Simultaneous input and state estimation for nonlinear systems with applications to flow field estimation ⋆

Huazhen Fang [a] Raymond A. de Callafon [a] Jorge Cortés [a]

[a] *Mechanical & Aerospace Engineering, University of California, San Diego, CA 92093, USA*

Abstract

This paper studies the problem of simultaneous input and state estimation (SISE) for nonlinear dynamical systems with and without direct input-output feedthrough. We take a Bayesian perspective to develop a sequential joint input and state estimation approach. Our scheme gives rise to a nonlinear Maximum a Posteriori optimization problem, which we solve using a classical Gauss-Newton method. The proposed approach generalizes a number of SISE methods presented in the literature. We illustrate the effectiveness of the proposed scheme for nonlinear systems with direct feedthrough in an oceanographic flow field estimation problem involving submersible buoys that measure position intermittently and acceleration continuously.

Key words: Bayesian filtering; input estimation; state estimation; nonlinear system.

1 Introduction

This paper deals with the problem of *nonlinear simultaneous input and state estimation (NL-SISE)*, which is concerned with simultaneously and sequentially estimating the unknown inputs and states of a nonlinear system from only the output measurements. Our motivation for this work comes from oceanographic applications where a group of submersible drogues are deployed to acquire data to reconstruct the three-dimensional flow field in a region of the ocean. The unknown ocean flows act as external inputs in the dynamic model of the drogue. Hence, the aim is to reconstruct the flow velocities (inputs) and the drogues' trajectories and velocities (states) given their position and acceleration measurements.

Literature review: The broad range of applications of simultaneous input and state estimation (SISE) in areas such as disturbance rejection, weather forecasting, and oceanography has stimulated continued research interest in this topic during the past decades. Early work in (Friedland, 1969) studies state estimation with unknown inputs which are modeled by stochastic processes with known wide-sense description (e.g., mean and covariance). Kitanidis (1987) employs minimum variance unbiased estimation (MVUE) to deal with the scenario of completely unknown inputs. Based on (Kitanidis, 1987), recent MVUE-based works (Darouach and Zasadzinski, 1997; Cheng et al., 2009) establish the conditions for the existence of unbiased estimators and the stability of the filters developed therein. In the above mentioned works, only the state estimation is conducted, leaving the unknown input estimation untackled. As input information is often as important as state information, SISE puts the emphasis on joint estimation of both. A majority of works consider linear discrete-time dynamical systems. An early contribution in this respect is (Mendel, 1977), in which a Kalman filter (KF) based approach is developed to estimate the states and white noises disturbances of a linear system, with the assumption of known noise covariance. Most current works treat the case of completely unknown inputs and build on existing state estimation techniques. Among them, we highlight KF (Hsieh, 2000, 2010, 2011), moving horizon estimation (MHE) (Pina and Botto, 2006), H_∞-filtering (You et al., 2008), sliding mode observers (Floquet et al., 2007), and MVUE (Gillijns and De Moor, 2007a,b). The MVUE-based filters in (Gillijns and De Moor, 2007a,b) are optimal among all linear unbiased state and input estimators in the sense of minimum mean square error. SISE for nonlinear systems is more challenging. As is well known, the extended KF (EKF) for nonlinear systems proceeds by linearizing about the current state estimate and us-

⋆ This work is partially supported by NSF award OCE-0941692. This material in this paper was not presented at any IFAC meeting. Corresponding author: Raymond A. de Callafon. Tel. +1-858-534-3166. Fax +1-858-822-3107.

Email addresses: `hzfang@ucsd.edu` (Huazhen Fang), `callafon@ucsd.edu` (Raymond A. de Callafon), `cortes@ucsd.edu` (Jorge Cortés).

ing the KF on the resulting system. However, such a linearization based technique is not truly useful in extending the Gillijns-De Moor's algorithms to nonlinear systems, because unknown inputs make it futile to linearize nonlinear functions where the state and input are coupled. Among the few results in the literature that study NL-SISE, we highlight (Corless and Tu, 1998; Ha and Trinh, 2004) on a special class of nonlinear systems that consists of a nominally linear part and a nonlinear part: the former work regards the nonlinear term as unknown state-dependent input to estimate, while the latter considers a Lipschitz nonlinear function with respect to both state and input.

Statement of contributions: The contributions of this paper are three-fold. First, a Bayesian framework is developed to deal with SISE. Our treatment extends the Bayesian approach employed in various other works for state and parameter estimation to jointly estimate the inputs and the states. Second, the `NL-SISE-wDF algorithm` and the `NL-SISE-w/oDF algorithm` are proposed within the Bayesian framework for nonlinear systems with and without direct input-output feedthrough, with the observation that research relevant to NL-SISE is still insufficient to date. Third, this paper combines theoretical studies and the application of oceanic flow field reconstruction. The proposed algorithms, as demanded by the application, have advantages in conceptual simplicity and practical effectiveness, and maintain a good balance between estimation performance and computational complexities.

2 Bayesian Paradigm for Nonlinear Systems with Direct Feedthrough

We consider nonlinear systems with direct feedthrough of the form

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{f}(\mathbf{u}_k, \mathbf{x}_k) + \mathbf{w}_k, \\ \mathbf{y}_k = \mathbf{h}(\mathbf{u}_k, \mathbf{x}_k) + \mathbf{v}_k, \end{cases} \tag{1}$$

where $\mathbf{u} \in \mathbb{R}^m$ is the input vector, $\mathbf{x} \in \mathbb{R}^n$ is the state vector, $\mathbf{y} \in \mathbb{R}^p$ is measurement vector, and $\mathbf{w} \in \mathbb{R}^n$ and $\mathbf{v} \in \mathbb{R}^m$ are mutually independent zero-mean white Gaussian noise sequences, with covariances $\mathbf{Q}_k$ and $\mathbf{R}_k$, respectively. The mappings $\mathbf{f} : \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}^n$ and $\mathbf{h} : \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}^p$ are the state transition and measurement functions, respectively, which is assumed to be C^1. We also assume $\nabla_{\mathbf{u}}\mathbf{h}$ has full rank. For the above system, our objective is to design a NL-SISE filter to estimate $\mathbf{u}_k$ and $\mathbf{x}_k$ from the measurement set $\mathcal{Y}_k = \{\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_k\}$ for each k.

Bayesian statistics have historically provided a framework for developing estimation schemes such as the classical KF and particle filters (Candy, 2009). A Bayesian estimator proceeds by estimating the probability density functions (pdf's) of unknown variables conditioned on available measurements. The goal in this section is to sequentially compute $p(\mathbf{u}_k, \mathbf{x}_k|\mathcal{Y}_k)$ from $p(\mathbf{u}_{k-1}, \mathbf{x}_{k-1}|\mathcal{Y}_{k-1})$. Like in Bayesian state estimation, this can also be accomplished in a two-step procedure of prediction and update.

Prediction is to determine $p(\mathbf{x}_k|\mathcal{Y}_{k-1})$. By the Chapman-Kolmogorov equation (Honerkamp, 1993), we have

$$p(\mathbf{x}_k|\mathcal{Y}_{k-1}) = \iint p(\mathbf{x}_k|\mathbf{u}_{k-1}, \mathbf{x}_{k-1}, \mathcal{Y}_{k-1}) \cdot p(\mathbf{u}_{k-1}, \mathbf{x}_{k-1}|\mathcal{Y}_{k-1}) \mathrm{d}\mathbf{u}_{k-1} \mathrm{d}\mathbf{x}_{k-1}.$$

We have $p(\mathbf{x}_k|\mathbf{u}_{k-1}, \mathbf{x}_{k-1}, \mathcal{Y}_{k-1}) = p(\mathbf{x}_k|\mathbf{u}_{k-1}, \mathbf{x}_{k-1})$, since $\mathbf{x}_k$ depends on $\mathbf{u}_{k-1}$ and $\mathbf{x}_{k-1}$, as the state equation in (1) is Markovian with order one. Hence, it follows that

$$p(\mathbf{x}_k|\mathcal{Y}_{k-1}) = \iint p(\mathbf{x}_k|\mathbf{u}_{k-1}, \mathbf{x}_{k-1}) \cdot p(\mathbf{u}_{k-1}, \mathbf{x}_{k-1}|\mathcal{Y}_{k-1}) \mathrm{d}\mathbf{u}_{k-1} \mathrm{d}\mathbf{x}_{k-1}, \tag{2}$$

where $p(\mathbf{x}_k|\mathbf{u}_{k-1}, \mathbf{x}_{k-1})$ can be determined via (1).

At time instant k, the measurement $\mathbf{y}_k$ can be used to update $p(\mathbf{x}_k|\mathcal{Y}_{k-1})$ and, at the same time, to jointly estimate the conditional pdf of $\mathbf{u}_k$ (since it is the first measurement containing information about $\mathbf{u}_k$) via $p(\mathbf{u}_k, \mathbf{x}_k|\mathcal{Y}_k)$. To proceed, we make the following assumption:

(A1) $\{\mathbf{u}_k\}$ is a *white* process, independent of $\mathbf{x}_0$, $\{\mathbf{w}_k\}$ and $\{\mathbf{v}_k\}$.

Here, 'white' means that $\mathbf{u}_k$ and $\mathbf{u}_l$ are independent random variables for $k \neq l$. Such a whiteness assumption is inspired by (Robinson, 1957), which has been a foundation for many seismic data processing algorithms. The intuitions underlying it are: (1) $\mathbf{u}_k$, completely unknown to us, may assume all possible values; (2) from the knowledge of $\mathbf{u}_k$ we cannot predict $\mathbf{u}_l$ for $k \neq l$. A similar treatment of $\{\mathbf{u}_k\}$ as a stochastic process is proposed in (Friedland, 1969), yet with its wide-sense description assumed known. By (A1), $\mathbf{u}_k$ is independent of $\mathbf{x}_k$ and $\mathcal{Y}_{k-1}$ (Gut, 2005, Theorem 10.4, pp. 71).

