A Sparse adaptive Bayesian filter for input estimation problems

Julian Ghibaudo, Mathieu Aucejo, Olivier de Smet

A Sparse adaptive Bayesian filter for input estimation problems

J. Ghibaudo, M. Aucejo and O. De Smet[a]

[a]*Laboratoire de Mécanique des Structures et des Systèmes Couplés, Conservatoire National des Arts et Métiers, 2 Rue Conté, 75003 Paris, France*

Abstract

The present paper introduces a novel Bayesian filter for estimating mechanical excitation sources in the time domain from a set of vibration measurements. The proposed filter is derived from a very general Bayesian formulation, unifying most of the state-of-the-art recursive filters developed in the last decade for solving input-state estimation problems. More specifically, the proposed Bayesian filter allows promoting the spatial sparsity of the estimated input vector, by assuming that the predicted input vector is a random vector with independent and identically distributed components following a generalized Gaussian distribution. To properly estimate the most probable parameters of the latter probability distribution, a nested Bayesian optimization is implemented. The validity of the proposed approach, called Sparse adaptive Bayesian Filter, is assessed both numerically and experimentally. In particular, the comparisons performed with some state-of-the-art filters show that the proposed strategy outperforms the existing filters in terms of input estimation accuracy and avoids the so-called drift effect.

Keywords: Linear inverse problem, Force localization, Space-time

*Corresponding author. E-mail address: mathieu.aucejo@lecnam.net

approach, Bayesian filter, Kalman filter.

1. Introduction

Force reconstruction is a current topic in the structural dynamics community, which emerges from the need to know the mechanical excitations acting on a structure for improving its design or monitoring its structural health. As an example, a mechanical impact (or shock), whether intended or not, can cause damages in structures, such as mechanical failures or a breakdown in the embedded devices. However, the actual impact is sometimes hard to specify, due to the potential difficulty in instrumenting the area of interest or to the lack of knowledge of its space-time characteristics (location, duration and intensity). Such practical considerations make the implementation of inverse identification methods necessary to identify the excitation sources acting on a mechanical structure from the measurement of kinematic quantities such as strain, displacement, velocity or acceleration. Unfortunately, inverse problems are generally mathematically ill-posed, so that low-level measurement noise can cause large reconstruction errors. For time domain applications, this adverse effect has been handled through the development of dedicated resolution strategies, such as dynamic programming [1–3], neural networks [4–7], virtual field method [8, 9], the force analysis technique [10, 11] or the sequential deconvolution [12, 13] to cite only a few of them. However, among all the methods existing in the literature, Tikhonov-like regularization and Kalman-like filtering are certainly the most widely used. Both methods find their roots in the early 1960's from the seminal works of Andrey Tikhonov [14] and Rudolf Kalman [15]. Despite their apparent dif-

ferences, they share some common features. First, the Kalman gain can be seen as a regularized inverse operator, similar to that defined in Tikhonov-like regularization. Second, both methods can be derived from the Bayesian formalism [16, 17], which allows a better understanding of the main assumptions underlying these strategies and paves the way for further developments and analyses. More precisely, Tikhonov-like regularization is a special type of Bayesian regularization, while Kalman-like filters belong to the general class of Bayesian filters.

From the Bayesian perspective, Tikhonov-like regularization addresses the ill-posedness problem by including in the formulation of the inverse problem some prior information on the noise corrupting the data and the time or space-time distribution of the excitation sources. This information is encoded respectively in the data-fidelity and regularization terms. Such a theoretical framework has led to the development of several methods in the context of time-domain force reconstruction. One can cite for instance the ℓ_2-regularization [18–21], the ℓ_1-regularization [22–24], the ℓ_q-regularization [25] or the $\ell_{2,q}$-regularization [26].

Whereas Tikhonov-like regularization operates on the whole set of measurements, Kalman-like filtering solves the inverse problem recursively using a prediction/estimation scheme at each time step. In this sense, it is an online approach requiring a lower computational cost than Tikhonov-like regularization, which is an offline strategy. An additional benefit of Kalman-like filtering over Tikhonov-like regularization relies in the possibility of jointly estimated the state of the system (strain, displacement or velocity for instance) and the external inputs (mechanical excitations) that it undergoes. One the

first Kalman-like filter proposed for this purpose is probably the Gillijns and De Moor filter (GDF) [27, 28], which carries out the input estimation prior to the state estimation. Consequently, the input-state estimation is performed sequentially and not strictly jointly. To this end, Lourens et al. introduce a fictitious state equation for the input in order to form an augmented state vector. This fictitious equation assumes a random walk model for the evolution of the input vector. In doing so, they derive the Augmented Kalman Filter (AKF), that estimates the augmented state vector using a standard Kalman filter [29]. The counterpart of this simplicity is the necessary tuning of the variance of the noise process associated to the input's fictitious equation, which is estimated offline from the L-curve principle. Unfortunately, the abovementioned filters suffer from the so-called drift effect in the estimations, especially when only acceleration measurements are used as input data. As an attempt to mitigate the drift effect, three alternative filters have been proposed. The first one is the Dual Kalman Filter (DKF) proposed by Eftekhar Azam et al. [30, 31]. In DKF, two standard Kalman filters are running in a sequential way, one for the input vector and one for the state vector. To do so, DKF makes use of the same fictitious state equation for the input vector as that implemented in AKF, whose noise variance also needs to be tuned offline. The second alternative is to extend the AKF. By doing this, Nayek proposes to model each component of the input vector as a Gaussian latent process instead of a random walk process [32]. Here again, an offline optimization procedure is required to properly tuned the hyperparameters governing the covariance functions related to each Gaussian process. Another extension is brought by Naets et al. [33]. The analytical analysis of

the observability properties of AKF leads to understand that the estimation based only on acceleration measurements is not observable, thus unreliable, as a drift is induced. To counterbalance this phenomenon, dummy measurements are introduced, which remove almost entirely the drift effect. Finally, Wei et al. also proposes a variation of the AKF, based on sparse constraint theory [34]. The estimations of the state and its covariance are corrected within a given range at every time step, and, under some assumptions, the drift is removed. The last alternative is the Sequential Bayesian Filter (SBF) developed by Sedehi et al. [35], which is a fully online strategy obtained by revisiting the input-state estimation problem from a Bayesian perspective. More specifically, in the prediction phase of SBF, the input vector at some time step is supposed to follow a multivariate Gaussian distribution with zero mean and covariance matrix equal to that estimated for the input vector at the previous time step. Furthermore, in SBF, the drift is avoided thanks to an additional trick consisting in modifying the predicted state vector used for estimating the input vector at a given time step. Despite their reasonable performances on the examples presented in the literature, none of the filters discussed above allows introducing some prior information on the spatial distribution of the input vector. As it will be shown later, this can lead to inaccurate reconstruction of the excitation field when the excitation sources are spatially sparse and a dense sensor network is used to measure the vibration field.

It is worth mentioning here that these practical situations can be encountered nowadays thanks to the advent of full-field measurements, such as Digital Image Correlation [36], high-speed holography [37] or infrared deflec-

tometry [38].

This paper introduces a Sparse adaptive Bayesian Filter (SaBF) aiming at avoiding the drift effect and promoting some kind of spatial sparsity of the excitation field if needed, while remaining purely online. To take this challenge, the proposed approach is derived from a very general Bayesian formulation of the sequential input-state estimation problem inspired by the work of Sedehi et al. [35]. However, the present paper goes one step further by showing that most of the state-of-the-art filters, namely, GDF, DKF and SBF, can be actually derived from the same Bayesian formulation. Practically, they mainly differ from each other by the hypothesis made on the input vector during the prediction step. This observation is at the roots of the proposed Bayesian Filter, which assumes that the predicted input vector is a random vector with independent and identically distributed components following a generalized Gaussian distribution characterized by its shape and scale parameters. To avoid a manual and cumbersome tuning of these hyperparameters, a nested Bayesian optimization is implemented to estimate their most probable values given the data available at each time step. In doing so, the proposed SaBF remains a strictly online strategy.

To properly describe and validate the SaBF, the present paper is divided into 5 parts. Section 2 describes the state-space representation generally used for dealing with input-state estimation problems. In section 3, the general Bayesian formulation of the joint input-state estimation problem at the roots of AKF is presented. The main steps and equations are developed to better highlight the main hypotheses and their influence on the computational form of this filter. This Bayesian formalism is extended in section 4 to

address the sequential input-state estimation problem. As it will be shown in this section, this extended Bayesian formulation provides a unique and unified framework to derive most of the state-of-the-art recursive filters, such as GDF, DKF and SBF. This section also emphasizes the main hypotheses made to obtain these filters, thus paving the way to SaBF, which is discussed in detail in section 5. In section 6, SaBF is applied in a purely numerical context in order to assess its robustness with respect to various practical parameters, such as the measurement noise, the sensors' density or the duration of the estimation (short-term or long-term). Finally, a real-world application is presented in section 7 to evaluate the performances of SaBF in operating conditions. In both numerical and experimental applications, SaBF is systematically compared to AKF, GDF, DKF and SBF, which allows putting into perspectives the overall performances of this novel Bayesian Filter.

2. Discretized state-space representation of dynamical systems

Bayesian filtering is based on the state-space representation of the dynamical system of interest. From a very general standpoint, it is composed of a state equation, describing the evolution of the system state at a certain time step from the knowledge of the system state and input at the previous time step, and an observation equation, encoding the relation between the measured data to the system state and input. Formally, the discretized state-space representation of a linear and time invariant system is expressed by:

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{w}_k^{\mathbf{x}} \\ \mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k + \mathbf{v}_k \end{cases}, \tag{1}$$

where $\mathbf{x}_k$, $\mathbf{u}_k$ and $\mathbf{y}_k$ are the state, input and output vectors at sample k, while $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$ and $\mathbf{D}$ are, respectively, the discretized state, input, output and feedthrough matrices. Here, $\mathbf{w}_k^{\mathbf{x}}$ denotes the Gaussian process noise with zero mean and covariance matrix $\mathbf{Q}_k^{\mathbf{x}}$ and $\mathbf{v}_k$ is the Gaussian measurement noise with zero mean and covariance matrix $\mathbf{R}_k$.

Several strategies have been developed to obtain the discretized state-space representation of dynamical systems given by Eq. (1), such as the zero-order-hold sampling technique [29], the Newmark integration scheme [39] or the Runge-Kutta method [40]. Here, it has been chosen to apply the explicit generalized-α integration scheme proposed by Aucejo et al. in Ref. [41] to a modally reduced order model of the considered dynamical system.

From a Bayesian perspective, the previous discretized state-space representation can be expressed as [17]:

$$\begin{cases} \mathbf{x}_{k+1} \sim p(\mathbf{x}_{k+1}|\mathbf{x}_k, \mathbf{u}_k) = \mathcal{N}(\mathbf{x}_{k+1}|\mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k, \mathbf{Q}_k^{\mathbf{x}}) \\ \mathbf{y}_k \sim p(\mathbf{y}_k|\mathbf{x}_k, \mathbf{u}_k) = \mathcal{N}(\mathbf{y}_k|\mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k, \mathbf{R}_k) \end{cases}, \tag{2}$$

where $\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ is the multivariate normal distribution with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$ associated to the random vector $\mathbf{x}$.

3. Bayesian formulation of the joint input-state estimation problem

This section introduces the Bayesian formulation at the roots of the Augmented Kalman Filter (AKF), which allows the joint estimation of the state and input vectors of a linear and time invariant dynamical system.

3.1. Probabilistic augmented state-space model

In the joint input-state estimation problem, the state vector $\mathbf{x}_k$ and the input vector $\mathbf{u}_k$ are computed jointly in a statistical sense. Such an estimation process requires the introduction of an augmented state $\mathbf{x}_k^{\mathbf{a}}$ defined such that:

$$\mathbf{x}_k^{\mathbf{a}} = \begin{bmatrix} \mathbf{x}_k \\ \mathbf{u}_k \end{bmatrix}. \tag{3}$$

However, to form the corresponding joint state-space representation, one has to introduce a fictitious state equation, describing the evolution of the input vector $\mathbf{u}_k$ from one time step to another. Generally, a random walk model is assumed, namely:

$$\mathbf{u}_{k+1} \sim p(\mathbf{u}_{k+1}|\mathbf{u}_k) = \mathcal{N}(\mathbf{u}_{k+1}|\mathbf{u}_k, \mathbf{Q}_k^{\mathbf{u}}), \tag{4}$$

implying that [29]:

$$\mathbf{u}_{k+1} = \mathbf{u}_k + \mathbf{w}_k^{\mathbf{u}}, \tag{5}$$

where $\mathbf{w}_k^{\mathbf{u}}$ is a Gaussian process noise with zero mean and covariance matrix $\mathbf{Q}_k^{\mathbf{u}}$. In doing so, the following probabilistic augmented state-space model is obtained:

$$\begin{cases} \mathbf{x}_{k+1}^{\mathbf{a}} \sim p(\mathbf{x}_{k+1}^{\mathbf{a}}|\mathbf{x}_k^{\mathbf{a}}) = \mathcal{N}(\mathbf{x}_{k+1}^{\mathbf{a}}|\mathbf{A}_{\mathbf{a}}\mathbf{x}_k^{\mathbf{a}}, \mathbf{Q}_k^{\mathbf{a}}) \\ \mathbf{y}_k \sim p(\mathbf{y}_k|\mathbf{x}_k^{\mathbf{a}}) = \mathcal{N}(\mathbf{y}_k|\mathbf{C}_{\mathbf{a}}\mathbf{x}_k^{\mathbf{a}}, \mathbf{R}_k) \end{cases}, \tag{6}$$

where

$$\mathbf{A}_{\mathbf{a}} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}, \quad \mathbf{C}_{\mathbf{a}} = \begin{bmatrix} \mathbf{C} & \mathbf{D} \end{bmatrix}, \quad \mathbf{Q}_k^{\mathbf{a}} = \begin{bmatrix} \mathbf{Q}_k^{\mathbf{x}} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_k^{\mathbf{u}} \end{bmatrix}, \tag{7}$$

where $\mathbf{I}$ is the identity matrix.

3.2. AKF revisited from the Bayesian perspective

The previous section suggests that the state vector and input vector can be jointly estimated from a standard Kalman Filter, which is known as the Augmented Kalman Filter (AKF) in the dedicated literature [29]. A standard Kalman Filter is a Bayesian Filter, whose purpose is to estimate the state vector $\mathbf{x}_k^{\mathbf{a}}$ given all the observations made up to the current time step k and denoted $\mathbf{y}_{1:k} = \{\mathbf{y}_1, \ldots, \mathbf{y}_k\}$. In probabilistic terms, the previous statement corresponds to the computation of the following filtering probability distribution:

$$p(\mathbf{x}_k^{\mathbf{a}}|\mathbf{y}_{1:k}) = \mathcal{N}(\mathbf{x}_k^{\mathbf{a}}|\widehat{\mathbf{x}}_k^{\mathbf{a}}, \mathbf{P}_k^{\mathbf{x}}). \tag{8}$$

To this end, the Bayesian Filter proceeds in three steps. After an initialization phase defining the prior distribution on the initial state vector, namely $p(\mathbf{x}_0^{\mathbf{a}})$, the second step consists, at some time step k, in computing the predictive probability distribution $p(\mathbf{x}_k^{\mathbf{a}}|\mathbf{y}_{1:k-1})$ corresponding to the prediction of the state vector $\mathbf{x}_k^{\mathbf{a}}$ at the time step k given all the observations made up to the previous time step $k-1$. Then, the predicted state vector is updated from the Bayes' rule to take into account the current observation $\mathbf{y}_k$. It results that the Bayesian formulation of the standard Kalman Filter can be summarized as follows [17]:

1. Initialization at $k = 0$

$$p(\mathbf{x}_0^{\mathbf{a}}) = \mathcal{N}(\mathbf{x}_0^{\mathbf{a}}|\widehat{\mathbf{x}}_0^{\mathbf{a}}, \mathbf{P}_0^{\mathbf{x}}), \tag{9}$$

   where the mean vector $\widehat{\mathbf{x}}_0^{\mathbf{a}}$ and the covariance matrix $\mathbf{P}_0^{\mathbf{x}}$ are known quantities.

2. Prediction step at time step k

$$\begin{aligned} p(\mathbf{x}_k^{\mathbf{a}}|\mathbf{y}_{1:k-1}) &= \int_{\mathbf{x}_k^{\mathbf{a}}} p(\mathbf{x}_k^{\mathbf{a}}|\mathbf{x}_{k-1}^{\mathbf{a}})\, p(\mathbf{x}_{k-1}^{\mathbf{a}}|\mathbf{y}_{1:k-1})\, d\mathbf{x}_{k-1}^{\mathbf{a}} \\ &= \mathcal{N}(\mathbf{x}_k^{\mathbf{a}}|\widetilde{\mathbf{x}}_k^{\mathbf{a}}, \widetilde{\mathbf{P}}_k^{\mathbf{x}}), \end{aligned} \tag{10}$$

where $\widetilde{\mathbf{x}}_k^{\mathbf{a}} = \mathbf{A}_{\mathbf{a}}\widehat{\mathbf{x}}_{k-1}^{\mathbf{a}}$ and $\widetilde{\mathbf{P}}_k^{\mathbf{x}} = \mathbf{A}_{\mathbf{a}}\mathbf{P}_{k-1}^{\mathbf{x}}\mathbf{A}_{\mathbf{a}}^{\mathsf{T}} + \mathbf{Q}_{k-1}^{\mathbf{a}}$.

3. Estimation step at time step k

$$\begin{aligned} p(\mathbf{x}_k^{\mathbf{a}}|\mathbf{y}_{1:k}) &\propto p(\mathbf{y}_k|\mathbf{x}_k^{\mathbf{a}})\, p(\mathbf{x}_k^{\mathbf{a}}|\mathbf{y}_{1:k-1}) \\ &= \mathcal{N}(\mathbf{x}_k^{\mathbf{a}}|\widehat{\mathbf{x}}_k^{\mathbf{a}}, \mathbf{P}_k^{\mathbf{x}}), \end{aligned} \tag{11}$$

where $\widehat{\mathbf{x}}_k^{\mathbf{a}} = \widetilde{\mathbf{x}}_k^{\mathbf{a}} + \mathbf{K}_k^{\mathbf{x}}(\mathbf{y}_k - \mathbf{C}_{\mathbf{a}}\,\widetilde{\mathbf{x}}_k^{\mathbf{a}})$ and $\mathbf{P}_k^{\mathbf{x}} = (\mathbf{I} - \mathbf{K}_k^{\mathbf{x}}\mathbf{C}_{\mathbf{a}})\widetilde{\mathbf{P}}_k^{\mathbf{x}}$. In the foregoing, $\mathbf{K}_k^{\mathbf{x}} = \widetilde{\mathbf{P}}_k^{\mathbf{x}}\mathbf{C}_{\mathbf{a}}^{\mathsf{T}}(\mathbf{C}_{\mathbf{a}}\widetilde{\mathbf{P}}_k^{\mathbf{x}}\mathbf{C}_{\mathbf{a}}^{\mathsf{T}} + \mathbf{R}_k)^{-1}$ is known as the Kalman gain.

The interested reader will find the resulting computational algorithm corresponding to the Augmented Kalman Filter in Appendix A.1.

4. Bayesian formulation of the sequential input-state estimation problem

This section extends the Bayesian formulation described previously to deal with the sequential input-state estimation problem. In particular, a Bayesian formulation, inspired by the work of Sedehi et al. [35], is first proposed. Then, we demonstrate how this general formulation allows recovering the main state-of-the-art sequential filters, i.e. GDF, SBF and DKF.

