

Chapter 2

Recursive Bayesian Estimation of Partially Observed Dynamic Systems

Abstract In the current Chapter, recursive Bayesian inference of partially observed dynamical systems is reviewed. As a tool for structural system identification, nonlinear Bayesian filters are applied to dual estimation problem of linear and nonlinear dynamical systems. In so doing, dual estimation of state and parameters of structural state space models is considered; EKF, SPKF, PF and EK-PF are used for parameter identification and state estimation. Dealing with a SDOF structure, it is shown that the hybrid EK-PF filter is able to furnish a reasonable estimation of parameters of nonlinear constitutive models. Assessment of SDOF systems is followed by identification of multi storey buildings. In this regard, performances of the EK-PF and EKF algorithms are compared, and it is concluded that they are nearly the same, and by an increase in the number of storeys of the building, both of the algorithms fail to provide an unbiased estimate of the parameters (stiffness of the storeys). Therefore, they are not reliable tools to monitor state and parameters of multi storey systems.

2.1 Introduction

Recursive inference of the dynamics of a system through noisy observations is normally pursued within a Bayesian framework. As a result, if there is a priori information available on probability distribution of observable quantities of the system and there is a correlation between observable and hidden quantities of the system, Bayes probability concept is employed to estimate probability distribution of the hidden state variables. Extensive variety of applications are exploited by using such approach namely: in econometrics to estimate volatility in the market (Ishihara and Omori 2012; Yang and Lee 2011; Miazhyńska et al. 2006), for a review on the literature see (Creal 2012). In field of robotics, this approach is applied to develop behaviors for robots (Lazkano et al. 2007), system identification of the robots (Ting et al. 2011), and their localization (Zhou and Sakane 2007). In biology, this approach is employed for molecular characterization of diseases

(Alvarado Mora et al. 2011), finding linkage in DNA (Allen and Darwiche 2008; Biedermann and Taroni 2012) and for characterization of genomic data (Caron et al. 2012). In image processing, this approach is used to diagnose diseases from medical images (Mitra et al. 2005), for image segmentation (Adelino and Ferreira da Silva 2009), and for image retrieval (Duan et al. 2005). Moreover, this approach is employed in the following fields such as: object tracking and radars (Jay et al. 2003; Velarde et al. 2008; White et al. 2009); in speech enhancement (Saleh and Niranjana 2001; Yahya et al. 2010); in mechanical characterization and parameter identification of materials (Corigliano and Mariani 2004, 2001a, b; Bittanti et al. 1984), mechanical system identification (Mariani and Ghisi 2007; Mariani and Corigliano 2005) and many other fields which are not mentioned for the sake of brevity. The aforementioned instances are just a few fields of application of Bayesian inference schemes; their diversity proves the versatility of such approach in problem solving processes.

Estimation of state and parameters of a structural system are simultaneously dealt with in a recursive fashion in this chapter of the monograph. As new observations become available, the information concerning the current state of the system, which is attained through a model of the system, is updated based on the measured observation. This objective is perceived by utilizing four recursive Bayesian filters, namely: the extended Kalman filter (EKF), the sigma-point Kalman filter (S-PKF), the particle filter (PF), and a newly proposed hybrid extended Kalman particle filter (EKPF). Therefore, to avoid shadowing effects of high dimensional structures, a single degree-of-freedom system has primarily been considered. The performances of the filters are standardized to simultaneously estimate state and parameters of a nonlinear constitutive model of the system. After the performance of the filters working with a single degree-of-freedom structure has been verified, we move to the analysis of multi degree-of-freedom (DOF) structures. To accomplish this aim, a shear type of buildings has been considered. It should be emphasized that although Bayesian filters under the study have been adopted in the other fields such as automatic control, their application in the field of structural engineering demands further investigations. The author of this book has coauthored three articles on peer reviewed international journals on this topics (Eftekhara Azam and Mariani 2012; Eftekhara Azam et al. 2012a, b). The proceeding parts of this Chapter is classified as follows: in Sect. 2.2, the dual estimation concept for simultaneous estimation of state and parameters of a state-space model is reviewed. General frames of the recursive Bayesian inference techniques are discussed in Sect. 2.3; moreover, the Kalman filter, as the optimal filter of linear state-space models is devoted to Sect. 2.4. Approximate Bayesian filters for nonlinear systems are dealt with in Sect. 2.5; furthermore in Sect. 2.6 the numerical results concerning dual estimation of states and parameters of single DOF and multi DOFs structures are presented. The Chapter is eventually concluded in Sect. 2.7, where the limitations filters under the study are discussed together with our remedy to solve the issue when applied to simultaneous state and parameter estimation of high dimensional problems.

2.2 Dual Estimation of States and Parameters of Mechanical Systems

In this research, the emphasis is on civil structures. Therefore, mechanical systems whose dynamics is governed by the famous set of ordinary differential equations are addressed which governs evolution of their dynamic:

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{D}\dot{\mathbf{u}} + \mathbf{R}(\mathbf{u}, t) = \mathbf{F}(t) \quad (2.1)$$

where \mathbf{M} is assigned as the mass matrix, \mathbf{D} represents the damping matrix; $\mathbf{R}(\mathbf{u}, t)$ stands for possibly displacement dependent internal force, whereas $\mathbf{F}(t)$ is designated as the loading vector; \mathbf{u} , $\dot{\mathbf{u}}$ and $\ddot{\mathbf{u}}$ are the nodal displacements, velocities and accelerations, respectively. Since measurements are normally completed in discrete time, our attention is limited to a discrete time formulation, where it is assumed that a part of displacements or accelerations of the system are measured in evenly spaced time grids.

To embed the mathematical model into algorithms designed for recursive Bayesian inference, we represent the dynamics of the system in a state-space form; the details concerning the state-space representation of the mathematical model (2.1) are presented in the following Sections. Throughout the book, displacement, velocity and acceleration quantities of the response of the structure are assigned by the word ‘state’ and we intend to use ‘parameters’ which represent the coefficients of the internal force term (in linear elastic case, components of the stiffness matrix). The state vector \mathbf{z} thus contains \mathbf{u} , $\dot{\mathbf{u}}$ and $\ddot{\mathbf{u}}$, namely:

$$\mathbf{z}_k = \begin{bmatrix} \mathbf{u}_k \\ \dot{\mathbf{u}}_k \\ \ddot{\mathbf{u}}_k \end{bmatrix} \quad (2.2)$$

while parameter vector $\boldsymbol{\vartheta}_k$ collects several unknown parameters of the system.

The state space representation of the system thus is expressed as:

$$\mathbf{z}_k = \mathbf{f}_k^z(\mathbf{z}_{k-1}; \boldsymbol{\vartheta}_{k-1}) + \mathbf{v}_k^z \quad (2.3)$$

$$\mathbf{y}_k = \mathbf{H}_k^z \mathbf{z}_k + \mathbf{w}_k \quad (2.4)$$

where for any time interval $[t_{k-1}t_k]$, $\mathbf{f}_k^z(\cdot)$ is a function of the state \mathbf{z}_{k-1} and parameters $\boldsymbol{\vartheta}_{k-1}$ of the system, and evolves the state of the system \mathbf{z}_{k-1} to obtain \mathbf{z}_k . \mathbf{H}_k^z quantifies the correlation between the state and the observable part of the system, at any given time instant; the name of Eq. (2.4), observation equation, originates from the aforementioned fact. \mathbf{v}_k^z and \mathbf{w}_k are the zero mean, uncorrelated Gaussian processes with covariance matrices \mathbf{V}^z and \mathbf{W} , respectively. Generally, observation equation may take any form; however, in the present study, it is reasonably assumed that observation process consists of a part of the state vector, namely displacements and/or accelerations of several representative points. As a result, the observation equation can be expressed as a sum of a linear mapping of

the state through a Boolean matrix (\mathbf{H}_k^z) and an additive, uncorrelated Gaussian noise stemming from uncertainty of measurement sensor.

In this study, the major mission of Bayesian filters, beyond estimating hidden part of the state vector, will be the calibration of system model parameters in an online method. At each time interval $[t_{k-1}t_k]$, on the basis of the information contained in the latest observation \mathbf{y}_k , the algorithms update former knowledge of the parameter $\boldsymbol{\vartheta}_{k-1}$ to yield $\boldsymbol{\vartheta}_k$. To accomplish this objective, dual estimation of states and parameters are considered; hence, the parameter vector $\boldsymbol{\vartheta}_k$ is increased by defining the state vector (Mariani and Corigliano 2005):

$$\mathbf{x}_k = \begin{bmatrix} \mathbf{z}_k \\ \boldsymbol{\vartheta}_k \end{bmatrix}. \quad (2.5)$$

In addition to the conventional form of state-space equation, which is composed of evolution and observation equations, dual estimation is pursued via an extra vectorial equation governing the evolution of the parameters over time according to:

$$\boldsymbol{\vartheta}_k = \boldsymbol{\vartheta}_{k-1} + \mathbf{v}_k^\vartheta. \quad (2.6)$$

The intuitive idea underlying this extra equation is to allow the unknown parameters of the system to change over time, starting from an initial guess and hopefully converge on an unbiased estimate. The possibility of variation to parameters is provided by white Gaussian fictitious noise \mathbf{v}_k^ϑ , added to parameter evolution. Moreover, the intensity of such a noise should be tuned in order to obtain an unbiased and converging estimate for the parameters (Bittanti and Savaresi 2000). The state-space equation governing evolution of the increased state thus is expressed as:

$$\mathbf{x}_k = \mathbf{f}_k(\mathbf{x}_{k-1}) + \mathbf{v}_k \quad (2.7)$$

$$\mathbf{y}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{w}_k \quad (2.8)$$

where $\mathbf{f}_k(\cdot)$, maps the extended state vector \mathbf{x}_k over time; therefore, it features both Eqs. (2.3) and (2.6) in one unique equation.

2.3 Recursive Bayesian Inference

The inference problem can be considered as recursively estimating the expected value $E[\mathbf{x}_k | \mathbf{y}_{1:k}]$ of the state of the system, conditioned on the observations. If the initial probability density function (PDF) $p(\mathbf{x}_0 | \mathbf{y}_0) = p(\mathbf{x}_0)$ of the state vector is known, the problem consists in estimating $p(\mathbf{x}_k | \mathbf{y}_{1:k})$, assuming that the conditional PDF $p(\mathbf{x}_{k-1} | \mathbf{y}_{1:k-1})$ is available. The problem can be broken down into in two stages of prediction and update. As far as the prediction stage, the Chapman–Kolmogorov

equation furnishes the a-priori estimate of the state PDF at t_k (Arulampalam et al. 2002):

$$p(\mathbf{x}_k|\mathbf{y}_{1:k-1}) = \int p(\mathbf{x}_k|\mathbf{x}_{k-1})p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k-1})d\mathbf{x}_{k-1}. \quad (2.9)$$

In the updating stage, as soon as the new observation \mathbf{y}_k becomes available, Bayes rule is profited to apply correction on the PDF of the state (Cadini et al. 2009):

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) = \varsigma p(\mathbf{y}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{y}_{1:k-1}) \quad (2.10)$$

where ς stands for a normalizing constant which depends on the likelihood function of the observation process. The Eqs. (2.9) and (2.10) collectively forge the basis for any Bayesian recursive inference scheme. The analytical solution of the integral in (2.9) is not possible except for a limited category of problems, namely systems which are formulated by linear state space equations and disturbed by uncorrelated white Gaussian noises (Eftekhar Azam et al. 2012a). In case of a general nonlinear problem, one should make recourse to approximate solutions, either by approximating the nonlinear evolution equations via linearization (Corigliano and Mariani 2004) or via discrete approximate representation of the PDF of the state vector (Mariani and Ghisi 2007; Doucet and Johansen 2009; Doucet and Johansen 2009). In the next Section, the major features of the analytical solution available for linear Gaussian state space model are examined, and is followed by the Sect. 2.5 which handles approximate solutions for nonlinear state-space models.

2.4 Linear Dynamic State Space Equations: Optimal Closed Form Estimator

As addressed in the preceding section, recursive Bayesian estimation of linear Gaussian state-space models can be calculated analytically. A linear discrete state-space model is considered which can be obtained by substituting the arbitrary evolution equation $\mathbf{f}_k(\cdot)$ in Eqs. (2.7) and (2.8) by a linear operator \mathbf{F}_k . Therefore, the state-space equations of such a system are expressed as:

$$\mathbf{x}_k = \mathbf{F}_k\mathbf{x}_{k-1} + \mathbf{v}_k, \quad (2.11)$$

$$\mathbf{y}_k = \mathbf{H}_k\mathbf{x}_k + \mathbf{w}_k. \quad (2.12)$$

If the primary probability distribution of the state is Gaussian, it is straightforward to display that a linear operator does not change the Gaussian PDF over time (Kalman 1960). That is, in the Chapman–Kolmogorov integral at any arbitrary time instant t_k , the functional form of both integrands is a priori known; moreover, $p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k-1})$ is constantly a Gaussian probability density function, and $p(\mathbf{x}_k|\mathbf{x}_{k-1})$ is by definition a Gaussian function as well. As a result, the integral can

Table 2.1 Kalman Filter algorithm

- Initialization at time t_0 :	$\hat{\mathbf{x}}_0 = \mathbb{E}[\mathbf{x}_0]$ $\mathbf{P}_0 = \mathbb{E}[(\mathbf{x}_0 - \hat{\mathbf{x}}_0)(\mathbf{x}_0 - \hat{\mathbf{x}}_0)^T]$
- At time t_k , for $k = 1, \dots, N_f$:	
• Prediction stage:	1. Evolution of state and prediction of covariance $\mathbf{x}_k^- = \mathbf{F}_k \mathbf{x}_{k-1}$ $\mathbf{P}_k^- = \mathbf{F}_k \mathbf{P}_{k-1} \mathbf{F}_k^T + \mathbf{V}$
• Update stage:	1. Calculation of Kalman gain: $\mathbf{G}_k = \mathbf{P}_k^- \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{W})^{-1}$ 2. Improve predictions using latest observation: $\hat{\mathbf{x}}_k = \mathbf{x}_k^- + \mathbf{G}_k (\mathbf{y}_k - \mathbf{H}_k \mathbf{x}_k^-)$ $\mathbf{P}_k = \mathbf{P}_k^- - \mathbf{G}_k \mathbf{H}_k \mathbf{P}_k^-$

be calculated analytically. Kalman introduced a well-known filter which is the optimal estimator for linear systems with uncorrelated Gaussian noise in his seminal study (Kalman 1960); the filter provides an online estimation of first and second order statistics of a state space model, and it includes a prediction stage which is simply an evolution of state over time. In the updating stage, by computing the Kalman gain \mathbf{G}_k , the filter enhances the predicted values furnished in previous stage. Readers are referred to Table 2.1 for a detailed description and algorithmic implementation of the Kalman filter (KF).

2.4.1 The Kalman Filter

In many real life problems, neither the dynamics of the system takes a linear form nor the uncertainties of transition equation which may be regarded as Gaussian distributions. Even if the initial distribution of the uncertainties could be assumed Gaussian, a nonlinear state-space model would change the distribution over time (Mariani and Ghisi 2007). Therefore, an optimal closed form solution will not be available for a general nonlinear problem (Doucet and Johansen 2009).

In a mechanical system, the source of nonlinearity might be the material response to loading (Corigliano and Mariani 2001a, b; Corigliano 1993); however, even if the material behavior is linear, dual estimation of states and parameter will result in a bilinear (nonlinear) state space model (Ljung 1999). We illustrate this issue via an intuitive example by considering the following linear state space model:

$$z_k = az_{k-1} + b + v_k^z \quad (2.13)$$

$$y_k = H z_k + w_k \quad (2.14)$$

where z_k and y_k denote the state and the observation of the system at a given time instant t_k ; a and b represent the linear transition for the state in a given time interval $[t_{k-1}, t_k]$, while H links the hidden state z_k to the observation process. v_k^z and w_k denote the zero mean white Gaussian processes which quantify evolution and measurement inaccuracies, respectively. In case one is only interested in estimating the state of the system z_k , we already know the Kalman filter furnishes the optimal estimation; however, let us imagine one is also interested in an online estimation of the parameters of the state space model. For the sake of simplicity, we assume that only parameter a is of interest. As aforementioned, the trick in dual estimation framework is to collect the unknown parameter a into the extended state vector \mathbf{x}_k and try to track the dynamics of such system via recursive Bayesian inference algorithms. It is noteworthy that even though parameter a is stationary by definition, the parameter is allowed to vary in the formulation of dual estimation. In this regard, a transition equation governing evolution of the parameter is introduced:

$$a_k = a_{k-1} + v_k^a. \quad (2.15)$$

Equation (2.15), together with (2.13) and (2.14), constitute the required state-space model for dual estimation of states and parameters. The augmented state vector \mathbf{x}_k thus becomes $\mathbf{x}_k = [z_k \quad a_k]^T$, where $\mathbf{x}_k(1) = z_k$ and $\mathbf{x}_k(2) = a_k$; consequently Eqs. (2.13–2.15) become:

$$\mathbf{x}_k(1) = \mathbf{x}_{k-1}(2)\mathbf{x}_{k-1}(1) + b \quad (2.16)$$

$$\mathbf{x}_k(2) = \mathbf{x}_{k-1}(2) + v_k^a \quad (2.17)$$

$$y_k = H\mathbf{x}_k(1) + w_k \quad (2.18)$$

or, in matrix form:

$$\begin{bmatrix} \mathbf{x}_k(1) \\ \mathbf{x}_k(2) \end{bmatrix} = \begin{bmatrix} \mathbf{x}_{k-1}(2)\mathbf{x}_{k-1}(1) \\ \mathbf{x}_{k-1}(2) \end{bmatrix} + \begin{bmatrix} v_k^z \\ v_k^a \end{bmatrix} + \begin{bmatrix} b \\ 0 \end{bmatrix} \quad (2.19)$$

$$y_k = [H \quad 0] \begin{bmatrix} \mathbf{x}_k(1) \\ \mathbf{x}_k(2) \end{bmatrix} + w_k. \quad (2.20)$$

It is evident that Eq. (2.19) is a nonlinear mapping over the given time interval $[t_{k-1}, t_k]$. The aforementioned fact, together with consideration that many real life problems are nonlinear, substantiates the need for Bayesian inference algorithms targeting general nonlinear, non-Gaussian problems. The following Section is devoted to review the approximate solutions available in the literature to deal recursive Bayesian estimation of nonlinear state-space models.

2.5 Nonlinear Dynamic State Space Equations: Approximate Bayesian Estimators

Most of the problems in the real problems, as well as all the problems related to the identification of the parameters of the systems by use of dual estimation concept lead to nonlinear state-space models. Hence, developing nonlinear versions of the KF seemed inevitable from the very beginning. Next subchapter reviews the main concepts behind the extension of the KF to the nonlinear problems.

