\partial x_j \partial x_k} g_{1_j}(x_1(t), \eta(t) + \psi(x_1(t), 0), \epsilon) g_{1_k}(x_1(t), \eta(t) \\
&\quad + \psi(x_1(t), 0), \epsilon), \\
G_{11_i}(x_1(t), \eta(t), \epsilon) &= g_{1_i}(x_1(t), \eta(t) + \psi(x_1(t), 0), \epsilon), \\
G_{21_l}(x_1(t), \eta(t), \epsilon) &= \epsilon \sum_{j=1}^{n_s} \frac{\partial \psi_l(x_1(t), 0)}{\partial x_j} g_{1_j}(x_1(t), \eta(t) + \psi(x_1(t), 0), \epsilon), \\
G_{22_l}(x_1(t), \eta(t), \epsilon) &= \sqrt{\epsilon} g_{2_l}(x_1(t), \eta(t) + \psi(x_1(t), 0), \epsilon).
\end{aligned}
$$

It should be noted that the reduced order slow subsystem that is given by

$$\dot{x}_1(t) = F_1(x_1(t), 0, 0) + g_1(x_1(t), 0, 0)\omega_1(t),\ x_1(t_0) = x_1(0), \tag{6.6}$$

at $\epsilon = 0$ has an equilibrium at $x_1(t) = 0$ and $\omega_1(t) = 0$, and the boundary-layer fast subsystem is given by

$$\frac{\mathrm{d}\eta}{\mathrm{d}\tau} = F_2(x_1(0), \eta(\tau), 0) + g_2(x_1(0), \eta(\tau), 0)w_2(\tau), \tag{6.7}$$

that has an equilibrium at $\eta(\tau) = 0$, where $x_1(0)$ is considered as a fixed parameter.

Definition 6.1. [204] Consider the nonlinear stochastic system

$$\dot{x}(t) = f(t, x) + \sum_{i=1}^{M} g_i(t, x)\omega_i(t),\ x(t_0) = x_0 \tag{6.8}$$

where $t \in \mathbb{R}^+$ is the time, $x = [x_1, ..., x_n]^{\mathrm{T}}$ is the state vector, $f(.),\ g_i(.) : \mathbb{R}^+ \times \mathbb{R}^n \to \mathbb{R}^n,\ i = 1, ..., M$ are nonlinear deterministic vector functions as $f(.) = [f_1, ..., f_n]^{\mathrm{T}},\ g_i = [g_{i_1}, ..., g_{i_n}]^{\mathrm{T}}$, and $\omega_i(t)$ is Gaussian noise process. Let us define the operator $\mathcal{L}^{\star}_{(6.8)}(.)$, where the index (6.8)

refers to the corresponding equation that this operator is applied to, as follows

$$\mathcal{L}^{\star}_{(6.8)}(.) = \frac{\partial(.)}{\partial t} + \sum_{i=1}^{n} f_i(t,x)\frac{\partial(.)}{\partial x_i} + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{M} g_{i_k}(t,x)g_{j_k}(t,x)\frac{\partial^2(.)}{\partial x_i \partial x_j}$$

We now state the following assumptions that are necessary for introducing Theorem 6.1 on exponential stability of the system (6.5) according to [204].

Assumption 6.3. A positive-definite function $V : \mathbb{R}^{n_s} \rightarrow \mathbb{R}^{+}$ exists which is twice differentiable with respect to $x_1(t)$, and positive constants $\alpha^{\star}_{x_1}$ and $\gamma_k,\ k = 1, ..., 4$ exist such that the following inequalities are satisfied:

$$\begin{aligned} &\gamma_1\|x_1(t)\|^2 \leq V(x_1(t)) \leq \gamma_2\|x_1(t)\|^2, \\ &\mathcal{L}^{\star}_{(6.2)}V(x_1(t)) \leq -2\alpha^{\star}_{x_1}V(x_1(t)), \\ &|\frac{\partial V}{\partial x_{1_i}}| \leq \gamma_3\|x\|,\ |\frac{\partial^2 V}{\partial x_{1_i}\partial x_{1_j}}| \leq \gamma_4,\ i,j = 1, ..., n_s. \end{aligned} \tag{6.9}$$

Assumption 6.4. A positive-definite function $W : \mathbb{R}^{n_s} \times \mathbb{R}^{n_f} \rightarrow \mathbb{R}^{+}$ exists which is continuously twice differentiable with respect to $\eta(t)$ and $x_1(0)$, and positive constants $\alpha^{\star}_{\eta}$ and $\nu_p,\ p = 1, ..., 5$ exist such that the following inequalities are satisfied for $i, j = 1, ..., n_s$, and $k, l = 1, ..., n_f$:

$$\begin{aligned} &\nu_1\|\eta(t)\|^2 \leq W(x_1(0),\eta(t)) \leq \nu_2\|\eta(t)\|^2, \\ &\mathcal{L}^{\star}_{(6.4)}W(x_1(0),\eta(t)) \leq -2\alpha^{\star}_{\eta}W((x_1(0),\eta(t))), \\ &|\frac{\partial W}{\partial x_{1_i}}| \leq \nu_3\|\eta\|,\ |\frac{\partial W}{\partial \eta_l}| \leq \nu_4\|\eta\|, \\ &|\frac{\partial^2 W}{\partial x_{1_i}(0)\partial \eta_k}| \leq \gamma_5,\ |\frac{\partial^2 W}{\partial \eta_k \partial \eta_l}| \leq \nu_5. \end{aligned} \tag{6.10}$$

Assumption 6.5. The functions $f_1(.),\ f_2(.),\ g_1(.)$, and $g_2(.)$ are continuously differentiable with respect to x_1 and x_2, the function $\psi(x_1(t),\epsilon)$ is twice continuously differentiable with respect to x_1, and a real number $M_1 > 0$ exists such that for all $x_1 \in \mathbb{R}^{n_s}$ and $x_2 \in \mathbb{R}^{n_f}$,

$i, j = 1, ..., n_s$, and $k, l = 1, ..., n_f$, we have

$$\begin{aligned} &|\frac{\partial f_{1_i}}{\partial x_{1_j}}| \leq M_1,\ |\frac{\partial f_{1_i}}{\partial x_{2_k}}| \leq M_1,\ |\frac{\partial f_{2_k}}{\partial x_{1_j}}| \leq M_1,\ |\frac{\partial f_{2_k}}{\partial x_{2_l}}| \leq M_1, \\ &|\frac{\partial \psi_k}{\partial x_{1_j}}| \leq M_1,\ |\frac{\partial g_{1_i}}{\partial x_{1_j}}| \leq M_1,\ |\frac{\partial g_{1_i}}{\partial x_{2_l}}| \leq M_1,\ |\frac{\partial g_{2_i}}{\partial x_{1_j}}| \leq M_1,\ |\frac{\partial g_{2_i}}{\partial x_{2_l}}| \leq M_1. \end{aligned} \tag{6.11}$$

Assumption 6.6. The continuous functions $k_{f_1},\ k_{f_2},\ k_{g_1},\ k_{g_2} : [0, \epsilon^\star) \to \mathbb{R}^+$, with $k_{f_1}(0) = k_{f_2}(0) = k_{g_1}(0) = k_{g_2}(0) = 0$ and positive constants $d_{f_2},\ d_{g_1}$, and d_{g_2} exist such that for all $x_1 \in \mathbb{R}^{n_s},\ x_2 \in \mathbb{R}^{n_f}$ and $\epsilon \in (0, \epsilon^\star),\ i = 1, ..., n_s,\ l = 1, ..., n_f$, we have

$$\begin{aligned} &|f_{1_i}(x_1, x_2, \epsilon) - f_{1_i}(x_1, x_2, 0)| \leq k_{f_1}(\epsilon)(|x_1| + |\eta|), \\ &|f_{2_l}(x_1, x_2, \epsilon) - f_{2_l}(x_1, x_2, 0)| \leq k_{f_2}(\epsilon)(|x_1| + |\eta|), \\ &|g_{1_i}(x_1, x_2, \epsilon) - g_{1_i}(x_1, x_2, 0)| \leq k_{g_1}(\epsilon)(|x_1| + |\eta|), \\ &|g_{2_l}(x_1, x_2, \epsilon) - g_{2_l}(x_1, x_2, 0)| \leq k_{g_2}(\epsilon)(|x_1| + |\eta|), \end{aligned} \tag{6.12}$$

where $\eta(t) = x_2(t) - \psi(x_1(t), \epsilon),\ k_{f_2}/\epsilon \leq d_{f_2},\ k_{g_1}/\epsilon \leq d_{g_1}$, and $k_{g_2}/\epsilon \leq d_{g_2}$.

The Main Criterion. Suppose that Assumptions 6.1-6.6 hold, and assume positive constants $\alpha_{x_1} < \alpha^\star_{x_1}$ and $\alpha_{x_2} < \alpha^\star_{x_2}$. Then, the positive constants $\epsilon^+,\ c$ and continuous functions $\alpha_s,\ \alpha_f : (0,\ \epsilon^\star)$, $\phi : (0,\ \epsilon^\star) \to \mathbb{R}^+$ exist such that the following conditions hold for $t_0 \in \mathbb{R}^{n_s}$ and $\eta_0 \in \mathbb{R}^{n_f}$, namely

1) For every $\epsilon \in (0,\ \epsilon^\star)$ and $t \geq t_0$, the solutions of (6.5) are bounded as follows:

$$\begin{aligned} &\mathbb{E}|x_1(t, t_0, x_1(0), \eta(0))| \leq c(|x_1(0)| + \phi(\epsilon)|\eta(0)|)\exp\{-\alpha_s(t - t_0)\} \\ &\mathbb{E}|\eta(t, t_0, x_1(0), \eta(0))| \leq c|\eta(0)|\exp\{-\frac{\alpha_f(\epsilon)}{\epsilon}(t - t_0)\} + \epsilon(|x_1(0)| + \phi(\epsilon)|\eta(0)|)\exp\{-\alpha_s(t - t_0)\}. \end{aligned} \tag{6.13}$$

and

2) $\lim_{\epsilon \to 0} \alpha_s(\epsilon) = \alpha_{x_1}$, $\lim_{\epsilon \to 0} \alpha_f(\epsilon) = \alpha_\eta$, and $\lim_{\epsilon \to 0} \phi(\epsilon) = 0$.

Theorem 6.1 *[204] If the Assumptions 6.1-6.6 hold, for any positive $\alpha_{x_1} < \alpha^{\star}_{x_1}$, a positive constant ϵ^{+} and a positive continuous function $\alpha_s : (0,\ \epsilon^{+}) \rightarrow \mathbb{R}^{+}$ exist such that for every $\epsilon \in (0, \epsilon^{+})$, the full-order system* (6.5) *is exponentially stable with the rate $\alpha_s(\epsilon)$ and $\lim_{\epsilon \rightarrow 0} \alpha_s(\epsilon) = \alpha_{x_1}$, and the gain of the exponential convergence of the full-order system remains finite.*

Finding an explicit and exact solution to $\psi(x_1(t), \epsilon)$ is extremely difficult in general, for example by using Gröbner formula, the solution to $\psi(x_1(t), \epsilon)$ can be locally computed as proposed in [205]. Therefore, a common method is to consider the Taylor series expansion [93, 206, 207] of $\psi(.)$ with respect to ϵ as

$$\psi(x_1(t), \epsilon) = \psi_0(x_1(t)) + \epsilon\psi_1(x_1(t)) + O(\epsilon^2). \tag{6.14}$$

Substituting $\psi(.)$ into $x_2(t)$ in Σ_ϵ and applying the Assumption 6.2 results in the zeroth-order slow model [207] as

$$\dot{x}_1(t) = f_1(x_1(t), \psi_0(x_1(t)), 0) + g_1(x_1(t), \psi_0(x_1(t)), 0)\omega_1(t), \tag{6.15}$$

which describes the slow dynamics of the system Σ_ϵ, when the solution to $x_1(t)$ in equation (6.15) is denoted by $x_{1_s}(t)$. The discrepancy between the response of the zeroth-order slow model (6.15) with $\epsilon = 0$ and that of the full model Σ_ϵ represents the fast dynamics. Furthermore, one can assume that for the time interval $t \in [t_0,\ T]$ over which $x_{1_s}(t)$ exists, the following approximation is satisfied,

$$x_1(t) = x_{1_s}(t) + O(\epsilon). \tag{6.16}$$

The second term in (6.14) is now used to specify and define a first-order slow dynamics

according to

$$\dot{x}_1(t) = f_1(x_1(t), \psi_0(x_1(t)) + \epsilon\psi_1(x_1(t)), \epsilon) + g_1(x_1(t), \psi_0(x_1(t)) + \epsilon\psi_1(x_1(t)), \epsilon)\omega_1(t). \tag{6.17}$$

This process can be extended similarly to higher order corrected slow dynamics.

To describe the behavior of $x_2(t)$ in the fast-time scale, as mentioned earlier it is conventional to define a fast time-scale by setting $\tau = \frac{t-t_0}{\epsilon}$, [93, 206], where $\tau = 0$ at $t = t_0$ implies that $\eta(\tau) = x_2(\tau) - \psi_0(x_1(t))$ is defined, such that

$$\frac{d\eta}{d\tau} = f_2(x_1(0), \eta(\tau) + \psi_0(x_1(0))) + O(\epsilon), \tag{6.18}$$

where $\eta(0) = x_2(0) - \psi_0(x_1(0))$. The solution for $\eta(\tau)$ from the above initial condition value problem is used as a boundary layer correction to $x_2(t)$ approximation as follows,

$$x_2(t) = \eta(\frac{t-t_0}{\epsilon}) + \psi_0(x_1(t)) + O(\epsilon). \tag{6.19}$$

In order for (6.19) to converge to the slow approximation of $x_2(t) = \psi_0(x_1(t)) + O(\epsilon)$ (as per Assumption 6.2), the correction term $\eta(\tau)$ must as $\tau \to \infty$ decay to some $O(\epsilon)$ quantity.

In what follows, the sampled-data representsation of the nonlinear singularly perturbed system Σ_ϵ is presented according to [208] which is essential for our further investigation of the proposed two-time scale estimation method that is based on the ensemble Kalman filtering approach.

It should be noted that in our proposed two-time scale ensemble Kalman filter (TTS-EnKF) approach, $x_{1_s}(t)$ is approximated and the boundary layer correction of $x_2(t)$ is performed at each time step. Therefore, Theorem 6.1 ensures that the error in the approximation of $x_1(t)$ and

$x_2(t)$ is bounded to be of the $O(\epsilon)$ magnitude.

Now before presenting our developed TTS-EnKF approach, let us introduce the "Exact" EnKF estimation method for the nonlinear singularly perturbed system (NSP) which requires the discritization of the system state equations. The comparative performance of our proposed TTS-EnKF with the "Exact" EnKF are provided in the subsequent sections. Consequently, an overview of NSP systems discritization will be presented in the subsection below.

6.2.1 Sampled-Data Nonlinear Singularly Perturbed Dynamics

In this subsection, we introduce the sampled-data scheme for nonlinear singularly perturbed dynamics which is essential for investigating the state estimation scheme that is based on the EnKF through an exact state estimation approach. In the exact EnKF approach for addressing the estimation of the fast and slow states of a nonlinear singularly perturbed system, the estimation is performed without the use of the slow and fast states decomposition. It should also be noted that EnKF method is only applicable to discrete-time systems [116], and this is the reason we do not develop EnKF for the continuous-time NSP systems.

Consider the continuous-time nonlinear singularly perturbed system that is described by Σ_ϵ. Let us assume that ι denotes a sampling period where over the time interval of the length ι, the following conditions are satisfied,

$$\begin{aligned} \omega_i(t) &:= \omega_{i_k},\ k\iota \leq t < (k+1)\iota,\ i = 1, 2, \\ \nu(t) &:= \nu_k,\ k\iota \leq t < (k+1)\iota. \end{aligned} \tag{6.20}$$

Consequently, the system Σ_ϵ is first rewritten as D_ϵ, according to

$$D_\epsilon : \begin{cases} \dot{x}_1(t) &= \mathfrak{f}_1(x_1(t), x_2(t), \omega(t), \epsilon),\ x_{1_0} = x_1(0), \\ \epsilon\dot{x}_2(t) &= \mathfrak{f}_2(x_1(t), x_2(t), \omega(t), \epsilon),\ x_{2_0} = x_2(0), \\ y(t) &= h(x_1(t), x_2(t), \epsilon)) + \nu(t), \end{cases} \tag{6.21}$$

where $\mathfrak{f}_1(.) : \mathbb{R}^{n_s} \times \mathbb{R}^{n_f} \times \mathbb{R} \to \mathbb{R}^{n_s}$ and $\mathfrak{f}_2(.) : \mathbb{R}^{n_s} \times \mathbb{R}^{n_f} \times \mathbb{R} \to \mathbb{R}^{n_f}$ are smooth functions in their arguments with obvious definitions from Σ_ϵ. The discrete-time representation of D_ϵ is approximated according to [208, 209] based on the following remark.

Remark 6.1. Assume that the fixed sampling period ι is sufficiently close to ϵ, such that one can express ι as $\iota = \alpha\epsilon$, where α is a real number close to one. The fast sampled-data model of D_ϵ is given by

$$D_z : \begin{cases} x_{1_{k+1}} &= x_{1_k} + \epsilon(\alpha\mathfrak{f}_1(x_{1_k}, x_{2_k}, \omega_{1_k}, \epsilon) + O(\alpha^2)) + O(\epsilon^2),\ x_{1_0} = x_1(0), \\ x_{2_{k+1}} &= x_{2_k} + \alpha\mathfrak{f}_2(x_{1_k}, x_{2_k}, \omega_{2_k}, \epsilon) + O(\alpha^2) + O(\epsilon),\ x_{2_0} = x_2(0), \\ y_k &= h(x_{1_k}, x_{2_k}, \epsilon) + \nu_k, \end{cases} \tag{6.22}$$

where the error due to the higher-order approximation of the system dynamics is also incorporated into the $O(\epsilon^2)$ term in $x_{1_{k+1}}$ and $O(\epsilon)$ term in $x_{2_{k+1}}$.

Definition 6.2. According to Remark 6.1, the discrete-time NSP system Σ_ϵ can be represented as

$$D_\epsilon : \begin{cases} x_{1_k} &= x_{1_{k-1}} + \epsilon(\alpha f_1(x_{1_{k-1}}, x_{2_{k-1}}, \epsilon) + \alpha g_1(x_{1_{k-1}}, x_{2_{k-1}}, \epsilon)\omega_{1_k} + O(\alpha^2)) + O(\epsilon^2), \\ x_{1_0} &= x_1(0), \\ x_{2_k} &= x_{2_{k-1}} + \alpha f_2(x_{1_{k-1}}, x_{2_{k-1}}, \epsilon) + \alpha\sqrt{\epsilon} g_2(x_{1_{k-1}}, x_{2_{k-1}}, \epsilon)\omega_{2_k} + O(\alpha^2) + O(\epsilon), \\ x_{2_0} &= x_2(0), \\ y_k &= h(x_{1_k}, x_{2_k}, \epsilon) + \nu_k, \end{cases} \tag{6.23}$$

The discrete-time dynamical model D_ϵ is utilized in the remainder of this work for designing

the state estimation and prediction schemes that are based on exact EnKF approach. This is motivated by the fact that to compare our accomplished results with our proposed TTS-EnKF method discretization of the entire system dynamics is not necessary. Hence, a general overview on the theory of EnKF is provided in the next subsection.

6.2.2 An Overview on Ensemble Kalman Filter (EnKF) Theory

As stated earlier, the EnKF is a suboptimal estimation methodology where by utilizing the Monte-Carlo integration, the Fokker Planck equation is approximately solved [120]. Consider a general discrete-time nonlinear system with the following dynamics and measurements

$$\begin{aligned} x_{k+1} &= f(x_k) + \omega_k, \\ y_k &= h(x_k) + \nu_k, \end{aligned} \tag{6.24}$$

where $x_k,\ \omega_k \in \mathbb{R}^n,\ y_k$, and $\nu_k \in \mathbb{R}^p$. The zero-mean white noise process ω_k and ν_k have covariance matrices Q_k and R_k, respectively. The ensemble Kalman filtering (EnKF) method is based on two main steps that are designated as the *a priori* state estimation (*forecast*) step and the *a posteriori* state estimation (*analysis*) step [115, 120].

First, at the time instant k, we generate N ensemble members from the forecasted (*a priori*) state estimates with a random sample error that is generated from a normal distribution with the covariance Q_k, where the ensembles are denoted by $\{\hat{x}^{(i)}_{k|k-1},\ i = 1, ..., N\}$ and generated from the dynamics,

$$\hat{x}^{(i)}_{k|k-1} = f(\hat{x}^{(i)}_{k-1|k-1}) + \omega^{(i)}_k, \tag{6.25}$$

where $i = 1, ..., N$ refers to the ensemble number, $\hat{x}^{(i)}_{k|k-1}$ denotes the i-th ensemble member in the forecast step, $\hat{x}^{(i)}_{k-1|k-1}$ denotes the estimated ensemble member from the previous analysis

step, and $\omega_k^{(i)}$ denote samples from a normal distribution with the covariance Q_k. Note that the sample error covariance matrix which is calculated from the members of $\omega_k^{(i)}$ converges to Q_k as $N \to \infty$.

The ensemble mean $\hat{\bar{x}}_{k|k-1}$ is defined as the most probable forecast estimate of the state according to the Gaussian probability distribution function (in the classic Kalman filter), as

$$\hat{\bar{x}}_{k|k-1} = \frac{1}{N} \sum_{i=1}^{N} \hat{x}_{k|k-1}^{(i)}. \tag{6.26}$$

The main idea in the EnKF is to replace the error covariance matrix in the state estimation process with the ensemble covariance matrix since the actual value of the state x_k is not actually known. Therefore, the so-called *a priori* ensemble perturbation matrix $E_{k|k-1} \in \mathbb{R}^{n \times N}$ around the ensemble mean is defined as

$$E_{x_{k|k-1}} = [\hat{x}_{k|k-1}^{(1)} - \hat{\bar{x}}_{k|k-1}, ..., \hat{x}_{k|k-1}^{(N)} - \hat{\bar{x}}_{k|k-1}]^{\mathrm{T}}, \tag{6.27}$$

and the output ensembles, their mean, and their ensemble perturbation matrix are accordingly computed as

$$\begin{aligned} \hat{y}_{k|k-1}^{(i)} &= h(\hat{x}_{k|k-1}^{(i)}), \\ \hat{\bar{y}}_{k|k-1} &= \frac{1}{N} \sum_{i=1}^{N} \hat{y}_{k|k-1}^{(i)}, \\ E_{y_{k|k-1}} &= [\hat{y}_{k|k-1}^{(1)} - \hat{\bar{y}}_{k|k-1}, ..., \hat{y}_{k|k-1}^{(N)} - \hat{\bar{y}}_{k|k-1}]^{\mathrm{T}}. \end{aligned} \tag{6.28}$$

Next the covariance matrices $P_{k|k-1}^{xx}$, $P_{k|k-1}^{yy}$, and $P_{k|k-1}^{xy}$ are approximated by $\hat{P}_{k|k-1}^{xx}$, $\hat{P}_{k|k-1}^{yy}$, and $\hat{P}_{k|k-1}^{xy}$, respectively as follows

$$\begin{aligned} \hat{P}_{k|k-1}^{xx} &\triangleq \frac{1}{N-1} E_{x_{k|k-1}} E_{x_{k|k-1}}^{\mathrm{T}}, \\ \hat{P}_{k|k-1}^{xy} &\triangleq \frac{1}{N-1} E_{x_{k|k-1}} E_{y_{k|k-1}}^{\mathrm{T}}, \\ \hat{P}_{k|k-1}^{yy} &\triangleq \frac{1}{N-1} E_{y_{k|k-1}} E_{y_{k|k-1}}^{\mathrm{T}}. \end{aligned} \tag{6.29}$$

In fact, the ensemble members mean is interpreted as the best forecast estimate of the state, and the spread of the ensemble members around the ensemble mean is assumed to be the error between the best estimate and the actual value of the state (which is unknown) [120, 122].

In the second step of the EnKF algorithm, which is known as the *analysis step* or *a posteriori* state estimation step in classical Kalman filter, the error between the observed measured outputs and the estimated outputs from the *forecast step* is utilized to reduce the error covariance of the *a posteriori* estimated state by applying the Kalman gain according to,

$$\hat{x}_{k|k}^{(i)} = \hat{x}_{k|k-1}^{(i)} + \hat{K}_k(y_k - \hat{y}_{k|k-1}^{(i)}), \tag{6.30}$$

where the Kalman gain $\hat{K}_k$ is defined as

$$\hat{K}_k = \hat{P}_{k|k-1}^{xy}(\hat{P}_{k|k-1}^{yy} + R_k)^{-1}. \tag{6.31}$$

Finally, the *a posteriori* error covariance matrix is approximated according to,

$$\hat{P}_{k|k}^{xx} \triangleq \frac{1}{N-1} E_{x_{k|k}} E_{x_{k|k}}^{\mathrm{T}}, \tag{6.32}$$

where $E_{x_{k|k}}$ is defined in (6.27) with $\hat{x}_{k|k-1}^{(i)}$ replaced by $\hat{x}_{k|k}^{(i)}$ and $\hat{\bar{x}}_{k|k-1}$ replaced by the mean of the analysis estimate ensemble members, $\hat{\bar{x}}_{k|k}$.

