Int. J. Appl. Math. Comput. Sci., 2020, Vol. 30, No. 2, 239–249
DOI: 10.34768/amcs-2020-0019

DECENTRALIZED AND DISTRIBUTED ACTIVE FAULT DIAGNOSIS: MULTIPLE MODEL ESTIMATION ALGORITHMS

ONDŘEJ STRAKA [a,*], IVO PUNČOCHÁŘ [a]

[a]Department of Cybernetics/European Centre of Excellence NTIS
University of West Bohemia
Univerzitní 8, 306 14 Pilsen, Czech Republic
e-mail: {straka30,ivop}@kky.zcu.cz

The paper focuses on active fault diagnosis (AFD) of large scale systems. The multiple model framework is considered and two architectures are treated: the decentralized and the distributed one. An essential part of the AFD algorithm is state estimation, which must be supplemented with a mechanism to achieve feasible implementation in the multiple model framework. In the paper, the generalized pseudo Bayes and interacting multiple model estimation algorithms are considered. They are reformulated for a given model of a large scale system. Performance of both AFD architectures is analyzed for different combinations of multiple model estimation algorithms using a numerical example.

Keywords: fault diagnosis, large scale systems, multiple models.

1. Introduction

Exceptional complexity and degree of integration of large scale systems (LSSs) result in their increased liability to faults. Since faults may potentially lead to failures with catastrophic consequences, it is vital to detect them reliably and as fast as possible by fault diagnosis (FD).

Two fundamental approaches to FD can be recognized that differ in their interaction with the monitored system. In the *passive* approach, the decisions generated by an FD system are based on passive observations of monitored system measurable quantities. This approach has been extensively studied by Isermann (2011) and Blanke *et al*. (2016) and has become widespread in the engineering practice (Gustafsson, 2009; Katipamula and Brambley, 2011). The *active* approach is characterized by the fact that, besides processing measurable quantities, the FD system generates an input signal to excite the monitored system. The purpose of the excitation is to obtain more information, which helps to detect faults that may be challenging to detect and isolate using passive FD.

In the last decade, the active FD (AFD) approach has become more popular (Ashari *et al*., 2012; Niemann and Poulsen, 2014; Punčochář *et al*., 2015; Raimondo *et al*., 2016). Since AFD for stochastic systems is a challenging problem, a partial simplification is achieved by considering only faults that fit into the multiple-model framework, where a monitored system can switch among several known modes of behavior (fault-free and faulty) at unknown time instants (Blackmore *et al*., 2008; Škach *et al*., 2016). The discussed class of faults is still large enough for the multiple-model framework to be deemed applicable in fault diagnosis problems of practical interest; see, e.g.,the works of Eide and Maybeck (1996), Zhang and Li (1998) or Hofbaur and Williams (2004) for particular examples.

Constraints on communication bandwidth and computational power are two main motivations for developing special methods for analysis and design of LSSs. A new AFD framework for stochastic LSSs was proposed by Punčochář and Straka (2019) and centralized, decentralized, and distributed architectures were discussed with the assumption that the continuous part of the state is available to the AFD system. Therefore, only the discrete part of the state had to be estimated. Since this assumption is rather unrealistic, the design of an AFD system for an LSS with noisy measurements was addressed by Straka and Punčochář (2019). This extension required a multiple model state estimation

*Corresponding author

algorithm capable of estimating both the discrete and the continuous part of the state. Since the optimal multiple model state estimation algorithm suffers from an exponential increase in computational and memory requirements, tractable approximate state estimators have been proposed in the literature (see, e.g., Watanabe and Tzafestas, 1993; Blom and Bar-Shalom, 1988).

The AFD system that was proposed by Straka and Punčochář (2019) utilized the generalized pseudo Bayes (GPB) algorithm of the second order (Watanabe and Tzafestas, 1993) (GPB2). Although several other multiple model estimation algorithms can be found in the literature, the most notable are the GPB algorithm of the first order (GPB1) and the interacting multiple model (IMM) algorithm (Blom and Bar-Shalom, 1988). They both present an attractive alternative to GPB2 because they are computationally cheaper. The goal of this paper is to design and analyze the performance of AFD systems with the decentralized and distributed architectures that utilize the GPB1 and IMM algorithms.

The paper is structured as follows. Section 2 provides specification of the LSS and the AFD problem. The solution to the latter in the decentralized and distributed architectures is summarized in Section 3. The GPB1 and IMM algorithms for the system specification considered are introduced in Section 4. The performance analysis is presented in Section 5, and Section 6 concludes the paper.

2. Problem formulation

This section formulates the AFD problem for an LSS. The block diagram of the AFD system is depicted in Fig. 1. The AFD nodes process observations made on the LSS, and generate decisions about faults and excitation that is fed back into the LSS.

2.1. LSS specification. Consider an LSS Σ described at a time instant $k \in \mathcal{T} \triangleq \{0, 1, 2, \ldots\}$ by the following discrete-time stochastic state-space model:

$$\Sigma: \quad \mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \boldsymbol{\mu}_k, \mathbf{u}_k) + \mathbf{F}(\boldsymbol{\mu}_k)\mathbf{w}_k, \tag{1a}$$

$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k, \boldsymbol{\mu}_k) + \mathbf{H}(\boldsymbol{\mu}_k)\mathbf{v}_k, \tag{1b}$$

where $\mathbf{x}_k \in \mathbb{R}^{D_x}$ is the continuous part of the state, $\boldsymbol{\mu}_k \in \mathcal{M}$ is the discrete part of the state, $\mathbf{u}_k \in \mathcal{U} \subseteq \mathbb{R}^{D_u}$ is the input, $\mathbf{w}_k \in \mathbb{R}^{D_x}$ is the state noise, $\mathbf{y}_k \in \mathbb{R}^{D_y}$ is the observation, and $\mathbf{v}_k$ is the measurement noise. The functions $\mathbf{f} : \mathbb{R}^{D_x} \times \mathcal{M} \times \mathcal{U} \mapsto \mathbb{R}^{D_x}$, $\mathbf{h} : \mathbb{R}^{D_x} \times \mathcal{M} \mapsto \mathbb{R}^{D_y}$, $\mathbf{F} : \mathcal{M} \mapsto \mathbb{R}^{D_x \times D_x}$, and $\mathbf{H} : \mathcal{M} \mapsto \mathbb{R}^{D_y \times D_y}$ are known. Each element of the discrete set $\mathcal{M}$ represents an index into a set of M possible models that can describe the behavior of the LSS Σ in fault-free and faulty conditions during one sampling period. The evolution of $\boldsymbol{\mu}_k$ is assumed to be described by a Markov chain with known transition probabilities,

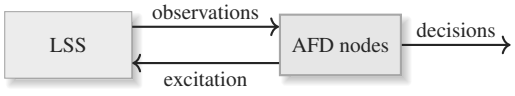

Fig. 1. Block diagram of the AFD system.

$$\Pr(\boldsymbol{\mu}_{k+1}|\boldsymbol{\mu}_k). \tag{2}$$

The Markov chain model can be useful even if the transition probabilities are not known precisely. In such a case, they still enable the user to specify whether a fault model is irreversible (the probability of leaving the fault model is zero), intermittent (the probability of returning to the fault-free model is nonzero), or progressive (the probability of switching to another fault model is nonzero). The initial conditions $\mathbf{x}_0$ and $\boldsymbol{\mu}_0$ are described by some known probability density function (PDF) $p(\mathbf{x}_0, \boldsymbol{\mu}_0)$. The state noise $\mathbf{w}_k$ and measurement noise $\mathbf{v}_k$ are described by sequences of known PDFs $p_{\mathbf{w}_k}$ and $p_{\mathbf{v}_k}$, respectively. Both noises are white, mutually independent, and independent of the initial conditions $\mathbf{x}_0$ and $\boldsymbol{\mu}_0$. Thus, for any $k \in \mathcal{T}$, we have[1]

$$p(\mathbf{x}_0, \boldsymbol{\mu}_0, \mathbf{w}_0^k, \mathbf{v}_0^k) = p(\mathbf{x}_0, \boldsymbol{\mu}_0) \prod_{i=0}^{k} p_{\mathbf{w}_i}(\mathbf{w}_i) p_{\mathbf{v}_i}(\mathbf{v}_i). \tag{3}$$

The variables $\mathbf{x}_k$ and $\boldsymbol{\mu}_k$ constitute the state of the LSS Σ,

$$\mathbf{s}_k \triangleq [\mathbf{x}_k^{\mathrm{T}}, \boldsymbol{\mu}_k^{\mathrm{T}}]^{\mathrm{T}} \in \mathcal{S} \triangleq \mathbb{R}^{D_x} \times \mathcal{M}. \tag{4}$$

Both the continuous part of the state $\mathbf{x}_k$ and the discrete part of the state $\boldsymbol{\mu}_k$ are unknown and can be observed indirectly through $\mathbf{y}_k$.