2.5.1 The Extended Kalman Filter

A simple remedy to deal with nonlinear state-space models is through an extension of the Kalman filter, where for each time instant t_k , the nonlinear state mapping $f_k(\mathbf{x}_{k-1})$ is linearized by a first order truncation of a Taylor series expansion around \mathbf{x}_{k-1} . To accomplish this goal, the Jacobian of the evolution equation is used as a surrogate for linear transition matrices in order to update covariance (Gelb 1974); subsequently, the Kalman gain is used to update state statistics. This procedure is the extension of the Kalman filter for nonlinear state space models; thus its name extended Kalman filter (EKF). The extended Kalman filter assumes the prior $p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k-1})$ to be Gaussian; however, even if initially Gaussian, a nonlinear mapping changes its probability distribution. Moreover, a severely nonlinear mapping of state might change the probability distribution into a tailed or a bimodal distribution (Adelino and Ferreira da Silva 2009; Van der Merwe 2004) and causes bias in the estimates furnished by the EKF. In addition, the approximation of the state mapping via its Jacobian is not accurate enough in several cases; for instance, it does not consider the stochastic nature of the state vector, and the effect of the neglected terms may become considerable. As a consequence, the approximation may lead to an inconsistent estimation of the covariance; hence, a bias or divergence may occur in estimation of the state (Julier and Uhlmann 1997). For a detailed description of EKF algorithm see Table 2.2, where $\nabla_x f_k(\mathbf{x})|_{\mathbf{x}=\mathbf{x}_{k-1}}$ denotes the Jacobian of $f_k(\mathbf{x})$ at $\mathbf{x} = \mathbf{x}_{k-1}$. To Alleviate the aforementioned issues posed by highly nonlinear models the first remedy has been the development of the sigma-point Kalman filter which will be discussed in the next subsection.

2.5.2 The Sigma-Point Kalman Filter

In case of severely nonlinear systems, the successive linearization approach may be inaccurate (Mariani 2009b). For certain problems, it may be practically difficult to adopt: in case of a non holonomic material behavior, to calculate the Jacobian,

Table 2.2 Extended Kalman filter algorithm

	- Initialization at time t_0 : $\hat{\mathbf{x}}_0 = \mathbb{E}[\mathbf{x}_0]$ $\mathbf{P}_0 = \mathbb{E}\left[(\mathbf{x}_0 - \hat{\mathbf{x}}_0)(\mathbf{x}_0 - \hat{\mathbf{x}}_0)^T\right]$
	- At time t_k , for $k = 1, \dots, N_f$:
	• Prediction stage:
	1. Computing process model Jacobian: $\mathbf{F}_k = \nabla_{\mathbf{x}} \mathbf{f}_k(\mathbf{x}) _{\mathbf{x}=\mathbf{x}_{k-1}}$
	2. Evolution of state and prediction of covariance:
	$\mathbf{x}_k^- = \mathbf{F}_k \mathbf{x}_{k-1}$ $\mathbf{P}_k^- = \mathbf{F}_k \mathbf{P}_{k-1} \mathbf{F}_k^T + \mathbf{V}$
	• Update stage:
	1. Calculation of Kalman gain: $\mathbf{G}_k = \mathbf{P}_k^- \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{W})^{-1}$
	2. Improve predictions using latest observation: $\hat{\mathbf{x}}_k = \mathbf{x}_k^- + \mathbf{G}_k (\mathbf{y}_k - \mathbf{H}_k \mathbf{x}_k^-)$ $\mathbf{P}_k = \mathbf{P}_k^- - \mathbf{G}_k \mathbf{H}_k \mathbf{P}_k^-$

one should know if the current state of the system proceeds toward loading or unloading (Mariani and Ghisi 2007). The difficulty in estimation of the Jacobian and also its inadequate accuracy has led to development of a category of derivative-free filters, called sigma-point Kalman filters, SPKF (Julier et al. 1995, 2000). The basic idea behind these filters is that it is easier to approximate a probability distribution compared to a nonlinear state-space model. A SPKF uses a deterministic set of quadrature points, called sigma-points, to handle the Chapman–Kolmogorov integral (Ito and Xiong 2000). This set of deterministic points can be used since a-priori distribution of the state is assumed to have a Gaussian functional form for all the time instants. Hence, it is possible to approximate it through a set of deterministic points which are parameterized through the mean and covariance of the state vector. The distribution of the state vector, a multivariate Gaussian probability distribution, at time t_{k-1} reads:

$$p(\mathbf{x}_{k-1} | \mathbf{y}_{1:k-1}) = \frac{1}{((2\pi)^n |\mathbf{P}_{k-1}|)^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{x}_{k-1} - \hat{\mathbf{x}}_{k-1})^T \mathbf{P}_{k-1}^{-1} (\mathbf{x}_{k-1} - \hat{\mathbf{x}}_{k-1})\right] \quad (2.21)$$

where $\hat{\mathbf{x}}_{k-1}$ and \mathbf{P}_{k-1} are the associated mean vector and covariance matrix of the state vector, respectively.

Once it is established that the a priori distribution of the state vector is a known Gaussian one, the Chapman–Kolmogorov integral can be recast as a Gaussian integral of the form $\int_{\mathbb{R}^n} \mu(\mathbf{x}) \omega(\mathbf{x}) d\mathbf{x}$, where $\mu(\cdot)$ is an arbitrary probability distribution, whereas $\omega(\cdot)$ denotes the a priori probability distribution of state vector. Hence (2.9) becomes (Ito and Xiong 2000):

$$\int \mu(\mathbf{x}_{k-1}) \frac{1}{((2\pi)^n |\mathbf{P}_{k-1}|)^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{x}_{k-1} - \hat{\mathbf{x}}_{k-1})^T \mathbf{P}_{k-1}^{-1} (\mathbf{x}_{k-1} - \hat{\mathbf{x}}_{k-1})\right] d\mathbf{x}_{k-1} \quad (2.22)$$

where $\omega(\cdot)$ is an arbitrary function of state vector. To numerically handle the Gaussian integral in (2.22), a discrete representation of (2.21) is necessary as done by a set of points which feature the same statistics of the original Gaussian distribution (Ito and Xiong 2000):

$$\boldsymbol{\chi}_j = \begin{cases} \sqrt{n + \rho} \mathbf{e}_j, & 1 \leq j \leq n \\ -\sqrt{n + \rho} \mathbf{e}_{j-n}, & n + 1 \leq j \leq 2n \\ \mathbf{0}, & 2n + 1 \end{cases} \quad (2.23)$$

and

$$\omega(\boldsymbol{\chi}_j) = \begin{cases} \frac{2\rho}{2(n + \rho)} & j = 2n + 1 \\ \frac{1}{2(n + \rho)} & 1 \leq j \leq 2n \end{cases} \quad (2.24)$$

where $\rho > 0$ is a constant and \mathbf{e}_k is the k th unit vector in \mathbb{R}^n . Julier and co-workers (Julier et al. 1995) proposed their S-PKF based on a quadrature rule which, for scalar functions, is identical to the Gauss-Hermit quadrature rule (Ito and Xiong 2000):

$$\int_{\mathbb{R}^n} \mu(\mathbf{x}) \omega(\mathbf{x}) d\mathbf{x} \approx \sum_{i=1}^{2n+1} \mu(\boldsymbol{\chi}_i) \omega(\boldsymbol{\chi}_i). \quad (2.25)$$

The $2n + 1$ quadrature points are the minimal number of points necessary to preserve the first and the second moments of a multivariate normal distribution (Julier et al. 1995). One can assume $\omega(\boldsymbol{\chi}_i)$ as quadrature weights, which in this case are constant in all time instants, while the quadrature points vary over time on the basis of the information contained in the covariance of the state, at $t = t_k$ the set of sigma-points are:

$$\boldsymbol{\chi}_{k,j} = \begin{cases} \hat{\mathbf{x}}_{k-1} & j = 2n + 1 \\ \hat{\mathbf{x}}_{k-1} + \psi \sqrt{\mathbf{P}_{k-1,j}} & 1 \leq j \leq n \\ \hat{\mathbf{x}}_{k-1} - \psi \sqrt{\mathbf{P}_{k-1,(j-n)}} & n + 1 \leq j \leq 2n \end{cases} \quad (2.26)$$

where $\hat{\mathbf{x}}_{k-1}$ denotes the expected value of the state and $\sqrt{\mathbf{P}_{k-1,j}}$ stands for j th column of square root of its associated covariance at $t = t_{k-1}$. This scheme outperforms the extended Kalman filter (Mariani and Ghisi 2007); for a detailed description of SPKF algorithm, see Table 2.3.

In Table 2.3, ω^{*j} and ω^j are weights adopted in the merging stage at the end of the time step, to build mean and covariance of the current state. ψ instead denotes, a time invariant scaling factor which renders possible capturing local effects of nonlinear functions. To enhance the performance, the scaling factor ψ should be

Table 2.3 Sigma-Point Kalman filter algorithm

<p>- Initialization at time t_0:</p> $\hat{\mathbf{x}}_0 = \mathbb{E}[\mathbf{x}_0]$ $\mathbf{P}_0 = \mathbb{E}\left[(\mathbf{x}_0 - \hat{\mathbf{x}}_0)(\mathbf{x}_0 - \hat{\mathbf{x}}_0)^T\right]$ <p>- At time t_k, for $k = 1, \dots, N_f$:</p> <ul style="list-style-type: none"> • Prediction stage: <ol style="list-style-type: none"> 1. Deploying sigma-points: $\boldsymbol{\chi}_{k,j}^- = \begin{cases} \hat{\mathbf{x}}_{k-1} & j = 2n + 1 \\ \hat{\mathbf{x}}_{k-1} + \psi \sqrt{\mathbf{P}_{k-1,j}} & 1 \leq j \leq n \\ \hat{\mathbf{x}}_{k-1} - \psi \sqrt{\mathbf{P}_{k-1,(j-n)}} & n + 1 \leq j \leq 2n \end{cases}$ 2. Evolution of the sigma points: $\boldsymbol{\chi}_{k,j} = \mathbf{f}_k(\boldsymbol{\chi}_{k,j}^-)$ 3. Estimation of the statistics: $\mathbf{x}_k^- = \sum_{j=1}^{2n+1} \omega^j \boldsymbol{\chi}_{k,j}$ $\mathbf{P}_k^- = \mathbf{R}_k + \mathbf{V}$ <p>where</p> $\mathbf{R}_k = \sum_{j=1}^{2n+1} \omega^{*j} (\boldsymbol{\chi}_{k,j} - \mathbf{x}_k^-) (\boldsymbol{\chi}_{k,j} - \mathbf{x}_k^-)^T$ <ul style="list-style-type: none"> • Update stage: <ol style="list-style-type: none"> 1. Calculation of Kalman gain: $\mathbf{G}_k = \mathbf{P}_k^- \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{W})^{-1}$ 2. Improve predictions using latest observation: $\hat{\mathbf{x}}_k = \mathbf{x}_k^- + \mathbf{G}_k (\mathbf{y}_k - \mathbf{H}_k \mathbf{x}_k^-)$ $\mathbf{P}_k = \mathbf{P}_k^- - \mathbf{G}_k \mathbf{H}_k \mathbf{P}_k^-$ 	<hr/>
---	-------

carefully calibrated to allow appropriate capturing of local nonlinearity effects, by tuning the distances of each sigma-point from the mean of a priori distribution of the variable. In the SPKF, the square root $\sqrt{\mathbf{P}_{k-1}}$ is calculated by a Choleski factorization. The subscript j refers to the j th column of the Choleski factor of the covariance.

The SPKF approach, similarly to the EKF, is based on the assumption that at each time instant, the a priori distribution of the state is Gaussian. To deal with a general class of nonlinear models, the particle filter approach has been developed by the academic society, the next subsection is devoted to highlight the main notions of it.

2.5.3 The Particle Filter

To deal with more general problems, it is a common practice to make recourse to Sequential Monte Carlo methods (Doucet and Johansen 2009) to handle the Chapman–Kolmogorov integral by numerical approximations. Sequential

Monte-Carlo methods make no explicit assumptions concerning the form of the posterior density $p(\mathbf{x}_{0:k}|\mathbf{y}_{1:k})$. These methods approximate the Chapman–Kolmogorov integrals in (2.9) through finite sums, adopting a sequential importance sampling on an adaptive stochastic grid. Within this frame, the particle filter implements an optimal recursive Bayesian estimation by recursively approximating the complete posterior state density. A set of N_p weighted particles $\mathbf{x}_k^{(i)}$, drawn from the posterior distribution $p(\mathbf{x}_{0:k}|\mathbf{y}_{1:k})$, is used to map the integrals. To accomplish this objective, the main trick is to represent the posterior PDF via Dirac delta functions pond at discrete sample points, namely the so-called particles. Without the loss of generality, one can write (Cadini et al. 2009):

$$p(\mathbf{x}_{0:k}|\mathbf{y}_{1:k}) = \int p(\boldsymbol{\varepsilon}_{0:k}|\mathbf{y}_{1:k}) \delta(\boldsymbol{\varepsilon}_{0:k} - \mathbf{x}_{0:k}) d\boldsymbol{\varepsilon}_k \quad (2.27)$$

where $\delta(\cdot)$ denotes the Dirac function.

Assuming the true posterior $p(\mathbf{x}_{1:k}|\mathbf{y}_{1:k})$ is known and can be sampled, an estimated of (2.27) is given by:

$$p(\mathbf{x}_{0:k}|\mathbf{y}_{1:k}) \approx \frac{1}{N_p} \sum_{i=1}^{N_p} \delta(\mathbf{x}_{0:k} - \mathbf{x}_{0:k}^i) \quad (2.28)$$

where \mathbf{x}_k^i are a set of random samples drawn from true posterior $p(\mathbf{x}_{0:k}|\mathbf{y}_{1:k})$. In practice, it is impossible to efficiently sample from the true posterior; a remedy is built by making recourse to the importance sampling, i.e. to sample state sequences from an arbitrarily chosen distribution $\pi(\mathbf{x}_{0:k}|\mathbf{y}_{1:k})$ called importance function. An unbiased estimate of $p(\mathbf{x}_{0:k}|\mathbf{y}_{1:k})$ can then still be made as (Cadini et al. 2009):

$$\begin{aligned} p(\mathbf{x}_{0:k}|\mathbf{y}_{1:k}) &= \int \pi(\boldsymbol{\varepsilon}_{0:k}|\mathbf{y}_{1:k}) \frac{p(\boldsymbol{\varepsilon}_k|\mathbf{y}_{1:k})}{\pi(\boldsymbol{\varepsilon}_{0:k}|\mathbf{y}_{1:k})} \delta(\boldsymbol{\varepsilon}_{0:k} - \mathbf{x}_{0:k}) d\boldsymbol{\varepsilon}_k \\ &\approx \frac{1}{N_s} \sum_{i=1}^{N_s} \omega_k^{*i} \delta(\mathbf{x}_{0:k} - \mathbf{x}_{0:k}^i) \end{aligned} \quad (2.29)$$

where

$$\omega_k^{*i} = \frac{p(\mathbf{y}_{1:k}|\mathbf{x}_{0:k}^i)p(\mathbf{x}_{0:k}^i)}{p(\mathbf{y}_{1:k})\pi(\mathbf{x}_{0:k}^i|\mathbf{y}_{1:k})} \quad (2.30)$$

is the importance weight associated to the state process \mathbf{x}_k^i . In practice, these weights are difficult to calculate, due to the need of evaluating the integral to normalize constant $p(\mathbf{y}_{1:k})$. Instead, the following weights are used (Gordon et al. 1993):

$$\omega_k^i = \frac{p(\mathbf{y}_{1:k}|\mathbf{x}_{0:k}^i)p(\mathbf{x}_{0:k}^i)}{\pi(\mathbf{x}_k^i|\mathbf{y}_{1:k})} \quad (2.31)$$

which are subsequently normalized according to:

$$\tilde{\omega}_k^i = \frac{\omega_k^i}{\sum_{j=1}^{N_s} \omega_k^j}. \quad (2.32)$$

Thus, estimate of the posterior distribution reads:

$$p(\mathbf{x}_{0:k} | \mathbf{y}_{1:k}) \approx \sum_{i=1}^{N_s} \tilde{\omega}_k^i \delta(\mathbf{x}_{0:k} - \mathbf{x}_{0:k}^i). \quad (2.33)$$

If the current state of the importance sampling function does not depend on future observations, i.e. if the importance sampling function satisfies the following condition (Van der Merwe 2004):

$$\begin{aligned} \pi(\mathbf{x}_{1:k} | \mathbf{y}_{1:k}) &= \pi(\mathbf{x}_1 | \mathbf{y}_1) \prod_{j=1}^k \pi(\mathbf{x}_j | \mathbf{x}_{1:j-1}, \mathbf{y}_{1:j}) \\ &= \pi(\mathbf{x}_k | \mathbf{x}_{1:k-1}, \mathbf{y}_{1:k}) \pi(\mathbf{x}_{1:k-1} | \mathbf{y}_{1:k-1}) \end{aligned} \quad (2.34)$$

and if states can be considered as a Markov process, through the assumption that the observations are conditionally independent, given the states we obtain (Van der Merwe 2004):

$$p(\mathbf{x}_{0:k}) = p(\mathbf{x}_{0:k}) \prod_{j=1}^k p(\mathbf{x}_j | \mathbf{x}_{j-1}) \quad (2.35)$$

$$p(\mathbf{y}_{1:k} | \mathbf{x}_{0:k}) = \prod_{j=1}^k p(\mathbf{y}_j | \mathbf{x}_j) \quad (2.36)$$

thus by using Eqs. (2.34–2.36) in (2.30), the recursive formula for the update of importance weights becomes (Van der Merwe 2004):

$$\omega_k^i = \omega_{k-1}^i \frac{p(\mathbf{y}_k | \mathbf{x}_k^i) p(\mathbf{x}_k^i | \mathbf{x}_{k-1}^i)}{\pi(\mathbf{x}_k^i | \mathbf{x}_{0:k-1}^i, \mathbf{y}_{1:k})}. \quad (2.37)$$

For filtering purposes, the estimation of the marginal probability density $p(\mathbf{x}_k | \mathbf{y}_k)$ of the full posterior is sufficient, that is, if $\pi(\mathbf{x}_k | \mathbf{x}_{1:k-1}, \mathbf{y}_{1:k})$ is substituted by $\pi(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{y}_k)$, the sampling proposal will only depend on \mathbf{x}_{k-1} and \mathbf{y}_k (Arulampalam et al. 2002). Consequently, the recursive formula for estimation and update of the non-normalized weights is expressed as (Arulampalam et al. 2002):

$$\omega_k^i = \omega_{k-1}^i \frac{p(\mathbf{y}_k | \mathbf{x}_k^i) p(\mathbf{x}_k^i | \mathbf{x}_{k-1}^i)}{\pi(\mathbf{x}_k^i | \mathbf{x}_{k-1}^i, \mathbf{y}_k)}. \quad (2.38)$$

The (2.38) provides a method to sequentially update the importance weights, given an appropriate choice of the proposal distribution $\pi(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{y}_k)$. Consequently, any expectations of the form $E[g(\mathbf{x}_k)] = \int g(\mathbf{x}_k) p(\mathbf{x}_{0:k} | \mathbf{y}_{1:k}) d\mathbf{x}_k$, $g(\cdot)$ being any given function can be approximated through $E[g(\mathbf{x}_k)] \approx \sum_{j=1}^{N_p} \omega_k^j g(\mathbf{x}_k^j)$.