4.1. General Bayesian formulation

In the sequential input-state estimation problem, the state vector $\mathbf{x}_k$ and the input vector $\mathbf{u}_k$ are computed sequentially, meaning that the estimation of both vectors are not performed simultaneously. From a very general

standpoint, this allows using the state-space representation given in Eq. (1) directly. For this purpose, the Bayesian formulation presented in section 3 to derive the standard Kalman Filter needs to be extended to account for the sequential nature of the filtering process. More specifically, this implies the introduction of predictive and filtering distributions associated to the input vector, namely $p(\mathbf{u}_k|\mathbf{y}_{1:k-1})$ and $p(\mathbf{u}_k|\mathbf{y}_{1:k})$. A careful analysis of the existing literature shows that the Bayesian formulation of the sequential input-state estimation problem can be divided into the following five steps:

1. Initialization at $k = 0$

   The initialization of the input and state vectors consists in defining the prior probability distributions over the initial input vector, $\mathbf{u}_0$, and the initial state vector, $\mathbf{x}_0$. Here, these prior probability distributions are defined as follows:

   $$p(\mathbf{u}_0) = \mathcal{N}(\mathbf{u}_0|\widehat{\mathbf{u}}_0, \mathbf{P}_0^{\mathbf{u}}) \quad \text{and} \quad p(\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_0|\widehat{\mathbf{x}}_0, \mathbf{P}_0^{\mathbf{x}}), \tag{12}$$

   where the mean vectors, $\widehat{\mathbf{u}}_0$ and $\widehat{\mathbf{x}}_0$, and the covariance matrices, $\mathbf{P}_0^{\mathbf{u}}$ and $\mathbf{P}_0^{\mathbf{x}}$, are known quantities.

   To complete the initialization step, one has to compute the predictive distribution $p(\mathbf{x}_1|\mathbf{y}_0)$. Assuming that the initial input and state vectors are statistically independent, one has:

   $$\begin{aligned} p(\mathbf{x}_1|\mathbf{y}_0) &= \int_{\mathbf{x}_0}\int_{\mathbf{u}_0} p(\mathbf{x}_1, \mathbf{x}_0, \mathbf{u}_0|\mathbf{y}_0)\, d\mathbf{x}_0\, d\mathbf{u}_0 \\ &= \int_{\mathbf{x}_0}\int_{\mathbf{u}_0} p(\mathbf{x}_1|\mathbf{x}_0, \mathbf{u}_0)\, p(\mathbf{x}_0)\, p(\mathbf{u}_0)\, d\mathbf{x}_0\, d\mathbf{u}_0 \\ &= \mathcal{N}(\mathbf{x}_1|\widetilde{\mathbf{x}}_1, \widetilde{\mathbf{P}}_1^{\mathbf{x}}), \end{aligned} \tag{13}$$

where $\widetilde{\mathbf{x}}_1 = \mathbf{A}\widehat{\mathbf{x}}_0 + \mathbf{B}\widehat{\mathbf{u}}_0$ and $\widetilde{\mathbf{P}}_1^{\mathbf{x}} = \mathbf{A}\mathbf{P}_0^{\mathbf{x}}\mathbf{A}^\mathsf{T} + \mathbf{B}\mathbf{P}_0^{\mathbf{u}}\mathbf{B}^\mathsf{T} + \mathbf{Q}_0$.

2. Prediction of the input vectors at time step k

   The prediction of the input vector relies on the definition of the predictive distribution $p(\mathbf{u}_k|\mathbf{y}_{1:k-1})$. However, as said in the preamble of this section, sequential input-state estimation methods are based on the state-space representation given by Eq. (1). It results that the predictive distribution $p(\mathbf{u}_k|\mathbf{y}_{1:k-1})$ remains unknown if no assumption is made on the shape or the evolution through time of the input vector. Consequently, some hypotheses must be made on this probability distribution in order to make the sequential input-state estimation possible. This is generally at this stage that the state-of-the-art sequential filters differ. To make the sequential Bayesian Filter rather general regarding the existing literature, it is assumed without loss of generality that the predictive probability distribution is a multivariate Gaussian distribution with mean $\widetilde{\mathbf{u}}_k$ and covariance matrix $\widetilde{\mathbf{P}}_k^{\mathbf{u}}$, that is:

$$p(\mathbf{u}_k|\mathbf{y}_{1:k-1}) = \mathcal{N}(\mathbf{u}_k|\widetilde{\mathbf{u}}_k, \widetilde{\mathbf{P}}_k^{\mathbf{u}}). \tag{14}$$

3. Estimation of the input vectors at time step k

   The estimation of the input vector requires the computation of the filtering probability distribution $p(\mathbf{u}_k|\mathbf{y}_{1:k})$, corresponding to the following marginal distribution:

$$p(\mathbf{u}_k|\mathbf{y}_{1:k}) = \int_{\mathbf{x}_k} p(\mathbf{u}_k|\mathbf{x}_k, \mathbf{y}_{1:k})\, p(\mathbf{x}_k|\mathbf{y}_{1:k})\, d\mathbf{x}_k. \tag{15}$$

   Unfortunately, the previous probability distribution can't be estimated directly, because the filtering distribution over the state vector $p(\mathbf{x}_k|\mathbf{y}_{1:k})$

is unknown at this stage. To pursue the filtering process, the filtering distribution is approximated by the predictive one, namely $p(\mathbf{x}_k|\mathbf{y}_{1:k-1})$. Hence, by making the previous approximation and applying the Bayes' rule to $p(\mathbf{u}_k|\mathbf{x}_k, \mathbf{y}_{1:k})$, the filtering probability distribution over the input vector becomes:

$$\begin{aligned} p(\mathbf{u}_k|\mathbf{y}_{1:k}) &\propto \int_{\mathbf{x}_k} p(\mathbf{y}_k|\mathbf{x}_k, \mathbf{u}_k)\, p(\mathbf{u}_k|\mathbf{y}_{1:k-1})\, p(\mathbf{x}_k|\mathbf{y}_{1:k-1})\, d\mathbf{x}_k \\ &= \mathcal{N}(\mathbf{u}_k|\widehat{\mathbf{u}}_k, \mathbf{P}_k^{\mathbf{u}}), \end{aligned} \tag{16}$$

where

$$\widehat{\mathbf{u}}_k = \widetilde{\mathbf{u}}_k + \mathbf{K}_k^{\mathbf{u}}(\mathbf{y}_k - \mathbf{C}\widetilde{\mathbf{x}}_k - \mathbf{D}\widetilde{\mathbf{u}}_k), \tag{17a}$$

$$\mathbf{P}_k^{\mathbf{u}} = (\mathbf{I} - \mathbf{K}_k^{\mathbf{u}}\mathbf{D})\widetilde{\mathbf{P}}_k^{\mathbf{u}} + \mathbf{K}_k^{\mathbf{u}}\mathbf{C}\widetilde{\mathbf{P}}_k^{\mathbf{x}}\mathbf{C}^{\mathsf{T}}\mathbf{K}_k^{\mathbf{u}\mathsf{T}}, \tag{17b}$$

$$\mathbf{K}_k^{\mathbf{u}} = \widetilde{\mathbf{P}}_k^{\mathbf{u}}\mathbf{D}^{\mathsf{T}}(\mathbf{D}\widetilde{\mathbf{P}}_k^{\mathbf{u}}\mathbf{D}^{\mathsf{T}} + \mathbf{R}_k)^{-1}. \tag{17c}$$

4. Estimation of the state vector at time step k

   As for the input vector, the estimation of the state vector requires the computation of the filtering probability distribution $p(\mathbf{x}_k|\mathbf{y}_{1:k})$, corresponding to the following marginal distribution:

$$\begin{aligned} p(\mathbf{x}_k|\mathbf{y}_{1:k}) &= \int_{\mathbf{u}_k} p(\mathbf{x}_k|\mathbf{u}_k, \mathbf{y}_{1:k})\, p(\mathbf{u}_k|\mathbf{y}_{1:k})\, d\mathbf{u}_k \\ &\propto \int_{\mathbf{u}_k} p(\mathbf{y}_k|\mathbf{x}_k, \mathbf{u}_k)\, p(\mathbf{x}_k|\mathbf{y}_{1:k-1})\, p(\mathbf{u}_k|\mathbf{y}_{1:k})\, d\mathbf{u}_k \\ &= \mathcal{N}(\mathbf{x}_k|\widehat{\mathbf{x}}_k, \mathbf{P}_k^{\mathbf{x}}), \end{aligned} \tag{18}$$

   where

$$\widehat{\mathbf{x}}_k = \widetilde{\mathbf{x}}_k + \mathbf{K}_k^{\mathbf{x}}(\mathbf{y}_k - \mathbf{C}\widetilde{\mathbf{x}}_k - \mathbf{D}\widehat{\mathbf{u}}_k), \tag{19a}$$

$$\mathbf{P}_k^{\mathbf{x}} = (\mathbf{I} - \mathbf{K}_k^{\mathbf{x}}\mathbf{C})\widetilde{\mathbf{P}}_k^{\mathbf{x}} + \mathbf{K}_k^{\mathbf{x}}\mathbf{D}\mathbf{P}_k^{\mathbf{u}}\mathbf{D}^{\mathsf{T}}\mathbf{K}_k^{\mathbf{x}\mathsf{T}}, \tag{19b}$$

$$\mathbf{K}_k^{\mathbf{x}} = \widetilde{\mathbf{P}}_k^{\mathbf{x}}\mathbf{C}^{\mathsf{T}}(\mathbf{C}\widetilde{\mathbf{P}}_k^{\mathbf{x}}\mathbf{C}^{\mathsf{T}} + \mathbf{R}_k)^{-1}. \tag{19c}$$

At this stage, it is possible to compute the cross-covariance matrix $\mathbf{P}_k^{\mathbf{xu}}$, which is defined such that:

$$\mathbf{P}_k^{\mathbf{xu}} = \mathbb{E}\left[(\mathbf{x}_k - \widehat{\mathbf{x}}_k)(\mathbf{u}_k - \widehat{\mathbf{u}}_k)^{\mathsf{T}}\right], \tag{20}$$

where $\mathbb{E}(\mathbf{x})$ is the expected value of the random vector $\mathbf{x}$. From Eq. (19a), it readily comes that:

$$\mathbf{P}_k^{\mathbf{xu}} = -\mathbf{K}_k^{\mathbf{x}}\mathbf{D}\mathbf{P}_k^{\mathbf{u}}. \tag{21}$$

5. Prediction of the state vector at time step $k+1$

   The last step of the sequential Bayesian Filter is the computation of the predictive distribution over the state vector at time step $k+1$, in order to continue the recursive process. The calculation of the latter probability distribution is as follows:

$$\begin{aligned} p(\mathbf{x}_{k+1}|\mathbf{y}_{1:k}) &= \int_{\mathbf{x}_k}\int_{\mathbf{u}_k} p(\mathbf{x}_{k+1}, \mathbf{x}_k, \mathbf{u}_k|\mathbf{y}_k)\, d\mathbf{x}_k\, d\mathbf{u}_k \\ &= \int_{\mathbf{x}_k}\int_{\mathbf{u}_k} p(\mathbf{x}_{k+1}|\mathbf{x}_k, \mathbf{u}_k)\, p(\mathbf{x}_k, \mathbf{u}_k|\mathbf{y}_{1:k})\, d\mathbf{x}_k\, d\mathbf{u}_k. \end{aligned} \tag{22}$$

   In the previous equation, $p(\mathbf{x}_k, \mathbf{u}_k|\mathbf{y}_{1:k})$ is the joint filtering probability distribution of the state and input vectors. Formally, it is expressed as:

$$p(\mathbf{x}_k, \mathbf{u}_k|\mathbf{y}_{1:k}) = p(\overline{\mathbf{x}}_k|\mathbf{y}_{1:k}) = \mathcal{N}(\overline{\mathbf{x}}_k|\widehat{\overline{\mathbf{x}}}_k, \overline{\mathbf{P}}_k), \tag{23}$$

   where

$$\widehat{\overline{\mathbf{x}}}_k = \begin{bmatrix} \widehat{\mathbf{x}}_k \\ \widehat{\mathbf{u}}_k \end{bmatrix} \quad \text{and} \quad \overline{\mathbf{P}}_k = \begin{bmatrix} \mathbf{P}_k^{\mathbf{x}} & \mathbf{P}_k^{\mathbf{xu}} \\ \mathbf{P}_k^{\mathbf{xu}^{\mathsf{T}}} & \mathbf{P}_k^{\mathbf{u}} \end{bmatrix}. \tag{24}$$

   Including the previous result in Eq. (22), one finally gets:

$$p(\mathbf{x}_{k+1}|\mathbf{y}_{1:k}) = \mathcal{N}(\mathbf{x}_{k+1}|\widetilde{\mathbf{x}}_{k+1}, \widetilde{\mathbf{P}}_{k+1}^{\mathbf{x}}), \tag{25}$$

where

$$\widetilde{\mathbf{x}}_{k+1} = \mathbf{A}\widehat{\mathbf{x}}_k + \mathbf{B}\widehat{\mathbf{u}}_k, \tag{26a}$$

$$\widetilde{\mathbf{P}}^{\mathbf{x}}_{k+1} = \begin{bmatrix}\mathbf{A} & \mathbf{B}\end{bmatrix} \begin{bmatrix}\mathbf{P}^{\mathbf{x}}_k & \mathbf{P}^{\mathbf{xu}}_k \\ \mathbf{P}^{\mathbf{xu}\mathsf{T}}_k & \mathbf{P}^{\mathbf{u}}_k\end{bmatrix} \begin{bmatrix}\mathbf{A}^{\mathsf{T}} \\ \mathbf{B}^{\mathsf{T}}\end{bmatrix} + \mathbf{Q}_k. \tag{26b}$$

4.2. Application to the main state of the art sequential filters

This section aims at demonstrating how the main state-of-the-art sequential filters, namely GDF, SBF and DKF, can all be derived from the general Bayesian formulation described in section 4.1. More specifically, the main hypotheses made for obtaining each filter are carefully explained in the light of the Bayesian interpretation.

4.2.1. Gillijns and De Moor Filter

In the Gillijns and De Moor Filter (GDF) introduced in Ref. [27], the authors implicitly state that all the predicted input vectors are equiprobable, thus reflecting the lack of prior knowledge of the spatial distribution of the input vector at some time step k. This assumption can be encoded by assuming that the predicted input vector follows a uniform distribution, that is:

$$\begin{aligned} p(\mathbf{u}_k|\mathbf{y}_{1:k-1}) &= \mathcal{U}(-a, a) \text{ with } a \text{ sufficiently large} \\ &\propto 1 \end{aligned} \tag{27}$$

The introduction of this predictive probability distribution into the general Bayesian formulation detailed in section 4.1 leads to a change in the

mathematical expression of the filtering probability distribution over the input vector, which becomes:

$$p(\mathbf{u}_k|\mathbf{y}_{1:k}) = \mathcal{N}(\mathbf{u}_k|\widehat{\mathbf{u}}_k, \mathbf{P}_k^{\mathbf{u}}), \tag{28}$$

where

$$\widehat{\mathbf{u}}_k = \mathbf{K}_k^{\mathbf{u}}(\mathbf{y}_k - \mathbf{C}\widetilde{\mathbf{x}}_k), \tag{29a}$$

$$\mathbf{P}_k^{\mathbf{u}} = (\mathbf{D}\mathbf{S}_k^{-1}\mathbf{D}^{\mathsf{T}})^{-1}, \tag{29b}$$

$$\mathbf{K}_k^{\mathbf{u}} = \mathbf{P}_k^{\mathbf{u}}\mathbf{D}^{\mathsf{T}}\mathbf{S}_k^{-1}, \tag{29c}$$

$$\mathbf{S}_k = \mathbf{C}\widetilde{\mathbf{P}}_k^{\mathbf{x}}\mathbf{C}^{\mathsf{T}} + \mathbf{R}_{\mathbf{k}}. \tag{29d}$$

Except this slight modification, all the remaining probability distributions remain unchanged. The resulting computational algorithm, known as GDF, is recalled in Appendix A.2.

4.2.2. Sequential Bayesian Filter

In the Sequential Bayesian Filter (SBF) proposed by Sedehi et al. [35], the sole assumption made by the authors is that the predicted input vector follows a multivariate Gaussian distribution with zero mean and covariance matrix $\mathbf{P}_{k-1}^{\mathbf{u}}$. Mathematically, this implies that the predictive probability distribution over the input vector is written:

$$p(\mathbf{u}_k|\mathbf{y}_{1:k-1}) = \mathcal{N}(\mathbf{u}_k|\mathbf{0}, \mathbf{P}_{k-1}^{\mathbf{u}}) \tag{30}$$

At first sight, this assumption is not supported by any physical rationale. According to the authors, such a predictive probability distribution implies that the input vector is independent of its estimation at the previous time step

and only relies on the related estimated covariance matrix [35]. The practical implication of this assumption on the whole filtering process is hard to assess. In addition to this assumption, the authors propose an additional trick to limit the drift effect. They argue that the drift effect is caused by the use of erroneous input estimations, that lead to errors accumulation with time [35]. To mitigate this adverse effect, the influence of the estimated input vector is removed from the predicted state vector only for estimating the input vector at the next time step. In other words, this means that Eq. (17a) becomes:

$$\widehat{\mathbf{u}}_k = \mathbf{K}_k^{\mathbf{u}}(\mathbf{y}_k - \mathbf{C}\mathbf{A}\widehat{\mathbf{x}}_{k-1}), \tag{31}$$

because $\widetilde{\mathbf{u}}_k = \mathbf{0}$ according to Eq. (30).

Based on the previous assumptions, one derives the computational form of SBF presented in Appendix A.3.

4.2.3. Dual Kalman Filter

As said in the introduction, the Dual Kalman Filter performs the input-state estimation by running two standard Kalman Filters in a sequential way. Practically, DKF makes first use of the fictitious state equation for the input vector defined in Eq. (4). From this, the predictive probability distribution for the input vector can be derived as follows:

$$\begin{aligned} p(\mathbf{u}_k|\mathbf{y}_{1:k-1}) &= \int_{\mathbf{u}_{k-1}} p(\mathbf{u}_k|\mathbf{u}_{k-1})\, p(\mathbf{u}_{k-1}|\mathbf{y}_{1:k-1})\, d\mathbf{u}_{k-1} \\ &= \mathcal{N}(\mathbf{u}_k|\widetilde{\mathbf{u}}_k, \widetilde{\mathbf{P}}_k^{\mathbf{u}}), \end{aligned} \tag{32}$$

where $\widetilde{\mathbf{u}}_k = \widehat{\mathbf{u}}_{k-1}$ and $\widetilde{\mathbf{P}}_k^{\mathbf{u}} = \mathbf{P}_{k-1}^{\mathbf{u}} + \mathbf{Q}_{k-1}^{\mathbf{u}}$.

In order to run two standard Kalman Filters sequentially, a set of assumptions must be introduced in the next steps of the filtering process. First,

the filtering distribution over the input vector is computed assuming that the predicted state vector $\widetilde{\mathbf{x}}_k$ is perfectly known and equal the estimated state vector at the previous time step $\widehat{\mathbf{x}}_{k-1}$. It results that the predictive distribution over the state vector defined in Eq. (16) should be such that $p(\mathbf{x}_k|\mathbf{y}_{1:k-1}) = \delta(\mathbf{x}_k - \widehat{\mathbf{x}}_{k-1})$. In doing so, one has:

$$p(\mathbf{u}_k|\mathbf{y}_{1:k}) = \mathcal{N}(\mathbf{u}_k|\widehat{\mathbf{u}}_k, \mathbf{P}_k^{\mathbf{u}}), \tag{33}$$

where $\widehat{\mathbf{u}}_k = \widetilde{\mathbf{u}}_k + \mathbf{K}_k^{\mathbf{u}}(\mathbf{y}_k - \mathbf{C}\widehat{\mathbf{x}}_{k-1} - \mathbf{D}\widetilde{\mathbf{u}}_k)$ and $\mathbf{P}_k^{\mathbf{u}} = (\mathbf{I} - \mathbf{K}_k^{\mathbf{u}}\mathbf{D})\widetilde{\mathbf{P}}_k^{\mathbf{u}}$. Here, the expression of the Kalman gain $\mathbf{K}_k^{\mathbf{u}}$ remains unchanged (see Eq. (17c)).