It should be pointed out that the perturbed observation concept can also be used in the analysis step in order to generate the *a posteriori* ensemble members [116]. This method takes advantage of parallel data assimilation cycles, where for $i = 1, ..., N$, the *a posteriori* ensemble members are updated by

$$\hat{x}_{k|k}^{(i)} = \hat{x}_{k|k-1}^{(i)} + \hat{K}_k(y_k^{(i)} - \hat{y}_{k|k-1}^{(i)}), \tag{6.33}$$

where $y_k^{(i)}$ denotes the perturbed observations that are given by

$$y_k^{(i)} = y_k + \nu_k^{(i)}, \tag{6.34}$$

where $\nu_k^{(i)}$ is a zero-mean random variable with normal distribution and covariance R_k. The sample error covariance matrix that is computed from $\nu_k^{(i)}$ converges to R_k as $N \to \infty$.

We are now in a position to propose and develop our proposed two-time scale estimation algorithm that is based on the EnKF for the nonlinear singularly perturbed system D_z in the next section.

6.3 Ensemble Kalman Filters for State Estimation of Nonlinear Two-Time Scale Systems

A popular method for formulation of health monitoring problems of dynamical systems, suggests to model the corresponding dynamics of the damage mechanism as a "slow" state that is augmented to the system "fast" dynamical equations. This augmentation results in a two-time scale system to be investigated for the health estimation and prediction steps within a health monitoring framework. In this thesis, a two-time scale filtering approach is developed for this purpose based on the ensemble Kalman filtering approach by taking advantages of the model reduction concepts. The performance of our proposed two-time scale ensemble Kalman filter is shown to be more accurate and less computationally intensive in terms of the equivalent flop complexity, as compared to the well-known particle filtering approach. By utilizing an augmentation of the state equations and damage mechanism, our developed two-time scale ensemble Kalman filter can then be applied for health monitoring of complex nonlinear systems. Specifically, in this chapter our proposed methodology is applied to a gas turbine engine where it is

assumed that it is affected by degradation phenomenon, i.e. erosion of the turbine, as the damage mechanism. Extensive comparative studies are conducted to validate and demonstrate the advantages and capabilities of our proposed framework and methodology.

6.3.1 Exact EnKF Filtering Approach for Nonlinear Singularly Perturbed Systems

Consider the overall system state vector $x_k = [x_{1_k}^{\mathrm{T}},\ x_{2_k}^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{n_s+n_f}$. Assume that the state vector is updated over time by evolving according to the dynamics $\hat{\bar{x}}_{k|k-1} = [\hat{\bar{x}}_{1_{k|k-1}}^{\mathrm{T}},\ \hat{\bar{x}}_{2_{k|k-1}}^{\mathrm{T}}]^{\mathrm{T}}$, where $\hat{\bar{x}}_{k|k-1}$ denotes the most probable *a priori* state which is obtained from the dynamical model D_z. Consider that the *a priori* state estimates have Gaussian distributions, as is the case in classical Kalman filters [210]. Now, let $F(x_{k|k-1})$ denote an approximation to the probability density function for $x_{k|k-1}$ at the current time step k. Therefore, it can be approximated by a Gaussian probability density function as

$$F(x_{k|k-1}) \sim \exp[-\frac{1}{2}(x_{k|k-1} - \hat{\bar{x}}_{k|k-1})^{\mathrm{T}}(P_{k|k-1})^{-1}(x_{k|k-1} - \hat{\bar{x}}_{k|k-1})], \tag{6.35}$$

where $P_{k|k-1}$ and $\hat{\bar{x}}_{k|k-1}$ denote *a priori* covariance matrix and the most probable state associated with $F(x_{k|k-1})$.

According to the EnKF methodology, the *a priori* state $\hat{\bar{x}}_{k|k-1}$ is approximated through N ensemble members $\{\hat{x}_{k|k-1}^{(i)},\ i = 1, ..., N\}$, that are generated from the system state trajectories D_z as $\hat{x}_{k|k-1}^{(i)} = [(\hat{x}_{1_{k|k-1}}^{(i)})^{\mathrm{T}},\ (\hat{x}_{2_{k|k-1}}^{(i)})^{\mathrm{T}}]^{\mathrm{T}}$, such that

$$\begin{aligned}
\hat{x}_{1_{k|k-1}}^{(i)} &= \hat{x}_{1_{k-1|k-1}}^{(i)} + \epsilon\alpha(f_1(\hat{x}_{1_{k-1|k-1}}^{(i)}, \hat{x}_{2_{k-1|k-1}}^{(i)}, \epsilon) + g_1(\hat{x}_{1_{k-1|k-1}}^{(i)}, \hat{x}_{2_{k-1|k-1}}^{(i)}, \epsilon)\omega_{1_k}^{(i)}), \\
\hat{x}_{2_{k|k-1}}^{(i)} &= \hat{x}_{2_{k-1|k-1}}^{(i)} + \alpha(f_2(\hat{x}_{1_{k-1|k-1}}^{(i)}, \hat{x}_{2_{k-1|k-1}}^{(i)}, \epsilon) + \sqrt{\epsilon}g_2(\hat{x}_{1_{k-1|k-1}}^{(i)}, \hat{x}_{2_{k-1|k-1}}^{(i)}, \epsilon)\omega_{2_k}^{(i)}),
\end{aligned} \tag{6.36}$$

where $\hat{x}^{(i)}_{1_{k-1|k-1}}$ and $\hat{x}^{(i)}_{2_{k-1|k-1}}$ refer to the updated ensemble members in the previous time step (these are computed in the time update step of the estimation algorithm).

Consider the *a priori* ensemble perturbations from their most probable state that is given by

$$\delta\hat{x}^{(i)}_{k|k-1} = \hat{x}^{(i)}_{k|k-1} - \hat{\bar{x}}_{k|k-1}, \tag{6.37}$$

such that $\frac{1}{N}\sum_{i=1}^{N}\delta\hat{x}^{(i)}_{k|k-1} = 0$, and $\delta\hat{x}^{(i)}_{k|k-1}$ is defined as the ensemble perturbation vector, N denotes the number of ensemble members, $N \geq n_s + n_f$, and $\hat{\bar{x}}_{k|k-1}$ denotes the most probable *a priori* state estimate that is approximated by the ensemble members as,

$$\hat{\bar{x}}_{k|k-1} = \frac{1}{N}\sum_{i=1}^{N}\hat{x}^{(i)}_{k|k-1},$$

Hence, the *a priori* state estimation error covariance matrix $P_{k|k-1}$ can be approximated using the EnKF approach as described in Section 6.2 as follows,

$$P_{k|k-1} = \hat{X}_{k|k-1}\hat{X}^{\mathrm{T}}_{k|k-1}, \tag{6.38}$$

where $\hat{X}_{k|k-1} = \frac{1}{\sqrt{N-1}}[\delta\hat{x}^{(1)}_{k|k-1}, ..., \delta\hat{x}^{(N)}_{k|k-1}]^{\mathrm{T}}$.

For the two-time scale singularly perturbed system D_z, we assume that the covariance matrix can be represented through the eigenvalues and their related eigenvectors of the *a priori* state estimation error covariance as,

$$P_{k|k-1} = \sum_{j=1}^{n_s+n_f}\lambda^{(j)}_k u^{(j)}_k [u^{(j)}_k]^{\mathrm{T}}, \text{ where } \lambda^{(1)}_k > ... > \lambda^{(n_s)}_k > ... > \lambda^{(n_s+n_f)}_k, \tag{6.39}$$

where the covariance matrix $P_{k|k-1}$ has $(n_s + n_f)$ eigenvalues $\lambda^{(j)}_k,\ j = 1, ..., n_s + n_f$, and its corresponding eigenvectors $u^{(j)}_k$.

Assumption 6.7. For the *a priori* state estimate ensemble members that are generated from the two-time scale system D_z, the *a priori* error covariance matrix $P_{k|k-1}$ has $n_s + n_f$ eigenvalues $\lambda^{(1)}_k > ... > \lambda^{(n_s)}_k > ... > \lambda^{(n_s+n_f)}_k$, where for $j = 1, ..., n_f$, it is assumed that $\frac{\lambda^{(n_s+j)}_k}{\lambda^{(n_s)}_k} \leq \epsilon$.

This means that the important uncertainties in the *a priori* state estimate tend to lie in a low-dimensional subspace of dimension n_s.

According to Assumption 6.7, it follows that the error covariance in all the other directions are much less than the variance $\sum_{j=1}^{n_s} \lambda_k^{(j)}$. In the next step, the measurement update is performed by using the most recent observations. Towards this end, let us introduce the output ensemble members as,

$$\hat{y}_{k|k-1}^{(i)} = h(\hat{x}_{1_{k|k-1}}^{(i)}, \hat{x}_{2_{k|k-1}}^{(i)}, \epsilon). \tag{6.40}$$

Next, the measurement ensemble perturbation matrix is computed from

$$\hat{Y}_{k|k-1} = \frac{1}{N-1}[\delta\hat{y}_{k|k-1}^{(1)}, ..., \delta\hat{y}_{k|k-1}^{(i)}]^{\mathrm{T}}, \tag{6.41}$$

where $\delta\hat{y}_{k|k-1}^{(i)} = \hat{y}_{k|k-1}^{(i)} - \frac{1}{N}\sum_{i=1}^{N} \hat{y}_{k|k-1}^{(i)}$. Hence, the output prediction error ensembles $\tilde{y}_{k|k-1}^{(i)}$ are obtained as

$$\tilde{y}_{k|k-1}^{(i)} = y_k - h(\hat{x}_{1_{k|k-1}}^{(i)}, \hat{x}_{2_{k|k-1}}^{(i)}, \epsilon). \tag{6.42}$$

After applying the Kalman gain to the filter model (6.36), the *a posteriori* ensemble members and their most probable *a posteriori* state estimates using the exact EnKF scheme are obtained from

$$\begin{pmatrix} \hat{x}_{1_{k|k}}^{(i)} \\ \hat{x}_{2_{k|k}}^{(i)} \end{pmatrix} = \begin{pmatrix} \hat{x}_{1_{k|k-1}}^{(i)} \\ \hat{x}_{2_{k|k-1}}^{(i)} \end{pmatrix} + K_k \tilde{y}_{k|k-1}^{(i)}, \tag{6.43}$$

where $K_k \in \mathbb{R}^{(n_s+n_f)\times n_y}$ denotes the Kalman gain that is designed using

$$K_k = P_{k|k-1}^{xy}(P_{k|k-1}^{yy} + R_k)^{-1}, \tag{6.44}$$

where $P_{k|k-1}^{xy} = \hat{X}_{k|k-1}\hat{Y}_{k|k-1}^{\mathrm{T}}$, and $P_{k|k-1}^{yy} = \hat{Y}_{k|k-1}\hat{Y}_{k|k-1}^{\mathrm{T}}$.

We are now in the position to introduce our proposed TTS-EnKF methodology for the NSP

system Σ_ϵ.

6.3.2 The TTS-EnKF Filtering Strategy

Our proposed TTS-EnKF strategy for state estimation of the NSP system Σ_ϵ is based on the decomposition of the fast and slow dynamics of the system according to Section 6.1. By invoking Assumption 6.2, one can approximate $x_2(t)$ as $x_2(t) = \psi_0(x_1(t))$. Consequently, the dynamics of the slow states $x_1(t)$ can be approximated by

$$\dot{x}_1(t) = f_1(x_1(t), \psi_0(x_1(t)), 0) + g_1(x_1(t), \psi_0(x_1(t), 0)\omega_1(t). \tag{6.45}$$

To design the EnKF corresponding to the slow states of the system, first we discretize its dynamics given by equation (6.45). Applying the Euler discretization procedure will lead to the following discrete-time slow model

$$x_{1_k} = x_{1_{k-1}} + \iota f_1(x_{1_{k-1}}, \psi_0(x_{1_{k-1}}), 0) + \iota g_1(x_{1_{k-1}}, \psi_0(x_{1_{k-1}}), 0)\omega_{1_k}, \tag{6.46}$$

where ι denotes the sampling period which is assumed to be sufficiently small that does not violate the two-time scale property of the system.

The slow states of the system that correspond to the first n_s largest eigenvalues of $P_{k|k-1}$, are estimated by our proposed slow filter through two main steps namely, the time update and the measurement update steps as follows:

1. Time update for the slow filter: Time update step is accomplished through the following procedure:

- *A priori* state ensemble members are generated according to,

$$\hat{x}^{(i)}_{1_{k|k-1}} = \hat{x}^{(i)}_{1_{k-1|k-1}} + \iota f_1(\hat{x}^{(i)}_{k-1|k-1}, \psi_0(\hat{x}^{(i)}_{1_{k-1|k-1}}), 0) + \iota g_1(\hat{x}^{(i)}_{1_{k-1|k-1}}, \psi_0(\hat{x}^{(i)}_{1_{k-1|k-1}}), 0)\omega^{(i)}_{1_k} \tag{6.47}$$

  where $\hat{x}^{(i)}_{1_{k-1|k-1}}$ denotes the i-th ensemble member of the slow states in the previous time step, $\psi_0(\hat{x}^{(i)}_{1_{k-1|k-1}})$ denotes the i-th ensemble member of the approximated fast state that is obtained from the reduced order model.

- *A priori* ensemble perturbation is generated from,

$$\begin{aligned} \hat{\bar{x}}_{1_{k|k-1}} &= \frac{1}{N}\sum_{i=1}^{N} \hat{x}^{(i)}_{k|k-1}, \\ \delta\hat{x}^{(i)}_{1_{k|k-1}} &= \hat{x}^{(i)}_{1_{k|k-1}} - \hat{\bar{x}}_{1_{k|k-1}},\ i = 1, ..., N. \end{aligned} \tag{6.48}$$

- *A priori* error covariances are computed according to,

$$\begin{aligned} \hat{X}_{1_{k|k-1}} &= \frac{1}{\sqrt{N-1}}[\delta\hat{x}^{(1)}_{1_{k|k-1}}, ..., \delta\hat{x}^{(N)}_{1_{k|k-1}}]^{\mathrm{T}}, \\ \check{P}^{\mathrm{s}}_{1_{k|k-1}} &= \hat{X}_{1_{k|k-1}}\hat{X}^{\mathrm{T}}_{1_{k|k-1}}, \end{aligned}$$

  where the covariance matrix $\check{P}^{\mathrm{s}}_{k|k-1}$ corresponds to the first n_s largest eigenvalues of $P_{1_{k|k-1}}$ in (6.38) for which the fast eigenvalues of $P_{k|k-1}$ that satisfy $\frac{\lambda^{(n_s+j)}_k}{\lambda^{(n_s)}_k} \leq \epsilon$, for $j = 1, ..., n_f$, are ignored.

2. Measurement update for the slow filter: For the measurement update step as the observations become available at the time instant k, the *a posteriori* state estimates of the first n_s slow states are obtained. In this step, the output equation in Σ_ϵ is replaced by its Taylor series expansion with respect to ϵ after substituting for x_{1_k} and x_{2_k} with $\hat{x}^{(i)}_{1_{k|k-1}}$ and $\psi_0(\hat{x}^{(i)}_{1_{k-1|k-1}})$, respectively. Therefore, the output ensembles are computed according to

$$\hat{y}^{(i)}_{k|k-1} = h_0(\hat{x}^{(i)}_{1_{k|k-1}}, \psi_0(\hat{x}^{(i)}_{1_{k-1|k-1}}), 0). \tag{6.49}$$

Following the above steps, the measurement ensemble perturbation matrix is obtained from

$$\hat{Y}_{k|k-1} = \frac{1}{N-1}[\delta\hat{y}^{(1)}_{k|k-1}, ..., \delta\hat{y}^{(i)}_{k|k-1}]^{\mathrm{T}}, \tag{6.50}$$

where $\delta\hat{y}^{(i)}_{k|k-1} = \hat{y}^{(i)}_{k|k-1} - \frac{1}{N}\sum_{i=1}^{N}\hat{y}^{(i)}_{k|k-1}$.

Consequently, the output prediction error ensembles $\tilde{y}^{(i)}_{k|k-1}$ are obtained from,

$$\tilde{y}^{(i)}_{k|k-1} = y_k - h_0(\hat{x}^{(i)}_{1_{k|k-1}}, \psi_0(\hat{x}^{(i)}_{1_{k-1|k-1}}), 0). \tag{6.51}$$

Furthermore, the *a posteriori* ensemble members corresponding to the slow states and their most probable *a posteriori* state estimate are obtained from the slow filter dynamics

$$\begin{aligned} \hat{x}^{(i)}_{1_{k|k}} &= \hat{x}^{(i)}_{1_{k|k-1}} + \check{K}^{\mathrm{s}}_k\tilde{y}^{(i)}_{k|k-1}, \\ \hat{\bar{x}}_{1_{k|k}} &= \hat{\bar{x}}_{1_{k|k-1}} + \check{K}^{\mathrm{s}}_k\tilde{\bar{y}}_{k|k-1}, \end{aligned} \tag{6.52}$$

where $\tilde{\bar{y}}_{k|k-1} = \frac{1}{N}\sum_{i=1}^{N}\tilde{y}^{(i)}_{k|k-1}$, and $\check{K}^{\mathrm{s}}_k \in \mathbb{R}^{n_s\times n_y}$ denotes the Kalman gain of the slow filter. In order to select the Kalman gain for the slow filter, the *a posteriori* error covariance matrix of the slow filter is defined according to the following definition.

Definition 6.3. The covariance matrices $\check{P}^{xy}_{k|k-1}$ and $\check{P}^{yy}_{k|k-1}$ associated with the EnKF are approximated by $\check{P}^{xy}_{k|k-1} = \hat{X}_{1_{k|k-1}}\hat{Y}^{\mathrm{T}}_{k|k-1}$ and $\check{P}^{yy}_{k|k-1} = \hat{Y}_{k|k-1}\hat{Y}^{\mathrm{T}}_{k|k-1}$, respectively. Furthermore, let the *a posteriori* estimation error be defined as $\tilde{x}_{1_{k|k}} = x_{1_k} - \hat{\bar{x}}_{1_{k|k}}$, so that the *a posteriori* error covariance matrix of the slow filter can be obtained as $\check{P}^{\mathrm{s}}_{k|k} = \mathbb{E}[\tilde{x}_{1_{k|k}}\tilde{x}^{\mathrm{T}}_{1_{k|k}}]$, where associated with the EnKF it is approximated by $\check{P}^{\mathrm{s}}_{k|k} = \hat{X}_{1_{k|k}}\hat{X}^{\mathrm{T}}_{1_{k|k}}$, where $\hat{X}_{1_{k|k}}$ corresponds to the ensemble perturbation matrix that is generated from the *a posteriori* estimation of the ensemble members.

Consequently, the following lemma which is inspired from the work [211] is utilized to obtain and select the Kalman gain corresponding to the slow filter.

Lemma 6.1. *Consider the cost function defined as* $J_k(\check{K}^{\mathrm{s}}_k) = \mathbb{E}[\tilde{x}^{\mathrm{T}}_{1_{k|k}}W_k\tilde{x}_{1_{k|k}}] = \mathrm{trace}(\check{P}^{\mathrm{s}}_{k|k}W_k)$,

where W_k denotes a positive definite matrix, $\tilde{x}_{1_{k|k}}$ denotes the a posteriori estimation error, and $\check{P}^{\mathrm{s}}_{k|k}$ denotes the a posteriori error covariance matrix corresponding to the slow system dynamics. The Kalamn gain $\check{K}^{\mathrm{s}}_k$ that minimizes this function is obtained as $\check{K}^{\mathrm{s}}_k = \check{P}^{xy}_{k|k-1}(\check{P}^{yy}_{k|k-1} + R_k)^{-1}$, where $\check{P}^{xy}_{k|k-1} = \hat{X}_{1_{k|k-1}}\hat{Y}^{\mathrm{T}}_{k|k-1}$ and $\check{P}^{yy}_{k|k-1} = \hat{Y}_{k|k-1}\hat{Y}^{\mathrm{T}}_{k|k-1}$.

Proof: To show this claim, we follow the same approach as the one in the classical Kalman filter to design the gain. All the covariance matrices in the Kalman gain are replaced by their equivalent covariance matrices that are approximated through the EnKF approach.

Assume that in the classical Kalman filter the *a posteriori* state estimation error is obtained from

$$\begin{aligned} \tilde{x}_{1_{k|k}} &= x_{1_k} - \hat{x}_{1_{k|k-1}} + \check{K}^{\mathrm{s}}_k(\hat{y}_{k|k-1} - y_k), \\ &= \tilde{x}_{1_{k|k-1}} + \check{K}^{\mathrm{s}}_k(\hat{y}_{k|k-1} - y_k), \end{aligned} \tag{6.53}$$

therefore, the covariance of the *a posteriori* state estimation error is obtained from

$$\mathbb{E}\{\tilde{x}_{1_{k|k}}\tilde{x}^{\mathrm{T}}_{1_{k|k}}\} = \mathbb{E}\{(\tilde{x}_{1_{k|k-1}} + \check{K}^{\mathrm{s}}_k(\hat{y}_{k|k-1} - y_k))(\tilde{x}_{1_{k|k-1}} + \check{K}^{\mathrm{s}}_k(\hat{y}_{k|k-1} - y_k))^{\mathrm{T}}\}. \tag{6.54}$$

Now, by expanding (6.54) one gets

$$\begin{aligned} J(\check{K}^{\mathrm{s}}_k) = \mathbb{E}\{\tilde{x}_{1_{k|k}}\tilde{x}^{\mathrm{T}}_{1_{k|k}}\} &= \mathbb{E}\{\tilde{x}_{1_{k|k-1}}\tilde{x}^{\mathrm{T}}_{1_{k|k-1}}\} + \mathbb{E}\{\check{K}^{\mathrm{s}}_k(\hat{y}_{k|k-1} - y_k)\tilde{x}^{\mathrm{T}}_{1_{k|k-1}}\} \\ &+ \mathbb{E}\{\tilde{x}_{1_{k|k-1}}(\hat{y}_{k|k-1} - y_k)^{\mathrm{T}}\check{K}^{\mathrm{s}^{\mathrm{T}}}_k\} + \mathbb{E}\{\check{K}^{\mathrm{s}}_k(\hat{y}_{k|k-1} - y_k)(\hat{y}_{k|k-1} - y_k)^{\mathrm{T}}\check{K}^{\mathrm{s}^{\mathrm{T}}}_k\} \\ &= \mathbb{E}\{\tilde{x}_{1_{k|k-1}}\tilde{x}^{\mathrm{T}}_{1_{k|k-1}}\} + \mathbb{E}\{\check{K}^{\mathrm{s}}_k(\hat{y}_{k|k-1} - y_k)\tilde{x}^{\mathrm{T}}_{1_{k|k-1}}\} \\ &+ \mathbb{E}\{\tilde{x}_{1_{k|k-1}}(\hat{y}_{k|k-1} - y_k)^{\mathrm{T}}\check{K}^{\mathrm{s}^{\mathrm{T}}}_k\} - \mathbb{E}\{\check{K}^{\mathrm{s}}_k(y_k - \hat{y}_{k|k-1})(y_k - \hat{y}_{k|k-1})^{\mathrm{T}}\check{K}^{\mathrm{s}^{\mathrm{T}}}_k\}) \end{aligned} \tag{6.55}$$

By taking the derivative of (6.55) in terms of the Kalman gain $\check{K}^{\mathrm{s}}_k$ and considering that the output process y_k is independent of the estimated state and measurement process, i.e. $\mathbb{E}\{\tilde{x}_{1_{k|k-1}}y^{\mathrm{T}}_k\} = 0$ and $\mathbb{E}\{\hat{y}_{k|k-1}y^{\mathrm{T}}_k\} = 0$, and also by noting that the covariance of the

measurement noise is defined as $\mathbb{E}\{y_k y_k^{\mathrm{T}}\} = R_k$, it now follows that

$$\begin{aligned}\frac{\partial J(\check{K}_k^{\mathrm{s}})}{\partial \check{K}_k^{\mathrm{s}}} = 0 \Longrightarrow \frac{\partial}{\partial \check{K}_k^{\mathrm{s}}}(\mathbb{E}\{\tilde{x}_{1_{k|k-1}}\tilde{x}_{1_{k|k-1}}^{\mathrm{T}}\} + \mathbb{E}\{\check{K}_k^{\mathrm{s}}\hat{y}_{k|k-1}\tilde{x}_{1_{k|k-1}}^{\mathrm{T}}\} + \mathbb{E}\{\tilde{x}_{1_{k|k-1}}\hat{y}_{k|k-1}^{\mathrm{T}}\check{K}_k^{\mathrm{s}^{\mathrm{T}}}\} \\ - \mathbb{E}\{\check{K}_k^{\mathrm{s}}\hat{y}_{k|k-1}\hat{y}_{k|k-1}^{\mathrm{T}}\check{K}_k^{\mathrm{s}^{\mathrm{T}}}\} - \check{K}_k^{\mathrm{s}}R_k\check{K}_k^{\mathrm{s}^{\mathrm{T}}}) = 0 \\ \Longrightarrow \check{K}_k^{\mathrm{s}} = \mathbb{E}\{\tilde{x}_{1_{k|k-1}}\hat{y}_{k|k-1}^{\mathrm{T}}\}(\mathbb{E}\{\hat{y}_{k|k-1}\hat{y}_{k|k-1}^{\mathrm{T}}\} + R_k)^{-1},\end{aligned} \tag{6.56}$$

where $\mathbb{E}\{\tilde{x}_{1_{k|k-1}}\hat{y}_{k|k-1}^{\mathrm{T}}\} = \check{P}_{k|k-1}^{xy}$ and $\mathbb{E}\{\hat{y}_{k|k-1}\hat{y}_{k|k-1}^{\mathrm{T}}\} = \check{P}_{k|k-1}^{yy}$, which are obtained in the EnKF scheme as $\check{P}_{k|k-1}^{xy} = \hat{X}_{1_{k|k-1}}\hat{Y}_{k|k-1}^{\mathrm{T}}$ and $\check{P}_{k|k-1}^{yy} = \hat{Y}_{k|k-1}\hat{Y}_{k|k-1}^{\mathrm{T}}$. This completes the proof of the lemma. ■

Finally, the main three steps that are required in the measurement update of the slow states are as follows:

(i) Measurement ensemble members and ensemble perturbation matrices are obtained according to (6.50),

(ii) Kalman gain selection is accomplished from Lemma 6.1,

(iii) *A posteriori* state estimation results are obtained according to (6.52).