In (Doucet 1997), it was shown that the proposal distribution $\pi(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{y}_k)$ minimizes the variance of the importance weights, conditional on \mathbf{x}_{k-1} and \mathbf{y}_k . Nonetheless, the distribution $p(\mathbf{x}_j|\mathbf{x}_{j-1})$ (i.e. the transition prior) is the most popular choice for the proposal distribution. Although it results in a Monte-Carlo variation higher than that obtained using the optimal proposal $\pi(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{y}_k)$, the importance weights are easily updated by simply evaluating the observation likelihood density $\pi(\mathbf{x}_k|\mathbf{x}_{k-1})$ for the sampled particle set, through (Cadini et al. 2009):

$$\omega_k^i = \omega_{k-1}^i p(\mathbf{y}_k|\mathbf{x}_k^i). \quad (2.39)$$

The variance of these importance weights increases stochastically over time (Doucet 1997); after a few time steps, one of the normalized importance weights tends to one, while the remaining weights tend to zero. To address this rapid degeneracy, a resampling stage may be used to eliminate samples with low importance weights, and duplicate samples with high importance weights. An intuitive explanation of particle filtering technique is expressed as: each sample \mathbf{x}_k^i might be a solution of the problem, and its associated weight ω_k^i signifies its probability of being the correct solution. In the resampling stage, the particles with higher probability are duplicated and in turn, the ones with lower probability are discarded. Such an approach somehow permits the filter to condense the cloud of particles around the peak probability zone. An algorithm built in this method was primarily proposed by Gordon et al. (1993), and has been called in different names such as bootstrap filter, condensation algorithm etc.; for a detailed algorithmic specification see Table 2.4.

It is worth underlining that the update stage in the particle filter algorithm is conducted via evolution of particle weights based on the probability of occurrence of each particle conditioned on the latest observation. After such weight evolution, the resampling stage is prescribed to alleviate the degeneracy issue, where ensemble of the samples is refined to increase the population of the samples which are more likely and decrease the lower probability population. To this end, different algorithms were proposed in the literature, namely e.g. stratified, systematic, or residual resampling. Accounting for sampling quality and computational complexity, the systematic resampling scheme adopted turns out to be the most favorable one in this study (Hol et al. 2006). The resampling stage is performed by drawing a random sample ζ_j from the uniform distribution over (0,1]; afterwards, the M th particle for which the value of the random number ζ_j is between values of the empirical cumulative distribution of particles at $M - 1$ and M is duplicated by resampling stage. Details of the systematic resampling (Kitagawa 1996) algorithm are shown in Table 2.5.

Since particle filter handles the current, the actual PDF of the state to draw particles in prediction stage, it can appropriately account for non-Gaussian densities. However, as the dimension of the state vector increases, computational costs associated with numerical integrations increase drastically. It is suggested, as a rough rule of thumb, not to apply particle filter to problems with dimension of state vector more than five (Li et al. 2004).

Table 2.4 Particle filter algorithm

- Initialization at time t_0 :

$$\hat{\mathbf{x}}_0 = \mathbb{E}[\mathbf{x}_0] \qquad \mathbf{x}_0^{(i)} = \hat{\mathbf{x}}_0$$

$$\mathbf{P}_0 = \mathbb{E}\left[(\mathbf{x}_0 - \hat{\mathbf{x}}_0)(\mathbf{x}_0 - \hat{\mathbf{x}}_0)^\top\right] \qquad \omega_0^{(i)} = p(\mathbf{y}_0|\mathbf{x}_0) \quad i = 1, \dots, N_P$$

- At time t_k , for $k = 1, \dots, N_T$:

- Prediction stage:
 1. Draw particles:

$$\mathbf{x}_k^{(i)} \sim p(\mathbf{x}_k|\mathbf{x}_{k-1}^{(i)}) \quad i = 1, \dots, N_P$$
- Update stage:
 1. Evolve weights:

$$\omega_k^{(i)} = \omega_{k-1}^{(i)} p(\mathbf{y}_k|\mathbf{x}_k^{(i)}) \quad i = 1, \dots, N_P$$
 2. Resampling, see Table 2.5.
 3. Compute expected value:

$$\hat{\mathbf{x}}_k = \sum_{i=1}^{N_P} \omega_k^{(i)} \mathbf{x}_k^{(i)}$$

Table 2.5 Systematic resampling algorithm

- At time t_k , for $j = 1, \dots, N_P$:

- draw a random sample ζ_j from uniform distribution over $(0,1]$
- find M that satisfies:

$$\sum_{i=1}^{M-1} \omega_k^{(i)} < \zeta_j \leq \sum_{i=1}^M \omega_k^{(i)}$$
- $\mathbf{x}_k^{(j)} \leftarrow \mathbf{x}_k^{(M)}$

The sampling distribution used in the generic particle filter can cause serious problems, since it is not the optimal one and conditioned on the latest observation. This fact leads to high computational costs, since the cloud of the samples fall far from the zones with high probability; therefore, many samples have to be drawn in order to make the algorithm to converge.

2.5.4 The Hybrid Extended Kalman Particle Filter

To alleviate the issues discussed in the previous subsection, our remedy is to keep using the same sampling distribution; however, after the samples are drawn, we improve the quality of the ensemble of the samples. Roughly speaking, once the samples are drawn, they are pushed by an extended Kalman filter toward the zones of higher probability in order to incorporate data from the latest observations into each sample.

The reason for exploiting the EKF instead of the SPKF, for enhancing the quality of sample ensemble, is twofold: first, the difficulty in tuning it in a way to have all the particles moved appropriately; second, the computational cost of the SPKF combined with particle filter can be significant, since both adopt numerical

Table 2.6 Hybrid extended Kalman particle filter algorithm

- Initialization at time t_0 :

$$\hat{\mathbf{x}}_0 = \mathbb{E}[\mathbf{x}_0] \quad \mathbf{x}_0^{(i)} = \hat{\mathbf{x}}_0$$

$$\mathbf{P}_0 = \mathbb{E}\left[(\mathbf{x}_0 - \hat{\mathbf{x}}_0)(\mathbf{x}_0 - \hat{\mathbf{x}}_0)^T\right] \quad \omega_0^{(i)} = p(\mathbf{y}_0|\mathbf{x}_0) \quad i = 1, \dots, N_P$$

- At time t_k , for $k = 1, \dots, N_T$:

• Prediction stage:

1. Draw particles:

$$\mathbf{x}_k^{(i)} \sim p\left(\mathbf{x}_k|\mathbf{x}_{k-1}^{(i)}\right) \quad i = 1, \dots, N_P$$

2. Push the particles toward the region of high probability through an EKF:

$$\mathbf{P}_k^{(i)-} = \mathbf{F}_k \mathbf{P}_{k-1}^{(i)} \mathbf{F}_k^T + \mathbf{V}$$

$$\mathbf{G}_k^{(i)} = \mathbf{P}_k^{(i)-} \mathbf{H}_k^T \left(\mathbf{H}_k \mathbf{P}_k^{(i)-} \mathbf{H}_k^T + \mathbf{W} \right)^{-1}$$

$$\mathbf{x}_k^{(i)} = \mathbf{x}_k^{(i)-} + \mathbf{G}_k^{(i)} \left(\mathbf{y}_k - \mathbf{H}_k \mathbf{x}_k^{(i)-} \right)$$

$$\mathbf{P}_k^{(i)} = \mathbf{P}_k^{(i)-} - \mathbf{G}_k^{(i)} \mathbf{H}_k \mathbf{P}_k^{(i)-} \quad i = 1, \dots, N_P$$

• Update stage:

1. Evolve weights:

$$\omega_k^{(i)} = \omega_{k-1}^{(i)} p\left(\mathbf{y}_k|\mathbf{x}_k^{(i)}\right) \quad i = 1, \dots, N_P$$

2. Resampling, see Table 2.5.

3. Compute expected value or other required statistics:

$$\hat{\mathbf{x}}_k = \sum_{i=1}^{N_P} \omega_k^{(i)} \mathbf{x}_k^{(i)}$$

approximations to handle the quadrature. That is, the EKF is combined with particle filter frames to update each particle based on the information contained in the latest observation, see Table 2.6.

In Table 2.6, \mathbf{F}_k represents the current Jacobians of mappings $f_k(\blacksquare)$.

In what follows, we will assess the performance of the filters through numerical examples. In the absence of experimental data, for validation of the algorithms, we rely on pseudo experimental data, i.e. numerical data resulting from direct analysis contaminated by white Gaussian processes substitute noisy measurements of the observable part of the state vector.

2.6 Numerical Results for Dual Estimation of Single Degree and Multi Degrees of Freedom Dynamic Systems

To numerically solve the set of ordinary differential equations that govern the dynamics of the system, a Newmark explicit integration scheme has been adopted. According to (Hughes 2000), the time marching algorithm within the time step $[t_{k-1} t_k]$ can be partitioned into:

- predictor stage:

$$\tilde{\mathbf{u}}_k = \mathbf{u}_{k-1} + \Delta t \dot{\mathbf{u}}_{k-1} + \Delta t^2 \left(\frac{1}{2} - \beta\right) \ddot{\mathbf{u}}_{k-1} \quad (2.40)$$

$$\dot{\tilde{\mathbf{u}}}_k = \dot{\mathbf{u}}_{k-1} + \Delta t(1 - \gamma) \ddot{\mathbf{u}}_{k-1}; \quad (2.41)$$

- explicit integrator:

$$\ddot{\mathbf{u}}_k = \mathbf{M}^{-1}(\mathbf{R}_k - (\mathbf{D} \dot{\tilde{\mathbf{u}}}_k + \mathbf{K} \tilde{\mathbf{u}}_k)); \quad (2.42)$$

- corrector stage:

$$\mathbf{u}_k = \tilde{\mathbf{u}}_k + \Delta t^2 \beta \ddot{\mathbf{u}}_k \quad (2.43)$$

$$\dot{\mathbf{u}}_k = \dot{\tilde{\mathbf{u}}}_k + \Delta t \gamma \ddot{\mathbf{u}}_k \quad (2.44)$$

where $\Delta t = t_k - t_{k-1}$ denotes the time step size. To ensure numerical stability in the linear regime, Δt needs to be upper bounded by Bathe (1996):

$$\Delta t_{cr} = \frac{T_n}{\pi} \quad (2.45)$$

where T_n is the period associated with the highest oscillation frequency. Even if Δt_{cr} can be increased in the reduced model, since higher order oscillations are filtered out of the numerical solution, in what follows we are keeping Δt constant in all the simulations. Hence, the speedup reported is therefore to be mainly linked to the reduction of the number of handled DOFs.

In Corigliano and Mariani (2001b), it was shown that structural effects may play a prominent role in system identification. They typically lead to shadowing effects, arising when the sensitivity of measurable variables (like, e.g. displacements or velocities) to constitutive parameters becomes negligible or falls out of the measurement range (i.e. they become comparable to round-off errors). Such structural effects practically lead to multiple solutions of the inverse problem in terms of model parameters update (all difficult to distinguish in the noisy environment), and filters provide biased or divergent calibrations, see e.g. (Corigliano and Mariani 2001a, b, 2004). To solely benchmark performance of the filters, we primarily focus on dynamics of a single degree-of-freedom structure. Once the performances of the filters are benchmarked by analyses concerning a single degree-of-freedom, then we move to the multi degrees of freedom structures to study the applicability of these methods in more realistic scenarios.

2.6.1 Single Degree-of-Freedom Dynamic System

Since we are interested in benchmarking the extended Kalman particle filter (EK-PF) when compared to other Bayesian filters tested in this study (i.e. the EKF, the SPKF and the PF), the aforementioned structural effects are avoided by focusing on an undamped single DOF system constituted by a mass (or rigid block) connected to the reference frame through a spring, see Fig. 2.1. The equation of motion of the system is expressed as:

$$M\ddot{u} + R(u) = F(t) \quad (2.46)$$

where m is the block mass; $R(u)$ is the spring force; $F(t)$ is the external load, which evolves in time; u and \ddot{u} are the displacement and acceleration of the block, respectively. The results can be easily extended to the damped case; in such situation, it is, however, important to have the system continuously to be (or permanently) excited, so as to avoid vibration amplitudes to progressively decrease in time, thereby loosing filter efficiency, see (Corigliano and Mariani 2004).

In this study, all the studied filters perform well for dual estimation of a linear SDOF structure; hence, the results are not discussed for the sake of brevity. Instead, to assess the filter performance, r is assumed to be a highly nonlinear, RFS-type function of the displacement u , i.e. of the spring elongation (Rose et al. 1981; Corigliano et al. 2006):

$$R(u) = a u \exp[-n u] \quad (2.47)$$

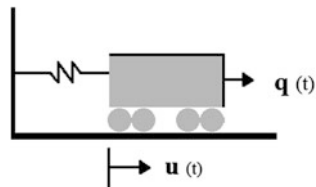
where a and n are unknown model parameters in need of tuning. Even if inspired by tight binding studies in atomistic simulations, law (2.47) is to be considered as phenomenological description of damaging processes taking place inside the spring: once a peak reaction is attained, softening (i.e. strength degradation) sets in and drives the state toward a smooth failure, occurring when $u \rightarrow +\infty$. Therefore, the two parameters a and n in (2.47) can be related to the strength r^M and the toughness G of the spring, through:

$$\begin{aligned} R^M &= \frac{a}{en} \\ G &= \int_0^{\infty} R du = \frac{a}{n^2} \end{aligned} \quad (2.48)$$

where e is the Nepero number.

Law (2.47) can be handled as a tensile envelope, with damage activation/deactivation conditions to be adopted to properly describe unloading/reloading paths, see e.g. (Mariani and Ghisi 2007). In accordance with previous papers (Mariani 2009a, b), we instead assume here that damage evolution is captured by strength degradation only, and model (2.45) is managed as a holonomic (nonlinear elastic) law.

Fig. 2.1 Single degree-of-freedom structural system



As aforementioned, we focus on pseudo-experimental (numerical) tests only. They consist in running direct analyses with known (target) values of model parameters, and then adding a white noise of assigned variance to the system output. This procedure allows us to obtain scattered measurements, which are then used to feed the filters.

In order to handle a stable system dynamics, followed by divergence (i.e. by $u \rightarrow +\infty$) due to the inception and growth of damage in the spring, the applied load $F(t)$ (see Eq. 2.47) has been assumed to monotonically increase in time according to:

$$F(t) = 0.5 + 0.0075t(\text{N}) \quad (2.49)$$

see also (Corigliano and Mariani 2004). With the mass initially at rest, this loading condition allows the system to be stable up to $t \cong 150$ s; beyond this threshold, softening in the spring becomes dominant (i.e. the transmitted force gets vanishing), and displacement u diverges.

In the analyses, the mass has been assumed $m = 9.72 \text{ N s}^2/\text{mm}$, see also (Corigliano and Mariani 2004). Measurements consist of the current mass displacement only, featuring a noise level characterized by a standard deviation $w = 0.05 \text{ mm}$.

The results relevant to the tracking of the whole system state (i.e. u , \dot{u} and \ddot{u}) are reported in Fig. 2.2, as obtained by running the EK-PF and, for comparison purposes, the PF and the S-PKF. In these plots, the dashed lines represent the target system response; the orange squared symbols are instead the discrete-time estimations furnished by the filters, and the blue circular symbols stand for the measurements (that are displacement values only). The figure illustrates that the three filters are all capable to track the initial, stable oscillations and the transition to the unstable regime due to inception of softening. Even if a high number of particles (500 in this analysis) has been adopted, the PF is not able to attain the same accuracy of the S-PKF; the EK-PF (run using 5 particles) is instead very accurate, performing slightly better than the S-PKF.

We now move to the system identification task. As usual [see, e.g. (Ljung 1999)], the following results have been obtained by setting the pivotal entries of \mathbf{P}_0 relevant to model parameters to be (at least) two orders of magnitude larger than those relevant to state variables. By this means, model calibration is enhanced, since information (actually, innovation) brought by measurements is trusted much more than the current estimates.

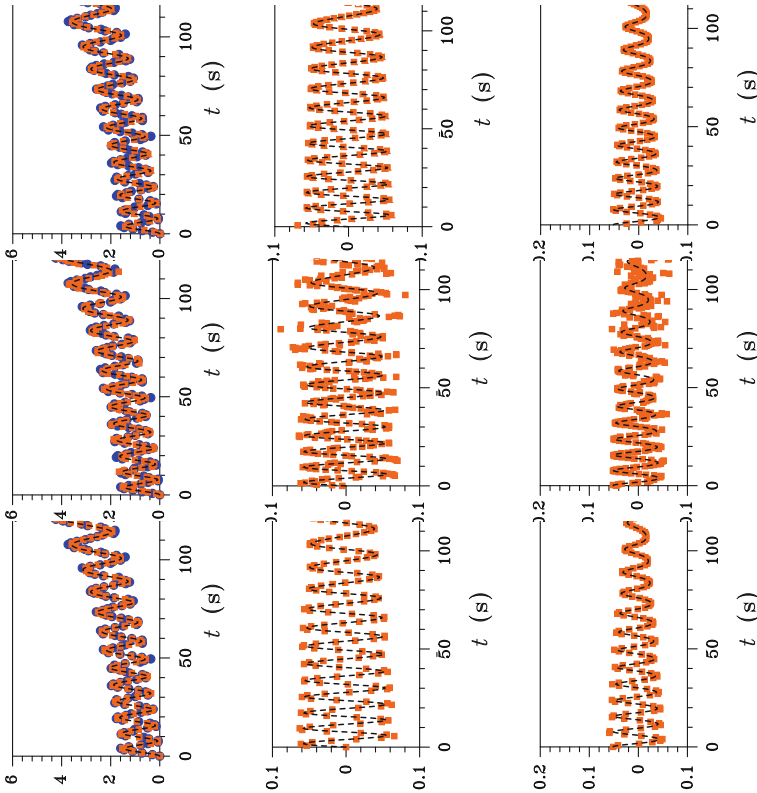


Fig. 2.2 State tracking. Comparison between target (*dashed lines*) and tracked (*orange squared symbols*) system evolution, in terms of: (*left column*) displacement u ; (*central column*) velocity \dot{u} ; (*right column*) acceleration \ddot{u} . Results obtained by running: (*top row*) EK-PF; (*middle row*) PF, and (*bottom row*) S-PKF

In terms of time evolution of the estimates of model parameters a and n , it is shown in Fig. 2.3 that they rapidly converge to the target values in the stable dynamic regime, independently of the initialization guess (here in the range between 50 and 150 % of the target values). The SPKF and the PF perform better than the EK-PF in the short-term time interval, featuring higher convergence rates without excessive oscillations of the estimates. However, as soon as the system stability threshold is approached, wild oscillations of increasing amplitude set in which leads to diverging model calibration furnished by SPKF and PF. On the contrary, the EK-PF does not show such wild oscillations, and always provides stable, unbiased estimates.

