Mechanical Systems and Signal Processing 209 (2024) 111060

Contents lists available at ScienceDirect

Mechanical Systems and Signal Processing

journal homepage: www.elsevier.com/locate/ymssp

Multiple local particle filter for high-dimensional system identification

Tianzhi Li, Claudio Sbarufatti, Francesco Cadini *

Dipartimento di Meccanica, Politecnico di Milano, Milan, Italy

ARTICLE INFO

Keywords:
Structural health monitoring
System identification
Particle filter
Curse of dimensionality
Decay of correlations

ABSTRACT

Nonlinearity and high dimensionality emerge as two primary challenges in the realm of system identification within the context of structural health monitoring (SHM) applications. Particle filter (PF) has been demonstrated efficient for nonlinear identification, but it suffers from the curse of dimensionality and behaves poorly in high-dimensional problems. The idea of state and measurement partitioning has been used in many PF algorithms to simplify high-dimensional identification problems into the identification of several lower-dimensional subgroups, but with very few applications to SHM problems. In this context, by combining multiple particle filters (MPF) with the decay of correlations property, this paper develops a novel multiple local particle filter (MLPF) for high-dimensional identification problems. A whole state vector is partitioned into several state subgroups, each consisting of fewer state components and then estimated by one PF through a novel likelihood including the local state and measurement vectors. The feasibility and efficiency of the proposed method are tested through a benchmark toy example, a case study of a twenty-story Bouc-Wen frame structure under ground motion, and an experimental study of fatigue delamination growth in composites.

1. Introduction

Structural health monitoring (SHM) has seen a lot of research in the field of system identification, utilizing various time-domain techniques such as least square method [1], extended Kalman filter [2,3], unscented Kalman filter [4,5], and particle filter (PF) [6–11]. These methods use a state space model to describe the dynamic behavior of a monitored structure and then provide the estimates of the parameters and state variables, which finally serve specific objectives, including diagnostics [1,12], prognostic [7–11], and structural control [3]. As the parameters and the state variables are usually combined to form an augmented state vector and coupled in both the process and measurement equations, problems with nonlinearity and high dimensionality will arise.

Particle filter (PF) has been demonstrated successful for a large variety of nonlinear SHM applications, such as structural identification [5,13], crack propagation prediction in metal [7,8,10,14,15], and fatigue damage prediction in composites [9,16,17]. However, these applications can involve the deployment of multiple sensors and the task of inferring tens or even hundreds of state variables and parameters, which will give rise to high dimensional problems. Illustrative examples consist of the identification of the state and parameters for a multiple-degree-of-freedom structure [12], the growth predictions of multiple fatigue cracks in a metallic plate [18,19], and the fatigue delamination shape prediction in composites [9]. More specifically, as more components (i.e., parameters and variables) are added into the state vector for estimation, the number of particles required by PF will be exponentially

* Corresponding author.
E-mail address: francesco.cadini@polimi.it (F. Cadini).

https://doi.org/10.1016/j.ymssp.2023.111060
Received 10 May 2023; Received in revised form 2 November 2023; Accepted 18 December 2023
Available online 4 January 2024

increased with the vector dimension, consequently leading to tremendous, or sometimes unaffordable, computation efforts [20,21]. This is well known as the curse of dimensionality. Very few PF modifications [6,22] are proposed in SHM practices to alleviate the effect of the curse of dimensionality. They consist in partitioning the state vector into the two state subgroups, namely linear and nonlinear ones, and then combining PF with the marginalization approach for estimation. However, the dimensions of either one subgroup or both of them could still be quite large in a high-dimensional system, limiting the usage of those methods.

The curse of dimensionality is a well-known issue when using PF in engineering applications beyond SHM, such as multiple target tracking [23], where a whole state vector always consists of the unknown components (e.g., the location coordinates) from multiple targets. A common modification in these applications is to partition the whole state vector into several state subgroups, each having the components of one target and then estimated through one PF, based on the assumptions of the independent posterior distribution and state evolution between the subgroups. By simplifying a problem of high-dimensional identification into that of multiple lower-dimensional identifications, this idea has led to the development of successful high-dimensional PF algorithms, including multiple particle filter (MPF) [24,25] and its extensions [23,26], and independent partition particle filter [27] and its extensions [28,29].

On one hand, state space models commonly used in SHM practices cannot always satisfy the assumption of independent state evolution. For example, when adopting a state partitioning strategy from multiple target tracking [24,25], a state vector containing multiple degrees of freedom (DOFs) can be partitioned into several subgroups, each consisting of components (e.g., velocity and displacement) from a single DOF. However, it always happens that one subgroup has to evolve with the state components from itself and another subgroup, for example, in the case of calculating the hysteretic displacement of the top story in a multiple-story Bouc-Wen frame using the velocities from that story and its lower story. This violates the aforementioned assumption and restricts the applicability of existing algorithms for structural system identification.

On the other hand, the likelihood calculation in a high-dimensional problem can impose a significant computation burden [30], as seen in the PF algorithms mentioned above, where calculating the likelihood of one subgroup involves either all state and measurement components [23–26] or partial state components and all measurements [27–29]. To improve the computation efficiency, a modified likelihood with fewer state and measurement components involved is always desirable. The decay of correlations property indicates that, as a high-dimensional system is spatially distributed over relatively large distances, the conditional distribution of one state subgroup should significantly depend on the collected measurements at its nearby location [31]. This allows for estimating the posterior distribution of each subgroup with its nearby measurements. Fewer measurements in posterior estimation bring to fewer state and measurement components in likelihood, as shown in the use of PF in the application of target tracking [32,33], providing a mechanism to reduce the computational cost.

In this study, we propose two contributions to high-dimensional structural identification. First, we extend the concept of state partitioning to account for dependent state evolution. Specifically, we partition one state vector into several state subgroups, each containing fewer state components and evolving with itself and additional state components. Next, we develop a novel multiple local particle filter (MLPF) by combining the decay of correlations property and the idea of state partitioning. The MLPF utilizes a novel likelihood function that includes fewer state and measurement components, resulting in computational savings. The feasibility of the extension is tested through a numerical simulation of a twenty-story Bouc-Wen frame structure under the ground motion and an experimental study of fatigue delamination growth in composites [9], and the efficiency of the proposed MLPF is validated by a toy example from [20], and the above numerical and experimental studies.

The remainder of this paper is organized as follows: Section 2 describes the particle filter, multiple particle filter, the extension about dependent state evolution, and the novel multiple local particle filter. Section 3 uses a toy example to illustrate the effect of curse of dimensionality on PF-based identification and to compare the performances of the three algorithms. Section 4 uses the numerical simulation of a twenty-story Bouc-Wen frame structure to validate the extension made for dependent state evolution and the superiority of MLPF over PF and MPF. The generalizability of MLPF to different SHM applications is tested by the experimental study of fatigue delamination growth in composites in Section 5. Finally, Section 6 concludes the paper with some topics for further work.

Table 1
Description of some essential symbols used in Section 2.

Symbol	Explanation	which algorithm requires the symbol		
		PF	MPF	MLPF
$\boldsymbol{X}_k$	State vector at k-th step	Y	Y	Y
$\boldsymbol{x}^j$	j-th state subgroup	–	Y	Y
$\boldsymbol{t}^j$	A vector of additional state components for state evolution of $\boldsymbol{x}^j$, obtained from process equation	–	Y	Y
$\boldsymbol{q}^j$	Local state vector for $\boldsymbol{x}^j$, obtained from local measurement equation	–	–	Y
$\boldsymbol{Y}_k$	Measurement vector at k-th step	Y	Y	Y
y_l	l-th measurement component	–	–	Y
$\boldsymbol{y}^j$	Local measurement vector for $\boldsymbol{x}^j$, obtained from local measurement equation	–	–	Y

Note: 'Y' stands for 'yes'.

2. Particle filter algorithms

In this Section, we first review some fundamental concepts related to the particle filter (PF) method, including the state space model, analytical calculation of posterior probability density function (PDF), Monte Carlo sampling, sequential importance sampling, and standard PF. Then, we introduce the multiple particle filter (MPF), which is designed for multiple target tracking, and propose an extension of dependent state evolution for MPF. We then develop the multiple local particle filter (MLPF) by introducing a new likelihood into the MPF. Finally, we propose a definition of local state and measurement vectors for MLPF and explain why MLPF can be more computationally efficient than MPF. Some essential symbols used in the section are given in Table 1.

2.1. Particle filter

The state space model for the system identification problem can be formulated as

$$\boldsymbol{X}_k = f(\boldsymbol{X}_{k-1}, \boldsymbol{u}_{k-1}, \boldsymbol{\omega}_{k-1}) \tag{1}$$

$$\boldsymbol{Y}_k = h(\boldsymbol{X}_k, \boldsymbol{u}_k, \boldsymbol{\eta}_k) \tag{2}$$

where the subscript k is the time step, $\boldsymbol{X}$, $\boldsymbol{u}$, and $\boldsymbol{Y}$ represent the state, input, and measurement vectors, $\boldsymbol{\omega}$ and $\boldsymbol{\eta}$ are the process and measurement noise vectors, and $f(\cdot)$ and $h(\cdot)$ are the process and measurement functions describing the system state evolution and the relationship between the states and the measurements, respectively.

From a Bayesian perspective, the problem of estimating the state vector $\boldsymbol{X}_k$ based on a sequence of the available measurements up to time step k $\boldsymbol{Y}_{1:k}$ is to recursively build the posterior probability density function (PDF) $p(\boldsymbol{X}_k|\boldsymbol{Y}_{1:k})$ as

$$p(\boldsymbol{X}_k|\boldsymbol{Y}_{1:k-1}) = \int p(\boldsymbol{X}_k|\boldsymbol{X}_{k-1}) p(\boldsymbol{X}_{k-1}|\boldsymbol{Y}_{1:k-1}) d\boldsymbol{X}_{k-1} \tag{3}$$

$$p(\boldsymbol{X}_k|\boldsymbol{Y}_{1:k}) \propto p(\boldsymbol{Y}_k|\boldsymbol{X}_k) p(\boldsymbol{X}_k|\boldsymbol{Y}_{1:k-1}) \tag{4}$$

where $\boldsymbol{Y}_{1:k}$ are the measurements collected from time step 1 to k, $\propto$ denotes proportionality, and $p(\boldsymbol{X}_k|\boldsymbol{X}_{k-1})$, $p(\boldsymbol{X}_k|\boldsymbol{Y}_{1:k-1})$, and $p(\boldsymbol{X}_k|\boldsymbol{Y}_{1:k})$ are the transition distribution, prior PDF, and likelihood function, respectively.

Obtaining an analytical solution of Eqs. (3) and (4) is rare in a nonlinear and non-Gaussian problem. As a result, Monte Carlo sampling is often used to approximate the posterior PDF with N particles $\{\boldsymbol{X}_k^i : i = 1, 2, \ldots, N\}$ as

$$p(\boldsymbol{X}_k|\boldsymbol{Y}_{1:k}) \approx \frac{1}{N} \sum_{i=1}^{N} \delta(\boldsymbol{X}_k - \boldsymbol{X}_k^i) \tag{5}$$

where $\delta(\cdot)$ is a Dirac-delta function.

Given the difficulty in sampling from the true posterior PDF, an alternative is to apply importance sampling to draw N samples from a properly chosen importance density $q(\boldsymbol{X})$ like $q(\boldsymbol{X}_k|\boldsymbol{Y}_{1:k})$ for building the PDF $p(\boldsymbol{X}_k|\boldsymbol{Y}_{1:k})$ as

$$p(\boldsymbol{X}_k|\boldsymbol{Y}_{1:k}) \approx \sum_{i=1}^{N} \widetilde{w}_k^i \delta(\boldsymbol{X}_k - \boldsymbol{X}_k^i) \tag{6}$$

where the weight $\widetilde{w}_k^i$ corresponding to the i-th particle $\boldsymbol{X}_k^i$ denotes the normalized form of the weight w_k^i calculated as

$$w_k^i \propto \frac{p(\boldsymbol{X}_k^i|\boldsymbol{Y}_{1:k})}{q(\boldsymbol{X}_k^i|\boldsymbol{Y}_{1:k})} \tag{7}$$

The computation effort of Eq. (7) will increase over time, as more measurements will be included in the vector $\boldsymbol{Y}_{1:k}$.

In order to improve the efficiency of computing the particle weight, the PDF $p(\boldsymbol{X}_{0:k}|\boldsymbol{Y}_{1:k})$ should be built with N particles $\{\boldsymbol{X}_{0:k}^i : i = 1, 2, \ldots, N\}$ from the density $q(\boldsymbol{X}_{0:k}|\boldsymbol{Y}_{1:k})$ as

$$p(\boldsymbol{X}_{0:k}|\boldsymbol{Y}_{1:k}) \approx \sum_{i=1}^{N} \widetilde{w}_k^i \delta(\boldsymbol{X}_{0:k} - \boldsymbol{X}_{0:k}^i) \tag{8}$$

then we have

$$\begin{aligned} w_k^i &\propto \frac{p(\boldsymbol{X}_{0:k}^i|\boldsymbol{Y}_{1:k})}{q(\boldsymbol{X}_{0:k}^i|\boldsymbol{Y}_{1:k})} \\ &\propto \frac{p(\boldsymbol{X}_{0:k-1}^i|\boldsymbol{Y}_{1:k-1})p(\boldsymbol{X}_k^i|\boldsymbol{X}_{k-1}^i)p(\boldsymbol{Y}_k|\boldsymbol{X}_k^i)}{q(\boldsymbol{X}_{0:k-1}^i|\boldsymbol{Y}_{1:k-1})q(\boldsymbol{X}_k^i|\boldsymbol{X}_{0:k-1}^i,\boldsymbol{Y}_{1:k})} \\ &\propto \widetilde{w}_{k-1}^i \frac{p(\boldsymbol{Y}_k|\boldsymbol{X}_k^i)p(\boldsymbol{X}_k^i|\boldsymbol{X}_{k-1}^i)}{q(\boldsymbol{X}_k^i|\boldsymbol{X}_{k-1}^i,\boldsymbol{Y}_k)} \end{aligned} \tag{9}$$

It is often convenient to choose the density $q(\boldsymbol{X}_k|\boldsymbol{X}_{k-1},\boldsymbol{Y}_k)$ to be the transition distribution,

$$q(\boldsymbol{X}_k|\boldsymbol{X}_{k-1},\boldsymbol{Y}_k) = p(\boldsymbol{X}_k|\boldsymbol{X}_{k-1}) \tag{10}$$

which yields

$$w_k^i \propto \widetilde{w}_{k-1}^i p(\boldsymbol{Y}_k|\boldsymbol{X}_k^i) \tag{11}$$

Finally, we can obtain the posterior PDF by [34,35]

$$\begin{aligned} p(\boldsymbol{X}_k|\boldsymbol{Y}_{1:k}) &= \int p(\boldsymbol{X}_{0:k}|\boldsymbol{Y}_{1:k})d\boldsymbol{X}_{0:k-1} \\ &\approx \int \sum_{i=1}^N \widetilde{w}_k^i \delta(\boldsymbol{X}_k - \boldsymbol{X}_k^i, \boldsymbol{X}_{0:k-1} - \boldsymbol{X}_{0:k-1}^i)d\boldsymbol{X}_{0:k-1} \\ &= \int \sum_{i=1}^N \widetilde{w}_k^i \delta(\boldsymbol{X}_k - \boldsymbol{X}_k^i)\delta(\boldsymbol{X}_{0:k-1} - \boldsymbol{X}_{0:k-1}^i)d\boldsymbol{X}_{0:k-1} \\ &= \sum_{i=1}^N \widetilde{w}_k^i \delta(\boldsymbol{X}_k - \boldsymbol{X}_k^i) \end{aligned} \tag{12}$$

Eqs. (8) ~ (12) give rise to the universally known sequential importance sampling (SIS) algorithm, which, however, is shown to suffer from particle degeneracy, i.e., most particles are assigned with a trivial weight after a few time steps. This phenomenon can be evaluated by the effective sample size N_{eff} as

$$N_{eff,k} = \frac{1}{\sum_{i=1}^N \left(\widetilde{w}_k^i\right)^2} \tag{13}$$

Once N_{eff} falls below a predefined threshold N_T, a resampling technique is adopted to eliminate the small-weight particles and duplicate the large-weight particles. A standard PF typically has a resampling step inserted into an SIS scheme [36,37], and its pseudo-code is given in Table 2.

The PF algorithm is widely recognized as an efficient approach for many nonlinear and non-Gaussian engineering problems. However, its practical application is often limited to low-dimensional state spaces due to the curse of dimensionality, which requires the number of particles to increase exponentially with the dimension of the state vector [20]. To overcome this challenge, one possible solution is to partition the state vector into several subgroups and then use a separate PF for each subgroup, such as that performed in multiple particle filter (MPF) [24,25].

Table 2
Pseudo-code of particle filter.

For $k = 0$	
	Initialization: draw N particles $\{\boldsymbol{X}_k^i : i = 1, 2, \ldots, N\}$ from the known $p(\boldsymbol{X}_0)$.
For $k = 1, 2, \ldots$	
	Prediction: draw N particles $\{\boldsymbol{X}_k^i : i = 1, 2, \ldots, N\}$ from $p(\boldsymbol{X}_k|\boldsymbol{X}_{k-1})$.
	Weight update: calculate the particle weight and the effective sample size $N_{eff,k}$ by Eqs. (11) and (13), respectively.
	If $N_{eff,k}/N$ is below the threshold N_T
	Resample for $\{\boldsymbol{X}_k^i : i = 1, 2, \ldots, N\}$ based on the normalized weights $\left\{\widetilde{w}_k^i : i = 1, 2, \ldots, N\right\}$ and then define each weight as $1/N$.
	Approximate the mean $\widehat{\boldsymbol{X}}_k$ by $\widehat{\boldsymbol{X}}_k = \sum_{i=1}^N \widetilde{w}_k^i \boldsymbol{X}_k^i$.