2.2. LSS decomposition. Although a centralized active fault detector could be designed for LSS Σ, this is not a computationally tractable approach. Therefore, a decomposition of the LSS and decentralized or distributed architectures of an active fault detector were considered by Punčochář and Straka (2019). The decomposition assumes that the LSS Σ consists of N subsystems that are weakly coupled through the state but have separate inputs and observations. Moreover, each subsystem has its own set of possible models that describe its behavior in fault-free and faulty conditions. Such a decomposition is

[1] A variable with the right subscript and superscript $\mathbf{x}_i^j \triangleq [\mathbf{x}_i^{\mathrm{T}}, \mathbf{x}_{i+1}^{\mathrm{T}}, \ldots, \mathbf{x}_j^{\mathrm{T}}]^{\mathrm{T}}$ with $j > i$ stands for the whole sequence of variables from the time instant i to j stacked into a column vector.

illustrated in Fig. 2 and the model (1) can be written as[2]

$$\Sigma: \underbrace{\begin{bmatrix} {}^1\mathbf{x}_{k+1} \\ \vdots \\ {}^N\mathbf{x}_{k+1} \end{bmatrix}}_{\mathbf{x}_{k+1}} = \underbrace{\begin{bmatrix} {}^1\mathbf{f}({}^1\mathbf{x}_k, {}^1\mu_k, {}^1\mathbf{u}_k) \\ \vdots \\ {}^N\mathbf{f}({}^N\mathbf{x}_k, {}^N\mu_k, {}^N\mathbf{u}_k) \end{bmatrix}}_{\mathbf{f}(\mathbf{x}_k, \boldsymbol{\mu}_k, \mathbf{u}_k)} + \underbrace{\begin{bmatrix} {}^1\mathbf{F}({}^1\mu_k) & & \\ & \ddots & \\ & & {}^N\mathbf{F}({}^N\mu_k) \end{bmatrix}}_{\mathbf{F}(\boldsymbol{\mu}_k)} \underbrace{\begin{bmatrix} {}^1\mathbf{w}_k \\ \vdots \\ {}^N\mathbf{w}_k \end{bmatrix}}_{\mathbf{w}_k}, \quad (5a)$$

$$\underbrace{\begin{bmatrix} {}^1\mathbf{y}_k \\ \vdots \\ {}^N\mathbf{y}_k \end{bmatrix}}_{\mathbf{y}_k} = \underbrace{\begin{bmatrix} {}^1\mathbf{h}({}^1\mathbf{x}_k, {}^1\mu_k) \\ \vdots \\ {}^N\mathbf{h}({}^N\mathbf{x}_k, {}^N\mu_k) \end{bmatrix}}_{\mathbf{h}(\mathbf{x}_k, \boldsymbol{\mu}_k)} + \underbrace{\begin{bmatrix} {}^1\mathbf{H}({}^1\mu_k) & & \\ & \ddots & \\ & & {}^N\mathbf{H}({}^N\mu_k) \end{bmatrix}}_{\mathbf{H}(\boldsymbol{\mu}_k)} \underbrace{\begin{bmatrix} {}^1\mathbf{v}_k \\ \vdots \\ {}^N\mathbf{v}_k \end{bmatrix}}_{\mathbf{v}_k}, \quad (5b)$$

where $\boldsymbol{\mu}_k \triangleq [{}^1\mu_k, \ldots, {}^N\mu_k]^{\mathrm{T}}$, ${}^n\mu_k \in {}^n\mathcal{M} \triangleq \{1, 2, \ldots, {}^nM\}$, $n \in \mathcal{N} \triangleq \{1, 2, \ldots, N\}$, $\mathcal{M} \triangleq {}^1\mathcal{M} \times \ldots \times {}^N\mathcal{M}$, $M = \prod_{n=1}^{N} {}^nM$, $\mathbf{u}_k \triangleq [{}^1\mathbf{u}_k^{\mathrm{T}}, \ldots, {}^N\mathbf{u}_k^{\mathrm{T}}]^{\mathrm{T}}$, ${}^n\mathbf{u}_k \in {}^n\mathcal{U} \in \mathbb{R}^{{}^nD_u}$, $D_u \triangleq \sum_{n=1}^{N} {}^nD_u, \mathcal{U} \triangleq {}^1\mathcal{U} \times \ldots \times {}^N\mathcal{U}$, ${}^n\mathbf{x}_k \in \mathbb{R}^{{}^nD_x}$, $D_x \triangleq \sum_{n=1}^{N} {}^nD_x$, ${}^n\mathbf{w}_k \in \mathbb{R}^{{}^nD_x}$ is described by $p_{{}^n\mathbf{w}_k}$, ${}^n\mathbf{y}_k \in \mathbb{R}^{{}^nD_y}$, $D_y \triangleq \sum_{n=1}^{N} {}^nD_y$, and ${}^n\mathbf{v}_k \in \mathbb{R}^{{}^nD_y}$ is described by $p_{{}^n\mathbf{v}_k}$. The LSS Σ given by (5) is assumed to satisfy the following independence conditions:

1. The initial states ${}^n\mathbf{x}_0$ and the initial model indices ${}^n\mu_0$ are independent and mutually independent,[3] i.e.,

$$p(\mathbf{x}_0, \boldsymbol{\mu}_0) = \prod_{n=1}^{N} p_{{}^n\mathbf{x}_0}({}^n\mathbf{x}_0) \Pr({}^n\mu_0).$$

2. The model indices ${}^n\mu_{k+1}$ are conditionally independent, i.e.,

$$\Pr(\boldsymbol{\mu}_{k+1} | \boldsymbol{\mu}_k) = \prod_{n=1}^{N} P({}^n\mu_{k+1} | \, {}^n\mu_k).$$

These independence conditions mean that the occurrence of a fault in a subsystem does not influence the probability of the occurrence of faults in other subsystems and also that the occurrences of the faults in the different subsystems, are independent.

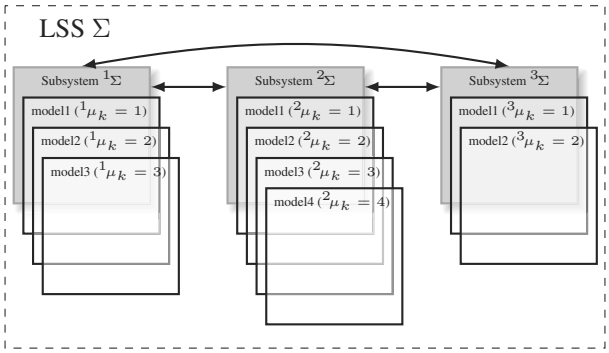

Fig. 2. Decomposition of an LSS into interconnected subsystems.

Thus, each subsystem ${}^n\Sigma$ is described by the following discrete-time stochastic model:

$${}^n\Sigma: \; {}^n\mathbf{x}_{k+1} = {}^n\mathbf{f}(\mathbf{x}_k, {}^n\mu_k, {}^n\mathbf{u}_k) + {}^n\mathbf{F}({}^n\mu_k) \, {}^n\mathbf{w}_k, \quad (6a)$$

$$\Pr({}^n\mu_{k+1} | \, {}^n\mu_k), \quad (6b)$$

$${}^n\mathbf{y}_k = {}^n\mathbf{h}({}^n\mathbf{x}_k, {}^n\mu_k) + {}^n\mathbf{H}({}^n\mu_k) \, {}^n\mathbf{v}_k, \quad (6c)$$

where the *local* state[4] ${}^n\mathbf{s}_k$ is defined as

$${}^n\mathbf{s}_k \triangleq [{}^n\mathbf{x}_k^{\mathrm{T}}, {}^n\mu_k]^{\mathrm{T}} \in {}^n\mathcal{S} \triangleq \mathbb{R}^{({}^nD_x)} \times {}^n\mathcal{M}, \quad (7)$$

variable ${}^n\mathbf{x}_k$ is a continuous part of the *local* state, and ${}^n\mu_k$ is a discrete part of the *local* state that represents an index into the set of possible models describing subsystem ${}^n\Sigma$. The set of models ${}^n\mathcal{M}$ includes one model representing the behavior of the subsystem in a *fault-free* condition, ${}^n\mu_k = 1$, and ${}^nM - 1$ models that represent the behavior of subsystem in a *faulty* condition, ${}^n\mu_k \in \{2, \ldots, {}^nM\}$. The faulty conditions are expressed by models with changed parameters (e.g., sensor gain). Note that the subsystems are coupled through $\mathbf{x}_k$ that appears in the dynamics (6a) of all subsystems.