To obtain insights into the superior performance of the EK-PF, Figs. 2.4 and 2.5 report the projections onto the two model parameter axes of the time evolution of the (smoothed) distribution of particles deployed by PF and EK-PF, respectively. It can be seen that step #2 of prediction stage of the Table 2.6 proves to be very

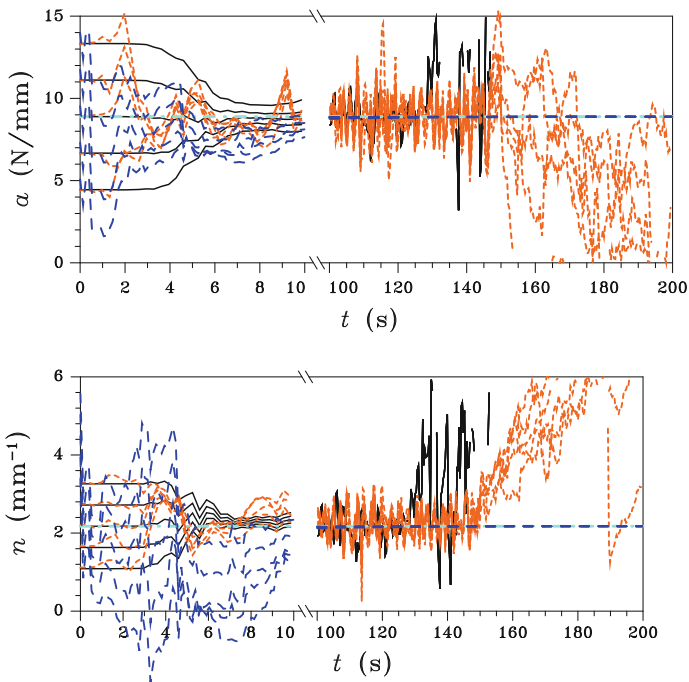


Fig. 2.3 Model calibration. Time evolution of estimated model parameters (*top row*) a and (*bottom row*) n , at varying initialization values. Results obtained by running: EK-PF (*long-dashed blue lines*), PF (*dashed orange lines*) and S-PKF (*continuous black lines*)

efficient in moving the particles toward the region of major interest, with distributions that are not spread over an extensive range of values. This eventually assists us to avoid divergence of the estimates.

Next, we study the performance of Bayesian filters for a slightly more difficult task: the dual estimation of a system having a bilinear constitutive model for its spring. The system is the same as before, but now the relationship between the force in spring R and the displacement u reads:

$$R = \begin{cases} k_1 u & \text{if } u < u_M \\ k_1 u_M + k_2(u - u_M) & \text{if } u > u_M \end{cases} \quad (2.50)$$

where k_1 denotes initial slope of the constitutive model of the spring; u_M is the limit at which spring constitutive model starts its bilinear behavior; and k_2 denotes the gradient of force–displacement after the displacement has exceeded u_M .

The strength of the constitutive law (2.50) lies in the versatility in simulating three different material behaviors, namely the linear-hardening, linear-perfect plastic and linear-softening. Under monotonically increasing loadings, depending on the k_2 value this bilinear constitutive law can be adopted to deal with identification of parameters of a structure whose behavior may not be known a priori.

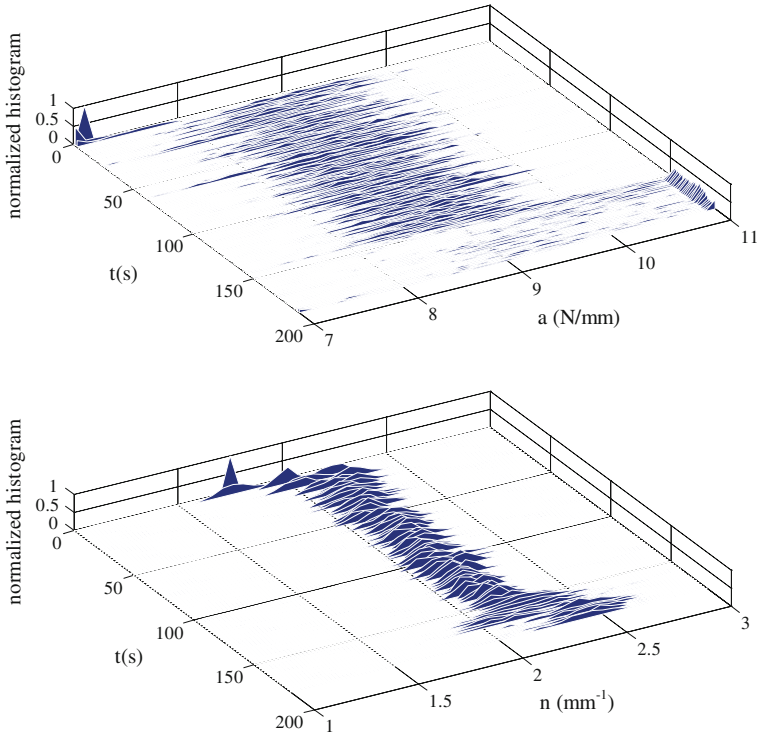


Fig. 2.4 PF, projections onto the parameters (*top*) a and (*bottom*) n axes of the evolution of particles

While dealing with joint state and parameter estimation, the main drawback of such constitutive law is the intricate interrelation of components of the state vector, when the parameter of the constitutive model are included into the state vector. Consider the state-space representation of the system, augmented state vector incorporates k_1 , k_2 and u_M so as:

$$\vartheta = \begin{bmatrix} k_1 \\ k_2 \\ u_M \end{bmatrix}. \quad (2.51)$$

At each time iteration, the evolution equation, based on the value of u_M may find two different functional form: if displacement of the spring is less than u_M , only the initial linear behavior of the spring gets involved; if displacement of the spring exceeds u_M , nonlinearity of spring affects the spring force. Thus filter has to decide which path to follow as long as deterministic information is not available for u_M . In what follows, the results of application of nonlinear versions of Kalman filters and Particle filter and also a hybrid extended Kalman particle will be presented. The results are organized in three sets, each one of the filtering algorithms

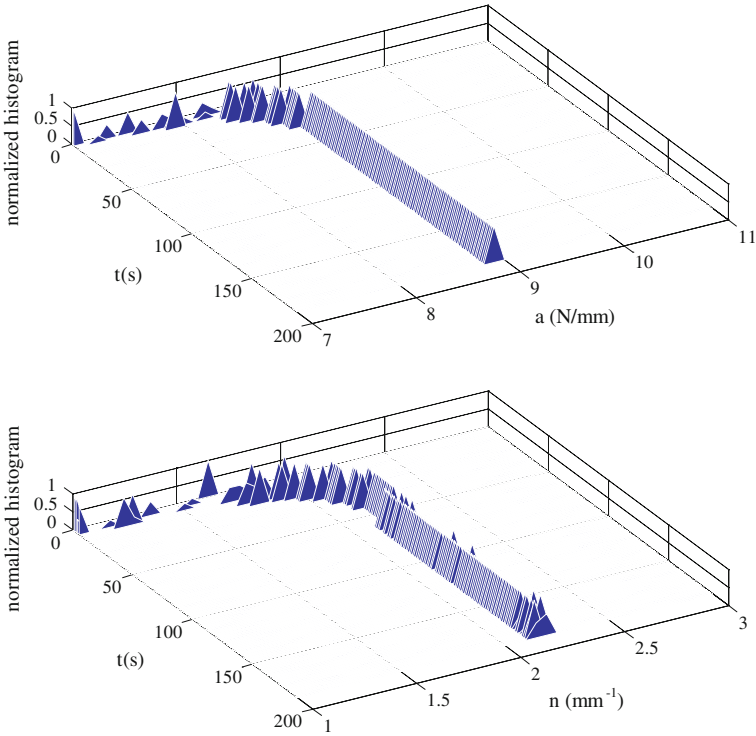


Fig. 2.5 EK-PF, projections onto the parameters (*top*) a and (*bottom*) n axes of the evolution of particles

is assessed when dealing with reference problems of each scenario: linear hardening, linear-perfectly plastic and linear-softening constitutive laws.

As aforementioned, in all the analyses, pseudo-experimental data are used instead of data coming from experiments; the numerical data contaminated by a zero mean additive white noise are therefore taken as observations of the system. The initial slope k_1 is always assumed to be 3.27 N/nm, while $k_2 = k_1/10$ for hardening, $k_2 = 0$ for plasticity and $k_2 = -k_1$ to mimic softening behavior. The value of the threshold of linear behavior u_M is set to 0.46 mm; the mass has been assumed $m = 9.72$ Ns²/mm, see also (Corigliano and Mariani 2004; Eftekhar Azam et al. 2012a). Measurements consist of the current mass displacement only, featuring a noise level characterized by a standard deviation $w = 0.01$ mm. In order to incept a nonlinear behavior due to damage in the spring, the applied load q has been assumed to monotonically increase in time according to (2.49). Since the main objective of this study is the calibration of constitutive parameters, we just include the plots of parameter estimation unless there is a specific reason to present state estimate plots.

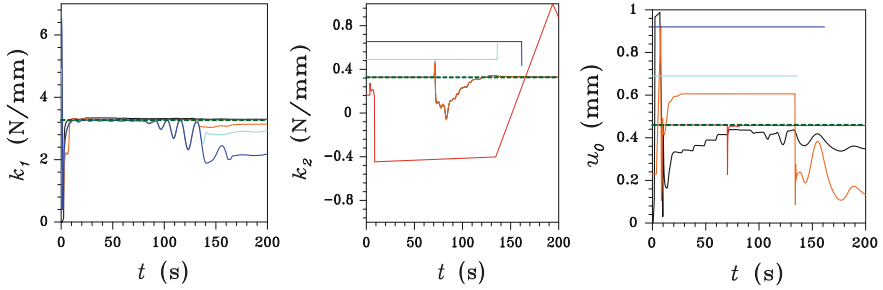


Fig. 2.6 Results of EKF for estimation of parameters of linear-hardening constitutive law

Figures 2.6, 2.7 and 2.8 show the performance of the EKF in simultaneous calibrating the three constitutive parameters of linear hardening, linear plastic and linear softening case, respectively. The filter is run for different initialization values; it is seen that except for the initializations from target values, in none of the scenarios the EKF is able to identify the constitutive parameters. As aforementioned, the EKF is a straight-forward extension of the Kalman filter, based on linearization of the evolution equation. It is suitably adopted for weakly nonlinear problems; however, if the nonlinearity is severe, such linearization is not accurate enough and poor performance is expected. It has to be underlined that tuning of the filter, in order to obtain unbiased estimate of parameters is not always easy, and we do not claim that we have tuned optimally the filters for different initializations and constitutive laws. In essence, three noise covariances associated with each parameter are tuning knobs of the system (Bittanti and Savaresi 2000). One has to notice that as the number of the parameters increases, their simultaneous tuning might become more difficult and algorithm appears to be practically inefficient.

Next, the results relevant to the performance of the SPKF are presented; even though SPKF has proved to outperform the EKF in many cases, it suffers from problem of positive definiteness of covariance matrix when dealing with parameter identification (Holmes et al. 2008), and also the tuning of the scale factor might become critical (Mariani 2009b). Figures 2.9, 2.10 and 2.11 present the results obtained by the SPKF when dealing with the three different scenarios of constitutive laws. Similar to the previous case, the filter is run with different initializations to see whether convergence is triggered from different starting points. It is seen that the performance of SPKF is quite poor, as it is not able to furnish unbiased estimates of the parameters, except for the case that the initial guess are set at the target values of parameters. We remind that in excess of three fictitious noise covariance to be tuned, within the SPKF algorithm also the scale factor should be tuned accurately; moreover, such a factor is used to allow the filter to capture local effects of nonlinearities of the evolution equation. Adding this to the three former parameters, one can see how delicate the task of tuning can become.

Since common extensions of the KF cannot furnish unbiased estimates of constitutive parameters, we make recourse to Particle filters as they are basically

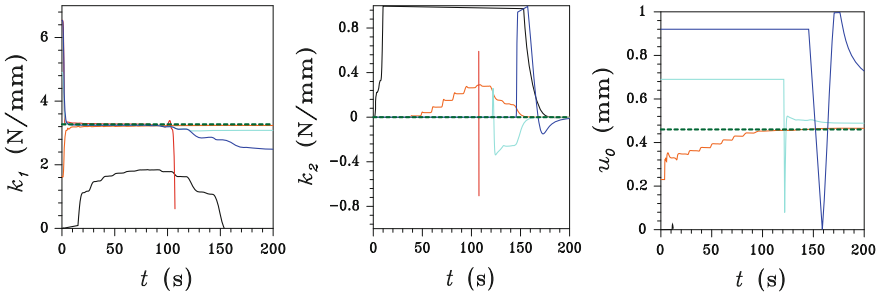


Fig. 2.7 Results of EKF for estimation of parameters of linear-plastic constitutive law

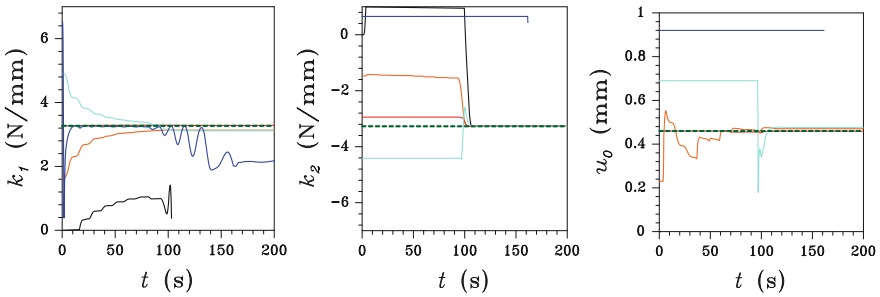


Fig. 2.8 Results of EKF for estimation of parameters of linear-softening constitutive law

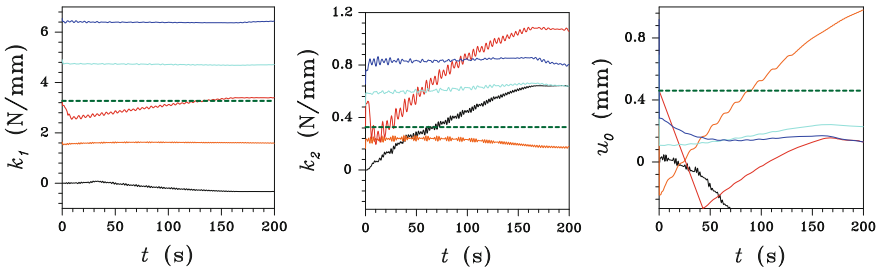


Fig. 2.9 Results of SPKF for estimation of parameters of linear-hardening constitutive law

designed for nonlinear systems with arbitrary uncertainty associated with them. Figures 2.12, 2.13 and 2.14 show the results of estimation of the parameters of linear-hardening, linear-perfect plastic and linear-softening constitutive model. Even though the particle filter is devised for nonlinear/non-Gaussian systems, it is seen through the graphs that it fails to estimate the parameters appropriately.

In designing a PF, it should be noticed that an appropriate initial guess of the distribution of the state of the system is essential to enhance the performance of the filter. Nevertheless, the value of the covariance of the noise for calibrating the

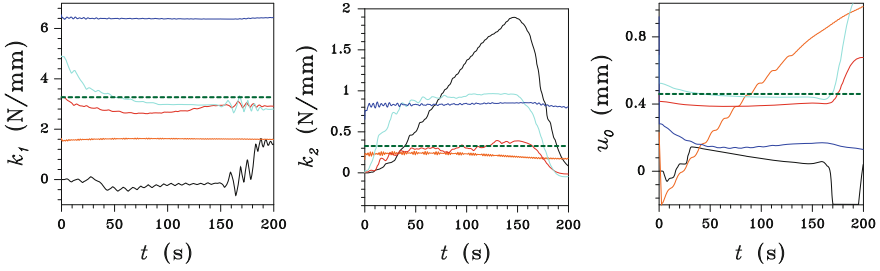


Fig. 2.10 Results of SPKF for estimation of parameters of linear-plastic constitutive law

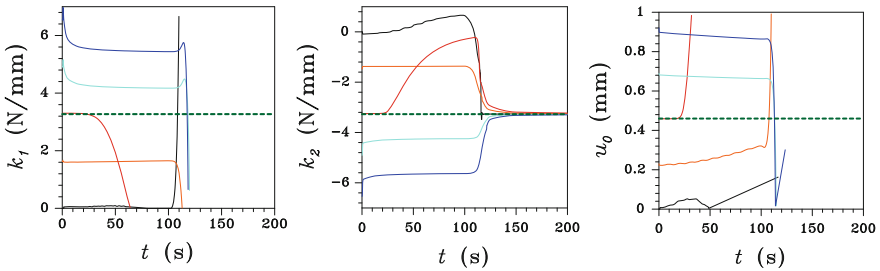


Fig. 2.11 Results of SPKF for estimation of parameters of linear-softening constitutive law

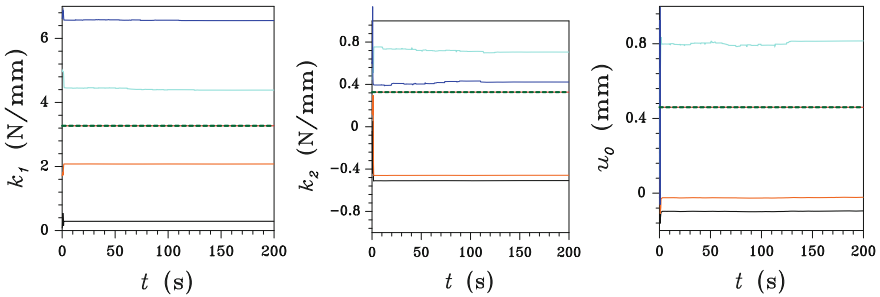


Fig. 2.12 Results of PF for estimation of parameters of linear-hardening constitutive law

parameters plays an important role (Arulampalam et al. 2002); moreover, they should be appropriately adjusted in order to let scattering of the samples in the feasible range of the parameter. We illustrate these issues via numerical examples. For ease of tuning, it is primarily assumed that we have quite reasonable a priori knowledge of k_1 and u_0 and aim to estimate only k_2 . Figures 2.15, 2.16, 2.17, 2.18, 2.19 and 2.20 show the results of analysis for estimation of k_2 . Looking at Figs. 2.15 and 2.18, they plot the time histories of estimation of the parameter k_2 , supposing that the values of k_1 and u_M are a priori known. Moving from Figs. 2.15, 2.16, 2.17 and 2.18, we have changed the intensity if the tuning noise to highlight its