2.2. Multiple particle filter

MPF was first proposed in [24,25] for multiple object tracking and then detailed in [23]. The procedure of MPF will be briefly given here, and the interested reader may refer to [23] for more details of the equations. First, a whole state vector $\boldsymbol{X}_k$ can be partitioned into b state subgroups as

$$\boldsymbol{X}_k = \begin{bmatrix} \boldsymbol{x}_k^1 \\ \boldsymbol{x}_k^2 \\ \vdots \\ \boldsymbol{x}_k^b \end{bmatrix} \tag{14}$$

where $\boldsymbol{x}_k^j$ denotes the j-th subgroup, i.e., a vector of state components from the j-th target. Each subgroup has the same state component as the others.

MPF always assumes each state subgroup to independently evolve from the rest, i.e.,

$$\boldsymbol{x}_k^j \sim p\left(\boldsymbol{x}_k^j | \boldsymbol{x}_{k-1}^j\right) \tag{15}$$

Then, by resorting to the assumption of posterior independence between each subgroup, it is possible to write:

$$\begin{aligned} p(\boldsymbol{X}_k|\boldsymbol{Y}_{1:k}) &= p\left(\boldsymbol{x}_k^1|\boldsymbol{Y}_{1:k}, \boldsymbol{x}_k^2, ..., \boldsymbol{x}_k^b\right) p\left(\boldsymbol{x}_k^2, \boldsymbol{x}_k^3, ..., \boldsymbol{x}_k^b|\boldsymbol{Y}_{1:k}\right) \\ &= p\left(\boldsymbol{x}_k^1|\boldsymbol{Y}_{1:k}\right) p\left(\boldsymbol{x}_k^2, \boldsymbol{x}_k^3, ..., \boldsymbol{x}_k^b|\boldsymbol{Y}_{1:k}\right) \\ &= \prod_{j=1}^{b} p\left(\boldsymbol{x}_k^j|\boldsymbol{Y}_{1:k}\right) \end{aligned} \tag{16}$$

It should be noted that, although the posterior probability density function (PDF) of each subgroup is not generally independent with the posterior PDFs of the other subgroups due to the correlations arising from the measurement equation, this assumption has been shown to be beneficial in high-dimensional particle filters [23]. Therefore, we adopt this assumption in this section, as well as in the subsequent analyses.

The estimation of the posterior PDF for a whole state vector can be consequently simplified into that for multiple subgroups. Take the j-th subgroup as a reference example. Let us define:

$$\boldsymbol{X}_k^{-j} = \begin{bmatrix} \boldsymbol{x}_k^1 \\ \vdots \\ \boldsymbol{x}_k^{j-1} \\ \boldsymbol{x}_k^{j+1} \\ \vdots \\ \boldsymbol{x}_k^b \end{bmatrix} \tag{17}$$

The posterior PDF $p\left(\boldsymbol{x}_k^j|\boldsymbol{Y}_{1:k}\right)$ can be then obtained through a Bayesian approach as

$$\begin{aligned} p\left(\boldsymbol{x}_k^j|\boldsymbol{Y}_{1:k}\right) &= \int p(\boldsymbol{X}_k|\boldsymbol{Y}_{1:k}) d\boldsymbol{X}_k^{-j} \\ &\propto \int p(\boldsymbol{Y}_k|\boldsymbol{X}_k) p\left(\boldsymbol{X}_k^{-j}|\boldsymbol{Y}_{1:k-1}\right) p\left(\boldsymbol{x}_k^j|\boldsymbol{X}_k^{-j}, \boldsymbol{Y}_{1:k-1}\right) d\boldsymbol{X}_k^{-j} \\ &= p\left(\boldsymbol{x}_k^j|\boldsymbol{Y}_{1:k-1}\right) \int p(\boldsymbol{Y}_k|\boldsymbol{X}_k) p\left(\boldsymbol{X}_k^{-j}|\boldsymbol{Y}_{1:k-1}\right) d\boldsymbol{X}_k^{-j} \end{aligned} \tag{18}$$

Eqs. (18) is difficult to be analytically calculated in a nonlinear and non-Gaussian problem. Thus, similarly to what is performed for PF, the sequential importance sampling is used to estimate the posterior PDF $p\left(\boldsymbol{x}_{0:k}^j|\boldsymbol{Y}_{1:k}\right)$ through N particles drawn from the importance density $q\left(\boldsymbol{x}_{0:k}^j|\boldsymbol{Y}_{1:k}\right)$, i.e.,

$$p\left(\boldsymbol{x}_{0:k}^j|\boldsymbol{Y}_{1:k}\right) \approx \sum_{i=1}^{N} \widetilde{w}_k^{j,i} \delta\left(\boldsymbol{x}_{0:k}^j - \boldsymbol{x}_{0:k}^{j,i}\right) \tag{19}$$

where $\widetilde{w}_k^{j,i}$ is the normalized form of particle weight $\widetilde{w}_k^{j,i}$ corresponding to the i-th particle for the j-th state subgroup.

The sequential calculation of the particle weights $\widetilde{w}_k^{j,i}$ is demonstrated as follows. First, we have

$$\begin{aligned} p(\boldsymbol{x}_{0:k}^{j}|\boldsymbol{Y}_{1:k}) &= \int p(\boldsymbol{X}_{0:k}|\boldsymbol{Y}_{1:k})d\boldsymbol{X}_{0:k}^{-j} \\ &\propto \int p(\boldsymbol{Y}_k|\boldsymbol{X}_k)p(\boldsymbol{X}_{0:k}|\boldsymbol{Y}_{1:k-1})d\boldsymbol{X}_{0:k}^{-j} \\ &= \int p(\boldsymbol{Y}_k|\boldsymbol{X}_k)p(\boldsymbol{x}_{0:k}^{j}|\boldsymbol{X}_{0:k}^{-j},\boldsymbol{Y}_{1:k-1})p(\boldsymbol{X}_{0:k}^{-j}|\boldsymbol{Y}_{1:k-1})d\boldsymbol{X}_{0:k}^{-j} \end{aligned} \tag{20}$$

With the assumption of independent state evolution and posterior distribution between each subgroup, we can have

$$p(\boldsymbol{x}_{0:k}^{j}|\boldsymbol{X}_{0:k}^{-j},\boldsymbol{Y}_{1:k-1}) = p(\boldsymbol{x}_{0:k}^{j}|\boldsymbol{Y}_{1:k-1}) \tag{21}$$

Eq. (20) then becomes

$$\begin{aligned} p(\boldsymbol{x}_{0:k}^{j}|\boldsymbol{Y}_{1:k}) &\propto p(\boldsymbol{x}_{0:k}^{j}|\boldsymbol{Y}_{1:k-1})\int p(\boldsymbol{Y}_k|\boldsymbol{X}_k)p(\boldsymbol{X}_{0:k}^{-j}|\boldsymbol{Y}_{1:k-1})d\boldsymbol{X}_{0:k}^{-j} \\ &= p(\boldsymbol{x}_k^{j}|\boldsymbol{x}_{k-1}^{j})p(\boldsymbol{x}_{0:k-1}^{j}|\boldsymbol{Y}_{1:k-1})\int p(\boldsymbol{Y}_k|\boldsymbol{X}_k)\int p(\boldsymbol{X}_{0:k-1}^{-j},\boldsymbol{X}_k^{-j}|\boldsymbol{Y}_{1:k-1})d\boldsymbol{X}_{0:k-1}^{-j}d\boldsymbol{X}_k^{-j} \\ &= p(\boldsymbol{x}_k^{j}|\boldsymbol{x}_{k-1}^{j})p(\boldsymbol{x}_{0:k-1}^{j}|\boldsymbol{Y}_{1:k-1})\int p(\boldsymbol{Y}_k|\boldsymbol{X}_k)p(\boldsymbol{X}_k^{-j}|\boldsymbol{Y}_{1:k-1})d\boldsymbol{X}_k^{-j} \end{aligned} \tag{22}$$

The authors in [23] propose that each PF has to approximate the prior PDF $p\left(\boldsymbol{X}_k^{-j}|\boldsymbol{Y}_{1:k-1}\right)$ through the Dirac function as

$$p(\boldsymbol{X}_k^{-j}|\boldsymbol{Y}_{1:k-1}) \approx \delta(\boldsymbol{X}_k^{-j} - \widehat{\boldsymbol{X}}_k^{-j}) \tag{23}$$

where

$$\widehat{\boldsymbol{X}}_k^{-j} = \begin{bmatrix} \sum_{i=1}^{N} \widetilde{w}_{k-1}^{1,i}\boldsymbol{x}_k^{1,i} \\ \vdots \\ \sum_{i=1}^{N} \widetilde{w}_{k-1}^{j-1,i}\boldsymbol{x}_k^{j-1,i} \\ \sum_{i=1}^{N} \widetilde{w}_{k-1}^{j+1,i}\boldsymbol{x}_k^{j+1,i} \\ \vdots \\ \sum_{i=1}^{N} \widetilde{w}_{k-1}^{b,i}\boldsymbol{x}_k^{b,i} \end{bmatrix} \tag{24}$$

Eq. (22) can then be written as

$$p(\boldsymbol{x}_{0:k}^{j}|\boldsymbol{Y}_{1:k}) \propto p(\boldsymbol{x}_{0:k-1}^{j}|\boldsymbol{Y}_{1:k-1})p(\boldsymbol{x}_k^{j}|\boldsymbol{x}_{k-1}^{j})p(\boldsymbol{Y}_k|\boldsymbol{x}_k^{j},\widehat{\boldsymbol{X}}_k^{-j}) \tag{25}$$

which yields

$$\begin{aligned} w_k^{j,i} &\propto \frac{p(\boldsymbol{x}_{0:k}^{j,i}|\boldsymbol{Y}_{1:k})}{q(\boldsymbol{x}_{0:k}^{j,i}|\boldsymbol{Y}_{1:k})} \\ &\propto \frac{p(\boldsymbol{x}_{0:k-1}^{j,i}|\boldsymbol{Y}_{1:k-1})p(\boldsymbol{x}_k^{j,i}|\boldsymbol{x}_{k-1}^{j,i})p(\boldsymbol{Y}_k|\boldsymbol{x}_k^{j,i},\widehat{\boldsymbol{X}}_k^{-j})}{q(\boldsymbol{x}_{0:k-1}^{j,i}|\boldsymbol{Y}_{1:k-1})q(\boldsymbol{x}_k^{j,i}|\boldsymbol{x}_{0:k-1}^{j,i},\boldsymbol{Y}_{1:k})} \\ &\propto \widetilde{w}_{k-1}^{j,i}\frac{p(\boldsymbol{Y}_k|\boldsymbol{x}_k^{j,i},\widehat{\boldsymbol{X}}_k^{-j})p(\boldsymbol{x}_k^{j,i}|\boldsymbol{x}_{k-1}^{j,i})}{q(\boldsymbol{x}_k^{j,i}|\boldsymbol{x}_{k-1}^{j,i},\boldsymbol{Y}_k)} \end{aligned} \tag{26}$$

With the importance density set as the prior distribution,

$$q(\boldsymbol{x}_k^{j}|\boldsymbol{x}_{k-1}^{j},\boldsymbol{Y}_k) = p(\boldsymbol{x}_k^{j}|\boldsymbol{x}_{k-1}^{j}) \tag{27}$$

the particle weight for each subgroup can be sequentially calculated as

$$w_k^{j,i} \propto \widetilde{w}_{k-1}^{j,i}p(\boldsymbol{Y}_k|\boldsymbol{x}_k^{j,i},\widehat{\boldsymbol{X}}_k^{-j}) \tag{28}$$

Finally, the posterior PDF $p\left(\boldsymbol{x}_k^{j}|\boldsymbol{Y}_{1:k}\right)$ can be similarly obtained through Eq. (12). One resampling step is implemented for each

Table 3
Pseudo-code of multiple particle filter.

For $k = 0$	
	Initialization: same as that in PF.
For $k = 1, 2, \ldots$	
	For each subgroup, independently **do**
	Prediction: draw N particles $\left\{\boldsymbol{x}_k^{j,i} : i = 1, 2, \ldots, N\right\}$ by Eq. (15).
	Mean calculation: $\widehat{\boldsymbol{x}}_k^j = \sum_{i=1}^N \widetilde{\boldsymbol{w}}_{k-1}^{j,i} \boldsymbol{x}_k^{j,i}$.
	For each subgroup, independently **do**
	Weight update: calculate the particle weight and the effective sample size $N_{eff,k}^j$ by Eqs. (28) and (13), respectively.
	Resample for $\left\{\boldsymbol{x}_k^{j,i} : i = 1, 2, \ldots, N\right\}$: same as that in PF.
	Approximate the mean $\widehat{\boldsymbol{x}}_k^j$ by $\widehat{\boldsymbol{x}}_k^j = \sum_{i=1}^N \widetilde{\boldsymbol{w}}_k^{j,i} \boldsymbol{x}_k^{j,i}$.

subgroup. The above summarizes the standard MPF algorithm [24,25], as given in Table 3. The effectiveness of MPF compared to PF has been demonstrated in multiple object tracking, as it simplifies the identification of the whole state vector into several lower-dimensional subgroups. However, this algorithm has two limitations.

First, MPF assumes that the evolution of each state subgroup must be independent from the others. However, in many SHM applications, we always encounter scenarios where state evolution is interdependent. Let us consider an application case of estimating the hysteretic displacement, velocity, and stiffness in a multiple-story Bouc-Wen frame. When following the state partitioning strategy from MPF [24,25], a state vector containing multiple degrees of freedom (DOFs) can be partitioned into several subgroups, each consisting of components (i.e., stiffness, velocity, and displacement) from a single DOF. The displacement of each story (except the first story) has to be calculated by the velocity from that story and its lower story, violating the assumption of independent state evolution. Second, the likelihood for each state subgroup is calculated using all state and measurement components, as shown in Eq. (28), which can be computationally expensive [30] and requires modifying the likelihood function to include fewer state and measurement components to improve efficiency.

2.3. Extension about dependent state evolution

Let us consider the aforementioned case of dependent state evolution, i.e., the j-th subgroup $\boldsymbol{x}^j$ has to evolve by itself and a vector of some additional components (defined as $\boldsymbol{t}^j$ in this study) as

$$\boldsymbol{x}_k^j = f^j\left(\boldsymbol{x}_{k-1}^j, \boldsymbol{t}_{k-1}^j, \boldsymbol{u}_{k-1}, \boldsymbol{\omega}_{k-1}^j\right) \tag{29}$$

The first contribution of this study is to extend the MPF method to accommodate the above scenario. The N particles at the prediction step can also be sampled from

$$\boldsymbol{x}_k^j \sim p\left(\boldsymbol{x}_k^j | \boldsymbol{x}_{k-1}^j, \widehat{\boldsymbol{t}}_{k-1}^j\right) \tag{30}$$

where $\widehat{\boldsymbol{t}}^j$ are the means of the samples for those state components.

MPF in the case of dependent state evolution is given below. The prior PDF of each state subgroup $p\left(\boldsymbol{x}_{0:k}^j | \boldsymbol{X}_{0:k}^{-j}, \boldsymbol{Y}_{1:k-1}\right)$ is related to the measurements and the other state subgroups. However, provided that each subgroup in a high-dimensional system may have a limited dependence on the other state subgroups, we can still assume that its prior distribution highly depends on the measurements rather than the other subgroups, which is considered beneficial in developing a high-dimensional PF. As a consequence, Eq. (21) is considered sound. Then, the prior PDF $p\left(\boldsymbol{x}_{0:k}^j | \boldsymbol{Y}_{1:k-1}\right)$ can be obtained by

$$\begin{aligned}
p\left(\boldsymbol{x}_{0:k}^j | \boldsymbol{Y}_{1:k-1}\right) &= \int p\left(\boldsymbol{x}_k^j, \boldsymbol{x}_{0:k-1}^j, \boldsymbol{t}_{k-1}^j | \boldsymbol{Y}_{1:k-1}\right) d\boldsymbol{t}_{k-1}^j \\
&= \int p\left(\boldsymbol{x}_k^j | \boldsymbol{Y}_{1:k-1}, \boldsymbol{t}_{k-1}^j, \boldsymbol{x}_{0:k-1}^j\right) p\left(\boldsymbol{x}_{0:k-1}^j, \boldsymbol{t}_{k-1}^j | \boldsymbol{Y}_{1:k-1}\right) d\boldsymbol{t}_{k-1}^j \\
&= \int p\left(\boldsymbol{x}_k^j | \boldsymbol{t}_{k-1}^j, \boldsymbol{x}_{k-1}^j\right) p\left(\boldsymbol{t}_{k-1}^j | \boldsymbol{x}_{0:k-1}^j, \boldsymbol{Y}_{1:k-1}\right) p\left(\boldsymbol{x}_{0:k-1}^j | \boldsymbol{Y}_{1:k-1}\right) d\boldsymbol{t}_{k-1}^j \\
&= p\left(\boldsymbol{x}_{0:k-1}^j | \boldsymbol{Y}_{1:k-1}\right) \int p\left(\boldsymbol{x}_k^j | \boldsymbol{t}_{k-1}^j, \boldsymbol{x}_{k-1}^j\right) \delta\left(\boldsymbol{t}_{k-1}^j - \widehat{\boldsymbol{t}}_{k-1}^j\right) d\boldsymbol{t}_{k-1}^j \\
&= p\left(\boldsymbol{x}_{0:k-1}^j | \boldsymbol{Y}_{1:k-1}\right) p\left(\boldsymbol{x}_k^j | \widehat{\boldsymbol{t}}_{k-1}^j, \boldsymbol{x}_{k-1}^j\right)
\end{aligned} \tag{31}$$

Provided with Eqs. (21) and (31), Eq. (20) then becomes

$$\begin{aligned} p\left(\boldsymbol{x}_{0:k}^{j}|\boldsymbol{Y}_{1:k}\right) &\propto p\left(\boldsymbol{x}_{0:k-1}^{j}|\boldsymbol{Y}_{1:k-1}\right)p\left(\boldsymbol{x}_{k}^{j}|\widehat{\boldsymbol{t}}_{k-1}^{j},\boldsymbol{x}_{k-1}^{j}\right)\int p(\boldsymbol{Y}_{k}|\boldsymbol{X}_{k})p\left(\boldsymbol{X}_{0:k}^{-j}|\boldsymbol{Y}_{1:k-1}\right)d\boldsymbol{X}_{0:k}^{-j} \\ &= p\left(\boldsymbol{x}_{0:k-1}^{j}|\boldsymbol{Y}_{1:k-1}\right)p\left(\boldsymbol{x}_{k}^{j}|\widehat{\boldsymbol{t}}_{k-1}^{j},\boldsymbol{x}_{k-1}^{j}\right)p\left(\boldsymbol{Y}_{k}|\boldsymbol{x}_{k}^{j},\widehat{\boldsymbol{X}}_{k}^{-j}\right) \end{aligned} \tag{32}$$

Following Eqs. (26) ~ (28), the particle weight can be sequentially calculated as

$$\begin{aligned} w_{k}^{j,i} &\propto \frac{p\left(\boldsymbol{x}_{0:k}^{j,i}|\boldsymbol{Y}_{1:k}\right)}{q\left(\boldsymbol{x}_{0:k}^{j,i}|\boldsymbol{Y}_{1:k}\right)} \\ &\propto \widetilde{w}_{k-1}^{j,i}\frac{p\left(\boldsymbol{Y}_{k}|\boldsymbol{x}_{k}^{j,i},\widehat{\boldsymbol{X}}_{k}^{-j}\right)p\left(\boldsymbol{x}_{k}^{j,i}|\boldsymbol{x}_{k-1}^{j,i},\widehat{\boldsymbol{t}}_{k-1}^{j}\right)}{q\left(\boldsymbol{x}_{k}^{j,i}|\boldsymbol{x}_{k-1}^{j,i},\boldsymbol{Y}_{k}\right)} \\ &= \widetilde{w}_{k-1}^{j,i}p\left(\boldsymbol{Y}_{k}|\boldsymbol{x}_{k}^{j,i},\widehat{\boldsymbol{X}}_{k}^{-j}\right) \end{aligned} \tag{33}$$

where the importance density $q\left(\boldsymbol{x}_{k}^{j,i}|\boldsymbol{x}_{k-1}^{j,i},\boldsymbol{Y}_{k}\right)$ is set as $p\left(\boldsymbol{x}_{k}^{j,i}|\boldsymbol{x}_{k-1}^{j,i},\widehat{\boldsymbol{t}}_{k-1}^{j}\right)$.