In the next step of the algorithm, the NSP system fast states are updated by assuming that the slow states are considered as constant at their initial values at $k-1$, i.e. $\hat{\bar{x}}_{k-1|k-1}$ for the time interval $[k-1, k)$.

Next, the same approach that is based on the EnKF is applied in order to obtain approximation to the NSP fast system states.

In our designed filter, it is assumed that $\tau = \frac{t-t_0}{\epsilon}$, so that the state equations in Σ_ϵ can be

rewritten as

$$\frac{\mathrm{d}x_{1_f}}{\mathrm{d}\tau} = \epsilon f_1(x_1(\tau), x_2(\tau), \epsilon) + \sqrt{\epsilon} g_1(x_1(\tau), x_2(\tau), \epsilon) w_1(\tau) \tag{6.57}$$

$$\frac{\mathrm{d}x_{2_f}}{\mathrm{d}\tau} = f_2(x_1(\tau), x_2(\tau), \epsilon) + g_2(x_1(\tau), x_2(\tau), \epsilon) w_2(\tau) \tag{6.58}$$

where the subscribe f denotes that the related states are in the fast-time scale (τ). Therefore, by setting $\epsilon = 0$ it results in

$$\begin{aligned} x_{1_f}(\tau) &= x_{1_f}(0) = x_1(t_0), \\ \dot{x}_{2_f}(\tau) &= f_2(x_{1_f}(0), x_{2_f}(\tau), 0) + g_2(x_{1_f}(0), x_{2_f}(\tau), 0) w_2(\tau), \end{aligned} \tag{6.59}$$

where the differentiation $\dot{x}_{2_f}(\tau)$ is defined with respect to the time scale τ. Consequently, the discrete-time model of (6.59) that will be utilized in design of the EnKF is obtained as

$$x_{2_{f_k}} = x_{2_{f_{k-1|k-1}}} + \iota f_2(\hat{\bar{x}}_{1_{k-1|k-1}}, x_{2_{f_{k-1|k-1}}}) + \iota g_2(\hat{\bar{x}}_{1_{k-1|k-1}}, x_{2_{f_{k-1|k-1}}}) w_{2_k}.$$

Note that $\tau \in [t_b,\ t_1]$, where $t_b > t_0$. Now, the fast filter state estimation is accomplished through two main steps, namely the time update and the measurement update steps.

3. Time Update of the Fast Filter: The time update is now performed in an n_f-dimensional space according to the following procedures,

   (a) *A priori* fast states ensemble generation according to,

   $$x^{(i)}_{2_{f_k}} = x^{(i)}_{2_{f_{k-1|k-1}}} + \iota f_2(\hat{\bar{x}}_{1_{k-1|k-1}}, x^{(i)}_{2_{f_{k-1|k-1}}}) + \iota g_2(\hat{\bar{x}}_{1_{k-1|k-1}}, x^{(i)}_{2_{f_{k-1|k-1}}}) w^{(i)}_{2_k},$$

   where $x^{(i)}_{2_{f_{k-1|k-1}}}$ denotes the fast ensemble members in the previous time step for $i = 1, ..., N$ ensembles.

   (b) *A priori* fast ensemble perturbation matrix generation that is based on the following

procedure,

$$\hat{\bar{x}}_{2_{f_{k|k-1}}} = \frac{1}{N}\sum_{i=1}^{N} \hat{x}^{(i)}_{2_{f_{k|k-1}}},$$

$$\delta\hat{x}^{(i)}_{2_{f_{k|k-1}}} = \hat{x}^{(i)}_{2_{f_{k|k-1}}} - \hat{\bar{x}}_{2_{f_{k|k-1}}},\ i = 1, ..., N,$$

$$\hat{X}_{2_{k|k-1}} = \frac{1}{\sqrt{N-1}}[x^{(1)}_{2_{f_{k|k-1}}} - \hat{\bar{x}}_{2_{f_{k|k-1}}}, ..., x^{(N)}_{2_{f_{k|k-1}}} - \hat{\bar{x}}_{f_{2_{k|k-1}}}]^{\mathrm{T}}.$$

In the next step, the *a posteriori* estimate of the fast system states are provided and approximated.

4. Measurement Update of the Fast Filter: It was pointed out earlier that the assumption that is used for health monitoring is that measurements are available at the fast-time scale of the system (unlike what is usually assumed in the analysis of two-time scale systems [206]). Therefore, the measurement update step should be performed for both slow and fast filters. The measurement update in this filter is also obtained through three main steps, namely, (i) measurement ensemble perturbation matrix computation, (ii) Kalman gain approximation, and (iii) *a posteriori* fast state estimation.

Definition 6.4. Let us define the output perturbation matrix $\hat{Y}^{f}_{k|k-1}$ as follows

$$\hat{Y}^{f}_{k|k-1} = \frac{1}{\sqrt{N-1}}[\hat{y}^{f^{(1)}}_{k|k-1} - \hat{\bar{y}}^{f}_{k|k-1}, ..., \hat{y}^{f^{(N)}}_{k|k-1} - \hat{\bar{y}}^{f}_{k|k-1}]^{\mathrm{T}}, \tag{6.60}$$

where $\hat{y}^{f^{(i)}}_{k|k-1} \triangleq h_0(\hat{\bar{x}}_{1_{k-1|k-1}}, \hat{x}^{(i)}_{2_{f_{k|k-1}}}),\ i = 1, ..., N$, and $\hat{\bar{y}}^{f}_{k|k-1} = \frac{1}{N}\sum_{i=1}^{N} \hat{y}^{f^{(i)}}_{k|k-1}$. Therefore, the following covariance matrices can be defined,

$$\breve{P}^{xy}_{k|k-1} = \hat{X}_{2_{k|k-1}} \hat{Y}^{f^{\mathrm{T}}}_{k|k-1},$$

$$\breve{P}^{yy}_{k|k-1} = \hat{Y}^{f}_{k|k-1} \hat{Y}^{f^{\mathrm{T}}}_{k|k-1}.$$

Consequently, the Kalman gain for the fast filter can be approximated according to the Lemma 6.1 and Definition 6.4 as,

$$\breve{K}^{f}_{k} = \breve{P}^{xy}_{k|k-1}(\breve{P}^{yy}_{k|k-1} + R_k)^{-1}. \tag{6.61}$$

Subsequently, the most probable *a posteriori* fast filter state estimate is obtained as

$$\begin{aligned} \hat{x}^{(i)}_{2_{f_{k|k}}} &= \hat{x}^{(i)}_{2_{f_{k|k-1}}} + \check{K}^{\mathrm{f}}_k \tilde{y}^{f^{(i)}}_{k|k-1}, \\ \hat{\tilde{x}}_{2_{f_{k|k}}} &= \hat{\tilde{x}}_{2_{f_{k|k-1}}} + \check{K}^{\mathrm{f}}_k \tilde{\tilde{y}}^{f}_{k|k-1}, \end{aligned} \tag{6.62}$$

where $\tilde{y}^{f^{(i)}}_{k|k-1} = y_k - \hat{y}^{f^{(i)}}_{k|k-1}$ and $\hat{\tilde{x}}_{2_{f_{k|k}}}$ is corrected according to the received observations by applying the Kalman gain $\breve{K}^{f}_k$.

Finally the most probable state vector $\hat{\tilde{x}}_{k|k}$ is updated according to

$$\hat{\tilde{x}}_{k|k} = [\hat{\tilde{x}}_{1_{k|k}},\ \hat{\tilde{x}}_{2_{f_{k|k}}}]^{\mathrm{T}}. \tag{6.63}$$

6.3.3 Error Analysis of the TTS-EnKF Algorithm

The convergence of conventional EnKF to the classical Kalman filter, and consequently to the optimal estimation of the system states for a linear problem has been addressed in [118]. However, in case when the problem is nonlinear the convergence of the EnKF to the optimal estimate is not guaranteed. Therefore, for our proposed TTS-EnKF approach, the boundedness of the estimated fast and slow states are analyzed under certain conditions and the error as a result of the decomposition of the full order dynamics into slow and fast dynamics is also analyzed. First, the following assumption is stated.

Assumption 6.8. Consider the system D_z for all $\{x_{1_k},\ \ x_{2_k}\}$, and $p \in [1, \infty)$ such that $\|x_{1_k}\|_p \leq b_1(k,p,\epsilon)$, $\|x_{2_k}\|_p \leq b_2(k,p,\epsilon)$, $\|f_1(x_{1_k}, x_{2_k}, \epsilon)\|_p \leq d_1(k,p,\epsilon)$, $\|f_2(x_{1_k}, x_{2_k}, \epsilon)\|_p \leq d_2(k,p)$, $\|g_1(x_{1_k}, x_{2_k}, \epsilon)\omega^{(i)}_{1_k}\|_p \leq d_3(k,p,\epsilon)$, $\|g_2(x_{1_k}, x_{2_k}, \epsilon)w^{(i)}_{2_k}\|_p \leq d_4(k,p,\epsilon)$, $\|\psi_0(x_{1_k})\| \leq c_2(k,p,\epsilon)$, and $\|h(x_{1_k}, x_{2_k}, \epsilon)\|_p \leq d_5(k,p,\epsilon)$ are bounded for some parameters c_i and d_j with $i = 1, 2$ and $j = 1, ..., 5$, where for a real positive number $p > 1$, the norm of a vector $x \in \mathbb{R}^{n\times 1}$ is defined as $\|x\|_p := (\sum_{i=1}^{n} |x_i|^p)^{\frac{1}{p}}$, with x_i for $i = 1, ..., n$ denoting the elements of the vector x.

The following theorem, guarantees the boundedness of the TTS-EnKF scheme *a posteriori* estimation error by considering the boundedness of the system state and output equations according to Assumption 6.8.

Theorem 6.2 *Consider the discrete-time nonlinear singularly perturbed system* (6.23)*. Let the state estimation problem be accomplished by utilizing the TTS-EnKF strategy through the a posteriori ensemble members update according to equations* (6.52) *and* (6.62) *for the slow and the fast states, respectively. Provided that Assumption 6.8 holds, then there exist parameters* $c_1(k,p,\epsilon)$ *and* $c_2(k,p,\epsilon)$ *for all* k *and all* $p \in [1,\infty)$ *such that* $\|\hat{x}^{(i)}_{1_{k|k}}\|_p \leq c_1(k,p,\epsilon)$ *and* $\|\hat{x}^{(i)}_{2_{k|k}}\|_p \leq c_2(k,p,\epsilon)$.

Proof: We invoke induction to show the result. For $k=0$, each $x^{(i)}_{1_0}$, and $x^{(i)}_{2_0}$ for $i=1,...,N$ is a normal distribution and bounded. Assume that for $k-1$, $\|\hat{x}^{(i)}_{1_{k-1|k-1}}\|_p \leq c_1(k-1,p,\epsilon)$ and $\|\hat{x}^{(i)}_{2_{k-1|k-1}}\|_p \leq c_2(k-1,p,\epsilon)$ for all i, then for the time instant k associated with the slow filter according to (6.52) we have

$$\hat{x}^{(i)}_{1_{k|k}} = \hat{x}^{(i)}_{1_{k|k-1}} + \check{K}^{\mathrm{s}}_k \tilde{y}^{(i)}_{k|k-1}.$$

By considering Assumption 6.8, from the boundedness of $\hat{x}^{(i)}_{1_{k-1|k-1}}$ and $\psi_0(\hat{x}^{(i)}_{1_{k-1|k-1}})$ one obtains $\|f_1(\hat{x}^{(i)}_{1_{k-1|k-1}}, \psi_0(\hat{x}^{(i)}_{1_{k-1|k-1}})\|_p \leq d_1(k-1,p,\epsilon)$ and $\|g_1(\hat{x}^{(i)}_{1_{k-1|k-1}}, \psi_0(\hat{x}^{(i)}_{1_{k-1|k-1}})\omega^{(i)}_{1_{k-1}}\|_p \leq d_3(k-1,p,\epsilon)$. Consequently, for the *a priori* state estimate of the slow state according to (6.47) we have

$$\|\hat{x}^{(i)}_{1_{k|k-1}}\|_p \leq c_1(k-1,p,\epsilon) + \epsilon\alpha d_1(k-1,p,\epsilon) + \epsilon\alpha d_3(k-1,p,\epsilon). \tag{6.64}$$

Now, by considering the boundedness of $\|\hat{x}^{(i)}_{1_{k|k-1}}\|_p$ according to (6.64) and Assumption 6.8, the

boundedness of the output equation follows since we have

$$\begin{aligned}\|\hat{y}_{k|k-1}^{(i)}\| &\leq \|h_0(\hat{x}_{k|k-1}^{(i)}, \psi_0(\hat{x}_{k|k-1}^{(i)})\| \\ &\leq d_5(k-1,p,\epsilon).\end{aligned} \tag{6.65}$$

Note that for computing of the Kalman gain in the measurement update step based on Lemma 6.1, we have $\check{K}_k^{\mathrm{s}} = \check{P}_{k|k-1}^{xy}(\check{P}_{k|k-1}^{yy} + R_k)^{-1}$, where $\check{P}_{k|k-1}^{xy} = \hat{X}_{1_{k|k-1}}\hat{Y}_{k|k-1}^{\mathrm{T}}$ and $\check{P}_{k|k-1}^{yy} = \hat{Y}_{k|k-1}\hat{Y}_{k|k-1}^{\mathrm{T}}$. Now, to show the boundedness of the Kalman gain we have to show the boundedness of $\check{P}_{k|k-1}^{xy}$ and $\check{P}_{k|k-1}^{yy}$, as follows

$$\begin{aligned}\|\check{P}_{k|k-1}^{xy}\|_p &= \frac{1}{N-1}\|(\hat{x}_{1_{k|k-1}}^{(i)} - \hat{\bar{x}}_{1_{k|k-1}})(\hat{y}_{k|k-1}^{(i)} - \hat{\bar{y}}_{k|k-1})^{\mathrm{T}}\|_p \\ &= \frac{1}{N-1}\|\hat{x}_{k|k-1}^{(i)}\hat{y}_{k|k-1}^{(i)^{\mathrm{T}}} - \hat{x}_{1_{k|k-1}}^{(i)}\hat{\bar{y}}_{k|k-1}^{\mathrm{T}} - \hat{\bar{x}}_{1_{k|k-1}}\hat{y}_{k|k-1}^{(i)^{\mathrm{T}}} + \hat{\bar{x}}_{1_{k|k-1}}\hat{\bar{y}}_{k|k-1}^{\mathrm{T}}\|_p \\ &\leq \frac{1}{N-1}(\|\hat{x}_{1_{k|k-1}}^{(i)}\hat{y}_{k|k-1}^{(i)^{\mathrm{T}}}\|_p + \|\hat{\bar{x}}_{1_{k|k-1}}\hat{\bar{y}}_{k|k-1}^{\mathrm{T}}\|_p).\end{aligned} \tag{6.66}$$

By invoking the Cauchy inequality [194] the two terms in the last inequality in (6.66) can be rewritten as

$$\begin{aligned}\|\hat{x}_{1_{k|k-1}}^{(i)}\hat{y}_{k|k-1}^{(i)^{\mathrm{T}}}\|_p &\leq \mathbb{E}(\|\hat{x}_{1_{k|k-1}}^{(i)}\|^p\|\hat{y}_{k|k-1}^{(i)^{\mathrm{T}}}\|^p)^{\frac{1}{p}}, \\ &\leq \mathbb{E}(\|\hat{x}_{1_{k|k-1}}^{(i)}\|^{2p})^{\frac{1}{2p}}\mathbb{E}(\|\hat{y}_{k|k-1}^{(i)^{\mathrm{T}}}\|^{2p})^{\frac{1}{2p}}, \\ &\leq \|\hat{x}_{1_{k|k-1}}^{(i)}\|_{2p}\|\hat{y}_{k|k-1}^{(i)^{\mathrm{T}}}\|_{2p}\end{aligned} \tag{6.67}$$

which yields

$$\begin{aligned}\|\check{P}_{k|k-1}^{xy}\|_p &\leq \frac{1}{N-1}(\|\hat{x}_{1_{k|k-1}}^{(i)}\|_{2p}\|\hat{y}_{k|k-1}^{(i)^{\mathrm{T}}}\|_{2p} + \|\hat{\bar{x}}_{1_{k|k-1}}\|_{2p}\|\hat{\bar{y}}_{k|k-1}^{\mathrm{T}}\|_{2p}) \\ &\leq \frac{2}{N-1}c_1(k-1,p,\epsilon)d_5(k-1,p,\epsilon).\end{aligned} \tag{6.68}$$

Similar to the derivations used in $\|\check{P}_{k|k-1}^{xy}\|_p$, for $\|\check{P}_{k|k-1}^{yy}\|_p$ we can also get

$$\begin{aligned}\|\check{P}_{k|k-1}^{yy}\|_p &\leq \frac{1}{N-1}(\|\hat{y}_{k|k-1}^{(i)}\hat{y}_{k|k-1}^{(i)^{\mathrm{T}}}\|_{2p} + \|\hat{\bar{y}}_{k|k-1}\hat{\bar{y}}_{k|k-1}^{\mathrm{T}})\|_{2p}) \\ &\leq \frac{1}{N-1}(\|\hat{y}_{k|k-1}^{(i)}\|_{2p}^2 + \|\hat{\bar{y}}_{k|k-1}\|_{2p}^2) \\ &\leq \frac{2}{N-1}d_5^2(k-1,p,\epsilon).\end{aligned} \tag{6.69}$$

Now to show the boundedness of $\check{K}_k^{\mathrm{s}}$, note that $P_{k|k-1}^{yy}$ is a symmetric and semi-positive definite

and R_k is a symmetric positive definite matrix. Therefore, we have

$$\|(\check{P}^{yy}_{k|k-1} + R_k)^{-1}\| \leq \|R_k^{-1}\| \leq \mathrm{cte}(k) \tag{6.70}$$

where $\mathrm{cte}(k)$ denotes a constant parameter at the time instant k. The inequality in (6.70) together with the bound on $\|\check{P}^{xy}_{k|k-1}\|_p$ according to (6.68) gives

$$\begin{aligned} \|\check{K}^{\mathrm{s}}_k\|_p &\leq \|\check{P}^{xy}_{k|k-1}\|_p cte(k) \\ &\leq \frac{2}{N-1} c_1(k-1,p,\epsilon) d_5(k-1,p,\epsilon) cte(k), \end{aligned} \tag{6.71}$$

where N is a sufficiently large number ($N \to \infty$).

Finally, to show the boundedness of the *a posteriori* slow state estimate, consider equation (6.52) that yields,

$$\begin{aligned} &\|x^{(i)}_{1_{k|k}}\|_p \leq \|x^{(i)}_{1_{k|k-1}}\|_p + \|\check{K}^{\mathrm{s}}_k \tilde{y}^{(i)}_{k|k-1}\|_p \\ &\leq c_1(k-1,p,\epsilon) + \epsilon\alpha(d_1(k-1,p,\epsilon) + d_3(k-1,p,\epsilon)) + \|\check{K}^{\mathrm{s}}_k\|_{2p} \|\hat{y}^{(i)}_{k|k-1}\|_{2p} \\ &\leq c_1(k-1,p,\epsilon) + \epsilon\alpha(d_1(k-1,p,\epsilon) + d_3(k-1,p,\epsilon)) \\ &+ \frac{2}{(N-1)} c_1(k-1,p,\epsilon) d_5^2(k-1,p,\epsilon) cte(k). \end{aligned} \tag{6.72}$$

Hence, by applying the Jensen's inequality [194] for any $\hat{x}^{(i)}_{1_{k|k}}$, we obtain

$$\|\frac{1}{N}\sum_{i=1}^{N} \hat{x}^{(i)}_{1_{k|k}}\|_p \leq \frac{1}{N}\sum_{i=1}^{N} \|\hat{x}^{(i)}_{1_{k|k}}\|_p,$$

which yields

$$\begin{aligned} \|\hat{\bar{x}}_{1_{k|k}}\|_p &\leq c_1(k-1,p,\epsilon) + \epsilon\alpha(d_1(k-1,p,\epsilon) + d_3(k-1,p,\epsilon)) \\ &+ \frac{2}{(N-1)} c_1(k-1,p,\epsilon) d_5^2(k-1,p,\epsilon) cte(k), \\ &\leq c_1(k,p,\epsilon). \end{aligned} \tag{6.73}$$

Now, we investigate the boundedness of the estimation error based on the error analysis associated with the *a posteriori* slow state estimation and the one that is obtained from the real reduced order system model. If we substitute $x_{2_{k-1}}$ in D_ϵ with $\psi_0(x_{1_{k-1}})$, the reduced

order model for estimating x_{s1_k} is obtained, where according to Lemma 6.1, and (6.16), $x_{1_k} = x_{s1_k} + O(\epsilon)$. In our developed TTS-EnKF filter, x_{s1_k} is estimated. Therefore, the estimation error is represented by,

$$e_{s1_k} = x_{s1_k} - \hat{\bar{x}}_{1_{k|k}},$$

where e_{s1_k} denotes the slow filter estimation error. Hence, an upper bound on this error can be obtained as

$$\begin{aligned}
\|e_{s1_k}\|_p &\leq \|x_{s1_k}\|_p + \|\hat{\bar{x}}_{1_{k|k}}\|_p + O(\epsilon), \\
&\leq \|x_{1_{k-1}}\|_p + \epsilon\alpha(\|f_1(x_{1_{k-1}}, x_{2_{k-1}})\|_p + \|g_1(x_{1_{k-1}}, x_{2_{k-1}})\omega_{1_k}\|_p) \\
&+ \epsilon O(\alpha^2) + O(\epsilon) + \|\hat{\bar{x}}_{1_{k|k}}\|_p, \\
&\leq 2c_1(k-1,p,\epsilon) + 2\epsilon\alpha(d_1(k-1,p,\epsilon) + d_3(k-1,p,\epsilon)) \\
&+ \frac{2}{(N-1)}c_1(k-1,p,\epsilon)d_5^2(k-1,p,\epsilon)cte(k) + O(\epsilon),
\end{aligned} \tag{6.74}$$

where the last inequality is obtained by applying Assumption 6.8 and replacing $\|\hat{\bar{x}}_{1_{k|k}}\|_p$ with the bound from (6.73). The error of $O(\epsilon)$ magnitude is added due to the resulting discritization error as shown in the derivation of D_z and considering that α is very close to 1. Now, as $N \to \infty$, if $c_1(k-1,p,\epsilon)d_5^2(k-1,p,\epsilon)cte(k) \ll N$, the term $\frac{2}{(N-1)}c_1(k-1,p,\epsilon)d_5^2(k-1,p,\epsilon)cte(k)$ will tend to zero, and one can approximate $2\epsilon\alpha(d_1(k-1,p,\epsilon) + d_3(k-1,p,\epsilon)) + O(\epsilon) = O(\epsilon)$. Consequently, the upper bound on the estimation error corresponding to the reduced order model is obtained as

$$\|e_{s1_k}\|_p \leq 2c_1(k-1,p,\epsilon) + O(\epsilon).$$

Now, the error in the estimation of x_{1_k} can be obtained as

$$e_{1_k} = x_{s1_k} + O(\epsilon) - \hat{\bar{x}}_{1_{k|k}}. \tag{6.75}$$

Similarly, the upper bound on e_{1_k} can be expressed as

$$\|e_{1_k}\|_p \leq 2c_1(k-1,p,\epsilon) + O(\epsilon).$$

In the next step, the boundedness of the *a posteriori* estimation of the fast states through the fast filter are investigated. Again by invoking induction we assume that $\|x^{(i)}_{1_{k-1|k-1}}\|_p \leq c_1(k-1,p,\epsilon)$ and $\|x^{(i)}_{2_{k-1|k-1}}\|_p \leq c_2(k-1,p,\epsilon)$. Using the same approach as in the slow filter, an upper bound on the estimated *a posteriori* fast state is obtained as

$$\begin{aligned}\|\hat{\tilde{x}}_{2_{k|k}}\|_p &\leq c_2(k-1,p,\epsilon) + \epsilon\alpha(d_2(k-1,p,\epsilon) + d_4(k-1,p,\epsilon)) \\ &+ \frac{2}{(N-1)}c_2(k-1,p,\epsilon)d_5^2(k-1,p,\epsilon)cte(k), \\ &\leq c_2(k,p,\epsilon).\end{aligned}$$

Therefore, the estimation error based on the discrepancy between the real x_{2_k} from D_ϵ and the estimated $\hat{\tilde{x}}_{2_{k|k}}$ can be expressed as,

$$e_{2_k} = x_{2_k} - \hat{\tilde{x}}_{2_{k|k}},$$

where e_{2_k} denotes the estimation error of the fast filter. Hence, an upper bound on this error can be obtained as

$$\begin{aligned}\|e_{2_k}\|_p &\leq \|x_{2_k}\|_p + \|\hat{\tilde{x}}_{2_{k|k}}\|_p, \\ &\leq \|x_{2_{k-1}}\|_p + \alpha(\|f_2(x_{1_{k-1}}, x_{2_{k-1}})\|_p + \|g_2(x_{1_{k-1}}, x_{2_{k-1}})\omega_{2_k}\|_p) + O(\alpha^2) + \|\hat{\tilde{x}}_{2_{k|k}}\|_p, \\ &\leq 2c_2(k-1,p,\epsilon) + \alpha(d_2(k-1,p,\epsilon) + d_4(k-1,p,\epsilon)) \\ &+ \epsilon\alpha(d_2(k-1,p,\epsilon) + d_4(k-1,p,\epsilon)) + \frac{2}{(N-1)}c_2(k-1,p,\epsilon)d_5^2(k-1,p,\epsilon)cte(k) \\ &+ O(\alpha^2), \\ &\leq 2c_2(k-1,p,\epsilon) + \alpha(d_2(k-1,p,\epsilon) + d_4(k-1,p,\epsilon)) + O(\alpha^2) + O(\epsilon).\end{aligned} \tag{6.76}$$

Finally, the error of the fast filter is propagated with the order of $O(\alpha)$, whereas for the slow

filter it is propagated with the order of $O(\epsilon)$. This completes the proof of the theorem. ■

In the next section, we have applied and extended our developed TTS-EnKF filter to the problem of long-term prediction of system states/health parameters for the health monitoring problem.