Second, the predictive distribution over the state vector at time k is calculated by assuming that the input and state vectors are uncorrelated (which is consistent with the previous assumptions) and that the estimated input vector at time step k is perfectly known. Hence, it becomes:

$$\begin{aligned} p(\mathbf{x}_k|\mathbf{y}_{1:k-1}) &= \int_{\mathbf{x}_{k-1}} p(\mathbf{x}_k|\mathbf{x}_{k-1}, \widehat{\mathbf{u}}_k)\, p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k-1})\, d\mathbf{x}_{k-1} \\ &= \mathcal{N}(\mathbf{x}_k|\widetilde{\mathbf{x}}_k, \widetilde{\mathbf{P}}_k^{\mathbf{x}}), \end{aligned} \tag{34}$$

where $\widetilde{\mathbf{x}}_k = \mathbf{A}\widehat{\mathbf{x}}_{k-1} + \mathbf{B}\widehat{\mathbf{u}}_k$, while $\widetilde{\mathbf{P}}_{k+1}^{\mathbf{x}} = \mathbf{A}\mathbf{P}_{k-1}^{\mathbf{x}}\mathbf{A}^{\mathsf{T}} + \mathbf{Q}_{k-1}^{\mathbf{x}}$.

Finally, the filtering distribution over the state vector is computed assuming that the estimated input vector $\widehat{\mathbf{u}}_k$ is perfectly known, i.e. $p(\mathbf{u}_k|\mathbf{y}_{1:k}) = \delta(\mathbf{u}_k - \widehat{\mathbf{u}}_k)$. Thus, it induces:

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) = \mathcal{N}(\mathbf{x}_k|\widehat{\mathbf{x}}_k, \mathbf{P}_k^{\mathbf{x}}), \tag{35}$$

where $\widehat{\mathbf{x}}_k = \widetilde{\mathbf{x}}_k + \mathbf{K}_k^{\mathbf{x}}(\mathbf{y}_k - \mathbf{C}\widetilde{\mathbf{x}}_k - \mathbf{D}\widehat{\mathbf{u}}_k)$ and $\mathbf{P}_k^{\mathbf{x}} = (\mathbf{I} - \mathbf{K}_k^{\mathbf{x}}\mathbf{C})\widetilde{\mathbf{P}}_k^{\mathbf{x}}$. Here, the expression of the Kalman gain $\mathbf{K}_k^{\mathbf{x}}$ remains unchanged (see Eq. (19c)).

The implementation of resulting filter, called DKF, is fully given in Appendix A.4.

5. Sparse adaptive Bayesian Filter

In the previous section, a general formulation of the sequential input-state estimation problem has been derived. It has also been shown that this formulation allows recovering most of the state-of-the-art filters. More precisely, a particular focus has been made on the assumptions at the core of each filter. The proposed derivation from the Bayesian formalism has demonstrated that the sequential Kalman-like filters existing in the literature used a different assumption to define the predictive probability distribution over the input vector $p(\mathbf{u}_k|\mathbf{y}_{1:k-1})$. This observation is actually at the root of the Sparse adaptive Bayesian Filter (SaBF), which aims at introducing in the formulation some kind of prior information on the spatial distribution of the input vector.

5.1. Mathematical intuition

Before detailing the derivation of SaBF, it is important to understand the mathematical intuition behind it. Actually, everything starts from the expression of the mean of the estimated input vector given by Eq. (17a), which can be rewritten as:

$$\widehat{\mathbf{u}}_k = \widetilde{\mathbf{u}}_k + \widetilde{\mathbf{P}}^{\mathbf{u}}_k \mathbf{D}^\mathsf{T} (\mathbf{D}\widetilde{\mathbf{P}}^{\mathbf{u}}_k \mathbf{D}^\mathsf{T} + \mathbf{R}_k)^{-1} (\mathbf{i}_k - \mathbf{D}\widetilde{\mathbf{u}}_k) \tag{36}$$

where $\mathbf{i}_k = \mathbf{y}_k - \mathbf{C}\widetilde{\mathbf{x}}_k$ is the innovation vector, which is a measure of the information brought by a new measurement.

From the following matrix inversion lemma:

$$\mathbf{B}\mathbf{A}^{\mathsf{T}}(\mathbf{A}\mathbf{B}\mathbf{A}^{\mathsf{T}}+\mathbf{C})^{-1}=(\mathbf{A}\mathbf{C}^{-1}\mathbf{A}^{\mathsf{T}}+\mathbf{B}^{-1})^{-1}\mathbf{A}^{\mathsf{T}}\mathbf{C}^{-1}, \tag{37}$$

Eq. (36) becomes:

$$\widehat{\mathbf{u}}_k=\widetilde{\mathbf{u}}_k+\left(\mathbf{D}^{\mathsf{T}}\mathbf{R}_k^{-1}\mathbf{D}+\left[\widetilde{\mathbf{P}}_k^{\mathbf{u}}\right]^{-1}\right)^{-1}\mathbf{D}^{\mathsf{T}}\mathbf{R}_k^{-1}(\mathbf{i}_k-\mathbf{D}\widetilde{\mathbf{u}}_k). \tag{38}$$

Readers, well-versed in regularization techniques, can immediately observe that Eq. (38) is actually the solution of the following minimization problem:

$$\widehat{\mathbf{u}}_k=\underset{\mathbf{u}_k}{\operatorname{argmin}}\,\frac{1}{2}\|\mathbf{i}_k-\mathbf{D}\mathbf{u}_k\|^2_{\mathbf{R}_k}+\frac{1}{2}\|\mathbf{u}_k-\widetilde{\mathbf{u}}_k\|^2_{\widetilde{\mathbf{P}}_k^{\mathbf{u}}}, \tag{39}$$

where $\|\mathbf{x}\|^2_{\mathbf{Q}}=\mathbf{x}^{\mathsf{T}}\mathbf{Q}^{-1}\mathbf{x}$ is the squared Mahalanobis distance.

From a Bayesian perspective, the solution of Eq. (39) is the Maximum A Posteriori (MAP) estimator of the following posterior probability distribution:

$$p(\mathbf{u}_k|\mathbf{i}_k)\propto p(\mathbf{i}_k|\mathbf{u}_k)\,p(\mathbf{u}_k), \tag{40}$$

where the likelihood function $p(\mathbf{i}_k|\mathbf{u}_k)$ and the prior distribution $p(\mathbf{u}_k)$ are defined such that:

$$p(\mathbf{i}_k|\mathbf{u}_k)=\mathcal{N}(\mathbf{i}_k|\mathbf{D}\mathbf{u}_k,\mathbf{R}_k)\;\;\text{and}\;\;p(\mathbf{u}_k)=\mathcal{N}(\mathbf{u}_k|\widetilde{\mathbf{u}}_k,\widetilde{\mathbf{P}}_k^{\mathbf{u}}). \tag{41}$$

The mathematical equivalence between Eqs. (36), (39) and (40) demonstrates that some prior knowledge of the spatial distribution of the input vector can be easily added in the formulation of the general sequential Bayesian Filter, introduced in section 4.1, through an adequate choice of the prior distribution $p(\mathbf{u}_k)$, corresponding to the predictive distribution $p(\mathbf{u}_k|\mathbf{y}_{1:k-1})$

in the Bayesian filtering formalism. This observation gives rise to the Sparse adaptive Bayesian Filter derived in the next section.

5.2. Formal derivation

The basic idea of the Sparse adaptive Bayesian Filter (SaBF) is to introduce some prior knowledge of the spatial distribution of the input vector. As explained in the previous section, this can be done by wisely choosing the predictive distribution $p(\mathbf{u}_k|\mathbf{y}_{1:k-1})$.

5.2.1. Preliminaries

In SaBF, this probability distribution is chosen so as to promote either the sparsity or the smoothness of the spatial distribution of the input vector. Practically, a probability distribution that meets this requirement is the univariate generalized Gaussian distribution [42]. If we further assume that the components of the input vector are independent and identically distributed, the predictive distribution over the input vector is thus chosen as the multivariate generalized Gaussian distribution with zero mean:

$$\begin{aligned} p(\mathbf{u}_k|\mathbf{y}_{1:k-1}) &= \mathcal{N}_g(\mathbf{u}_k|\mathbf{0}, \tau_k, q_k) \\ &= \left(\frac{q_k^{1-\frac{1}{q_k}}}{2\Gamma(\frac{1}{q_k})} \right)^{N_u} \tau_k^{\frac{N_u}{q_k}} \exp\left(-\frac{\tau_k}{q_k} \|\mathbf{u}_k\|_{q_k}^{q_k} \right), \end{aligned} \tag{42}$$

where:

- q_k is the shape parameter of the distribution at the time step k. Its value is defined in $\mathbb{R}^{+*}$;
- $\| \bullet \|_q$ is the ℓ_q-norm ($q \geq 1$) or quasi-norm ($q < 1$);

- τ_k is the scale parameter of the distribution at the time step k;
- N_u is the number of components of the input vector;
- $\Gamma(x)$ is the gamma function.

It is worth mentioning here that the choice of the previous multivariate Gaussian distribution offers some flexibility for encoding one's prior knowledge of the spatial distribution of the input vector, since it allows enforcing the sparsity of the input vector when $q_k \leq 1$ or its smoothness when $q_k = 2$ [43].

To comply with the Bayesian formulation presented in section 4.1, it must be noted that the following Gaussian approximation [44]:

$$\mathcal{N}_g(\mathbf{u}_k|\mathbf{0}, \tau_k, q_k) \;\propto\; \mathcal{N}(\mathbf{u}_k|\mathbf{0}, \mathbf{W}_k^{-1}/\tau_k), \tag{43}$$

holds for an adequate choice of the scale and shape parameters (τ_k, q_k) and the matrix $\mathbf{W}$. The latter matrix has to satisfy the Mercer's condition (positive, definite, symmetric).

From the proposed predictive distribution and the previous approximation, Eq. (38) becomes:

$$\widehat{\mathbf{u}}_k = (\mathbf{D}\mathbf{R}_k^{-1}\mathbf{D}^\mathsf{T} + \tau_k\mathbf{W}_k)^{-1}\mathbf{D}^\mathsf{T}\mathbf{R}_k^{-1}\,\mathbf{i}_k. \tag{44}$$

The latter equation allows identifying the Kalman gain $\mathbf{K}_k^{\mathbf{u}}$ and the predictive covariance matrix $\widetilde{\mathbf{P}}_k^{\mathbf{u}}$ as:

$$\mathbf{K}_k^{\mathbf{u}} = (\mathbf{D}\mathbf{R}_k^{-1}\mathbf{D}^\mathsf{T} + \tau_k\mathbf{W}_k)^{-1}\mathbf{D}^\mathsf{T}\mathbf{R}_k^{-1} \quad \text{and} \quad \widetilde{\mathbf{P}}_k^{\mathbf{u}} = (\tau_k\mathbf{W}_k)^{-1}. \tag{45}$$

At this stage, it remains to determine the scale and shape parameters (τ_k, q_k) and the matrix $\mathbf{W}_k$ satisfying the Gaussian approximation. To this end, a Bayesian optimization is proposed to determine at each time step the most probable values of all the parameters of the problem, including $\widehat{\mathbf{u}}_k$, given the innovation $\mathbf{i}_k$.

5.2.2. Optimal computation of $\mathbf{u}_k$*,* τ_k *and* q_k

In the present paper, the optimal parameters, $\mathbf{u}_k$, τ_k and q_k, are computed as the Maximum A Posteriori (MAP) solution of the following Bayesian optimization problem:

$$(\widehat{\mathbf{u}}_k, \widehat{\tau}_k, \widehat{q}_k) = \underset{(\mathbf{u}_k, \tau_k, q_k)}{\operatorname{argmax}}\, p(\mathbf{u}_k, \tau_k, q_k | \mathbf{i}_k), \tag{46}$$

where the posterior distribution $p(\mathbf{u}_k, \tau_k, q_k|\mathbf{i}_k)$ is given by:

$$p(\mathbf{u}_k, \tau_k, q_k|\mathbf{i}_k) \propto p(\mathbf{i}_k|\mathbf{u_k})\, p(\mathbf{u}_k|\tau_k, q_k)\, p(\tau_k)\, p(q_k), \tag{47}$$

where the prior distribution $p(\mathbf{u}_k|\tau_k, q_k) = p(\mathbf{u}_k|\mathbf{y}_{1:k-1})$, while the likelihood function $p(\mathbf{i}_k|\mathbf{u}_k)$ is the multivariate Gaussian distribution with known covariance matrix $\mathbf{R}_k$ defined in Eq. (41).

The solution of the previous optimization problem can be obtained by maximizing the full conditional probability distributions associated to each parameter. In this respect, the solution of the complete Bayesian optimization problem given by Eq. (46) can be found by solving iteratively the fol-

lowing equations:

$$\widehat{\tau}_k = \underset{\tau_k}{\operatorname{argmax}}\; p(\widehat{\mathbf{u}}_k|\tau_k, q_k)\, p(\tau_k), \tag{48a}$$

$$\widehat{q}_k = \underset{q}{\operatorname{argmax}}\; p(\widehat{\mathbf{u}}_k|\tau_k, q_k)\, p(q_k), \tag{48b}$$

$$\widehat{\mathbf{u}}_k = \underset{\mathbf{u}_k}{\operatorname{argmax}}\; p(\mathbf{u}_k|\mathbf{i}_k)\, p(\mathbf{u}_k|\tau_k, q_k). \tag{48c}$$

To complete the formulation of the Bayesian optimization problem, it remains to specify the prior probability distribution over the scale and shape parameters, τ_k and q_k. To determine the adequate prior distribution $p(\tau_k)$, one has to notice that the scale parameter τ_k is a strictly positive real number. For this reason, this prior distribution is chosen as Gamma distribution, which is defined such that:

$$\mathcal{G}(\tau_k|\alpha_t, \beta_t) = \frac{\beta_t^{\alpha_t}}{\Gamma(\alpha_t)} \tau_t^{\alpha_t - 1} \exp(-\beta_t \tau_k) \quad \text{with} \quad \alpha_t > 0,\, \beta_t > 0, \tag{49}$$

where α_t and β_t are respectively the scale parameter and the rate parameter of the distribution. However, the order of magnitude of τ_t is generally unknown in practice. That is why, the prior distribution over τ_t must be weakly informative to avoid biasing the optimization process. This requirement leads us to set $\alpha_t = 1$ and $\beta_t = 10^{-18}$.

The choice of the prior probability distribution over the shape parameter q_k follows the same philosophy as that used for selecting the prior distribution of τ_k. Here, one has to note that the shape parameter is a strictly positive and bounded real number. Indeed, the existing literature shows that it is generally comprised in the interval $]0, 2]$ [43, 45]. Keeping in mind the conjugacy property, a reasonable choice is the truncated inverse Gamma distribution:

$$\mathcal{IG}_T(q_k|\alpha_q, \beta_q) \propto \mathcal{IG}(q|\alpha_q, \beta_q)\, \mathbb{I}_{[l_b, u_b]}(q_k), \tag{50}$$

where:

- $\mathcal{IG}(q_k|\alpha_q, \beta_q)$ is the inverse Gamma distribution defined as:

$$\mathcal{IG}(q_k|\alpha_q, \beta_q) = \frac{\beta_q^{\alpha_q}}{\Gamma(\alpha_q)}(1/q_k)^{\alpha_q+1}\exp(-\beta_q/q_k) \tag{51}$$

- $\mathbb{I}_{[l_b,u_b]}(q_k)$ is the truncation function restricting the support of the inverse Gamma distribution between the lower bound l_b and the upper bound u_b.

Practically, even if the support of q_k is specified in the interval $]0, 2]$, its actual value is only vaguely known. To translate this information into mathematical terms, one sets $\alpha_q = 1$, $\beta_q = 10^{-18}$ and $(l_b, u_b) = (0.01, 2)$.

Now that the problem is fully specified, the MAP solution of the systems of equations (48) can be computed. Practically, it is easier to solve the dual minimization problem, consisting in computing the parameters minimizing the opposite of the logarithm of the full conditional probability distributions. In doing so, one has:

$$\widehat{\tau}_k = \underset{\tau_k}{\operatorname{argmin}}\ \tau_k\left(\beta_t\, q_k + \|\mathbf{u}_k\|_{q_k}^{q_k}\right) - (N_u + q_k(\alpha_t - 1))\log(\tau_k), \tag{52a}$$

$$\widehat{q}_k = \underset{q}{\operatorname{argmin}}\ f(q_k|\mathbf{u}_k, \tau_k) \quad \text{for } q_k \in [l_b, u_b], \tag{52b}$$

$$\widehat{\mathbf{u}}_k = \underset{\mathbf{u}_k}{\operatorname{argmin}}\ \frac{1}{2}\|\mathbf{i}_k - \mathbf{D}\mathbf{u}_k\|_{\mathbf{R}_k}^2 + \frac{\tau_k}{q_k}\|\mathbf{u}_k\|_{q_k}^{q_k}, \tag{52c}$$

where:

$$\begin{aligned} f(q_k|\mathbf{u}_k, \tau_k) = N_u\ \log\Gamma(1/q_k) - N_u\frac{\log\tau_k}{q_k} + \frac{\tau_k\|\mathbf{u}_k\|_{q_k}^{q_k} + \beta_q}{q_k} \\ - \left(N_u\left(1 - \frac{1}{q_k}\right) - \alpha_q - 1\right)\log q_k. \end{aligned} \tag{53}$$

As already mentioned, the resolution of the previous system of equations can only be performed in an iterative manner, since the optimal value of a given parameter depends on the values of the others. It results that the resolution process at a time step k can be summarized as follows:

1. Initialization

   a. Set $\widehat{q}_k^{(0)} = \widehat{q}_{k-1}$ and $\widehat{\mathbf{u}}_k^{(0)} = \widehat{\mathbf{u}}_{k-1}$

2. **while** convergence is not reached

   a. Compute $\widehat{\tau}_k^{(i)}$ from Eq. (52a) given $\widehat{\mathbf{u}}_k^{(i-1)}$ and $\widehat{q}_k^{(i-1)}$

   b. Compute $\widehat{q}_k^{(i)}$ from Eq. (52b) given $\widehat{\mathbf{u}}_k^{(i-1)}$ and $\widehat{\tau}_k^{(i)}$

   c. Compute $\widehat{\mathbf{u}}_k^{(i)}$ from Eq. (52c) given $\widehat{\tau}_k^{(i)}$ and $\widehat{q}_k^{(i)}$

   end while

3. **return** $\widehat{\mathbf{u}}_k$, $\widehat{\tau}_k$ and $\widehat{q}_k$

The practical implementation of the previous procedure requires some comments regarding the resolution of the set of equations (52).