2.4. Novel multiple local particle filter

This Section presents the proposed multiple local particle filter (MLPF), which maintains the state partitioning approach and all the assumptions from the MPF. The decay of correlations property indicates that, as a high-dimensional system like a multiple-DOF structure is spatially distributed over relatively large distances, the conditional distribution of one state subgroup should be highly correlated to the measurements collected at its nearby location [31].

First, let us consider that the j-th state subgroup $\boldsymbol{x}_{k}^{j}$ has a few measurements in its nearby location, then define those measurements as its local measurement vector $\boldsymbol{y}_{k}^{j}$. Then, Eq. (16) is modified accordingly,

$$p(\boldsymbol{X}_{k}|\boldsymbol{Y}_{1:k}) = \prod_{j=1}^{b} p\left(\boldsymbol{x}_{k}^{j}|\boldsymbol{Y}_{1:k-1},\boldsymbol{y}_{k}^{j}\right) \tag{34}$$

The posterior PDF of the j-th state subgroup $p\left(\boldsymbol{x}_{k}^{j}|\boldsymbol{Y}_{1:k-1},\boldsymbol{y}_{k}^{j}\right)$ can be described as

$$\begin{aligned} p\left(\boldsymbol{x}_{k}^{j}|\boldsymbol{Y}_{1:k-1},\boldsymbol{y}_{k}^{j}\right) &= \int p\left(\boldsymbol{x}_{k}^{j},\boldsymbol{q}_{k}^{j}|\boldsymbol{Y}_{1:k-1},\boldsymbol{y}_{k}^{j}\right)d\boldsymbol{q}_{k}^{j} \\ &\propto p\left(\boldsymbol{x}_{k}^{j}|\boldsymbol{Y}_{1:k-1}\right)\int p\left(\boldsymbol{y}_{k}^{j}|\boldsymbol{x}_{k}^{j},\boldsymbol{q}_{k}^{j}\right)p\left(\boldsymbol{q}_{k}^{j}|\boldsymbol{Y}_{1:k-1}\right)d\boldsymbol{q}_{k}^{j} \end{aligned} \tag{35}$$

where $\boldsymbol{q}_{k}^{j}$ is a vector of additional state components required for calculating the likelihood $p\left(\boldsymbol{y}_{k}^{j}|\boldsymbol{x}_{k}^{j},\boldsymbol{q}_{k}^{j}\right)$, and it is defined as the local state vector for the j-th state subgroup. The criterion for selecting the local state and measurement vectors for each state subgroup will be again explained in Sections 2.5, 4.3, and 5.4.

Just like the cases of PF and MPF recalled above, the analytical solution of Eq. (35) can hardly be obtained. The sequential importance sampling is therefore used to estimate the posterior PDF $p\left(\boldsymbol{x}_{0:k}^{j}|\boldsymbol{Y}_{1:k-1},\boldsymbol{y}_{k}^{j}\right)$ as

$$p\left(\boldsymbol{x}_{0:k}^{j}|\boldsymbol{Y}_{1:k-1},\boldsymbol{y}_{k}^{j}\right) \approx \sum_{i=1}^{N}\widetilde{w}^{j,i}\delta\left(\boldsymbol{x}_{0:k}^{j}-\boldsymbol{x}_{0:k}^{j,i}\right) \tag{36}$$

Following similar assumptions and procedures in Eqs. (20) ~ (25), we have

$$\begin{aligned} p\left(\boldsymbol{x}_{0:k}^{j}|\boldsymbol{Y}_{1:k-1},\boldsymbol{y}_{k}^{j}\right) &= \int p\left(\boldsymbol{x}_{0:k}^{j},\boldsymbol{q}_{0:k}^{j}|\boldsymbol{Y}_{1:k-1},\boldsymbol{y}_{k}^{j}\right)d\boldsymbol{q}_{0:k}^{j} \\ &\propto \int p\left(\boldsymbol{y}_{k}^{j}|\boldsymbol{x}_{k}^{j},\boldsymbol{q}_{k}^{j}\right)p\left(\boldsymbol{x}_{0:k}^{j},\boldsymbol{q}_{0:k}^{j}|\boldsymbol{Y}_{1:k-1}\right)d\boldsymbol{q}_{0:k}^{j} \\ &= p\left(\boldsymbol{x}_{0:k}^{j}|\boldsymbol{Y}_{1:k-1}\right)\int p\left(\boldsymbol{y}_{k}^{j}|\boldsymbol{x}_{k}^{j},\boldsymbol{q}_{k}^{j}\right)p\left(\boldsymbol{q}_{0:k}^{j}|\boldsymbol{Y}_{1:k-1}\right)d\boldsymbol{q}_{0:k}^{j} \\ &= p\left(\boldsymbol{x}_{0:k}^{j}|\boldsymbol{Y}_{1:k-1}\right)\int p\left(\boldsymbol{y}_{k}^{j}|\boldsymbol{x}_{k}^{j},\boldsymbol{q}_{k}^{j}\right)\int p\left(\boldsymbol{q}_{0:k-1}^{j},\boldsymbol{q}_{k}^{j}|\boldsymbol{Y}_{1:k-1}\right)d\boldsymbol{q}_{0:k-1}^{j}d\boldsymbol{q}_{k}^{j} \\ &= p\left(\boldsymbol{x}_{k}^{j}|\boldsymbol{x}_{k-1}^{j}\right)p\left(\boldsymbol{x}_{0:k-1}^{j}|\boldsymbol{Y}_{1:k-1}\right)\int p\left(\boldsymbol{y}_{k}^{j}|\boldsymbol{x}_{k}^{j},\boldsymbol{q}_{k}^{j}\right)p\left(\boldsymbol{q}_{k}^{j}|\boldsymbol{Y}_{1:k-1}\right)d\boldsymbol{q}_{k}^{j} \\ &= p\left(\boldsymbol{x}_{k}^{j}|\boldsymbol{x}_{k-1}^{j}\right)p\left(\boldsymbol{x}_{0:k-1}^{j}|\boldsymbol{Y}_{1:k-2},\boldsymbol{y}_{k-1}^{j}\right)\int p\left(\boldsymbol{y}_{k}^{j}|\boldsymbol{x}_{k}^{j},\boldsymbol{q}_{k}^{j}\right)\delta\left(\boldsymbol{q}_{k}^{j}-\widehat{\boldsymbol{q}}_{k}^{j}\right)d\boldsymbol{q}_{k}^{j} \\ &= p\left(\boldsymbol{x}_{k}^{j}|\boldsymbol{x}_{k-1}^{j}\right)p\left(\boldsymbol{x}_{0:k-1}^{j}|\boldsymbol{Y}_{1:k-2},\boldsymbol{y}_{k-1}^{j}\right)p\left(\boldsymbol{y}_{k}^{j}|\boldsymbol{x}_{k}^{j},\widehat{\boldsymbol{q}}_{k}^{j}\right) \end{aligned} \tag{37}$$

Table 4
Pseudo-code of novel multiple local particle filter.

For $k = 0$	
	Initialization: same as that in PF.
For $k = 1, 2, \ldots$	
	For each subgroup, independently **do**
	Prediction and **Mean calculation**: same as that in MPF.
	For each subgroup, independently **do**
	Weight update: calculate the weight and the effective sample size $N^j_{eff,k}$ by Eqs. (38) and (13), respectively.
	Resample and **Approximate**: same as that in MPF.

Similarly to Eqs. (26) ~ (28), the particle weight can be sequentially calculated as

$$\begin{aligned} w_k^{j,i} &\propto \frac{p\left(\boldsymbol{x}_{0:k}^{j,i}\middle|\boldsymbol{Y}_{1:k-1},\boldsymbol{y}_k^j\right)}{q\left(\boldsymbol{x}_{0:k}^{j,i}\middle|\boldsymbol{Y}_{1:k-1},\boldsymbol{y}_k^j\right)} \\ &\propto \frac{p\left(\boldsymbol{x}_{0:k-1}^{j,i}\middle|\boldsymbol{Y}_{1:k-2},\boldsymbol{y}_{k-1}^j\right)p\left(\boldsymbol{x}_k^{j,i}\middle|\boldsymbol{x}_{k-1}^{j,i}\right)p\left(\boldsymbol{y}_k^j\middle|\boldsymbol{x}_k^{j,i},\widehat{\boldsymbol{q}}_k^j\right)}{q\left(\boldsymbol{x}_{0:k-1}^{j,i}\middle|\boldsymbol{Y}_{1:k-1},\boldsymbol{y}_k^j\right)q\left(\boldsymbol{x}_k^{j,i}\middle|\boldsymbol{x}_{0:k-1}^{j,i},\boldsymbol{Y}_{1:k-1},\boldsymbol{y}_k^j\right)} \\ &= \frac{p\left(\boldsymbol{x}_{0:k-1}^{j,i}\middle|\boldsymbol{Y}_{1:k-2},\boldsymbol{y}_{k-1}^j\right)p\left(\boldsymbol{x}_k^{j,i}\middle|\boldsymbol{x}_{k-1}^{j,i}\right)p\left(\boldsymbol{y}_k^j\middle|\boldsymbol{x}_k^{j,i},\widehat{\boldsymbol{q}}_k^j\right)}{q\left(\boldsymbol{x}_{0:k-1}^{j,i}\middle|\boldsymbol{Y}_{1:k-2},\boldsymbol{y}_{k-1}^j\right)q\left(\boldsymbol{x}_k^{j,i}\middle|\boldsymbol{x}_{k-1}^{j,i},\boldsymbol{y}_k^j\right)} \\ &= w_{k-1}^{j,i}\frac{p\left(\boldsymbol{x}_k^{j,i}\middle|\boldsymbol{x}_{k-1}^{j,i}\right)p\left(\boldsymbol{y}_k^j\middle|\boldsymbol{x}_k^{j,i},\widehat{\boldsymbol{q}}_k^j\right)}{q\left(\boldsymbol{x}_k^{j,i}\middle|\boldsymbol{x}_{k-1}^{j,i},\boldsymbol{y}_k^j\right)} \\ &\propto \widetilde{w}_{k-1}^{j,i}p\left(\boldsymbol{y}_k^j\middle|\boldsymbol{x}_k^{j,i},\widehat{\boldsymbol{q}}_k^j\right) \end{aligned} \tag{38}$$

The posterior PDF $p\left(\boldsymbol{x}_k^j\middle|\boldsymbol{Y}_{1:k-1},\boldsymbol{y}_k^j\right)$ can finally be obtained by Eq. (12), the same as already performed for PF and MPF. In the case where the independent state evolution, i.e., Eq. (15), cannot be satisfied, one can resort to Eq. (30) for calculating the prior, which is similar to what has been performed for MPF, thus not detailed here.

By embedding a novel likelihood $p\left(\boldsymbol{y}_k^j\middle|\boldsymbol{x}_k^{j,i},\widehat{\boldsymbol{q}}_k^j\right)$ into the MPF scheme, multiple local particle filter (MLPF) can be presented as outlined in Table 4. The difference between MPF and MLPF is that the latter uses a reduced number of components in its likelihood, resulting in superior computation efficiency.

2.5. Definition of local state and measurement vectors

The criterion for selecting the local measurement and state vectors for each state subgroup is presented here. Considering that the measurement $\boldsymbol{Y}$ is a d-dimensional vector as

$$\boldsymbol{Y}_k = \begin{bmatrix} y_{1,k} & y_{2,k} & \cdots & y_{d,k} \end{bmatrix}^T \tag{39}$$

the measurement equation Eq. (2) can be then partitioned into d equations as

$$\begin{cases} y_{1,k} = h_1\left(\boldsymbol{p}_{1,k},\boldsymbol{u}_k,\eta_{1,k}\right) \\ y_{2,k} = h_2\left(\boldsymbol{p}_{2,k},\boldsymbol{u}_k,\eta_{2,k}\right) \\ \vdots \\ y_{d,k} = h_d\left(\boldsymbol{p}_{d,k},\boldsymbol{u}_k,\eta_{d,k}\right) \end{cases} \tag{40}$$

where $\boldsymbol{p}_{l,k}$ is a vector of state components required for modeling the l-th equation.

The components of the state subgroup $\boldsymbol{x}_k^j$ exist in one or some rows of Eq. (40), all of which are selected to re-formulate a local measurement equation,

$$\boldsymbol{y}_k^j = h^j\left(\boldsymbol{x}_k^j,\boldsymbol{q}_k^j,\boldsymbol{u}_k,\boldsymbol{\eta}_k^j\right) \tag{41}$$

where $\boldsymbol{q}_k^j$ and $\boldsymbol{y}_k^j$ are the local state and measurement vectors for that subgroup. For example, let us consider the components of the first subgroup $\boldsymbol{x}_k^1$ exist in the first three rows of Eq. (40), i.e.,

$$\begin{cases} y_{1,k} = h_1\left(\boldsymbol{p}_{1,k},\boldsymbol{u}_k,\eta_{1,k}\right) \\ y_{2,k} = h_2\left(\boldsymbol{p}_{2,k},\boldsymbol{u}_k,\eta_{2,k}\right) \\ y_{3,k} = h_3\left(\boldsymbol{p}_{3,k},\boldsymbol{u}_k,\eta_{3,k}\right) \end{cases} \tag{42}$$

Thus, the local measurement vector $\boldsymbol{y}_k^1$ consists of the first three measurements. All the state components from the three vectors $\boldsymbol{p}_{1,k}$,

$\boldsymbol{p}_{2,k}$, and $\boldsymbol{p}_{3,k}$ can be used to build a new vector, which can be then divided into two vectors $\boldsymbol{x}_k^1$ and $\boldsymbol{q}_k^1$. This will be explained by a numerical example in Sections 4.3 and an experimental study in Section 5.4.

When calculating the particle weights for one subgroup, MPF employs a likelihood function that requires the samples from that subgroup and the mean values of the samples from some additional state components. As a result, the formers are expected to have a significant contribution to the distribution of normalized weight compared to the latter. The same observation can be drawn for MLPF, despite using a reduced number of components in its likelihood. Consequently, in comparison to MPF, the proposed MLPF can deliver similarly accurate estimation with an equivalent number of particles, while demanding reduced computational effort in likelihood calculation.

3. Toy example

A toy example taken from [20] is adopted for illustrating the effect of curse of dimensionality on PF-based system identification and comparing the performances of the three algorithms in terms of accuracy and computational time. Its state space model is given as follows,

$$\boldsymbol{X}_k \sim \mathcal{N}(0, \mathrm{I}) \tag{43}$$

$$\boldsymbol{Y}_k = \boldsymbol{X}_k + \boldsymbol{\eta}_k \tag{44}$$

where $\mathcal{N}(0, \mathrm{I})$ denotes the zero-mean Gaussian distribution with an identity matrix I serving as the covariance, and the measurement noise $\boldsymbol{\eta}_k$ is also sampled from the distribution $\mathcal{N}(0, \mathrm{I})$. The state and measurement with a duration of two hundred steps are created through Eqs. (43) and (44). The size of the matrix I is increased from 10 to 90 with step 10, thus resulting in nine sets of target states and measurements for estimation.

For all three algorithms, the importance density is set as $\mathcal{N}(0, \mathrm{I})$, resampling procedure and process noise are not included, and the coefficients in likelihood are defined by the distribution of measurement noise [20]. We consider that both MPF and MLPF use the same state partitioning. Then, we examine the performance of MPF and MLPF under three different subgroup sizes, i.e., one state subgroup consists of two, five, or ten state components. Notably, the assumption of independent state evolution for each subgroup is inherently met in this example. Furthermore, there is no need to define the local state vector in MLPF, because the likelihood function for each subgroup only involves the state components of that subgroup.

The root-mean-square error (RMSE) of the state estimates is taken as the performance metric to evaluate the estimation accuracy,

$$RMSE = \sqrt{\frac{1}{T}\sum_{k=1}^{T}\frac{1}{D_s}\sum_{l=1}^{D_s}\left(\widehat{X}_{l,k} - X_{l,k,true}\right)^2} \tag{45}$$

where X_l denotes the l-th state component of the state vector, and T is the number of total time steps. A lower RMSE means a more accurate estimation.