2.3. AFD problem specification. AFD strives to design a function that transforms the complete available information to a decision about the faults (subsystem models) and to an input, whose role is to excite the system to improve the detection quality. The active fault detector can be described at a time instant $k \in \mathcal{T}$ as

$$\Delta: \quad \begin{bmatrix} \mathbf{d}_k \\ \mathbf{u}_k \end{bmatrix} = \boldsymbol{\rho}_k(\mathbf{I}_0^k) = \begin{bmatrix} \boldsymbol{\sigma}_k(\mathbf{I}_0^k) \\ \boldsymbol{\gamma}_k(\mathbf{I}_0^k) \end{bmatrix}, \quad (8)$$

where $\mathbf{I}_0^k \triangleq [(\mathbf{y}_0^k)^{\mathrm{T}}, (\mathbf{u}_0^{k-1})^{\mathrm{T}}]^{\mathrm{T}} \in \mathcal{I}^k$ denotes complete available information observed up to the time step $k \in \mathcal{T}$

[2] A variable or a function with the left superscript pertains to the corresponding subsystem (e.g., ${}^n\Sigma$), whereas a variable or a function without the left superscript relates to the whole LSS (e.g., Σ).

[3] With a slight abuse of terminology, the function $p(\mathbf{s}_k) = p(\mathbf{x}_k, \boldsymbol{\mu}_k)$ will be called a PDF although $\boldsymbol{\mu}_k$ is a discrete random variable. More formal notation would require using the cumulative distribution function instead of the PDF or the Dirac delta function.

[4] For convenience, the term *local* state is used to denote the part of the state $\mathbf{s}_k$ pertaining to a subsystem, even though it is not a state of the subsystem due to coupling.

with $\mathcal{I}^k \triangleq \mathbb{R}^{(k+1)D_y} \times \mathcal{U}^k$, $\mathcal{U}^k \triangleq \mathcal{U} \times \ldots \times \mathcal{U}$. The vector $\mathbf{d}_k \triangleq [{}^1d_k, {}^2d_k, \ldots, {}^Nd_k]^{\mathrm{T}} \in \mathcal{M}$ consists of the decisions ${}^nd_k \in {}^n\mathcal{M}$ about the model indices ${}^n\mu_k$, $\boldsymbol{\sigma}_k : \mathcal{I}^k \mapsto \mathcal{M}$ represents the fault detector at the time step k, and $\boldsymbol{\gamma}_k : \mathcal{I}^k \mapsto \mathcal{U}$ is a function describing the input signal generator.

The optimal active fault detector is determined such that the following additive discounted criterion is minimized:[5]

$$J = \lim_{F\to\infty} \mathrm{E}\left\{\sum_{k=0}^{F} \eta^k L^{\mathrm{d}}(\boldsymbol{\mu}_k, \mathbf{d}_k)\right\}, \tag{9}$$

where $\eta \in (0,1)$ is a chosen discount factor and $L^{\mathrm{d}} : \mathcal{M} \times \mathcal{M} \mapsto \mathbb{R}^+$ is a detection cost function that allows different costs to be assigned for selecting the vector of decisions $\mathbf{d}_k$ when the vector of model indices $\boldsymbol{\mu}_k$ is actually effective. This paper assumes that the costs are not related across the subsystems, and thus the following additive detection cost function is used:

$$L^{\mathrm{d}}(\boldsymbol{\mu}_k, \mathbf{d}_k) = \sum_{n=1}^{N} {}^nL^{\mathrm{d}}\left({}^n\mu_k, {}^nd_k\right), \tag{10}$$

where ${}^nL^{\mathrm{d}} : {}^n\mathcal{M} \times {}^n\mathcal{M} \mapsto \mathbb{R}^+$ is a function that assigns costs to selecting decision ${}^nd_k$ while the true model index is ${}^n\mu_k$. The cost function can represent true economic costs of the missed detection, false alarm and incorrect fault isolation. If these costs are not available in a particular problem, they can be regarded as tuning parameters that shape the behavior of the active fault detector (Punčochář and Šimandl, 2014).

3. AFD using decentralized and distributed architectures

This section describes the AFD problem reformulation for the centralized architecture, and then approximate solutions based on the decentralized and distributed architectures are presented.

3.1. Perfect state information model. Since the active fault detector has access only to inputs and outputs of the LSS but not to its state, the formulated problem belongs to the class of imperfect state information problems (Bertsekas, 2012). These are usually solved by being reformulated as perfect state information problems. The reformulation assumes that the active fault detector can be split into a given state estimator and an unknown mapping that transforms the state estimate to the input and decision. Then, the aim is to design only this mapping.

The optimal state estimate represented by the conditional PDF $p(\mathbf{s}_k|\mathbf{I}_0^k)$ (Bar-Shalom *et al.*, 2001) is usually approximated using a finite number of statistics that can be collected into an *information state* $\boldsymbol{\xi}_k \in \mathcal{G}$. The time evolution of the information state is then described by the following perfect state information model:

$$\boldsymbol{\xi}_{k+1} = \boldsymbol{\phi}\left(\boldsymbol{\xi}_k, \mathbf{u}_k, \mathbf{y}_{k+1}\right), \tag{11}$$

where $\boldsymbol{\phi} : \mathcal{G} \times \mathcal{U} \times \mathbb{R}^{D_y} \mapsto \mathcal{G}$ is a function that represents the composition of the state estimator and the LSS model Σ. The future output $\mathbf{y}_{k+1}$ is regarded in this model as a random disturbance described by the conditional pdf $p(\mathbf{y}_{k+1}|\boldsymbol{\xi}_k, \mathbf{u}_k)$.

3.2. AFD for the perfect state information model. Given the information state $\boldsymbol{\xi}_k$, it suffices to consider the active fault detector as a time invariant system that is described at a time step $k \in \mathcal{T}$ as

$$\Delta : \quad \begin{bmatrix}\mathbf{d}_k \\ \mathbf{u}_k\end{bmatrix} = \begin{bmatrix}\bar{\boldsymbol{\sigma}}(\boldsymbol{\xi}_k) \\ \bar{\boldsymbol{\gamma}}(\boldsymbol{\xi}_k)\end{bmatrix}, \tag{12}$$

where $\bar{\boldsymbol{\sigma}} : \mathcal{G} \mapsto \mathcal{M}$ and $\bar{\boldsymbol{\gamma}} : \mathcal{G} \mapsto \mathcal{U}$ are unknown functions. The detection cost function for the perfect state information model equivalent to L^{d} in (9) can be shown (Punčochář and Šimandl, 2014) to satisfy

$$\bar{L}^{\mathrm{d}}(\boldsymbol{\xi}_k, \mathbf{d}_k) = \sum_{n=1}^{N} \sum_{{}^n\mu_k} {}^nL^{\mathrm{d}}({}^n\mu_k, {}^nd_k) \Pr({}^n\mu_k|\mathbf{I}_0^k). \tag{13}$$

Having the reformulated problem specification, the active fault detector is determined by finding a function $V : \mathcal{G} \mapsto \mathbb{R}$ that solves the Bellman functional equation (Bertsekas, 2012; Vrabie *et al.*, 2013),

$$V(\boldsymbol{\xi}_k) = \min_{\mathbf{d}'\in\mathcal{M}} \bar{L}^{\mathrm{d}}(\boldsymbol{\xi}_k, \mathbf{d}') + \eta \min_{\mathbf{u}'\in\mathcal{U}} \mathrm{E}\left\{V(\boldsymbol{\xi}_{k+1})|\boldsymbol{\xi}_k, \mathbf{u}_k = \mathbf{u}'\right\}. \tag{14}$$

The Bellman function V can be computed off-line using the *a priori* known PDF $p(\mathbf{x}_{k+1}|\mathbf{x}_k, \boldsymbol{\mu}_k, \mathbf{u}_k)$, transition probabilities $\Pr(\boldsymbol{\mu}_{k+1}|\boldsymbol{\mu}_k)$, measurement PDF $p(\mathbf{y}_k|\mathbf{x}_k, \boldsymbol{\mu}_k)$, cost function $\bar{L}^{\mathrm{d}}$, and discount factor η. Then, the decisions and inputs can be determined on-line by solving much simpler optimization problems. Clearly, the Bellman function V does not need to be known to write the detector as

$$\mathbf{d}_k = \bar{\boldsymbol{\sigma}}^*(\boldsymbol{\xi}_k) = \arg\min_{\mathbf{d}'\in\mathcal{M}} \bar{L}^{\mathrm{d}}(\boldsymbol{\xi}_k, \mathbf{d}'). \tag{15}$$

Note that the detector is determined by the choice of the detection cost function (10). On the other hand, the input signal generator uses the Bellman function V to calculate the input $\mathbf{u}_k$ as

$$\mathbf{u}_k = \bar{\boldsymbol{\gamma}}^*(\boldsymbol{\xi}_k) = \arg\min_{\mathbf{u}'\in\mathcal{U}} \mathrm{E}\left\{V\left(\boldsymbol{\xi}_{k+1}\right)|\boldsymbol{\xi}_k, \mathbf{u}_k = \mathbf{u}'\right\}. \tag{16}$$

[5] The operator $\mathrm{E}\{\cdot\}$ denotes the expectation over all involved random variables.