Computation of $\widehat{\tau}_k^{(i)}$

The computation of the optimal scale parameter $\widehat{\tau}_k^{(i)}$ is rather straightforward, since the solution of the minimization problem given by Eq. (52a) can be analytically calculated. Indeed, when applying the first-order optimality condition, one easily finds:

$$\widehat{\tau}_k^{(i)} = \frac{N_u + \widehat{q}_k^{(i-1)}(\alpha_t - 1)}{\beta_t \, \widehat{q}_k^{(i-1)} + \left\| \widehat{\mathbf{u}}_k^{(i-1)} \right\|_{\widehat{q}_k^{(i-1)}}^{\widehat{q}_k^{(i-1)}}}. \tag{54}$$

Computation of $\widehat{q}_k^{(i)}$

To compute the optimal value of the shape parameter at an iteration i, several algorithms can be used such as bound-constrained minimization methods [46, 47]. However, because the function $f\left(q_k|\widehat{\mathbf{u}}_k^{(i-1)}, \widehat{\tau}_k^{(i)}\right)$ is explicitly defined, it is computationally more efficient to apply a brute-force search approach. Consequently, it has been chosen to evaluate $f\left(q_k|\widehat{\mathbf{u}}_k^{(i-1)}, \widehat{\tau}_k^{(i)}\right)$ for a discrete set $\bar{q}_k \in [0.01, 2]$, having a resolution $\Delta\bar{q}_k = 0.02$. The optimal shape parameter $\widehat{q}_k^{(i)}$ is the value of $\bar{q}_k$ such that $f\left(\widehat{q}_k^{(i)}|\widehat{\mathbf{u}}_k^{(i-1)}, \widehat{\tau}_k^{(i)}\right)$ is minimum.

Computation of $\widehat{u}_k^{(i)}$

The optimal input vector $\widehat{\mathbf{u}}_k^{(i)}$ is solution of the following minimization problem:

$$\widehat{\mathbf{u}}_k^{(i)} = \underset{\mathbf{u}_k}{\operatorname{argmin}} \frac{1}{2}\|\mathbf{i}_k - \mathbf{D}\mathbf{u}_k\|_{\mathbf{R}_k}^2 + \frac{\widehat{\tau}_k^{(i)}}{\widehat{q}_k^{(i)}}\|\mathbf{u}_k\|_{\widehat{q}_k^{(i)}}^{\widehat{q}_k^{(i)}}. \tag{55}$$

Unfortunately, the previous minimization problem has no explicit solution and requires the implementation of an iterative resolution procedure, based on the application of the first-order optimality condition. At a given iteration j of the iterative process, one has:

$$\widehat{\mathbf{u}}_k^{(i,j)} = \left(\mathbf{D}\mathbf{R}_k^{-1}\mathbf{D}^{\mathsf{T}} + \widehat{\tau}_k^{(i)}\mathbf{W}_k^{(i,j)}\right)^{-1}\mathbf{D}^{\mathsf{T}}\mathbf{R}_k^{-1}\,\mathbf{i}_k, \tag{56}$$

where $\mathbf{W}_k^{(i,j)}$ is a positive definite diagonal weighting matrix expressed as:

$$\mathbf{W}_k^{(i,j)} = \operatorname{diag}\left(w_{k,1}^{(i,j)}, \ldots, w_{k,n}^{(i,j)}, \ldots, w_{k,N_u}^{(i,j)}\right), \tag{57}$$

where:

$$w_{k,n}^{(i,j)} = \max\left(\epsilon_i, \left|\widehat{u}_{k,n}^{(i,j-1)}\right|\right)^{\widehat{q}_k^{(i)}-2}. \tag{58}$$

In the previous equation, $\widehat{u}_{k,n}^{(i,j-1)}$ is the n^{th} component of the input vector $\widehat{\mathbf{u}}_k^{(i,j-1)}$ and ϵ_i is a small real positive number avoiding infinite weights when

$\left|\frac{\widehat{u}_{k,n}^{(i,j-1)}}{\widehat{\mathbf{u}}_k^{(i,0)}}\right| \to 0$ and $\widehat{q}_k^{(i)} < 2$. Its value is computed so that 5% of the values of $\left|\frac{\widehat{u}_{k,n}^{(i,j-1)}}{\widehat{\mathbf{u}}_k^{(i,0)}}\right|$ are less than or equal to ϵ_i.

To complete the description if this iterative sub-process, it should be mentioned that the initial solution of the iterative process is computed from Eq. (56) by taking $\mathbf{W}_k^{(i,0)} = \mathbf{I}$, while the process is converged when the relative error δ, defined such that:

$$\delta\left(\widehat{\mathbf{u}}_k^{(i,j-1)}, \widehat{\mathbf{u}}_k^{(i,j)}\right) = \frac{\left\|\widehat{\mathbf{u}}_k^{(i,j)} - \widehat{\mathbf{u}}_k^{(i,j-1)}\right\|_2^2}{\left\|\widehat{\mathbf{u}}_k^{(i,j-1)}\right\|_2^2}, \tag{59}$$

is less than some tolerance, sets here to 10^{-8}. Finally, at convergence of the iterative process after J iterations, one has:

$$\widehat{\mathbf{u}}_k^{(i)} = \widehat{\mathbf{u}}_k^{(i,J)} \quad \text{and} \quad \mathbf{W}_k^{(i)} = \mathbf{W}_k^{(i,J)}. \tag{60}$$

Convergence monitoring of the optimization process

As for the iterative computation of $\widehat{\mathbf{u}}_k^{(i)}$, the optimization process is stopped when the relative error $\delta\left(\widehat{\mathbf{u}}_k^{(i-1)}, \widehat{\mathbf{u}}_k^{(i)}\right)$ is less than 10^{-8}. At convergence of the optimization process after I iterations, one gets:

$$\widehat{q}_k = \widehat{q}_k^{(I)}, \quad \widehat{\tau}_k = \widehat{\tau}_k^{(I)}, \quad \widehat{\mathbf{u}}_k = \widehat{\mathbf{u}}_k^{(I)} \quad \text{and} \quad \mathbf{W}_k = \mathbf{W}_k^{(I)}. \tag{61}$$

As a final step, $\widehat{\tau}_k$ and $\mathbf{W}_k$ must be introduced in Eq. (45) to compute the Kalman gain $\mathbf{K}_k^{\mathbf{u}}$ and the predictive covariance matrix $\widetilde{\mathbf{P}}_k^{\mathbf{u}}$ and thus to continue the filtering process.

5.3. Computational algorithms

To provide a global overview of the proposed Sparse adaptive Bayesian Filter, the different computational steps are gathered into Algorithms 1 and 2.

More precisely, Algorithm 1 describes the general implementation of SaBF, while Algorithm 2 summarizes the nested Bayesian optimization process introduced in section 5.2.2.

Algorithm 1: Sparse adaptive Bayesian Filter – SaBF

Input: $\mathbf{y}_k$, $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$, $\widehat{\mathbf{u}}_0$, $\mathbf{P}_0^{\mathbf{u}}$, $\widehat{\mathbf{x}}_0$, $\mathbf{P}_0^{\mathbf{x}}$, $\mathbf{Q}_k^{\mathbf{x}}$, $\mathbf{R}_k$, $\widehat{q}_0$

Output: $\widehat{\mathbf{u}}_k$, $\mathbf{P}_k^{\mathbf{u}}$, $\widehat{\mathbf{x}}_k$, $\mathbf{P}_k^{\mathbf{x}}$

0. Initialization

$$\widetilde{\mathbf{x}}_1 = \mathbf{A}\widehat{\mathbf{x}}_0 + \mathbf{B}\widehat{\mathbf{u}}_0$$

$$\widetilde{\mathbf{P}}_1^{\mathbf{x}} = \mathbf{A}\mathbf{P}_0^{\mathbf{x}}\mathbf{A}^{\mathsf{T}} + \mathbf{B}\mathbf{P}_0^{\mathbf{u}}\mathbf{B}^{\mathsf{T}} + \mathbf{Q}_0^{\mathbf{x}}$$

for *each time step* $k > 0$ **do**

1. Input estimation

$$\mathbf{i}_k = \mathbf{y}_k - \mathbf{C}\widetilde{\mathbf{x}}_k$$

$$\left(\widehat{\mathbf{u}}_k, \mathbf{K}_k^{\mathbf{u}}, \widetilde{\mathbf{P}}_k^{\mathbf{u}}, \widehat{q}_k\right) = \text{OptimalEstimation}(\mathbf{D}, \mathbf{i}_k, \mathbf{R}, \widehat{\mathbf{u}}_{k-1}, \widehat{q}_{k-1})$$

$$\mathbf{P}_k^{\mathbf{u}} = \left(\mathbf{I} - \mathbf{K}_k^{\mathbf{u}}\mathbf{D}\right)\widetilde{\mathbf{P}}_k^{\mathbf{u}}\left(\mathbf{I} - \mathbf{K}_k^{\mathbf{u}}\mathbf{D}\right)^{\mathsf{T}} + \mathbf{K}_k^{\mathbf{u}}\left(\mathbf{C}\widetilde{\mathbf{P}}_k^{\mathbf{x}}\mathbf{C}^{\mathsf{T}} + \mathbf{R}_k\right)\mathbf{K}_k^{\mathbf{u}\mathsf{T}}$$

2. State estimation

$$\mathbf{K}_k^{\mathbf{x}} = \widetilde{\mathbf{P}}_k^{\mathbf{x}}\mathbf{C}^{\mathsf{T}}\left(\mathbf{C}\widetilde{\mathbf{P}}_k^{\mathbf{x}}\mathbf{C}^{\mathsf{T}} + \mathbf{R}_k\right)^{-1}$$

$$\widehat{\mathbf{x}}_k = \widetilde{\mathbf{x}}_k + \mathbf{K}_k^{\mathbf{x}}\left(\mathbf{i}_k - \mathbf{D}\widehat{\mathbf{u}}_k\right)$$

$$\mathbf{P}_k^{\mathbf{x}} = \left(\mathbf{I} - \mathbf{K}_k^{\mathbf{x}}\mathbf{C}\right)\widetilde{\mathbf{P}}_k^{\mathbf{x}}\left(\mathbf{I} - \mathbf{K}_k^{\mathbf{x}}\mathbf{C}\right)^{\mathsf{T}} + \mathbf{K}_k^{\mathbf{x}}\left(\mathbf{D}\mathbf{P}_k^{\mathbf{u}}\mathbf{D}^{\mathsf{T}} + \mathbf{R}_k\right)\mathbf{K}_k^{\mathbf{x}\mathsf{T}}$$

$$\mathbf{P}_k^{\mathbf{xu}} = -\mathbf{K}_k^{\mathbf{x}}\mathbf{D}\mathbf{P}_k^{\mathbf{u}}$$

3. State prediction

$$\widetilde{\mathbf{x}}_{k+1} = \mathbf{A}\widehat{\mathbf{x}}_k + \mathbf{B}\widehat{\mathbf{u}}_k$$

$$\widetilde{\mathbf{P}}_{k+1}^{\mathbf{x}} = \begin{bmatrix}\mathbf{A} & \mathbf{B}\end{bmatrix}\begin{bmatrix}\mathbf{P}_k^{\mathbf{x}} & \mathbf{P}_k^{\mathbf{xu}} \\ \mathbf{P}_k^{\mathbf{ux}} & \mathbf{P}_k^{\mathbf{u}}\end{bmatrix}\begin{bmatrix}\mathbf{A}^{\mathsf{T}} \\ \mathbf{B}^{\mathsf{T}}\end{bmatrix} + \mathbf{Q}_k^{\mathbf{x}}$$

end

Algorithm 2: OptimalEstimation

Input: $\mathbf{D}$, $\mathbf{i}_k$, $\mathbf{R}_k$, $\widehat{\mathbf{u}}_{k-1}$, $\widehat{q}_{k-1}$

Output: $\widehat{\mathbf{u}}_k$, $\mathbf{K}_k^{\mathbf{u}}$, $\widetilde{\mathbf{P}}_k^{\mathbf{u}}$, $\widehat{q}_k$

0. Initialization

$\widehat{q}_k^{(0)} = \widehat{q}_{k-1}$, $\widehat{\mathbf{u}}_k^{(0)} = \widehat{\mathbf{u}}_{k-1}$

while $\delta < tol$ **do**

$i = i + 1$

1. Compute $\widehat{\tau}_k^{(i)}$ from Eq. (54) given $\widehat{\mathbf{u}}_k^{(i-1)}$ and $\widehat{q}_k^{(i-1)}$.
2. Compute $\widehat{q}_k^{(i)}$ from the brute force approach applied to Eq. (53) given $\widehat{\mathbf{u}}_k^{(i-1)}$ and $\widehat{\tau}_k^{(i)}$.
3. Compute $\widehat{\mathbf{u}}_k^{(i)}$ and $\mathbf{W}_k^{(i)}$ from Eqs. (55)-(60) given $\widehat{q}_k^{(i)}$ and $\widehat{\tau}_k^{(i)}$.
4. Convergence monitoring - Computation of δ:
$$\delta = \frac{\left\| \widehat{\mathbf{u}}_k^{(i)} - \widehat{\mathbf{u}}_k^{(i-1)} \right\|_2^2}{\left\| \widehat{\mathbf{u}}_k^{(i-1)} \right\|_2^2}$$

end

$\widehat{q}_k = \widehat{q}_k^{(I)}$, $\widehat{\tau}_k = \widehat{\tau}_k^{(I)}$ and $\mathbf{W}_k = \mathbf{W}_k^{(I)}$

$\widetilde{\mathbf{P}}_k^{\mathbf{u}} = (\widehat{\tau}_k \mathbf{W}_k)^{-1}$

6. Numerical experiment

This section introduces the application of the SaBF in a purely numerical context. Its robustness with respect to various practical parameters, such as the measurement noise, the sensors' density or the duration of the estimation (short-term or long-term) is assessed and its performances are compared to those of some state-of-the-art filters. Only the estimation of the input vector is targeted here, even though the estimation of the state vector can be of primary interest in some applications. Hence, the problem is oriented to the solution of the force identification problem rather than to the state reconstruction through the measures.

6.1. Problem definition

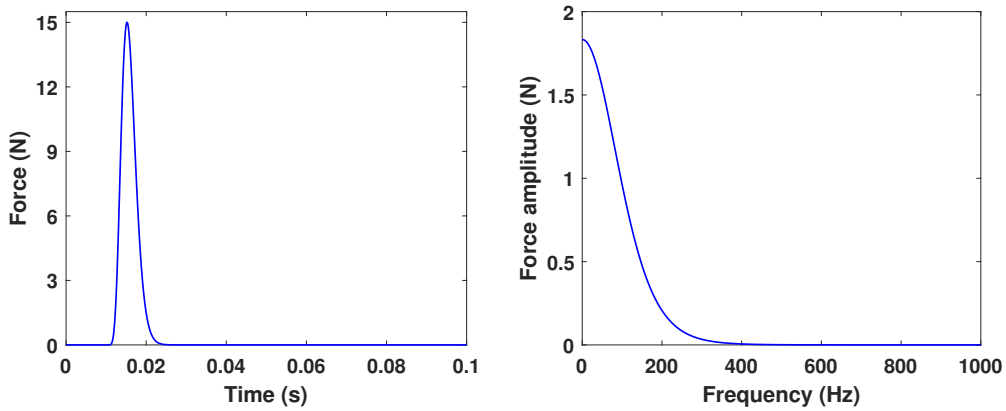
The structure under consideration is a simply supported stainless steel beam. The beam has a length of $L = 3\text{m}$, with a cross-sectional area $S = 1060\text{mm}^2$ and quadratic moment of inertia $I = 171\text{mm}^4$. The material properties are $E = 210\text{GPa}$ for the Young's modulus and $\rho = 7850\text{kg/m}^3$ for the density. The modal damping factor is set to 0.01.

In the present numerical application, it is supposed that the beam undergoes a hammer impact at location $x_{\text{exc}} = 0.98\text{m}$, measured from its left end. This type of excitation can be modeled by a Gamma-like function of shape and scale parameters p and θ [26], that is:

$$u_{\text{ref}}(t) = u_0 \left(\frac{t}{p\theta}\right)^p \exp\left(-\frac{t}{\theta} + p\right), \tag{62}$$

where u_0 is the force intensity. In this example, it is assumed that the structure is impacted by a hammer equipped with a soft rubber tip in order to

excite only the low frequency modes of the beam. To reflect this assumption, the parameters of the input excitation are chosen such that $u_0 = 15\text{N}$, $p = 8.7$ and $\theta = 0.6\text{ms}$. Furthermore, a delay of 8ms is added at the beginning of the signal to simulate a pre-trigger. This reference force is shown in Fig. 1. It must be noted here that the applied hammer excitation has a cut-off frequency around 500 Hz.



(a) (b)

Figure 1: Synthesized hammer impact excitation signal - (a) Time domain representation and (b) Frequency domain representation.

Along the beam, a set of 20 sensors are mounted on the structure to measure kinematic variables, such as strain, displacement, velocity or acceleration. Here, it is supposed that only acceleration data are available, since they lead to better input reconstructions when Kalman-like filters are to be used [26, 29, 31, 35]. To synthesize these measured data, a modally reduced order model of the beam is used. The considered modal basis contains the first 53 analytical vibration modes, corresponding to modes having

a resonance frequency below 1kHz. The noiseless acceleration data are computed over 0.1s from an unconditionally stable and second-order accurate Newmark's integration scheme using a time step size of 10µs. Then, for simulating the measurement process, the computed data are corrupted by an additive Gaussian white noise with a controlled signal-to-noise ratio (SNR) set to 25 dB.

To properly implement all the filters compared in this paper, it remains to define the initial conditions $(\widehat{\mathbf{x}}_0, \mathbf{P}_0^{\mathbf{x}})$ and $(\widehat{\mathbf{u}}_0, \mathbf{P}_0^{\mathbf{u}})$ as well as the covariance matrices $\mathbf{Q}_k^{\mathbf{x}}$ and $\mathbf{R}_k$. Here, the initial covariance matrices $\mathbf{P}_0^{\mathbf{x}}$ and $\mathbf{P}_0^{\mathbf{u}}$ are assumed isotropic with a variance set to 10^{-20}, while the initial state and input vectors $\widehat{\mathbf{x}}_0$ and $\widehat{\mathbf{u}}_0$ are null vectors. The noise covariances $\mathbf{Q}_k^{\mathbf{x}}$ and $\mathbf{R}_k$ are supposed isotropic and constant over the time, with respective values of 10^{-20} and 10^{-2}. For the SaBF, the initial shape parameter has also to be specified. Here, $\widehat{q}_0$ is set to 1, because the spatial distribution of the input vector should be sparse. For AKF and DKF, the covariance matrix associated to the fictitious equation on the input vector is supposed isotropic and its variance is adjusted following the procedure described in Refs. [29, 31].

In absence of contradictory information, all the values given above form the default configuration. In addition, it should be noting here that colocated configurations are required for instantaneous inversion, i.e. the estimation of the system input without a time delay [48]. This practically implies that the input vector is estimated at sensors locations.

Finally, to quantify the accuracy of the estimated input vector, three indicators are introduced. First, the Global Relative Error (GRE) is used

to evaluate the accuracy of the estimation over the whole structure. Its expression is given in the following equation:

$$\mathrm{GRE} = \frac{\|\widehat{\mathbf{u}} - \mathbf{u}_{\mathrm{ref}}\|_1}{\|\mathbf{u}_{\mathrm{ref}}\|_1}. \tag{63}$$

where $\mathbf{u}_{\mathrm{ref}}$ and $\widehat{\mathbf{u}}$ are respectively the reference and estimated input vectors defined over the whole estimation duration.

The two other indicators are local. The Peak Error (PE) is representing the relative error between the maximal intensity of the hammer impact estimated at the actual point force location $\widehat{u}^{\mathrm{exc}}$ and that of the reference one $u_{\mathrm{ref}}^{\mathrm{exc}}$. This indicator is mathematically expressed as:

$$\mathrm{PE} = \frac{\widehat{u}^{\mathrm{exc}} - u_{\mathrm{ref}}^{\mathrm{exc}}}{u_{\mathrm{ref}}^{\mathrm{exc}}}. \tag{64}$$

Its sign allows knowing whether the estimated value is higher (or lower) than the targeted value.