Similarly to that performed in [20], the number of particles used for each algorithm is determined by

$$\begin{cases} N_{PF} = 10 \times 2^a \\ N_{MPF} = a + 1 \\ N_{MLPF} = a + 1 \end{cases} \tag{46}$$

where the number a is increased from 0 with a step of 1, until the estimates obtained from PF, MPF, or MLPF are no worse than those simply relying on the measurements alone, i.e., the RMSE of the estimates is lower than that of the measurements,

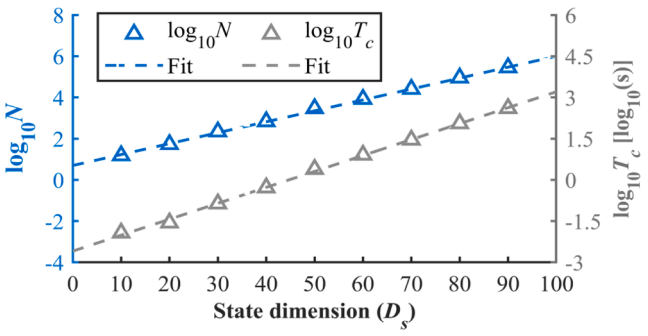
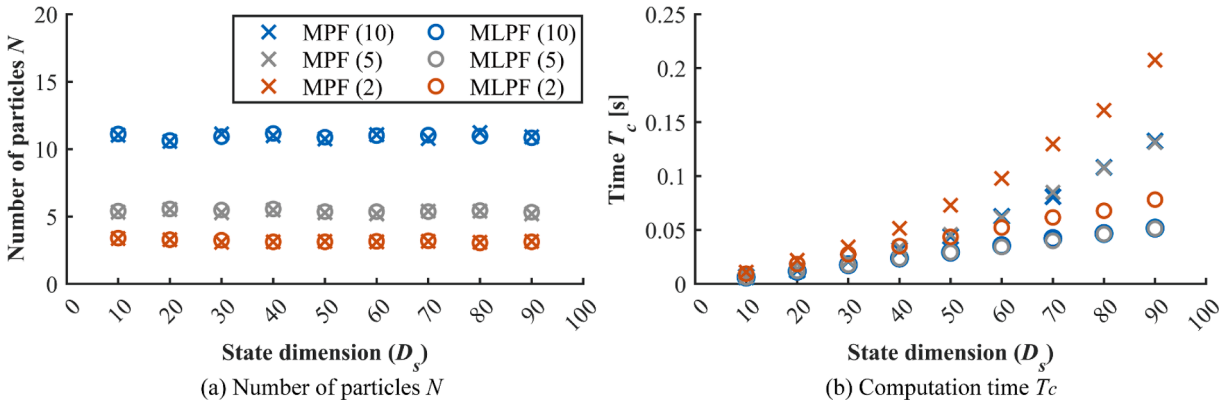


Fig. 1. Numbers of particles and computation times required by PF for satisfying the criterion at increasing state dimensions.



(a) Number of particles N (b) Computation time Tc

Fig. 2. Numbers of particles and computation time required by MPF and MLPF for satisfying the criterion at increasing state dimensions. Note: the numbers '2′, '5′, and '10′ within the brackets denote the number of state components at each state subgroup.

$$\sqrt{\frac{1}{T}\sum_{k=1}^{T}\frac{1}{D_s}\sum_{l=1}^{D_s}\left(\widehat{X}_{l,k}-X_{l,k,true}\right)^2}\leqslant\sqrt{\frac{1}{T}\sum_{k=1}^{T}\frac{1}{D_s}\sum_{l=1}^{D_s}\left(Y_{l,k}-X_{l,k,true}\right)^2} \tag{47}$$

where Y_l denotes the l-th component of the measurement vector.

Fig. 1 plots the numbers of particles and computational times at increasing state dimensions required by PF, averaged from twenty-five independent runs. The computation time and the number of particles can be formulated as a function of state dimension as

$$\begin{cases}\log_{10}N_{PF}=0.053D_s+0.698\\ \log_{10}T_{c,PF}=0.058D_s-2.593\end{cases} \tag{48}$$

which leads to the same conclusion from [20], i.e., the required number of particles and computation time appear to exponentially increase with the state dimension.

Fig. 2 (a) plots the numbers of particles at increasing state dimensions required by MPF and MLPF, averaged from twenty-five independent runs. The values in the brackets, i.e., 2, 5, and 10, indicate the number of state components in each state subgroup. By using state partitioning, both MPF and MLPF require significantly fewer particles than PF. Furthermore, the number of particles remains stable as state dimensions increase, demonstrating the potential of MPF and MLPF for mitigating the curse of dimensionality. Notably, the fewer components a state subgroup possesses, the fewer particles that MPF and MLPF require. At the same time, the performances of MPF and MLPF in terms of accuracy are quite similar, i.e., they can provide similarly accurate estimates with the same number of particles. The reason has been given in Section 2.4 and again explained here. When calculating the particle weights for one subgroup, both MPF and MLPF resort to a likelihood function that requires the samples from that subgroup and the mean values of the samples from some additional state components. Those samples are expected to more significantly contribute to the distribution of normalized weight compared to the mean values.

Fig. 2 (b) plots the average computation times at increasing state dimensions required by MPF and MLPF. MLPF exhibits faster computation time than the MPF, because the former always has fewer state and measurement components adopted for calculating the likelihood for each subgroup. By taking the example of defining the subgroup size as 10, the computation time as a function of state dimension can be formulated as

$$\begin{cases}T_{c,MPF}=1.570\times10^{-5}D_s^2-4.874\times10^{-6}D_s+6.095\times10^{-3}\\ T_{c,MLPF}=5.755\times10^{-4}D_s+6.944\times10^{-4}\end{cases} \tag{49}$$

which suggests that the advantage of the MLPF over the MPF becomes more evident in a higher-dimensional system. This is because the number of components required for likelihood calculation in MPF increases with the state dimension in this example, whereas the MLPF does not experience such a limitation. This demonstrates that MLPF yields similar levels of estimation accuracy with the same number of particles as MPF, while requiring less computation time.

Additionally, when the subgroup size is two, the MPF and the MLPF require more computation time compared to the other cases, because of the increasing number of PFs used at each step. These results suggest that reducing the subgroup size can improve computation efficiency until the subgroup size becomes too small.

4. Numerical study of a twenty-story frame structure

Section 4 intends to validate the extension made for dependent state evolution and the superiority of MLPF over PF and MPF.

Sections 4.1 and 4.2 provide details on the development of dynamic responses for a twenty-story Bouc-Wen frame under the ground motion and state space modeling, respectively. Section 4.3 explains the process of partitioning the state into some subgroups and defining their corresponding local state and measurement vectors. Typical runs of estimating the stiffness, velocity, and displacement from PF, MPF, and MLPF are discussed in Section 4.4, and the comparison in terms of estimation accuracy and computation effort is further elaborated in Section 4.5. Additionally, Section 4.6 presents their performance with a limited number of measurements.

4.1. Data generation

A twenty-story Bouc-Wen frame structure under the ground motion serves as the numerical example for comparing the performances of the three PF algorithms, and its structural dynamic equation can be built as

$$\begin{bmatrix} m_1 & & & \\ & m_2 & & \\ & & \ddots & \\ & & & m_{20} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_{20} \end{bmatrix} + \begin{bmatrix} c_1+c_2 & -c_2 & & \\ -c_2 & c_2+c_3 & -c_3 & \\ & & \ddots & \\ & & -c_{20} & c_{20} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_{20} \end{bmatrix} + \begin{bmatrix} s_1+s_2 & -s_2 & & \\ -s_2 & s_2+s_3 & -s_3 & \\ & & \ddots & \\ & & -s_{20} & s_{20} \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_{20} \end{bmatrix} = -\begin{bmatrix} m_1 \\ m_2 \\ \vdots \\ m_{20} \end{bmatrix} a_g \tag{50}$$

where the subscript l denotes the l-th story, a_g means the ground motion, a, v, c, m, and s mean the acceleration, velocity, damping, mass, and stiffness, respectively, and the hysteretic displacement z can be obtained from

$$\dot{z}_l = (v_l - v_{l-1}) - \beta|v_l - v_{l-1}||z_l|^{n-1} z_l - \gamma(v_l - v_{l-1})|z_l|^n \tag{51}$$

where β, γ, and n are the Bouc-Wen parameters.

For simulating the dynamic responses, the mass, damping, and stiffness for each story are 1 kg, 0.25 Ns/m, and 9 N/m, respectively [5], and the parameters β, γ, and n are 800 m^{-2}, 400 m^{-2}, and 2, respectively [38]. The first five-second El-Centro NS earthquake signal serves as the ground motion a_g $\left[m/s^2\right]$. The structural dynamic responses are calculated at 400 Hz sampling frequency by the MATLAB ordinary differential equation solver 'ode45'. Both the acceleration of each story and the ground motion are assumed to be measured, and they are corrupted by zero-mean Gaussian noises with standard deviations (STDs) equal to 1 % of the root-mean-square (RMS) value of the acceleration of that story and 1 % of the RMS value of the ground motion, respectively.

4.2. State space modeling

With the stiffness, velocity, and hysteretic displacement of each story to be estimated, the state space model can be formulated as

$$\boldsymbol{X}_k = \begin{bmatrix} s_{1,k} \\ \vdots \\ v_{1,k} \\ \vdots \\ z_{1,k} \\ z_{2,k} \\ \vdots \end{bmatrix} = \begin{bmatrix} s_{1,k-1} \\ \vdots \\ v_{1,k-1} \\ \vdots \\ z_{1,k-1} \\ z_{2,k-1} \\ \vdots \end{bmatrix} + \begin{bmatrix} \omega_{1,k-1} \\ \vdots \\ a_{1,k-1}\Delta t + \omega_{21,k-1} \\ \vdots \\ \Delta t\left(v_{1,k-1} - \beta\left|v_{1,k-1}\right|\left|z_{1,k-1}\right|^{n-1} z_{1,k-1} - \gamma v_{1,k-1}\left|z_{1,k-1}\right|^n\right) + \omega_{41,k} \\ \Delta t\left(\left(v_{2,k-1} - v_{1,k-1}\right) - \beta\left|v_{2,k-1} - v_{1,k-1}\right|\left|z_{1,k-1}\right|^{n-1} z_{1,k-1} - \gamma\left(v_{2,k-1} - v_{1,k-1}\right)\left|z_{1,k-1}\right|^n\right) + \omega_{42,k} \\ \vdots \end{bmatrix} \tag{52}$$

$$\boldsymbol{Y}_k = \begin{bmatrix} a_{1,k} \\ \vdots \\ a_{20,k} \end{bmatrix} = -\begin{bmatrix} a_{g,k} \\ \vdots \\ a_{g,k} \end{bmatrix} - \begin{bmatrix} \frac{1}{m_1} & & \\ & \ddots & \\ & & \frac{1}{m_{20}} \end{bmatrix} \left(\begin{bmatrix} c_1+c_2 & -c_2 & \\ & \ddots & \\ & -c_{20} & c_{20} \end{bmatrix} \begin{bmatrix} v_{1,k} \\ \vdots \\ v_{20,k} \end{bmatrix} + \begin{bmatrix} s_{1,k}+s_{2,k} & -s_{2,k} & \\ & \ddots & \\ & -s_{20,k} & s_{20,k} \end{bmatrix} \begin{bmatrix} z_{1,k} \\ \vdots \\ z_{20,k} \end{bmatrix} \right) + \begin{bmatrix} \eta_{1,k} \\ \vdots \\ \eta_{20,k} \end{bmatrix} \tag{53}$$

where Δt is the time interval, $\{\omega_i : i = 1, 2, \ldots, 60\}$ and $\{\eta_i : i = 1, 2, \ldots, 20\}$ are the zero-mean Gaussian process and measurement noises, respectively. The Bouc-Wen components and the multiplication of the unknown stiffness parameter and the displacement result in the nonlinearity of this model [5,6], and the sixty unknown state components to be estimated account for the high dimensionality.

The numbers of particles for the PF, the MPF, and the MLPF are 80000, 800, and 800, respectively. Note that 80,000 particles are approximately the maximum number of particles that the authors' computer could afford. Their other parameters are the same as defined as follows. The threshold N_T for systematic resampling is 0.5 [6]. The samples for initial stiffnesses are randomly selected within the same uniform distribution, i.e., $\{s_{1,0}, s_{2,0}, \ldots, s_{20,0}\} \sim \mathrm{U}(4, 12)$. The initial velocity and displacement are zero. The displacement and velocity are calculated by trapezoidal integration from the measured acceleration. The STDs of process noises for velocity, displacement, and stiffness are 0.01 % of root-mean-square (RMS) values of the calculated velocity and displacement, and 0.001, respectively. The STD for each story in the likelihood is taken as 10 % of the RMS value of the acceleration of that story.

4.3. Definition of state subgroup and local vectors

The application of multiple target tracking considers the state components from one target as one state subgroup. This also applies to a multiple-DOF structure. Specifically, the state components from one DOF, i.e., stiffness, velocity, and displacement from one story, can be taken as one subgroup. Thus, for both the MPF and MLPF, this numerical example has twenty state subgroups,

$$\begin{cases} \boldsymbol{x}_k^1 = [\, s_{1,k} \quad v_{1,k} \quad z_{1,k} \,]^T \\ \boldsymbol{x}_k^2 = [\, s_{2,k} \quad v_{2,k} \quad z_{2,k} \,]^T \\ \qquad\qquad \vdots \\ \boldsymbol{x}_k^{20} = [\, s_{20,k} \quad v_{20,k} \quad z_{20,k} \,]^T \end{cases} \tag{54}$$

As shown by the process equation Eq. (52), the first state subgroup can independently evolve, while the other subgroup needs to evolve by resorting to itself and one extra state component t^j. Let us take the second state subgroup as the reference example. The state evolution of that subgroup can be determined from Eq. (52) as

$$\boldsymbol{x}_k^2 = \begin{bmatrix} s_{2,k} \\ v_{2,k} \\ z_{2,k} \end{bmatrix} = \begin{bmatrix} s_{2,k-1} \\ v_{2,k-1} \\ z_{2,k-1} \end{bmatrix} + \begin{bmatrix} \omega_{2,k-1} \\ a_{2,k-1}\Delta t + \omega_{22,k-1} \\ \Delta t\Big(\big(v_{2,k-1} - v_{1,k-1}\big) - \beta \big| v_{2,k-1} - v_{1,k-1} \big| \big| z_{1,k-1} \big|^{n-1} z_{1,k-1} - \gamma \big(v_{2,k-1} - v_{1,k-1}\big) \big| z_{1,k-1} \big|^n \Big) + \omega_{42,k} \end{bmatrix} \tag{55}$$

and it has to resort to itself and one additional component v_1. The same procedure can be applied to the other state subgroup, and the extra component for the j-th subgroup can be defined as

$$t^j = v_{j-1} \tag{56}$$

The selection of the local state and measurement vectors for each subgroup in MLPF is explained by taking the first two state subgroups $\boldsymbol{x}_k^1$ and $\boldsymbol{x}_k^2$ as the reference example. The measurement equation Eq. (53) can be reformulated as

$$\begin{cases} a_{1,k} = \dfrac{-a_{g,k} - [\, c_1 + c_2 \quad -c_2 \,] \begin{bmatrix} v_{1,k} \\ v_{2,k} \end{bmatrix} - [\, s_{1,k} + s_{2,k} \quad -s_{2,k} \,] \begin{bmatrix} z_{1,k} \\ z_{2,k} \end{bmatrix}}{m_1} + \eta_{1,k} \\ a_{2,k} = \dfrac{-a_{g,k} - [\, -c_2 \quad c_2 + c_3 \quad -c_3 \,] \begin{bmatrix} v_{1,k} \\ v_{2,k} \\ v_{3,k} \end{bmatrix} - [\, -s_{2,k} \quad s_{2,k} + s_{3,k} \quad -s_{3,k} \,] \begin{bmatrix} z_{1,k} \\ z_{2,k} \\ z_{3,k} \end{bmatrix}}{m_2} + \eta_{2,k} \\ \qquad\qquad \vdots \\ a_{20,k} = \dfrac{-a_{g,k} - [\, -c_{20} \quad c_{20} \,] \begin{bmatrix} v_{19,k} \\ v_{20,k} \end{bmatrix} - [\, -s_{20,k} \quad s_{20,k} \,] \begin{bmatrix} z_{19,k} \\ z_{20,k} \end{bmatrix}}{m_{20}} + \eta_{20,k} \end{cases} \tag{57}$$

where the three state components of the subgroup $\boldsymbol{x}_k^1$ exist in the first and second rows only. Thus, the local measurement equation for that subgroup can be obtained as

$$\boldsymbol{y}_k^1 = \begin{bmatrix} a_{1,k} \\ a_{2,k} \end{bmatrix} = \begin{bmatrix} \dfrac{-a_{g,k} - [\, c_1 + c_2 \quad -c_2 \,] \begin{bmatrix} v_{1,k} \\ v_{2,k} \end{bmatrix} - [\, s_{1,k} + s_{2,k} \quad -s_{2,k} \,] \begin{bmatrix} z_{1,k} \\ z_{2,k} \end{bmatrix}}{m_1} + \eta_{1,k} \\ \dfrac{-a_{g,k} - [\, -c_2 \quad c_2 + c_3 \quad -c_3 \,] \begin{bmatrix} v_{1,k} \\ v_{2,k} \\ v_{3,k} \end{bmatrix} - [\, -s_{2,k} \quad s_{2,k} + s_{3,k} \quad -s_{3,k} \,] \begin{bmatrix} z_{1,k} \\ z_{2,k} \\ z_{3,k} \end{bmatrix}}{m_2} + \eta_{2,k} \end{bmatrix} \tag{58}$$