Thus, the optimal input generator works in a feedback manner to improve the detection quality.

3.3. Approximate solution. The costs of computing the Bellman function V are extreme for LSSs as the dimension of the information state $\boldsymbol{\xi}_k$ can be high due to the dimension of the continuous part of the state $\mathbf{x}_k$ itself, the form of the sufficient statistics,[6] and the overall number of the models M. To reduce computational costs of both the off-line and on-line parts of the design, the AFD systems proposed by Punčochář and Straka (2019) as well as Straka and Punčochář (2019) considered decentralized and distributed architectures depicted in Figs. 3 and 4, respectively.

In the decentralized AFD architecture, each subsystem ${}^{n}\Sigma$ is monitored by an AFD node ${}^{n}\Delta$, which can access only the subsystem observations ${}^{n}\mathbf{y}_k$, knows only the models describing the subsystem behavior, and generates input ${}^{n}\mathbf{u}_k$ and decision ${}^{n}d_k$. To achieve such isolation of AFD nodes, approximate subsystem models are introduced by neglecting the coupling among the subsystems (Punčochář and Straka, 2019). For such a decoupled LSS, the AFD problem is solved for each subsystem separately. This means that each subsystem has its own information state of a smaller dimension than the LSS information state, and the Bellman function computation is tractable.

In the distributed AFD architecture, AFD nodes can communicate some information, which is used together with the full LSS model for computing the decisions. As a result, these decisions are more accurate than those computed within the decentralized architecture. Calculation of the Bellman function for the full LSS model in the distributed architecture would require the same effort as for the centralized architecture (Punčochář and Straka, 2019). Hence, for the sake of tractability, the Bellman function calculation is done in the same manner as in the decentralized architecture. This results in a lower quality of the excitation input in comparison with the centralized architecture. Thus, in both AFD architectures, the Bellman function is computed using approximate models that neglect the coupling among the subsystems.

4. Estimation algorithm for the perfect state information model

The perfect state information model (11) includes a state estimation algorithm that calculates the information state $\boldsymbol{\xi}_{k+1}$ based on $\boldsymbol{\xi}_k$, $\mathbf{u}_k$, and $\mathbf{y}_{k+1}$. The number of statistics collected in the information state $\boldsymbol{\xi}_k$ increases exponentially as time step k progresses forward if the optimal multiple model estimation algorithm is used. Hence, a computationally tractable approximate estimation algorithm is needed to prevent such an increase.

A GPB algorithm keeps track of the h-step history of the model sequences. Whenever the time step $k \geq h$, the GPB algorithm merges the model sequences $\boldsymbol{\mu}_{k-h}^{k}$ after the filtering step of the state estimation algorithm to obtain sufficient statistics for the model sequences $\boldsymbol{\mu}_{k-h+1}^{k}$. The GPB1 algorithm keeps a single estimate of the continuous part $\mathbf{x}_k$ and probabilities of the models. The GPB2 algorithm keeps an estimate of the continuous part for each model together with their probabilities (Bar-Shalom and Li, 1993). Blom and Bar-Shalom (1988) proposed the IMM algorithm as an alternative to GPB2 with a similar accuracy but lower computational costs.

Straka and Punčochář (2019) employed the GPB2 algorithm. This paper focuses on utilization of the GPB1 and IMM algorithms. These are described in the literature usually for the following form of the dynamics (1a):

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \boldsymbol{\mu}_{k+1}, \mathbf{u}_k) + \mathbf{F}(\boldsymbol{\mu}_{k+1})\mathbf{w}_k,$$

i.e., the dynamics depend on $\boldsymbol{\mu}_{k+1}$ related to time $k+1$ whereas the model considered in this paper depends on $\boldsymbol{\mu}_k$. Hence, both the GPB1 and IMM algorithms must be first reformulated, which will be done for simplicity for the whole LSS model. Then, their application in AFD with decentralized and distributed architectures will be discussed.

4.1. GPB1 algorithm. It is assumed that the filtering PDF $p(\mathbf{x}_k, \boldsymbol{\mu}_k|\mathbf{I}_0^k)$ can be factorized as

$$p(\mathbf{x}_k, \boldsymbol{\mu}_k|\mathbf{I}_0^k) = p(\mathbf{x}_k|\mathbf{I}_0^k)\Pr(\boldsymbol{\mu}_k|\mathbf{I}_0^k), \tag{17}$$

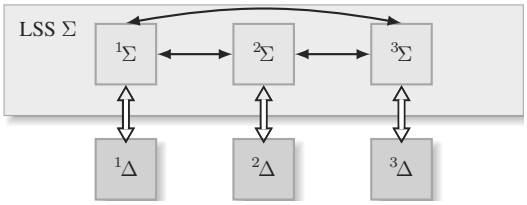

Fig. 3. Decentralized AFD system architecture.

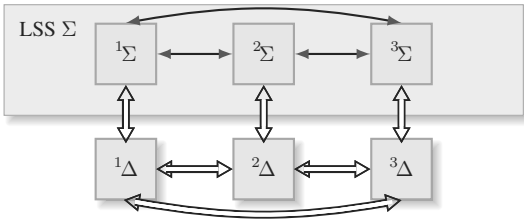

Fig. 4. Distributed AFD system architecture.

[6] Depending on the state estimation algorithm, the sufficient statistics may be represented by, e.g., the conditional mean and the covariance matrix of $\mathbf{x}_k$ in the case of the Kalman filter or weighted particles in the case of the particle filter.

and thus it is given by the single PDF $p(\mathbf{x}_k|\mathbf{I}_0^k)$ and M probabilities $\Pr(\boldsymbol{\mu}_k|\mathbf{I}_0^k)$.

Prediction step. The conditional independence is enforced and the predictive PDF is approximated as

$$p(\mathbf{x}_{k+1}, \boldsymbol{\mu}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k) \approx p(\mathbf{x}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k) \Pr(\boldsymbol{\mu}_{k+1}|\mathbf{I}_0^k), \quad (18)$$

where the probabilities $\Pr(\boldsymbol{\mu}_{k+1}|\mathbf{I}_0^k)$ are obtained as

$$\Pr(\boldsymbol{\mu}_{k+1}|\mathbf{I}_0^k) = \sum_{\boldsymbol{\mu}_k \in \mathcal{M}} \Pr(\boldsymbol{\mu}_k^{k+1}|\mathbf{I}_0^k), \quad (19)$$

and the PDF $p(\mathbf{x}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k)$ can be expressed as

$$p(\mathbf{x}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k) = \sum_{\boldsymbol{\mu}_k \in \mathcal{M}} \Pr(\boldsymbol{\mu}_k|\mathbf{I}_0^k) \times \int p(\mathbf{x}_k|\mathbf{I}_0^k) p(\mathbf{x}_{k+1}|\mathbf{x}_k, \boldsymbol{\mu}_k, \mathbf{u}_k) \mathrm{d}\mathbf{x}_k. \quad (20)$$

The mixture PDF $p(\mathbf{x}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k)$ is approximated by a simpler PDF $\hat{p}(\mathbf{x}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k)$ by the moment matching technique where only a small number of moments is retained (Bar-Shalom and Li, 1993). The probabilities $\Pr(\boldsymbol{\mu}_k^{k+1}|\mathbf{I}_0^k)$ appearing in (19) are given as

$$\Pr(\boldsymbol{\mu}_k^{k+1}|\mathbf{I}_0^k) = \Pr(\boldsymbol{\mu}_{k+1}|\boldsymbol{\mu}_k) \Pr(\boldsymbol{\mu}_k|\mathbf{I}_0^k). \quad (21)$$

The predictive PDF $p(\mathbf{s}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k)$ is then given by the PDF $\hat{p}(\mathbf{x}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k)$ and M probabilities $\Pr(\boldsymbol{\mu}_{k+1}|\mathbf{I}_0^k)$. Note that the transition PDF $p(\mathbf{x}_{k+1}|\mathbf{x}_k, \boldsymbol{\mu}_k, \mathbf{u}_k)$ is specified by (1a) and the PDF $p_{\mathbf{w}_k}$.