Using the Bayes' rule repeatedly, we obtain

$$p(\mathbf{u}_k, \mathbf{x}_k|\mathcal{Y}_k) = \frac{p(\mathbf{y}_k|\mathbf{u}_k, \mathbf{x}_k, \mathcal{Y}_{k-1}) p(\mathbf{u}_k, \mathbf{x}_k|\mathcal{Y}_{k-1})}{p(\mathbf{y}_k|\mathcal{Y}_{k-1})}.$$

Note that $p(\mathbf{y}_k|\mathbf{u}_k, \mathbf{x}_k, \mathcal{Y}_{k-1}) = p(\mathbf{y}_k|\mathbf{u}_k, \mathbf{x}_k)$ due to the fact that $\mathbf{y}_k$ entirely depends on $\mathbf{u}_k$ and $\mathbf{x}_k$, and that $p(\mathbf{u}_k, \mathbf{x}_k|\mathcal{Y}_{k-1}) = p(\mathbf{x}_k|\mathcal{Y}_{k-1}) p(\mathbf{u}_k)$ as a result of $\mathbf{u}_k$'s independence from $\mathbf{x}_k$ and $\mathcal{Y}_{k-1}$. Consequently,

$$p(\mathbf{u}_k, \mathbf{x}_k|\mathcal{Y}_k) = \frac{p(\mathbf{y}_k|\mathbf{u}_k, \mathbf{x}_k) p(\mathbf{x}_k|\mathcal{Y}_{k-1}) p(\mathbf{u}_k)}{p(\mathbf{y}_k|\mathcal{Y}_{k-1})}$$

It is seen that $p(\mathbf{u}_k)/p(\mathbf{y}_k|\mathcal{Y}_{k-1})$ plays the role of a proportionality coefficient because

$$\frac{p(\mathbf{u}_k)}{p(\mathbf{y}_k|\mathcal{Y}_{k-1})} = \frac{1}{\iint p(\mathbf{y}_k|\mathbf{u}_k, \mathbf{x}_k)p(\mathbf{x}_k|\mathcal{Y}_{k-1})\mathrm{d}\mathbf{u}_k\mathrm{d}\mathbf{x}_k}.$$

Thus we have

$$p(\mathbf{u}_k, \mathbf{x}_k|\mathcal{Y}_k) \propto p(\mathbf{y}_k|\mathbf{u}_k, \mathbf{x}_k)p(\mathbf{x}_k|\mathcal{Y}_{k-1}), \tag{3}$$

where $p(\mathbf{y}_k|\mathbf{u}_k, \mathbf{x}_k)$ can be determined from the output equation in (1), and $p(\mathbf{x}_k|\mathcal{Y}_{k-1})$ is obtained in the prediction procedure (2).

Sequentially updating (2) and (3) not only provides the Bayesian solution to SISE in the presence of direct feedthrough, but also yields a statistical framework, within which different SISE methods can be developed. Our next step is to derive the `NL-SISE-wDF algorithm` using MAP by virtue of (2) and (3).

3 NL-SISE Algorithm for Nonlinear Systems with Direct Feedthrough

This section develops the `NL-SISE-wDF algorithm` for nonlinear systems with direct feedthrough. We begin with some multivariate Gaussian distribution assumptions and then proceed to describe the prediction and update steps of our scheme developed in the context of the Bayesian paradigm of Section 2.

3.1 Assumptions

We make the following assumptions:

(A2) $p(\mathbf{u}_k, \mathbf{x}_k|\mathcal{Y}_k) \sim \mathcal{N}\left(\begin{bmatrix}\hat{\mathbf{u}}_k \\ \hat{\mathbf{x}}_k^+\end{bmatrix}, \begin{bmatrix}\mathbf{P}_k^{\mathbf{u}} & \mathbf{P}_k^{\mathbf{ux}} \\ (\mathbf{P}_k^{\mathbf{ux}})^\top & \mathbf{P}_k^{\mathbf{x}+}\end{bmatrix}\right)$,

(A3) $p(\mathbf{y}_k|\mathbf{u}_k, \mathbf{x}_k) \sim \mathcal{N}\left(\mathbf{h}(\mathbf{u}_k, \mathbf{x}_k), \mathbf{R}_k\right)$,

(A4) $p(\mathbf{x}_k|\mathcal{Y}_{k-1}) \sim \mathcal{N}\left(\hat{\mathbf{x}}_k^-, \mathbf{P}_k^{\mathbf{x}-}\right)$,

where $\hat{\mathbf{u}}_k$ is the estimate of $\mathbf{u}_k$ given $\mathcal{Y}_k$ with associated covariance $\mathbf{P}_k^{\mathbf{u}}$, $\hat{\mathbf{x}}_k^-$ and $\hat{\mathbf{x}}_k^+$ are the estimates of $\mathbf{x}_k$ given $\mathcal{Y}_{k-1}$ and $\mathcal{Y}_k$ with covariances $\mathbf{P}_k^{\mathbf{x}-}$ and $\mathbf{P}_k^{\mathbf{x}+}$, respectively.

Ideally, if knowledge of $p(\mathbf{u}_k, \mathbf{x}_k|\mathcal{Y}_k)$ is available for each k, $\hat{\mathbf{u}}_k$ and $\hat{\mathbf{x}}_k^+$ can be simply obtained by MAP or conditional mean calculation. However, it is often intractable to compute $p(\mathbf{u}_k, \mathbf{x}_k|\mathcal{Y}_k)$ accurately for nonlinear systems. In order to overcome the problem, (A2)-(A4) are made to approximately describe the pdf's, thus paving the way for development of a sequentially updating NL-SISE algorithm from the Bayesian paradigm. Gaussianity assumptions analogous to (A2)-(A4) are commonly held in estimation algorithms for nonlinear systems, e.g., (Anderson and Moore, 1979; Bell and Cathey, 1993; Spinello and Stilwell, 2010).

3.2 Prediction Step

To develop the state prediction procedure, let us use (2). Define

$$\hat{\mathbf{x}}_k^- = \arg\max_{\mathbf{x}_k} p(\mathbf{x}_k|\mathcal{Y}_{k-1}). \tag{4}$$

Consider the first-order Taylor series expansion of $\mathbf{f}(\mathbf{u}_k, \mathbf{x}_k)$ around $(\hat{\mathbf{u}}_k, \hat{\mathbf{x}}_k^+)$

$$\mathbf{f}(\mathbf{u}_k, \mathbf{x}_k) \approx \mathbf{f}(\hat{\mathbf{u}}_k, \hat{\mathbf{x}}_k^+) + \nabla\mathbf{f}(\hat{\mathbf{u}}_k, \hat{\mathbf{x}}_k^+)\begin{bmatrix}\mathbf{u}_k - \hat{\mathbf{u}}_k \\ \mathbf{x}_k - \hat{\mathbf{x}}_k^+\end{bmatrix}, \tag{5}$$

where $\nabla\mathbf{f} = \begin{bmatrix}\nabla_{\mathbf{u}}\mathbf{f} & \nabla_{\mathbf{x}}\mathbf{f}\end{bmatrix}$. Then by (A2), (2) and (5), the solution to (4) is given by

$$\hat{\mathbf{x}}_k^- = \mathbf{f}(\hat{\mathbf{u}}_{k-1}, \hat{\mathbf{x}}_{k-1}^+), \tag{6}$$

with associated prediction error covariance $\mathbf{P}_k^{\mathbf{x}-}$ given by

$$\mathbf{P}_k \approx \nabla\mathbf{f}(\hat{\mathbf{u}}_{k-1}, \hat{\mathbf{x}}_{k-1}^+)\begin{bmatrix}\mathbf{P}_{k-1}^{\mathbf{u}} & \mathbf{P}_{k-1}^{\mathbf{ux}} \\ (\mathbf{P}_{k-1}^{\mathbf{ux}})^\top & \mathbf{P}_{k-1}^{\mathbf{x}+}\end{bmatrix} \cdot \nabla\mathbf{f}^\top(\hat{\mathbf{u}}_{k-1}, \hat{\mathbf{x}}_{k-1}^+) + \mathbf{Q}_{k-1}. \tag{7}$$

The equations (6) and (7) constitute the prediction formulae together, computing the state predicts and prediction error covariances, respectively. The computation at time instant k utilizes only $\hat{\mathbf{u}}_{k-1}$ and $\hat{\mathbf{x}}_{k-1}^+$, thus cutting down on storage of past data to relieve the computational burden.

3.3 Update Step

Define

$$\begin{bmatrix}\hat{\mathbf{u}}_k \\ \hat{\mathbf{x}}_k^+\end{bmatrix} = \arg\max_{\mathbf{u}_k, \mathbf{x}_k} p(\mathbf{u}_k, \mathbf{x}_k|\mathcal{Y}_k). \tag{8}$$

The approach used in (Bell and Cathey, 1993; Spinello and Stilwell, 2010) can be modified to address the above maximization problem to obtain $\hat{\mathbf{u}}_k$ and $\hat{\mathbf{x}}_k^+$, by applying the Gauss-Newton method to approximate the MAP estimates.