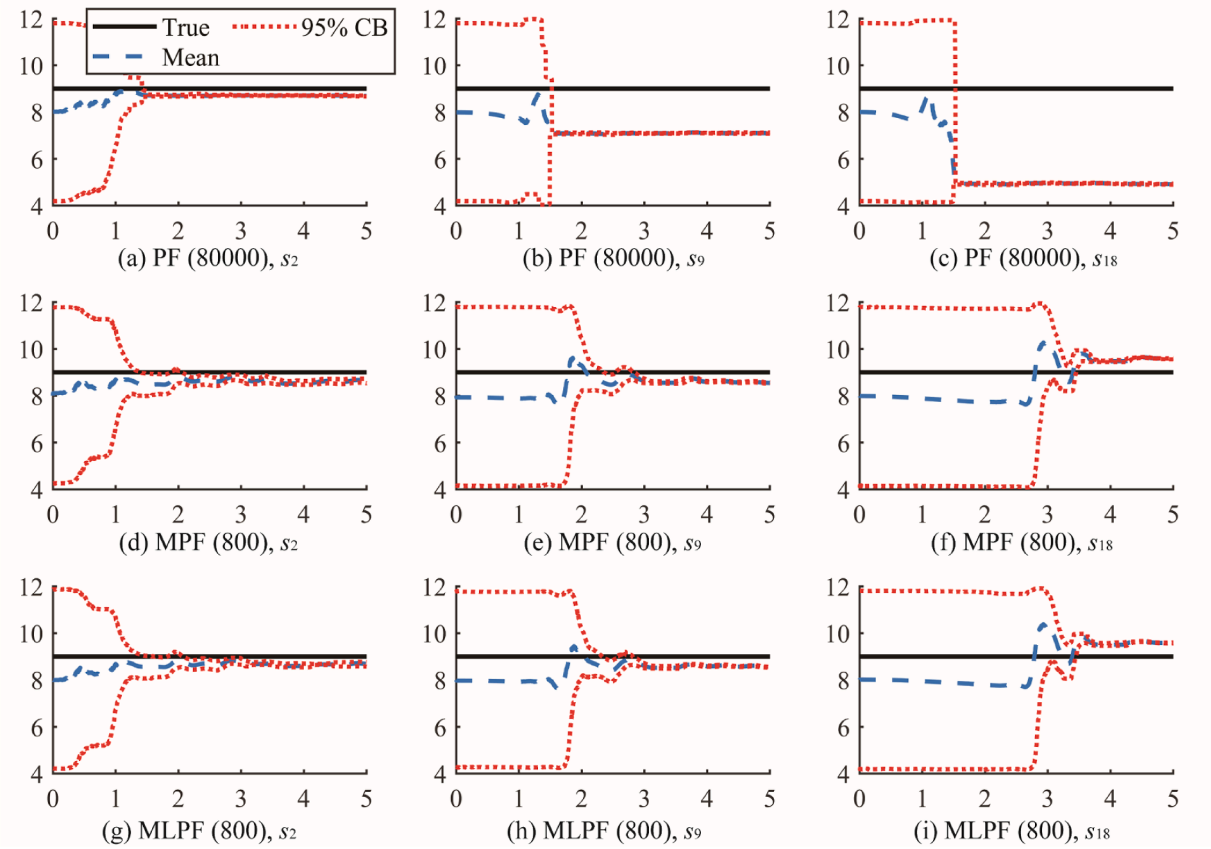


Fig. 3. Stiffness estimations from the three algorithms. Note: (i) the x and y labels are 'time [s]' and 'stiffness [N/m]', respectively; (ii) the number within the bracket and 'CB' means the number of particles used in one run and confidence boundary, respectively.

and its local state and measurement vectors can then be determined as

$$\begin{cases} \boldsymbol{y}_k^1 = [\, a_{1,k} \quad a_{2,k} \,]^T \\ \boldsymbol{q}_k^1 = [\, s_{2,k} \quad v_{2,k} \quad z_{2,k} \quad s_{3,k} \quad v_{3,k} \quad z_{3,k} \,]^T \end{cases} \tag{59}$$

The same selection procedure applies to the subgroup $\boldsymbol{x}_k^2$ to determine its local state and measurement vectors,

$$\begin{cases} \boldsymbol{y}_k^2 = [\, a_{1,k} \quad a_{2,k} \quad a_{3,k} \,]^T \\ \boldsymbol{q}_k^2 = [\, s_{1,k} \quad v_{1,k} \quad z_{1,k} \quad s_{3,k} \quad v_{3,k} \quad z_{3,k} \quad s_{4,k} \quad v_{4,k} \quad z_{4,k} \,]^T \end{cases} \tag{60}$$

4.4. Comparing PF, MPF, and MLPF: A single run

One typical run of estimating the stiffness, velocity, and displacement by each algorithm is discussed in this section. The estimations of the three selected stiffness parameters, namely s_2, s_9, and s_{18}, are shown in Fig. 3, where those parameters converge in about three seconds but with different rates. Additionally, due to the initial acceleration of the higher story remaining close to zero for a longer time, the measurement calculated by its stiffness weakly contributes to the likelihood at initial time steps, resulting in a slower convergence for the stiffness of the higher story compared to the lower story.

In the case of PF, the resampling procedure eliminates the samples of the two stiffness s_9 and s_{18} at the initial steps, with little concern about whether their distributions are around the true values or not. This leads to a quick loss of particle diversity, and consequently, an inaccurate estimation, as shown in Fig. 3 (b) and (c). However, by partitioning one state vector into several state subgroups, both MPF and MLPF algorithms keep the initial diversities of samples for the two parameters better, resulting in more accurate estimates than the PF, as shown in Fig. 3 (e), (f), (h), and (i). This also demonstrates the successful extension of state partition strategy to the case of dependent state evolution.

Additionally, the accuracy of estimations from the MPF and the MLPF is quite similar. This is because the additional components in the likelihood of MPF have a limited effect on the distribution of normalized weight, leading to similar estimation results between the two algorithms.

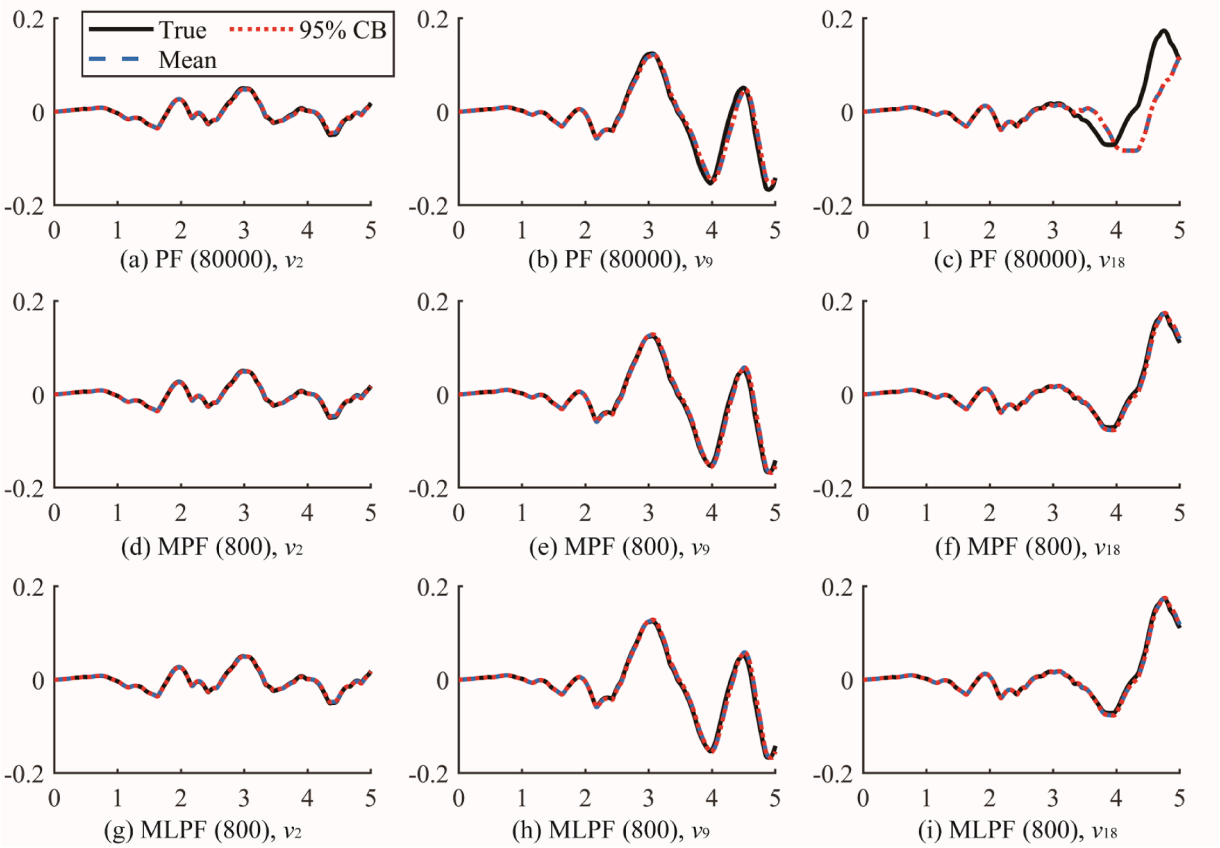


Fig. 4. Velocity estimations from the three algorithms. Note: (i) the x and y labels are 'time [s]' and 'velocity [m/s]', respectively; (ii) the number within the bracket and 'CB' means the number of particles used in one run and confidence boundary, respectively.

Finally, the velocity and displacement estimations in Figs. 4 and 5, respectively, yield the same conclusions as drawn from stiffness estimation, demonstrating the robustness of both the MPF and the MLPF.

4.5. Comparing PF, MPF, and MLPF: Accuracy and computation effort

To provide a more comprehensive comparison of stiffness estimation accuracies and computation efforts from the three algorithms, each algorithm is implemented 25 independent times to provide 25 sets of estimates. The Root-mean-square error (RMSE) and relative error (RE) serve as the performance metrics to evaluate the estimation accuracy of each run,

$$RMSE = \sqrt{\frac{1}{T}\sum_{k=1}^{T}\left(\widehat{s}_k - s_{true,k}\right)^2} \tag{61}$$

$$RE = \frac{\left|\widehat{s}_T - s_{true,T}\right|}{s_{true,T}} \tag{62}$$

where T is the number of time steps in PF implementation. Note that a lower RMSE or RE indicates a more accurate estimation. Fig. 6 presents the RMSE and RE averaged from 25 runs for each algorithm, highlighting the observations made in Figs. 3 - 5.

The computation efforts of PF, MPF, and MLPF are compared here. Table 5 presents the average computation time of one run for each algorithm on an AMD Ryzen 9 3950X 16-Core Processor. Consistent with the observations made in Section 3, both MPF and MLPF require a significantly reduced number of particles (or computation time) compared to PF, while still delivering more accurate estimates. Additionally, MLPF achieves similar levels of accuracy with the same number of particles as MPF, but with fewer components in the likelihood calculation, making it more computationally efficient.

4.6. Identification with limited measurements

The accelerations of all stories were gathered for system identification in the above numerical study. However, in-field applications

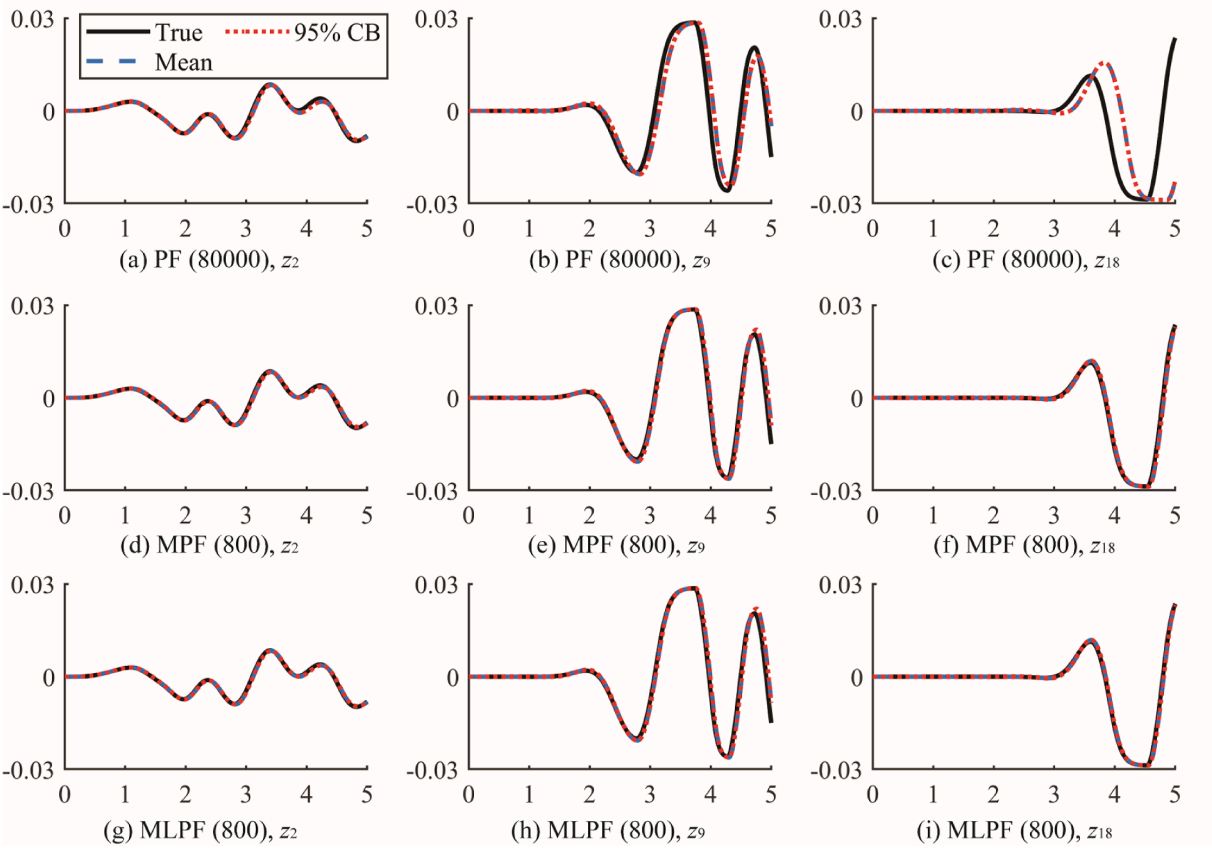


Fig. 5. Displacement estimations from the three algorithms. Note: (i) the *x* and *y* labels are 'time [s]' and 'displacement [m]', respectively; (ii) the number within the bracket and 'CB' means the number of particles used in one run and confidence boundary, respectively.

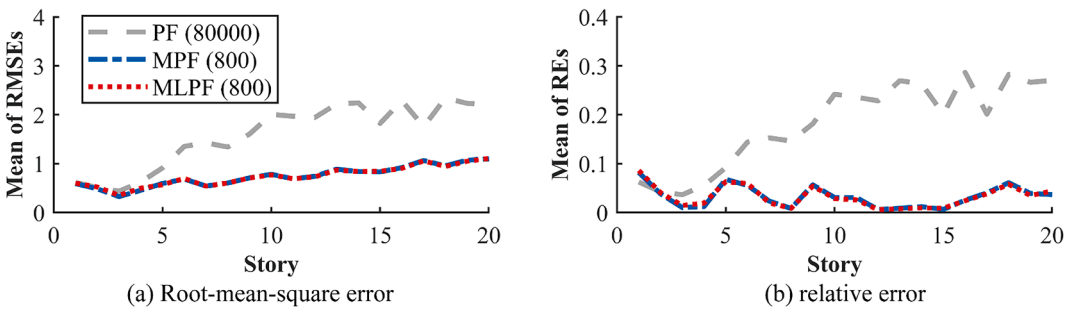


Fig. 6. RMSE and RE averaged from the 25 runs of each algorithm. Note: the number within the bracket means the number of particles used for one run.

Table 5
Average computation time of one run for each algorithm.

	PF	MPF	MLPF
Number of particles	80,000	800	800
Computation time [s]	1354.7	162.1	71.7

may only have a limited number of sensors installed for monitoring. Some Kalman filter algorithms have been adapted to perform system identification with limited measurements [39,40], while there have been few attempts to do the same with PF. Nevertheless, PF, MPF, and MLPF can still function in such scenarios, albeit with a modified likelihood calculation. More specifically, the likelihood

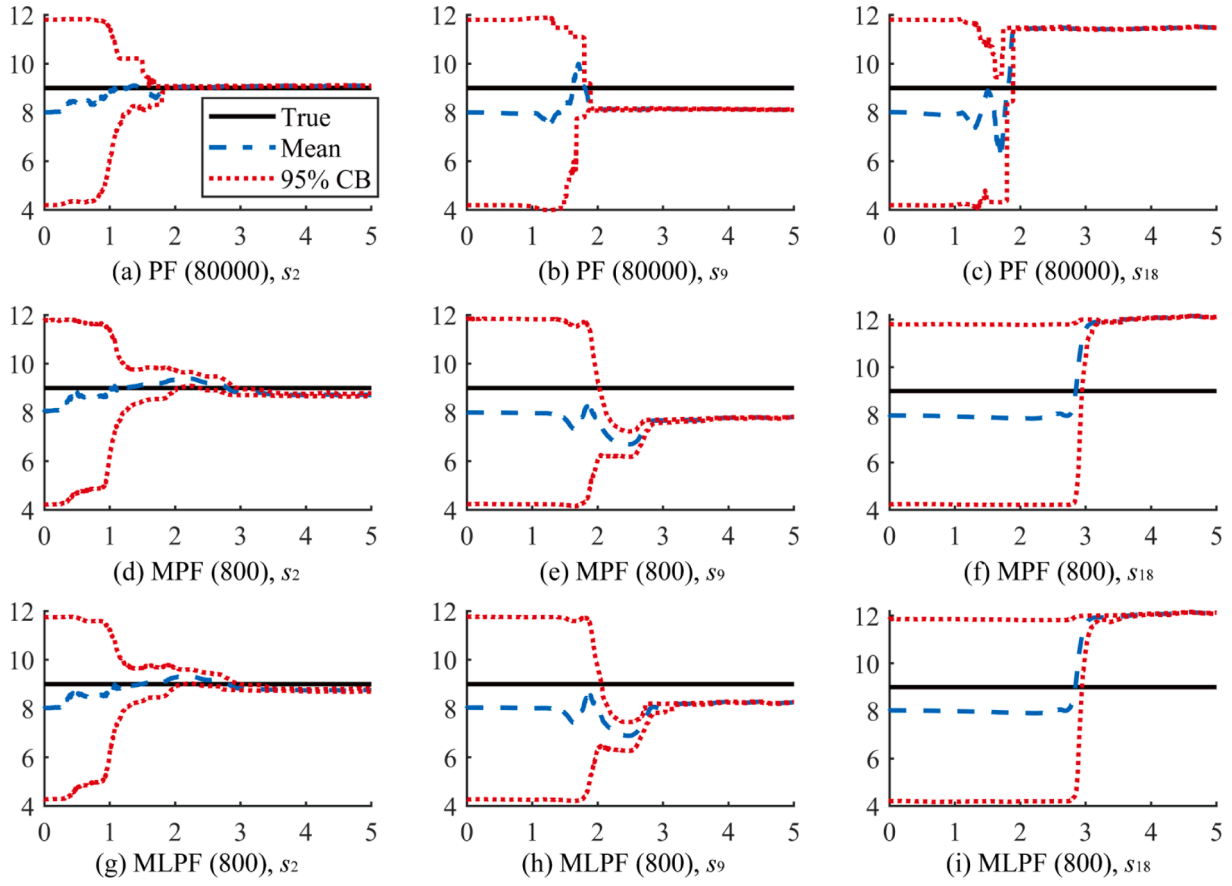


Fig. 7. Stiffness estimations from the three algorithms with limited measurements. Note: (i) the *x* and *y* labels are 'time [s]' and 'stiffness [m/s]', respectively; (ii) the number within the bracket and 'CB' means the number of particles used in one run and confidence boundary, respectively.