Filtering step. The filtering PDF $p(\mathbf{s}_{k+1}|\mathbf{I}_0^{k+1})$ is computed by the Bayesian relations in the form given by the PDF $p(\mathbf{x}_{k+1}|\mathbf{I}_0^{k+1})$ and M probabilities $\Pr(\boldsymbol{\mu}_{k+1}|\mathbf{I}_0^{k+1})$. The probabilities are computed as

$$\Pr(\boldsymbol{\mu}_{k+1}|\mathbf{I}_0^{k+1}) = \frac{p(\mathbf{y}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k, \boldsymbol{\mu}_{k+1}) \Pr(\boldsymbol{\mu}_{k+1}|\mathbf{I}_0^k)}{\sum_{\boldsymbol{\mu}_{k+1}} p(\mathbf{y}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k, \boldsymbol{\mu}_{k+1}) \Pr(\boldsymbol{\mu}_{k+1}|\mathbf{I}_0^k)}, \quad (22)$$

with

$$p(\mathbf{y}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k, \boldsymbol{\mu}_{k+1}) = \int p(\mathbf{y}_{k+1}|\mathbf{x}_{k+1}, \boldsymbol{\mu}_{k+1}) \times \hat{p}(\mathbf{x}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k) \, \mathrm{d}\mathbf{x}_{k+1}. \quad (23)$$

An approximation to the PDF $p(\mathbf{x}_{k+1}|\mathbf{I}_0^{k+1})$ is calculated by replacing the mixture

$$\sum_{\boldsymbol{\mu}_{k+1}} p(\mathbf{x}_{k+1}|\mathbf{I}_0^{k+1}, \boldsymbol{\mu}_{k+1}) \Pr(\boldsymbol{\mu}_{k+1}|\mathbf{I}_0^{k+1})$$

with a simpler PDF using the moment matching technique, where $p(\mathbf{x}_{k+1}|\mathbf{I}_0^{k+1}, \boldsymbol{\mu}_{k+1})$ is given by

$$p(\mathbf{x}_{k+1}|\mathbf{I}_0^{k+1}, \boldsymbol{\mu}_{k+1}) = \frac{p(\mathbf{y}_{k+1}|\mathbf{x}_{k+1}, \boldsymbol{\mu}_{k+1}) \hat{p}(\mathbf{x}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k)}{p(\mathbf{y}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k, \boldsymbol{\mu}_{k+1})}. \quad (24)$$

Note that the measurement PDF $p(\mathbf{y}_{k+1}|\mathbf{x}_{k+1}, \boldsymbol{\mu}_{k+1})$ is specified by (1b) and the PDF $p_{\mathbf{v}_k}$.

4.2. IMM algorithm. It is assumed that the filtering PDF is given by M PDFs $p(\mathbf{x}_k|\mathbf{I}_0^k, \boldsymbol{\mu}_k)$ and M probabilities $\Pr(\boldsymbol{\mu}_k|\mathbf{I}_0^k)$.

Prediction step. The predictive PDF can be written as

$$p(\mathbf{x}_{k+1}, \boldsymbol{\mu}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k) = \Pr(\boldsymbol{\mu}_{k+1}|\mathbf{I}_0^k) p(\mathbf{x}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k, \boldsymbol{\mu}_{k+1}), \quad (25)$$

where the probabilities $\Pr(\boldsymbol{\mu}_{k+1}|\mathbf{I}_0^k)$ are given by (19) and the PDF $p(\mathbf{x}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k, \boldsymbol{\mu}_{k+1})$ can be expressed as

$$p(\mathbf{x}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k, \boldsymbol{\mu}_{k+1}) = \sum_{\boldsymbol{\mu}_k \in \mathcal{M}} p(\mathbf{x}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k, \boldsymbol{\mu}_k) \Pr(\boldsymbol{\mu}_k|\boldsymbol{\mu}_{k+1}, \mathbf{I}_0^k). \quad (26)$$

The probabilities $\Pr(\boldsymbol{\mu}_k|\boldsymbol{\mu}_{k+1}, \mathbf{I}_0^k)$ are called mixing probabilities and they are given as

$$\Pr(\boldsymbol{\mu}_k|\boldsymbol{\mu}_{k+1}, \mathbf{I}_0^k) = \frac{\Pr(\boldsymbol{\mu}_k^{k+1}|\mathbf{I}_0^k)}{\Pr(\boldsymbol{\mu}_{k+1}|\mathbf{I}_0^k)}, \quad (27)$$

where $\Pr(\boldsymbol{\mu}_k^{k+1}|\mathbf{I}_0^k)$ is given by (21). The predictive PDF $p(\mathbf{x}_{k+1}, \boldsymbol{\mu}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k)$ is represented by M PDFs $p(\mathbf{x}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k, \boldsymbol{\mu}_k)$ and M^2 probabilities $\Pr(\boldsymbol{\mu}_k^{k+1}|\mathbf{I}_0^k)$.

Mixing. The mixture PDF $p(\mathbf{x}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k, \boldsymbol{\mu}_{k+1})$ given by (26) is approximated for each $\boldsymbol{\mu}_{k+1}$ using a simpler PDF $\hat{p}(\mathbf{x}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k, \boldsymbol{\mu}_{k+1})$ by the moment matching technique. After the mixing, the unknown state $\mathbf{s}_{k+1}$ is described by M PDFs $\hat{p}(\mathbf{x}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k, \boldsymbol{\mu}_{k+1})$ and M probabilities $\Pr(\boldsymbol{\mu}_{k+1}|\mathbf{I}_0^k)$.

Filtering step. Finally, the filtering PDF $p(\mathbf{s}_{k+1}|\mathbf{I}_0^{k+1})$ is computed by the Bayesian relations in the form given by M PDFs $p(\mathbf{x}_{k+1}|\mathbf{I}_0^{k+1}, \boldsymbol{\mu}_{k+1})$ and M probabilities $\Pr(\boldsymbol{\mu}_{k+1}|\mathbf{I}_0^{k+1})$. The probabilities are computed using (22) but the PDF $p(\mathbf{y}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k, \boldsymbol{\mu}_{k+1})$ is computed as

$$p(\mathbf{y}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k, \boldsymbol{\mu}_{k+1}) = \int p(\mathbf{y}_{k+1}|\mathbf{x}_{k+1}, \boldsymbol{\mu}_{k+1}) \times \hat{p}(\mathbf{x}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k, \boldsymbol{\mu}_{k+1}) \mathrm{d}\mathbf{x}_{k+1}. \quad (28)$$

The PDF $p(\mathbf{x}_{k+1}|\mathbf{I}_0^{k+1}, \boldsymbol{\mu}_{k+1})$ is given as

$$p(\mathbf{x}_{k+1}|\mathbf{I}_0^{k+1}, \boldsymbol{\mu}_{k+1}) = \frac{p(\mathbf{y}_{k+1}|\mathbf{x}_{k+1}, \boldsymbol{\mu}_{k+1})\hat{p}(\mathbf{x}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k, \boldsymbol{\mu}_{k+1})}{p(\mathbf{y}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k, \boldsymbol{\mu}_{k+1})}. \quad (29)$$

4.3. Comparison of estimation algorithms. The time and measurement updates of the GPB1, IMM, and GPB2 estimation algorithms are illustrated in Figs. 5, 6, and 7, respectively. The first column shows the number of PDFs and probabilities that are computed at individual steps of the estimation algorithms. In the prediction step, all estimation algorithms compute M PDFs and M^2 probabilities. In the filtering step, both the GPB1 and IMM algorithms compute M PDFs and M probabilities, while the GPB2 algorithm computes M^2 PDFs and M^2 probabilities. Since the mixing and merging have similar computational costs, the GPB2 algorithm is more costly compared to the IMM. The GPB1 algorithm is deemed to be the least costly. The computational costs of the algorithms also depend on the implementations of the individual steps and will be illustrated in the numerical example.

The estimation algorithms also differ in the dimension of the information state $\boldsymbol{\xi}_k$. While the filtering PDF is approximated in the IMM and GPB2 algorithms by M PDFs and M probabilities, the GPB1 algorithm uses only a single PDF and M probabilities to approximate the filtering PDF. Thus, for GPB1 the information state has a lower dimension as it contains sufficient statistics corresponding to a single PDF, while the information states for the IMM and GPB2 contain M instances of such statistics.

