


Augmented Physics-Based Machine Learning for Navigation and Tracking

TALES IMBIRIBA , Member, IEEE
Northeastern University, Boston, MA USA

ONDŘEJ STRAKA , Member, IEEE

JINDŘICH DUNÍK , Senior Member, IEEE
University of West Bohemia, Plzen, Czech Republic

PAU CLOSAS , Senior Member, IEEE
Northeastern University, Boston, MA USA

This article presents a survey of the use of artificial intelligence/machine learning (AI/ML) techniques in navigation and tracking applications, with a focus on the dynamical models typically involved in corresponding state estimation problems. When physics-based models are either not available or not able to capture the complexity of the actual dynamics, recent works explored the use of deep learning models. This article tradeoffs both models and presents promising solutions in between, whereby physics-based models are augmented by data-driven components. The article uses two target tracking examples, both with synthetic and real data, to illustrate the various choices of the models and their parameters, highlighting their benefits and challenges. Finally, the article provides some conclusions and an outlook for future research in this relevant area.

Manuscript received 3 March 2023; revised 1 June 2023 and 18 October 2023; accepted 21 October 2023. Date of publication 31 October 2023; date of current version 11 June 2024.

DOI: No. 10.1109/TAES.2023.3328853

Refereeing of this contribution was handled by K. Yu.

This work was supported in part by the National Science Foundation under Grant ECCS-1845833 and Grant CCF-2326559.

Authors' addresses: Tales Imbiriba and Pau Closas are with the Institute for Experiential AI and the Department of Electrical and Computer Engineering, Northeastern University, 360 Huntington Avenue, Boston, MA 02115 USA, E-mail: (talesim@ece.neu.edu; closas@northeastern.edu); Ondrej Straka and Jindrich Dunik are with the Department of Cybernetics, the University of West Bohemia, 30614 Plzen, Czech Republic, E-mail: (straka30@kky.zcu.cz; dunikj@kky.zcu.cz). (*Corresponding author: Tales Imbiriba.*)

0018-9251 © 2023 IEEE

I. INTRODUCTION

Navigation is a method for determining or planning the position and course of an object (e.g., ship, aircraft, or ground vehicle) by means of geometry, astronomy, sensor data, etc. Common sensors include radio signal receivers, inertial measurement units, and electro-optical (EO) sensors. Reliable and accurate navigation is a key prerequisite for transportation efficiency. Tracking, similarly to navigation, aims to find accurate positioning information. Still, in contrast to navigation, the information is surveyed by a third party without using measurements from sensors onboard the tracked object.

The ever-increasing navigation safety and accuracy requirements force the continuous development of sensors, models, and methods for processing sensor information, estimating immeasurable quantities directly, and fusing the available navigation information. For a long time, the processing, estimation, and fusion methods have relied on purely physics-based models (PBMs) to characterize the sensor operation, the relation of the sensor measurements to quantities of interest, and the dynamics of these quantities using mathematical relations. Efforts to make use of the vast amount of information available recently for navigation and tracking (N&T) purposes lead to complicated methods and models with, sometimes, limited utility or even the unavailability of suitable models.

The growing capabilities of currently available computational resources [1] caused a new wave and explosion of N&T methods that use available data to improve their performance or to learn the correct functioning purely from data. The use of artificial intelligence/machine learning (AI/ML) algorithms has been reported in almost all types of N&T systems for land, marine, and air vehicles, either manned or unmanned [2], [3], [4], [5], [6]. AI/ML-enhanced N&T systems have been developed either as a proof of concept or with an intended application in an area without a certification need (e.g., supervised autonomous vehicles). However, AI/ML-enhanced navigation for safety-critical applications cannot be certified due to the lack of legislation and performance operational standards. For instance, those applications requiring design assurance level (DAL) A, DAL B with failure condition catastrophic or hazardous according to DO-178C [7], [8]; or those supporting or even allowing autonomous operation in critical phases of flight, such as landing using an instrument landing system. To address those limitations, the European Union Aviation Safety Agency (EASA), ensuring civil aviation safety, published an AI road map for its use in navigation applications [9]. The road map defines the areas where the AI can support the design of aviation systems,¹ proposes a time frame, and identifies the critical issues. Some of the frequently mentioned issues, which need to be urgently solved, are

¹Mentioned topics include, for example, navigation, air traffic management, maintenance, or safety risk management.