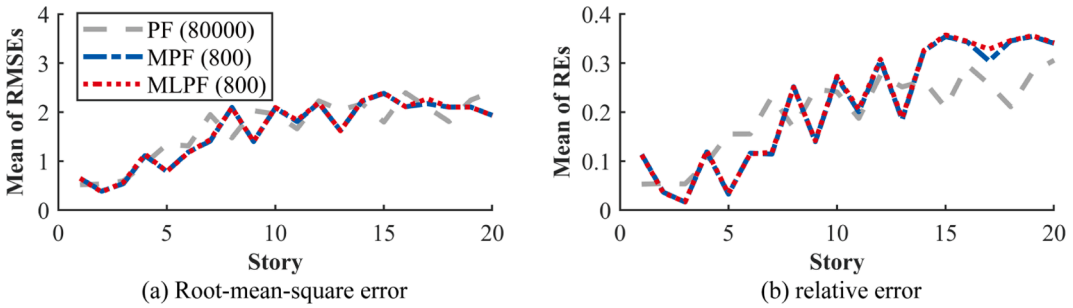


Fig. 8. RMSE and RE averaged from the 25 runs of each algorithm with limited measurements. Note: the number within the bracket means the number of particles used for one run.

for either a whole state vector or one subgroup is calculated only using the collected accelerations, while, after the resampling step, the accelerations from all the DOFs must be calculated using the measurement equation, and then used for the state evolution at next time step.

Let us consider that only the accelerations of the 1st, 3rd, 5th, …, and 19th stories are measured and then adopted for likelihood calculation. The state space model, state and measurement partitioning, and initialization for the three algorithms remain unchanged. The stiffness estimations from one typical run are given in Fig. 7, and the RMSE and RE averaged from the 25 runs are presented in Fig. 8. All the estimates converge in about three seconds, the same as in Fig. 3. On the other hand, the estimation accuracies from all three PFs decrease with fewer measurements, especially for the stiffnesses at higher stories, thus justifying the importance of sufficient sensors in the SHM system.

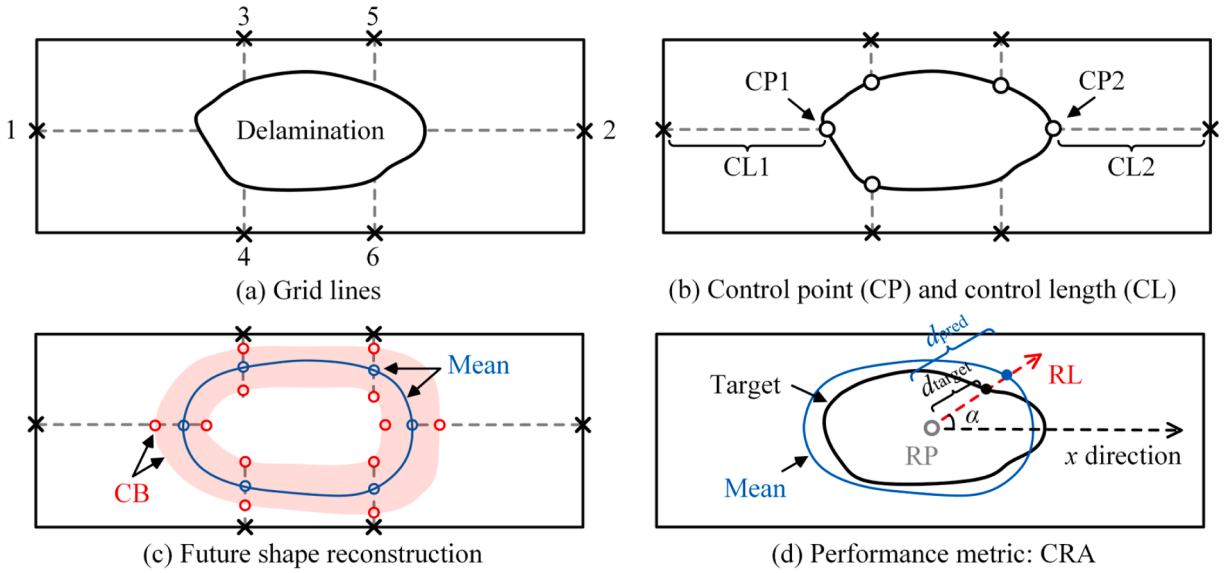


Fig. 9. Control points (CPs), and control lengths (CLs), and their predictions.Note: (i) the blue and red dots denote the mean and the confidence boundaries (CBs) of the predicted control points (CPs), respectively; (ii) the blue curve and the red part denote the mean and the confidence boundaries (CBs) of predicted shape, respectively; and (iii) 'RP' and 'RL' mean 'reference point', and 'reference line', respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

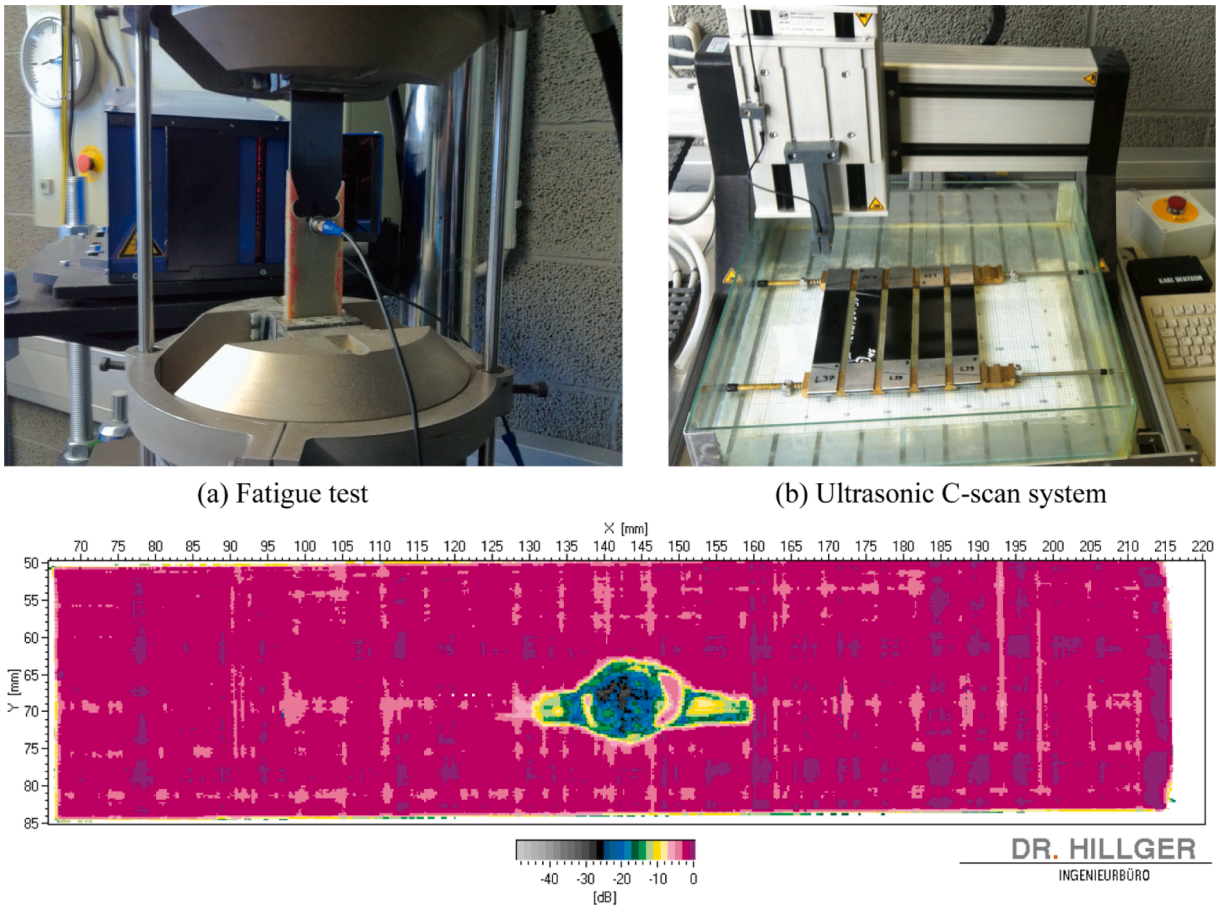


Fig. 10. Fatigue test for composite panel with ultrasonic C-scan monitoring system.

5. Experimental study of delamination shape prediction in composites

The generalizability of MLPF to different SHM applications is tested here. The shape prediction method and composite fatigue test are introduced in Sections 5.1 and 5.2, respectively. The state space modeling is given in Section 5.3. The prognostic results using MLPF and those using PF and MLPF are provided in Sections 5.4 Section 5.5, respectively.

5.1. Shape prediction method

The method has been recently published in [9] by some of the authors, thus it will be briefly mentioned here. Interested readers can refer to this publication for further details. The discretization of the delamination shape is depicted in Fig. 9 (a) and (b). First, the image is segmented by some grid lines with their layout (i.e., the locations of the six cross markers) kept unchanged for all the other images. Then, some control points (CPs) can be determined at which the grid lines intersect the delamination contour, and one control length (CL) is taken as the distance between one CP and its corresponding cross marker. Each CL has its own evolution model, which is refined using the latest measurement through a state estimation algorithm like PF. This updated model then informs the future behavior of the CL.

Fig. 9 (c) utilizes one representative example to illustrate how the shape is reconstructed for one future step through the predicted CLs. First, the means and confidence boundaries (CBs) of the future CL are used for locating the corresponding CPs (the blue and red dots). Then, the latter can be adopted for reconstructing the future shapes through a cubic spline interpolation. Fig. 9 (b) shows the cumulative relative accuracy (CRA) proposed in [9] for evaluating the shape prediction accuracy at each load cycle step. Provided with multiple reference lines (RLs), each of which is a ray arising from a reference point (RP, simply defined as the center of the image in this study) along a certain angle α, the CRA of the shape predicted at kf-step based the estimates at k-th step can be written as

$$CRA_{k,kf} = \frac{1}{N_r}\sum_{rl=1}^{N_r}\left(1 - \frac{\left|d_{target}^{k,kf,rl} - d_{pred}^{k,kf,rl}\right|}{d_{target}^{k,kf,rl}}\right) \tag{63}$$

where N_r means the number of RLs, d_{target} is the distance between the RP and the target shape along with one RL, whereas d_{pred} is the distance between the RP and the predicted shape. The CRA indicator is a number within the range of $[-\infty, 1]$, where a higher CRA represents a more accurate prediction.

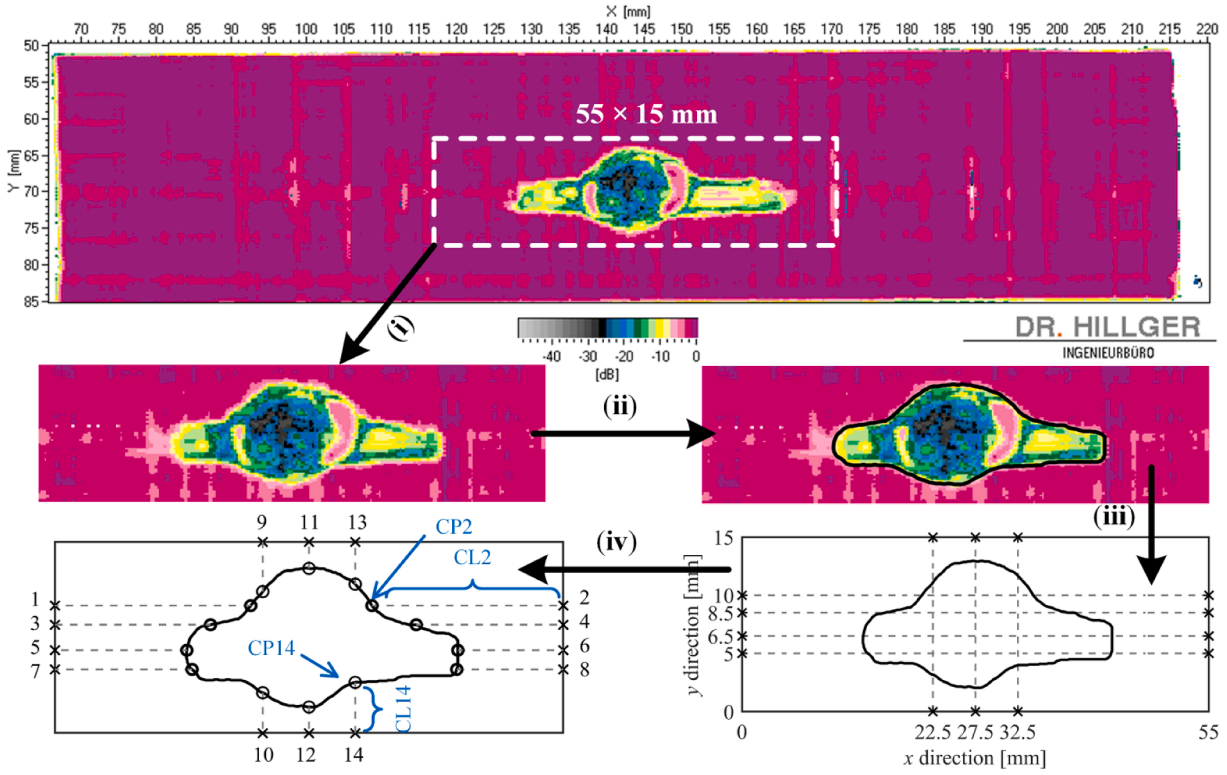


Fig. 11. Image processing.

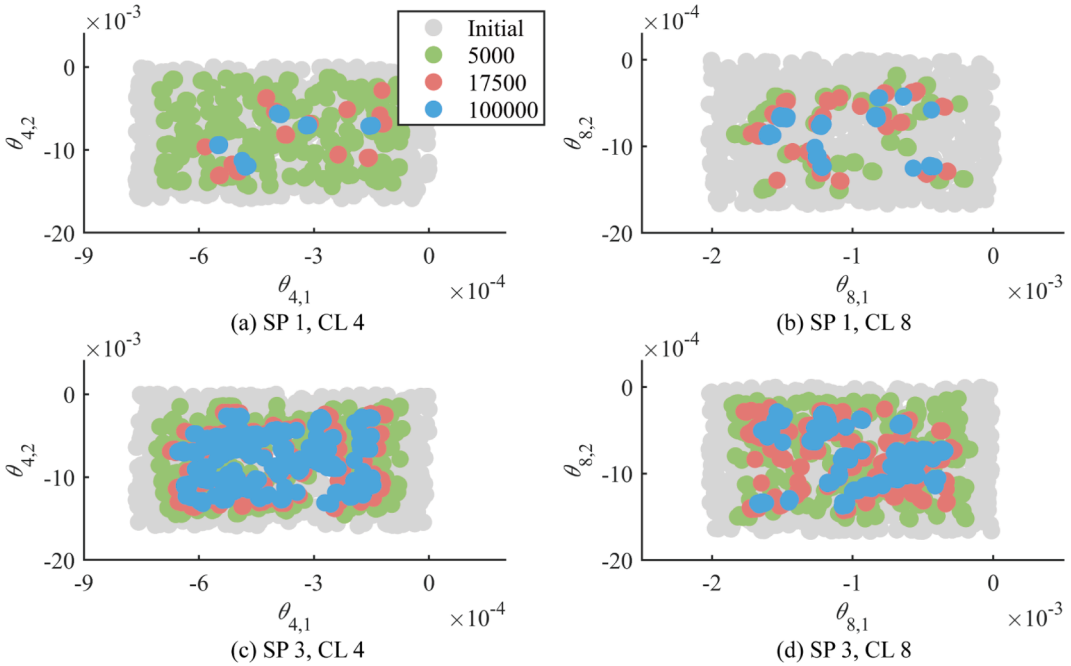


Fig. 12. MLPF-based parameter estimation results for two selected control lengths from specimens SP 1 and SP 3.

5.2. Experimental setup

Fig. 10 (a) shows that the run-to-failure fatigue experiments of three cross-ply carbon-epoxy laminate specimens were conducted with a servo-hydraulic *Instron/Schenk*® 100 kN tension–tension test machine. The laminate has a size of 250 × 35 × 2mm. It was manufactured by Cycom 977–2-35–12 k HTS *prepreg* material with the stacking sequence of $[0/90]_{4S}$, resulting in 18 plies in total. An impact with 3.8 Joule impact energy was produced by a semi-spherical striker drop weight tower and then applied to the panel center, leading to initial damage like delamination and modulus reduction. The fatigue tests were then carried out at a maximum applied load of 49kN, a loading frequency of 6Hz, and a stress ratio of 0.1.

Fig. 10 (b) shows that the *USPC 3040 DAC* from Ingenieurbüro Dr. Hillger® C-scan system was adopted for offline monitoring. A sample C-scan image obtained during these tests is given in Fig. 10 (c), where the *x*-direction is the load direction.