6.4 Prediction Scheme Based on Two-Time Scale EnKF

In this section, our previously developed two-time scale EnKF scheme is utilized for long-term prediction of the nonlinear system's slowly time-varying health parameters as well as its fast states. This problem is generally considered as the second module in development of an integrated framework for health monitoring of complex engineering systems.

6.4.1 Prediction Framework Based on the Two-Time Scale EnKF Strategy

The main challenge in the prediction problem is that the prediction errors increase as the prediction horizon is extended. This problem is directly related to absence of actual observations after the time instant k, so that information on actual observations cannot be used for reducing the resulting error covariances in the *a posteriori* state estimation process.

In our proposed framework, we follow the main idea of the EnKF which substitutes the actual states with their sequence of ensemble members to obtain the estimation error where we

now also replace the observation vector in the measurement update step with the approximated observation vector in both the slow and fast filters that result from the approximated observation ensembles.

Our proposed prediction scheme also consists of two filters, namely the slow and the fast filters for updating the health parameters as well as the states, respectively. Consequently, our proposed prediction algorithm based on the two-time scale EnKF scheme can be summarized as follows.

6.4.2 Prediction of the Slow States

Our prediction strategy is performed through two main steps (similar to the estimation strategy), namely time update and measurement update (based on the approximated measurements).

Time Update for the Slow Filter

The state vector for this filter is defined as $x_{\mathrm{s}_{1_k}}$. Consequently, the ensemble members are generated in the time update step from the following equations for the $l > 1$ step ahead prediction horizon

$$\hat{x}^{(i)^-}_{1_{k+l|k}} = \hat{x}^{(i)^+}_{1_{k+l-1|k}} + \iota f_1(\hat{x}^{(i)^+}_{1_{k+l-1|k}}, \psi_0(\hat{x}^{(i)^+}_{1_{k+l-1|k}}), 0) + \iota g_1(\hat{x}^{(i)^+}_{1_{k+l-1|k}}, \psi_0(\hat{x}^{(i)^+}_{1_{k+l-1|k}}), 0)\omega^{(i)}_{1_{k+l-1}}, \tag{6.77}$$

where $i = 1, ..., N$, and the superscript $(-)$ refers to the predicted state in the previous time step before performing the covariance correction in the measurement update step while the

superscript $(+)$ refers to the approximated value of the state after performing the measurement update step.

The most probable states and their corresponding ensemble perturbations are generated according to

$$\begin{aligned} \hat{\bar{x}}^{-}_{1_{k+l|k}} &= \frac{1}{N}\sum_{i=1}^{N} \hat{x}^{(i)^{-}}_{1_{k+l|k}}, \\ \delta\hat{x}^{(i)^{-}}_{1_{k+l|k}} &= \hat{x}^{(i)^{-}}_{1_{k+l|k}} - \hat{\bar{x}}^{-}_{1_{k+l|k}}. \end{aligned} \tag{6.78}$$

We now define the vector $\hat{X}^{-}_{\mathrm{s}_{1_{k+l|k}}} = \hat{X}^{-}_{1_{k+l|k}} = \frac{1}{\sqrt{N-1}}[\delta\hat{x}^{(1)^{-}}_{1_{k+l|k}}, ..., \delta\hat{x}^{(N)^{-}}_{1_{k+l|k}}]^{\mathrm{T}}$.

Measurement Update for the Slow Filter

For the measurement update step, as stated earlier due to absence of observations one requires to apply another alternative approach to reduce the prediction error in this step. We suggest to utilize the following approximation for the l-step ahead prediction of the observation vector, namely

$$y^{\mathrm{s}}_{k+l} \approx h_0(\hat{\bar{x}}^{-}_{1_{k+l|k}}, \psi_0(\hat{\bar{x}}^{-}_{1_{k+l|k}}), 0), \tag{6.79}$$

where y^{s}_{k+l} denotes the predicted observations of the slow filter. Hence, the approximated observation vector according to (6.79) is utilized in the slow filter to predict the system slow states. To summarize, the prediction scheme for the slow filter is performed according to the following steps:

1. The output perturbation matrix is computed from: $\hat{Y}^{\mathrm{s}}_{k+l|k} = [\delta\hat{y}^{\mathrm{s}^{(1)}}_{k+l|k}, ..., \delta\hat{y}^{\mathrm{s}^{(N)}}_{k+l|k}]^{\mathrm{T}}$, where for $i = 1, .., N$, $\delta\hat{y}^{\mathrm{s}^{(i)}}_{k+l|k} = h_0(\hat{x}^{(i)^{-}}_{1_{k+l|k}}, \psi_0(\hat{x}^{(i)^{-}}_{1_{k+l|k}}), 0) - \frac{1}{N}\sum_{i=1}^{N} h_0(\hat{x}^{(i)^{-}}_{1_{k+l|k}}, \psi_0(\hat{x}^{(i)^{-}}_{1_{k+l|k}}), 0)$,

2. The Kalman gain is computed from: $\check{K}^{\mathrm{s}}_{k+l} = \check{P}^{\mathrm{xy}}_{k+l|k}(\check{P}^{\mathrm{yy}}_{k+l|k} + R_k)^{-1}$, where $\check{P}^{\mathrm{xy}}_{k+l|k} = \hat{X}^{-}_{1_{k+l|k}}\hat{Y}^{\mathrm{s}^{\mathrm{T}}}_{k+l|k}$ and $\check{P}^{\mathrm{yy}}_{k+l|k} = \hat{Y}^{\mathrm{s}}_{k+l|k}\hat{Y}^{\mathrm{s}^{\mathrm{T}}}_{k+l|k}$.

3. The prediction of *a posteriori* state ensemble members is computed from:

$$\hat{x}^{(i)^{+}}_{1_{k+l|k}} = \hat{x}^{(i)^{-}}_{1_{k+l|k}} + \check{K}^{\mathrm{s}}_{k+l}\tilde{y}^{\mathrm{s}^{(\mathrm{i})}}_{k+l|k}, \tag{6.80}$$

   where $\tilde{y}^{\mathrm{s}^{(i)}}_{k+l|k} = y^{\mathrm{s}}_{k+l} - h_0(\hat{x}^{(i)^{-}}_{1_{k+l|k}}, \psi_0(\hat{x}^{(i)^{-}}_{1_{k+l|k}}), 0)$.

4. The most probable *a posteriori* state estimate is computed from: $\hat{\bar{x}}^{+}_{1_{k+l|k}} = \frac{1}{N}\sum_{i=1}^{N}\hat{x}^{(i)^{+}}_{1_{k+l|k}}$.

In the subsequent subsection, the prediction scheme for predicting the fast states of the system is provided in detail.

6.4.3 Prediction of the Fast States

The prediction scheme of this filter is also performed through two main steps, namely the time update and the measurement update where the ensemble perturbations update is also performed in this step.

Time Update for the Fast Filter

For this filter, the slow states of the system are considered as fixed and equal to their most probable predicted values obtained from the previous time step, i.e., $x_{1_{k+l}} \approx \hat{\bar{x}}^{(+)}_{1_{k+l-1|k}}$. Therefore,

the time update step is performed according to the following equations

$$\begin{aligned}
&\hat{x}_{1_{k+l|k}} = \hat{\bar{x}}^{(+)}_{1_{k+l-1|k}}, \\
&\hat{\bar{x}}^{(i)^-}_{2_{k+l|k}} = \hat{x}^{(i)^+}_{2_{k+l-1|k}} + \iota f_2(\hat{\bar{x}}^{(+)}_{1_{k+l-1|k}}, x^{(i)^+}_{2_{k+l-1|k}}) + \iota g_2(\hat{\bar{x}}^{(+)}_{1_{k+l-1|k}}, x^{(i)^+}_{2_{k+l-1|k}}) + w^{(i)}_{2_{k+l}},
\end{aligned} \tag{6.81}$$

where $\hat{x}^{(i)^+}_{2_{k+l-1|k}}$ denotes the predicted fast ensembles from the previous time step for $i = 1, ..., N$ members. We can now define the vector $\hat{X}^{\mathrm{f}^-}_{k+l|k} = \hat{X}^-_{2_{k+l|k}} = \frac{1}{\sqrt{N-1}}[\delta\hat{x}^{(1)^-}_{2_{k+l|k}}, ..., \delta\hat{x}^{(N)^-}_{2_{k+l|k}}]^{\mathrm{T}}$.

Measurement Update for the Fast Filter

For the measurement update step, similar to the slow filter we suggest to utilize an approximation for the l-step ahead prediction of the observation vector as follows,

$$y^{\mathrm{f}}_{k+l} \approx h_0(\hat{\bar{x}}^+_{1_{k+l-1|k}}, \hat{\bar{x}}^-_{2_{k+l|k}}), \tag{6.82}$$

where y^{f}_{k+l} denotes the predicted observations from the fast filter. Therefore, the approximated observation vector according to (6.82) is used in the fast filter to predict the system fast states according to the following steps:

1. The output perturbation matrix is computed from: $\hat{Y}^{\mathrm{f}}_{k+l|k} = [\delta\hat{y}^{\mathrm{f}^{(1)}}_{k+l|k}, ..., \delta\hat{y}^{\mathrm{f}^{(N)}}_{k+l|k}]^{\mathrm{T}}$, where for $i = 1, .., N$, $\delta\hat{y}^{\mathrm{f}^{(i)}}_{k+l|k} = h_0(\hat{\bar{x}}^+_{1_{k+l-1|k}}, \hat{x}^{(i)^-}_{2_{k+l|k}}) - \frac{1}{N}\sum_{i=1}^{N} h_0(\hat{\bar{x}}^+_{1_{k+l-1|k}}, \hat{x}^{(i)^-}_{2_{k+l|k}})$,

2. The Kalman gain is computed from: $K^{\mathrm{f}}_{k+l} = \breve{P}^{\mathrm{xy}}_{k+l|k}(\breve{P}^{\mathrm{yy}}_{k+l|k} + R_k)^{-1}$, where $\breve{P}^{\mathrm{xy}}_{k+l|k} = \hat{X}^-_{2_{k+l|k}}\hat{Y}^{\mathrm{f}^{\mathrm{T}}}_{k+l|k}$, and $\breve{P}^{\mathrm{yy}}_{k+l|k} = \hat{Y}^{\mathrm{f}}_{k+l|k}\hat{Y}^{\mathrm{f}^{\mathrm{T}}}_{k+l|k}$.

3. The prediction of *a posteriori* state ensemble members of fast states is computed from: $\hat{x}^{(i)^+}_{2_{k+l|k}} = \hat{x}^{(i)^-}_{2_{k+l|k}} + \breve{K}^{\mathrm{f}}_{k+l}\tilde{y}^{\mathrm{f}^{(i)}}_{k+l|k}$, where $\tilde{y}^{\mathrm{f}^{(i)}}_{k+l|k} = y^{\mathrm{f}}_{k+l} - h_0(\hat{\bar{x}}^+_{1_{k+l-1|k}}, \hat{x}^{(i)^-}_{2_{k+l|k}})$.

4. The most probable *a posteriori* state estimate is computed from: $\hat{\bar{x}}^+_{2_{k+l|k}} = \frac{1}{N}\sum_{i=1}^{N} \hat{x}^{(i)^+}_{2_{k+l|k}}$.

However, it is obvious that as the step-ahead prediction horizon is extended, errors in the prediction of the system states do also increase. The following theorem which is inspired from Theorem 6.2, provides bounds on the state estimation results for the TTS-EnKF as a function of the l-step ahead prediction horizon.

Theorem 6.3 *Let Assumption 6.8 and Theorem 6.2 results hold. The l-step ahead prediction error of the slow and fast states of the system that is provided in* (6.23) *and that utilize the TTS-EnKF scheme remains bounded with an error of the order of $(l+2)O(\epsilon)$ for $\hat{\bar{x}}^{+}_{1_{k+l|k}}$ and of the order $(l+1)O(\alpha^2)+(l+2)O(\epsilon)$ for $\hat{\bar{x}}^{+}_{2_{k+l|k}}$.*

Proof: From the prediction scheme it is known that the predicted state from the previous time step is utilized to predict the state in the next time instant. Therefore, the error due to discretization of the system as well as the fast-slow decomposition of the system states do propagate through the prediction algorithm to the future time steps. Therefore, by back substituting the predicted values of $\hat{x}^{(i)^-}_{1_{k+j|k}}$ and $\hat{x}^{(i)^-}_{2_{k+j|k}}$ for $j=0,...,l-1$ into equations (6.77) and (6.80) for the slow filter, we can obtain

$$\begin{aligned}
\hat{x}^{(i)^+}_{1_{k+l|k}} &= \hat{x}^{(i)^+}_{1_{k+l-1|k}} + \epsilon\alpha(f_1(\hat{x}^{(i)^+}_{1_{k+l-1|k}}, \psi_0(\hat{x}^{(i)^+}_{1_{k+l-1|k}}), 0) + g_1(\hat{x}^{(i)^+}_{1_{k+l-1|k}}, \psi_0(\hat{x}^{(i)^+}_{1_{k+l-1|k}}), 0)\omega^{(i)^+}_{1_{k+l}}) \\
&\quad + \check{K}^{\mathrm{s}}_{k+l}\tilde{y}^{\mathrm{s}^{(\mathrm{i})}}_{k+l|k} \\
&= \hat{x}^{(i)^+}_{1_{k|k}} + \epsilon\alpha\sum_{j=0}^{l-1}(f_1(\hat{x}^{(i)^+}_{1_{k+j|k}}, \psi_0(\hat{x}^{(i)^+}_{1_{k+j|k}}), 0) + g_1(\hat{x}^{(i)^+}_{1_{k+j|k}}, \psi_0(\hat{x}^{(i)^+}_{1_{k+j|k}}), 0)\omega^{(i)}_{k+j+1}) \\
&\quad + \sum_{j=1}^{l}\check{K}^{\mathrm{s}}_{k+j}\tilde{y}^{\mathrm{s}^{(\mathrm{i})}}_{k+j|k}
\end{aligned}$$

Now, a bound on the predicted state $\hat{x}^{(i)^+}_{1_{k+l|k}}$ and prediction error can be obtained by considering the Assumption 6.8 and Theorem 6.2 as follows

$$\begin{aligned}
\|\hat{x}^{(i)^+}_{1_{k+l|k}}\| &\leq c_1(k,p,\epsilon) + \alpha\epsilon \sum_{j=0}^{l-1}(d_1(k+j,p,\epsilon) + d_3(k+j,p,\epsilon)) \\
&\quad + \sum_{j=1}^{l} \frac{2}{(N-1)} c_1(k+j,p,\epsilon) d_5^2(k+j,p,\epsilon) cte(k+j) \\
\|e_{\mathrm{s}_{1_{k+l}}}\| &\leq c_1(k+l-1,p,\epsilon) + c_1(k,p,\epsilon) + \epsilon\alpha(d_1(k+l-1,p,\epsilon) + d_3(k+l-1,p,\epsilon)) \\
&\quad + \alpha\epsilon \sum_{j=0}^{l-1}(d_1(k+j,p,\epsilon) + d_3(k+j,p,\epsilon)) \\
&\quad + \sum_{j=1}^{l} \frac{2}{(N-1)} c_1(k+j,p,\epsilon) d_5^2(k+j,p,\epsilon) cte(k+j) + O(\epsilon),
\end{aligned}$$

assuming $l \ll N$, and the bound on the prediction error corresponding to the slow states is given by

$$\|e_{\mathrm{s}_{1_{k+l}}}\| \leq c_1(k+l-1,p,\epsilon) + c_1(k,p,\epsilon) + (l+2)O(\epsilon).$$

The same procedure can be applied to $\hat{x}^{(i)^+}_{2_{k+l|k}}$ to obtain the higher bound on the l-step ahead

prediction as follows

$$
\begin{aligned}
\hat{x}^{(i)^+}_{2_{k+l|k}} &= \hat{x}^{(i)^+}_{2_{k+l-1|k}} + \alpha\epsilon(f_2(\hat{\bar{x}}^{+}_{1_{k+l-1|k}}, \hat{x}^{(i)^+}_{2_{k+l-1|k}}) + g_2(\hat{\bar{x}}^{+}_{1_{k+l-1|k}}, \hat{x}^{(i)^+}_{2_{k+l-1|k}})\omega^{(i)}_{2_{k+l}}) + \check{K}^{\mathrm{f}}_{k+l}\tilde{y}^{\mathrm{f}(\mathrm{i})}_{k+l|k} \\
&= \hat{x}^{(i)^+}_{2_{k|k}} + \alpha\epsilon\sum_{j=0}^{l-1}(f_2(\hat{\bar{x}}^{+}_{1_{k+j|k}}, \hat{x}^{(i)^+}_{2_{k+j|k}}) + g_2(\hat{\bar{x}}^{+}_{1_{k+j|k}}, \hat{x}^{(i)^+}_{2_{k+j|k}})\omega^{(i)}_{2_{k+j}}) + \sum_{j=1}^{l}\check{K}^{\mathrm{f}}_{k+j}\tilde{y}^{\mathrm{f}(\mathrm{i})}_{k+j|k}, \\
&\leq c_2(k,p,\epsilon) + \alpha\epsilon\sum_{j=0}^{l-1}(d_2(k+j,p,\epsilon) + d_4(k+j,p,\epsilon)) \\
&+ \sum_{j=1}^{l}\frac{2}{N-1}c_2(k+j,p,\epsilon)d_5^2(k+j,p,\epsilon)cte(k+j).
\end{aligned}
$$

$$
\begin{aligned}
\|e_{2_{k+l}}\| &\leq c_2(k+l-1,p,\epsilon) + c_2(k,p,\epsilon) + \alpha(d_2(k+l-1,p,\epsilon) + d_4(k+l-1,p,\epsilon)) + O(\epsilon) \\
&+ O(\alpha^2) + \alpha\epsilon\sum_{j=0}^{l-1}(d_2(k+j,p,\epsilon) + d_4(k+j,p,\epsilon)) \\
&+ \sum_{j=1}^{l}(\frac{2}{(N-1)}c_2(k+j,p,\epsilon)d_5^2(k+j,p,\epsilon)cte(k+j) + O(\epsilon) + O(\alpha^2)), \\
&\leq c_2(k+l-1,p,\epsilon) + c_2(k,p,\epsilon) + \alpha(d_2(k+l-1,p,\epsilon) + d_4(k+l-1,p,\epsilon)) \\
&+ (l+2)O(\epsilon) + (l+1)O(\alpha^2)
\end{aligned}
$$

Therefore, the highest bounds on the l-step ahead predicted states as well as the order of the propagated error as a function of the prediction horizon are obtained. This completes the proof of the theorem. ■

The results from the Theorem 6.3 show that although the l-step prediction of the system states remains bounded for a bounded l, however as the prediction horizon extends the errors due to approximation of the exact system into slow and fast subsystems cause additional errors in the resulting predictions. Therefore, the prediction horizon should be chosen carefully such that ignoring the slow-fast decomposition errors in the developed TTS-EnKF scheme cannot be significant. In Section 6.5, our developed TTS-EnKF estimation/prediction strategies are ap-

plied to track the degradation phenomenon and its propagation prediction for long-term horizon in a gas turbine engine system.

In addition to the prediction accuracy of our methods that have been developed for nonlinear systems based on the nonlinear filtering strategies, the computational cost of our developed schemes when implemented is also an important issue that should be investigated to determine a trade-off between the accuracy and the cost. In the next subsection, the computational cost of our developed TTS-EnKF method is quantified and is compared with the well-known particle filtering approach in prediction applications.

6.4.4 Complexity Analysis of the TTS-EnKF Estimation/Prediction Schemes

In this subsection, the computational complexity of our proposed TTS-EnKF estimation/prediction schemes are quantitatively obtained and analyzed. The analysis is based on the number of floating-point operations (flops) that are required by the analyzed algorithm, known as equivalent flop (EF) analysis. The dimension and definitions of certain entities that are used in the EF analysis of our proposed TTS-EnKF state estimation and prediction schemes are provided in Table 6.1. Given that the computational complexity of certain common matrix manipulations as given in Chapter 3, the EF complexity of our proposed scheme is now summarized in Table 6.2 for the slow state estimation/prediction module and in Table 6.3 for the fast state estimation/prediction module. The EF quantities in the two tables correspond to only one iteration of the scheme. The coefficient c_1 is used to represent the complexity of the random number generation.

Table 6.1: Definition and Dimension of the entities in the TTS-EnKF estimation/prediction Schemes

Variable	Dimension	Definition
x_{1_k}	$\mathbb{R}^{n_s}$	slow state vector
x_{2_k}	$\mathbb{R}^{n_f}$	fast state vector
e_{1_k}	$\mathbb{R}^{n_s}$	modeling error to the slow state
e_{2_k}	$\mathbb{R}^{n_f}$	modeling error to the fast state
$\breve{K}_k^{\mathrm{s}}$	$\mathbb{R}^{n_s \times n_y}$	Kalman gain for the slow filter
$\breve{K}_k^{\mathrm{f}}$	$\mathbb{R}^{n_f \times n_y}$	Kalman gain for the fast filter
y_t	$\mathbb{R}^{n_y}$	measurement output
$\mathfrak{f}_1(.)$	$\mathbb{R}^{n_s \times 1}$	slow state dynamic function
$\mathfrak{f}_2(.)$	$\mathbb{R}^{n_f \times 1}$	fast state dynamic function
$h(.)$	$\mathbb{R}^{n_y \times 1}$	observation function

Table 6.2: The Equivalent Complexity for the slow states estimation/prediction step for the TTS-EnKF scheme.