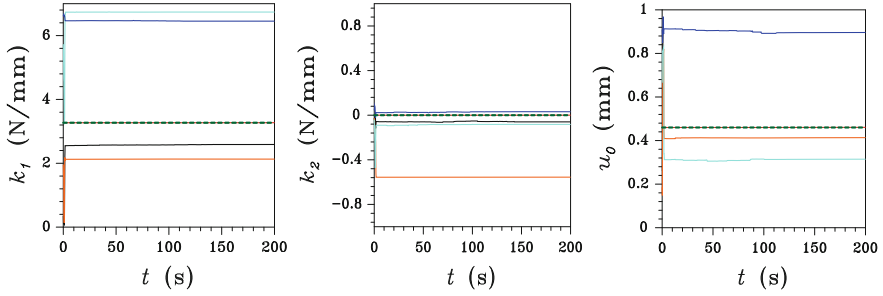


Fig. 2.13 Results of PF for estimation of parameters of linear-plastic constitutive law

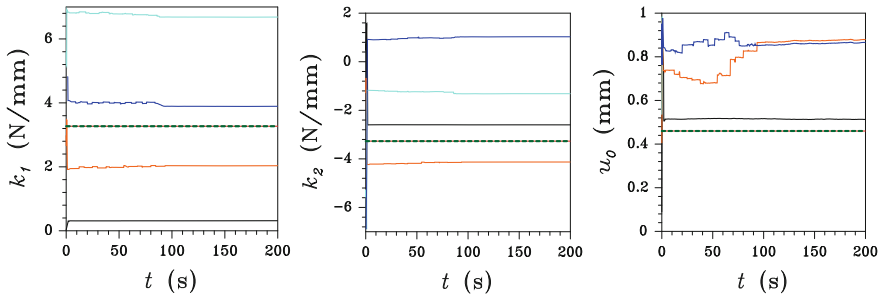


Fig. 2.14 Results of PF for estimation of parameters of linear-softening constitutive law

importance in the parameter estimation. In both cases the initial value of the parameter is set to 50 % of the target value. In the graph shown in Fig. 2.15, the value of the noise for tuning k_2 is set to $10^{-2} \text{ N}^2/\text{mm}^2$, which permit the evolution of the particles finally converge to the target value. On the contrary, the noise value equal to $10^{-4} \text{ N}^2/\text{mm}^2$ which is used to obtain the results shown in Figs. 2.18, 2.19 and 2.20, does not let the algorithm to sample efficiently, and the ensemble of the particles does not finally converge to the target values of the parameters.

To compare the performance of the particle filter when the tuning noise intensity varies, one can confront Figs. 2.16 and 2.19. At $t = 100$ s, as the parameter k_2 enters in the system evolution due to the inception of nonlinearity, for the case with the noise equal to $10^{-4} \text{ N}^2/\text{mm}^2$, estimates of the states of the system diverge, while in with the noise equal to $10^{-2} \text{ N}^2/\text{mm}^2$ states are estimated unbiasedly. This corroborates the idea that a small value for tuning noise intensity prevents the cloud of the particles to efficiently approximate the a posteriori distribution of the state. To investigate this issue in more details, we have focused on the histograms of the particles and their associated weights at $t = 130$ s, where there is a sharp change in the estimation of displacements (see Fig. 2.17). Looking at the histograms and particle weights shown in Fig. 2.20, it is seen that the cloud of the particles, shown via histogram, are far from the observation vicinity (the red

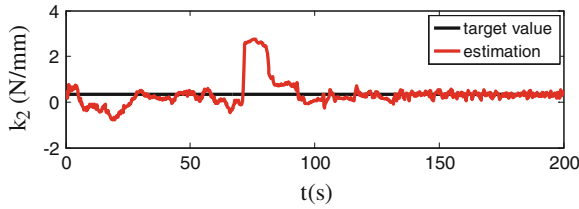


Fig. 2.15 Parameter estimates while noise covariance is set appropriately ($10^{-2} \text{N}^2/\text{mm}^2$)

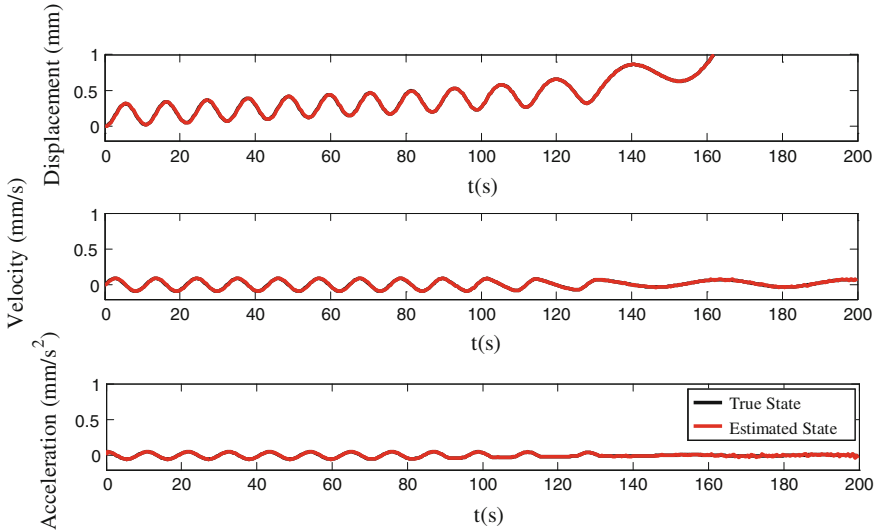


Fig. 2.16 State estimates when noise covariance is set appropriately ($10^{-2} \text{N}^2/\text{mm}^2$)

vertical line), where the distance of the closest bin to the observation is about 0.15 mm. As a consequence, in Fig. 2.20 all of the particles have found equal normalized weights; their distance from the observation vicinity is too far, as a consequence the associated probability with each particle becomes less than the round-off errors. On the contrary, looking at the same time instant in the case in which estimates are converging target values, it is seen that the distance of the closest been to the observation is about 0.004 mm; thus, in Fig. 2.18 the particles closer to observation have found a more significant normalized weight whereas other have smaller weights. Such diversity of weights shows that the particles are distributed in a zone which is close to the observation.

In what precedes, it has been shown that the proper choice of noise covariance has fundamental effects on the performance of PF. In case of dealing with one single parameter, it is not difficult to tune the filter; however, while dealing with more parameters, finding the right combination might become difficult. To address the

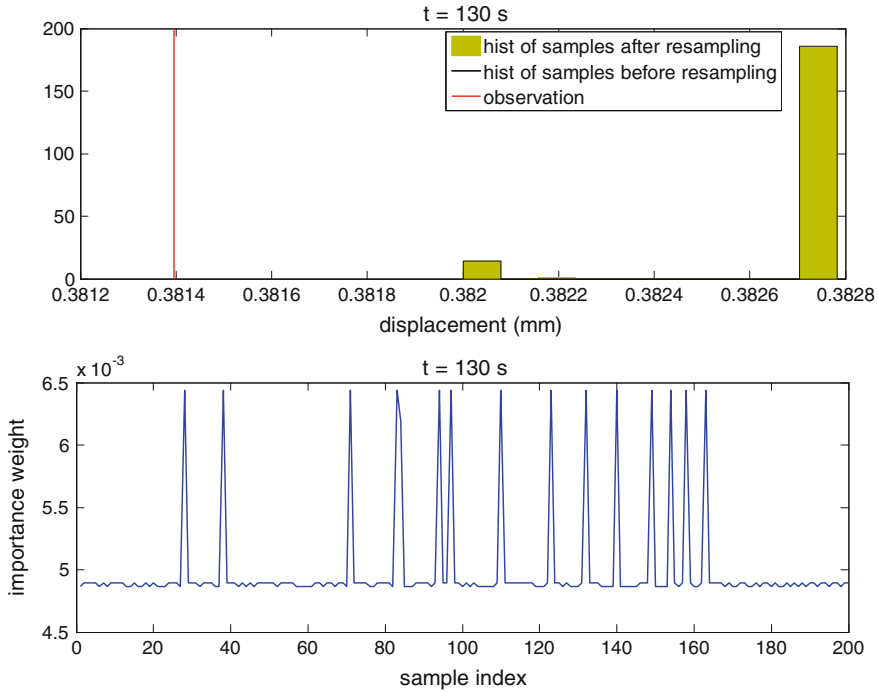


Fig. 2.17 Histogram of observable part of state vector (*top*) and associated sample weights (*bottom*) though through the *top figure* it seems that the sample has degenerated, through the *bottom* it is seen that many samples have significant weights. Also notice that samples are distributed in a close neighborhood of observation (*red vertical line*)

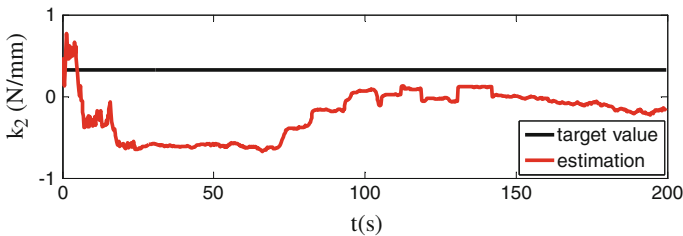


Fig. 2.18 Parameter estimates when noise covariance is not set appropriately ($10^{-4} \text{ N}^2/\text{mm}^2$)

issues induced by simultaneous track of the three parameters shown in Figs. 2.11, 2.12, 2.13 and 2.14, for instance the step-function like behavior seen in Fig. 2.14 when calibrating u_0 , we focus on the state estimation time histories, see Fig. 2.21, and consider the jump at $t = 34$ s. To have a closer look at what happens while this jump occurs, once again we make use of histogram of the distribution of the particles in two time instants: the beginning of the time step; the end of the time step. Before

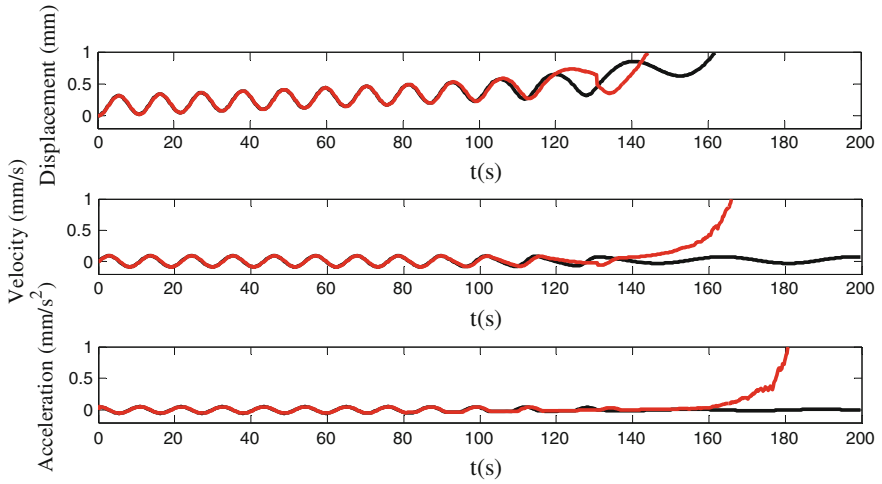


Fig. 2.19 State estimates when noise covariance is not set appropriately ($10^{-4} \text{ N}^2/\text{mm}^2$)

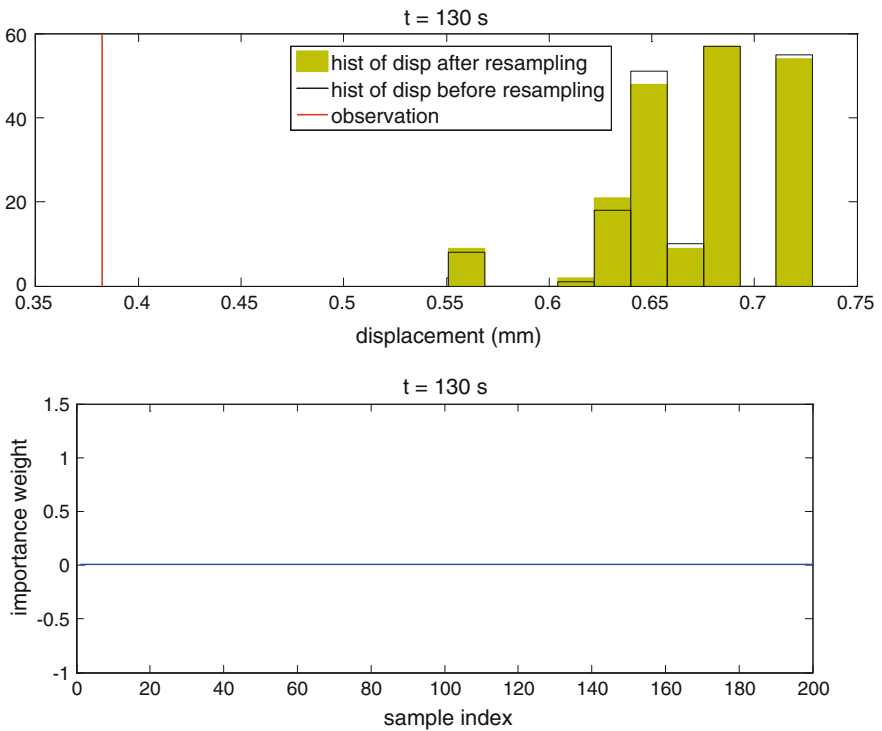


Fig. 2.20 Histogram of observable part of state vector (*top*) and associated samples weights (*bottom*) though from *top* it is seen that the sample cloud is quite far from observation neighborhood (*vertical red line*) consequently none of the particles find significant weights

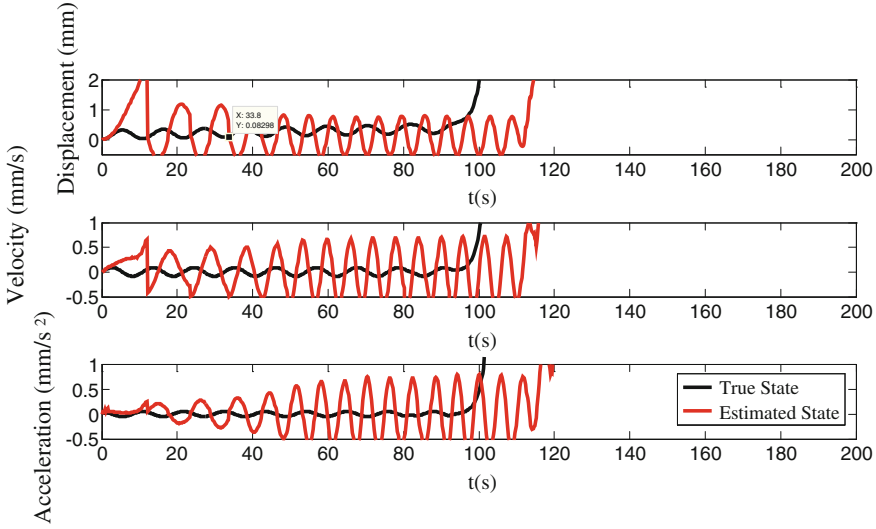


Fig. 2.21 State estimation by PF, linear softening constitutive law

proceeding with this objective, let us review again the particle filter algorithm. The procedure is triggered by drawing a number of N_p samples from a Gaussian distribution, then at each time instant t_k the same number of samples are drawn from transition prior. By transition prior, we mean a Gaussian distribution which its mean equals to the value of evolved estimated state at previous time step t_{k-1} while its covariance equals to the covariance of the process noise. This procedure practically is equal to generation of N_p Gaussian random numbers, and adding to them the value of \mathbf{x}_k which is evolved through evolution function. In the next stage, the probability of realization of each sample is computed. In this study, it is assumed that observation equation is contaminated by a white Gaussian process; hence, calculation of the probability of realization will be a function of a norm of the distance of the particle from the observation. The functional form of a multivariate Gaussian distribution reads as:

$$p(\mathbf{z}) = \frac{1}{\sqrt{2\pi|\boldsymbol{\Sigma}|}} e^{-\frac{1}{2}(\mathbf{z}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{z}-\boldsymbol{\mu})} \quad (2.52)$$

where $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ denote mean and covariance of the state vector, respectively; $|\cdot|$ stands for the determinant of the matrix. Within the PF algorithm, the above mentioned formula is used to compute the probability of realization associated with each particle $\mathbf{x}_k^{(i)}$, according to :

$$p(\mathbf{y}_k | \mathbf{x}_k^{(i)}) = \frac{1}{\sqrt{2\pi|\mathbf{W}|}} e^{-\frac{1}{2}(\mathbf{y}_k - \mathbf{h}_k(\mathbf{x}_k^{(i)}))^T \mathbf{W}^{-1}(\mathbf{y}_k - \mathbf{h}_k(\mathbf{x}_k^{(i)}))}. \quad (2.53)$$

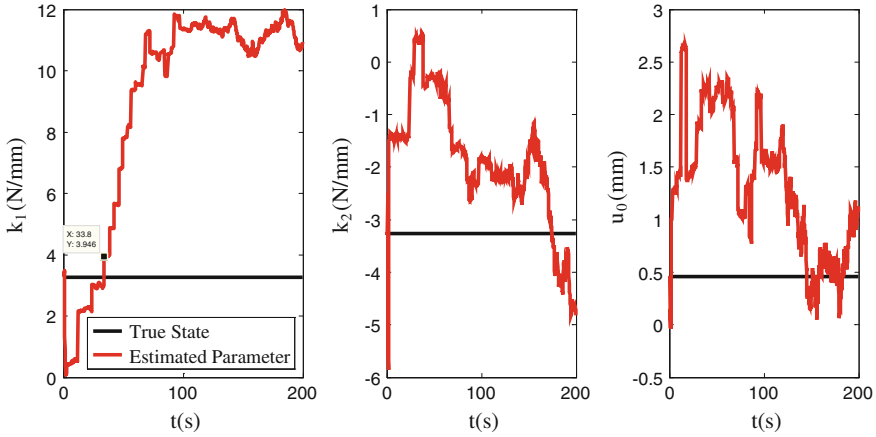


Fig. 2.22 Parameter estimation by PF, linear softening constitutive law

However, in case the observable part of the cloud of particles is too far from the observation y_k , the calculated probability will equal zero due to round off errors. To cope with ill-conditioning, it is set to a small value. As a result, all the particles will find an equal weight. In this condition, at the resampling stage, the resampled cloud will not change considerably, and will be similar to the previously existing cloud of particles. If the observable part of the cloud of particles approaches to observation vicinity (i.e. the zone in which at least a number of the probabilities are not affected by round-off error) a sharp change in the estimation of the state will occur. The gradient of such change in estimation of the observable part of state vector is obviously toward improvement in the estimate; however, the hidden (unobserved) part of state entries may or may not change in the direction to converge to an unbiased estimate, as seen in Fig. 2.22.