Referring to (3) and (A3)-(A4), we define the MAP cost function $L(\mathbf{u}_k, \mathbf{x}_k) = p(\mathbf{u}_k, \mathbf{x}_k|\mathcal{Y}_k)$. Hence,

$$L(\mathbf{u}_k, \mathbf{x}_k) = \lambda \cdot \exp\left[-\boldsymbol{\alpha}_k^\top\mathbf{R}_k^{-1}\boldsymbol{\alpha}_k - \boldsymbol{\beta}_k^\top(\mathbf{P}_k^{\mathbf{x}-})^{-1}\boldsymbol{\beta}_k\right],$$

where λ combines all the constants, $\boldsymbol{\alpha}_k = \mathbf{y}_k - \mathbf{h}(\mathbf{u}_k, \mathbf{x}_k)$ and $\boldsymbol{\beta}_k = \mathbf{x}_k - \hat{\mathbf{x}}_k^-$. It is easier to deal with the logarithmic cost function $\ell(\mathbf{u}_k, \mathbf{x}_k) = -\ln L(\mathbf{u}_k, \mathbf{x}_k)$:

$$\ell(\mathbf{u}_k, \mathbf{x}_k) = \delta + \mathbf{r}^\top(\mathbf{u}_k, \mathbf{x}_k)\mathbf{r}(\mathbf{u}_k, \mathbf{x}_k), \tag{9}$$

where $\delta = -\ln\lambda$ and

$$\mathbf{r}(\mathbf{u}_k, \mathbf{x}_k) = \begin{bmatrix} \mathbf{R}_k^{-\frac{1}{2}}\boldsymbol{\alpha}_k \\ (\mathbf{P}_k^{\mathbf{x}-})^{-\frac{1}{2}}\boldsymbol{\beta}_k \end{bmatrix}.$$

Then an equivalence to (8) is given by

$$\begin{bmatrix} \hat{\mathbf{u}}_k \\ \hat{\mathbf{x}}_k^+ \end{bmatrix} = \arg\min_{\mathbf{u}_k, \mathbf{x}_k} \ell(\mathbf{u}_k, \mathbf{x}_k). \tag{10}$$

An analytical solution to the nonlinear MAP optimization in (10) is often rather difficult to derive. However, indeed being a nonlinear least-squares problem, it can be numerically addressed using the Gauss-Newton method (Björck, 1996).

To find $\hat{\mathbf{u}}_k$ and $\hat{\mathbf{x}}_k^+$, the classical Gauss-Newton method iteratively computes the sequences of approximations $\hat{\mathbf{u}}_k^{(i)}$ and $\hat{\mathbf{x}}_k^{+(i)}$, where (i) denotes the iteration step. Specifically,

$$\begin{aligned} \hat{\boldsymbol{\xi}}_k^{(i+1)} &= \hat{\boldsymbol{\xi}}_k^{(i)} - \left[\nabla_{\boldsymbol{\xi}}^\top \mathbf{r}\left(\hat{\boldsymbol{\xi}}_k^{(i)}\right) \nabla_{\boldsymbol{\xi}} \mathbf{r}\left(\hat{\boldsymbol{\xi}}_k^{(i)}\right)\right]^{-1} \\ &\quad \cdot \nabla_{\boldsymbol{\xi}}^\top \mathbf{r}\left(\hat{\boldsymbol{\xi}}_k^{(i)}\right) \cdot \mathbf{r}\left(\hat{\boldsymbol{\xi}}_k^{(i)}\right), \end{aligned} \tag{11}$$

where $\boldsymbol{\xi}_k = \begin{bmatrix}\mathbf{u}_k^\top & \mathbf{x}_k^\top\end{bmatrix}^\top$, and $\nabla_{\boldsymbol{\xi}}\mathbf{r} = \begin{bmatrix}\nabla_{\mathbf{u}}\mathbf{r} & \nabla_{\mathbf{x}}\mathbf{r}\end{bmatrix}$.

We may let the initial guesses be $\hat{\mathbf{u}}_k^{(0)} = 0$ and $\hat{x}_k^{+(0)} = \hat{\mathbf{x}}_k^-$ for convenience, though they can be set to arbitrary values. The iteration continues until the iteration step (i) reaches the preselected maximum $i_{\max}$ or the difference between two consecutive iterations is less than a preselected small value $\epsilon > 0$. Then $\hat{\mathbf{u}}_k^{(i)}$ and $\hat{\mathbf{x}}_k^{+(i)}$ obtained in the final iteration will be assigned to $\hat{\mathbf{u}}_k$ and $\hat{\mathbf{x}}_k^+$, respectively.

The iteration process in (11) refines the input and state estimates continually by re-evaluating the joint estimator around the latest estimated input and state operating point. Despite demanding more computational power, the iteration based refinement enhances not only the estimation performance but also the robustness to nonlinearities. A balance will also be achieved between computational complexities and estimation performance by selecting a proper stopping condition.

The estimation error covariance is equal to the inverse of the Fisher information matrix, as is common in MAP estimators under Gaussian distributions (Mutambara, 1998). Then we have

$$\begin{bmatrix} \mathbf{P}_k^{\mathbf{u}} & \mathbf{P}_k^{\mathbf{ux}} \\ (\mathbf{P}_k^{\mathbf{ux}})^\top & \mathbf{P}_k^{\mathbf{x}+} \end{bmatrix} = \boldsymbol{\mathcal{I}}^{-1}(\hat{\mathbf{u}}_k, \hat{\mathbf{x}}_k^+), \tag{12}$$

where the Fisher information matrix $\boldsymbol{\mathcal{I}}$ is defined as

$$\boldsymbol{\mathcal{I}} = \begin{bmatrix} \boldsymbol{\mathcal{I}}^{\mathbf{u}} & \boldsymbol{\mathcal{I}}^{\mathbf{ux}} \\ (\boldsymbol{\mathcal{I}}^{\mathbf{ux}})^\top & \boldsymbol{\mathcal{I}}^{\mathbf{x}} \end{bmatrix} = \mathrm{E}\left(\begin{bmatrix} \nabla_{\mathbf{u}}^\top \ell \\ \nabla_{\mathbf{x}}^\top \ell \end{bmatrix} \begin{bmatrix} \nabla_{\mathbf{u}}\ell & \nabla_{\mathbf{x}}\ell \end{bmatrix}\right). \tag{13}$$

The explicit formulae for the gradients are as follows:

$$\begin{aligned} &\nabla_{\mathbf{u}}\mathbf{r} = \begin{bmatrix} -\mathbf{R}^{-\frac{1}{2}}\nabla_{\mathbf{u}}\mathbf{h} \\ \mathbf{0} \end{bmatrix}, \quad \nabla_{\mathbf{x}}\mathbf{r} = \begin{bmatrix} -\mathbf{R}^{-\frac{1}{2}}\nabla_{\mathbf{x}}\mathbf{h} \\ (\mathbf{P}^{\mathbf{x}-})^{-\frac{1}{2}} \end{bmatrix}, \\ &\nabla_{\mathbf{u}}\ell = \mathbf{r}^\top\nabla_{\mathbf{u}}\mathbf{r} = \boldsymbol{\alpha}^\top\mathbf{R}^{-1}\nabla_{\mathbf{u}}\mathbf{h}, \\ &\nabla_{\mathbf{x}}\ell = \mathbf{r}^\top\nabla_{\mathbf{x}}\mathbf{r} = \boldsymbol{\alpha}^\top\mathbf{R}^{-1}\nabla_{\mathbf{x}}\mathbf{h} + \boldsymbol{\beta}^\top\left(\mathbf{P}^{\mathbf{x}-}\right)^{-1}. \end{aligned}$$

Hence, $\boldsymbol{\mathcal{I}}$ is given by

$$\boldsymbol{\mathcal{I}} = \begin{bmatrix} \nabla_{\mathbf{u}}^\top\mathbf{h}\mathbf{R}^{-1}\nabla_{\mathbf{u}}\mathbf{h} & \nabla_{\mathbf{u}}^\top\mathbf{h}\mathbf{R}^{-1}\nabla_{\mathbf{x}}\mathbf{h} \\ \nabla_{\mathbf{x}}^\top\mathbf{h}\mathbf{R}^{-1}\nabla_{\mathbf{u}}\mathbf{h} & \nabla_{\mathbf{x}}^\top\mathbf{h}\mathbf{R}^{-1}\nabla_{\mathbf{x}}\mathbf{h} + (\mathbf{P}^{\mathbf{x}-})^{-1} \end{bmatrix}. \tag{14}$$

Remark 1 *(Improvements to the Gauss-Newton method).* While the basic Gauss-Newton iteration shown in (11) solves linear problems within only a single iteration and has fast local convergence on mildly nonlinear problems, it may suffer from divergence for some nonlinear problems. To improve the convergence performance, a damping coefficient $\alpha^{(i)} > 0$ can be added:

$$\begin{aligned} \hat{\boldsymbol{\xi}}_k^{(i+1)} &= \hat{\boldsymbol{\xi}}_k^{(i)} - \alpha^{(i)}\left[\nabla_{\boldsymbol{\xi}}^\top \mathbf{r}\left(\hat{\boldsymbol{\xi}}_k^{(i)}\right) \nabla_{\boldsymbol{\xi}} \mathbf{r}\left(\hat{\boldsymbol{\xi}}_k^{(i)}\right)\right]^{-1} \\ &\quad \cdot \nabla_{\boldsymbol{\xi}}^\top \mathbf{r}\left(\hat{\boldsymbol{\xi}}_k^{(i)}\right) \cdot \mathbf{r}\left(\hat{\boldsymbol{\xi}}_k^{(i)}\right). \end{aligned} \tag{15}$$