Instruction	Mult.	Add	Func. Eval.	Other
$\omega_{k+l}^{(i)} = [\omega_{1_{k+l}}^{(i)}, \omega_{2_{k+l}}^{(i)}]^{\mathrm{T}} = \sqrt{Q_k}\mathrm{randn}(n_s+n_f, N)$	$(n_s+n_f)^2 N$	$-$	$-$	$N(n_s+n_f)c_1$
$\hat{x}_{1_{k+l\|k}}^{(i)^-} = \hat{x}_{1_{k+l-1\|k}}^{(i)^+} + \iota(f_1(\hat{x}_{1_{k+l-1\|k}}^{(i)^+}, \psi_0(\hat{x}_{1_{k+l-1\|k}}^{(i)^+}), 0) + g_1(\hat{x}_{1_{k+l-1\|k}}^{(i)^+}, \psi_0(\hat{x}_{1_{k+l-1\|k}}^{(i)^+}), 0)\omega_{k+1}^{(i)})$	Nn_s	Nn_s	$N(n_s+n_f)$	$-$
$\delta\hat{x}_{1_{k+l\|k}}^{(i)^-} = \hat{x}_{1_{k+l\|k}}^{(i)^-} - \hat{\bar{x}}_{1_{k+l\|k}}^{-}$, $\hat{X}_{1_{k+l\|k}}^{-} = \frac{1}{\sqrt{N-1}}[\delta\hat{x}_{1_{k+l\|k}}^{(1)^-}, ..., \delta\hat{x}_{1_{k+l\|k}}^{(N)^-}]$	Nn_s	Nn_s	$-$	$-$
$\delta\hat{y}_{k+l\|k}^{\mathrm{s}(i)} = h_0(\hat{x}_{1_{k+l\|k}}^{(i)^-}, \psi_0(\hat{x}_{1_{k+l\|k}}^{(i)^-}), 0) - \frac{1}{N}\sum_{i=1}^{N} h_0(\hat{x}_{1_{k+l\|k}}^{(i)^-}, \psi_0(\hat{x}_{1_{k+l\|k}}^{(i)^-}), 0)$, $\hat{Y}_{k+l\|k}^{\mathrm{s}} = \frac{1}{\sqrt{N-1}}[\delta\hat{y}_{k+l\|k}^{\mathrm{s}(1)}, ..., \delta\hat{y}_{k+l\|k}^{\mathrm{s}(N)}]$	n_y	$2Nn_y$	Nn_y	$-$
$\check{P}_{k+l\|k}^{xy} = \hat{X}_{1_{k+l\|k}}^{-}\hat{Y}_{k+l\|k}^{\mathrm{s}\mathrm{T}}$	Nn_sn_y	$(N-1)n_sn_y$	$-$	$-$
$\check{P}_{k+l\|k}^{yy} = \hat{Y}_{k+l\|k}^{\mathrm{s}}\hat{Y}_{k+l\|k}^{\mathrm{s}\mathrm{T}}$	Nn_y^2	$(N-1)n_y$	$-$	$-$
$\breve{K}_{k+l}^{\mathrm{s}} = \check{P}_{k+l\|k}^{xy}(\check{P}_{k+l\|k}^{yy} + R_k)^{-1}$	$n_y^3 + n_sn_y^2$	$n_y^2 + n_sn_y(n_y-1)$	$-$	$-$
$\tilde{y}_{k+l\|k}^{\mathrm{s}(i)} = h_0(\hat{\bar{x}}_{1_{k+l\|k}}^{-}, \psi_0(\hat{\bar{x}}_{1_{k+l\|k}}^{-}), 0) - h_0(\hat{x}_{1_{k+l\|k}}^{(i)^-}, \psi_0(\hat{x}_{1_{k+l\|k}}^{(i)^-}), 0)$	$-$	n_y	n_y+n_f	$-$
$\hat{x}_{1_{k+l\|k}}^{(i)^+} = \hat{x}_{1_{k+l\|k}}^{-} + \breve{K}_{k+l\|k}^{\mathrm{s}}\tilde{y}_{k+l\|k}^{\mathrm{s}(i)}$	Nn_sn_y	n_sN	$-$	$-$
$\hat{\bar{x}}_{1_{k+l\|k}}^{+} = \frac{1}{N}\sum_{i=1}^{N}\hat{x}_{1_{k+l\|k}}^{(i)^+}$	n_s	Nn_s	$-$	$-$
Total	$N(n_s^2+n_f^2+n_y^2+2n_sn_f$ $+2n_sn_y+2n_s)+n_y^3$ $+2n_sn_y^2+n_y+2n_s$	$N(6n_s+4n_y$ $+n_sn_y)+n_y^2-n_y$ $+n_sn_y^2-2n_sn_y$	$N(n_s+n_f$ $+n_y)+n_y$ $+n_f$	$Nc_1(n_s+n_f)$

Table 6.3: The Equivalent Complexity for the fast state estimation/prediction step for the TTS-EnKF scheme.

Instruction	Mult.	Add	Func. Eval.	Other
$\hat{x}_{2_{k+l\|k}}^{(i)^-} = \hat{x}_{2_{k+l-1\|k}}^{(i)^+} + \iota(f_2(\hat{\bar{x}}_{1_{k+l-1\|k}}^{+}, \hat{x}_{2_{k+l-1\|k}}^{(i)^+}) + g_2(\hat{\bar{x}}_{1_{k+l-1\|k}}^{+}, \hat{x}_{2_{k+l-1\|k}}^{(i)^+})w_{2_{k+l}}^{(i)})$	$2Nn_f$	$2Nn_f$	Nn_f	$-$
$\delta\hat{x}_{2_{k+l\|k}}^{(i)^-} = \hat{x}_{2_{k+l\|k}}^{(i)^-} - \hat{\bar{x}}_{2_{k+l\|k}}^{-}$, $\hat{X}_{2_{k+l\|k}}^{-} = \frac{1}{\sqrt{N-1}}[\delta\hat{x}_{2_{k+l\|k}}^{(1)^-}, ..., \delta\hat{x}_{2_{k+l\|k}}^{(N)^-}]$	Nn_f	Nn_f	$-$	$-$
$\delta\hat{y}_{k+l\|k}^{\mathrm{f}(i)} = h_0(\hat{\bar{x}}_{1_{k+l-1\|k}}^{+}, \hat{x}_{2_{k+l\|k}}^{(i)^-}) - \frac{1}{N}\sum_{i=1}^{N}\hat{y}_{k+l\|k}^{\mathrm{f}(i)}$, $\hat{Y}_{k+l\|k}^{\mathrm{f}} = \frac{1}{\sqrt{N-1}}[\delta\hat{y}_{k+l\|k}^{\mathrm{f}(1)}, ..., \delta\hat{y}_{k+l\|k}^{\mathrm{f}(N)}]$	n_y	$2Nn_y$	Nn_y	$-$
$\check{P}_{k+l\|k}^{xy} = \hat{X}_{2_{k+l\|k}}^{-}\hat{Y}_{k+l\|k}^{\mathrm{f}\mathrm{T}}$	Nn_fn_y	$2(N-1)n_fn_y$	$-$	$-$
$\check{P}_{k+l\|k}^{yy} = \hat{Y}_{k+l\|k}^{\mathrm{f}}\hat{Y}_{k+l\|k}^{\mathrm{f}\mathrm{T}}$	Nn_y^2	$(N-1)n_y$	$-$	$-$
$\breve{K}_{k+l}^{\mathrm{f}} = \check{P}_{k+l\|k}^{xy}(\check{P}_{k+l\|k}^{yy} + R_k)^{-1}$	$n_y^3 + n_fn_y^2$	$n_y^2 + n_fn_y(n_y-1)$	$-$	$-$
$\tilde{y}_{k+l\|k}^{\mathrm{f}(i)} = h_0(\hat{\bar{x}}_{1_{k+l-1\|k}}^{+}, \hat{x}_{2_{k+l\|k}}^{-}) - h_0(\hat{\bar{x}}_{1_{k+l-1\|k}}^{+}, \hat{x}_{2_{k+l\|k}}^{(i)^-})$	n_f	$N(n_f+n_y)$	n_y	$-$
$\hat{x}_{2_{k+l\|k}}^{(i)^+} = \hat{x}_{1_{k+l\|k}}^{-} + \breve{K}_{k+l\|k}^{\mathrm{f}}\tilde{y}_{k+l\|k}^{\mathrm{f}(i)}$	n_fn_yN	n_fN	$-$	$-$
$\hat{\bar{x}}_{k+l\|k}^{\mathrm{f}+} = \frac{1}{N}\sum_{i=1}^{N}\hat{x}_{k+l\|k}^{\mathrm{f}(i)^+}$	n_f	n_fN	$-$	$-$
Total	$N(3n_f+2n_fn_y+n_y^2)$ $+n_y^3+n_fn_y^2+2n_f$ $+n_y$	$N(6n_f+3n_y+n_fn_y)$ $+n_fn_y^2+n_y^2-2n_fn_y-n_s$	$N(n_f+n_y)$ $+n_y$	$-$

Table 6.4: The Total Equivalent Complexity of the Filters

Prediction Method	Total Equivalent Complexity
DLM-Based Prediction Method [185]	$C_A(n_s,n_f,c_1,c_2,c_3,c_4,N) \approx N(3n_s^2+5n_f^2+6n_f+2n_fn_y+7n_y+3n_s$ $+c_1(n_s+n_f)+c_2(n_s+n_f)+c_3n_s)$
Standard PF-Based Prediction Method [25]	$C_B(n_s,n_f,c_1,c_3,N) \approx N(3n_s^2+3n_f^2+6n_sn_f+(1+c_1+c_3)n_s$ $+(1+c_1+c_3)n_f+n_y)$
TTS-EnKF Prediction Method (this work)	$C_C(n_s,n_f,c_1,N) \approx N(n_s^2+n_f^2+2n_y^2+2n_sn_f+3n_sn_y+3n_fn_y$ $+(9+c_1)n_s+(11+c_1)n_f+9n_y)$

Our goal here is to develop a comprehensive measure and comparison between the complexity of our proposed TTS-EnKF prediction algorithm with other commonly used and well-known particle filtering (PF) estimation/prediction schemes [55, 185]. Towards this end, the complexity of the particle filters prediction algorithm with a regularized structure is also presented in Table 6.4. This has already been used for prediction purposes in various applications as in [25]. We have also included our previously developed prediction algorithm that is based on combination of the particle filters with dynamically linear models (DLM) [185]. In Table 6.4, where c_1 denotes the complexity of the random number generation, c_2 denotes the complexity of the resampling step of the particle filtering algorithm, c_3 denotes the complexity corresponding to the regularization step of the particle filtering algorithm, and c_4 denotes the complexity corresponding to the DLM models construction. The EF complexity of the DLM-based prediction method, the standard PF-based prediction method, and the TTS-EnKF prediction method are denoted by $C_A(n_s, n_f, c_1, c_2, c_3, c_4, N)$, $C_B(n_s, n_f, c_1, c_3, N)$, and $C_C(n_s, n_f, c_1, N)$, respectively. In the above first two methods N represents the number of particles, whereas in the last method N represents the number of ensembles that are chosen in the TTS-EnKF approach.

From the results that are presented in this table, it follows that the PF-based prediction methods yield a computationally more intensive implementation cost. This is quantified by the EF complexity (which is a measure of the algorithm time complexity) due to presence of resampling (c_2) and/or regularization (c_3) steps that deal with ordering as one of the most complex implementation procedures [170].

6.5 Development of a Health Monitoring of a Gas Turbine Engine

The considered application of our proposed two-time scale EnKF method for health monitoring and prognosis of a gas turbine engine is presented in this section. The approach can be used for failure prognosis of the engine, when the system is assumed to be affected by health degradation phenomenon. Our proposed prediction scheme is demonstrated to be capable of tracking the system health parameters that enjoy a slow time dynamics as compared to the other gas turbine dynamical system states that enjoy a fast time dynamics. Moreover, the performance of our proposed two-time scale EnKF method is evaluated and investigated under a general degradation scenario in the turbine component due to the erosion phenomenon. The main idea of our method is to model the dynamics of the system health parameters and augment them to the gas turbine system state equations to achieve a more accurate estimation as well as prediction results. Therefore, the gas turbine engine model as presented in [157] is modified in this work to include the dynamical model that is associated with the system health parameters.

6.5.1 Overall Model Overview

The formulation for degradation damage of a gas turbine engine is now proposed as follows. In this new methodology the system health parameters, which have a slowly time-varying behavior (due to the fault vector), are modeled as state variables with slow dynamics. The most important aspect of this modeling process is that the degradation is assumed to have started from the

starting time of the engine/turbine operation. This assumption is not very restrictive since a real engine is subjected to various types of degradation (such as erosion) from the first initiation of its operation that propagate during the time.

For the class of nonlinear systems that we are investigating here (the gas turbine application), the health parameters of the system are denoted by θ and are considered to be smooth functions of the system states (fast states) and time, i.e., $\theta(x,t)$. The effects of the degradation is modeled by a multiplicative time-varying vector function, $k(t,\epsilon)$, known as a fault vector, where $0 < \epsilon \ll 1$, is a sufficiently small parameter that quantifies the time-scale separation. In other words, the health parameter is represented by

$$\theta(x,t) = k(t,\epsilon)\theta_1(x(t)), \tag{6.83}$$

where $\theta_1(x(t))$ is a smooth function of x. The function $k(t,\epsilon) \in \mathcal{C}^2$ has an asymptotic power series expansion of ϵ [1] [212], i.e., for $k(t,\epsilon)$ and its first derivative we have,

$$\begin{aligned} k(t,\epsilon) &= k^0(t,0) + \epsilon\frac{\partial}{\partial\epsilon}k(t,0) + O(\epsilon^2), k^0(t,0) = 0, \\ \dot{k}(t,\epsilon) &= \dot{k}^0(t,0) + \epsilon\frac{\partial}{\partial\epsilon}\dot{k}(t,0) + O(\epsilon^2), \dot{k}^0(t,0) = 0, \end{aligned} \tag{6.84}$$

where $\dot{k} = \frac{\partial}{\partial t}k(t,\epsilon)$. Hence, the dynamics of the health parameters that are augmented to the system state equations are obtained as,

$$\dot{\theta}(x,t) = \dot{k}(t,\epsilon)\theta_1(x(t)) + k(t,\epsilon)\frac{\partial\theta_1}{\partial x}\dot{x}. \tag{6.85}$$

By considering the expansions given in equation (6.84), the system state equations including the augmented health parameter states, can be represented in the standard singularly perturbed form by introducing a new time variable $\tau = \epsilon t$, as follows

$$\begin{aligned} \epsilon\frac{dx}{d\tau} &= f(x,\theta,\epsilon,\tau), \\ \frac{d\theta}{d\tau} &= g(x,\theta,\epsilon,\tau), \end{aligned} \tag{6.86}$$

[1] A function $f(\epsilon)$ has an asymptotic power series expansion if as $\epsilon \to 0$, $f(\epsilon) \approx \sum_{j=0}^{\infty} f_j\epsilon^j$

where the time derivatives are taken with respect to τ, $x \in \mathbb{R}^{n_x}$, $\theta \in \mathbb{R}^{n_\theta}$ and $f : \mathbb{R}^{n_x} \times \mathbb{R}^{n_\theta} \times \mathbb{R}^2 \to \mathbb{R}^{n_x}$, $g : \mathbb{R}^{n_x} \times \mathbb{R}^{n_\theta} \times \mathbb{R}^2 \to \mathbb{R}^{n_\theta}$ belong to $\mathcal{C}^2$. In the following simulation scenarios that are conducted the effects of the turbine erosion degradation on the gas turbine system health propagation are investigated. Therefore, the dynamics of the mass flow capacity and efficiency of the turbine are augmented to the system state equations.

The mathematical model of a gas turbine that is used in this work is a single spool jet engine that was developed in [157] and presented in Chapter 2. The four engine states are the combustion chamber pressure and temperature, $P_{\rm CC}$ and $T_{\rm CC}$, respectively, the spool speed S, and the nozzle outlet pressure $P_{\rm NLT}$. The continuous-time state space model of the gas turbine is given as follows,

$$\begin{aligned}
\dot{T}_{\rm CC} &= \frac{1}{c_{\rm v} m_{\rm cc}}[(c_{\rm p} T_{\rm C} m_{\rm C} + \eta_{\rm CC} H_{\rm u} m_{\rm f} - c_{\rm p} T_{\rm CC} \theta_{m_{\rm T}}) - c_{\rm v} T_{\rm CC}(m_{\rm C} + m_{\rm f} - \theta_{m_{\rm T}})], \\
\dot{S} &= \frac{\eta_{\rm mech} \theta_{m_{\rm T}} c_{\rm p}(T_{\rm CC} - T_{\rm T}) - m_{\rm C} c_{\rm p}(T_{\rm C} - T_{\rm d})}{{\rm J} S (\frac{\pi}{30})^2}, \\
\dot{P}_{\rm CC} &= \frac{P_{\rm CC}}{T_{\rm CC}} \frac{1}{c_{\rm v} m_{\rm cc}}[(c_{\rm p} T_{\rm C} m_{\rm C} + \eta_{\rm CC} H_{\rm u} m_{\rm f} - c_{\rm p} T_{\rm CC} \theta_{m_{\rm T}}) - c_{\rm v} T_{\rm CC}(m_{\rm C} + m_{\rm f} - \theta_{m_{\rm T}})] \\
&+ \frac{\gamma {\rm R} T_{\rm CC}}{V_{\rm CC}}(m_{\rm C} + m_{\rm f} - \theta_{m_{\rm T}}), \\
\dot{P}_{\rm NLT} &= \frac{T_{\rm M}}{V_{\rm M}}(\theta_{m_{\rm T}} + \frac{\beta}{\beta + 1} m_{\rm C} - m_{\rm Nozzle}),
\end{aligned} \tag{6.87}$$

The five gas turbine measured outputs are considered to be the compressor temperature (y_1), the combustion chamber pressure (y_2), the spool speed (y_3), the nozzle outlet pressure (y_4), and the turbine temperature (y_5), namely

$$\begin{aligned}
y_1 &= T_{\rm C} = T_{\rm diffuser}[1 + \frac{1}{\eta_{\rm C}}[(\frac{P_{\rm CC}}{P_{\rm diffuzer}})^{\frac{\gamma-1}{\gamma}} - 1]], \\
y_2 &= P_{\rm CC},\ y_3 = S,\ y_4 = P_{\rm NLT}, \\
y_5 &= T_{\rm T} = T_{\rm CC}[1 - \theta_{\eta_{\rm T}}(1 - (\frac{P_{\rm NLT}}{P_{\rm CC}})^{\frac{\gamma-1}{\gamma}})].
\end{aligned} \tag{6.88}$$

By augmenting the turbine health parameters to the system state equations we obtain

$$\dot{\theta}_{\eta_{\rm T}} = \dot{k}_1(t,\epsilon)\eta_{\rm T}(S,P_{\rm CC}) + k_1(t,\epsilon)(\tfrac{\partial \eta_{\rm T}}{\partial S}\dot{S} + \tfrac{\partial \eta_{\rm T}}{\partial \beta}\dot{\beta}), \tag{6.89}$$

$$\dot{\theta}_{m_{\rm T}} = \dot{k}_2(t,\epsilon)m_{\rm T}(S,P_{\rm CC}) + k_2(t,\epsilon)(\tfrac{\partial m_{\rm T}}{\partial S}\dot{S} + \tfrac{m_{\rm T}}{\partial \beta}\dot{\beta}). \tag{6.90}$$

where the physical significance of all the above model parameters is provided in Table 2.1, and the functions $k_1(.)$ and $k_2(.)$ model manifestation in the turbine health parameters due to erosion and are considered as polynomial functions with asymptotic series expansion of ϵ. These functions are chosen as $k_1(t,\epsilon) = 1 - \epsilon t$ and $k_2(t,\epsilon) = 1 + 0.5\epsilon t$, in order to model the erosion degradation as a linear degradation model [213]. The functions $\eta_{\rm T}(S,\beta)$ and $m_{\rm T}(S,P_{\rm CC})$ are obtained as polynominal functions by curve-fitting from the turbine performance maps as utilized in [157] as follows,

$$\begin{aligned}
\eta_{\rm T}(S,\beta) &= 1.31 - 2.622(\frac{S}{S_{\rm ref}}) + 0.3739\beta + 3.691(\frac{S}{S_{\rm ref}})^2 + 0.3125\frac{S}{S_{\rm ref}}\beta - 1.119\beta^2 \\
&\quad - 2.076(\frac{S}{S_{\rm ref}})^3 + 1.36(\frac{S}{S_{\rm ref}})^2\beta - 1.254(\frac{S}{S_{\rm ref}})\beta^2 + 0.9428\beta^3 \\
m_{\rm T}(S,\beta) &= 2.765 + 1.779(\frac{S}{S_{\rm ref}}) + 11.49\beta - 0.6761(\frac{S}{S_{\rm ref}})^2 - 4.16(\frac{S}{S_{\rm ref}})\beta - 20.85\beta^2 \\
&\quad + 0.4114(\frac{S}{S_{\rm ref}})^2\beta + 3.509(\frac{S}{S_{\rm ref}})\beta^2 + 17.99\beta^3 + 0.3475(\frac{S}{S_{\rm ref}})^2\beta^2 \\
&\quad - 1.655(\frac{S}{S_{\rm ref}})\beta^2 - 5.815\beta^4
\end{aligned} \tag{6.91}$$

where $S_{\rm ref}$ is a reference design parameter chosen as 12000, and the bypass ratio β is computed from the following function based on the surface-fitting of the turbine performance maps in [157]

$$\begin{aligned}
\beta(S,P_{\rm CC},P_{\rm NLT}) = &- 0.1107 - 0.9083(\frac{S}{S_{\rm ref}}) + 0.4225(\frac{P_{\rm CC}}{P_{\rm NLT}}) + 0.4334(\frac{S}{S_{\rm ref}})^2 \\
&- 0.09009(\frac{S}{S_{\rm ref}})(\frac{P_{\rm CC}}{P_{\rm NLT}}) - 0.002457(\frac{P_{\rm CC}}{P_{\rm NLT}})^2
\end{aligned} \tag{6.92}$$

In order to model the overall gas turbine engine state equations with the turbine health parameters in the two-time scale framework, it is assumed that $\tau = \epsilon t$, so that one can rewrite

the system equation (6.87) as

$$\begin{aligned}
\epsilon\frac{\mathrm{d}T_{\mathrm{CC}}}{\mathrm{d}\tau} &= \frac{1}{c_{\mathrm{v}}m_{\mathrm{cc}}}[(c_{\mathrm{p}}T_{\mathrm{C}}m_{\mathrm{C}} + \eta_{\mathrm{CC}}H_{\mathrm{u}}m_{\mathrm{f}} - c_{\mathrm{p}}T_{\mathrm{CC}}\theta_{m_{\mathrm{T}}}) - c_{\mathrm{v}}T_{\mathrm{CC}}(m_{\mathrm{C}} + m_{\mathrm{f}} - \theta_{m_{\mathrm{T}}})],\\
\epsilon\frac{\mathrm{d}S}{\mathrm{d}\tau} &= \frac{\eta_{\mathrm{mech}}\theta_{m_{\mathrm{T}}}c_{\mathrm{p}}(T_{\mathrm{CC}} - T_{\mathrm{T}}) - m_{\mathrm{C}}c_{\mathrm{p}}(T_{\mathrm{C}} - T_{\mathrm{d}})}{\mathrm{J}S(\frac{\pi}{30})^2},\\
\epsilon\frac{\mathrm{d}P_{\mathrm{CC}}}{\mathrm{d}\tau} &= \frac{P_{\mathrm{CC}}}{T_{\mathrm{CC}}}\frac{1}{c_{\mathrm{v}}m_{\mathrm{cc}}}[(c_{\mathrm{p}}T_{\mathrm{C}}m_{\mathrm{C}} + \eta_{\mathrm{CC}}H_{\mathrm{u}}m_{\mathrm{f}} - c_{\mathrm{p}}T_{\mathrm{CC}}\theta_{m_{\mathrm{T}}}) - c_{\mathrm{v}}T_{\mathrm{CC}}(m_{\mathrm{C}} + m_{\mathrm{f}} - \theta_{m_{\mathrm{T}}})]\\
&+ \frac{\gamma\mathrm{R}T_{\mathrm{CC}}}{V_{\mathrm{CC}}}(m_{\mathrm{C}} + m_{\mathrm{f}} - \theta_{m_{\mathrm{T}}}),\\
\epsilon\frac{\mathrm{d}P_{\mathrm{NLT}}}{\mathrm{d}\tau} &= \frac{T_{\mathrm{M}}}{V_{\mathrm{M}}}(\theta_{m_{\mathrm{T}}} + \frac{\beta}{\beta+1}m_{\mathrm{C}} - m_{\mathrm{Nozzle}}).
\end{aligned} \tag{6.93}$$

Similarly associated with the turbine health parameters we have

$$\begin{aligned}
\frac{\mathrm{d}\theta_{\eta_{\mathrm{T}}}}{\mathrm{d}\tau} &= -\epsilon\eta_{\mathrm{T}}(S,\beta) + (1-\tau)(\frac{\partial\eta_{\mathrm{T}}}{\partial S}\frac{\mathrm{d}S}{\mathrm{d}\tau} + \frac{\partial\eta_{\mathrm{T}}}{\partial\beta}(\frac{\partial\beta}{\partial P_{\mathrm{CC}}}\frac{\mathrm{d}P_{\mathrm{CC}}}{\mathrm{d}\tau} + \frac{\partial\beta}{\partial P_{\mathrm{NLT}}}\frac{\mathrm{d}P_{\mathrm{NLT}}}{\mathrm{d}\tau})),\\
\frac{\mathrm{d}\theta_{m_{\mathrm{T}}}}{\mathrm{d}\tau} &= 0.5\epsilon m_{\mathrm{T}}(S,\beta) + (1+0.5\tau)(\frac{\partial m_{\mathrm{T}}}{\partial S}\frac{\mathrm{d}S}{\mathrm{d}\tau} + \frac{\partial m_{\mathrm{T}}}{\partial\beta}(\frac{\partial\beta}{\partial P_{\mathrm{CC}}}\frac{\mathrm{d}P_{\mathrm{CC}}}{\mathrm{d}\tau} + \frac{\partial\beta}{\partial P_{\mathrm{NLT}}}\frac{\mathrm{d}P_{\mathrm{NLT}}}{\mathrm{d}\tau})).
\end{aligned} \tag{6.94}$$

The reduced order slow model that is obtained by setting $\epsilon = 0$ in (6.93), and substituting P_{NLT} from the equation of y_5 in $\epsilon\frac{\mathrm{d}S}{\mathrm{d}\tau} = 0$, yields the following algebraic equations

$$\begin{aligned}
T_{\mathrm{CC}} &= \frac{c_pT_{\mathrm{C}}m_{\mathrm{C}} + \eta_{\mathrm{CC}}H_um_f}{c_v(m_c + m_f - \theta_{m_{\mathrm{T}}}) + c_p\theta_{m_{\mathrm{T}}}},\\
P_{\mathrm{CC}} &= \frac{\gamma Rc_vm_{\mathrm{CC}}}{V_{\mathrm{CC}}}T_{\mathrm{CC}}^2\frac{(\theta_{m_{\mathrm{T}}} - m_{\mathrm{C}} - m_f)}{(c_pT_{\mathrm{C}}m_{\mathrm{C}} + \eta_{\mathrm{CC}}H_um_f - c_pT_{\mathrm{CC}}\theta_{m_{\mathrm{T}}}) - c_vT_{\mathrm{CC}}(m_{\mathrm{C}} + m_f - \theta_{m_{\mathrm{T}}})},\\
\beta &= \frac{m_{\mathrm{Nozzle}} - \theta_{m_{\mathrm{T}}}}{m_{\mathrm{C}} - m_{\mathrm{Nozzle}} + \theta_{m_{\mathrm{T}}}},\\
P_{\mathrm{NLT}} &= P_{\mathrm{CC}}(1 + \frac{m_{\mathrm{C}}(T_{\mathrm{C}} - T_d)}{\eta_{mech}\theta_{m_{\mathrm{T}}}T_{\mathrm{CC}}\theta_{\eta_{\mathrm{T}}}})^{\frac{\gamma}{\gamma-1}},
\end{aligned} \tag{6.95}$$

where S can also be approximated by replacing β from (6.92) as its positive root.