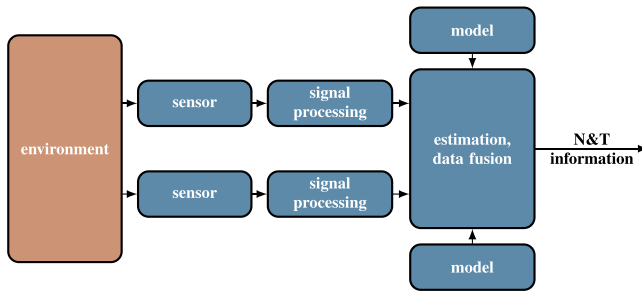


Fig. 1. Simplified block scheme of a N&T system.

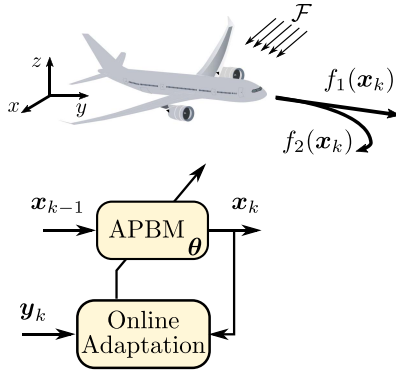


Fig. 2. State estimation often assumes dynamics for target states \mathbf{x}_k , which are sometimes inaccurately described by PBMs due to, for instance, unmodeled external forces \mathcal{F} or dynamics.

navigation information trustworthiness, predictability, explainability, and a lack of standardized methods for evaluation. Throughout the manuscript we adopt the taxonomy presented in [9] regarding the definitions of AI and ML. Note that the EASA road map is not the only or the first document on this topic. For example, in the 1990s, the Federal Aviation Administration published a technical note on AI for maintenance, monitoring, and control of airway facilities [10].

This article aims to provide a brief overview of AI/ML techniques applied in sensor information processing, system modeling, estimation, and data fusion, which are core parts of any N&T system (c.f. Fig. 1). State-space modeling possibly utilizing AI/ML techniques is the focal point of the article, leading either to completely data-driven models or to augmenting physics-based state-space models by efficiently leveraging and integrating data-driven models, as illustrated in Fig. 2 in a top-level view. In the picture, an object is tracked (e.g., an airplane) through standard state-estimation techniques [11], [12], which typically make assumptions on the object dynamics to estimate its state better. In practice, there are inaccuracies in the process. For instance, several dynamical models might be available; there might be unmodeled forces or effects (\mathcal{F}) either because these are unknown or too complex to model, or models might have time-varying fluctuations not easily captured through parsimonious models. The AI/ML-based modeling approaches are analyzed in detail and compared using two numerical tracking examples.

The rest of this article is organized as follows. Section II provides a brief overview of AI/ML techniques in sensor information processing, estimation, and data fusion, with pointers to surveyed literature. Section III, then, gives an overview of physics-based and AI/ML techniques used in modeling N&T systems and describes selected essential techniques in detail. Two illustrative target tracking examples are discussed in Section IV to provide additional insights on the use of AI modeling in N&T contexts. Section V delves into the challenges of employing AI in N&T and gives pointers for future research directions. Finally, Section VI concludes this article with final remarks and perspectives.

II. USE OF AI/ML TECHNIQUES IN NAVIGATION TECHNOLOGIES

This section provides a brief review of the use of AI/ML within N&T. Particularly, the literature has been surveyed to cover: sensors, where the most pervasive technologies in N&T are tackled; models, which are just briefly mentioned since that is deeply discussed in Section III, at the core of this article; and estimation/fusion components, which are relevant in any N&T system, as introduced in Fig. 1.