It can be proven that the damped Gauss-Newton iteration keeps moving to the critical point in a descent direction for sufficiently small $\alpha^{(i)} > 0$, thus guaranteeing its local convergence. In fact, it is usually globally convergent. Yet $\alpha^{(i)}$ must be selected with caution to ensure the viability of the damped Gauss-Newton, and a few methods have been proposed, e.g., the Armijo-Goldstein step length principle. A further improvement is to introduce a stabilizing term:

$$\begin{aligned} \hat{\boldsymbol{\xi}}_k^{(i+1)} &= \hat{\boldsymbol{\xi}}_k^{(i)} - \alpha^{(i)}\left[\nabla_{\boldsymbol{\xi}}^\top \mathbf{r}\left(\hat{\boldsymbol{\xi}}_k^{(i)}\right) \nabla_{\boldsymbol{\xi}} \mathbf{r}\left(\hat{\boldsymbol{\xi}}_k^{(i)}\right) + \delta^{(i)}\mathbf{D}^{(i)}\right]^{-1} \\ &\quad \cdot \nabla_{\boldsymbol{\xi}}^\top \mathbf{r}\left(\hat{\boldsymbol{\xi}}_k^{(i)}\right) \cdot \mathbf{r}\left(\hat{\boldsymbol{\xi}}_k^{(i)}\right), \end{aligned}$$

whereby the rank deficiency problem of $\left(\nabla_{\boldsymbol{\xi}}^{\top}\mathbf{r}\nabla_{\boldsymbol{\xi}}\mathbf{r}\right)$ that may appear in (11) and (15) can be avoided, given that $\delta^{(i)} > 0$ and $\mathbf{D}^{(i)}$ is a specified SPD matrix. This is known as the trust region method or Levenberg-Marquardt method. For more details about Gauss-Newton-type methods, the reader is referred to (Björck, 1996). •

3.4 The NL-SISE-wDF algorithm

Putting Sections 3.2 and 3.3 together yields the `NL-SISE-wDF algorithm`, formally described in Table 1, for nonlinear systems of the form (1).

Initialize: $k = 0$, $\hat{\boldsymbol{\xi}}_0^+ = \mathrm{E}(\boldsymbol{\xi}_0)$, $\mathbf{P}_0^{\boldsymbol{\xi}+} = p_0\mathbf{I}$, where p_0 is typically a large positive value
repeat
  $k \leftarrow k+1$
  Prediction:
  State prediction via (6)
  Computation of prediction error covariance via (7)
  Update:
  Initialize: $i = 0$, $\hat{\mathbf{u}}_k^{(0)} = 0$, $\hat{\mathbf{x}}_k^{+(0)} = \hat{\mathbf{x}}_k^-$
  while $i < i_{\max}$ $\left(\text{or } \left\|\hat{\boldsymbol{\xi}}_k^{+(i)} - \hat{\boldsymbol{\xi}}_k^{+(i-1)}\right\| > \epsilon > 0\right)$ **do**
    Gauss-Newton based joint input and state estimation via (11)
    $i \leftarrow i+1$
  end while
  $\hat{\mathbf{u}}_k = \hat{\mathbf{u}}_k^{(i_{\max})}$, $\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^{+(i_{\max})}$
  Computation of joint estimation error covariance via (12)-(14)
until no more measurements arrive

Table 1
`NL-SISE-wDF algorithm` (NL-SISE for Nonlinear Systems with Direct Feedthrough).

The `NL-SISE-wDF algorithm` introduces a novel Bayesian perspective to addressing the SISE problems, since a large body of recent work considers the problem from the viewpoint of filter design and optimal gain selection. Another advantage is that it can be applied to nonlinear systems in general form, instead of being restricted to some special ones.

The work (Gillijns and De Moor, 2007b) presents an optimal recursive SISE filter design approach for linear systems with direct feedthrough. The following result shows that the `NL-SISE-wDF algorithm` is a generalization of the Gillijns-De Moor's algorithm. Its proof is given in the Appendix.

Theorem 1 *The NL-SISE-wDF algorithm applied to a linear system with direct feedthrough yields the same input and state estimation as in (Gillijns and De Moor, 2007b).*

Remark 2 *(Generality of the Bayesian paradigm).* The algorithm in (Gillijns and De Moor, 2007b) is based on MVUE. Thus Theorem 1 partially reflects the fact that the Bayesian paradigm in (2)-(3) offers a general framework to solve SISE problems. This is further stressed by the fact that the Gillijns-De Moor's algorithm can be directly developed from the Bayesian paradigm in conjunction with MAP estimation, without using the numerical Gauss-Newton method, under the assumptions: (1) $\mathbf{x}_0 \sim \mathcal{N}(\hat{\mathbf{x}}_0^+, \mathbf{P}_0^{\mathbf{x}+})$; (2) $\{\mathbf{w}_k\}$ and $\{\mathbf{v}_k\}$ are zero-mean white Gaussian; (3) $\mathbf{x}_0$, $\{\mathbf{w}_k\}$ and $\{\mathbf{v}_k\}$ are independent of each other; (4) $\{\mathbf{u}_k\}$ is white Gaussian and independent of $\mathbf{x}_0$, $\{\mathbf{w}_k\}$ and $\{\mathbf{v}_k\}$. Compared to the assumptions made for the classical Kalman filter, (4) is the only additional requirement, which ensures that the state propagation and output measurement sequences are Gaussian distributed. •

4 Nonlinear Systems without Direct Feedthrough

Consider a nonlinear system described by equations of the form

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{f}(\mathbf{u}_k, \mathbf{x}_k) + \mathbf{w}_k, \\ \mathbf{y}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k, \end{cases} \tag{16}$$

where no direct feedthrough exists. In this situation, the input estimation is delayed by one time unit, considering that the first measurement containing information about $\mathbf{u}_{k-1}$ is $\mathbf{y}_k$. Therefore, it is needed to sequentially update $p(\mathbf{u}_{k-1}, \mathbf{x}_k|\mathcal{Y}_k)$. We impose the same assumption as (A1) to $\mathbf{u}_k$ for the system in (16), i.e., $\{\mathbf{u}_k\}$ is a white process independent of $\mathbf{x}_0$, $\{\mathbf{w}_k\}$ and $\{\mathbf{v}_k\}$. The Bayesian paradigm constructed accordingly is given by

$$p(\mathbf{u}_{k-1}, \mathbf{x}_k|\mathcal{Y}_k) \propto p(\mathbf{y}_k|\mathbf{x}_k) \int p(\mathbf{x}_k|\mathbf{u}_{k-1}, \mathbf{x}_{k-1}) \cdot p(\mathbf{x}_{k-1}|\mathcal{Y}_{k-1})\mathrm{d}\mathbf{x}_{k-1}. \tag{17}$$

It is also assumed that

$$p(\mathbf{u}_{k-1}, \mathbf{x}_k|\mathcal{Y}_k) \sim \mathcal{N}\left(\begin{bmatrix}\hat{\mathbf{u}}_{k-1} \\ \hat{\mathbf{x}}_k\end{bmatrix}, \begin{bmatrix}\mathbf{P}_{k-1}^{\mathbf{u}} & \mathbf{P}_{k-1,k}^{\mathbf{ux}} \\ (\mathbf{P}_{k-1,k}^{\mathbf{ux}})^\top & \mathbf{P}_k^{\mathbf{x}}\end{bmatrix}\right), \tag{18}$$

$$p(\mathbf{y}_k|\mathbf{x}_k) \sim \mathcal{N}\left(\mathbf{h}(\mathbf{x}_k), \mathbf{R}_k\right). \tag{19}$$

From (17), a MAP cost function can be defined as done previously in (9), that is,

$$\ell(\mathbf{u}_{k-1}, \mathbf{x}_k) = \delta + \mathbf{r}^\top(\mathbf{u}_{k-1}, \mathbf{x}_k)\mathbf{r}(\mathbf{u}_{k-1}, \mathbf{x}_k). \tag{20}$$

Here, δ is a constant and

$$\mathbf{r}(\mathbf{u}_{k-1}, \mathbf{x}_k) = \begin{bmatrix}\mathbf{R}_k^{-\frac{1}{2}}\boldsymbol{\rho}_k \\ \boldsymbol{\Pi}^{-\frac{1}{2}}(\mathbf{u}_{k-1})\boldsymbol{\zeta}_k\end{bmatrix},$$

where $\boldsymbol{\rho}_k = \mathbf{y}_k - \mathbf{h}(\mathbf{x}_k)$, $\boldsymbol{\zeta}_k = \mathbf{x}_k - \mathbf{f}(\mathbf{u}_{k-1}, \hat{\mathbf{x}}_{k-1})$, and $\boldsymbol{\Pi}(\mathbf{u}_k) = \nabla_{\mathbf{x}}\mathbf{f}(\mathbf{u}_k, \hat{\mathbf{x}}_k)\mathbf{P}_k^{\mathbf{x}}\nabla_{\mathbf{x}}^\top \mathbf{f}(\mathbf{u}_k, \hat{\mathbf{x}}_k) + \mathbf{Q}_k$. Using the Gauss-Newton method to compute the estimates $\hat{\mathbf{u}}_{k-1}$ and $\hat{\mathbf{x}}_k$,

$$\hat{\boldsymbol{\sigma}}_k^{(i+1)} = \hat{\boldsymbol{\sigma}}_k^{(i)} - \left[\nabla_{\boldsymbol{\sigma}}^\top \mathbf{r}\left(\hat{\boldsymbol{\sigma}}_k^{(i)}\right)\nabla_{\boldsymbol{\sigma}}\mathbf{r}\left(\hat{\boldsymbol{\sigma}}_k^{(i)}\right)\right]^{-1} \cdot \nabla_{\boldsymbol{\sigma}}^\top \mathbf{r}\left(\hat{\boldsymbol{\sigma}}_k^{(i)}\right)\cdot \mathbf{r}\left(\hat{\boldsymbol{\sigma}}_k^{(i)}\right), \tag{21}$$

where $\boldsymbol{\sigma}_k = \left[\mathbf{u}_{k-1}^\top \ \mathbf{x}_k^\top\right]^\top$ and $\nabla_{\boldsymbol{\sigma}}\mathbf{r} = \left[\nabla_{\mathbf{u}}\mathbf{r} \ \nabla_{\mathbf{x}}\mathbf{r}\right]$.