The last indicator is the Correlation Coefficient (CC), which defines the overall accuracy of the estimated excitation history, $\widehat{\mathbf{u}}^{\mathrm{exc}}$, with respect to the reference one, $\mathbf{u}_{\mathrm{ref}}^{\mathrm{exc}}$, at the actual impact point location:

$$\mathrm{CC} = \frac{\mathbf{u}_{\mathrm{ref}}^{\mathrm{exc}}\ \widehat{\mathbf{u}}^{\mathrm{exc}^{\mathsf{T}}}}{\|\mathbf{u}_{\mathrm{ref}}^{\mathrm{exc}}\|_2\ \|\widehat{\mathbf{u}}^{\mathrm{exc}}\|_2}. \tag{65}$$

6.2. Identification of a hammer impact

As a first application, SaBF is compared to AKF, DKF and GDF on the default configuration described in the previous section. It should be mentioned here, that SBF is not considered here and in the rest of the paper, because it diverges for the parameters used in the paper to tune the filter (see Appendix B). One of the main causes of divergence is related to the trick

introduced by Sedehi et al. in Ref. [35] and recalled in Eq. (31) to mitigate the drift effect. Other parameters may explain this result, but a thorough analysis of the tuning of SBF is outside the scope of this paper.

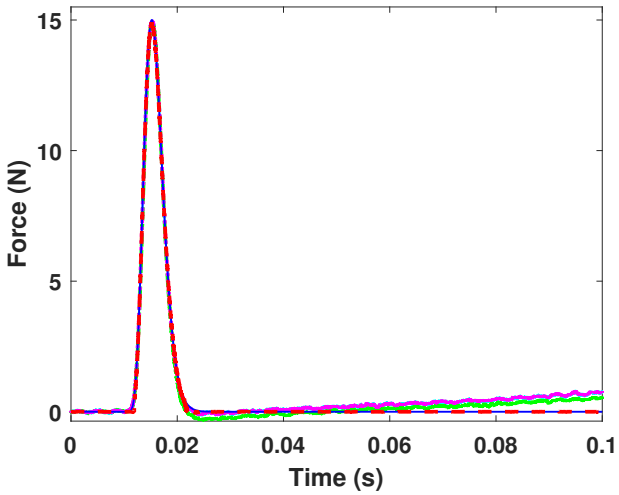


Figure 2: Time history of the estimated input obtained from the default configuration - (—) Reference, (– –) SaBF, (· · ·) AKF, (– · –) DKF and (– –) GDF.

The estimation of the input time history presented in Fig. 2 shows that the peak value estimated by all filters is close to the reference value of 15N. The peak error of each filter is below 1%, as exposed in the Table 1. Moreover, another important aspect of the reconstruction is the behavior after having identified the hammer impact: the residual value that exists leads the filter to diverge instead of remaining constant. This is the so-called drift effect, which characterizes the way the identified variable tends to deviate from a constant value, null here. That drift is clearly observable on all filters from $t = 50$ms, except on SaBF where the estimation stagnates, with a standard deviation of 0.18mN.

Table 1 gathers the values of the three indicators, GRE, PE and CC, for each filter compared in this study. It shows that the global error is significantly reduced while using the SaBF compared to the state-of-the-art filters, whereas the peak error is acceptable. Moreover, the correlation coefficient between the reference force and the estimate input history is the highest for the SaBF.

Table 1: Table of indicators for each filter tested for the default configuration

Filter	**GRE** (%)	**PE** (%)	**CC** (%)
GDF	75.49	-0.2	99.2
DKF	59.32	-0.4	99.5
AKF	74.88	-0.2	99.2
SaBF	3.95	-0.6	99.9

Finally, as shown in Fig. 2 and Table 1, SaBF outperforms the existing Kalman-like filters in the identification of the input time history. Nevertheless, all filters identify well the impact location. The Fig. 3 afterwards shows the estimated location and time history of the impact, according to each filter.

For this experimental configuration, the shape parameter q is computed by following the iterative procedure described in Algorithm 2. Although this parameter is mathematically in $\mathbb{R}^{+*}$, it is restricted to the range $]l_b; u_b]$ with $l_b = 0$ and $u_b = 2$ in the implementation as classically done in the literature. This parameter is optimised at each time-step with the minimization of the cost function of the Eq. (53). As shown in Fig. 4, the initial value q_0 doesn't affect the final result. Finally, considering the initial value of the shape

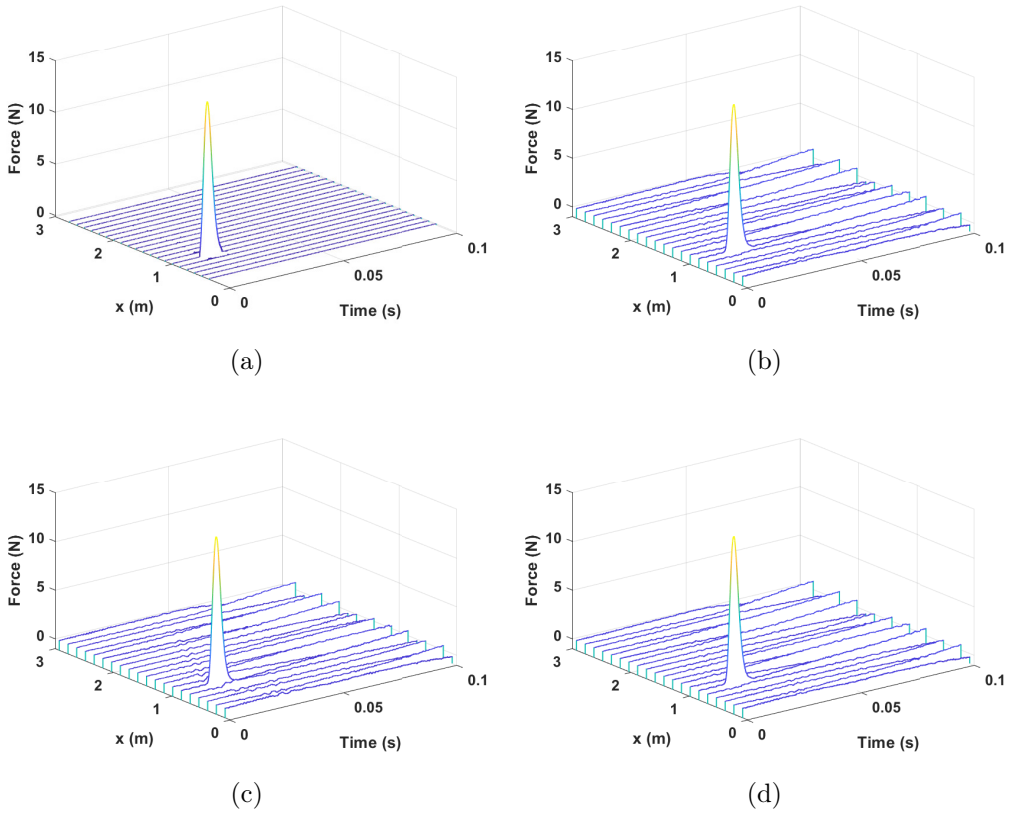


Figure 3: Estimated input distribution from the default configuration – (a) SaBF, (b) AKF, (c) DKF and (d) GDF.

parameter at each iteration (before the optimisation), two choices are possible : $q_k = q_0$, set to 1 for example, and $q_k = q_{k-1}$. The second one has been chosen, as it allows to obtain a better calculation speed. Actually, as the source is fixed and remains sparse along the time, q_k is quite close to q_{k-1}.

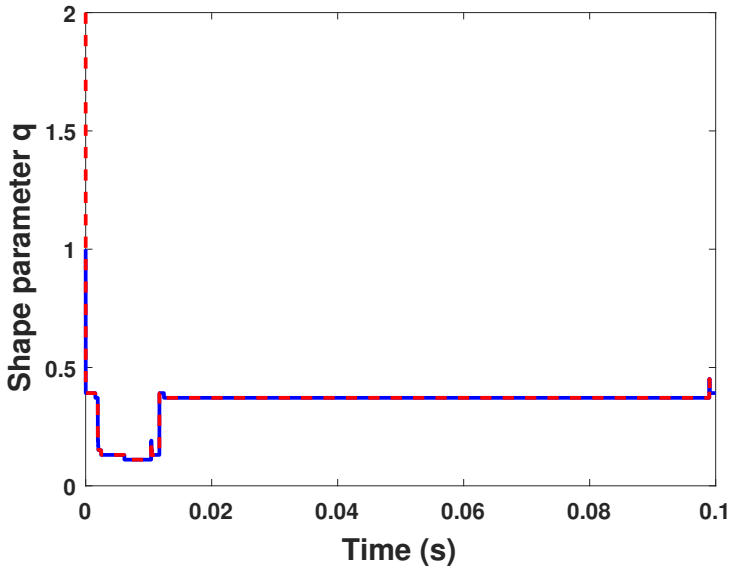


Figure 4: Time history of q_{opt} with (—) $q_0 = 1$ and (−−) $q_0 = 2$.

6.3. Influence of the experimental parameters

In the engineering practice, some parameters can have an important influence on the quality of the estimated solutions. Among those, the measurement noise level and the number of sensors are of particular interest. Consequently, this section aims at evaluating the influence of these parameters on the performances of all the filters considered in this paper.

6.3.1. Influence of the measurement noise

The noise added to acceleration data is increased, from a signal-to-noise ratio of 25dB to 15dB. This rise is reflected in the results, especially on the time history. As shown in Fig. 5, all the state-of-the-art filters are drifted up after the impact, whereas the time history after impact of the SaBF is still constant with a null value and a standard deviation around 0.15N. So, the higher the noise, the steeper the slope of this drift is for AKF, DKF and GDF.

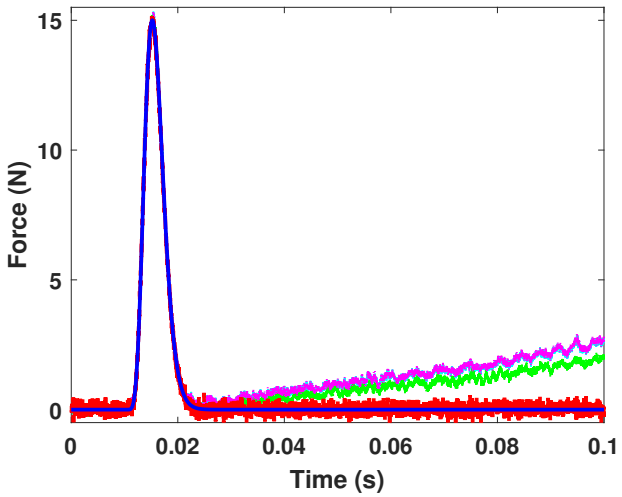


Figure 5: Time history of the estimated input for a SNR of 15dB - (—) Reference, (– –) SaBF, (· · ·) AKF, (– · –) DKF and (– –) GDF.

When looking at the values of the performance indicators given in Table 2, one can notice that the proposed SaBF produces significantly better results in terms of global identification even though DKF provides a slightly better identification of the peak value. Furthermore, even though the impact is correctly located, a drift appears on almost all the identification points,

Table 2: Table of indicators for each filter tested for a SNR of 15dB.

Filter	**GRE** (%)	**PE** (%)	**CC** (%)
GDF	242.7	2.2	90.3
DKF	190.2	1.27	94.1
AKF	240.8	2.2	90.1
SaBF	22.84	1.43	99.8

except for SaBF. As the data are noisy, Kalman-like filters tends to predict and correct the estimation with regards to non-null data. Hence, the predicted values drift progressively, the sign of the estimation only depends on the initial mean of the noise, which cannot be controlled. The high value of the correlation coefficient for the SaBF is mainly due to the removal of the drift (see Fig. 6).

6.3.2. Influence of the number of sensors

The number of sensors controls the total amount of data that the filter uses to perform the input-state estimation. Hence, the process may last longer to complete the global computation. By doubling this number, from 20 to 40 sensors, it appears that the state-of-the-art filters behave in a worse way. Actually, as the problem is under-determined, a greater amount of information does not lead to a more accurate estimation. As shown in Fig. 7 and in the same way as it happens when the measurement noise increases, all the state-of-the-art filters are drifted up, whereas SaBF remains constant with a null value and a standard deviation around 0.15mN.

Thus, both the global relative error and the peak error for SaBF are the

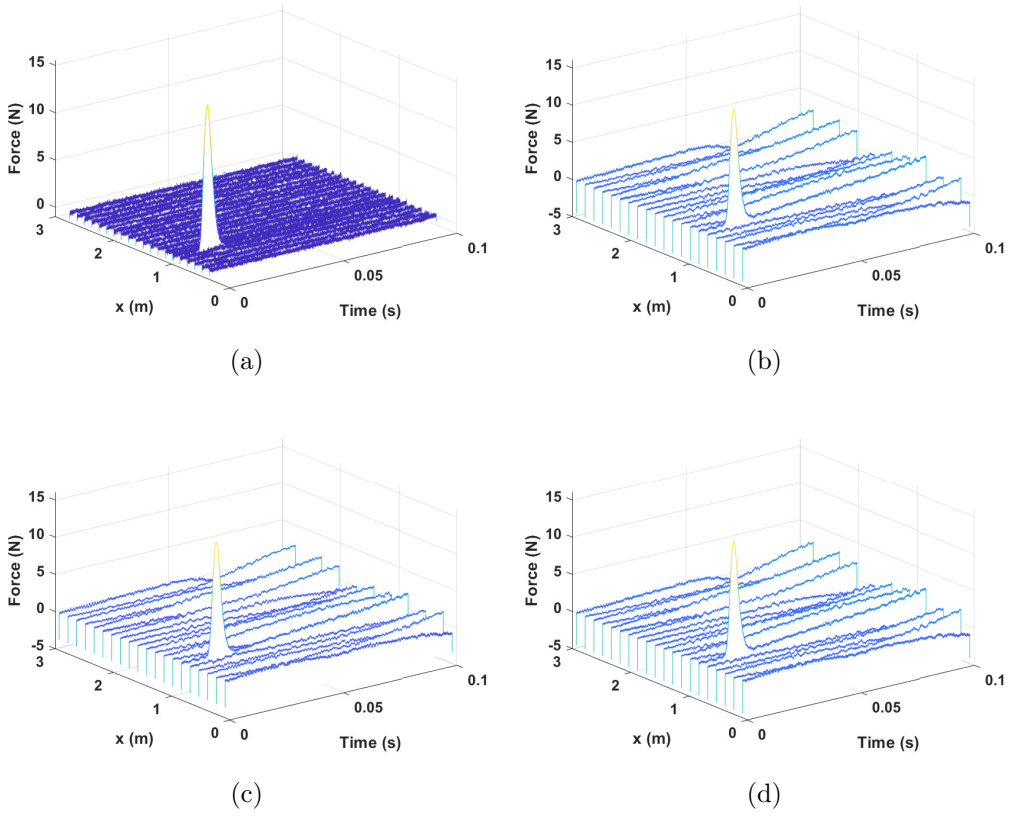


Figure 6: Spatial distribution of the input estimated for a SNR of 15dB – (a) SaBF, (b) AKF, (c) DKF and (d) GDF.

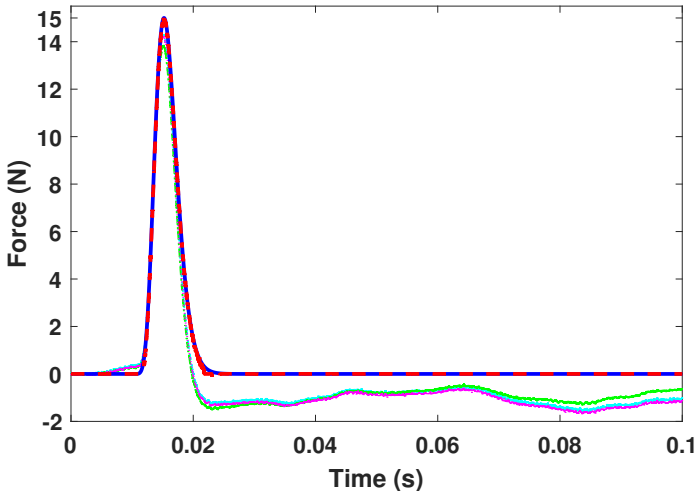


Figure 7: Time history of the estimated input obtained for a set 40 sensors - (—) Reference, (– –) SaBF, (· · ·) AKF, (– · –) DKF and (– –) GDF.

lowest of all introduced filters, as shown in Table 3, while the correlation coefficient is still very close to the maximal value (99.98 %). As for the

Table 3: Table of indicators for each filter tested with 40 sensors

Filter	**GRE** (%)	**PE** (%)	**CC** (%)
GDF	550.0	-4.33	92.7
DKF	483.9	-7.53	92.7
AKF	552.6	-4.8	91.4
SaBF	4.20	0.07	99.9

measurement noise, the number of sensors has a great influence on the presence of drifting on the identification points: doubling the number of sensors leads to a higher mean drift and an offset on the final results for all of the state-of-the-art filters. These two phenomena lead to a significant error after

0.1s: the force estimated by the filters reaches almost 50% of the peak value (around 7N) on points that are not excited at all. These erroneous values can be observed on Fig. 8.

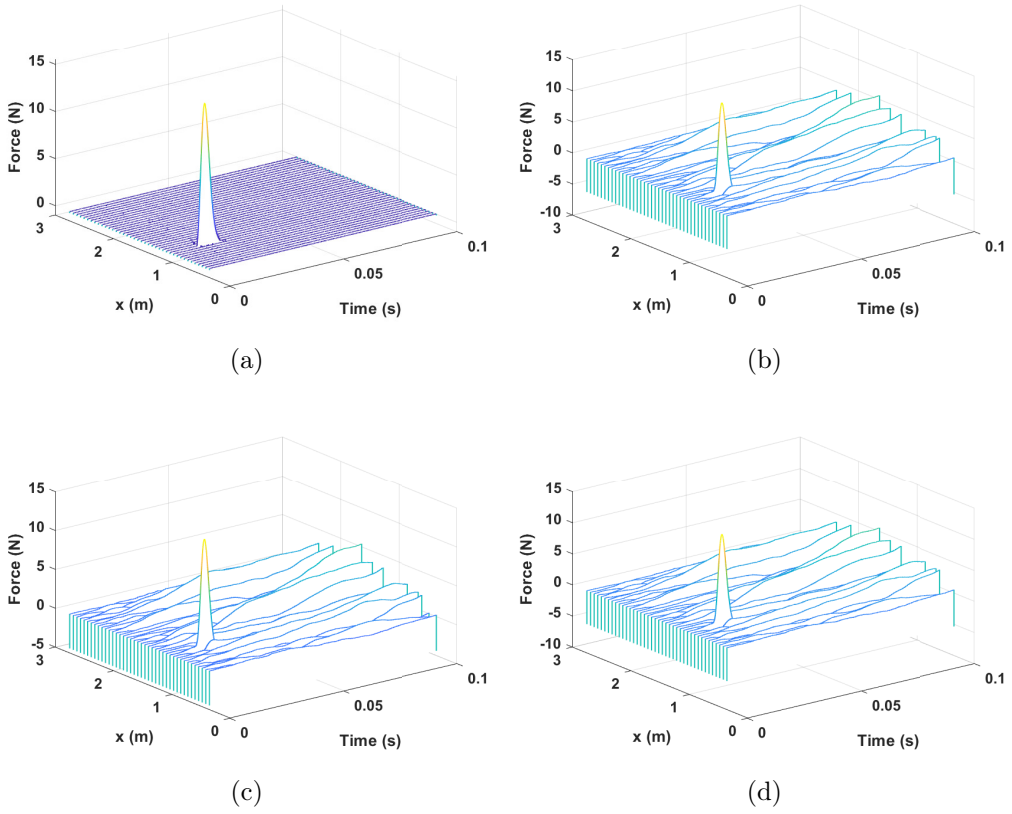


Figure 8: Spatial distribution of the input estimated at 40 locations – (a) SaBF, (b) AKF, (c) DKF and (d) GDF.