The terms η_{T} and m_{T} are polynomial functions of P_{CC} and S (following (6.87)-(6.92)) which are dependent on the performance maps of the turbine and in our simulations we follow

a numerical algorithm to compute the derivatives of these maps in terms of P_{CC} and S.

To discretize the above continuous-time model, the first order approximation of the algorithm that was presented in Remark 6.1 was used which shows an acceptable result for estimation of both the fast and the slow states of the system with a sampling period of $T_s = 1$ msec ($\iota = 0.001$).

6.5.2 Simulation Scenario

In the simulation scenario considered here the engine is assumed to be subjected to degradation damage due to turbine erosion. This causes a gradual drift in the system health parameters, and as a result the system states. A slowly varying linear degradation model is applied to the turbine health parameters during the 500 cycles of operation that causes a 3% drop in the turbine efficiency and 1.5% increase in its mass flow capacity. The time-scale separation parameter ϵ is selected as 0.005 to provide this degradation rate in the turbine.

6.5.3 Erosion Estimation Results

Our developed two-time scale filtering methodology is now utilized for estimating the system states as well as the turbine health parameters that are represented as the augmented slow states to the gas turbine state equations. The results corresponding to the percent of the mean absolute error (MAE) within an estimation window of 5 seconds for different number of ensemble members are presented in Table 6.5 using our proposed TTS-EnKF estimation scheme. We

now compare the errors that are obtained from this method with the ones that are obtained by using the "exact" EnKF approach (that is when no fast-slow decomposition of the overall states of the system is performed), where the same scenario having the same number of ensembles are applied. The MAE$\%$ results obtained corresponding to this method are presented in Table 6.6. It should be pointed out that the exact EnKF approach does not converge, due to numerical ill-conditioning, when the number of ensembles is less than 20 (N/C in the table denotes Not Convergent).

It is noted that the covariance matrix in the exact EnKF method is dependent to the time-scale separation parameter ϵ, which can cause non-singularity of the covariance matrix in some scenarios and as a result divergence of the Kalman filtering algorithm due to ill-conditioning of the estimation problem (in calculating the Kalman gain). This problem in exact EnKF method is more obvious for smaller values of ϵ and/or low number of ensemble members in the EnKF algorithm.

A comparison between the TTS-EnKF and the exact EnKF estimation results shows that although the exact method is not capable of performing the system state estimation for lower number of ensembles, the TTS-EnKF approach is still capable of performing the estimation objective with a fewer number of ensembles, and consequently it can yield a less computationally costly algorithm.

The results presented in Tables 6.5 indicate that by increasing the number of ensemble members to more than $N = 100$ does not necessarily result in a more accurate estimation performance. The best estimation results that are achieved are for 100 ensembles with the maximum

percentage of the mean absolute error (MAE $\%$) of 0.95% for the state estimation obtained for the nozzle pressure, and the MAE$\%$ of 2.28% for the output estimation obtained for the turbine pressure. However, due to the approximations we made to obtain the algebraic equations in Subsection for turbine health parameters, in this specific scenario with $\epsilon = 0.005$, in almost all cases, the exact EnKF method results in a lower MAE$\%$ for both the state and output estimations. Moreover, the discrepancy between the TTS-EnKF and the exact EnKF approaches for the output estimation problem is lower than that of the state estimation problem.

By selecting $N = 100$, the estimation results for states and outputs associated with both methods are depicted in Figures 6.1 and 6.2, respectively. The results shown in these two figures confirm that although the exact method has lower errors for estimating the fast states of the system, the TTS-EnKF method shows a more accurate result for estimating the system slow states, specially the turbine mass flow capacity.

Table 6.5: Estimation MAE$\%$ using different number of ensembles for the TTS-EnKF method (a) states and (b) measurement outputs.

(a)

State	$N = 10$	$N = 50$	$N = 100$	$N = 200$
P_{CC}	0.7481	0.7440	0.6532	0.6510
N	0.1185	0.0806	0.0515	0.0495
T_{CC}	0.1220	0.0668	0.0613	0.0611
P_{NLT}	1.1822	1.1774	0.9521	0.9213
$\theta_{\eta_{\mathrm{T}}}$	0.6831	0.5938	0.4281	0.4210
$\theta_{m_{\mathrm{T}}}$	0.0614	0.0342	0.0322	0.0341

(b)

Output	$N = 10$	$N = 50$	$N = 100$	$N = 200$
T_{C}	0.4118	0.3013	0.2451	0.2510
P_{C}	1.5231	1.4867	1.3047	1.3045
N	1.1148	0.0806	0.0655	0.06122
T_{T}	0.3147	0.2338	0.2001	0.2170
P_{T}	2.6250	2.6287	2.2830	2.3030

Table 6.6: Estimation MAE% using different number of ensembles for the exact EnKF method (a) states and (b) measurement outputs (N/C means not convergent).

(a)

State	$N = 10$	$N = 50$	$N = 100$	$N = 200$
P_{CC}	N/C	0.3355	0.3022	0.3020
N	N/C	0.0504	0.0492	0.0497
T_{CC}	N/C	0.0714	0.0661	0.0670
P_{NLT}	N/C	0.2254	0.2142	0.2145
$\theta_{\eta_{\mathrm{T}}}$	N/C	0.3021	0.2815	0.2781
$\theta_{m_{\mathrm{T}}}$	N/C	0.0746	0.0526	0.0532

(b)

Output	$N = 10$	$N = 50$	$N = 100$	$N = 200$
T_{C}	N/C	0.1589	0.1322	0.1323
P_{C}	N/C	1.1821	1.1620	1.1400
N	N/C	0.0504	0.0454	0.0427
T_{T}	N/C	0.1353	0.1132	0.1151
P_{T}	N/C	2.3484	2.2550	2.2260

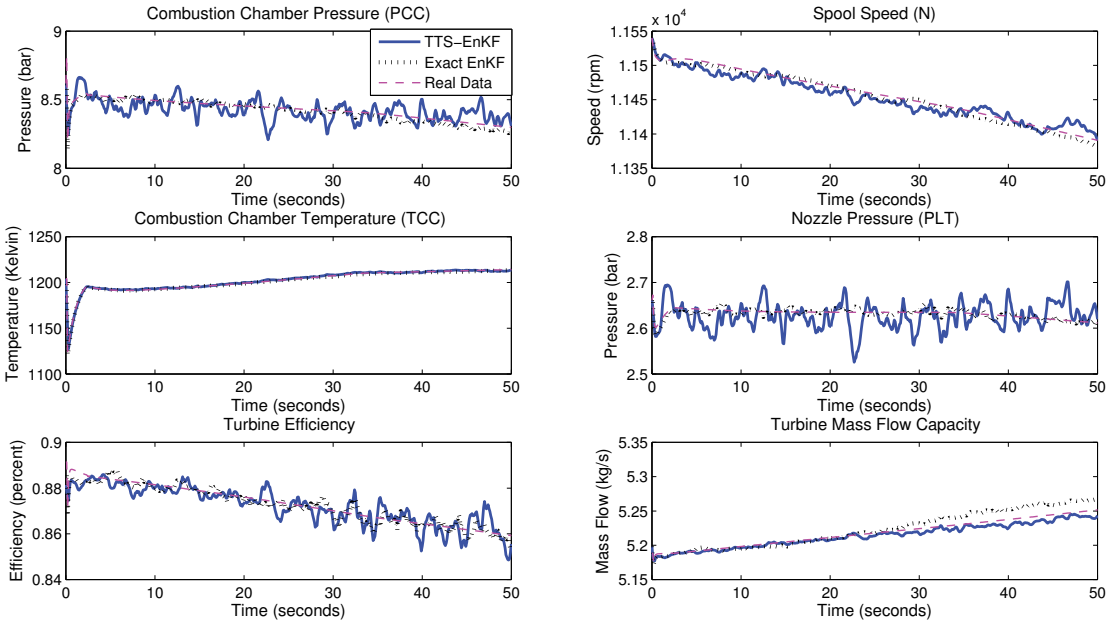


Figure 6.1: Estimated states corresponding to $N = 100$ by using the TTS-EnKF and the Exact EnKF approaches.

To show the effect of ϵ in the performance of both exact EnKF, and TTS-EnKF methods, the degradation scenario is repeated with different ϵ magnitudes and selecting $N = 100$, as presented in Tables 6.7, and 6.8. The summarized results in these two tables show that the estimation accuracy of the TTS-EnKF is not affected by ϵ, whereas exact EnKF estimation

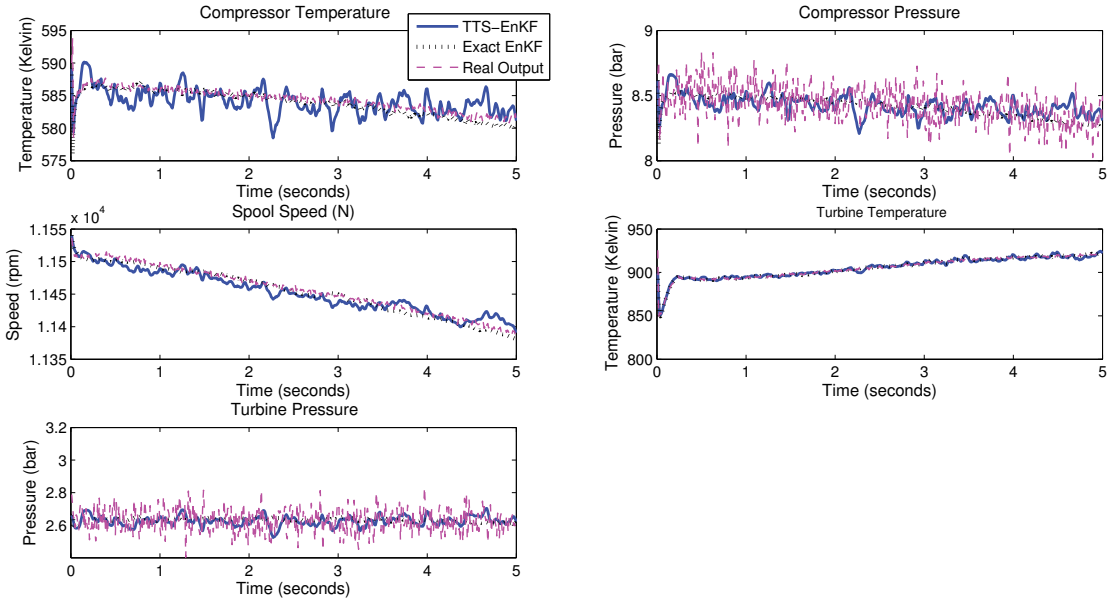


Figure 6.2: Estimated outputs corresponding to $N = 100$ by using the TTS-EnKF and the Exact EnKF approaches.

accuracy is highly related to ϵ such that for ϵ values less than or equal to 0.001 the algorithm becomes ill-conditioned and cannot converge.

Table 6.7: Estimation MAE$\%$ using different values of ϵ for the TTS-EnKF method (a) states and (b) measurement outputs.

(a)

State	$\epsilon = 0.005$	$\epsilon = 0.003$	$\epsilon = 0.001$	$\epsilon = 0.0001$
P_{CC}	0.6532	0.6481	0.6320	0.6505
N	0.0515	0.05000	0.05325	0.05260
T_{CC}	0.0613	0.0608	0.0615	0.0611
P_{NLT}	0.9521	0.9484	0.9511	0.9491
$\theta_{\eta_{\mathrm{T}}}$	0.4281	0.4312	0.4255	0.4380
$\theta_{m_{\mathrm{T}}}$	0.0322	0.0356	0.0327	0.0351

(b)

Output	$\epsilon = 0.005$	$\epsilon = 0.003$	$\epsilon = 0.001$	$\epsilon = 0.0001$
T_{C}	0.2451	0.2266	0.2219	0.2205
P_{C}	1.3047	1.3164	1.3083	1.3085
N	0.0655	0.0602	0.0637	0.0652
T_{T}	0.2001	0.2109	0.2012	0.2149
P_{T}	2.2830	2.2583	2.2530	2.3072

Table 6.8: Estimation MAE$\%$ using different values of ϵ for the exact EnKF method (a) states and (b) measurement outputs (N/C means not convergent).

(a)

State	$\epsilon = 0.005$	$\epsilon = 0.003$	$\epsilon = 0.001$	$\epsilon = 0.0001$
P_{CC}	0.3022	0.6255	N/C	N/C
N	0.0492	0.0651	N/C	N/C
T_{CC}	0.0661	0.1200	N/C	N/C
P_{NLT}	0.2142	0.3541	N/C	N/C
$\theta_{\eta_{\mathrm{T}}}$	0.2815	0.4537	N/C	N/C
$\theta_{m_{\mathrm{T}}}$	0.0526	0.1070	N/C	N/C

(b)

Output	$\epsilon = 0.005$	$\epsilon = 0.003$	$\epsilon = 0.001$	$\epsilon = 0.0001$
T_{C}	0.1322	0.2220	N/C	N/C
P_{C}	1.1620	1.2811	N/C	N/C
N	0.0454	0.0567	N/C	N/C
T_{T}	0.1132	0.2112	N/C	N/C
P_{T}	2.2550	2.6372	N/C	N/C

6.5.4 Erosion Prediction Results

In this scenario, our prediction strategy that is developed based on the two-time scale EnKF method is utilized to predict the propagation of the system states (fast states) and the turbine health parameters (slow states) when the degradation due to the erosion has affected the system during its entire operating horizon (that is 500 cycles of flight).

For the prediction case study, $N = 100$ is selected for both the TTS-EnKF and the exact EnKF schemes. However, the prediction case study also includes the classical PF-based prediction method [55] (according to the results of Subsection 6.4.4) using 100 particles in order to compare the execution time of all the three methods as a measure of EF complexity that was described in Subsection 6.4.4.

The prediction horizon is also extended from the 100 steps-ahead to 500 steps-ahead and the MAE$\%$ results corresponding to the first and the last prediction windows are provided in Tables

6.9 and 6.10, respectively. These results are in accordance with those shown in Figures 6.3 and 6.4 for the state and the output prediction results. From the results shown in the two tables and two figures, one can realize that the PF-based prediction algorithm with 100 particles does not show an acceptable prediction performance. In other words, beyond the 100 steps-ahead prediction horizon the MAE$\%$ increases drastically for both the state and output prediction results. On the other hand, as the prediction horizon extends, the MAE$\%$ also increases for prediction results associated with both the exact EnKF and the TTS-EnKF approaches. However, the TTS-EnKF scheme shows a better prediction accuracy results as compared to the exact EnKF method. For example, the maximum 100 steps-ahead MAE$\%$ for $\theta_{\eta_{\mathrm{T}}}$ prediction using the TTS-EnKF method is 0.34%, whereas it is around 0.42% for the exact EnKF scheme. We emphasize here again that for this specific scenario with $\epsilon = 0.005$ the exact EnKF method does not diverge.

Finally, the execution time associated with one iteration of each scheme is measured and provided in Table 6.11. The results show a large difference between the PF-based prediction scheme execution time and that associated with and compared to the EnKF-based approaches.

Table 6.9: 100-step ahead prediction MAE$\%$ using three different methods (a) states and (b) measurement outputs.

(a)

State	TTS-EnKF	Exact EnKF	PF-Based Method
P_{CC}	0.2118	0.2843	0.4149
N	0.0474	0.0816	0.1017
T_{CC}	0.1220	0.1437	0.1653
P_{NLT}	0.2854	0.3778	0.6373
$\theta_{\eta_{\mathrm{T}}}$	0.3439	0.4283	0.5030
$\theta_{m_{\mathrm{T}}}$	0.0087	0.0099	0.0109

(b)

Output	TTS-EnKF	Exact EnKF	PF-Based Method
T_{C}	0.1052	0.1232	0.1895
P_{C}	1.3338	1.3285	1.3801
N	0.0474	0.0816	0.1017
T_{T}	0.1989	0.2090	0.1937
P_{T}	1.8963	1.8666	1.9765

Table 6.10: 500-step ahead prediction MAE% using three different methods (a) states and (b) measurement outputs.

(a)

State	TTS-EnKF	Exact EnKF	PF-Based Method
P_{CC}	1.0542	1.2994	3.5630
N	0.5168	0.6374	1.7717
T_{CC}	0.5700	0.7287	1.9753
P_{NLT}	1.2063	1.5131	4.0037
$\theta_{\eta_{\mathrm{T}}}$	1.8358	2.1622	6.4120
$\theta_{m_{\mathrm{T}}}$	0.0287	0.0342	0.1037

(b)

Output	TTS-EnKF	Exact EnKF	PF-Based Method
T_{C}	0.3993	0.4859	1.4007
P_{C}	1.6270	1.7990	3.8850
N	0.5168	0.6374	1.7717
T_{T}	1.1358	1.3814	3.9448
P_{T}	2.2675	2.3229	4.1315

Table 6.11: Time Complexity Analysis for the TTS-EnKF, Exact EnKF and the PF-Based Prediction Methods in seconds corresponding to one iteration of each scheme.

Method	Best Scenario	Average Scenario	Worst Scenario
TTS-EnKF	1.1676	1.3112	2.9235
Exact EnKF	0.8898	0.9310	1.0020
PF-based	24.4211	33.5490	64.0247

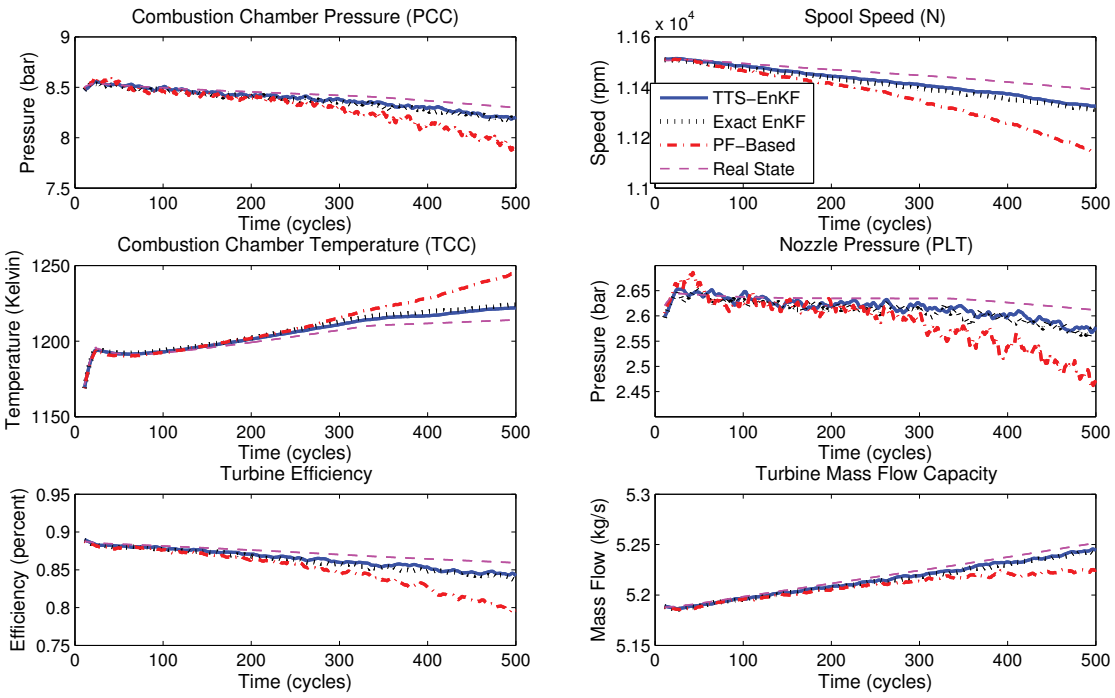


Figure 6.3: Predicted states corresponding to $N = 100$ by using the TTS-EnKF, the Exact EnKF and the PF-based approaches.

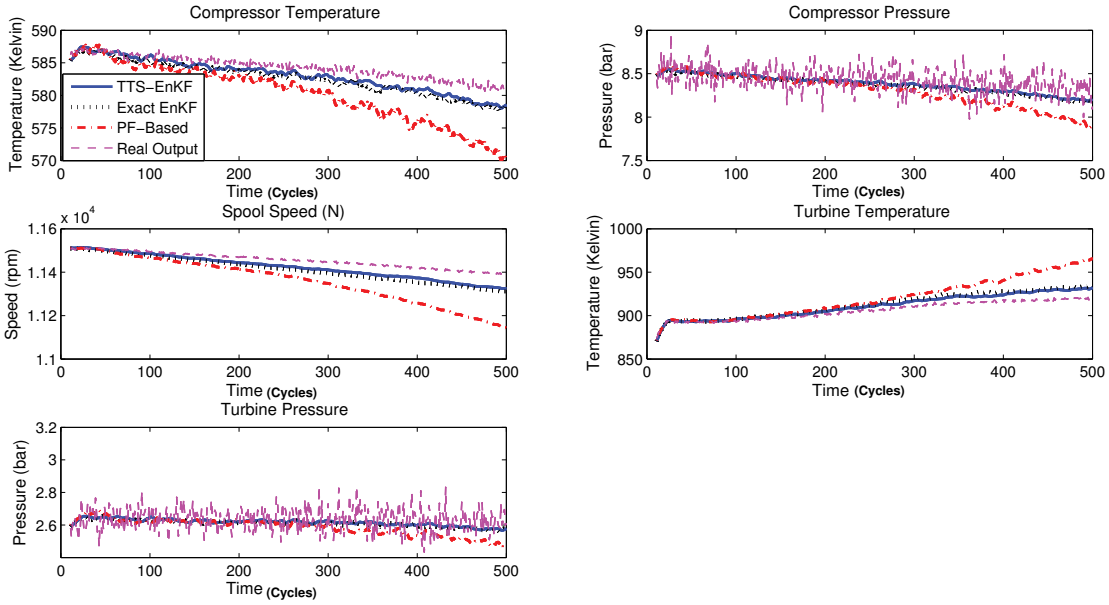


Figure 6.4: Predicted outputs corresponding to $N = 100$ by using the TTS-EnKF, the Exact EnKF and the PF-based approaches.