An optional initial condition is given by $\hat{\boldsymbol{\sigma}}_k^{(0)} = 0$, and the finally obtained $\hat{\boldsymbol{\sigma}}_k^{i_{\max}}$ will then be assigned to $\hat{\mathbf{u}}_{k-1}$ and $\hat{\mathbf{x}}_k$. A small damping coefficient and a stabilizing term can also be added to overcome possible divergence and matrix singularity problems. The associated estimation error covariance matrix can be computed by evaluating the Fisher information matrix at $\hat{\mathbf{u}}_{k-1}$ and $\hat{\mathbf{x}}_k$, i.e.,

$$\begin{bmatrix} \mathbf{P}_{k-1}^{\mathbf{u}} & \mathbf{P}_{k-1,k}^{\mathbf{ux}} \\ \left(\mathbf{P}_{k-1,k}^{\mathbf{ux}}\right)^\top & \mathbf{P}_k^{\mathbf{x}} \end{bmatrix} = \boldsymbol{\mathcal{I}}^{-1}(\hat{\mathbf{u}}_{k-1}, \hat{\mathbf{x}}_k), \tag{22}$$

where the definition of $\boldsymbol{\mathcal{I}}$ is identical to (13).

The l-th column of the gradient matrix of $\mathbf{r}$ with respect to (w.r.t.) $\mathbf{u}$ is given by,

$$\frac{\partial \mathbf{r}}{\partial \mathbf{u}_l} = \begin{bmatrix} \mathbf{0} \\ -\boldsymbol{\Pi}^{-\frac{1}{2}}\frac{\partial \mathbf{f}}{\partial \mathbf{u}_l} - \frac{1}{2}\boldsymbol{\Pi}^{\frac{1}{2}}\boldsymbol{\Pi}^{-1}\frac{\partial \boldsymbol{\Pi}}{\partial \mathbf{u}_l}\boldsymbol{\Pi}^{-1}\boldsymbol{\zeta} \end{bmatrix}.$$

The following relation is used here:

$$\frac{\partial \mathbf{X}^{\frac{1}{2}}}{\partial \tau} = -\frac{1}{2}\mathbf{X}^{\frac{1}{2}}\mathbf{X}^{-1}\frac{\partial \mathbf{X}}{\partial \tau}\mathbf{X}^{-1},$$

where $\mathbf{X}$ is a symmetric positive definite matrix dependent on τ (Spinello and Stilwell, 2010). The l-th column of the gradient matrix of $\mathbf{r}$ w.r.t. x is

$$\frac{\partial \mathbf{r}}{\partial \mathbf{x}_l} = \begin{bmatrix} -\mathbf{R}^{-\frac{1}{2}}\frac{\partial \mathbf{h}}{\partial \mathbf{x}_l} \\ \boldsymbol{\Pi}^{-\frac{1}{2}}\mathbf{e}_l \end{bmatrix}, \quad \nabla_{\mathbf{x}}\mathbf{r} = \begin{bmatrix} -\mathbf{R}^{-\frac{1}{2}}\nabla_{\mathbf{x}}\mathbf{h} \\ \boldsymbol{\Pi}^{-\frac{1}{2}} \end{bmatrix},$$

where $\mathbf{e}_l$ is the standard basis vector with a 1 in the l-th element and 0's elsewhere. The lj-th entries of $\nabla_{\mathbf{u}}^\top \mathbf{r}\nabla_{\mathbf{u}}\mathbf{r}$, $\nabla_{\mathbf{u}}^\top \mathbf{r}\nabla_{\mathbf{x}}\mathbf{r}$ and $\nabla_{\mathbf{x}}^\top \mathbf{r}\nabla_{\mathbf{x}}\mathbf{r}$ are expressed as, respectively,

$$\begin{aligned}
\frac{\partial \mathbf{r}^\top}{\partial \mathbf{u}_l}\frac{\partial \mathbf{r}}{\partial \mathbf{u}_j} &= \frac{\partial \mathbf{f}^\top}{\partial \mathbf{u}_l}\Pi^{-1}\frac{\partial \mathbf{f}}{\partial \mathbf{u}_j} + \frac{1}{2}\boldsymbol{\zeta}^\top\boldsymbol{\Pi}^{-1} \\
&\quad \cdot\left(\frac{\partial \boldsymbol{\Pi}}{\partial \mathbf{u}_l}\boldsymbol{\Pi}^{-1}\frac{\partial \mathbf{f}}{\partial \mathbf{u}_j} + \frac{\partial \boldsymbol{\Pi}}{\partial \mathbf{u}_j}\boldsymbol{\Pi}^{-1}\frac{\partial \mathbf{f}}{\partial \mathbf{u}_l}\right) \\
&\quad + \frac{1}{4}\boldsymbol{\zeta}^\top\boldsymbol{\Pi}^{-1}\frac{\partial \boldsymbol{\Pi}}{\partial \mathbf{u}_l}\boldsymbol{\Pi}^{-1}\frac{\partial \boldsymbol{\Pi}}{\partial \mathbf{u}_j}\Pi^{-1}\boldsymbol{\zeta}, \\
\frac{\partial \mathbf{r}^\top}{\partial \mathbf{u}_l}\frac{\partial \mathbf{r}}{\partial \mathbf{x}_j} &= -\frac{\partial \mathbf{f}^\top}{\partial \mathbf{u}_l}\boldsymbol{\Pi}^{-1}\mathbf{e}_j - \frac{1}{2}\boldsymbol{\zeta}^\top\boldsymbol{\Pi}^{-1}\frac{\partial \boldsymbol{\Pi}}{\partial \mathbf{u}_l}\boldsymbol{\Pi}^{-1}\mathbf{e}_j, \\
\frac{\partial \mathbf{r}^\top}{\partial \mathbf{x}_l}\frac{\partial \mathbf{r}}{\partial \mathbf{x}_j} &= \frac{\partial \mathbf{h}^\top}{\partial \mathbf{x}_l}\mathbf{R}^{-1}\frac{\partial \mathbf{h}}{\partial \mathbf{x}_j} + \mathbf{e}_l^\top\boldsymbol{\Pi}^{-1}\mathbf{e}_j, \\
\nabla_{\mathbf{x}}^\top \mathbf{r}\nabla_{\mathbf{x}}\mathbf{r} &= \nabla_{\mathbf{x}}^\top \mathbf{h}\mathbf{R}^{-1}\nabla_{\mathbf{x}}\mathbf{h} + \boldsymbol{\Pi}^{-1}.
\end{aligned}$$

Then we have

$$\begin{aligned}
\frac{\partial \ell}{\partial \mathbf{u}_l} &= \mathbf{r}^\top\frac{\partial \mathbf{r}}{\partial \mathbf{u}_l} = -\boldsymbol{\zeta}^\top\boldsymbol{\Pi}^{-1}\frac{\partial \mathbf{f}}{\partial \mathbf{u}_l} - \frac{1}{2}\boldsymbol{\zeta}^\top\boldsymbol{\Pi}^{-1}\frac{\partial \boldsymbol{\Pi}}{\partial \mathbf{u}_l}\boldsymbol{\Pi}^{-1}\boldsymbol{\zeta}, \\
\frac{\partial \ell}{\partial \mathbf{x}_l} &= \mathbf{r}^\top\frac{\partial \mathbf{r}}{\partial \mathbf{x}_l} = -\boldsymbol{\rho}^\top\mathbf{R}^{-1}\frac{\partial \mathbf{h}}{\partial \mathbf{x}_l} + \boldsymbol{\zeta}^\top\boldsymbol{\Pi}^{-1}\mathbf{e}_l, \\
\nabla_{\mathbf{x}}\ell &= -\boldsymbol{\rho}^\top\mathbf{R}^{-1}\nabla_{\mathbf{x}}\mathbf{h} + \boldsymbol{\zeta}^\top\boldsymbol{\Pi}^{-1}.
\end{aligned}$$

To compute the Fisher information matrix, $\mathrm{E}\left(\nabla_{\mathbf{u}}^\top \ell\nabla_{\mathbf{u}}\ell\right)$, $\mathrm{E}\left(\nabla_{\mathbf{u}}^\top \ell\nabla_{\mathbf{x}}\ell\right)$ and $\mathrm{E}\left(\nabla_{\mathbf{x}}^\top \ell\nabla_{\mathbf{x}}\ell\right)$ are needed. Their lj-th entries are