To visualize the phenomenon, the time evolution of displacement and parameters of the system are shown in the same plot, see Fig. 2.23. Now we regard a few time intervals of interest, and look at the histograms of particles at some time instants picked before and after the jump, we keep the time instant $t = 11.92$ s as reference instant.

In Fig. 2.24 it is seen that cloud of particles is not including the observation and the distance of the closest bin to the observation is about 0.2 mm (the value of the observation is indicated by a red vertical bar in the graph). Consequently, all the probabilities become zero, due to the round-off errors. To cope with the problem of ill-conditioning caused by the zero probabilities, in case of a zero probability, it is set to the smallest value that the computer program used accounts for it. That is, all the particles find the same weight. Figure 2.26 shows the histograms of k_1 , k_2 and u_0 respectively. As a consequence of the equal weights of the particles; it is seen that, before and after resampling stage, the histograms are not changed (Fig. 2.25).

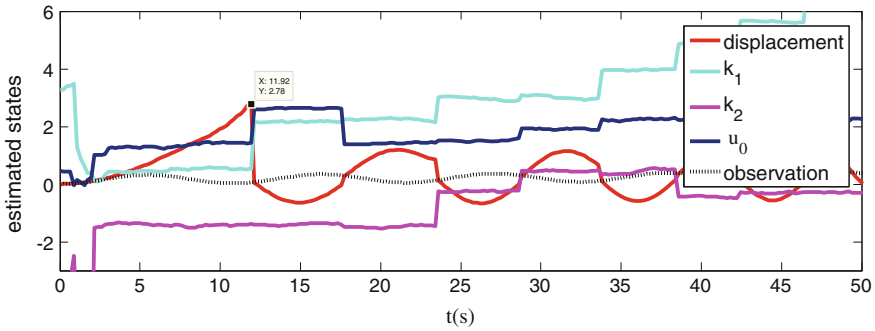


Fig. 2.23 State and parameter estimation by use of PF

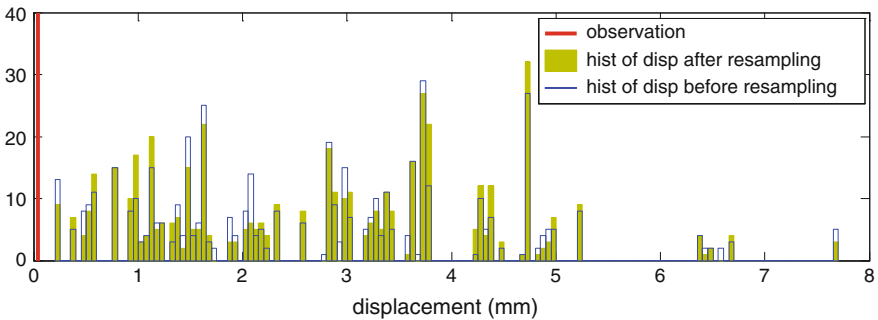


Fig. 2.24 Histogram of estimated displacements @ $t = 11.7$ s

Now let us look at $t = 11.92$, plots included in Figs. 2.27, 2.28 and 2.29 look considerably similar to previous time instant $t = 11.7$ s; however it seems that the cloud of samples is now closer to observation, as seen Fig. 2.27.

In what follows, histograms related to time instant $t = 12.13$ s are assessed. First see Fig. 2.30, in which the histogram of displacements is shown. Again, the red bar signifies the value observation y_k at related time instant, at its intersection with horizontal axis. It is seen that they are scattered throughout a wide interval; however, some particles have approached observation vicinity, as close as required to have non-zero weights for a couple of the particles, see Fig. 2.32.

To have a more clear idea, in Fig. 2.31 we have enlarged the vicinity of observation and histogram of resampled particles in order to highlight the changes in the particle cloud after resampling stage. We have to remark that the plot is an enlargement also in ordinate. It is clearly seen that a few particles (represented via blue histogram) have reached quite close to observation (red bar) so that their associated weight has become significant (see Fig. 2.32); as a consequence, in the resampling stage, the particles far from observation neighborhood are eliminated, and the ones close to it are duplicated. Figure 2.32 shows the weights associated

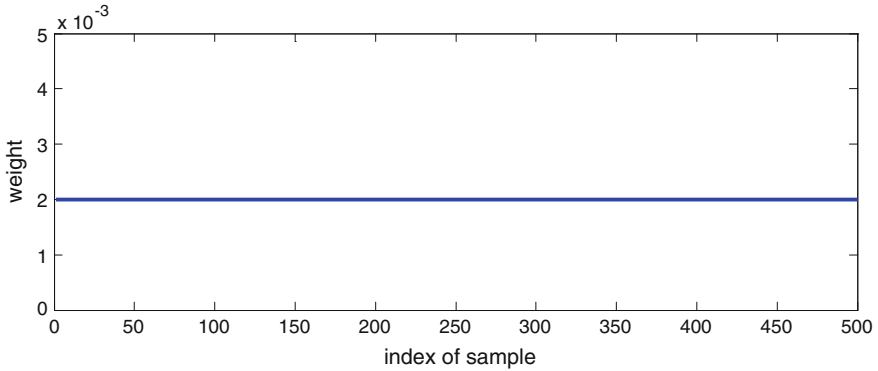


Fig. 2.25 Weights associated with each particle @ $t = 11.7$ s before resampling

with each particle. The peaks in Fig. 2.32 are the normalized weights associated with each particle before the resampling stage. The closer ones have visible peaks; there are also several peaks which are not visible in Fig. 2.32; once enlarged, those become visible as well; however they are about ten (see Fig. 2.33), nearly negligible when compared with the number of particles which in this case is 500.

As it is seen in Fig. 2.34, resampled particles do not necessarily move toward the target value; this is due to the fact that a wrong set of parameters has accompanied the shift of the samples toward the observation vicinity. Figure 2.34 well described the reason of failure of the PF in estimating states and parameters namely the distance of could of samples from observation vicinity. In order to alleviate such a problem, a remedy is to push the cloud of the samples toward observation vicinity. It can be done by employing the EKF: in each iteration, the EKF is used to update each particle by considering the information contained in the latest observation (de Freitas et al. 2000). More precisely, in the sampling stage, the samples are drawn from the transition prior; afterwards, each sample is updated by the EKF and so is pushed toward the observation vicinity. To some extent, this approach alleviates the problems arouse by choosing a suboptimal sampling distribution namely the transition prior. Figures 2.35, 2.36 and 2.37 show performance of a generic PF enhanced by the EKF. It is seen that such approach substantially improves the estimate of the parameters of the system.

To allow a clear understanding of the algorithm, let us look more closely at Fig. 2.37. Filter results from the initialization at 50 % of the target values is chosen just as an example. Figures 2.38 and 2.39 show the state and parameter estimation obtained through the EK-PF. It is seen that an excellent convergence is achieved. Figure 2.40 supports the idea that, by updating each individual particle within cloud of samples via EKF, the ensemble has to approach the zones of high probability.

As one can see in Fig. 2.40, after the EKF stage is implemented, the cloud of the samples drawn in the sampling stage moves toward the red bar (observation vicinity). In the resampling stage, the particles with higher probabilities are duplicated, and the ones with lower probability are eliminated; consequently, the

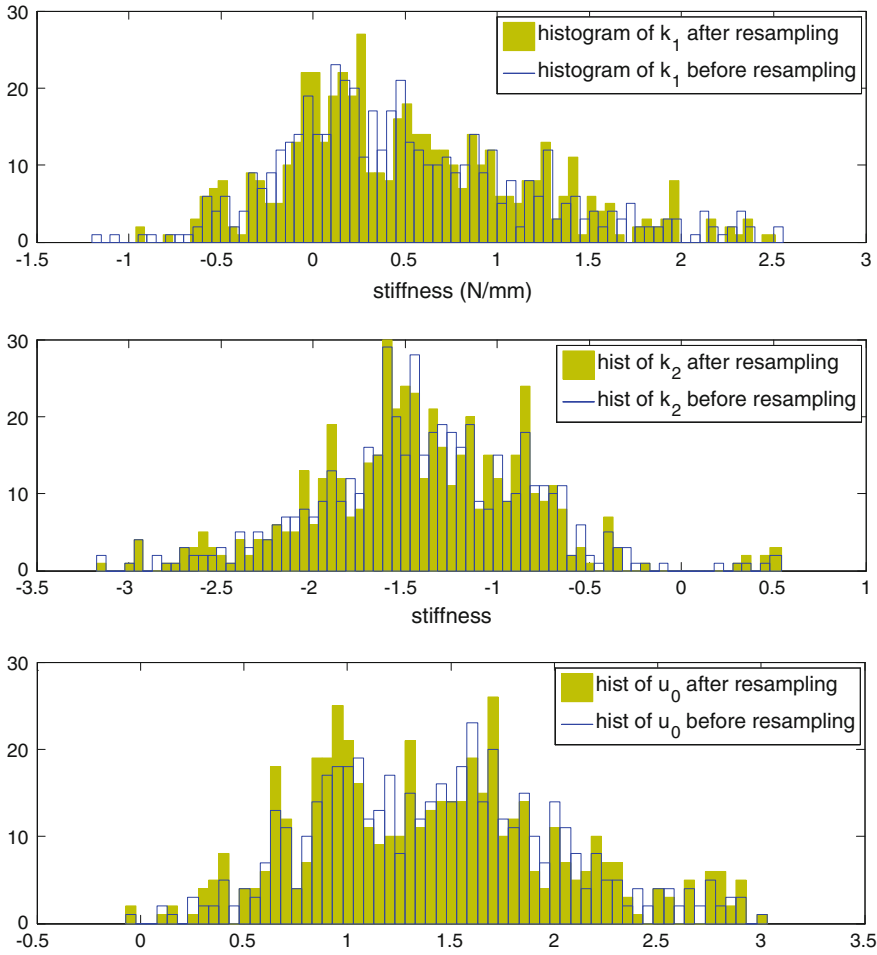


Fig. 2.26 Histogram of estimated parameters before and after resampling stage @ $t = 11.7$ s, top k_1 ; middle k_2 ; bottom u_0

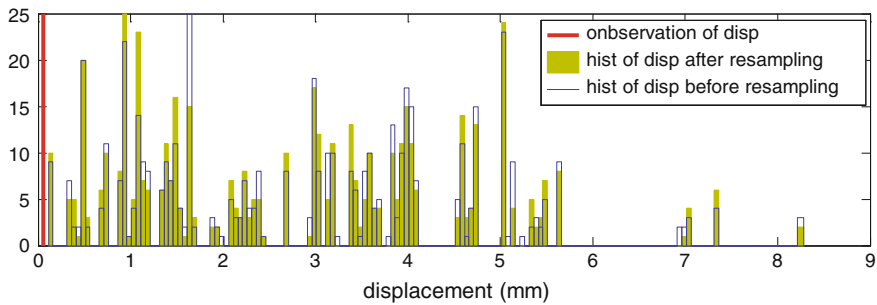


Fig. 2.27 Histogram of displacements @ $t = 11.92$ s

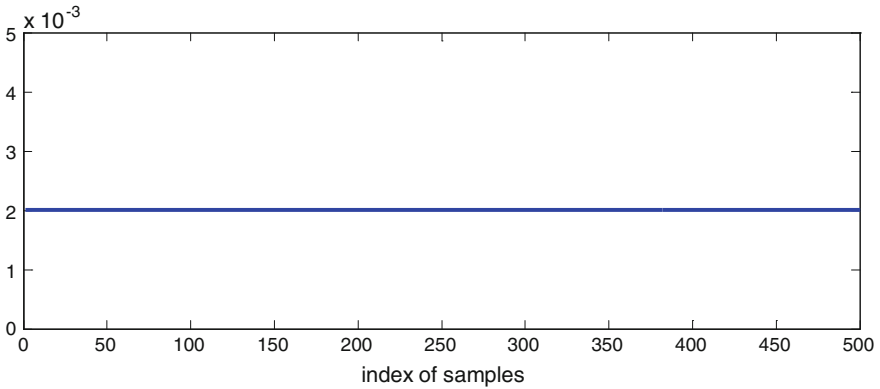


Fig. 2.28 Weights associated with each particle @ $t = 11.92$ s before resampling

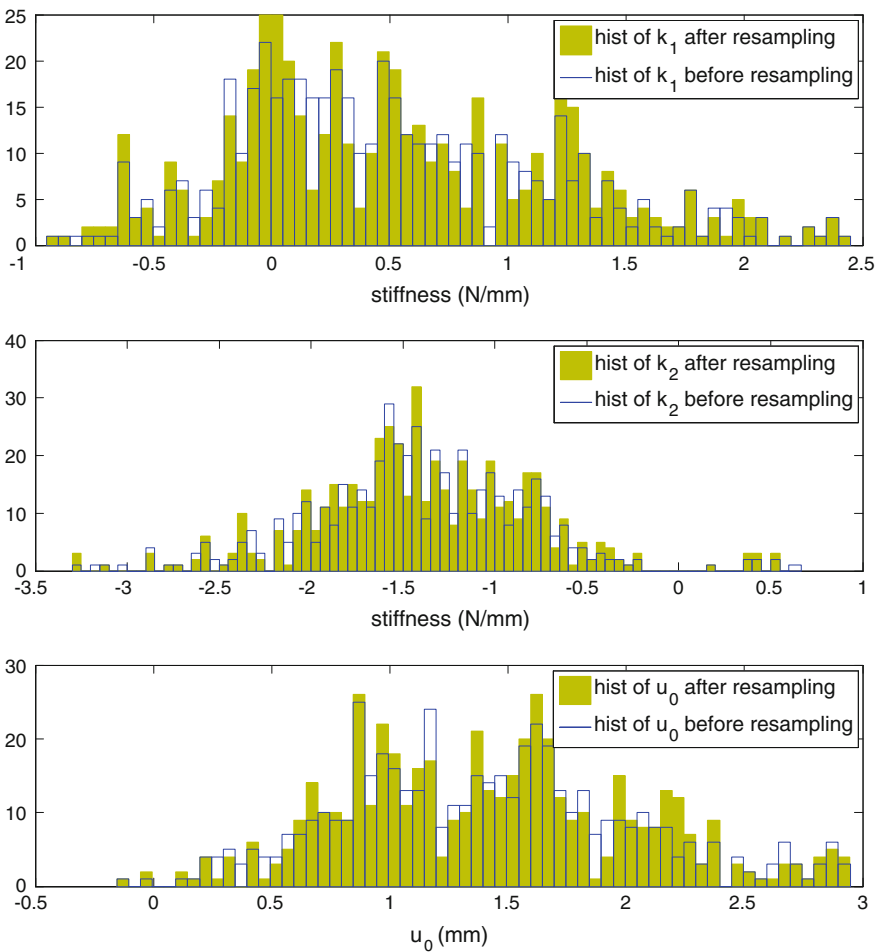


Fig. 2.29 Histograms of estimated parameters before and after resampling stage @ $t = 11.92$ s, top k_1 ; middle k_2 ; bottom u_0

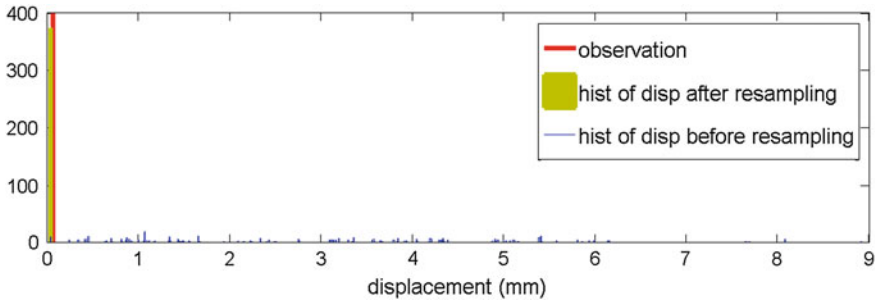


Fig. 2.30 Histogram of displacements @ $t = 12.13$ s

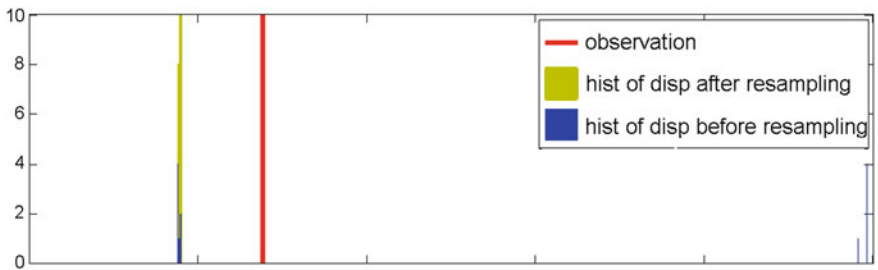


Fig. 2.31 Close up of histogram of displacements @ $t = 12.13$ s

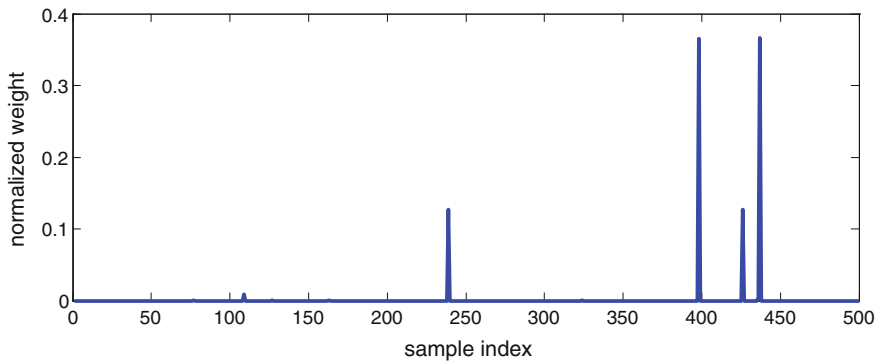


Fig. 2.32 Weights associated with each particle @ $t = 12.13$ s before resampling

cloud of the samples once again approaches the observation vicinity. Assessing other time instants always reveals the same results.