Fig. 11 shows the two steps for extracting the 'true' delamination shape from the raw ultrasonic C-scan image. First, a 55 × 15mm rectangle sub-image, centered on the delamination, is extracted from the raw image at each load cycle step. Then, as the unit of db stands for the depth of delamination, the delamination contour is selected as the area limit where the signal amplitude is less than −6db, by which we ensure that the delamination is considered deep enough within the laminate thickness. The delamination shapes extracted at different load cycles for the three specimens SP 1, SP 2, and SP 3 are given in Appendix A.1. For each specimen, the delamination shape always grows under a similar pattern, i.e., it propagates along the loading direction only, thus enabling the possibility of its prediction. Note that SP 2 will be used for training some MLPF hyperparameters, while SP 1 and SP 3 for testing.

5.3. State space modeling and partitioning

Given the coupling of each CL evolution, the CL as a function of the number of load cycles can be built as follows:

$$\begin{cases} \begin{bmatrix} \theta_{1,k} \\ \vdots \\ \theta_{14,k} \\ CL_{1,k} \\ \vdots \\ CL_{1,k} \end{bmatrix} = \begin{bmatrix} \theta_{1,k-1} + \omega_{1,k-1} \\ \vdots \\ \theta_{14,k-1} + \omega_{14,k-1} \\ CL_{1,k-1} + f_1\left(CL_{1,k-1}, \cdots, CL_{14,k-1}, \Delta N_k, \theta_{1,k}, \omega_{15,k-1}\right) \\ \vdots \\ CL_{14,k-1} + f_{14}\left(CL_{1,k-1}, \cdots, CL_{14,k-1}, \Delta N_k, \theta_{14,k}, \omega_{28,k-1}\right) \end{bmatrix} \\ Y_k = \left[CL_{1,k} \quad \cdots \quad CL_{14,k} \right]^T + \eta_k \end{cases} \tag{64}$$

where $f_l(\cdot)$ is the evolution function for the l-th CL, $\boldsymbol{\theta}_l$ is a vector of its function parameters, $\boldsymbol{\omega}_l$ and ω_{l+14} are the process noise for the parameter vector and CL, respectively, and $\boldsymbol{\eta}$ is the measurement noise vector. Given the satisfactory performance of three-dimension

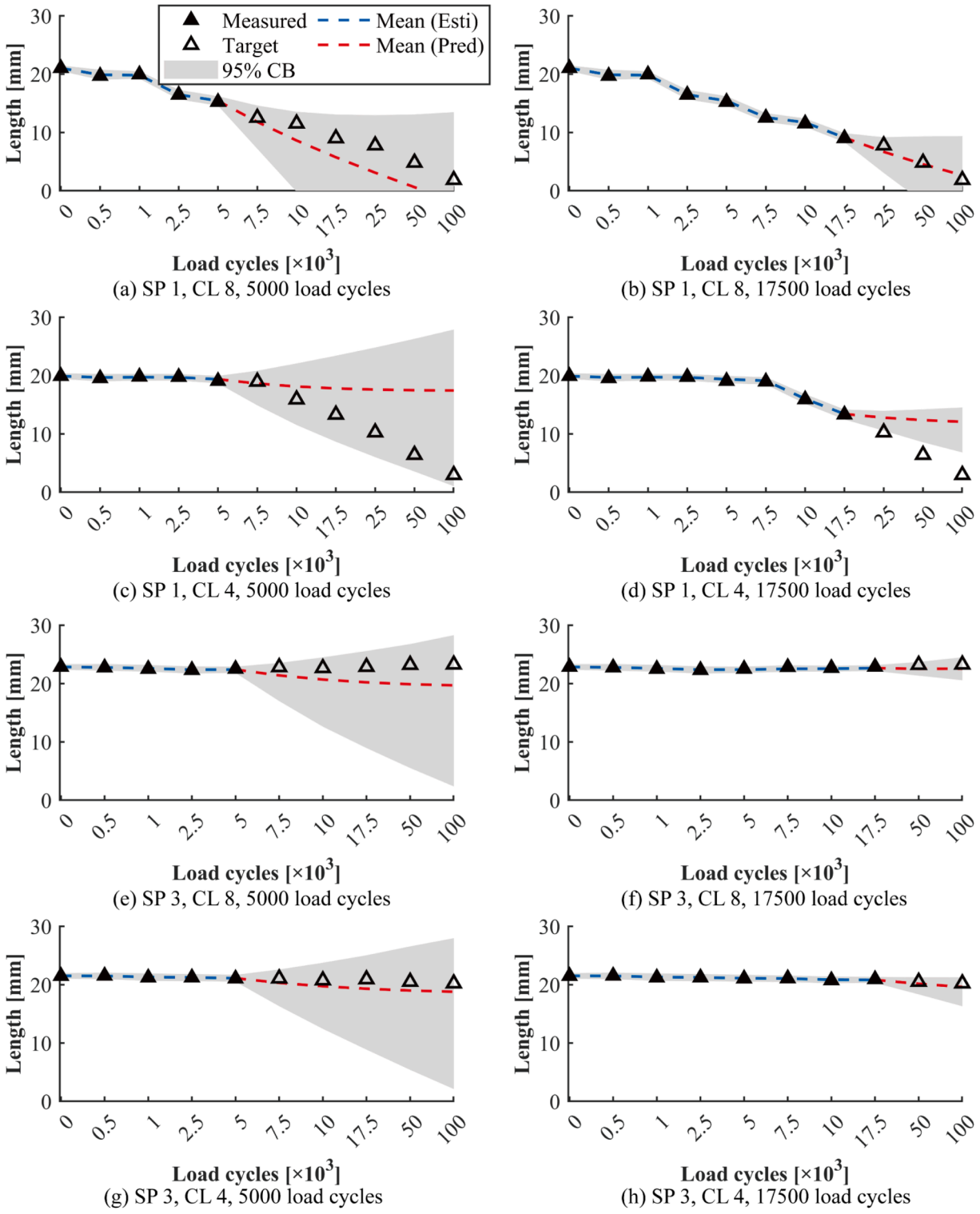
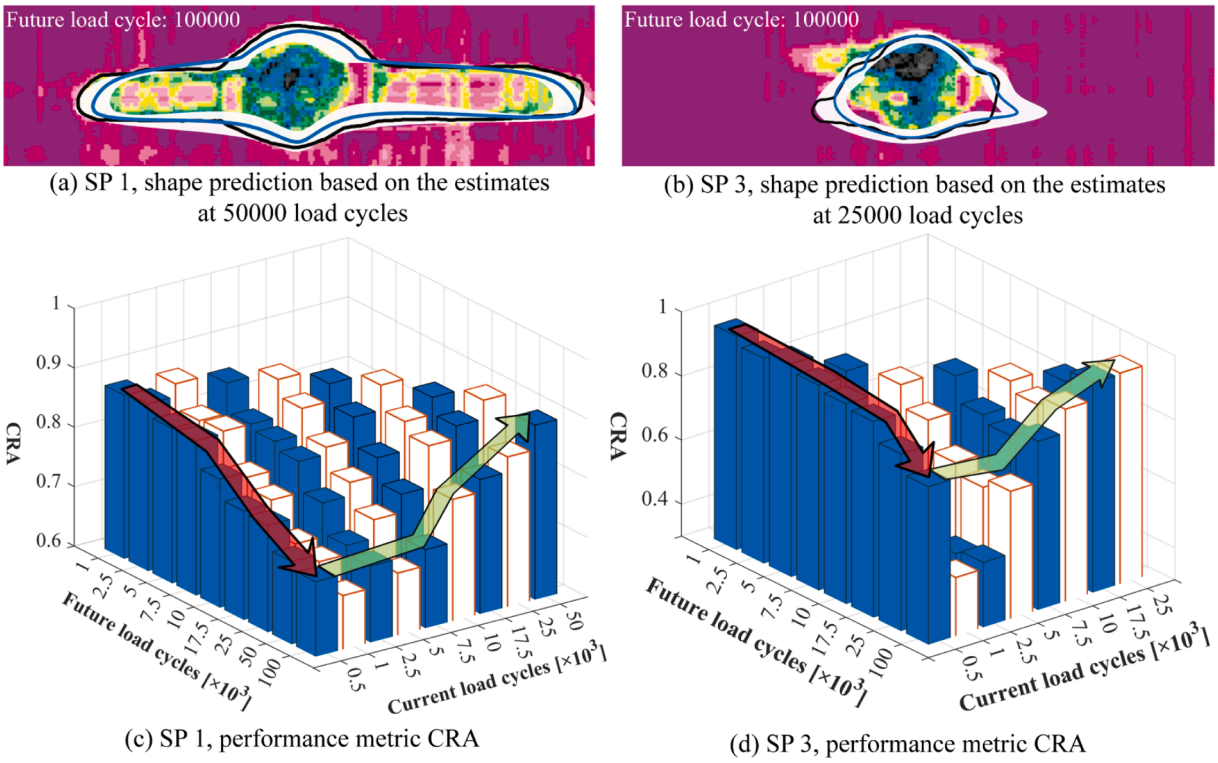


Fig. 13. MLPF-based predictions for two control lengths with the estimates at two selected load cycles for specimens SP 1 and SP 3.



(a) SP 1, shape prediction based on the estimates at 50000 load cycles

(b) SP 3, shape prediction based on the estimates at 25000 load cycles

(c) SP 1, performance metric CRA

(d) SP 3, performance metric CRA

Fig. 14. MLPF-based Shape predictions and cumulative relative accuracies (CRAs) for specimens SP 1 and SP 3.

curve-surface fitting as shown in [18], it is used to describe the evolution of the l-th CL with the number of load cycles as

$$CL_{l,k} = CL_{l,k-1} + e^{\omega_{l+14,k-1}}\left(\theta_{l,1,k} \times \Delta N_k^2 + \theta_{l,2,k} \times \Delta N_k + \theta_{l,3,k} \times \left(\frac{CL_{1,k-1} + \cdots + CL_{14,k-1}}{14}\right) + \theta_{l,4,k}\right) \tag{65}$$

where $\boldsymbol{\theta}_l = \left[\theta_{l,1} \quad \theta_{l,2} \quad \theta_{l,3} \quad \theta_{l,4}\right]^T$, and $\omega_{l+14} \sim \mathcal{N}\left(-\frac{\sigma^2}{2}, \sigma^2\right)$ is the unbiased Gaussian process noise with standard deviation σ [41]. Note that one may build the CL evolution model through many other modeling options, such as Gaussian process, and neural network. But this will not be discussed in detail here, as it is out of the scope of this work. Additionally, kernel smoothing [42] is adopted to improve the parameter estimation performance, due to its demonstrated performance in the work of the same authors [8].

$$\boldsymbol{\theta}_{l,k} = \sqrt{1-ks^2}\boldsymbol{\theta}_{l,k-1} + \left(1 - \sqrt{1-ks^2}\right)\widehat{\boldsymbol{\theta}}_{l,k-1} + \boldsymbol{\omega}_{l,k-1} \tag{66}$$

where ks is the kernel smoothing parameter, and it is set as 0.3 in this study.

The hyperparameters of PF, MPF, and MLPF have been selected through the work of some of the same authors [9] and a trial-and-error procedure. Same with Section 4, the particle numbers for PF, MPF, and MLPF are 80000, 800, and 800, respectively. Systematic resampling is implemented at each time step. The initial uniform distribution of each CL is created by the same CL from specimen SP 2, i.e., $CL_{l,0} \sim \mathrm{U}(CL_{l,0,\mathrm{SP2}} - 0.5, CL_{l,0,\mathrm{SP2}} + 0.5)$. The initial uniform distribution of each parameter is created by the coefficient fitted by specimen SP 2, e.g., $\theta_{l,1,0} \sim \mathrm{U}(0, 2 \times \theta_{l,1,\mathrm{SP2}})$, which has been well elaborated in [9]. The standard deviation σ in the process noise ω_{l+14} is defined as 2, and the process noise for each parameter is defined as 2 % of its corresponding coefficient fitted from specimen SP 2.

Following the previous partitioning rule, one state subgroup consists of the state components from one CL. Thus, this experimental example has fourteen state subgroups,

$$\begin{cases} \boldsymbol{x}_k^1 = \left[CL_{1,k} \quad \theta_{1,1,k} \quad \theta_{1,2,k} \quad \theta_{1,3,k} \quad \theta_{1,4,k}\right]^T \\ \boldsymbol{x}_k^2 = \left[CL_{2,k} \quad \theta_{2,1,k} \quad \theta_{2,2,k} \quad \theta_{2,3,k} \quad \theta_{2,4,k}\right]^T \\ \qquad\qquad\qquad \vdots \\ \boldsymbol{x}_k^{14} = \left[CL_{14,k} \quad \theta_{14,1,k} \quad \theta_{14,2,k} \quad \theta_{14,3,k} \quad \theta_{14,4,k}\right]^T \end{cases} \tag{67}$$

where each subgroup needs to evolve by resorting to its own state components and all the other CLs. Same with Section 3, the local measurement vector for one state subgroup only consists of one measured CL, and there is no need to define the local state vector in

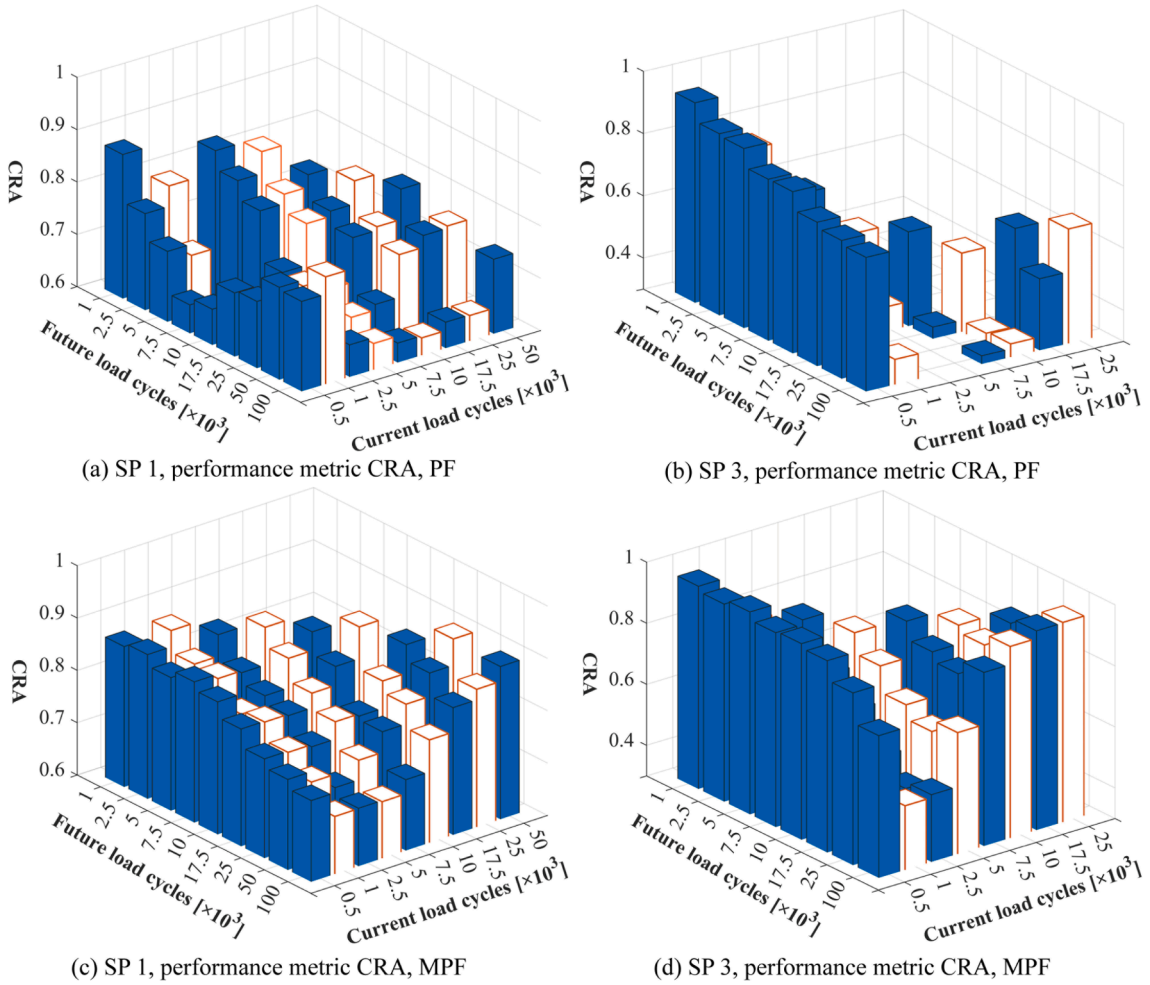


Fig. 15. Cumulative relative accuracies (CRAs) for specimens SP 1 and SP 3 using PF and MPF.

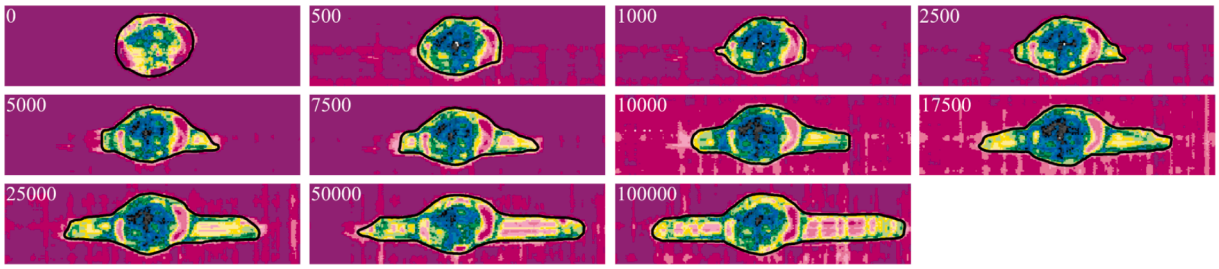


Fig. A1. Ultrasonic C-scan delamination images at different load cycles from specimen SP 1.

MLPF, because the likelihood function for each subgroup only involves the CL of that subgroup.

5.4. Prognostic results from MLPF

MLPF is applied to specimens SP 1 and SP 3 for providing the estimates of CLs and parameters, which can be used for future length and shape predictions. Fig. 12 shows the parameter estimation results for the two selected CLs for the two specimens. The distributions of the two evolution parameters, namely $\theta_{l,1}$ and $\theta_{l,2}$, gradually shrink during the estimation process, thus proving a good convergence.