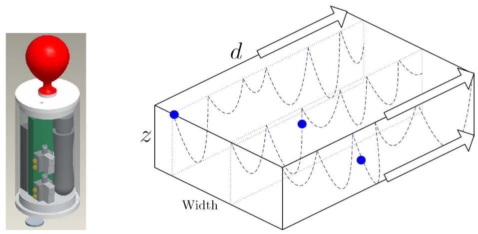
$$\begin{aligned}
\mathrm{E}\left(\frac{\partial \ell^\top}{\partial \mathbf{u}_l}\frac{\partial \ell}{\partial \mathbf{u}_j}\right) &= \frac{\partial \mathbf{f}^\top}{\partial \mathbf{u}_l}\boldsymbol{\Pi}^{-1}\frac{\partial \mathbf{f}}{\partial \mathbf{u}_j} + \frac{1}{4}\mathrm{tr}\left(\frac{\partial \boldsymbol{\Pi}}{\partial \mathbf{u}_l}\Pi^{-1}\frac{\partial \boldsymbol{\Pi}}{\partial \mathbf{u}_j}\boldsymbol{\Pi}^{-1}\right), \\
\mathrm{E}\left(\frac{\partial \ell^\top}{\partial \mathbf{u}_l}\frac{\partial \ell}{\partial \mathbf{x}_j}\right) &= -\frac{\partial \mathbf{f}^\top}{\partial \mathbf{u}_l}\boldsymbol{\Pi}^{-1}\mathbf{e}_j, \\
\mathrm{E}\left(\frac{\partial \ell^\top}{\partial \mathbf{x}_l}\frac{\partial \ell}{\partial \mathbf{x}_j}\right) &= \frac{\partial \mathbf{h}^\top}{\partial \mathbf{x}_l}\boldsymbol{\Pi}^{-1}\frac{\partial \mathbf{h}}{\partial \mathbf{x}_j} + \mathbf{e}_l^\top\boldsymbol{\Pi}^{-1}\mathbf{e}_j, \\
\mathrm{E}\left(\nabla_{\mathbf{x}}^\top \ell\nabla_{\mathbf{x}}\ell\right) &= \nabla_{\mathbf{x}}^\top \mathbf{h}\mathbf{R}^{-1}\nabla_{\mathbf{x}}\ell + \boldsymbol{\Pi}^{-1}.
\end{aligned}$$

The above equations lead to the `NL-SISE-w/oDF algorithm`, summarized in Table 2.

Initialize: $k = 0$, $\hat{\boldsymbol{\sigma}}_0 = \mathrm{E}(\boldsymbol{\sigma}_0)$, $\mathbf{P}_0^{\boldsymbol{\sigma}} = p_0\mathbf{I}$, where p_0 is a large positive value
repeat
  $k \leftarrow k + 1$
  Initialize: $i = 0$, $\hat{\mathbf{u}}_k^{(0)} = 0$, $\hat{\mathbf{x}}_k^{(0)} = 0$
  while $i < i_{\max}$ $\left(\text{or } \left\|\hat{\boldsymbol{\sigma}}_k^{+(i)} - \hat{\boldsymbol{\sigma}}_k^{+(i-1)}\right\| > \epsilon > 0\right)$ **do**
    Gauss-Newton based joint input and state estimation via (21)
    $i \leftarrow i + 1$
  end while
  $\hat{\mathbf{u}}_k = \hat{\mathbf{u}}_k^{(i_{\max})}$, $\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^{(i_{\max})}$
  Computation of joint estimation error covariance via (22)
until no more measurements arrive

Table 2
`NL-SISE-w/oDF algorithm` (NL-SISE for Nonlinear Systems without Direct Feedthrough).

In (Gillijns and De Moor, 2007a), a sequential SISE algorithm is proposed for linear systems without direct feedthrough. The derivation is also based on filter design and MVUE as in (Gillijns and De Moor, 2007b). In fact, the `NL-SISE-w/oDF algorithm` reduces to (Gillijns and De Moor, 2007a) in the linear case, as we show



(a) (b)

Fig. 1. (a) Schematic view of the buoyancy-controlled drogue; (c) scenario for estimation of a cubic unidirectional flow domain using the drogues (solid circles: drogues; dashed lines: trajectory of a drogue).

next. The proof is similar to that of Theorem 1 and is therefore omitted here.

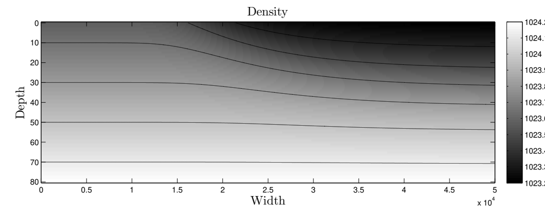
Theorem 2 *If the* `NL-SISE-w/oDF algorithm` *is applied to a linear system without direct feedthrough, it yields the same input and state estimation as in (Gillijns and De Moor, 2007a).*

It should be pointed out that the optimal state estimation gain matrix in (Gillijns and De Moor, 2007a) cannot be determined uniquely. One among all possible options is given by Eqn. (20) therein, which matches the state estimation gain that is obtained with the `NL-SISE-w/oDF algorithm`.

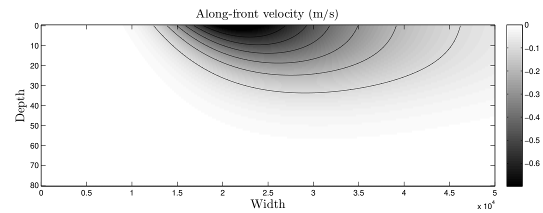
Remark 3 *(Generality of the Bayesian paradigm – continued).* Gillijns and De Moor (2007a) give the same state update as (Kitanidis, 1987; Darouach and Zasadzinski, 1997) and the same input update as (Hsieh, 2000). Theorem 2 indicates that these methods can be regarded as special cases of the `NL-SISE-w/oDF algorithm`. By analogy, if the conditions (1)-(4) proposed in Remark 2 are also valid for the system in (16), the algorithm in (Gillijns and De Moor, 2007a) can be directly derived from the Bayesian paradigm by MAP estimation. •

5 Application to Flow Field Estimation with Buoyancy-Controlled Drogues

The study of flow fields is a fundamental problem in oceanography. Flows play a key role in phenomena such as the transportation of nutrients, the motion of biological species in their early life, and the diffusion of contaminants and algal blooms. To reconstruct flow fields, we consider a group of buoyancy-controlled drogues (Colgan, 2006; Han et al., 2010) capable of arbitrary vertical migration behaviors in the ocean, see Fig. 1(a). While under water, the drogues can store a time record of depth, acceleration, and other relevant oceanographic quantities such as temperature and salinity. When they rise to surface, the information is transmitted to a central server for analysis and assimilation. Although it is not a technical requirement, we consider a cubic unidirectional flow domain, in which the flow is along a specific direction through the entire cross section with a continuous velocity and in parallel streamlines, as shown in Fig. 1(b).



(a)



(b)

Fig. 2. Flow field resulting from density gradients: (a) depth–time cross section of the density profile; (b) depth-time cross section of the along-front velocity field.

5.1 Drogue dynamics

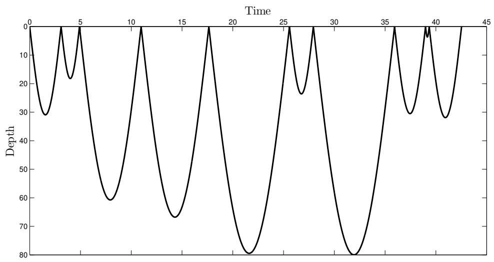
For a drogue, the flow velocity $v_d(z)$ along the direction of its displacement d is time-stationary and dependent only on the drogue depth z. The dynamics of the drogue within the flow field is described in (Booth, 1981):

$$m\ddot{d}(t) = c \cdot \left| v_d(z(t)) - \dot{d}(t) \right| \cdot \left(v_d(z(t)) - \dot{d}(t) \right), \quad (23)$$

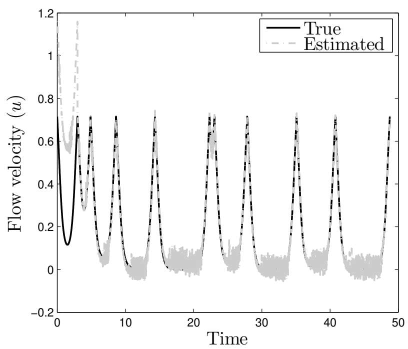
where m is the constant rigid mass and c is the drag parameter that quantifies the drag or resistance exercised on the drogue in the flow field. The motion of the drogue is characterized by an irregularly submerging/surfacing pattern — it submerges and moves underwater for a certain duration, then surfaces, and repeats the process again. No matter whether it is underwater or on the surface, the depth z and acceleration $\ddot{d}$ are measurable; however, the position d can only be measured when it is at surface.

From (23), we define two state variables $x_1 := d$ and $x_2 := \dot{d}$. Further, $v_d(z,t)$ can be viewed as the unknown external input into the drogue dynamics, naturally implying the definition of $u := v_d(z)$. Via finite-difference discretization with sampling period of T, the following state-space model is obtained to describe the dynamics of the drogue:

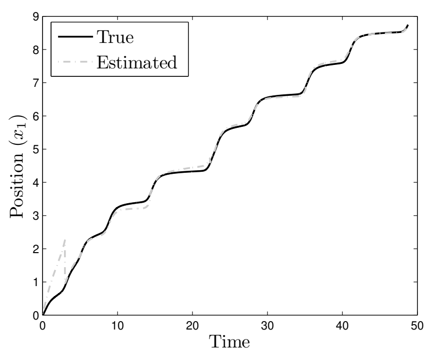
$$\Sigma : \begin{cases} \mathbf{x}_{k+1} = \mathbf{f}(u_k, \mathbf{x}_k) + \mathbf{w}_k, \\ \mathbf{y}_k = \mathbf{h}(u_k, \mathbf{x}_k) + \mathbf{v}_k, \end{cases} \quad (24)$$



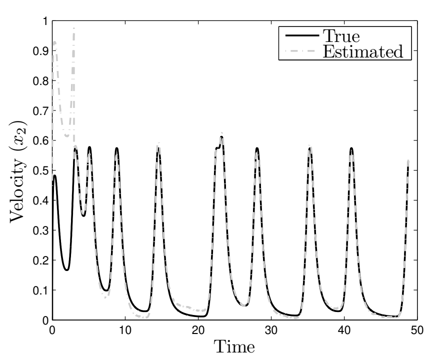
(a)



(b)

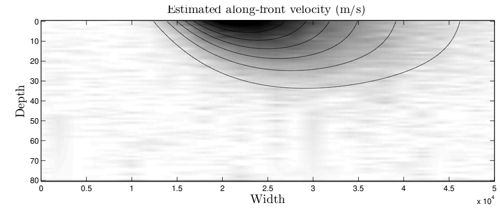


(c)

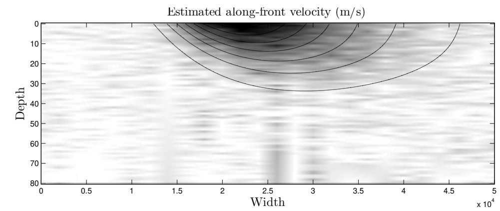


(d)

Fig. 3. (a) depth profile; (b) u vs. $\hat{u}$; (c) x_1 vs. $\hat{x}_1$; (d) x_2 vs. $\hat{x}_2$.