An extensive assessment of the performances of the Bayesian filters, when dealing with highly nonlinear dynamics of a SDOF system, has been presented. Though the studied mechanical system has only one degree-of-freedom, the

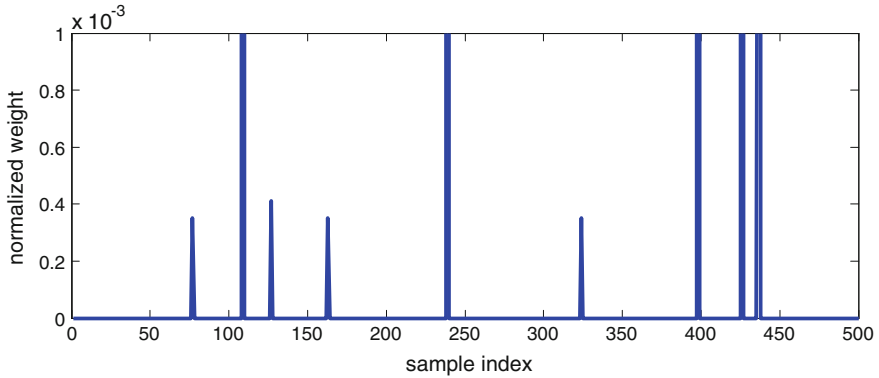


Fig. 2.33 Close up plot of weights associated with each particle @ $t = 12.13$ s before resampling

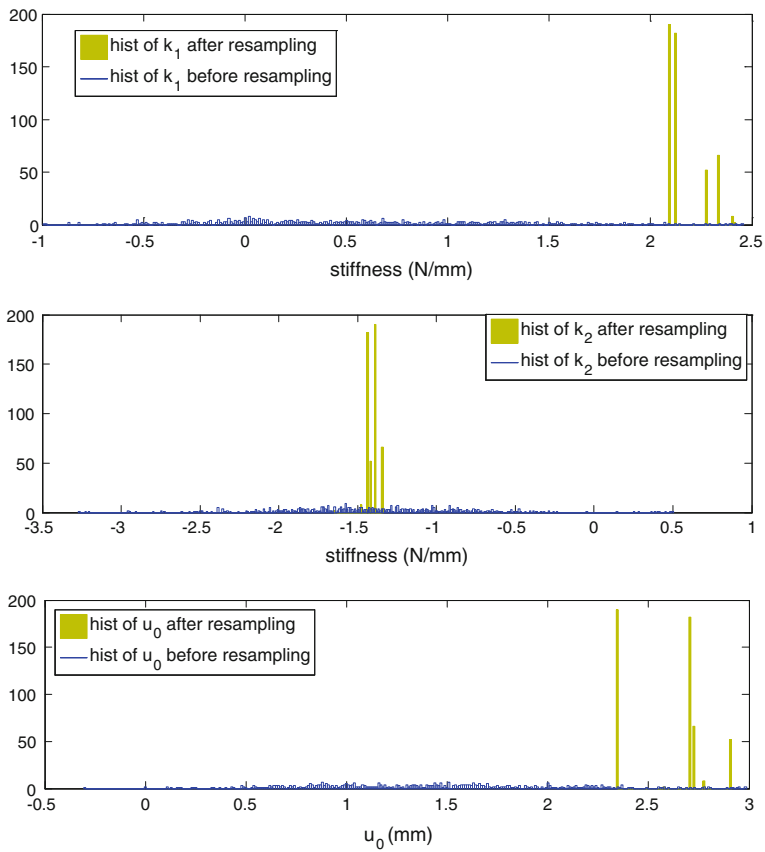


Fig. 2.34 Histogram of estimated parameters before and after resampling stage @ $t = 12.13$ s, *top* k_1 ; *middle* k_2 ; *bottom* u_0

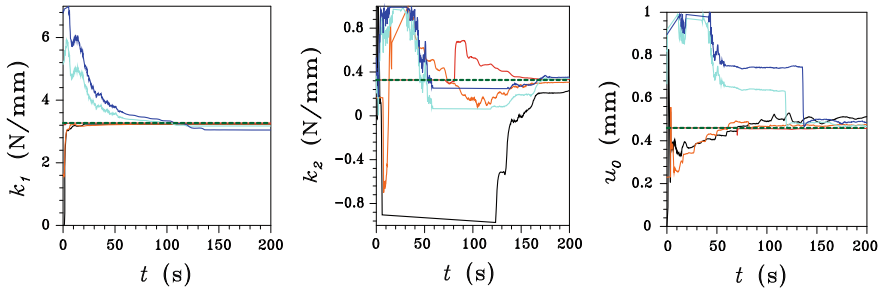


Fig. 2.35 Results of EK-PF for estimation of parameters of linear-hardening constitutive law

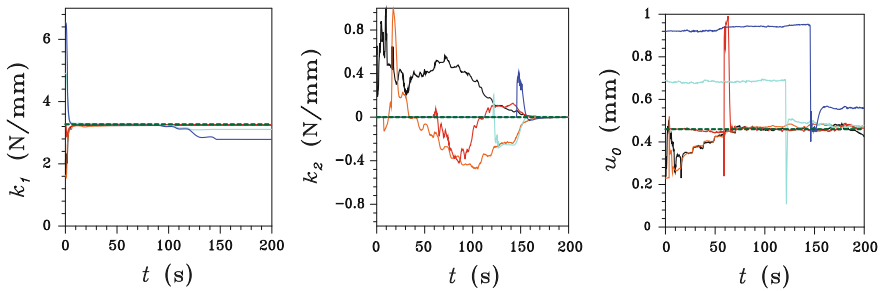


Fig. 2.36 Results of EK-PF for estimation of parameters of linear-plastic constitutive law

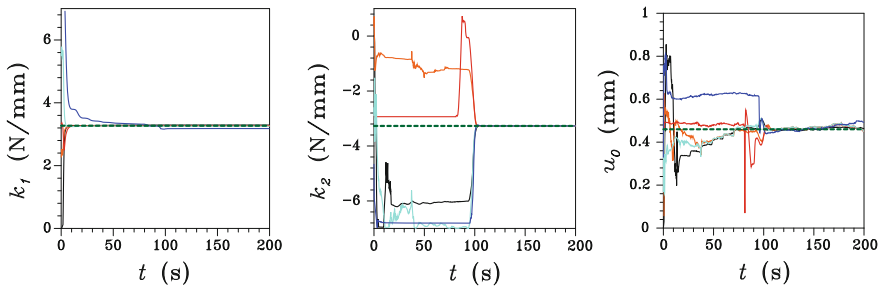


Fig. 2.37 Results of EK-PF for estimation of parameters of linear-softening constitutive law

extended state vector has three state components (displacement, velocity and acceleration) and 2 or 3 parameters (in case of a exponential softening constitutive law two parameters are to be calibrated, whereas in a bilinear one three parameters exist); consequently the extended state vector is multivariate even in present case. It was observed that the EKF, SPKF and PF all fail to furnish satisfactory results concerning identification of the parameters of the system, whereas EK-PF provides quite reasonable estimation of the states and parameters: for the exponential

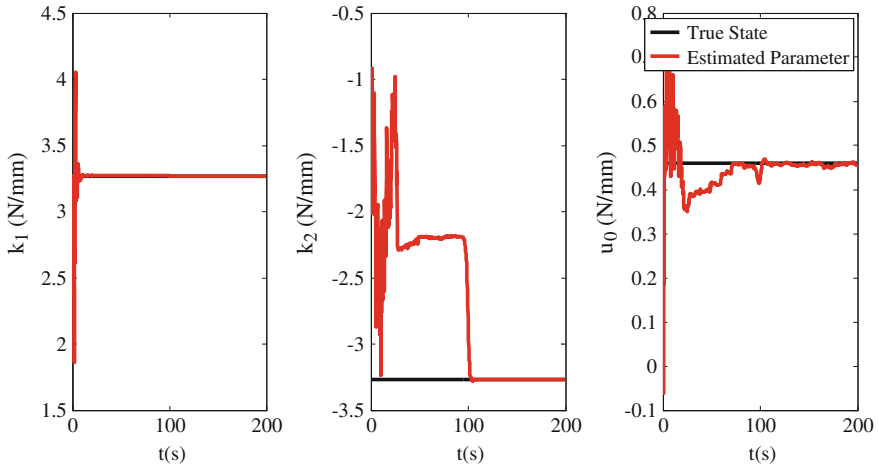


Fig. 2.38 Parameter estimation via EK-PF for a linear softening constitutive law

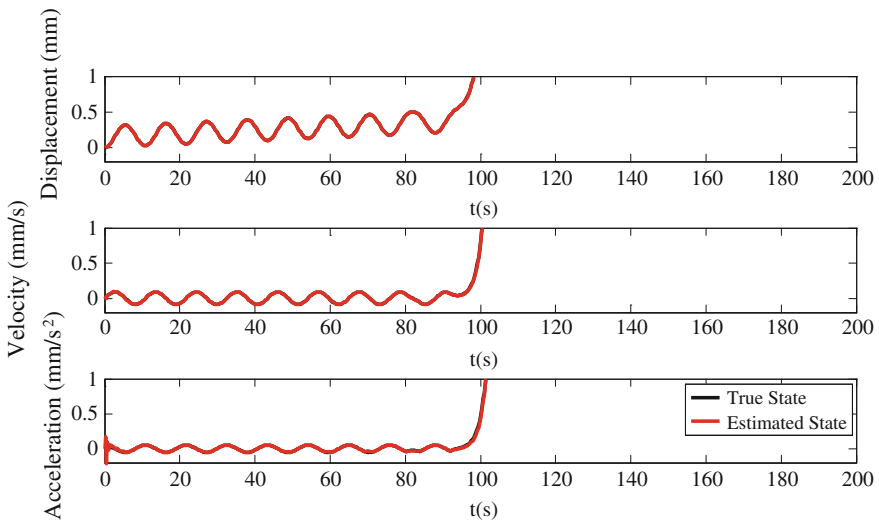


Fig. 2.39 State estimation via EK-PF for a linear softening constitutive law

behavior of the spring the results are unbiased for an extensive range of initializations; for the bilinear spring behavior EK-PF, in some cases, it converges to unbiased solutions, and in some others, it converges to values affected by small biases.

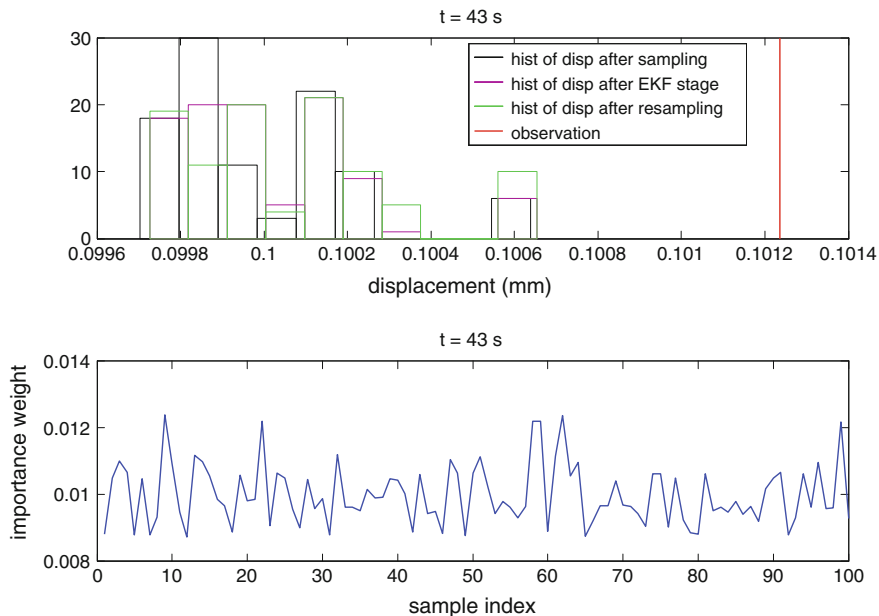


Fig. 2.40 Top histograms of displacement of the system at sampling (black hist), after EKF implemented on each sample (magenta hist) and after resampling stage (green hist), bottom associated importance weight with each particle

2.6.2 Multi Degrees-of-Freedom Dynamic System

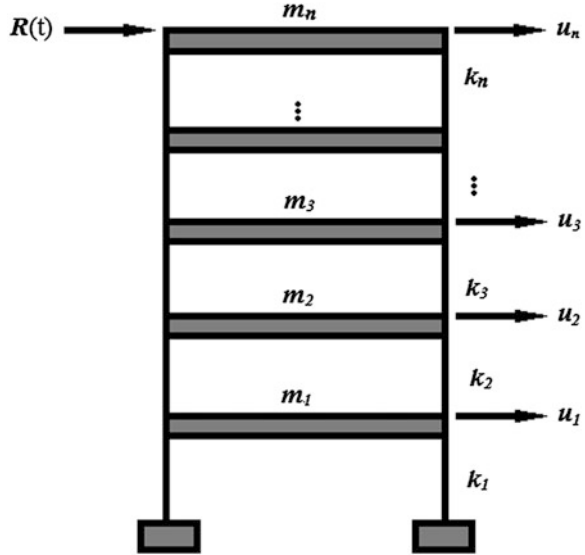
In this Section, dual estimation of state and parameters of a shear type building is studied, as seen in Fig. 2.41. To start with the simplest case, we focus on the linear elastic response. By neglecting dissipating phenomena, the governing equations of motion thus is expressed as:

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \mathbf{F}(t) \quad (2.54)$$

where \mathbf{M} and \mathbf{K} denote the stationary mass matrix and stiffness matrix, respectively:

$$\mathbf{M} = \begin{bmatrix} m_1 & & & \\ & m_2 & & \\ & & \ddots & \\ & & & m_n \end{bmatrix} \quad (2.55)$$

Fig. 2.41 Schematic view of a shear building



$$\mathbf{K} = \begin{bmatrix} k_1 + k_2 & -k_2 & & & & \\ -k_2 & k_2 + k_3 & & & & \\ & & \ddots & & & \\ & & & k_{n-1} + k_n & -k_n & \\ & & & -k_n & k_n & \end{bmatrix} \quad (2.56)$$

whereas $\mathbf{F}(t)$ is the external loading vector; in general, $\mathbf{F}(t)$ can be any kind of loading. However in this study, we assume that it is a harmonic force applied to the top floor:

$$\mathbf{F}(t) = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \varrho \sin \omega t \end{bmatrix} \quad (2.57)$$

where ϱ and ω are the amplitude and the frequency of the excitation, respectively. To numerically solve (2.54), the Newmark explicit time integrator has been used, see Eqs. (2.40–2.44).

To write the equations in a discrete state-space form, we introduce an extended state \mathbf{z} that, at each time instant t_k , includes \mathbf{u} , $\dot{\mathbf{u}}$ and $\ddot{\mathbf{u}}$ according to:

$$\mathbf{z}_k = \begin{bmatrix} \mathbf{u}_k \\ \dot{\mathbf{u}}_k \\ \ddot{\mathbf{u}}_k \end{bmatrix}. \quad (2.58)$$

The state-space form of (2.54) then reads:

$$\mathbf{z}_k = \mathbf{A}_k \mathbf{z}_{k-1} + \mathbf{B}_k \quad (2.59)$$

where

$$\mathbf{A}_k = \begin{bmatrix} \mathbf{I} - \beta \Delta t^2 \mathbf{M}^{-1} \mathbf{K} & \Delta t \mathbf{I} - \beta \Delta t^3 \mathbf{M}^{-1} \mathbf{K} & -\beta \left(\frac{1}{2} - \beta \right) \Delta t^4 \mathbf{M}^{-1} \mathbf{K} + \Delta t^2 \left(\frac{1}{2} - \beta \right) \mathbf{I} \\ -\gamma \Delta t \mathbf{M}^{-1} \mathbf{K} & \mathbf{I} - \gamma \Delta t^2 \mathbf{M}^{-1} \mathbf{K} & -\gamma \left(\frac{1}{2} - \beta \right) \Delta t^3 \mathbf{M}^{-1} \mathbf{K} + \Delta t (1 - \gamma) \mathbf{I} \\ -\mathbf{M}^{-1} \mathbf{K} & -\Delta t \mathbf{M}^{-1} \mathbf{K} & -\Delta t^2 \left(\frac{1}{2} - \beta \right) \mathbf{M}^{-1} \mathbf{K} \end{bmatrix} \quad (2.60)$$

and

$$\mathbf{B}_k = \begin{bmatrix} \beta \Delta t^2 \mathbf{M}^{-1} \mathbf{R}_k \\ \gamma \Delta t \mathbf{M}^{-1} \mathbf{R}_k \\ \mathbf{M}^{-1} \mathbf{R}_k \end{bmatrix} \quad (2.61)$$

In this study, it is assumed that displacements and accelerations of the floors can be measured; thus the observation equation is written as:

$$\mathbf{y}_k = \mathbf{H} \mathbf{z}_k + \mathbf{w}_k \quad (2.62)$$

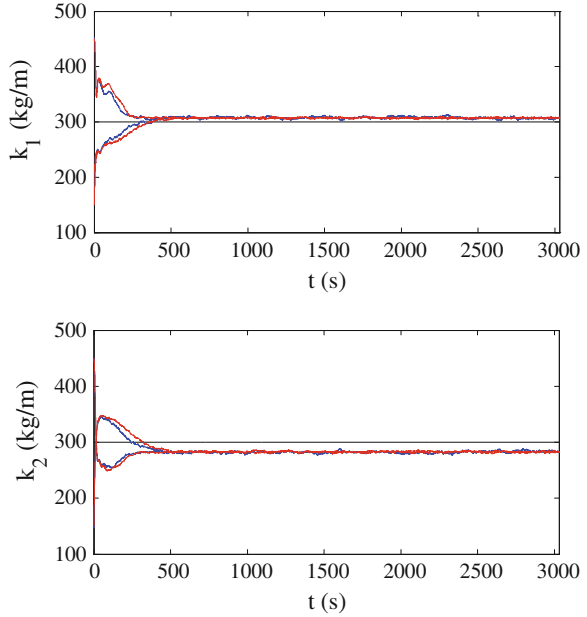
where \mathbf{H} denotes a Boolean matrix of appropriate dimension, which links the observation process to the state of the system; \mathbf{w}_k denotes the associated measurement noise; β and γ are parameters of the Newmark integration algorithm. For the dual estimation, the model parameter vector results:

$$\boldsymbol{\vartheta} = \begin{bmatrix} k_1 \\ k_2 \\ \vdots \\ k_n \end{bmatrix}. \quad (2.63)$$

In the numerical analysis, we deal with a multiple-story shear building, featuring the same stiffness and mass values at each floor. We start by considering the smallest possible number of floors (say two), and see how many parameters are calibrated unbiasedly. In this regard, we assume $m_i = 25 \text{ kg}$ and $k_i = 300 \text{ kg/m}$ ($i = 1 : n$). The outcomes of state estimation and parameter calibration are a function of the quality and quantity of the information provided to the algorithms; by *quality*, we intend the accuracy of measurement devices, accuracy of the model of the system and initialization guess; by *quantity*, the number of degrees of freedom, whose evolution in time is measured, is intended.