Fig. 13 show the predictions for the two selected CLs based on their PF parameter estimates at two selected load cycles. The predictions of the posterior PDFs have their averages approaching the target with the increasing load cycle step, and the 95 %

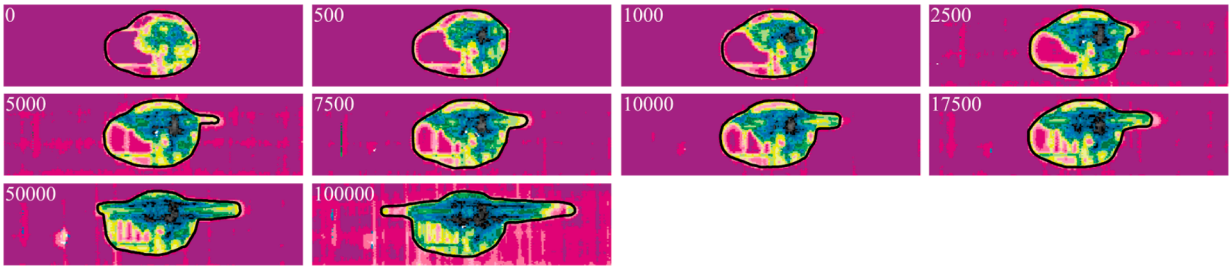


Fig. A2. Ultrasonic C-scan delamination images at different load cycles from specimen SP 2.

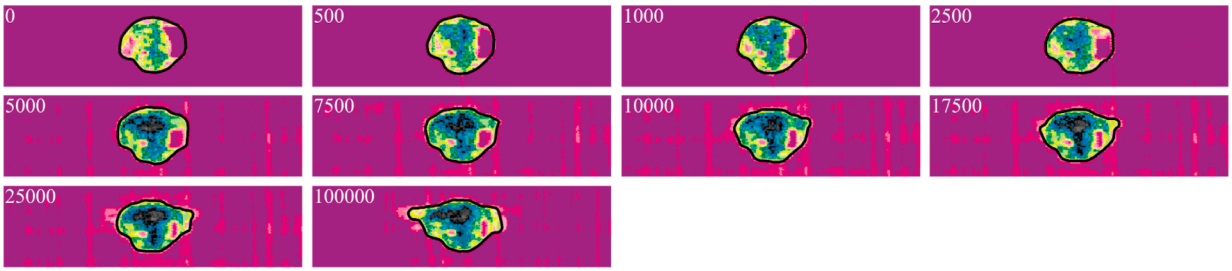


Fig. A3. Ultrasonic C-scan delamination images at different load cycles from specimen SP 3.

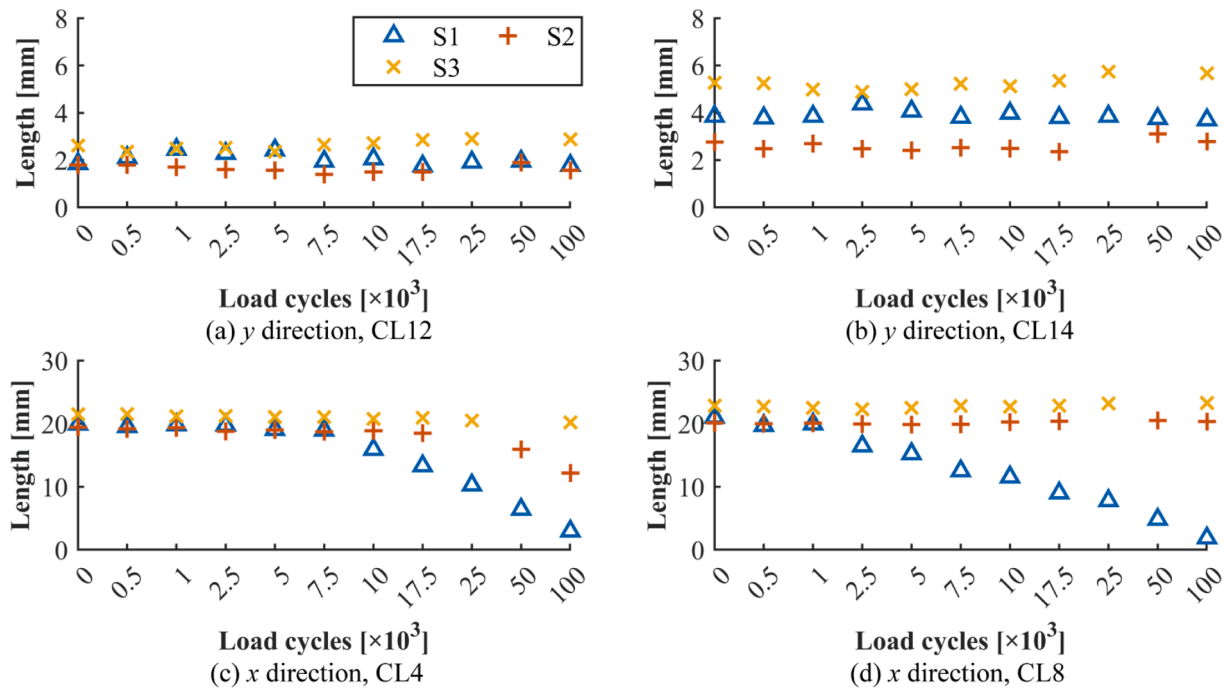


Fig. A4. Growths of four selected control lengths from specimens SP 1 – SP 3.

confidence boundaries (CBs) including the target.

Following the strategy given in Section 5.1, the estimates at those steps can be located by the future CLs, and then used for future delamination shaping. For simplicity, only the delamination shapes predicted with the estimates at one selected load cycle are given in Fig. 14 (a) and (b). Then, by defining the center of the delamination image (i.e., [27.5mm, 7.5mm]) as the reference point (RP), and $\boldsymbol{\alpha} = [0, 5, 10, 15, 45, 90, 135, 165, 170, 175, 180, 185, 190, 195, 235, 270, 315, 345, 350, 355]$ degrees with respect to the x-direction, the CRA prognostic performance metrics are shown in Fig. 14 (c) and (d), where the shape prediction based on the estimates at one certain step seems less accurate as the prediction step horizon increases (red arrow), because of the growing uncertainties of

delamination growth ahead in the future. On the other hand, the prediction at one specific future step gets more accurate if more delamination images are adopted for model updating (green arrow), due to the convergence of the evolution parameters. All the above conclusions demonstrate the superior performance of MLPF in high-dimensional system identification, and consequently prognostic.

5.5. Comparing PF, MPF, and MLPF

Both PF and MPF are applied to specimens SP 1 and SP 3 for estimating CLs and parameters, which can be then used for future shape predictions. For one run, the computation time of applying the PF, MPF, and MLPF to specimen SP1 are 0.251 s, 0.337 s, and 27.430 s. For the sake of simplicity, only the CRA results from PF and MPF are given in Fig. 15. Consistent with the observations made in Sections 3 and 4, both MPF and MLPF require a significantly reduced number of particles (or computation time) to deliver accurate shape predictions. Additionally, the prediction accuracies from the MPF and the MLPF are quite similar, while the latter requires less computation time, demonstrating the computation efficiency of MLPF over MPF.

6. Conclusion

The application of particle filter in a high-dimensional system is limited due to the curse of dimensionality. By combining multiple particle filter (MPF) and the decay of correlations property, this paper has developed a novel multiple local particle filter (MLPF) for system identification. One state vector is partitioned into several state subgroups, each containing the components from one DOF or one control length (CL) and then estimated by one PF. The superior performances of the MLPF over the MPF and the PF with respect to computation effort and estimation accuracy have been proved by the identification results of the toy example, the numerical twenty-story Bouc-Wen frame structure under the ground motion, and the experimental fatigue delamination shape prediction in composites.

The number of particles required by the PF will exponentially increase with the state dimension, while the number of particles required by the MPF or the MLPF will remain relatively stable. This proves the abilities of the MPF and the MLPF in alleviating the curse of dimensionality. the MLPF can provide similarly accurate estimations as the MPF under the same number of particles but requires less computation time due to fewer components involved in the likelihood. This advantage of the MLPF over the MPF will be even more apparent in a higher-dimensional system, where the number of components within the likelihood of the MPF will increase with a higher state dimension, while the number of components for the MLPF may not increase as much.

For the system identification of the twenty-story frame structure, it has been found that the stiffness of the story closer to the ground motion shows faster convergence, possibly due to its higher contribution to the likelihood calculation at the initial steps. Furthermore, PF algorithms can work with partial acceleration measurements but provide less accurate estimates. Finally, the generalizability of MLPF to different SHM applications has been validated by the delamination shape prediction in composites, where the future shape can be accurately predicted by the MLPF estimates.

The structures used in this study has been found to be simple, thus their state partitioning processes are straightforward, e.g., the state components from one DOF or control length. However, this may not be as simple for the other SHM applications, especially those tailored for complex and 3D structures. To address this challenge, it is recommended that a sensitivity analysis should be conducted to determine the appropriate state partitioning and the required local vectors. Alternatively, an adaptive state partitioning approach could be developed to more fully explore the potential of this strategy for high-dimensional problems. By taking these steps, researchers can better tailor their SHM models to specific applications and ensure the most effective use of available data. Moreover, the proposed MLPF usually requires a sufficient number of measurements, which may not always be possible and still calls for an advanced system identification method with a limited number of measurements.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgment

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 859957. The authors would appreciate Prof. Manuel Chiachío and Prof. Juan Chiachío from University of Granada for providing the experimental data. Also, the authors would like to thank the Institute of Polymers and Composites at Hamburg University of Technology (TUHH), for their collaborative support regarding the production of samples and the collection of monitoring data.

Appendix A1

Figs. A1-A3 present the images of the delamination and its contour at different load cycles from specimens S1, S2, and S3, respectively. Fig. A4 shows four selected CL evolutions from the three specimens.

References

[1] S.S. Law, K. Zhang, Z.D. Duan, Structural damage detection from coupling forces between substructures under support excitation, Eng. Struct. 32 (2010) 2221–2228.
[2] S. Pan, D. Xiao, S. Xing, S.S. Law, P. Du, Y. Li, A general extended Kalman filter for simultaneous estimation of system and unknown inputs, Eng. Struct. 109 (2016) 85–98.
[3] J. He, Y.L. Xu, S. Zhan, Q. Huang, Structural control and health monitoring of building structures with unknown ground excitations: experimental investigation, J. Sound Vib. 390 (2017) 23–38.
[4] Y. Ding, B.Y. Zhao, B. Wu, X.C. Zhang, L.N. Guo, Simultaneous identification of structural parameter and external excitation with an improved unscented kalman filter, Adv. Struct. Eng. 18 (2015) 1981–1998.
[5] E.N. Chatzi, A.W. Smyth, The unscented Kalman filter and particle filter methods for nonlinear structural system identification with non-collocated heterogeneous sensing, Struct. Control Health Monit. 16 (2009) 99–123.
[6] A. Olivier, A.W. Smyth, Particle filtering and marginalization for parameter identification in structural systems, Struct. Control Health Monit. 24 (2017) e1874.
[7] T. Li, L. Lomazzi, F. Cadini, C. Sbarufatti, J. Chen, S. Yuan, Numerical simulation-aided particle filter-based damage prognosis using Lamb waves, Mech. Syst. Sig. Process. 178 (2022) 109326.
[8] T. Li, C. Sbarufatti, F. Cadini, J. Chen, S. Yuan, Particle filter-based hybrid damage prognosis considering measurement bias, Struct. Control Health Monit. 29 (2021) e2914.
[9] T. Li, F. Cadini, M. Chiachío, J. Chiachío, C. Sbarufatti, Particle filter-based delamination shape prediction in composites subjected to fatigue loading, Struct. Health Monit. 14759217221116041 (2022).
[10] J. Chen, S. Yuan, H. Wang, W. Yang, On particle filter improvements for on-line crack growth prognosis with guided wave monitoring, Smart Mater. Struct. 28 (2019) 035011.
[11] T. Li, J. Chen, S. Yuan, F. Cadini, C. Sbarufatti, Particle filter-based damage prognosis using online feature fusion and selection, Mech. Syst. Sig. Process. 203 (2023) 110713.
[12] F. Cadini, L. Lomazzi, M. Ferrater Roca, C. Sbarufatti, M. Giglio, Neutralization of temperature effects in damage diagnosis of MDOF systems by combinations of autoencoders and particle filters, Mech. Syst. Sig. Process. 162 (2022) 108048.
[13] E.N. Chatzi, A.W. Smyth, Particle filter scheme with mutation for the estimation of time-invariant parameters in structural health monitoring applications, Struct. Control Health Monit. 20 (2013) 1081–1095.
[14] J. Chen, S. Yuan, C. Sbarufatti, X. Jin, Dual crack growth prognosis by using a mixture proposal particle filter and on-line crack monitoring, Reliab. Eng. Syst. Saf. 215 (2021) 107758.
[15] F. Zhao, X. Zhou, C. Wang, L. Dong, S.N. Atluri, Setting Adaptive Inspection Intervals in Helicopter Components, AIAA Journal, Based on a Digital Twin, 2023, pp. 1–14.
[16] D. Cristiani, C. Sbarufatti, M. Giglio, Damage diagnosis and prognosis in composite double cantilever beam coupons by particle filtering and surrogate modelling, Struct. Health Monit. 1475921720960067 (2020).
[17] J. Chiachío, M. Chiachío, S. Sankararaman, A. Saxena, K. Goebel, Condition-based prediction of time-dependent reliability in composites, Reliab. Eng. Syst. Saf. 142 (2015) 134–147.
[18] L. Wang, C. Zhang, C. Tao, H. Ji, Y. Yang, J. Qiu, Prediction of multiple fatigue crack growth based on modified Paris model with particle filtering framework, Mech. Syst. Sig. Process. 190 (2023) 110124.
[19] J. Chen, S. Yuan, X. Jin, On-line prognosis of fatigue cracking via a regularized particle filter and guided wave monitoring, Mech. Syst. Sig. Process. 131 (2019) 1–17.
[20] C. Snyder, T. Bengtsson, P. Bickel, J. Anderson, Obstacles to high-dimensional particle filtering, Mon. Weather Rev. 136 (2008) 4629–4640.
[21] C. Snyder, Particle filters, the "optimal" proposal and high-dimensional systems, in: Proceedings of the ECMWF Seminar on Data Assimilation for Atmosphere and Ocean, 2011, pp. 1–10.
[22] R. Sajeeb, C.S. Manohar, D. Roy, A conditionally linearized Monte Carlo filter in non-linear structural dynamics, Int. J. Non Linear Mech. 44 (2009) 776–790.
[23] L. Úbeda-Medina, J. Grajal, Sigma-point multiple particle filtering, Signal Process. 160 (2019) 271–283.
[24] M.F. Bugallo, T. Lu, P.M. Djuric, Target tracking by multiple particle filtering, IEEE Aerospace Conference IEEE 2007 (2007) 1–7.
[25] P.M. Djuric, T. Lu, M.F. Bugallo, Multiple particle filtering, 2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07, 2007, pp. III-1181-III-1184.
[26] P. Closas, M.F. Bugallo, Improving accuracy by iterated multiple particle filtering, IEEE Signal Process Lett. 19 (2012) 531–534.
[27] M. Orton, W. Fitzgerald, A Bayesian approach to tracking multiple targets using sensor arrays and particle filters, IEEE Trans. Signal Process. 50 (2002) 216–223.
[28] C. Kreucher, K. Kastella, A.O. Hero, Multitarget tracking using the joint multitarget probability density, IEEE Trans. Aerosp. Electron. Syst. 41 (2005) 1396–1414.
[29] Y. Wei, M.R. Morelande, L. Kong, J. Yang, A computationally efficient particle filter for multitarget tracking using an independence approximation, IEEE Trans. Signal Process. 61 (2013) 843–856.
[30] T. Li, S. Sun, J.M. Corchado, T.P. Sattar, S. Si, Numerical fitting-based likelihood calculation to speed up the particle filter, Int. J. Adapt Control Signal Process. 30 (2016) 1583–1602.
[31] P. Rebeschini, R. Van Handel, Can local particle filters beat the curse of dimensionality? Ann. Appl. Probab. 25 (2015) 2809–2866.
[32] P. Chavali, A. Nehorai, Hierarchical particle filtering for multi-modal data fusion with application to multiple-target tracking ☆, Signal Process. 97 (2014) 207–220.
[33] J. MacCormick, M. Isard, Partitioned Sampling, Articulated Objects, and Interface-Quality Hand Tracking, Springer, Berlin Heidelberg, Berlin, Heidelberg, 2000, pp. 3–19.
[34] L.A. Úbeda Medina, Robust techniques for multiple target tracking and fully adaptive radar= Técnicas robustas para seguimiento de múltiples blancos y radar adaptativo, Telecomunicacion (2018).
[35] R.N. Bracewell, R.N. Bracewell, The Fourier transform and its applications, McGraw-hill New York, 1986.
[36] M.S. Arulampalam, S. Maskell, N. Gordon, T. Clapp, A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking, IEEE Trans. Signal Process. 50 (2002) 174–188.
[37] N.J. Gordon, D.J. Salmond, A.F.M. Smith, Novel approach to nonlinear/non-Gaussian Bayesian state estimation, IEE Proceedings F - Radar and Signal Processing 140 (1993) 107–113.

[38] Y. Lei, D. Xia, K. Erazo, S. Nagarajaiah, A novel unscented Kalman filter for recursive state-input-system identification of nonlinear systems, Mech. Syst. Sig. Process. 127 (2019) 120–135.
[39] Y. Lei, M. He, C. Liu, S.-Z. Lin, Identification of tall shear buildings under unknown seismic excitation with limited output measurements, Adv. Struct. Eng. 16 (2013) 1839–1849.
[40] Y. Li, Y. Luo, H.-P. Wan, C.-B. Yun, Y. Shen, Identification of earthquake ground motion based on limited acceleration measurements of structure using Kalman filtering technique, Struct. Control Health Monit. 27 (2020) e2464.
[41] M. Corbetta, C. Sbarufatti, M. Giglio, M.D. Todd, Optimization of nonlinear, non-Gaussian Bayesian filtering for diagnosis and prognosis of monotonic degradation processes, Mech. Syst. Sig. Process. 104 (2018) 305–322.
[42] J. Liu, M. West, Combined parameter and state estimation in simulation-based filtering, sequential monte carlo methods in practice, Springer (2001) 197–223.