(a)



(b)

Fig. 4. Flow field estimation: (a)-(b) estimated at noise levels of $\mathbf{R} = 0.001$, 0.01, respectively.

Here,

$$\mathbf{f}(u_k, \mathbf{x}_k) = \begin{bmatrix} x_{1,k} + T \cdot x_{2,k} \\ x_{2,k} + T \cdot \frac{c}{m} \cdot |u_k - x_{2,k}| \cdot (u_k - x_{2,k}) \end{bmatrix},$$

$$\mathbf{h}(u_k, \mathbf{x}_k) = \begin{cases} \varphi_k(u_k, \mathbf{x}_k) & \text{when underwater,} \\ \begin{bmatrix} \varphi_k(u_k, \mathbf{x}_k) \\ \phi(u_k, \mathbf{x}_k) \end{bmatrix} & \text{when at surface,} \end{cases}$$

with $\varphi_k(u_k, \mathbf{x}_k) = \frac{c}{m} \cdot |u_k - x_{2,k}| \cdot (u_k - x_{2,k})$ and $\phi_k(u_k, \mathbf{x}_k) = x_{1,k}$.

Note that independent white Gaussian noises $\mathbf{w}$ and $\mathbf{v}$ are added in (24). This is a nonlinear model with unknown input and direct feedthrough. The `NL-SISE-wDF algorithm` is applicable to acquire the information estimates of not only the velocities of the flow field (unknown input variable) but also the trajectory and velocities of the drogue (state variables).

5.2 Numerical Simulation

The cross-sectional view of the considered cubic flow domain is shown in Fig. 2. It has dimensions of $(0, 5.0 \times 10^4)\text{m} \times (0, 80)\text{m}$. Fig. 2(a) illustrates the fronts, which are regions of strong horizontal density gradients in the ocean. They are usually sites of strong currents and eddy formation. Under the combined influence of the density gradient and Coriolis force, an along-front velocity field is generated, as given in Fig. 2(b), where the velocities are directed downward through the plane of the paper.

Let 51 drogues be deployed evenly along the width direction. The following parameters are used in the simulations presented next: mass of drogue $m = 1.5$Kg, drag parameter $c = 30\text{N} \cdot \text{s}^2/\text{m}^2$, sampling time $T = 0.01$s, process noise covariance $\mathbf{Q} = 0.001$ and measurement noise covariance $\mathbf{R} = 0.001$ (Han et al., 2010). Inserting the parameters into the state space model in (24), we then apply the `NL-SISE-wDF algorithm` to each drogue.

Let us first examine the simulation on the drogue released at the middle point (2.5×10^4m) which has the depth profile in Fig. 3(a) (other profile options are certainly allowed). Fig. 3(b)-3(d) makes comparisons between the actual and estimated values for u, x_1 and x_2, respectively. It is observed that the estimated values approach fast to the true ones for all the input and state variables. Identical results are obtained for the other drogues, and thus omitted here to save space.

Putting together the input estimation results of all drogues, we can reconstruct the entire flow field. The estimated flow field is shown in Fig. 4(a). From direct comparison with the true one in Fig. 2(b), it is seen that the estimation delicately captures the changes of the velocities at the central area, with an adequate accuracy achieved in general. To evaluate the `NL-SISE-wDF algorithm` further, we let $\mathbf{R} = 0.01$, respectively, to investigate the reconstruction performance. The results are illustrated in Fig. 4(b), from which we see that the overall estimation performance is still satisfactory, despite some deterioration as a result of the increase in noise level. The simulations show the efficacy of the proposed `NL-SISE-wDF algorithm` to provide reliable estimates for the challenging problem of flow field estimation using buoyancy-controlled drogues.

6 Conclusions

We have investigated the simultaneous input and state estimation problem for nonlinear systems with and without direct feedthrough. Building on a general Bayesian statistical paradigm for SISE of nonlinear systems, we have designed two algorithms that construct state and input estimators using Maximum a Posteriori optimization. The algorithms are sequential, consisting of prediction and update stages; furthermore, each update involves iterative searching as a result of using a Gauss-Newton method to minimize the nonlinear MAP cost function. We have shown that the proposed algorithms generalize existing methods for input and/or state estimation of linear systems. We have illustrated the soundness of the `NL-SISE-wDF algorithm` in an oceanographic flow field estimation problem where flow velocity profiles are estimated from the motion of a group of buoyancy-controlled drogues. Satisfactory performance is observed in the simulations.

Future work will include joint input and state observability analysis. This topic has been partially solved for linear systems but remains an open challenge for nonlinear systems. Another topic of interest is extension of other well-regarded linear SISE approaches to nonlinear systems.

Appendix – Proof of Theorem 1

Proof of Theorem 1: Consider applying Algorithm 1 to the linear discrete-time system in (Gillijns and De Moor, 2007b). In this case, the gradients are given by

$$\nabla_{\mathbf{x}}\mathbf{f} = \mathbf{A}_k, \quad \nabla_{\mathbf{u}}\mathbf{f} = \mathbf{G}_k, \quad \nabla_{\mathbf{x}}\mathbf{h} = \mathbf{C}_k, \quad \nabla_{\mathbf{u}}\mathbf{h} = \mathbf{H}_k.$$

Following (6)-(7), the state prediction is now given by

$$\hat{\mathbf{x}}_k^- = \mathbf{A}_{k-1}\hat{\mathbf{x}}_{k-1}^+ + \mathbf{G}_{k-1}\hat{\mathbf{u}}_{k-1},$$

and the prediction error covariance is

$$\mathbf{P}_k^{\mathbf{x}-} = \begin{bmatrix}\mathbf{A}_{k-1} & \mathbf{G}_{k-1}\end{bmatrix}\begin{bmatrix}\mathbf{P}_{k-1}^{\mathbf{x}+} & \mathbf{P}_{k-1}^{\mathbf{xu}} \\ \mathbf{P}_{k-1}^{\mathbf{ux}} & \mathbf{P}_{k-1}^{\mathbf{u}}\end{bmatrix}\begin{bmatrix}\mathbf{A}_{k-1}^\top \\ \mathbf{G}_{k-1}^\top\end{bmatrix} + \mathbf{Q}_{k-1}.$$

The prediction equations match the time update in (Gillijns and De Moor, 2007b).

For joint input and sate estimation, the Gauss-Newton method can be implemented in a single iteration for the linear system. Thus (11) is transformed into

$$\begin{bmatrix}\hat{\mathbf{u}}_k \\ \hat{\mathbf{x}}_k^+\end{bmatrix} = \begin{bmatrix}\mathbf{0} \\ \hat{\mathbf{x}}_k^-\end{bmatrix} - \begin{bmatrix}\mathbf{\Phi}_{11} & \mathbf{\Phi}_{12} \\ \mathbf{\Phi}_{21} & \mathbf{\Phi}_{22}\end{bmatrix}^{-1}\begin{bmatrix}\mathbf{\Omega}_1 \\ \mathbf{\Omega}_2\end{bmatrix}\left(\mathbf{y}_k - \mathbf{C}_k\hat{\mathbf{x}}_k^-\right),$$

where

$$\begin{aligned}
\mathbf{\Phi}_{11} &= \mathbf{H}_k^\top\mathbf{R}_k^{-1}\mathbf{H}_k, \quad \mathbf{\Phi}_{12} = \mathbf{\Phi}_{21}^\top = \mathbf{H}_k^\top\mathbf{R}_k^{-1}\mathbf{C}_k,\\
\mathbf{\Phi}_{21} &= \mathbf{\Phi}_{12}^\top, \quad \mathbf{\Phi}_{22} = \mathbf{C}_k^\top\mathbf{R}_k^{-1}\mathbf{C}_k + (\mathbf{P}_k^{\mathbf{x}-})^{-1},\\
\mathbf{\Omega}_1 &= -\mathbf{H}_k^\top\mathbf{R}_k^{-1}, \quad \mathbf{\Omega}_2 = -\mathbf{C}_k^\top\mathbf{R}_k^{-1}.
\end{aligned}$$