To complete this part on the influence of the number of sensors, the same study is done with a reduced set of 9 sensors. In this situation, the filters are expected to provide similar performances. Figs. 9 and 10 and Table 4 confirm this intuition, since the results are rather similar for all filters. To

sum it up, all the results presented in this section highlight the robustness of SaBF with respect to the number of sensors.

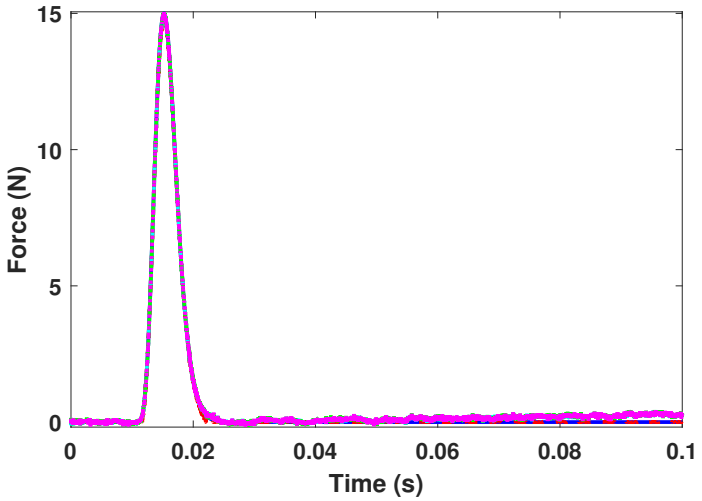


Figure 9: Time history of the estimated input obtained for a set 9 sensors - (—) Reference, (– –) SaBF, (· · ·) AKF, (– · –) DKF and (– –) GDF.

Table 4: Table of indicators for each filter tested for a simulation with 9 sensors.

Filter	**GRE** (%)	**PE** (%)	**CC** (%)
GDF	12.45	0.33	99.92
DKF	12.41	0.4	99.92
AKF	12.55	0.33	99.92
SaBF	2.81	0.01	99.99

6.3.3. Influence of the total duration

As observed in the previous subsection, the total duration of the experiment combined with the drift induces a significant error on the input estima-

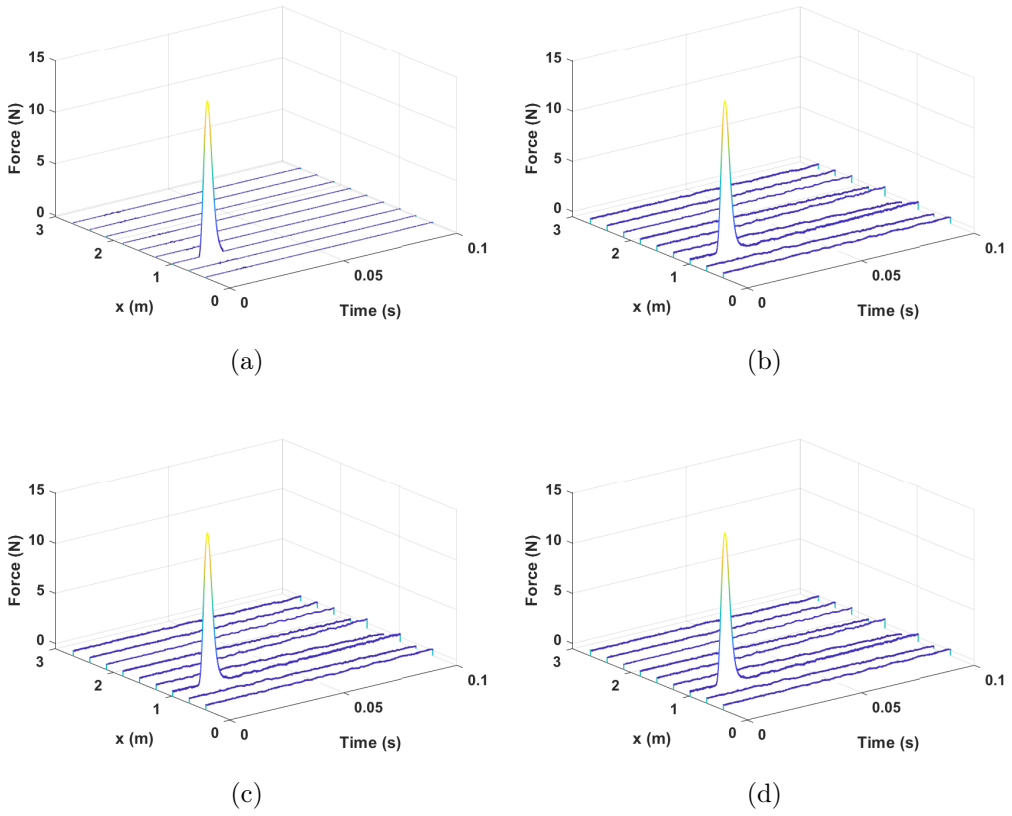


Figure 10: Spatial distribution of the input estimated at 9 locations – (a) SaBF, (b) AKF, (c) DKF and (d) GDF

tion. This error accumulates at each time step, following an unpredictable behavior at each identification point. Hence, as shown in Fig. 11, AKF, DKF and GDF provide almost the same wrong estimation of the input. On the other hand, the force identified by SaBF remains constant and null, with a standard deviation around 39.4µN.

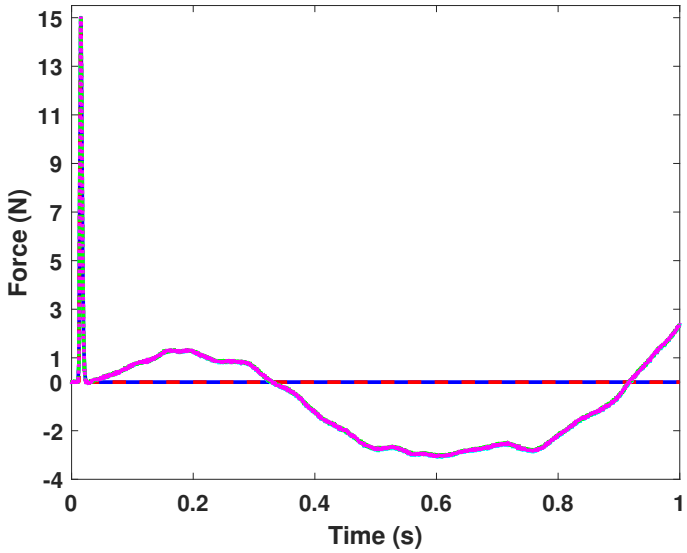


Figure 11: Time history of the estimated input computed over 1s - (—) Reference, (– –) SaBF, (· · ·) AKF, (– · –) DKF and (– –) GDF.

As expected, the effect of the drift is much more important over a long duration: when observing all the identification points, it seems that a distributed force is applied all along the beam for the state-of-the-art filters (see Fig. 11). The maximal value of the absolute error reached by these spurious forces is around 10.8N, which is around the nominal value of the impact. Hence, the identification of impacts with these filters cannot be done online: even though a shock would have been identified in terms of intensity and

location, the estimation is wrong the next second and could potentially lead to misunderstand the actual phenomenon that happens.

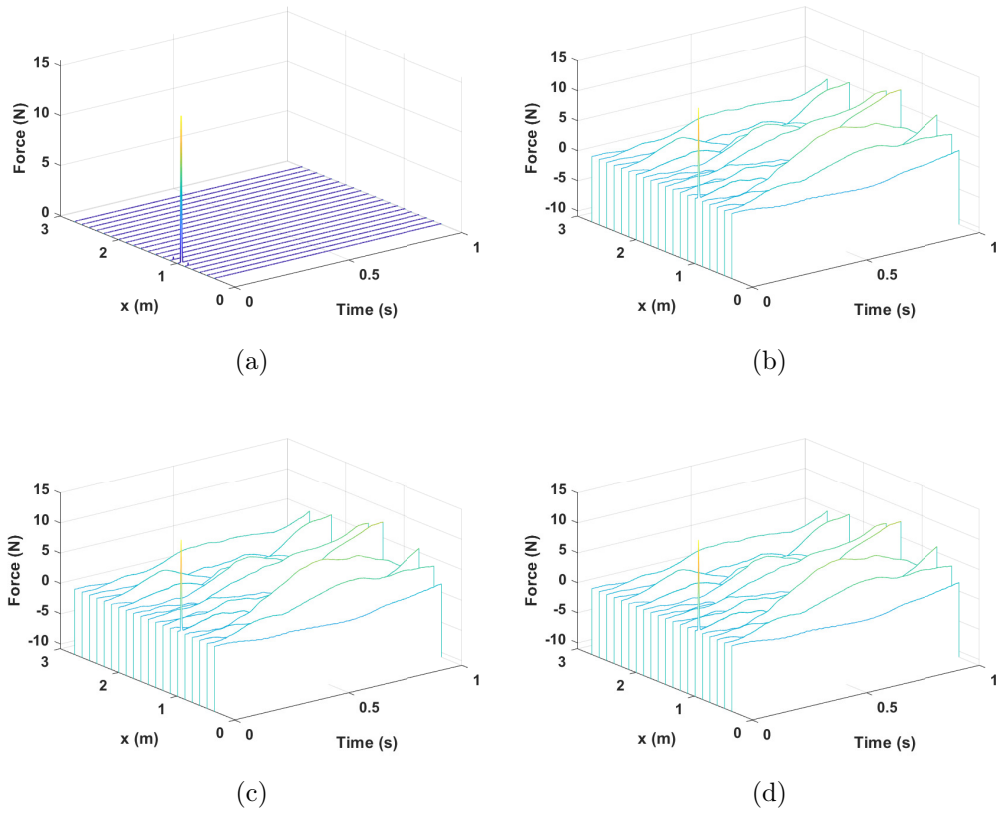


Figure 12: Spatial distribution of the input estimated over 1s – (a) SaBF, (b) AKF, (c) DKF and (d) GDF.

The performance indicators, collected in Table 5, demonstrate that the input estimation is totally biased by the presence of the drift for AKF, DKF and GDF. Thus, even though the peak amplitude is well estimated by all the filters, SaBF outperforms all the presented filters over long duration.

Table 5: Table of indicators for each filter tested for a 1 second simulation.

Filter	**GRE** (%)	**PE** (%)	**CC** (%)
GDF	520.1	0.2	41.4
DKF	517.8	0.27	41.8
AKF	518.6	0.28	41.7
SaBF	6.41	-1.5	99.9

7. Real-world application

This section aims at confirming the conclusions drawn in the numerical experiment on a real-world application. The basic motivation of this experimental validation is to assess the performances of the proposed Bayesian Filter in operating conditions. In what follows, two parameters are carefully studied. The first one is the influence of the measurement noise level, while the second one is related to the behavior of the filter over the long term. As for the numerical validation, SaBF is compared to some state-of-the-art filters.

7.1. Description of the experimental set-up

The structure under test is a thin aluminum plate of 60cm in length, 40cm in width and 6mm in thickness, clamped along its length in a wooden support. The effective width of the plate resulting from the mounting conditions is 39.1cm. To perform all the subsequent measurements, the system is suspended to a rigid structure through a set of elastic bungee cords (see Fig. 13).

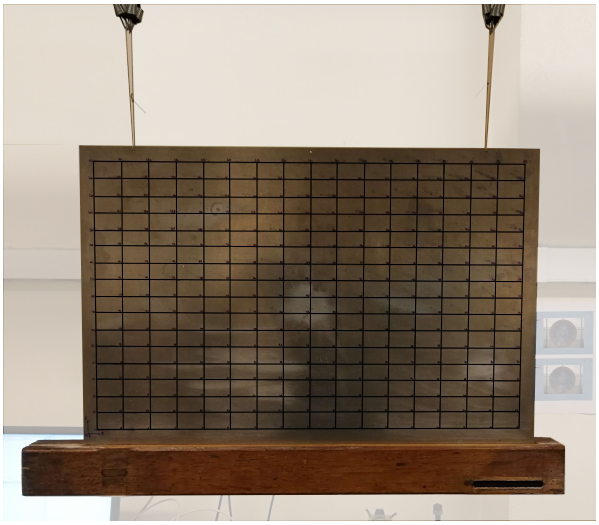


Figure 13: Structure under test - Clamped plate suspended to a rigid frame by bungee cords

7.1.1. State-space representation

As it has been done in the numerical experiment, the explicit generalized-α integration scheme is chosen to compute the modally reduced state-space matrices of the considered dynamical system [41]. Because one of the boundary conditions is not perfectly known, the modal basis of the structure has been obtained from an Experimental Modal Analysis (EMA). More precisely, a roving hammer test has been conducted on a grid of 17×17 points (see Fig. 13) using three reference accelerometers to measure a set of Frequency Response Functions. By applying a MDOF curve fitting algorithm, the first 83 vibration modes have been extracted. For the purpose of the proposed experimental validation only the first 34 vibration modes have been retained in the modal basis. This actually corresponds to the modes having a resonance frequency below 2kHz.

Regarding the parameters of the filters, the initial state and input conditions must be specified as well as the covariance matrices associated to the process and measurement noises. For this experimental applications, the values of these parameters are the same as those defined for the numerical application (see section 6.1).

7.1.2. Data acquisition

In the present experiment, a set of 4 accelerometers have been mounted on the structure to measure the vibrations resulting from a hammer impact. To satisfy the instantaneous inversion constraint, the input force and acceleration data have been located at some nodes of the grid used to perform the EMA. The locations and identification numbers (ID) of the hammer impact and the acceleration measurements are presented in Fig. 14.

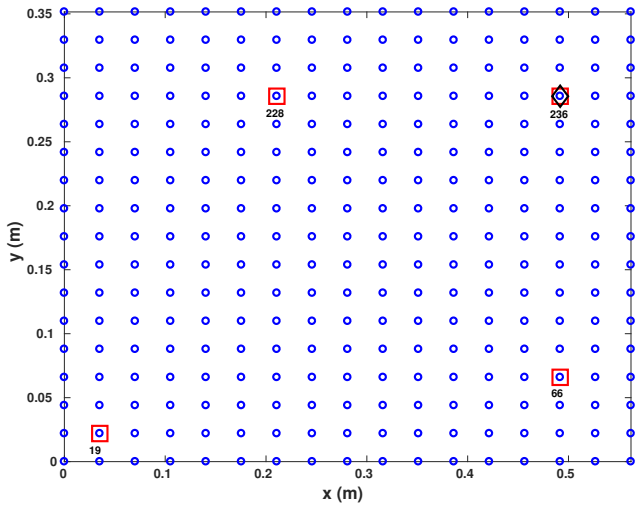
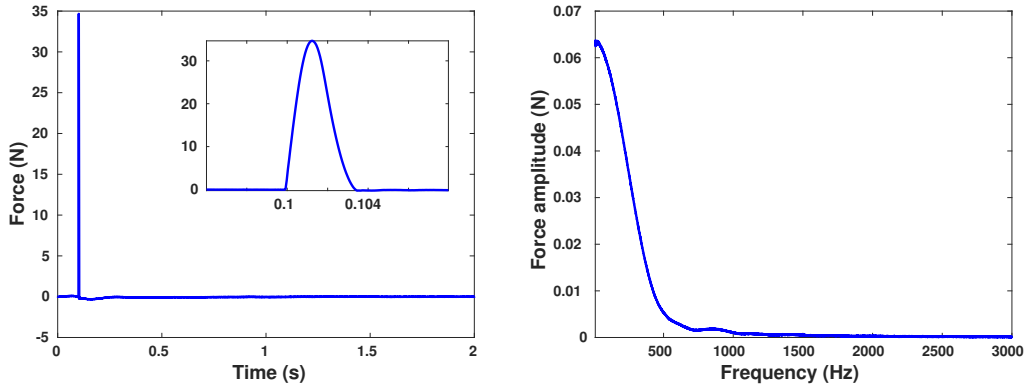


Figure 14: Location of the accelerometers and the hammer impact on the grid used to perform the EMA – – (○) EMA grid, (□) Accelerometers and (◊) Hammer impact

Practically, the impact hammer is equipped with a soft rubber tip in order to excite only the low frequency modes of the plate, i.e. below 1kHz. In doing so, the state-space model, built using all the modes having a resonance frequency below 2 kHz, is expected to be well-representative of the considered system. As shown in Fig. 15, the considered hammer impact excites the structure for frequencies below 1kHz, while the maximum intensity is 34.65N. However, to mitigate the influence of out-of-band modes during the system inversion, acceleration data are filtered at 2kHz and resampled at 16 384Hz following the recommendations of Lourens et al. [29].



(a) (b)

Figure 15: Hammer impact measured at node 236 – (a) Time history and (b) Frequency spectrum

7.2. Reconstruction of the hammer impact

An analysis, similar to that proposed in section 6, is done hereafter from the acceleration data measured on the impacted plate. The time history and the location of the impact are estimated for a set of 4 sensors over a time

window of approximately 0.1s. Compared to the numerical experiment that has been conducted, only SaBF and GDF provide a correct identification of the intensity of the peak, as it is shown in the following Fig. 16. AKF identifies a 20N peak, then it tends toward an offset of 4.55N. A similar behavior is adopted by DKF, but in a worse way: the identified peak is around 5N while the final value remains at -1.3N. Finally, a slight drift is observed on the GDF result, which should be increased with the noise level and a longer estimation duration.

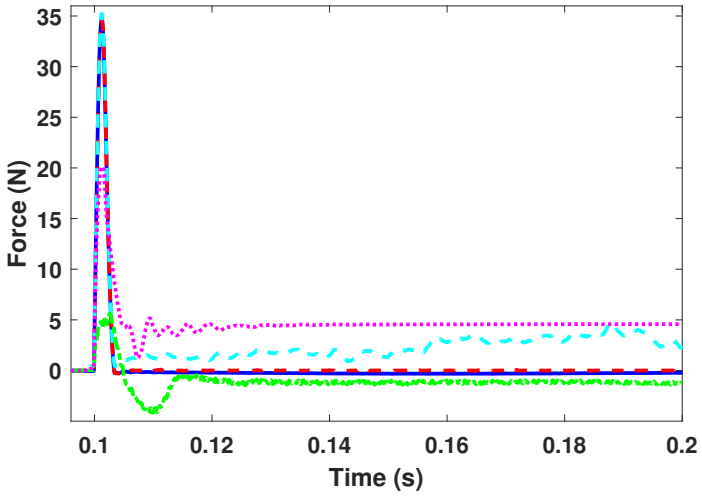


Figure 16: Time history of the estimated impact of the hammer locally on node 236 - (—) Reference, (– –) SaBF, (· · ·) AKF, (– · –) DKF and (– –) GDF.

The performance indicators allow quantifying the likelihood of the input estimation with the reference signal (see Table 6). As expected, those values are not acceptable for DKF and AKF. However, even though the GDF provides a great estimation of the peak, its global error is the worst, as the drift is very important at the other identification nodes (see Fig. 17).

Table 6: Table of indicators for each filter tested in the reference configuration.

Filter	**GRE** (%)	**PE** (%)	**CC** (%)
GDF	142.9	1.76	82.9
DKF	100.5	-83.78	46.9
AKF	87.8	-42.02	45.9
SaBF	1.1	0.35	99.8

The steep drift of the GDF is remarkable on the spatial distribution, even for a short experiment (here, 0.1s), whereas SaBF estimations on each node that are not subjected to a shock are completely flattened and nil. Here again, SaBF outperforms the other filters, whether in terms of peak detection, location or deviation after the impact during the rest of the experimentation.

7.2.1. Influence of the measurement noise level

In order to observe the effect of the noise level on the estimation, a Gaussian white noise is added to the measured data, so as to obtain a SNR around 15dB. As observed previously, the noise level leads to an increase of the slope of the drift. As GDF is the only filter that induces this phenomenon, it is only visible on its estimation. On the same duration, the absolute error, which reached 5N with a 25dB SNR, is currently at almost 20N. The peak value that was well estimated before, is now overtaking the targeted value. Meanwhile, the same problem of offset and low peak estimation is renewed for AKF and DKF.