This research focuses on the study of the effects of an increasing number of parameters in dual estimation of multi-dimensional mechanical systems. It has to be highlighted that the observable quantity is considered to be the displacement of the top floor only. Covariance of the measurement noise is assumed to be $2.7 \times 10^{-6} \text{ m}^2$; the initial covariance of states (displacement, velocity and acceleration) is supposed to be very small (10^{-10} m^2), whereas diagonal entries of initial

Fig. 2.42 EKF (red line) and EK-PF (blue line) performances for calibration of a two-storey shear building stiffness's. The black line always represents the target value

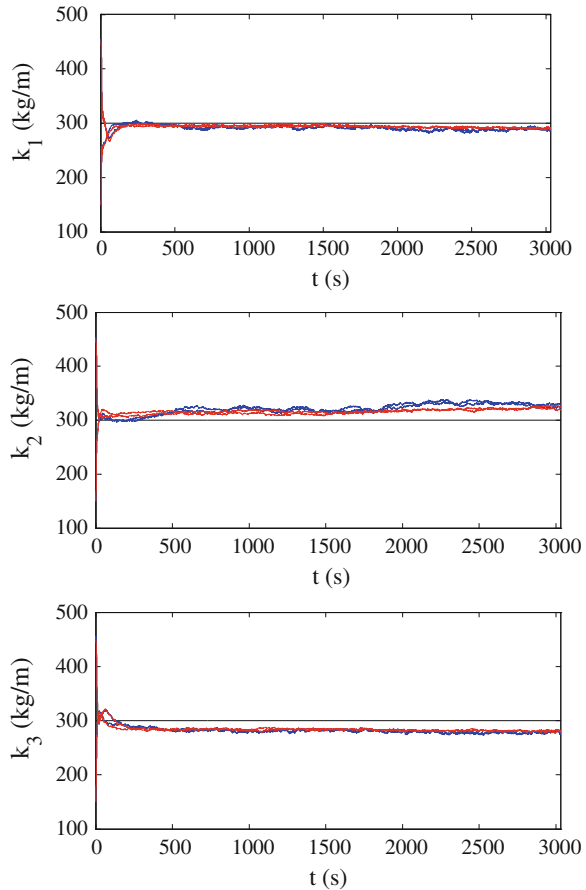


covariance of unknown parameters are assumed to be $10 \text{ kg}^2/\text{m}^2$. In all the analyses, the covariance of the fictitious noise for tuning the parameters is set to $10^{-3} \text{ kg}^2/\text{m}^2$. Since states are always tracked unbiasedly, for the sake of brevity, the relevant results are not reported.

To ensure the algorithm has reached an unbiased estimate, it is a common practice to run analysis starting from different initializations; in case all converge to the same estimate, then it may be most likely an unbiased estimate. In this case, we initialize the analyses by values 50 % less and 50 % more than target value. We begin our numerical assessment by study of a two DOF structure and report the results of parameter estimation in Fig. 2.42: it is observed that two filters exhibit the same performances. In EK-PF procedure, 20 particles are deployed; by increasing the number of particles to 200, changes are visible in the plots of Fig. 2.42. Hence, the number of the particles was fixed to 20.

Though by increasing the number of particles toward infinity, particle filter can furnish unbiased estimates (Cadini et al. 2009), in practice, such a number of particles may be intractable for the current power of computational tools. By increasing the number of unknown parameters, it is observed that the bias in the estimates becomes more visible. In Fig. 2.43, it is seen that again both EKF and EK-PF exhibit the same performance; however, the bias in the estimates is increased when compared to a 2-DOF system. Moving to a 3-DOF and 4-DOF system, Figs. 2.43 and 2.44 reports the results when three and four inter-storey stiffnesses has to be estimated, respectively. Comparing with the case of a 2-storey shear building, again the bias in the estimate of the parameters increases.

Fig. 2.43 EKF (red line) and EK-PF (blue line) performances for calibration of a three-storey shear building stiffness's. The black line always represents the target value



By exploring the literature concerning online methods for the identification of structures, one will see that most of it is focused on shear building structures with less than four stories [e.g. see (Chatzi et al. 2010; Gao and Lu 2006; Koh et al. 1995; Xie and Feng 2011)]. We avoid showing the results concerning estimation of more complicated structures, since they confirm the same trend observed in this reported part of the analysis. As the dimension of the state vector (hence the number of the parameters) increases, estimation of the parameters become increasingly difficult; in the jargon of dynamic programming, such a problem is termed *curse of dimensionality* (Bellman 1957). Powell (2007) illustrates this issue via an intuitive example: if state space has i dimensions and if each state component can take j possible values then we might have i^j possible states, i.e. by a linear increase in dimension of state vector, the dimension of the space of possibilities increases exponentially.

A possible remedy, for problems featuring high dimensionalities, is represented by searching for a possible subspace capturing the main variation in data; in

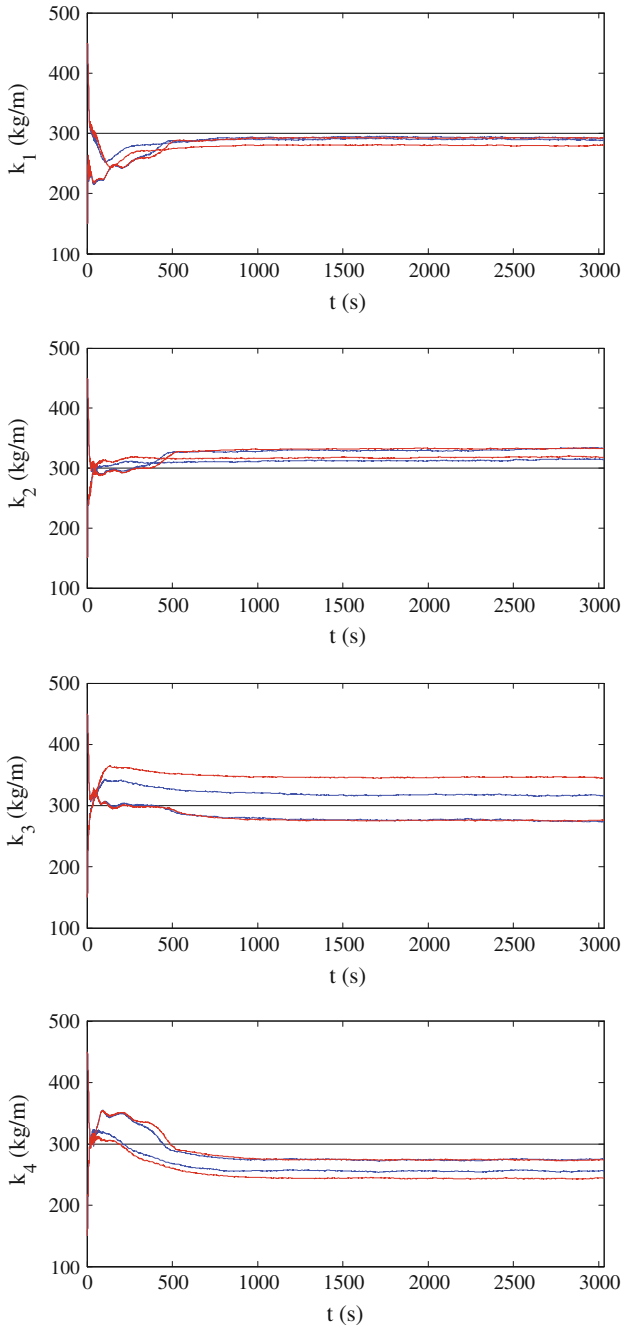


Fig. 2.44 EKF (red line) and EK-PF (blue line) performances for calibration of a four-storey shear building stiffness's. The black line always represents the target value

forthcoming Chapters, applicability of Proper Orthogonal Decomposition (POD) is primarily shown in constructing reduced order models, and afterwards such a model will be embedded in filtering schemes.

2.7 Summary and Conclusions

In this chapter, recursive Bayesian inference of partially observed dynamical systems has been reviewed. As a tool for structural system identification, nonlinear Bayesian filters are applied to dual estimation problem of linear and nonlinear dynamical systems. Dealing with a SDOF structure, it has been shown that the hybrid EK-PF filter is able to furnish a reasonable estimation of parameters of nonlinear constitutive models. Assessment of SDOF systems is followed by identification of multi storey buildings. In this regard, performances of the EK-PF and EKF algorithms are compared, and it has been concluded that they are nearly the same, and by an increase in the number of storeys of the building, the algorithms fail to provide an unbiased estimate of the parameters (stiffness of the storeys). Therefore, they are not reliable tools to monitor state and parameters of multi storey systems.

To develop a robust algorithm to monitor health of the structures via recursive Bayesian inference, we will make recourse to model order reduction of the dynamic systems. To this end, next Chapter reviews important features of proper orthogonal decomposition and its application to model order reduction of dynamic systems.

References

- Adelino R, da Silva Ferreira (2009) Bayesian mixture models of variable dimension for image segmentation. *Comput Methods Programs Biomed* 94:1–14
- Allen D, Darwiche A (2008) RC_Link: genetic linkage analysis using Bayesian networks. *Int J Approximate Reasoning* 48:499–525
- Alvarado Mora MV, Romano CM, Gomes-Gouvêa MS, Gutierrez MF, Botelho L, Carrilho FJ, Pinho JRR (2011) Molecular characterization of the Hepatitis B virus genotypes in Colombia: a Bayesian inference on the genotype F. *Infect, Genet Evol* 11:103–108
- Arulampalam MS, Maskell S, Gordon N, Clapp T (2002) A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Trans Sig Process* 50:174–188
- Bathe K (1996) *Finite element procedures*. Prentice-Hall Inc, Upper Saddle River
- Bellman RE (1957) *Dynamic programming*. Princeton University Press, Princeton
- Biedermann A, Taroni F (2012) Bayesian networks for evaluating forensic DNA profiling evidence: a review and guide to literature. *Forensic Sci Int: Genet* 6(2):147–157
- Bittanti S, Savaresi SM (2000) On the parameterization and design of an extended Kalman filter frequency tracker. *IEEE Trans Autom Control* 45:1718–1724
- Bittanti S, Maier G, Nappi A (1984) Inverse problems in structural elastoplasticity: a Kalman filter approach. In: Sawczukand A, Bianchi G (eds) *Plasticity today*. Applied Science Publications, London, pp 311–329

- Cadini F, Zio E, Avram D (2009) Monte Carlo-based filtering for fatigue crack growth estimation. *Probab Eng Mech* 24:367–373
- Caron F, Doucet A, Gottardo R (2012) On-line change point detection and parameter estimation with application to genomic data. *Stat Comput* 22:579–595
- Chatzi EN, Smyth AW, Masri SF (2010) Experimental application of on-line parametric identification for nonlinear hysteretic systems with model uncertainty. *Struct Saf* 32:326–337
- Corigliano A (1993) Formulation, identification and use of interface models in the numerical analysis of composite delamination. *Int J Solids Struct* 30:2779–2811
- Corigliano A, Mariani S (2001a) Parameter identification of a time-dependent elastic-damage interface model for the simulation of debonding in composites. *Compos Sci Technol* 61:191–203
- Corigliano A, Mariani S (2001b) Simulation of damage in composites by means of interface models: parameter identification. *Compos Sci Technol* 61:2299–2315
- Corigliano A, Mariani S (2004) Parameter identification in explicit structural dynamics: performance of the extended Kalman filter. *Comput Methods Appl Mech Eng* 193:3807–3835
- Corigliano A, Mariani S, Pandolfi A (2006) Numerical analysis of rate-dependent dynamic composite delamination. *Compos Sci Technol* 66:766–775
- Creal D (2012) A survey of sequential Monte Carlo methods for economics and finance. *Econ Rev* 31(3):245–296
- de Freitas JFG, Niranjan MA, Gee AH, Doucet A (2000) Sequential Monte Carlo methods to train neural network models. *Neural Comput* 12:955–993
- Doucet A (1997) Monte Carlo methods for Bayesian estimation of hidden Markov models: application to radiation signals. (unpublished) doctoral dissertation, University Paris-Sud Orsay
- Doucet A, Johansen AM (2009) A tutorial on particle filtering and smoothing: fifteen years later. *Handbook of Nonlinear Filtering* 12:656–704
- Duan L, Gao W, Zeng W, Zhao D (2005) Adaptive relevance feedback based on Bayesian inference for image retrieval. *Signal Process* 85:395–399
- Eftekhar Azam S, Mariani S (2012) Dual estimation of partially observed nonlinear structural systems: a particle filter approach. *Mech Res Commun* 46:54–61
- Eftekhar Azam S, Ghisi A, Mariani S (2012a) Parallelized sigma-point Kalman filtering for structural dynamics. *Comp Struct* 92–93, pp. 193–205
- Eftekhar Azam S, Bagherinia M, Mariani S (2012b) Stochastic system identification via particle and sigma-point Kalman filtering. *Scientia Iranica A*, 19:982–991
- Gao F, Lu Y (2006) A Kalman-filter based time-domain analysis for structural damage diagnosis with noisy signals. *J Sound Vib* 297:916–930
- Gelb A (1974) *Applied optimal estimation*. MIT Press, Cambridge
- Gordon NJ, Salmond DJ, Smith AFM (1993) Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE proceedings F*, vol 140, pp 107–113
- Hol JD, Schon TB, Gustafsson F (2006) On resampling algorithms for particle filtering. In: *Proceedings of nonlinear statistical signal processing workshop 2006*, pp 79–82
- Holmes S, Klein G, Murray DW (2008) A square root unscented Kalman filter for visual monoSLAM. In: *Proceedings—EEE international conference on robotics and automation*, p 3710
- Hughes TJR (2000) *The finite element method. Linear static and dynamic finite element analysis*. Dover, New York
- Ishihara T, Omori Y (2012) Efficient Bayesian estimation of a multivariate stochastic volatility model with cross leverage and heavy-tailed errors. *Comput Stat Data Anal* 56(11):3674–3689
- Ito K, Xiong K (2000) Gaussian filters for nonlinear filtering problems. *IEEE Trans Autom Control* 45:910–927
- Jay E, Philippe Ovarlez J, Declercq D, Duvaut P (2003) BORD: Bayesian optimum radar detector. *Signal Process* 83:1151–1162
- Julier SJ, Uhlmann JK (1997) New extension of the Kalman filter to nonlinear systems. *Proceedings of SPIE—the international society for optical engineering*, pp 182–193
- Julier SJ, Uhlmann JK, Durrant-Whyte HF (1995) New approach for filtering nonlinear systems. In: *Proceedings of the American control conference*, pp 1628–1632

- Julier S, Uhlmann J, Durrant-Whyte HF (2000) A new method for the nonlinear transformation of means and covariances in filters and estimators. *IEEE Trans Autom Control* 45:477–482
- Kalman RE (1960) A new approach to linear filtering and prediction problems. *J Basic Eng* 82:35–45
- Kitagawa G (1996) Monte Carlo filter and smoother for non-Gaussian nonlinear state space models. *J Comput Graphical Stat* 5:1–25
- Koh CG, See LM, Balendra T (1995) Determination of storey stiffness of three-dimensional frame buildings. *Eng Struct* 17:179–186
- Lazkano E, Sierra B, Astigarraga A, Martínez-Otzeta JM (2007) On the use of Bayesian Networks to develop behaviours for mobile robots. *Rob Auton Syst* 55:253–265
- Li P, Goodall R, Kadirkamanathan V (2004) Estimation of parameters in a linear state space model using a Rao-Blackwellised particle filter. *IEE proceedings: control theory and applications*, vol 151. pp 727–738
- Ljung L (1999) *System identification. Theory for the user*, 2nd edn. Prentice Hall, Englewood Cliffs
- Mariani S (2009a) Failure of layered composites subject to impacts: constitutive modeling and parameter identification issues. In: Mendes G, Lago B (eds) *Strength of materials*. Nova Science Publishers, New York, pp 97–131
- Mariani S (2009b) Failure assessment of layered composites subject to impact loadings: a finite element, sigma-point Kalman filter approach. *Algorithms* 2:808–827
- Mariani S, Corigliano A (2005) Impact induced composite delamination: state and parameter identification via joint and dual extended Kalman filters. *Comput Methods Appl Mech Eng* 194:5242–5272
- Mariani S, Ghisi A (2007) Unscented Kalman filtering for nonlinear structural dynamics. *Nonlinear Dyn* 49:131–150
- Miazhyńska T, Frühwirth-Schnatter S, Dorffner G (2006) Bayesian testing for non-linearity in volatility modeling. *Comput Stat Data Anal* 51:2029–2042
- Mitra SK, Lee T, Goldbaum M (2005) A Bayesian network based sequential inference for diagnosis of diseases from retinal images. *Pattern Recogn Lett* 26:459–470
- Powell WB (2007) *Approximate dynamic programming: solving the curse of dimensionality*. Princeton University Press, Princeton
- Rose JH, Ferrante J, Smith JR (1981) Universal binding energy curves for metals and bimetallic interfaces. *Phys Rev Lett* 47:675–678
- Saleh GMK, Niranjana M (2001) Speech enhancement using a Bayesian evidence approach. *Comput Speech Lang* 15:101–125
- Ting J, D'Souza A, Schaal S (2011) Bayesian robot system identification with input and output noise. *Neural Networks* 63:99–108
- Van der Merwe, R. 2004, *Sigma-point Kalman filters for probabilistic inference in dynamic state-space models*, Oregon Health and Science University
- Velarde LGC, Migon HS, Alcoforado DA (2008) Hierarchical Bayesian models applied to air surveillance radars. *Eur J Oper Res* 184:1155–1162
- White OL, Safaeinili A, Plaut JJ, Stofan ER, Clifford SM, Farrell WM, Heggy E, Picardi G (2009) MARSIS radar sounder observations in the vicinity of Ma'adim Vallis, Mars. *Icarus* 201:460–473
- Xie Z, Feng J (2011) Real-time nonlinear structural system identification via iterated unscented Kalman filter. *Mechanical Syst Signal Process* 28:309–322
- Yahya AA, Mahmud R, Ramli AR (2010) Dynamic Bayesian networks and variable length genetic algorithm for designing cue-based model for dialogue act recognition. *Comput Speech Lang* 24:190–218
- Yang S, Lee J (2011) Predicting a distribution of implied volatilities for option pricing. *Expert Syst Appl* 38:1702–1708
- Zhou H, Sakane S (2007) Mobile robot localization using active sensing based on Bayesian network inference. *Rob Auton Syst* 55:292–305



<http://www.springer.com/978-3-319-02558-2>

Online Damage Detection in Structural Systems
Applications of Proper Orthogonal Decomposition, and
Kalman and Particle Filters

Eftekhar Azam, S.

2014, XII, 135 p. 87 illus., Softcover

ISBN: 978-3-319-02558-2