6.6 Conclusion

In this chapter, a novel two-time scale estimation filter is developed and designed based on an ensemble Kalman filtering (En-KF) approach to estimate the fast and slow states of a nonlinear system. One of the main applications of our proposed estimation strategy is in investigating the health monitoring and damage tracking problems. Based on our developed estimation algorithm, a two-time scale prediction methodology is also proposed to predict the long-term behavior of the system dynamical states. Our proposed estimation and prediction methodologies were applied to a gas turbine engine system to illustrate and validate our results when the system is affected by a gradual degradation damage. The resulting estimation and prediction observations indicate an acceptable performance of our methods and confirm that our strategy is quiet suitable for further investigation in the domain of health and condition monitoring research.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

In this thesis, the health monitoring framework for nonlinear systems was considered to address the prognosis problem in dynamical systems. Towards this end, two different problem formulation strategies have been utilized to include the damage mechanism which affects the system dynamics. Consequently, each damage modeling formulation strategy has led to a different health monitoring framework. Both of these frameworks include two main steps namely, the estimation and the prediction steps that are based on nonlinear filtering methodologies. The developed estimation and prediction methodologies in this thesis are considered as either enhancement of the current Monte Carlo based estimation/prediction methods or proposing new filtering methodologies.

In Chapter 2, the necessary background information regarding the gas turbine engine system, as the main case study considered in this thesis, have been presented. The degradation due to fouling and erosion damages have also been formulated in this chapter.

In Chapter 3, a dual estimation methodology is developed for both time-varying parameters and states of a nonlinear system based on the Recursive Prediction Error (RPE) concept and the Particle Filtering (PF) scheme. Our developed methodology in based on a concurrent implementation of state and parameter estimation filters as opposed to using a single filter for simultaneously estimating the augmented states and parameters. The convergence and stability of our proposed dual estimation strategy are shown formally to be guaranteed under certain conditions. The proposed dual estimation framework is then utilized for addressing the challenging problem of fault diagnosis of nonlinear systems. The performance capabilities of our proposed fault diagnosis methodology is demonstrated and evaluated by the application to a gas turbine engine through accomplishing state and parameter estimation under simultaneous and concurrent component fault scenarios. The health parameters of the system are considered to be slowly time-varying during the engine operation. Extensive simulation results are provided to substantiate and justify the superiority of our proposed fault diagnosis methodology when compared to another well-known alternative diagnostic technique that is available in the literature.

In Chapter 4, an improved method for uncertainty management in long-term prediction of nonlinear systems by using particle filters was developed. In our proposed approach, an observation forecasting scheme is developed to extend the system observation profiles (as time-series) to future. Particles are then propagated to future time instants according to a resampling algo-

rithm instead of considering constant weights for the particles propagation in the prediction step. The uncertainty in the long-term prediction of the system states and parameters are managed by utilizing dynamic linear models for development of an observation forecasting scheme. This task is addressed through an outer adjustment loop for adaptively changing the sliding observation injection window based on the Mahalanobis distance criterion. Our proposed approach is then applied to predict the health condition of a gas turbine engine that is affected by degradations in the system health parameters for demonstrating and illustrating the capabilities and performance characteristics of developed schemes.

In Chapter 5, the proposed prediction method in Chapter 4 is utilized to develop a hybrid architecture for prognosis and health monitoring of nonlinear systems by integration of model-based and computationally intelligent-based techniques. In our proposed framework the well-known particle filter method is utilized to estimate the states as well as the health parameters of the system. Simultaneously, the system observations are predicted through an observation forecasting scheme that is developed based on neural network paradigms to construct observation profiles for future time horizons. Our proposed on-line training process for observation forecasting enables the neural network model to track the non-ergodic changes in the observations profiles, whereas such behavior happens as a result of hidden damage that affects the system health parameters. The forecasted observations are utilized in the particle filters to predict evolution of the system states as well as health parameters (these are considered to be time-varying due to effects of degradation and damage) into future time horizons. The proposed hybrid architecture enables one to select health signatures for determining the remaining useful life (RUL) of the system or its components not only based on the system observations but also by taking into

account the system health parameters that are not physically measurable. Our proposed hybrid health monitoring methodology is constructed based on the framework which is not dependent on the structure of the neural network model utilized in the implementation of the observation forecasting scheme, however changing the neural network model structure in this framework does not affect the prediction accuracy of the entire health prediction algorithm, significantly. As a case study, our proposed hybrid approach is also applied to predict the health condition of a gas turbine engine when it is affected by fouling and erosion degradation and fault damages.

In Chapter 6, another popular method for formulation of health monitoring problem in dynamical systems is utilized which suggests to model the dynamics of the damage mechanism as a slow state augmented to the system fast dynamical equations. This augmentation results in a two-time scale system to be investigated in the system health estimation and prediction steps of the health monitoring framework. In this chapter, a two-time scale filtering approach is developed for this purpose based on the ensemble Kalman filtering approach by taking advantage of the model reduction concept. The performance of the proposed two-time scale ensemble Kalman filter is shown to be more accurate and less expensive in terms of equivalent flop complexity, as compared to the well-known particle filtering approach. By utilizing the augmentation of state equations and damage mechanism, our developed two-time scale ensemble Kalman filter is applied for health monitoring of a gas turbine engine when it is assumed to be affected by degradation phenomenon, i.e. erosion of the turbine, as the damage mechanism.

In conclusion, two main frameworks for addressing the health monitoring problem of nonlinear systems which are subjected to degradation damage, are presented. While in the first

framework (Chapters 3 to 5) it is assumed that the dynamic of the damage is not known and it is modeled based on its effect on the system health parameters, in the second framework the mathematical model corresponding to damage propagation is assumed to be known (Chapter 6). Therefore, the prediction results obtained from the second framework can be extended for longer prediction horizon as compared to the ones obtained from the first framework. However, in the case that more accurate model for damage propagation is accessible, the second approach is preferred, otherwise the first approach can be more general for addressing the health monitoring and prognosis problems in nonlinear systems.

The proposed methodologies in this thesis are applied to component fault diagnosis and/or failure prognosis of a gas turbine engine model simulating under different degradation scenarios. All the simulations and codings are performed by utilizing the powerful MATLAB software.

7.2 Suggestions for Future Work

Some of the future extensions of the present research are as follows:

1. To develop an optimal, computationally efficient parameter estimation approach based on particle filters (which is an open problem in the domain of estimation theory) and to investigate the optimality gap of using the proposed sub-optimal solutions for parameter estimation problem.

2. To extend the proposed framework based on particle filters to include more complicated damage models of the physical dynamical systems. The uncertainty in the damage dy-

namics can be modelled as time-varying parameters to be estimated.

3. To extend the proposed methodology for prediction based on the combination of particle filters and observation forecasting module by utilizing multi-variate time series forecasting methods such that the prediction horizon can be extended with lower prediction error.

4. To develop a hybrid health monitoring and prognosis framework based on the combination of data-driven and EnKF approaches to make case for utilization of EnKF method for problems with non-Gaussian process and measurement noises.

5. To investigate the utilization of the EnKF approach as a solid and computationally efficient alternative method for particle filters in prognosis and health monitoring applications and identify the limitations to achieve this goal. The EnKF approach has not been studied extensively in health monitoring and prognosis applications.

6. To develop an optimal solution for the EnKF problem inspired from the convergence results for particle filters as developed in [169] considering that in EnKF resampling step does not exist. The convergence of EnKF to Kalman filter in linear case has only been shown so far.

7. To propose novel prognosis metrics to quantify the judgements regarding the RUL and EOL predictions which would not be application specific. Most of the prognosis metrics which are available in the literature are ad-hoc and application specific. On the other hand, they only rely on the RUL and EOL prediction results whereas in prognosis the RUL prediction accuracy is closely related to the performance of either estimation or prediction steps before predicting RUL and/or EOL. Therefore, introducing a metric that is capa-

ble of taking into account the effect of uncertainty regarding the estimation/prediction approach which is selected to predict RUL, can be rewarding.

Bibliography

[1] S. Haykin. *Neural networks and learning machines*, volume 3. Pearson Education Upper Saddle River, 2009.

[2] J. Liu and M. West. *Combined parameter and state estimation in simulation-based filtering*. Springer, 2001.

[3] G. Poyiadjis, S. Singh, and A. Doucet. Gradient-free maximum likelihood parameter estimation with particle filters. In *American Control Conference*, pages 3062–3067, 2006.

[4] N. Daroogheh, N. Meskin, and K. Khorasani. Particle filtering for state and parameter estimation in gas turbine engine fault diagnostics. In *American Control Conference*, pages 4343–4349, 2013.

[5] N. Daroogheh, N. Meskin, and K. Khorasani. Particle filter-based fault diagnosis of nonlinear systems using a dual particle filter scheme. *arXiv preprint arXiv:1606.08806*, 2016.

[6] R. Sekhon, H. Bassily, and J. Wagner. A comparison of two trending strategies for gas turbine performance prediction. *Journal of Engineering for Gas Turbines and Power*, 130(4):041601, 2008.

[7] A. Grall, L. Dieulle, C. Berenguer, and M. Roussignol. Continuous-time predictive-maintenance scheduling for a deteriorating system. *IEEE Transactions on Reliability*, 51(2):141–150, Jun 2002.

[8] A. D. Kenyon, V. M. Catterson, S. D. J. McArthur, and J. Twiddle. An agent-based implementation of hidden markov models for gas turbine condition monitoring. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 44(2):186–195, 2014.

[9] D. Codetta-Raiteri and L. Portinale. Dynamic Bayesian networks for fault detection, identification, and recovery in autonomous spacecraft. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 45(1):13–24, 2015.

[10] D. Lefebvre. Fault diagnosis and prognosis with partially observed petri nets. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 44(10):1413–1424, 2014.

[11] M. Yu, D. Wang, M. Luo, and L. Huang. Prognosis of hybrid systems with multiple incipient faults: Augmented global analytical redundancy relations approach. *IEEE Transactions on Systems, Man, and Cybernetics: Part A: Systems and Humans*, 41(3):540–551, 2011.

[12] A. J. Trappey, D. W. Hsiao, and L. Ma. Maintenance chain integration using petri-net enabled multiagent system modeling and implementation approach. *IEEE Transactions*

on Systems, Man, and Cybernetics: Part C: Applications and Reviews, 41(3):306–315, 2011.

[13] H. Gu, S. Zhang, and L. Ma. Process analysis for performance evaluation of prognostics methods orienting to engineering application. In *International Conference on Quality, Reliability, Risk, Maintenance, and Safety Engineering (ICQR2MSE)*, pages 681–686, 2012.

[14] B. Pourbabaee, N. Meskin, and K. Khorasani. Robust sensor fault detection and isolation of gas turbine engines subjected to time-varying parameter uncertainties. *Mechanical Systems and Signal Processing*, 76:136–156, 2016.

[15] B. Pourbabaee, N. Meskin, and K. Khorasani. Sensor fault detection, isolation, and identification using multiple-model-based hybrid Kalman filter for gas turbine engines. *IEEE Transactions on Control Systems Technology*, 24(4):1184–1200, 2016.

[16] M. Navi, M. Davoodi, and N. Meskin. Sensor fault detection and isolation of an industrial gas turbine using partial kernel PCA. *IFAC-PapersOnLine*, 48(21):1389–1396, 2015.

[17] B. Pourbabaee, N. Meskin, and K. Khorasani. Sensor fault detection and isolation using multiple robust filters for linear systems with time-varying parameter uncertainty and error variance constraints. In *IEEE Conference on Control Applications (CCA)*, pages 382–389, 2014.

[18] B. Pourbabaee, N. Meskin, and K. Khorasani. Multiple-model based sensor fault diagnosis using hybrid Kalman filter approach for nonlinear gas turbine engines. In *American Control Conference*, pages 4717–4723, 2013.

[19] E. Tsoutsanis, N. Meskin, M. Benammar, and K. Khorasani. An efficient component map generation method for prediction of gas turbine performance. In *ASME Turbo Expo 2014: Turbine Technical Conference and Exposition*. American Society of Mechanical Engineers.

[20] E. Tsoutsanis, N. Meskin, M. Benammar, and K. Khorasani. Performance-based prognosis scheme for industrial gas turbines. In *IEEE Conference on Prognostics and Health Management (PHM)*, pages 1–8, 2015.

[21] E. Tsoutsanis and N. Meskin. Forecasting the health of gas turbine components through an integrated performance-based approach. In *IEEE International Conference on Prognostics and Health Management (ICPHM)*, pages 1–8, 2016.

[22] E. Tsoutsanis, N. Meskin, M. Benammar, and K. Khorasani. A component map tuning method for performance prediction and diagnostics of gas turbine compressors. *Applied Energy*, 135:572–585, 2014.

[23] E. Tsoutsanis, N. Meskin, M. Benammar, and K. Khorasani. A dynamic prognosis scheme for flexible operation of gas turbines. *Applied Energy*, 164:686–701, 2016.

[24] P. Shetty, D. Mylaraswamy, and T. Ekambaram. A hybrid prognostic model formulation and health estimation of auxiliary power units. *Journal of Engineering for Gas Turbines and Power*, 130(2):021601–1–021601–9, 2008.

[25] M. E. Orchard. *A Particle Filtering-based Framework for On-line Fault Diagnosis and Failure Prognosis*. PhD thesis, Georgia Institute of Technology, 2006.

[26] S. Yin, S. X. Ding, and D. Zhou. Diagnosis and prognosis for complicated industrial systems part I. *IEEE Transactions on Industrial Electronics*, 63(4):2501–2505, 2016.

[27] S. Yin, S. X. Ding, and D. Zhou. Diagnosis and prognosis for complicated industrial systems part II. *IEEE Transactions on Industrial Electronics*, 63(5):3201–3204, 2016.

[28] L. Liao and F. Köttig. A hybrid framework combining data-driven and model-based methods for system remaining useful life prediction. *Applied Soft Computing*, 44:191–199, 2016.

[29] J. Son, Q. Zhou, S. Zhou, X. Mao, and M. Salman. Evaluation and comparison of mixed effects model based prognosis for hard failure. *IEEE Transactions on Reliability*, 62(2):379–394, 2013.

[30] X. S. Si, W. Wang, C. Hu, D. Zhou, and M. G. Pecht. Remaining useful life estimation based on a nonlinear diffusion degradation process. *IEEE Transactions on Reliability*, 61(1):50–67, 2012.

[31] D. Chelidze, J. P. Cusumano, and A. Chatterjee. A dynamical systems approach to damage evolution tracking, part 1: description and experimental application. *Journal of Vibration and Acoustics*, 124(2):250–257, 2002.

[32] D. Chelidze, J. P. Cusumano, and A. Chatterjee. A dynamical systems approach to damage evolution tracking, part 2: Model-based validation and physical interpretation. *Transactions of the ASME*, 124(2):250–257, 2002.

[33] D. Chelidze and J. P. Cusumano. A dynamical systems approach to failure prognosis. *Journal of Vibration and Acoustics*, 126(1):2–8, 2004.

[34] D. Chelidze and M. Liu. Dynamical systems approach to fatigue damage identification. *Journal of Sound and Vibration*, 281(3):887–904, 2005.

[35] J. P. Cusumano and A. Chatterjee. Steps towards a qualitative dynamics of damage evolution. *International Journal of Solids and Structures*, 37(44):6397–6417, 2000.

[36] D. Chelidze. Multimode damage tracking and failure prognosis in electromechanical systems. In *AeroSense*, pages 1–12. International Society for Optics and Photonics, 2002.

[37] J. Luo, K. R. Pattipati, L. Qiao, and S. Chigusa. Model-based prognostic techniques applied to a suspension system. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, 38(5):1156–1168, 2008.

[38] M. Orchard, G. Kacprzynski, K. Goebel, B. Saha, and G. Vachtsevanos. Advances in uncertainty representation and management for particle filtering applied to prognostics.

In *IEEE International conference on Prognostics and health management*, pages 1–6, 2008.

[39] B. Saha and K. Goebel. Modeling Li-Ion battery capacity depletion in a particle filtering framework. pages 2909–2924, 2009.

[40] B. Saha, K. Goebel, S. Poll, and J. Christophersen. Prognostics methods for battery health monitoring using a Bayesian framework. 58(2):291–296, 2009.

[41] B. Saha and K. Goebel. Uncertainty management for diagnostics and prognostics of batteries using Bayesian techniques. In *IEEE Aerospace Conference*, pages 1–8, 2008.

[42] M. Daigle and K. Goebel. Model-based prognostics with fixed-lag particle filters. In *Annual Conference of the Prognostics and Health Management Society*, 2009.

[43] M. Orchard, F. Tobar, and G. Vachtsevanos. Outer feedback correction loops in particle filtering-based prognostic algorithms: Statistical performance comparison. *Studies in Informatics and Control*, 18(4):295–304, 2009.

[44] X. Guan, Y. Liu, A. Saxena, J. Celaya, and K. Goebel. Entropy-based probabilistic fatigue damage prognosis and algorithmic performance comparison. In *Annual Conference of the Prognostics and Health Management Society*, volume 9, 2009.

[45] M. Daigle and K. Goebel. Improving computational efficiency of prediction in model-based prognsotics using the unscented transformation. Technical report, 2010.

[46] B. Saha and K. Goebel. Model adaptation for prognostics in a particle filtering framework. *International Journal of Prognostics and Health Management*, 2:61–70, 2011.

[47] S. H. Nguyen and D. Chelidze. New invariant measures to track slow parameter drifts in fast dynamical systems. *Nonlinear Dynamics*, 79(2):1207–1216, 2015.

[48] Y. Qian, R. Yan, and R. X. Gao. A multi-time scale approach to remaining useful life prediction in rolling bearing. *Mechanical Systems and Signal Processing*, 83:549–567, 2017.

[49] J. Korbicz, Z. Fathi, and W. Fred Ramirez. State estimation schemes for fault detection and diagnosis in dynamic systems. *International Journal of Systems Science*, 24(5):985–1000, 1993.

[50] D. Simon and D. L. Simon. Aircraft turbofan engine health estimation using constrained Kalman filtering. *Journal of engineering for Gas Turbines and Power*, 127(2):323–328, 2005.

[51] P. Baraldi, F. Mangili, and E. Zio. A Kalman filter-based ensemble approach with application to turbine creep prognostics. *IEEE Transactions on Reliability*, 61(4):966–977, 2012.

[52] B. Pattipati, C. Sankavaram, K. Pattipati, Y. Zhang, M. Howell, and M. Salman. Multiple model moving horizon estimation approach to prognostics in coupled systems. In *IEEE AUTOTESTCON*, 2011.

[53] H. J. Kushner and H. Joseph. *Probability methods for approximations in stochastic control and for elliptic equations*, volume 129. Academic Press New York, 1977.

[54] V. E. Beneš. Exact finite-dimensional filters for certain diffusions with nonlinear drift. *Stochastics: An International Journal of Probability and Stochastic Processes*, 5(1-2):65–92, 1981.

[55] A. Doucet, N. De Freitas, and N. Gordon. *Sequential Monte Carlo Methods in Practice*. Springer Verlag, 2001.

[56] A. Bensoussan, R. Glowinski, and A. Raşcanu. Approximation of some stochastic differential equations by the splitting up method. *Applied Mathematics & Optimization*, 25:81–106, 1992.

[57] S. Lototsky and B. L. Rozovskii. Recursive multiple wiener integral expansion for nonlinear filtering of diffusion processes. *Stochastic Processes and Functional Analysis, Lecture Notes in Pure and Appl. Math*, 186:199–208, 1997.

[58] D. Crisan, J. Gaines, and T. Lyons. Convergence of a branching particle method to the solution of the Zakai equation. *SIAM Journal on Applied Mathematics*, 58:1568–1590, 1998.

[59] I. Arasaratnam and S. Haykin. Cubature Kalman filters. *IEEE Transactions on Automatic Control*, 54(6):1254–1269, 2009.

[60] N. J. Gordon, D. J. Salmond, and A. F. Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. In *IEE Proceedings of Radar and Signal Processing*, volume 140, 1993.

[61] P. Del Moral. Nonlinear filtering: Interacting particle resolution. *Comptes Rendus de l'Académie des Sciences-Series I-Mathematics*, 325:653–658, 1997.

[62] A. Bensoussan, R. Glowinski, and A. Rascanu. Approximation of the Zakai equation by the splitting up method. *SIAM Journal on Control and Optimization*, 28:1420–1431, 1990.

[63] P. Florchinger and F. Le Gland. Time-discretization of the Zakai equation for diffusion processes observed in correlated noise. *Stochastics: An International Journal of Probability and Stochastic Processes*, 35:233–256, 1991.

[64] M. Sun and R. Glowinski. Pathwise approximation and simulation for the Zakai filtering equation through operator splitting. *Calcolo*, 30:219–239, 1993.

[65] H. C. Yeong, J. H. Park, and N. S. Namachchivaya. Particle filters in a multiscale environment: With application to the lorenz-96 atmospheric model. *Stochastics and Dynamics*, 11:569–591, 2011.

[66] V. Kadirkamanathan, P. Li, M. H. Jaward, and S. G. Fabri. A sequential Monte Carlo filtering approach to fault detection and isolation in nonlinear systems. In *the 39th IEEE Conference on Decision and Control*, volume 5, pages 4341–4346, 2000.

[67] P. Li and V. Kadirkamanathan. Particle filtering based likelihood ratio approach to fault diagnosis in nonlinear stochastic systems. *IEEE Transactions on Systems, Man and Cybernetics. Part C, Applications and reviews*, 31(3):337–343, 2001.

[68] M. Zajac. Online fault detection of a mobile robot with a parallelized particle filter. *Neurocomputing*, 126:151–165, 2014.

[69] B. Zhao, R. Skjetne, M. Blanke, and F. Dukan. Particle filter for fault diagnosis and robust navigation of underwater robot. *IEEE Transactions on Control Systems Technology*, 22(6):2399–2407, 2014.

[70] Q. Zhang, F. Campillo, F. Cerou, and F. Legland. Nonlinear system fault detection and isolation based on bootstrap particle filters. In *44th IEEE Conference on Decision and Control, and European Control Conference. CDC-ECC'05*, pages 3821–3826, 2005.

[71] G. Vachtsevanos, G. Georgoulas, and G. Nikolakopoulos. Fault diagnosis, failure prognosis and fault tolerant control of aerospace/unmanned aerial systems. In *24th IEEE Mediterranean Conference on Control and Automation (MED)*, pages 366–371, 2016.

[72] Z. Duan, Z. Cai, and Y. Jin-xia. Fault diagnosis and fault tolerant control for wheeled mobile robots under unknown environments: A survey. In *the IEEE International Conference on Robotics and Automation*, pages 3428–3433, 2005.

[73] V. Verma. *Tractable particle filters for robot fault diagnosis*. PhD thesis, Stanford University, 2004.

[74] F. Hutter, R. Dearden, et al. The Gaussian particle filter for diagnosis of non-linear systems. In *the 5th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, 2003.

[75] S. Tafazoli and X. Sun. Hybrid system state tracking and fault detection using particle filters. *IEEE Transactions on Control Systems Technology*, 14(6):1078–1087, 2006.

[76] A. Doucet and V. B. Tadić. Parameter estimation in general state-space models using particle methods. *Annals of the institute of Statistical Mathematics*, 55(2):409–422, 2003.

[77] M. West. Mixture models, Monte Carlo, Bayesian updating, and dynamic models. *Computing Science and Statistics*, 24:325–325, 1993.

[78] T. Flury and N. Shephard. Bayesian inference based only on simulated likelihood: particle filter analysis of dynamic economic models. *Econometric Theory*, 27(05):933–956, 2011.

[79] N. Kantas, A. Doucet, S. Singh, and J. Maciejowski. An overview of sequential Monte Carlo methods for parameter estimation in general state-space models. In *the IFAC Symposium on System Identification (SYSID)*, 2009.

[80] G. Poyiadjis, A. Doucet, and S. Singh. Particle methods for optimal filter derivative: Application to parameter estimation. In *the IEEE International Conference on Acoustics, Speech, and Signal Processing, (ICASSP'05)*, volume 5, pages v–925, 2005.

[81] G. Poyiadjis, A. Doucet, and S. Singh. Particle approximations of the score and observed information matrix in state space models with application to parameter estimation. *Biometrika*, 98(1):65–80, 2011.

[82] T. Schön, A. Wills, and B. Ninness. System identification of nonlinear state-space models. *Automatica*, 47(1):39–49, 2011.

[83] J. Westerborn and J. Olsson. Efficient particle-based online smoothing in general hidden Markov models. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8003–8007, 2014.