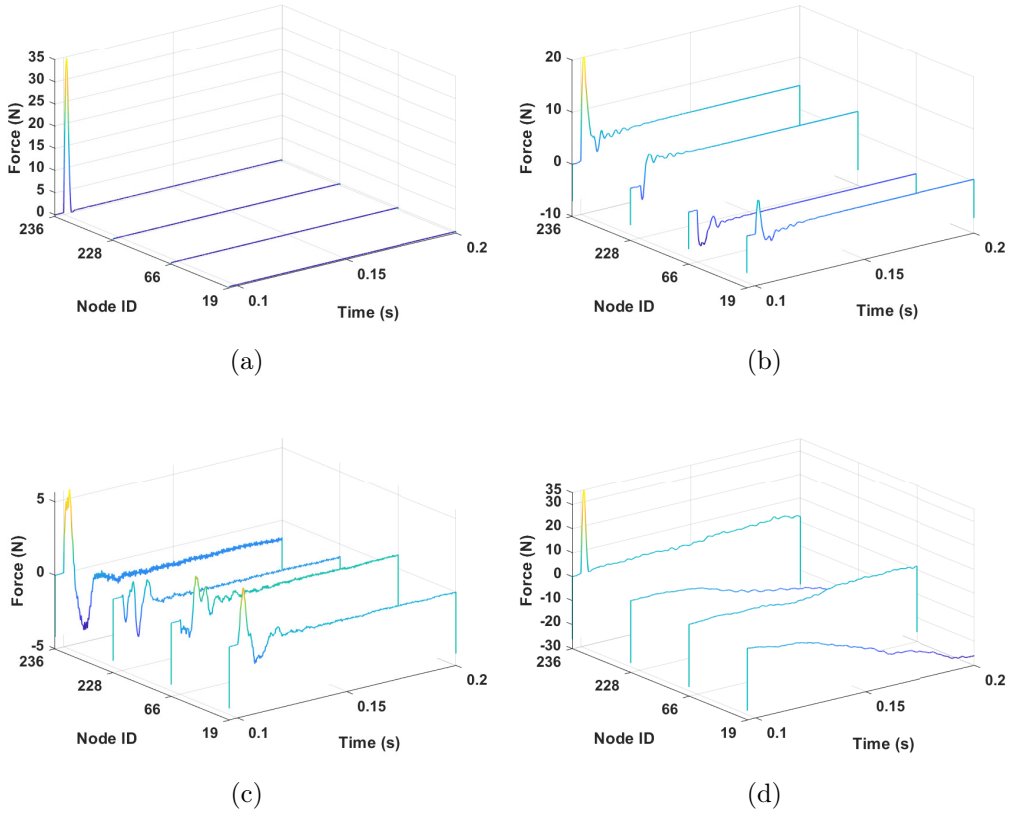


Figure 17: Spatial distribution of the estimated impact of the hammer – (a) SaBF, (b) AKF, (c) DKF and (d) GDF

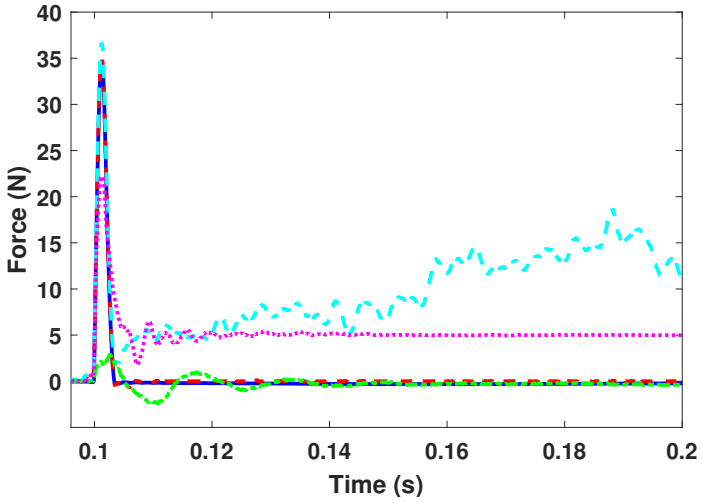


Figure 18: Time history of the estimated impact of the hammer locally on node 236 for an estimated SNR of 15dB - (—) Reference, (– –) SaBF, (· · ·) AKF, (– · –) DKF and (– –) GDF.

As expected in the default experimental configuration, even though the impact is well located, the higher the noise level, the higher the drift (see Fig. 18). It soars to reach a higher absolute value than the nominal force applied. From Fig. 19, AKF seems to be the more efficient identifying solution after SaBF. This observation is reinforced by the global indicator GRE, which is the second lowest of the Table 7 after the one of SaBF.

7.2.2. Influence of the estimation duration

Finally, the total duration of the experiment is investigated. Instead of a 0.1s time window, it has been decided to increase it to 0.9s, in order to observe the behavior of the each filter. As already highlighted, SaBF, AKF and DKF provide a constant force estimation after 0.05s, whereas GDF follows an unpredictable path. Hence, its values diverge noticeably at the excitation

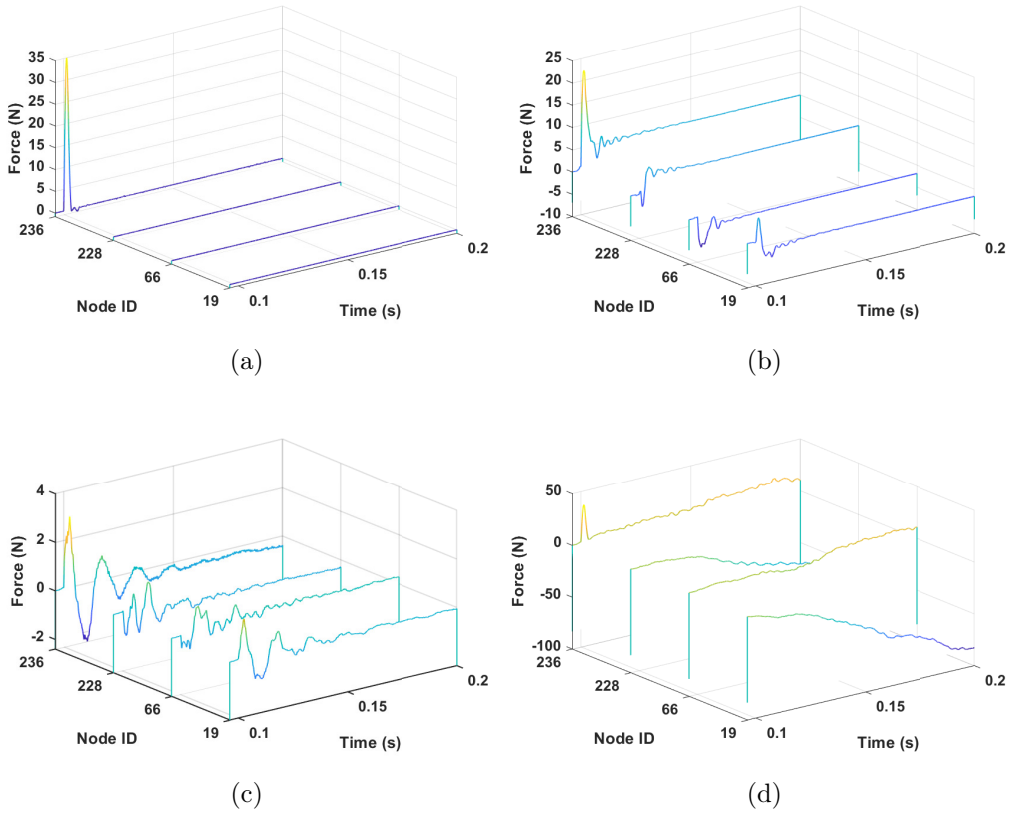


Figure 19: Spatial distribution of the estimated impact of the hammer for an estimated SNR of 15 dB – (a) SaBF, (b) AKF, (c) DKF and (d) GDF.

Table 7: Table of indicators for each filter tested for SNR = 15dB.

Filter	**GRE** (%)	**PE** (%)	**CC** (%)
GDF	466.3	5.92	33.1
DKF	100.5	-91.6	45.6
AKF	77.69	-36.05	45.0
SaBF	1.93	1.36	99.8

point, as shown the Fig. 20, as well as at the identification points of Fig. 21. This trend is reflected by the performance indicators presented in Table 8. Finally, it worth mentioning here that the performances SaBF are almost unaffected by a change in the measurement noise level or the estimation duration.

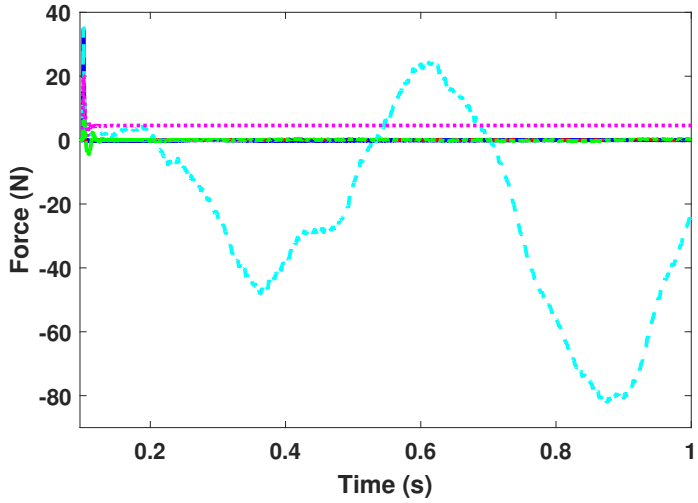


Figure 20: Time history of the estimated impact of the hammer locally on node 236 for a 1 second simulation - (—) Reference, (– –) SaBF, (· · ·) AKF, (– · –) DKF and (– –) GDF.

8. Conclusion

In this paper, a novel sequential Bayesian Filter has been developed for solving the input-state estimation problem. The initial motivation of the paper was to introduce some prior knowledge of the spatial distribution of the sources exciting a structure. To this end, a general Bayesian formulation of the problem has been presented. It is interesting to note that the proposed

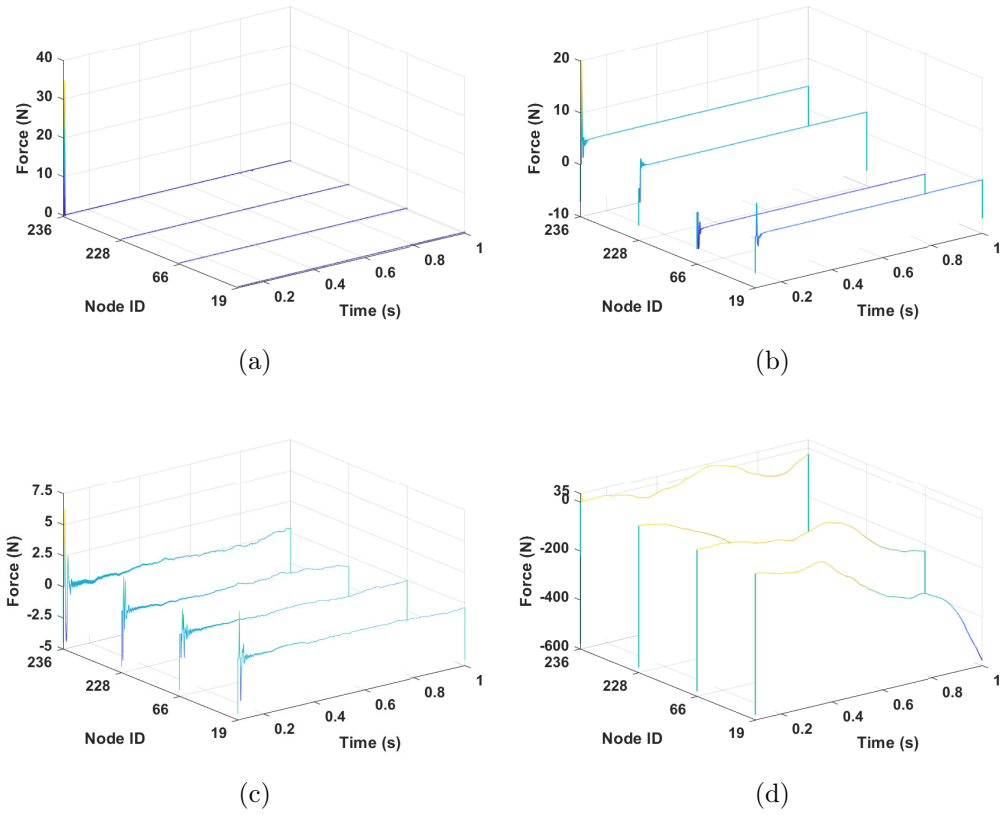


Figure 21: Spatial distribution of the estimated impact of the hammer for a 1 second simulation – (a) SaBF, (b) AKF, (c) DKF and (d) GDF.

Table 8: Table of indicators for each filter tested for a 1 second simulation.

Filter	**GRE** (%)	**PE** (%)	**CC** (%)
GDF	4055.6	1.76	6.07
DKF	100.1	-81.96	14.35
AKF	87.78	-42.02	12.67
SaBF	1.10	0.35	99.6

formulation is at the roots of most of the state-of-the-art filters developed in the literature, such as AKF, GDF, DKF or SBF. By clearly describing and detailing the hypotheses made by each filter, it has been demonstrated that the existing filters mainly differ by the assumption made in the definition of predictive probability distribution over the input vector. From this observation, it has been concluded that one's prior knowledge of the input vector can be properly reflected by an adequate choice of this probability distribution. Here, it has been assumed that the predictive input vector followed a multivariate generalized Gaussian distribution, given rise to the Sparse adaptive Bayesian Filter (SaBF) discussed in this paper. The proposed approach contains several parameters that are optimally estimated from a nested Bayesian optimization procedure.

A numerical experiment has been conducted to compare the performances of AKF, GDF, DKF, SBF and SaBF with respect to several key parameters, such as the measurement noise, the sensors' density or the estimation duration. The obtained results show that SaBF outperforms standard filters, especially because it is a purely online approach that allows inferring the spatial sparsity pattern of the input vector from the data and the model. This sparsity adaption property has a nice side effect, since it avoids the drift that appears when only acceleration measurements are used. This conclusion has been confirmed experimentally.

Finally, this paper demonstrates through the proposed general Bayesian framework that new Bayesian filters can be obtained by choosing different set of hypotheses. This is a topic of ongoing research.

A. Computational algorithms of input-state estimation problems

This appendix summarizes the computational algorithms of the state-of-the-art methods introduced in sections 3 and 4.

A.1. Augmented Kalman Filter

Algorithm 3: Augmented Kalman Filter – AKF

Input: $\mathbf{y}_k$, $\mathbf{A_a}$, $\mathbf{C_a}$, $\widehat{\mathbf{x}}_0^{\mathbf{a}}$, $\mathbf{P}_0^{\mathbf{x}}$, $\mathbf{Q}_k^{\mathbf{a}}$, $\mathbf{R}_k$

Output: $\widehat{\mathbf{x}}_k^{\mathbf{a}}$, $\mathbf{P}_k^{\mathbf{x}}$

for *each time step* $k > 0$ **do**

1. State prediction

$$\widetilde{\mathbf{x}}_k^{\mathbf{a}} = \mathbf{A_a}\widehat{\mathbf{x}}_{k-1}^{\mathbf{a}}$$

$$\widetilde{\mathbf{P}}_k^{\mathbf{x}} = \mathbf{A_a}\mathbf{P}_{k-1}^{\mathbf{x}}\mathbf{A}_{\mathbf{a}}^{\mathsf{T}} + \mathbf{Q}_{k-1}^{\mathbf{a}}$$

2. State estimation

$$\mathbf{K}_k^{\mathbf{x}} = \widetilde{\mathbf{P}}_k^{\mathbf{x}}\mathbf{C}_{\mathbf{a}}^{\mathsf{T}}(\mathbf{C_a}\widetilde{\mathbf{P}}_k^{\mathbf{x}}\mathbf{C}_{\mathbf{a}}^{\mathsf{T}} + \mathbf{R}_k)^{-1}$$

$$\widehat{\mathbf{x}}_k^{\mathbf{a}} = \widetilde{\mathbf{x}}_k^{\mathbf{a}} + \mathbf{K}_k^{\mathbf{x}}(\mathbf{y}_k - \mathbf{C_a}\,\widetilde{\mathbf{x}}_k^{\mathbf{a}})$$

$$\mathbf{P}_k^{\mathbf{x}} = (\mathbf{I} - \mathbf{K}_k^{\mathbf{x}}\mathbf{C_a})\widetilde{\mathbf{P}}_k^{\mathbf{x}}$$

end

Algorithm 4: Gillijns and De Moor Filter – GDF

Input: $\mathbf{y}_k$, $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$, $\widehat{\mathbf{x}}_0$, $\mathbf{P}_0^{\mathbf{x}}$, $\widehat{\mathbf{u}}_0$, $\mathbf{P}_0^{\mathbf{u}}$, $\mathbf{Q}_k^{\mathbf{x}}$, $\mathbf{R}_k$

Output: $\widehat{\mathbf{u}}_k$, $\mathbf{P}_k^{\mathbf{u}}$, $\widehat{\mathbf{x}}_k$, $\mathbf{P}_k^{\mathbf{x}}$

0. Initialization

$$\widetilde{\mathbf{x}}_1 = \mathbf{A}\widehat{\mathbf{x}}_0 + \mathbf{B}\widehat{\mathbf{u}}_0$$
$$\widetilde{\mathbf{P}}_1^{\mathbf{x}} = \mathbf{A}\mathbf{P}_0\mathbf{A}^{\mathsf{T}} + \mathbf{B}\mathbf{P}_0^{\mathbf{u}}\mathbf{B}^{\mathsf{T}} + \mathbf{Q}_{\mathbf{0}}^{\mathbf{x}}$$

for *each time step* $k > 0$ **do**

1. Input estimation

$$\mathbf{S}_k = \mathbf{C}\widetilde{\mathbf{P}}_k^{\mathbf{x}}\mathbf{C}^{\mathsf{T}} + \mathbf{R}_k$$
$$\mathbf{P}_k^{\mathbf{u}} = (\mathbf{D}\mathbf{S}_k^{-1}\mathbf{D}^{\mathsf{T}})^{-1}$$
$$\mathbf{K}_k^{\mathbf{u}} = \mathbf{P}_k^{\mathbf{u}}\mathbf{D}^{\mathsf{T}}\mathbf{S}_k^{-1}$$
$$\widehat{\mathbf{u}}_k = \mathbf{K}_k^{\mathbf{u}}(\mathbf{y}_k - \mathbf{C}\widetilde{\mathbf{x}}_k)$$

2. State estimation

$$\mathbf{K}_k^{\mathbf{x}} = \widetilde{\mathbf{P}}_k^{\mathbf{x}}\mathbf{C}^{\mathsf{T}}\mathbf{S}_k^{-1}$$
$$\widehat{\mathbf{x}}_k = \widetilde{\mathbf{x}}_k + \mathbf{K}_k^{\mathbf{x}}(\mathbf{y}_k - \mathbf{C}\widetilde{\mathbf{x}}_k - \mathbf{D}\widehat{\mathbf{u}}_k)$$
$$\mathbf{P}_k^{\mathbf{x}} = (\mathbf{I} - \mathbf{K}_k^{\mathbf{x}}\mathbf{C})\widetilde{\mathbf{P}}_k^{\mathbf{x}} + \mathbf{K}_k^{\mathbf{x}}\mathbf{D}\mathbf{P}_k^{\mathbf{u}}\mathbf{D}^{\mathsf{T}}\mathbf{K}_k^{\mathbf{x}\mathsf{T}}$$
$$\mathbf{P}_k^{\mathbf{xu}} = -\mathbf{K}_k^{\mathbf{x}}\mathbf{D}\mathbf{P}_k^{\mathbf{u}}$$