Finally, even if the optimal state estimator is used, AFD can be achieved only for a certain class of systems. The approximations taken by the GPB1, IMM or GPB2 algorithms may render this class even smaller, as can be illustrated for the GPB1 algorithm. Due to the enforced independence (18) in the prediction step of GPB1, the likelihood $p(\mathbf{y}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k, \boldsymbol{\mu}_{k+1})$ given in (23) differs for individual $\boldsymbol{\mu}_{k+1}$ only if the measurement equations (1b) depend on $\boldsymbol{\mu}_k$. If the measurement equations are the same for all $\boldsymbol{\mu}_k$, i.e., $p(\mathbf{y}_k|\mathbf{x}_k, \boldsymbol{\mu}_k) = p(\mathbf{y}_k|\mathbf{x}_k)$, then it follows from (23) that $p(\mathbf{y}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k, \boldsymbol{\mu}_{k+1}) = p(\mathbf{y}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k)$ and (22) reduces to

$$\Pr(\boldsymbol{\mu}_{k+1}|\mathbf{I}_0^{k+1}) = \Pr(\boldsymbol{\mu}_{k+1}|\mathbf{I}_0^k). \quad (30)$$

In such a case, not only is the input inconsequential, but also measurements are not taken into account and decisions are made only based on the model (2) and $\Pr(\boldsymbol{\mu}_0)$. Note that this particular issue does not occur for the IMM and GPB2 estimation algorithms.

4.4. GPB1 and IMM algorithms in AFD with decentralized and distributed architectures. An estimation algorithm is used in AFD twice. First, it is employed as part of the perfect state information model for the off-line design of the Bellman function (14). Second, it is used for on-line computation of the statistics included in the information state $\boldsymbol{\xi}_k$, which is subsequently employed to generate the decision (15) and the input (16).

Although the GPB1 and IMM algorithms were introduced in the previous section using the whole LSS model, an estimation algorithm actually operates on the subsystems of the LSS when decentralized and distributed architectures are considered. As explained in Section 3.3, approximate models of subsystems, which neglect the coupling among the subsystems, are employed during off-line design regardless of the architecture. Thus, an estimation algorithm produces information state ${}^n\boldsymbol{\xi}_k$ representing the local estimate $p({}^n\mathbf{s}_k|\mathbf{I}_0^k)$ using approximate models of (6) for each subsystem individually.

The use of an estimation algorithm varies during on-line estimation depending on the architecture in use. In the decentralized one, the estimation algorithm computes ${}^n\boldsymbol{\xi}_k$ similarly to off-line design. In the distributed architecture, it is, however, supplemented with a fusion step that is used to cope with the coupling

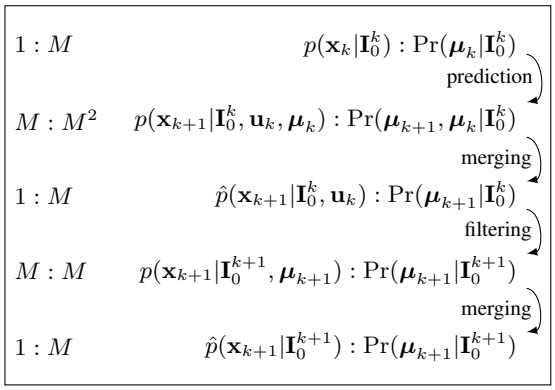

Fig. 5. Time and measurement updates of the GPB1 algorithm.

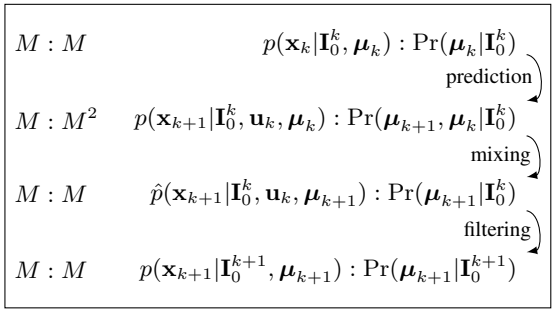

Fig. 6. Time and measurement updates of the IMM algorithm.

and leads to estimates of $\mathbf{x}_k$. If the coupling in the model dynamics (6a) is to be respected when the prediction PDF $p(\mathbf{s}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k)$ is calculated, AFD nodes must communicate their filtering estimates. The AFD node ${}^n\Delta$ sends its filtering estimate, i.e., the PDF $p({}^n\mathbf{s}_k|\mathbf{I}_0^{k-1}, {}^n\mathbf{y}_k, {}^n\mathbf{u}_{k-1})$, in the form of the information vector ${}^n\boldsymbol{\xi}_k$ to all other AFD nodes. The information vector is a function of past data $\mathbf{I}_0^{k-1}$ related to the whole LSS Σ and present data ${}^n\mathbf{y}_k$, and ${}^n\mathbf{u}_{k-1}$ related only to the subsystem ${}^n\Sigma$. Once the AFD node receives estimates of all other AFD nodes, it can fuse them to obtain $p(\mathbf{s}_k|\mathbf{I}_0^k)$.

The fusion must respect the fact that the estimates of ${}^n\mathbf{x}_k$ for $n \in \mathcal{N}$ given by the PDF $p({}^n\mathbf{x}_k|\mathbf{I}_0^{k-1}, {}^n\mathbf{y}_k, {}^n\mathbf{u}_{k-1})$ are clearly dependent, but the degree of dependence is unknown. If the sufficient statistics are represented by means and covariance matrices, the unknown dependency issue can be solved with the covariance intersection technique (Julier and Uhlmann, 1997). If the statistics are represented by weighted particles, the techniques proposed by, e.g., Tslil *et al*. (2018) or Ajgl and Šimandl (2011) can be used.

Usage of the IMM algorithm during one time step of the on-line part of the AFD algorithm is illustrated for two AFD nodes in Fig. 8 for the decentralized architecture and in Fig. 9 for the distributed architecture. One time step of the perfect state information model is indicated in the figures by the dashed arrow on the left side. Note that usage of the GPB1 and GPB2 algorithms is analogical.

5. Numerical illustration

To conserve space, the GPB1, IMM, and GPB2 estimation algorithms are compared by means of a simple numerical example only. The comparison will contain several active fault detectors involving different combinations of the multiple-model estimation algorithms used in the off-line and on-line stages of the AFD algorithm to analyze their impacts on the stages. The cases with the GPB1 algorithm in the off-line stage are of special interest as such combinations have low computational and memory costs resulting from the low dimension of the information

$$
\begin{aligned}
&M : M && p(\mathbf{x}_k|\mathbf{I}_0^k, \boldsymbol{\mu}_k) : \Pr(\boldsymbol{\mu}_k|\mathbf{I}_0^k) \\
&&& \text{prediction} \\
&M : M^2 && p(\mathbf{x}_{k+1}|\mathbf{I}_0^k, \mathbf{u}_k, \boldsymbol{\mu}_k) : \Pr(\boldsymbol{\mu}_k^{k+1}|\mathbf{I}_0^k) \\
&&& \text{filtering} \\
&M^2 : M^2 && p(\mathbf{x}_{k+1}|\mathbf{I}_0^{k+1}, \boldsymbol{\mu}_k^{k+1}) : \Pr(\boldsymbol{\mu}_k^{k+1}|\mathbf{I}_0^{k+1}) \\
&&& \text{merging} \\
&M : M && \hat{p}(\mathbf{x}_{k+1}|\mathbf{I}_0^{k+1}, \boldsymbol{\mu}_{k+1}) : \Pr(\boldsymbol{\mu}_{k+1}|\mathbf{I}_0^{k+1})
\end{aligned}
$$

Fig. 7. Time and measurement updates of the GPB2 algorithm.

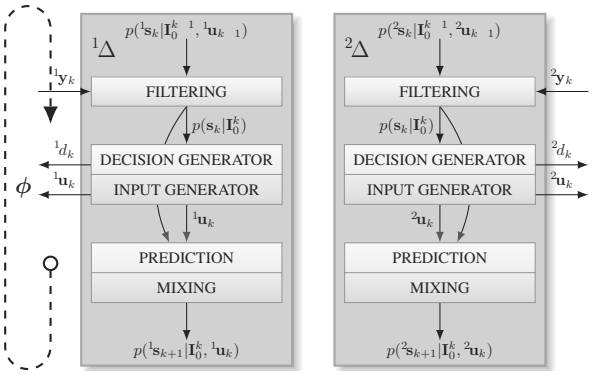

Fig. 8. Scheme of the decentralized AFD algorithm.