[84] A. Doucet, M. K. Pitt, G. Deligiannidis, and R. Kohn. Efficient implementation of Markov chain Monte Carlo when using an unbiased likelihood estimator. *Biometrika*, page asu075, 2015.

[85] J. Sun, H. Zuo, and M. G. Pecht. Advances in sequential Monte Carlo methods for joint state and parameter estimation applied to prognostics. In *Prognostics and System Health Managment Confernece*, pages 1–7. IEEE, 2011.

[86] B. E. Olivares, C. Muñoz, M. E. Orchard, and J. F. Silva. Particle-filtering-based prognosis framework for energy storage devices with a statistical characterization of state-of-health regeneration phenomena. *IEEE Transactions on Instrumentation and Measurement*, 62(2):364–376.

[87] M. Jouin, R. Gouriveau, D. Hissel, M. Péra, and N. Zerhouni. Particle filter-based prognostics: Review, discussion and perspectives. *Mechanical Systems and Signal Processing*, 72:2–31, 2016.

[88] M. Samadi and M. Saif. Health monitoring of li-ion batteries: A particle filtering approach. In *IEEE 24th International Symposium on Industrial Electronics (ISIE)*, pages 831–836, 2015.

[89] D. A. Pola, H. F. Navarrete, M. E. Orchard, R. S. Rabie, M. A. Cerda, B. E. Olivares, J. F. Silva, P. A. Espinoza, and A. Perez. Particle-filtering-based discharge time prognosis for lithium-ion batteries with a statistical characterization of use profiles. *IEEE Transactions on Reliability*, 64(2):710–720, 2015.

[90] M. Orchard, F. Tobar, and G. Vachtsevanos. Outer feedback correction loops in particle filtering-based prognostic algorithms: Statistical performance comparison. *Studies in Informatics and Control*, 18(4):295–304, 2009.

[91] M. Orchard, L. Tang, B. Saha, K. Goebel, and G. Vachtsevanos. Risk-sensitive particle-filtering-based prognosis framework for estimation of remaining useful life in energy storage devices. *Studies in Informatics and Control*, 19(3):209–218, 2010.

[92] S. Sankararaman. Significance, interpretation, and quantification of uncertainty in prognostics and remaining useful life prediction. *Mechanical Systems and Signal Processing*, 52:228–247, 2015.

[93] H. Khalil and J. Grizzle. *Nonlinear systems*, volume 3. Prentice hall New Jersey, 1996.

[94] V. R. Saksena, J. O'reilly, and P. Kokotovic. Singular perturbations and time-scale methods in control theory: survey 1976–1983. *Automatica*, 20(3):273–293, 1984.

[95] H. E. Rauch. Order reduction in estimation with singular perturbation. In *4 th Symposium on Nonlinear Estimation Theory and Its Applications*, pages 231–241, 1974.

[96] A. H. Haddad. Linear filtering of singularly perturbed systems. *IEEE Transactions on Automatic Control*, 21(4):515–519, 1976.

[97] Z. Gajic and M. Lim. A new filtering method for linear singularly perturbed systems. *IEEE Transactions on Automatic Control*, 39(9):1952–1955, 1994.

[98] X. Shen, M. Rao, and Y. Ying. Decomposition method for solving Kalman filter gains in singularly perturbed systems. *Optimal Control Applications and Methods*, 14(1):67–73, 1993.

[99] K. W. Chang. Singular perturbations of a general boundary value problem. *SIAM Journal on Mathematical Analysis*, 3(3):520–526, 1972.

[100] D. S. Naidu and A. Rao. *Singular perturbation analysis of discrete control systems*, volume 1154. Springer-Verlag Berlin, 1985.

[101] D. S. Naidu. *Singular perturbation methodology in control systems*. Number 34. IET, 1988.

[102] B. S. Kim, Y. J. Kim, and M. T. Lim. LQG control for nonstandard singularly perturbed discrete-time systems. *Journal of dynamic systems, measurement, and control*, 126(4):860–864, 2004.

[103] M. R. Azimi-Sadjadi and K. Khorasani. Reduced order strip Kalman filtering using singular perturbation method. *IEEE Transactions on Circuits and Systems*, 37(2):284–290, 1990.

[104] A. Rao and D. S. Naidu. Singular perturbation method for Kalman filter in discrete systems. In *IEE Proceedings on Control Theory and Applications*, volume 131, pages 39–46, 1984.

[105] D. S. Naidu, C. Charalambous, K. L. Moore, and M. A. Abdelrahma. H_∞-optimal control of singularly perturbed discrete-time systems, and risk-sensitive control. In *the 33rd IEEE Conference on Decision and Control*, volume 2, pages 1706–1711, 1994.

[106] M. Aliyu and E. Boukas. H_∞-filtering for singularly perturbed nonlinear systems. *International Journal of Robust and Nonlinear Control*, 21(2):218–236, 2011.

[107] J. H. Park, H. C. Yeong, and N. S. Namachchivaya. Particle filters in a multiscale environment: homogenized hybrid particle filter. *Journal of Applied Mechanics*, 78(6):061001, 2011.

[108] P. Imkeller, N. S. Namachchivaya, N. Perkowski, and H. C. Yeong. A homogenization approach to multiscale filtering. *Procedia IUTAM*, 5:34–45, 2012.

[109] N. Daroogheh, N. Meskin, and K. Khorasani. Robust hybrid EKF approach for state estimation in multi-scale nonlinear singularly perturbed systems. In *53rd IEEE Conference on Decision and Control*, pages 1047–1054, 2014.

[110] A. Gelb. *Applied optimal estimation*. MIT press, 1974.

[111] F. E. Daum. Exact finite-dimensional nonlinear filters. *IEEE Transactions on Automatic Control*, 31(7):616–622, 1986.

[112] A. J. Krener and A. Duarte. A hybrid computational approach to nonlinear estimation. In *the 35th IEEE Conference on Decision and Control*, volume 2, pages 1815–1819, 1996.

[113] C. Jaganath, A. Ridley, and D. S. Bernstein. A SDRE-based asymptotic observer for nonlinear discrete-time systems. In *the American Control Conference*, pages 3630–3635, 2005.

[114] A. Smith, A. Doucet, N. de Freitas, and N. Gordon. *Sequential Monte Carlo methods in practice*. Springer Science & Business Media, 2013.

[115] G. Evensen. Sequential data assimilation for nonlinear dynamics: the ensemble Kalman filter. In *Ocean Forecasting*, pages 97–116. Springer, 2002.

[116] G. Evensen. The ensemble Kalman filter: Theoretical formulation and practical implementation. *Ocean dynamics*, 53(4):343–367, 2003.

[117] S. Gillijns and B. De Moor. Model error estimation in ensemble data assimilation. *Nonlinear Processes in Geophysics*, 14(1):59–71, 2007.

[118] J. Mandel, L. Cobb, and J. D. Beezley. On the convergence of the ensemble Kalman filter. *Applications of Mathematics*, 56(6):533–541, 2011.

[119] H. Moradkhani, S. Sorooshian, H. Gupta, and P. R. Houser. Dual state–parameter estimation of hydrological models using ensemble Kalman filter. *Advances in Water Resources*, 28(2):135–147, 2005.

[120] S. Gillijns, O. B. Mendoza, J. Chandrasekar, B. De Moor, D. S. Bernstein, and A. Ridley. What is the ensemble kalman filter and how well does it work? In *American Control Conference*, pages 6–pp, 2006.

[121] G. Triantafyllou, I. Hoteit, X. Luo, K. Tsiaras, and G. Petihakis. Assessing a robust ensemble-based Kalman filter for efficient ecosystem data assimilation of the cretan sea. *Journal of Marine Systems*, 125:90–100, 2013.

[122] G. Evensen. The ensemble Kalman filter for combined state and parameter estimation. *IEEE Control Systems*, 29(3):83–104, 2009.

[123] E. Ott, B. R. Hunt, I. Szunyogh, A. V. Zimin, E. J. Kostelich, M. Corazza, E. Kalnay, D. Patil, and J. A. Yorke. A local ensemble Kalman filter for atmospheric data assimilation. *Tellus A*, 56(5):415–428, 2004.

[124] J. L. Anderson. An ensemble adjustment Kalman filter for data assimilation. *Monthly weather review*, 129(12):2884–2903, 2001.

[125] J. L. Anderson and S. L. Anderson. A Monte Carlo implementation of the nonlinear filtering problem to produce ensemble assimilations and forecasts. *Monthly Weather Review*, 127(12):2741–2758, 1999.

[126] P. Sakov and P. R. Oke. A deterministic formulation of the ensemble Kalman filter: an alternative to ensemble square root filters. *Tellus A*, 60(2):361–371, 2008.

[127] S. J. Greybush, E. Kalnay, T. Miyoshi, K. Ide, and B. R. Hunt. Balance and ensemble Kalman filter localization techniques. *Monthly Weather Review*, 139(2):511–522, 2011.

[128] S. Otsuka and T. Miyoshi. A Bayesian optimization approach to multimodel ensemble kalman filter with a low-order model. *Monthly Weather Review*, 143(6):2001–2012, 2015.

[129] I. Fukumori and P. Malanotte-Rizzoli. An approximate kaiman filter for ocean data assimilation: An example with an idealized gulf stream model. *Journal of Geophysical Research: Oceans (1978–2012)*, 100(C4):6777–6793, 1995.

[130] M. A. Cane, A. Kaplan, R. N. Miller, B. Tang, E. C. Hackert, and A. J. Busalacchi. Mapping tropical pacific sea level: Data assimilation via a reduced state space Kalman filter. *Journal of Geophysical Research: Oceans (1978–2012)*, 101(C10):22599–22617, 1996.

[131] G. Evensen. Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *Journal of Geophysical Research: Oceans*, 99(C5):10143–10162, 1994.

[132] E. Kwiatkowski and J. Mandel. Convergence of the square root ensemble Kalman filter in the large ensemble limit. *SIAM/ASA Journal on Uncertainty Quantification*, 3(1):1–17, 2015.

[133] M. K. Tippett, J. L. Anderson, C. H. Bishop, T. M. Hamill, and J. S. Whitaker. Ensemble square root filters. *Monthly Weather Review*, 131(7):1485–1490, 2003.

[134] C. H. Bishop, B. J. Etherton, and S. J. Majumdar. Adaptive sampling with the ensemble transform Kalman filter. part I: Theoretical aspects. *Monthly weather review*, 129(3):420–436, 2001.

[135] R. J. Lorentzen and G. Nævdal. An iterative ensemble Kalman filter. *IEEE Transactions on Automatic Control*, 56(8):1990–1995, 2011.

[136] G. Niu. *Data-driven Technology for Engineering System Health Management: Design Approach, Feature Construction, Fault Diagnosis, Prognostics, Fusion and Decisions*. Springer, 2016.

[137] S. Shrikhande, P. Varde, and D. Datta. Prognostics and health management: Methodologies & soft computing techniques. In *Current Trends in Reliability, Availability, Maintainability and Safety Conference*, pages 213–227. 2016.

[138] L. Cristaldi, G. Leone, R. Ottoboni, S. Subbiah, and S. Turrin. A comparative study on data-driven prognostic approaches using fleet knowledge. In *IEEE International Instrumentation and Measurement Technology Conference Proceedings*, pages 1–6, 2016.

[139] V. George, L. Frank, R. Michael, H. Andrew, and W. Biqing. Intelligent fault diagnosis and prognosis for engineering systems, 2006.

[140] ZN.S. Vanini, N. Meskin, and K. Khorasani. Multiple-model sensor and components fault diagnosis in gas turbine engines using autoassociative neural networks. *Journal of Engineering for Gas Turbines and Power*, 136(9):091603, 2014.

[141] ZN.S. Vanini, K. Khorasani, and N. Meskin. Fault detection and isolation of a dual spool gas turbine engine using dynamic neural networks and multiple model approach. *Information Sciences*, 259:234–251, 2014.

[142] M. S. Jha, G. Dauphin-Tanguy, and B. Ould-Bouamama. Particle filter based hybrid prognostics for health monitoring of uncertain systems in bond graph framework. *Mechanical Systems and Signal Processing*, 75:301–329, 2016.

[143] R. Razavi-Far, M. Farajzadeh-Zanjani, S. Chakrabarti, and M. Saif. Data-driven prognostic techniques for estimation of the remaining useful life of lithium-ion batteries. In *IEEE International Conference on Prognostics and Health Management (ICPHM)*, pages 1–8, 2016.

[144] J. Lee, F. Wu, W. Zhao, M. Ghaffari, L. Liao, and D. Siegel. Prognostics and health management design for rotary machinery systems, reviews, methodology and applications. *Mechanical Systems and Signal Processing*, 42(1):314–334, 2014.

[145] E. Tsoutsanis, N. Meskin, M. Benammar, and K. Khorasani. Transient gas turbine performance diagnostics through nonlinear adaptation of compressor and turbine maps. *Journal of Engineering for Gas Turbines and Power*, 137(9):091201, 2015.

[146] S. Rahme and N. Meskin. Adaptive sliding mode observer for sensor fault diagnosis of an industrial gas turbine. *Control Engineering Practice*, 38:57–74, 2015.

[147] A. K. Jardine, D. Lin, and D. Banjevic. A review on machinery diagnostics and prognostics implementing condition-based maintenance. *Mechanical systems and signal processing*, 20(7):1483–1510, 2006.

[148] E. Tsoutsanis, N. Meskin, M. Benammar, and K. Khorasani. Dynamic performance simulation of an aeroderivative gas turbine using the matlab simulink environment. In *ASME 2013 International Mechanical Engineering Congress and Exposition*. American Society of Mechanical Engineers.

[149] M. Schwabacher. A survey of data-driven prognostics. In *the AIAA Infotech Aerospace Conference*, pages 1–5, 2005.

[150] C. Chen, B. Zhang, and G. Vachtsevanos. Prediction of machine health condition using neuro-fuzzy and Bayesian algorithms. *IEEE Transactions on instrumentation and Measurement*, 61(2):297–306, 2012.

[151] B. T. Thumati, M. A. Feinstein, and S. Jagannathan. A model-based fault detection and prognostics scheme for takagi–sugeno fuzzy systems. *IEEE Transactions on Fuzzy Systems*, 22(4):736–748, 2014.

[152] N. Puggina and M. Venturini. Development of a statistical methodology for gas turbine prognostics. *Journal of Engineering for Gas Turbines and Power*, 134(2):022401, 2012.

[153] Y. Zhou and M. Huang. Lithium-ion batteries remaining useful life prediction based on a mixture of empirical mode decomposition and ARIMA model. *Microelectronics Reliability*, 2016.

[154] Z. Wang, L. Liu, H. Zhang, and G. Xiao. Fault-tolerant controller design for a class of nonlinear MIMO discrete-time systems via online reinforcement learning algorithm. *IEEE Transactions on Systems, Man and Cybernetics*, 45(5):611–622, 2016.

[155] C. Chen, B. Zhang, G. Vachtsevanos, and M. Orchard. Machine condition prediction based on adaptive neuro–fuzzy and high-order particle filtering. *IEEE Transactions on Industrial Electronics*, 58(9):4353–4364, 2011.

[156] F. O. Heimes. Recurrent neural networks for remaining useful life estimation. In *IEEE Prognostics and Health Management Conference (PHM)*, pages 1–6, 2008.

[157] E. Naderi, N. Meskin, and K. Khorasani. Nonlinear fault diagnosis of jet engines by using a multiple model-based approach. *Journal of Engineering for Gas Turbines and Power*, 134(1):011602, 2012.

[158] N. Meskin, E. Naderi, and K. Khorasani. A multiple model-based approach for fault diagnosis of jet engines. *IEEE Transactions on Control Systems Technology*, 21(1):254–262, 2013.

[159] N L R Group. Gsp software, June 2011.

[160] M. Naeem, R. Singh, and D. Probert. Implications of engine deterioration for creep life. *Applied energy*, 60(4):183–223, 1998.

[161] N. Daroogheh, A. Vatani, M. Gholamhossein, and K. Khorasani. Engine life evaluation based on a probabilistic approach. In *ASME International Mechanical Engineering Congress and Exposition*, pages 347–358, 2012.

[162] M. Naeem, R. Singh, and D. Probert. Implications of engine deterioration for a high-pressure turbine-blade's low-cycle fatigue (LCF) life-consumption. *International journal of fatigue*, 21(8):831–847, 1999.

[163] S. Haykin. *Kalman filtering and neural networks*. Wiley Online Library, 2001.

[164] F. L. Lewis and F. Lewis. *Optimal estimation: with an introduction to stochastic control theory*. Wiley New York et al., 1986.

[165] L. Ljung and T. Söderström. *Theory and practice of recursive identification*. MIT press, 1983.

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

[167] C. Musso, N. Oudjane, and F. LeGland. Improving regularised particle filters. *Sequential Monte Carlo methods in practice*, pages 247–271, 2001.

[168] A. Budhiraja, L. Chen, and C. Lee. A survey of numerical methods for nonlinear filtering problems. *Physica D: Nonlinear Phenomena*, 230(1):27–36, 2007.

[169] H. Xiao-Li, T. Schon, and L. Ljung. A general convergence result for particle filtering. *IEEE Transactions on Signal Processing*, 59(7):3424–3429, 2011.

[170] R. Karlsson, T. Schön, and F. Gustafsson. Complexity analysis of the marginalized particle filter. *IEEE Transactions on Singnal Processing*, 53(11):4408–4411, 2005.

[171] V. Venkatasubramanian, R. Rengaswamy, K. Yin, and S.N. Kavuri. A review of process fault detection and diagnosis: Part I: Quantitative model-based methods. *Computers & Chemical Engineering*, 27(3):293–311, 2003.

[172] R. Isermann. Process fault detection based on modeling and estimation methods - a survey. *Automatica*, 20(4):387–404, 1984.

[173] D. Wang, M. Yu, C. B. Low, and S. Arogeti. *Model-based Health Monitoring of Hybrid Systems*. Springer, 2013.

[174] S. Simani. *Model-based fault diagnosis in dynamic systems using identification techniques*. PhD thesis, dell Universita di Ferrara, 2003.

[175] R. Kohavi and F. Provost. Confusion matrix. *Machine Learning*, 30(2-3):271–274, 1998.

[176] A. Saxena, J. Celaya, B. Saha, S. Saha, and K. Goebel. Metrics for offline evaluation of prognostic performance. *International Journal of Prognostics and Health Management*, 1(1):4–23, 2010.

[177] A. Saxena, J. Celaya, E. Balaban, K. Goebel, B. Saha, S. Saha, and M. Schwabacher. Metrics for evaluating performance of prognostic techniques. In *International Conference on Prognostics and Health Management*, 2008.

[178] S. Uckun, K. Goebel, and P. Lucas. Standardizing research methods for prognostics. In *International Conference on Prognostics and Health Management*, 2008.

[179] X. Guan, Y. Liu, R. Jha, A. Saxena, J. Celaya, and K. Geobel. Comparison of two probabilistic fatigue damage assessment approaches using prognostic performance metrics. *International Journal of Prognostics and Health Management*, 1(005), 2011.

[180] M. Daigle and K. Goebel. Multiple damage progression paths in model-based prognostics. In *IEEE Aerospace Conference*, 2011.

[181] M. E. Orchard, L. Tang, K. Goebel, and G. Vachtsevanos. A novel RSPF approach to prediction of high-risk, low-probability failure events. In *Annual Conference of the Prognostics and Health Management Society*, 2009.

[182] L. Tang, M. E. Orchard, K. Goebel, and G. Vachtsevanos. Novel metrics and methodologies for the verification and validation of prognostic algorithms. In *IEEE Aerospace Conference*, pages 1–8, 2011.

[183] J. Harrison and M. West. *Bayesian Forecasting & Dynamic Models*. Springer, 1999.

[184] A. Soylemezoglu, S. Jagannathan, and C. Saygin. Mahalanobis-taguchi system as a multi-sensor based decision making prognostics tool for centrifugal pump failures. *IEEE Transactions on Reliability*, 60(4):864–878, 2011.

[185] N. Daroogheh, N. Meskin, and K. Khorasani. A novel particle filter parameter prediction scheme for failure prognosis. In *American Control Conference, 2014*, pages 1735–1742, 2014.

[186] A. C. Harvey. *Forecasting, Structural Time Series Models and the Kalman Filter*. Cambridge University Press, 1990.

[187] G. Verdier and A. Ferreira. Adaptive Mahalanobis distance and k -nearest neighbor rule for fault detection in semiconductor manufacturing. *IEEE Transactions on Semiconductor Manufacturing*, 24(1):59–68, 2011.

[188] L. Ljung. System Identification: Theory for the User. *Prentice Hall Information and System Sciences Series, New Jersey*, 7632, 1987.

[189] J. D. Hamilton. *Time series analysis*, volume 2. Princeton university press Princeton, 1994.

[190] S. Jaggia. Forecasting with ARMA models. *Case Studies In Business, Industry And Government Statistics*, 4(1):59–65, 2014.

[191] E. Ahmed, A. Clark, and G. Mohay. A novel sliding window based change detection algorithm for asymmetric traffic. In *IEEE International Conference on Network and Parallel Computing*, pages 168–175, 2008.

[192] R. A. Smith. Matrix equation xa+bx=c. *SIAM Journal on Applied Mathematics*, 16(1):198–201, 1968.

[193] N. Oudjane and C. Musso. Progressive correction for regularized particle filters. In *the Third IEEE International Conference on Information Fusion*, volume 2, pages THB2–10, 2000.

[194] G. H. Hardy, J. E. Littlewood, and G. Polya. Inequalities. reprint of the 1952 edition. cambridge mathematical library, 1988.

[195] A. Doucet, S. Godsill, and C. Andrieu. On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and computing*, 10(3):197–208, 2000.

[196] M. Daigle, I. Roychoudhury, S. Narasimhan, S. Saha, B. Saha, and K. Goebel. Investigating the effect of damage progression model choice on prognostics performance. In *the Annual Conference of the Prognostics and Health Management Society*, 2011.

[197] M. Naeem, R. Singh, and D. Probert. Implications of engine's deterioration upon an aero-engine HP turbine blade's thermal fatigue life. *International journal of fatigue*, 22(2):147–160, 2000.

[198] Y. Li and P. Nilkitsaranont. Gas turbine performance prognostic for condition-based maintenance. *Applied Energy*, 86:2152–2161, 2009.

[199] N. Daroogheh, A. Baniamerian, N. Meskin, and K. Khorasani. Prognosis and health monitoring of nonlinear systems using a hybrid scheme through integration of particle filters and neural networks. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, PP(99):1–15, 2016.

[200] A. Doucet, S. Godsill, and C. Andrieu. On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and computing*, 10(3), 2000.

[201] Q. Zhang and A. Benveniste. Wavelet networks. *IEEE Transactions on Neural Networks*, 3(6):889–898, 1992.

[202] M. T. Hagan and M. B. Menhaj. Training feedforward networks with the marquardt algorithm. *IEEE Transactions on Neural Networks*, 5(6):989–993, 1994.

[203] X. Si, W. Wang, C. Hu, and D. Zhou. Remaining useful life estimation–a review on the statistical data driven approaches. *European Journal of Operational Research*, 213(1):1–14, 2011.

[204] L. Socha. Exponential stability of singularly perturbed stochastic systems. *IEEE Transactions on Automatic Control*, 45(3):576–580, 2000.

[205] J. P. Barbot and N. Pantalos. Using symbolic calculus for singularly perturbed nonlinear systems. In *Algebraic Computing in Control*, pages 40–49. Springer, 1991.

[206] P. Kokotovic, H. Khali, and J. O'reilly. *Singular perturbation methods in control: analysis and design*, volume 25. Society for Industrial Mathematics, 1987.

[207] M. Spong, K. Khorasani, and P. Kokotovic. An integral manifold approach to the feedback control of flexible joint robots. *IEEE Journal on Robotics and Automation*, 3(4):291–300, 1987.

[208] J. P. Barbot, M. Djemai, S. Monaco, and D. Normand-Cyrot. Analysis and control of nonlinear singularly perturbed systems under sampling. *Control and Dynamic Systems*, 79:203–246, 1996.

[209] J. P. Barbot, S. Monaco, D. Normand-Cyrot, and N. Pantalos. Discretization schemes for nonlinear singularly perturbed systems. In *the 30th IEEE Conference on Decision and Control*, pages 443–448, 1991.

[210] R. E. Kalman. A new approach to linear filtering and prediction problems. *Journal of Fluids Engineering*, 82(1):35–45, 1960.

[211] B. Teixeira, J. Chandrasekar, H. J. Palanthandalam-Madapusi, L. Tôrres, L. Aguirre, and D. S. Bernstein. Gain-constrained kalman filtering for linear and nonlinear systems. *IEEE Transactions on Signal Processing*, 56(9):4113–4123, 2008.

[212] C. Kuehn. *Multiple Time Scale Dynamics*. Springer, 2015.

[213] Rolls Royce. *The jet engine*. John Wiley & Sons, 2015.