3. State prediction

$$\widetilde{\mathbf{x}}_{k+1} = \mathbf{A}\widehat{\mathbf{x}}_k + \mathbf{B}\widehat{\mathbf{u}}_k$$
$$\widetilde{\mathbf{P}}_{k+1}^{\mathbf{x}} = \begin{bmatrix}\mathbf{A} & \mathbf{B}\end{bmatrix}\begin{bmatrix}\mathbf{P}_k^{\mathbf{x}} & \mathbf{P}_k^{\mathbf{xu}} \\ \mathbf{P}_k^{\mathbf{xu}\mathsf{T}} & \mathbf{P}_k^{\mathbf{u}}\end{bmatrix}\begin{bmatrix}\mathbf{A}^{\mathsf{T}} \\ \mathbf{B}^{\mathsf{T}}\end{bmatrix} + \mathbf{Q}_k^{\mathbf{x}}$$

end

Algorithm 5: Sequential Bayesian Filter – SBF

Input: $\mathbf{y}_k, \mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}, \widehat{\mathbf{x}}_0, \mathbf{P}^{\mathbf{x}}_0, \widehat{\mathbf{u}}_0, \mathbf{P}^{\mathbf{u}}_0, \mathbf{Q}^{\mathbf{x}}_k, \mathbf{R}_k$

Output: $\widehat{\mathbf{u}}_k$, $\mathbf{P}^{\mathbf{u}}_k$, $\widehat{\mathbf{x}}_k$, $\mathbf{P}^{\mathbf{x}}_k$

0. Initialization

$$\widetilde{\mathbf{x}}_1 = \mathbf{A}\widehat{\mathbf{x}}_0 + \mathbf{B}\widehat{\mathbf{u}}_0$$

$$\widetilde{\mathbf{P}}^{\mathbf{x}}_1 = \mathbf{A}\mathbf{P}_0\mathbf{A}^{\mathsf{T}} + \mathbf{B}\mathbf{P}^{\mathbf{u}}_0\mathbf{B}^{\mathsf{T}} + \mathbf{Q}^{\mathbf{x}}_{\mathbf{0}}$$

for *each time step* $k > 0$ **do**

1. Input estimation

$$\mathbf{K}^{\mathbf{u}}_k = \mathbf{P}^{\mathbf{u}}_{k-1}\mathbf{D}^{\mathsf{T}}(\mathbf{D}\mathbf{P}^{\mathbf{u}}_{k-1}\mathbf{D}^{\mathsf{T}} + \mathbf{R}_k)^{-1}$$

$$\widehat{\mathbf{u}}_k = \mathbf{K}^{\mathbf{u}}_k(\mathbf{y}_k - \mathbf{C}\mathbf{A}\widehat{\mathbf{x}}_{k-1})$$

$$\mathbf{P}^{\mathbf{u}}_k = (\mathbf{I} - \mathbf{K}^{\mathbf{u}}_k\mathbf{D})\mathbf{P}^{\mathbf{u}}_{k-1}(\mathbf{I} - \mathbf{K}^{\mathbf{u}}_k\mathbf{D})^{\mathsf{T}} + \mathbf{K}^{\mathbf{u}}_k(\mathbf{C}\widetilde{\mathbf{P}}^{\mathbf{x}}_k\mathbf{C}^{\mathsf{T}} + \mathbf{R}_k)\mathbf{K}^{\mathbf{u}\mathsf{T}}_k$$

2. State estimation

$$\mathbf{K}^{\mathbf{x}}_k = \widetilde{\mathbf{P}}^{\mathbf{x}}_k\mathbf{C}^{\mathsf{T}}(\mathbf{C}\widetilde{\mathbf{P}}^{\mathbf{x}}_k\mathbf{C}^{\mathsf{T}} + \mathbf{R}_k)^{-1}$$

$$\widehat{\mathbf{x}}_k = \widetilde{\mathbf{x}}_k + \mathbf{K}^{\mathbf{x}}_k(\mathbf{y}_k - \mathbf{C}\widetilde{\mathbf{x}}_k - \mathbf{D}\widehat{\mathbf{u}}_k)$$

$$\mathbf{P}^{\mathbf{x}}_k = (\mathbf{I} - \mathbf{K}^{\mathbf{x}}_k\mathbf{C})\widetilde{\mathbf{P}}^{\mathbf{x}}_k(\mathbf{I} - \mathbf{K}^{\mathbf{x}}_k\mathbf{C})^{\mathsf{T}} + \mathbf{K}^{\mathbf{x}}_k(\mathbf{D}\mathbf{P}^{\mathbf{u}}_k\mathbf{D}^{\mathsf{T}} + \mathbf{R}_{\mathbf{k}})\mathbf{K}^{\mathbf{x}\mathsf{T}}_k$$

$$\mathbf{P}^{\mathbf{xu}}_k = -\mathbf{K}^{\mathbf{x}}_k\mathbf{D}\mathbf{P}^{\mathbf{u}}_k$$

3. State prediction

$$\widetilde{\mathbf{x}}_{k+1} = \mathbf{A}\widehat{\mathbf{x}}_k + \mathbf{B}\widehat{\mathbf{u}}_k$$

$$\widetilde{\mathbf{P}}^{\mathbf{x}}_{k+1} = \begin{bmatrix}\mathbf{A} & \mathbf{B}\end{bmatrix}\begin{bmatrix}\mathbf{P}^{\mathbf{x}}_k & \mathbf{P}^{\mathbf{xu}}_k \\ \mathbf{P}^{\mathbf{xu}\mathsf{T}}_k & \mathbf{P}^{\mathbf{u}}_k\end{bmatrix}\begin{bmatrix}\mathbf{A}^{\mathsf{T}} \\ \mathbf{B}^{\mathsf{T}}\end{bmatrix} + \mathbf{Q}^{\mathbf{x}}_k$$

end

Algorithm 6: Dual Kalman Filter – DKF

Input: $\mathbf{y}_k$, $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$, $\widehat{\mathbf{x}}_0, \mathbf{P}^{\mathbf{x}}_0, \widehat{\mathbf{u}}_0, \mathbf{P}^{\mathbf{u}}_0, \mathbf{Q}^{\mathbf{x}}_k$, $\mathbf{Q}^{\mathbf{u}}_k$, $\mathbf{R}_k$

Output: $\widehat{\mathbf{u}}_k$, $\mathbf{P}^{\mathbf{u}}_k$, $\widehat{\mathbf{x}}_k$, $\mathbf{P}^{\mathbf{x}}_k$

for *each time step* $k > 0$ **do**

1. Input prediction

$$\widetilde{\mathbf{u}}_k = \widehat{\mathbf{u}}_{k-1}$$

$$\widetilde{\mathbf{P}}^{\mathbf{u}}_k = \mathbf{P}^{\mathbf{u}}_{k-1} + \mathbf{Q}^{\mathbf{u}}_{k-1}$$

2. Input estimation

$$\mathbf{K}^{\mathbf{u}}_k = \widetilde{\mathbf{P}}^{\mathbf{u}}_k \mathbf{D}^{\mathsf{T}} (\mathbf{D}\widetilde{\mathbf{P}}^{\mathbf{u}}_k \mathbf{D}^{\mathsf{T}} + \mathbf{R}_k)^{-1}$$

$$\widehat{\mathbf{u}}_k = \widetilde{\mathbf{u}}_k + \mathbf{K}^{\mathbf{u}}_k (\mathbf{y}_k - \mathbf{C}\widehat{\mathbf{x}}_{k-1} - \mathbf{D}\widetilde{\mathbf{u}}_k)$$

$$\mathbf{P}^{\mathbf{u}}_k = (\mathbf{I} - \mathbf{K}^{\mathbf{u}}_k \mathbf{D})\widetilde{\mathbf{P}}^{\mathbf{u}}_k$$

3. State prediction

$$\widetilde{\mathbf{x}}_k = \mathbf{A}\widehat{\mathbf{x}}_{k-1} + \mathbf{B}\widehat{\mathbf{u}}_k$$

$$\widetilde{\mathbf{P}}^{\mathbf{x}}_k = \mathbf{A}\mathbf{P}^{\mathbf{x}}_{k-1}\mathbf{A}^{\mathsf{T}} + \mathbf{Q}^{\mathbf{x}}_{k-1}$$

4. State estimation

$$\mathbf{K}^{\mathbf{x}}_k = \widetilde{\mathbf{P}}^{\mathbf{x}}_k \mathbf{C}^{\mathsf{T}} (\mathbf{C}\widetilde{\mathbf{P}}^{\mathbf{x}}_k \mathbf{C}^{\mathsf{T}} + \mathbf{R}_k)^{-1}$$

$$\widehat{\mathbf{x}}_k = \widetilde{\mathbf{x}}_k + \mathbf{K}^{\mathbf{x}}_k (\mathbf{y}_k - \mathbf{C}\widetilde{\mathbf{x}}_k - \mathbf{D}\widehat{\mathbf{u}}_k)$$

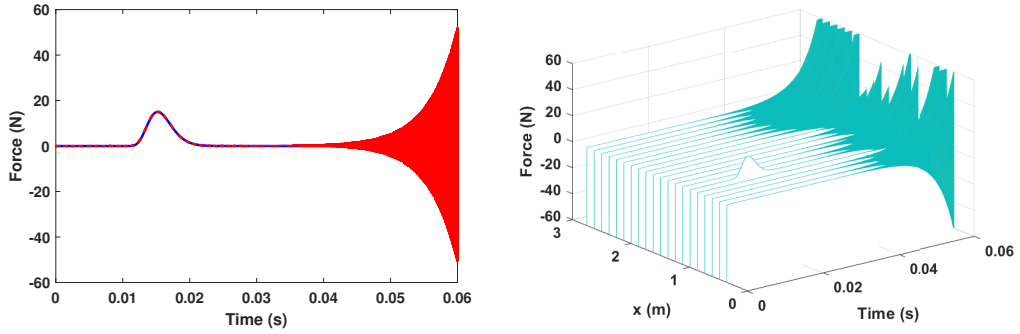
$$\mathbf{P}^{\mathbf{x}}_k = (\mathbf{I} - \mathbf{K}^{\mathbf{x}}_k \mathbf{C})\widetilde{\mathbf{P}}^{\mathbf{x}}_k$$

end

B. Numerical results obtained with SBF

This section presents the results obtained with SBF for the default configuration of the numerical application presented in section 6. As explained in the core of the text and illustrated in Fig. B.1, SBF diverges for the set

of parameters defined in section 6.1. The same observation has been made when using experimental data. That is why, SBF has not been considered for comparison in the present paper.



(a) (b)

Figure B.1: Input estimation from SBF for the default configuration of the numerical application – (a) Time history and (b) Spatial distribution – (—) Reference and (– –) SBF

References

[1] H. R. Busby and D. M. Trujillo. Solution of an inverse dynamics problem using an eigenvalue reduction technique. *Computers & Structures*, 25 (1):109–117, 1987.

[2] L. J. L. Nordström. A dynamic programming algorithm for input estimation on linear time-variant systems. *Computers Methods in Applied Mechanics and Engineering*, 195:6407–6427, 2006.

[3] A. González, C. Rowley, and E. J. OBrien. A general solution to the

identification of moving vehicle forces on a bridge. *International Journal for Numerical Methods in Engineering*, 75:335–354, 2008.

[4] R. T. Jones, J. S. Sirkis, and E. J. Friebele. Detection of impact location and magnitude for isotropic plates using neural networks. *Journal of Intelligent Material Systems and Structures*, 8:90–99, 1997.

[5] K. Worden and W. J. Staszewski. Impact location and quantification on a composite panel using neural networks and a genetic algorithm. *Strain*, 36 (2):61–68, 2000.

[6] J. M. Zhou, L. Dong, W. Guan, and J. Yan. Impact load identification of nonlinear structures using deep recurrent neural network. *Mechanical Systems and Signal Processing*, 133:106292, 2019.

[7] J. Tabian, H. Fu, and Z. S. Khodaei. A convolutional neural network for impact detection and characterization of complex composite structures. *Sensors*, 19:4933, 2019.

[8] P. O'Donoghue, O. Robin, and A. Berry. Time-resolved identification of mechanical loadings on plates using the virtual field method and deflectometry measurement. *Strain*, 54:e12258, 2018.

[9] P. O'Donoghue, O. Robin, and A. Berry. Time-space identification of mechanical impacts and distributed random excitations on plates and membranes. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 233 (18):6436–6447, 2019.

[10] N. Aujogue and A. Ross. Transient force analysis technique to identify time-variying loads and defects on plates. In *Proceedings of the 26th International Congress on Sound and Vibration*, Montréal, Canada, 2019.

[11] C. Pezerat, Q. Leclere, E. Le Roux, and J.-H. Thomas. Adapting the correction for CFAT application in time domain. In *Proceedings of Forum Acusticum*, Lyon, France, 2020.

[12] D. Bernal and A. Ussia. Sequential deconvolution input reconstruction. *Mechanical Systems and Signal Processing*, 50-51:41–55, 2015.

[13] D. Bernal. Non-recursive sequential input deconvolution. *Mechanical Systems and Signal Processing*, 82:296–306, 2017.

[14] A. N. Tikhonov. Solution of incorrectly formulated problems and the regularization method. *Soviet Mathematics*, 4:1035–1038, 1963.

[15] R. E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME - Journal of Basic Engineering*, 82:35–45, 1960.

[16] A. Tarantola. *Inverse problem theory and methods for model parameter estimation*. SIAM Philadelphia, 2005.

[17] S. Särkkä. *Bayesian filtering and smoothing*. Cambridge University Press, 2013.

[18] E. Jacquelin, A. Bennani, and P. Hamelin. Design of sensor networks for instantaneous inversion of modally reduced order models in structural

dynamics. *Mechanical Systems and Signal Processing*, 52-53:628–644, 2015.

[19] Y.-M. Mao, X.-L. Guo, and Y. Zhao. Experimental study of hammer impact identification on a steel cantilever beam. *Experimental techniques*, 34:82–85, 2010.

[20] S. Samagassi, A. Khamlichi, A. Driouach, and E. Jacquelin. Reconstruction of multiple impact forces by wavelet relevance vector machine approach. *Journal of Sound and Vibration*, 359:56–67, 2015.

[21] G. Yan, H. Sun, and O. Büyüköztürk. Impact load identification for composite structures using bayesian regularization and unscented kalman filter. *Structural control and health monitoring*, 24:e1910, 2017.

[22] D. Ginsberg and C.-P. Fritzen. New approach for impact detection by finding sparse solution. In *Proceedings of ISMA 2014*, Leuven, Belgium, 2014.

[23] B. Qiao, X. Zhang, J. Gao, and X. Chen. Impact-force sparse reconstruction from highly incomplete and inaccurate measurements. *Journal of Sound and Vibration*, 376:72–94, 2016.

[24] M. Kirchner, J. Croes, F. Cosco, and W. Desmet. Exploiting input sparsity for joint state/input moving horizon estimation. *Mechanical Systems and Signal Processing*, 101:237–253, 2018.

[25] Q. Li and Q. Lu. A hierarchical bayesian method for vibration-based time domain for reconstruction problems. *Journal of Sound and Vibration*, 421:190–204, 2018.

[26] M. Aucejo, O. De Smet, and J.-F. Deü. On a space-time regularization for force reconstruction problems. *Mechanical Systems and Signal Processing*, 118:549–567, 2019.

[27] S. Gillijns and B. De Moor. Unbiased minimum-variance input and state estimation for linear discrete-time systems with direct feedthrough. *Automatica*, 43:934–937, 2007.

[28] E. Lourens, C. Papadimitriou, S. Gillijns, E. Reynders, G. De Roeck, and G. Lombaert. Joint input-response estimation for structural systems based on reduced-order models and vibration data from a limited number of sensors. *Mechanical Systems and Signal Processing*, 29:310–327, 2012.

[29] E. Lourens, E. Reynders, G. De Roeck, G. Degrande, and G. Lombaert. An augmented kalman filter for force identification in structural dynamics. *Mechanical Systems and Signal Processing*, 27:446–460, 2012.

[30] S. Eftekhar Azam, E. Chatzi, and C. Papadimitriou. A dual kalman filter approach for state estimation via output-only acceleration measurements. *Mechanical Systems and Signal Processing*, 60-61:866–886, 2015.

[31] S. Eftekhar Azam, E. Chatzi, C. Papadimitriou, and A. Smyth. Experimental validation of kalman-type filters for online and real-time state and input estimation. *Journal of Vibration and Control*, 23 (15):2494–2519, 2017.

[32] R. Nayek, S. Chakraborty, and S. Narasimhan. A gaussian latent force model for joint input-state estimation in linear structural systems. *Mechanical Systems and Signal Pocessing*, 128:497–530, 2019.

[33] F. Naets, J. Cuadrado, and W. Desmet. Stable force identification in structural dynamics using kalman filtering and dummy-measurements. *Mechanical Systems and Signal Processing*, 50-51:235–248, 2015.

[34] D. Wei, D. Li, and J. Huang. Improved force identification with augmented kalman filter based on the sparse constraint. *Mechanical Systems and Signal Processing*, 167, 2022.

[35] O. Sedehi, C. Papadimitriou, D. Teymouri, and L. S. Katafygiotis. Sequential bayesian estimation of state and input in dynamical systems using outpout-only measurements. *Mechanical Systems and Signal Processing*, 131:659–688, 2019.

[36] M. A. Sutton, J.-J. Orteu, and H. W. Schreier. *Image correlation for shape, motion and deformation measurements: Basic concepts, theory and applications*. Springer, Boston, MA, 2009.

[37] J. Poittevin, F. Gautier, C. Pezerat, and P. Picart. High-speed holographic metrology: principle, limitations and application to vibroacoustics of structures. *Optical Engineering*, 55 (12):121717, 2016.

[38] O. Robin, P. O'Donoghue, A. Berry, V. Farley, and K. Prithipaul. Full-field vibration measurements on a cantilever beam under impact using visible and infrared deflectometry. *Applied Acoustics*, 183:108294, 2021.

[39] K. Liu, S. S. Law, X. Q. Zhu, and Y. Xia. Explicit form of an implicit method for inverse force identification. *Journal of Sound and Vibration*, 333:730–744, 2014.

[40] T. Lai, T.-H. Yi, H.-N. Li, and X. Fu. An explicit fourth-order runge-kutta method for dynamic force identification. *International Journal of Structural Stability and Dynamics*, 17 (10):1750120, 2017.

[41] M. Aucejo, O. De Smet, and J.-F. Deü. Practical issues on the applicability of kalman filtering for reconstructing mechanical sources in structural dynamics. *Journal of Sound and Vibration*, 442:45–70, 2019.

[42] S. Nadarajah. A generalized normal distribution. *Journal of Applied Statistics*, 32 (7):685–694, 2005.

[43] M. Aucejo and O. De Smet. An optimal bayesian regularization for force reconstruction problems. *Mechanical Systems and Signal Processing*, 126:98–115, 2019.

[44] Mathieu Aucejo and Olivier de Smet. On a full bayesian inference for force reconstruction problems. *Mechanical Systems and Signal Processing*, 104:36–59, 2018.

[45] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University Press, 2004.

[46] S. G. Nash. Newton-type minimization via lanczos method. *SIAM Journal of Numerical Analysis*, 21:770–778, 1984.

[47] R. H. Byrd, P. Lu, and J. Nocedal. A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific and Statistical Computing*, 16 (5):1190–1208, 1995.

[48] K. Maes, E. Lourens, K. Van Nimmen, E. Reynders, G. De Roeck, and G. Lombaert. Force reconstruction: analysis and regularization of a deconvolution problem. *Journal of Sound and Vibration*, 265 (1):81–107, 2003.