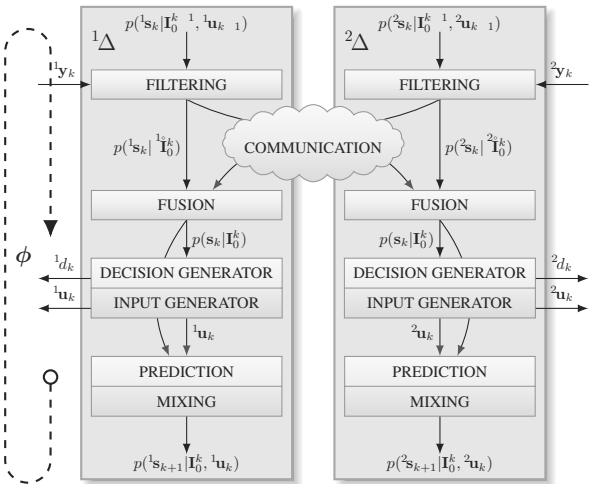

Fig. 9. Scheme of the distributed AFD algorithm.

state. Both decentralized and distributed architectures are evaluated.

5.1. System specification. Let us consider a system Σ that consists of two weakly coupled multiple-model linear subsystems

$$
\begin{aligned}
{}^1\Sigma : {}^1\mathbf{x}_{k+1} &= {}^1\mathbf{A}({}^1\mu_k)\mathbf{x}_k + {}^1B({}^1\mu_k)\,{}^1u_k + {}^1G({}^1\mu_k)\,{}^1w_k, \\
{}^1y_k &= {}^1C({}^1\mu_k)\,{}^1\mathbf{x}_k + {}^1H({}^1\mu_k)\,{}^1v_k, \\
{}^2\Sigma : {}^2\mathbf{x}_{k+1} &= {}^2\mathbf{A}({}^2\mu_k)\mathbf{x}_k + {}^2B({}^2\mu_k)\,{}^2u_k + {}^2G({}^2\mu_k)\,{}^2w_k, \\
{}^2y_k &= {}^2C({}^2\mu_k)\,{}^2\mathbf{x}_k + {}^2H({}^2\mu_k)\,{}^2v_k,
\end{aligned}
$$

where both subsystems have two models with

$$
\begin{aligned}
{}^1\mathbf{A}(1) &= [0.76\ 0.01], \quad {}^1B(1) = 0.12, \quad {}^1G(1) = \sqrt{0.003}, \\
&\qquad\qquad {}^1C(1) = 0.9, \quad {}^1H(1) = 0.01, \\
{}^1\mathbf{A}(2) &= [0.86\ 0.03], \quad {}^1B(2) = 0.14, \quad {}^1G(2) = \sqrt{0.003}, \\
&\qquad\qquad {}^1C(2) = 1, \quad {}^1H(2) = 0.01,
\end{aligned}
$$

$$^2\mathbf{A}(1) = [0.02\ 0.87], \quad ^2B(1) = 0.13, \quad ^2G(1) = \sqrt{0.002},$$
$$^2C(1) = 0.9, \quad ^2H(1) = 0.01,$$
$$^2\mathbf{A}(2) = [0.01\ 0.775], \quad ^2B(2) = 0.15, \quad ^2G(2) = \sqrt{0.002},$$
$$^2C(2) = 1, \quad ^2H(2) = 0.01.$$

The transition probabilities between models for each subsystem are given in Table 1, where Model 1 represents the fault-free behavior of the subsystem and Model 2 represents the faulty behavior. The faults of both subsystems are represented as changes in the static gain, time constant, and sensor gain. The state noises ^1w_k and ^2w_k have both standard Gaussian PDF $p_{^1w_k} = p_{^2w_k} = \mathcal{N}\{0,1\}$. The measurement noises ^1v_k and ^2v_k have both also standard Gaussian PDF $p_{^1v_k} = p_{^2v_k} = \mathcal{N}\{0,1\}$. The initial condition $\mathbf{x}_0$ has Gaussian PDF $\mathcal{N}\{\mathbf{0}, 0.01 \cdot \mathbf{I}\}$ and initial $\boldsymbol{\mu}_0$ has probability $\Pr(\boldsymbol{\mu}_0 = [1\ 1]^{\mathrm{T}}) = 1$, which means that each subsystem is fault-free in the beginning. The admissible inputs of subsystems are $^1\mathcal{U} = {}^2\mathcal{U} = \{-1, 0, 1\}$. The detection cost function $^nL^{\mathrm{d}}$ is the zero-one function $^nL^{\mathrm{d}}(^n\mu_k, {}^nd_k) = 1 - \delta_{^n\mu_k, {}^nd_k}$, where $\delta_{i,j}$ is the Kronecker delta. The discount factor is $\eta = 0.9$.

5.2. AFD algorithm specification. Since the individual models are Gaussian and linear, estimation algorithms employ the Kalman filter to compute the estimate of the continuous part of the state $^n\mathbf{x}_k$ during the prediction and filtering steps. Therefore, the information state $\boldsymbol{\xi}_k$ includes the conditional mean and covariance matrix as relevant statistics.

When off-line design was carried out in the decentralized architecture using the IMM or GPB2, the information state was five-dimensional for both AFD nodes, i.e., $^n\boldsymbol{\xi}_k \in \mathbb{R}^5$, $n = 1, 2$. It consisted of the scalar mean $\mathrm{E}[^n\mathbf{x}_k|\mathbf{I}_0^k]$ and variance $\mathrm{var}[^n\mathbf{x}_k|\mathbf{I}_0^k]$ for both models and the probability of the first model. The approximate Bellman function was designed using the value iteration algorithm, performed over the grids given as $[-1.5 : 0.1 : 1.5] \times [-1.5 : 0.1 : 1.5] \times [1.9 \cdot 10^{-4} : 5 \cdot 10^{-6} : 2 \cdot 10^{-4}] \times [1.9 \cdot 10^{-4} : 5 \cdot 10^{-6} : 2 \cdot 10^{-4}] \times [0 : 0.02 : 1]$ with 441 099 discrete states. When the design was carried out using GPB1, the information state was three-dimensional and consisted of a scalar mean and variance and the probability of the first model. The value iteration algorithm was performed over the grids given as $[-1.5 : 0.1 : 1.5] \times [1.9 \cdot 10^{-4} : 5 \cdot 10^{-6} : 2 \cdot 10^{-4}] \times [0 : 0.02 : 1]$, with 4 743 discrete states. To use the Bellman function computed by GPB1 with the IMM and GPB2 during on-line estimation, the dimension of the information state has to be reduced from five to three by merging the Gaussian mixture terms. On the other hand, to use the Bellman function computed by the IMM or GPB2 with GPB1 during on-line estimation, the dimension of the information state has to be enlarged by duplication of the sufficient statistics.

Table 1. Transition probabilities of the modes.

$^1\Sigma$			$^2\Sigma$		
	$^1\mu_k$			$^2\mu_k$	
$^1\mu_{k+1}$	1	2	$^2\mu_{k+1}$	1	2
1	0.99	0.02	1	0.99	0.01
2	0.01	0.98	2	0.01	0.99

5.3. Comparison. The performance of AFD algorithms was evaluated using 10^5 Monte Carlo (MC) simulations each run over the finite time horizon $F = 400$. The estimate $\hat{J}$ of the criterion obtained by the MC simulations, time requirements of a single run of the algorithm (i.e., on-line decision generation)[7] denoted as $T_{\text{on-line}}$, computational costs of all iterations of the value iteration algorithm used in the off-line design of the Bellman function denoted as $T_{\text{off-line}}$, and the number of iterations are given in Tables 2 and 3 for decentralized and distributed architectures, respectively. Note that the same values of $T_{\text{off-line}}$ apply to the distributed architecture because both architectures use the same Bellman functions.

For the decentralized architecture, it follows that the lowest criterion value, i.e., the most precise fault detection, is achieved by the detector with the GPB2 estimation algorithm used in both off-line and on-line stages. In fact, all combinations of the IMM and GPB2 perform similarly. For the distributed architecture, the best detection quality is achieved by the combination of the IMM algorithm in the off-line stage and the GPB2 algorithm in the on-line stage. Here, again all combinations of the IMM and GPB2 perform similarly.

The active fault detector involving combinations of the GPB1 algorithm in the off-line stage and the IMM or GPB2 algorithms in the on-line stage achieves a slightly lower detection quality than the detector with the best combination (by 15 % and 13 %, respectively), but the computational costs of the off-line stage involving the GPB1 algorithm are by two orders of magnitude lower than for the IMM or GPB2 algorithms. This also holds for memory requirements.

The computational costs of the on-line stage are in accordance with the expectations, i.e., the GPB1 algorithm is the cheapest one and the GPB2 algorithm is more demanding than the IMM one. The differences in computational costs between the IMM and GPB2 are rather small due to the fact that the subsystems are scalar. The difference would be more pronounced for higher

[7] All the numerical simulations were performed using the R2017b version of the Matlab® software running on a PC equipped with an Intel® CoreTM i7–4790 CPU (3.60 [GHz]).