Let Δ be the inverse of Φ. Then we have

$$\begin{aligned}
\mathbf{\Delta}_{11} &= (\mathbf{\Phi}_{11} - \mathbf{\Phi}_{12}\mathbf{\Phi}_{22}^{-1}\mathbf{\Phi}_{21})^{-1} = (\mathbf{H}_k^\top\tilde{\mathbf{R}}_k^{-1}\mathbf{H}_k)^{-1},\\
\mathbf{\Delta}_{12} &= -\mathbf{\Delta}_{11}\mathbf{\Phi}_{12}\mathbf{\Phi}_{22}^{-1}\\
&= -(\mathbf{H}_k^\top\tilde{\mathbf{R}}_k^{-1}\mathbf{H}_k)^{-1}\mathbf{H}_k^\top\mathbf{R}_k^{-1}\mathbf{C}_k\mathbf{\Phi}_{22}^{-1},\\
\mathbf{\Delta}_{21} &= \mathbf{\Delta}_{12}^\top,\\
\mathbf{\Delta}_{22} &= \mathbf{\Phi}_{22}^{-1} + \mathbf{\Phi}_{22}^{-1}\mathbf{\Phi}_{21}\mathbf{\Delta}_{11}\mathbf{\Phi}_{12}\mathbf{\Phi}_{22}^{-1}\\
&= \mathbf{\Phi}_{22}^{-1} + \mathbf{\Phi}_{22}^{-1}\mathbf{C}_k^\top\mathbf{R}_k^{-1}\mathbf{H}_k(\mathbf{H}_k^\top\tilde{\mathbf{R}}_k^{-1}\mathbf{H}_k)^{-1}\\
&\quad \cdot \mathbf{H}_k^\top\mathbf{R}_k^{-1}\mathbf{C}_k\mathbf{\Phi}_{22}^{-1},
\end{aligned}$$

where $\tilde{\mathbf{R}}_k := \mathbf{C}\mathbf{P}_k^{\mathbf{x}-}\mathbf{C}_k^\top + \mathbf{R}_k$, as is defined in Gillijns and De Moor (2007b). The input estimation formula can be expressed as

$$\hat{\mathbf{u}}_k = \underbrace{-(\mathbf{\Delta}_{11}\mathbf{\Omega}_1 + \mathbf{\Delta}_{12}\mathbf{\Omega}_2)}_{\mathbf{M}_k}(\mathbf{y}_k - \mathbf{C}_k\hat{\mathbf{x}}_k^-)$$

$$= \mathbf{M}_k(\mathbf{y}_k - \mathbf{C}_k\hat{\mathbf{x}}_k^-).$$

It is verifiable that $\mathbf{M}_k = \left(\mathbf{H}_k^\top \tilde{\mathbf{R}}_k^{-1}\mathbf{H}_k\right)^{-1} \mathbf{H}_k^\top \tilde{\mathbf{R}}_k^{-1}$.

For state estimation, we have

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- \underbrace{-(\mathbf{\Delta}_{21}\mathbf{\Omega}_1 + \mathbf{\Delta}_{22}\mathbf{\Omega}_2)}_{\mathbf{L}_k}(\mathbf{y}_k - \mathbf{C}_k\hat{\mathbf{x}}_k^-).$$

It follows that

$$\mathbf{L}_k = \mathbf{\Phi}_{22}^{-1}\mathbf{C}_k^\top \mathbf{R}_k^{-1}(\mathbf{I} - \mathbf{H}_k\mathbf{M}_k) = \mathbf{K}_k(\mathbf{I} - \mathbf{H}_k\mathbf{M}_k),$$

where $\mathbf{K}_k = \mathbf{\Phi}_{22}^{-1}\mathbf{C}_k^\top \mathbf{R}_k^{-1} = \mathbf{P}_k^{\mathbf{x}-}\mathbf{C}_k^\top \tilde{\mathbf{R}}_k^{-1}$.

Hence,

$$\begin{aligned}\hat{\mathbf{x}}_k^+ &= \hat{\mathbf{x}}_k^- + \mathbf{L}_k(\mathbf{y}_k - \mathbf{C}_k\hat{\mathbf{x}}_k^-)\\ &= \hat{\mathbf{x}}_k^- + \mathbf{K}_k(\mathbf{I} - \mathbf{H}_k\mathbf{M}_k)(\mathbf{y}_k - \mathbf{C}_k\hat{\mathbf{x}}_k^-)\\ &= \hat{\mathbf{x}}_k^- + \mathbf{K}_k(\mathbf{y}_k - \mathbf{C}_k\hat{\mathbf{x}}_k^- - \mathbf{H}_k\hat{\mathbf{u}}_k).\end{aligned}$$

It is seen that the input and state estimation formulae are exactly those in (Gillijns and De Moor, 2007b).

Finally, we investigate the estimation error covariances. It is found in the linear situation, the Fisher information matrix $\boldsymbol{\mathcal{I}}$ is equivalent to $\mathbf{\Phi}$. Therefore,

$$\begin{aligned}\mathbf{P}_k^{\mathbf{u}} &= \mathbf{\Delta}_{11} = (\mathbf{H}_k^\top \tilde{\mathbf{R}}_k^{-1}\mathbf{H}_k)^{-1},\\ \mathbf{P}_k^{\mathbf{x}} &= \mathbf{\Delta}_{22} = \mathbf{P}_k^{\mathbf{x}-} - \mathbf{K}_k(\tilde{\mathbf{R}}_k - \mathbf{H}_k\mathbf{P}_k^{\mathbf{u}}\mathbf{H}_k^\top)\mathbf{K}_k^\top,\\ \mathbf{P}_k^{\mathbf{xu}} &= (\mathbf{P}_k^{\mathbf{ux}})^\top = \mathbf{\Delta}_{21} = -\mathbf{K}_k\mathbf{H}_k\mathbf{P}_k^{\mathbf{u}}.\end{aligned}$$

since $\mathbf{\Phi}_{22}^{-1} = \mathbf{P}_k^{\mathbf{x}-} - \mathbf{K}_k\tilde{\mathbf{R}}_k\mathbf{K}_k^\top$. The above covariance formulae are their exact match counterparts in (Gillijns and De Moor, 2007b). The proof is complete. □

References

Anderson, B. D. O., Moore, J. B., 1979. Optimal Filtering. Prentice-Hall, Englewood Cliffs.

Bell, B., Cathey, F., 1993. The iterated Kalman filter update as a Gauss-Newton method. IEEE Transactions on Automatic Control 38 (2), 294 –297.

Björck, Å., 1996. Numerical Methods for Least Squares Problems. SIAM, Philadelphia.

Booth, D. A., 1981. On the use of drogues for measuring subsurface ocean currents. Ocean Dynamics 34, 284–294.

Candy, J. V., 2009. Bayesian Signal Processing: Classical, Modern and Particle Filtering Methods. Wiley-Interscience, New York, NY, USA.

Cheng, Y., Ye, H., Wang, Y., Zhou, D., 2009. Unbiased minimum-variance state estimation for linear systems with unknown input. Automatica 45 (2), 485–491.

Colgan, C., 2006. Underwater laser shows. Explorations, Scripps Institution of Oceanography 12 (4), 20–27.

Corless, M., Tu, J., 1998. State and input estimation for a class of uncertain systems. Automatica 34 (6), 757–764.

Darouach, M., Zasadzinski, M., 1997. Unbiased minimum variance estimation for systems with unknown exogenous inputs. Automatica 33 (4), 717 – 719.

Floquet, T., Edwards, C., Spurgeon, S. K., 2007. On sliding mode observers for systems with unknown inputs. International Journal of Adaptive Control & Signal Processing 21 (8-9), 638–656.

Friedland, B., 1969. Treatment of bias in recursive filtering. IEEE Transactions on Automatic Control 14 (4), 359–367.

Gillijns, S., De Moor, B., 2007a. Unbiased minimum-variance input and state estimation for linear discrete-time systems. Automatica 43 (1), 111–116.

Gillijns, S., De Moor, B., 2007b. Unbiased minimum-variance input and state estimation for linear discrete-time systems with direct feedthrough. Automatica 43 (5), 934–937.

Gut, A., 2005. Probability: A Graduate Course. Springer, New York.

Ha, Q. P., Trinh, H., 2004. State and input simultaneous estimation for a class of nonlinear systems. Automatica 40 (10), 1779–1785.

Han, Y., De Callafon, R. A., Cortés, J., Jaffe, J., Jun. 2010. Dynamic modeling and pneumatic switching control of a submersible drogue. In: International Conference on Informatics in Control, Automation and Robotics. Vol. 2. Funchal, Madeira, Portugal, pp. 89–97.

Honerkamp, J., 1993. Stochastic Dynamical Systems: Concepts, Numerical Methods, Data Analysis. Wiley, New York, NY, USA.

Hsieh, C.-S., 2000. Robust two-stage Kalman filters for systems with unknown inputs. IEEE Transactions on Automatic Control 45 (12), 2374–2378.

Hsieh, C.-S., 2010. On the optimality of two-stage Kalman filtering for systems with unknown inputs. Asian Journal of Control 12 (4), 510–523.

Hsieh, C.-S., 2011. Optimal filtering for systems with unknown inputs via the descriptor Kalman filtering method. Automatica 47 (10), 2313–2318.

Kitanidis, P. K., 1987. Unbiased minimum-variance linear state estimation. Automatica 23 (6), 775–778.

Mendel, J., 1977. White-noise estimators for seismic data processing in oil exploration. IEEE Transactions on Automatic Control 22 (5), 694–706.

Mutambara, A. G. O., 1998. Decentralized Estimation and Control for Multisensor Systems. CRC Press, Inc., Boca Raton, FL, USA.

Pina, L., Botto, M. A., 2006. Simultaneous state and input estimation of hybrid systems with unknown inputs. Automatica 42 (5), 755–762.

Robinson, E., 1957. Predictive decomposition of seismic traces. Geophysics 29, 767–778.

Spinello, D., Stilwell, D., 2010. Nonlinear estimation with state-dependent Gaussian observation noise. IEEE Transactions on Automatic Control 55 (6), 1358–1366.

You, F.-Q., Wang, F.-L., Guan, S.-P., 2008. Hybrid estimation of state and input for linear discrete time-varying systems: A game theory approach. Acta Automatica Sinica 34 (6), 665–669.