Table 2. Performance of the decentralized AFD architecture for different multiple model estimation algorithms.

on-line \ off-line	$\hat{J}$ GPB1	IMM	GPB2	$T_{\text{on-line}}$
GPB1	1.525	1.592	1.572	0.122 s
IMM	0.939	0.822	0.817	0.141 s
GPB2	0.934	0.817	0.814	0.144 s
$T_{\text{off-line}}$	70 s	6 916 s	6 552 s	
#iterations	36	30	28	

Table 3. Performance of the distributed AFD architecture for different multiple model estimation algorithms.

on-line \ off-line	$\hat{J}$ GPB1	IMM	GPB2	$T_{\text{on-line}}$
GPB1	1.391	1.375	1.380	0.399 s
IMM	0.915	0.804	0.806	0.441 s
GPB2	0.909	0.801	0.802	0.499 s

dimensions of the continuous part of the local states. The results confirm that distributed architecture achieves a better detection quality than the decentralized one.

6. Conclusion

The paper dealt with active fault diagnosis of large-scale stochastic systems within the multiple model framework. State estimation is one of the key components of active fault diagnosis algorithms and for the framework considered its feasible implementation requires a technique to reduce the number of state estimate PDF components. In the literature, the GPB1, IMM, and GPB2 algorithms are often used for this purpose. Formerly, decentralized and distributed AFD algorithms were proposed that used the GPB2 algorithm.

This paper focused on using the GPB1 and IMM algorithms. They were reformulated to suit the system specification considered. Then, their performance was analyzed with a simple numerical example, where different multiple model estimation algorithms were used for off-line design and on-line estimation. The detection quality and computational costs were compared for both decentralized and distributed architectures. From the numerical example it follows that for the distributed architecture a good compromise between the detection quality and computational and memory requirements is GPB1 used for off-line design and the IMM or GPB2 used for on-line estimation. However, it should be noted that the GPB1 algorithm is suitable for the active fault detector only for systems with faults affecting the measurement equation.

Acknowledgment

This work was supported by the Czech Science Foundation (project no. GA18-08531S) and by the project LO1506 of the Czech Ministry of Education, Youth and Sports.

References

Ajgl, J. and Šimandl, M. (2011). Particle based probability density fusion with differential Shannon entropy criterion, *Proceedings of the 14th International Conference on Information Fusion, Chicago, IL, USA*, pp. 1–8.

Ashari, A.E., Nikoukhah, R. and Campbell, S.L. (2012). Active robust fault detection in closed-loop systems: Quadratic optimization approach, *IEEE Transactions on Automatic Control* **57**(10): 2532–2544.

Bar-Shalom, Y. and Li, X. (1993). *Estimation and Tracking: Principles, Techniques, and Software*, Artech House, Boston, MA.

Bar-Shalom, Y., Li, X.R. and Kirubarajan, T. (2001). *Estimation with Applications to Tracking and Navigation*, John Wiley & Sons, New York, NY.

Bertsekas, D.P. (2012). *Dynamic Programming and Optimal Control*, 4th Edn, Athena Scientific, Belmont, MA.

Blackmore, L., Rajamanoharan, S. and Williams, B.C. (2008). Active estimation for jump Markov linear systems, *IEEE Transactions on Automatic Control* **53**(10): 2223–2236.

Blanke, M., Kinnaert, M., Lunze, J. and Staroswiecki, M. (2016). *Diagnosis and Fault-tolerant Control*, 3rd Edn, Springer-Verlag, Berlin.

Blom, H.A.P. and Bar-Shalom, Y. (1988). The interacting multiple model algorithm for systems with Markovian switching coefficients, *IEEE Transactions on Automatic Control* **33**(8): 780–783.

Eide, P. and Maybeck, P.S. (1996). An MMAE failure detection system for the F-16, *IEEE Transactions on Aerospace and Electronic Systems* **32**(3): 1125–1136.

Gustafsson, F. (2009). Automotive safety systems, *IEEE Signal Processing Magazine* **26**(4): 32–47.

Hofbaur, M.W. and Williams, B.C. (2004). Hybrid estimation of complex systems, *IEEE Transactions on Systems, Man, and Cybernetics B: Cybernetics* **34**(5): 2178–2191.

Isermann, R. (2011). *Fault-Diagnosis Applications*, Springer, Heidelberg.

Julier, S.J. and Uhlmann, J.K. (1997). A non-divergent estimation algorithm in the presence of unknown correlations, *Proceedings of the 1997 American Control Conference, Albuquerque, NM, USA*, Vol. 4, pp. 2369–2373.

Katipamula, S. and Brambley, M.R. (2011). Methods for fault detection, diagnostics, and prognostics for building systems: A Review. Part II, *HVAC&R Research* **11**(2): 169–187.

Niemann, H. and Poulsen, N.K. (2014). Active fault detection in MIMO systems, *Proceedings of the 2014 American Control Conference, Portland, OR, USA*, pp. 1975–1980.

Punčochář and Straka, O. (2019). Non-centralized active fault diagnosis for stochastic systems, *2019 American Control Conference, Philadelphia, PA, USA*, pp. 5052–5057.

Punčochář, I. and Šimandl, M. (2014). On infinite horizon active fault diagnosis for a class of non-linear non-Gaussian systems, *International Journal of Applied Mathematics and Computer Science* **24**(4): 795–807, DOI: 10.2478/amcs-2014-0059.

Punčochář, I., Široký, J. and Šimandl, M. (2015). Constrained active fault detection and control, *IEEE Transactions on Automatic Control* **60**(1): 253–258.

Raimondo, D.M., Marseglia, G.R., Braatz, R.D. and Scott, J.K. (2016). Closed-loop input design for guaranteed fault diagnosis using set-valued observers, *Automatica* **74**: 107–117.

Škach, J., Punčochář, I. and Lewis, F.L. (2016). Optimal active fault diagnosis by temporal-difference learning, *Proceedings of the 55th IEEE Conference on Decision and Control, Las Vegas, NV, USA*, pp. 2146–2151.

Straka, O. and Punčochář (2019). Decentralized and distributed active fault diagnosis for stochastic systems with indirect observations, *22nd International Conference on Information Fusion, Ottawa, Canada*, pp. 1–8.

Tslil, O., Aharon, O. and Carmi, A. (2018). Distributed estimation using particles intersection, *Proceedings of the 21st International Conference on Information Fusion, Cambridge, UK*, pp. 1653–1660.

Vrabie, D., Vamvoudakis, K.G. and Lewis, F.L. (2013). *Optimal Adaptive Control and Differential Games by Reinforcement Learning Principles*, 1st Edn, Institution of Engineering and Technology, London.

Watanabe, K. and Tzafestas, S.G. (1993). Generalized pseudo-Bayes estimation and detection for abruptly changing systems, *Journal of Intelligent and Robotic Systems* **7**(1): 95–112.

Zhang, Y. and Li, X.-R. (1998). Detection and diagnosis of sensor and actuator failures using IMM estimator, *IEEE Transactions on Aerospace and Electronic Systems* **34**(4): 1293–1313.

Ondřej Straka received his MSc degree in cybernetics and control engineering and his PhD degree in cybernetics from the University of West Bohemia, Pilsen, Czech Republic, in 1998 and 2004, respectively. Since 2015, he has been an associate professor with the Department of Cybernetics, University of West Bohemia. He is the head of the Identification and Decision Making Research Group (IDM), NTIS—New Technologies for the Information Society. He has participated in a number of projects on fundamental research and applied research. He has been involved in development of several software frameworks for nonlinear state estimation and system identification. He has published over 70 journal and conference papers in renowned journals. His current research interests include local and global nonlinear state estimation methods, system identification, performance evaluation, and fault detection. Doctor Straka was a recipient of the Werner von Siemens Excellence Award in 2014 for most the important result in basic research.

Ivo Punčochář received his MSc degree in cybernetics and control engineering and his PhD degree in cybernetics from the University of West Bohemia, Pilsen, Czech Republic, in 2003 and 2008, respectively. Since 2014 he has been a senior researcher at the research center called New Technologies for the Information Society at the University of West Bohemia. He is also a member of the Identification and Decision Making Research Group established there. He has participated in several projects on fundamental and applied research dealing with fault detection, state estimation, and GNSS based safe positioning. He has published over 30 conference and journal papers. His primary research interests include active fault detection, optimal stochastic control and global navigation satellite systems. He was a member of the team that received the Werner von Siemens Excellence Award in 2014 for the most important outcome of basic research.

Received: 9 September 2019
Revised: 25 November 2019
Re-revised: 14 February 2020
Accepted: 6 April 2020