A. Sensors Information Processing

1) *Satellite-Based Global Position, Navigation, and Timing:* Global Navigation Satellite System (GNSS) is the de facto technology for position, navigation, and timing applications, when it is available [13], [14], [15], [16], [17]. GNSS encompasses global and regional positioning systems that, based on constellations of monitored satellites, enable receivers to compute range estimates to provide reliable position and timing absolute measures. Since their early development, GNSS receivers have been designed following physics-based principles, the same as the majority of communication systems [18]. Recently, the use of data-driven receiver design has been substantially explored, showing remarkable improvements under nonnominal conditions where physics-based methods are not accurately representing reality [6].

GNSS literature is flourishing with articles on using (deep) learning models and methods to address otherwise challenging problems; see [19] for a comprehensive survey. For instance, machine learning is becoming a popular approach for the detection, classification, and mitigation of GNSS interferences (such as jammers or spoofers). Ferre et al. [20] discussed the use of support vector machine (SVM) and convolutional neural network (CNN) models to detect and classify jamming signals, whereas a neural networks (NN) was used for evil waveforms detection in [21]. A distributed approach for jammer classification training was proposed in [22], whereas a distributed localization algorithm was presented in [23]. In [24], a multi-layer perceptron (MLP) network is built together with an adaptive notch filter for mitigation of narrowband interference. In the context of spoofing attacks, the authors in [25], [26], [27], [28], and [29] proposed a variety of NN-based supervised

ML approaches for spoofing detection (methods including CNN, MLP, recurrent neural networks (RNNs) based on the long-short term memory (LSTM) and C-support vector machine (C-SVM) with the principal component analysis). ML data-driven models have also been considered to counteract the effects of multipath. Primarily, most works use models to enhance the observables processed by the receiver chain, for instance, [30], [31], [32], [33], [34], [35], and [36]. Alternatively, some works investigate data-driven methods at the signal level; for instance, the authors in [37] and [38] proposed a GNSS signal acquisition method based on various deep learning models; Orabi et al. [39] used a deep neural network (DNN) to substitute the loop discriminator in multipath scenarios; and Li et al. [40] proposed a data-driven approach enhancing the signal correlation component, such that it can augment the optimal physics-based solution with multipath datasets. More articles can be found showing the popularity of this topic [41], [42], [43], [44], [45].

GNSS is well known for its use in remote sensing and Earth observation as a passive radar in some contexts. In remote sensing, the use of ML has also been growing; for instance, Brum et al. [46] proposed a NN for earthquake detection, and the authors in [47] and [48] proposed to track hurricanes and detect sea ice with a CNN. Various deep NNs were discussed in [49] for different environmental remote sensing problems. In ionospheric scintillations and tropospheric wet delay estimation, [50] and [51] explored the topic of detecting scintillation. The work in [52] explored the forecasting of low-latitude ionospheric conditions, whereas estimation of GNSS atmospheric-induced delays was performed using an MLP [53], CNN [54], and employing an ANN for tropospheric wet delay estimation [55]. The comprehensive survey [56] discusses the use of ML in ionospheric scintillation monitoring and estimation.

2) *Inertial Sensors*: Inertial sensors are represented by accelerometers (measuring the specific force) and gyroscopes (measuring the angular rate). Their measurements are integrated using the inertial navigation equations (INEs) in an inertial navigation system (INS) to calculate the position and attitude estimates [57], [58]. The INS is a traditional self-contained system with dual properties w.r.t. the GNSS, such as short-term stability and independence on externally broadcast signals.

Similarly to the GNSS, the algorithms of the AI/ML have also been used in the inertial area, namely in *design* of the inertial sensors and *processing* of inertial measurements [2], [59]. Whereas the former group aims at algorithms for, e.g., sensor life prediction, the latter, the more popular, group includes algorithms for inertial sensor calibration, error (or bias) estimation and compensation, and inertial data processing to get navigation information. For example, in [60], a calibration of the camera and inertial measurement unit is treated using reinforcement learning with the goal of learning which sequences of motion contain helpful information to obtain calibration parameters with sufficient accuracy. In [61], a strap-down gyroscope drift rate is modeled by the time series neural networks

of sigmoid type, and in, e.g., [62] and [63], the NN-based temperature drift modeling of the MEMS-type gyroscope and accelerometer is discussed. The AI/ML algorithms have also been used in the processing and postprocessing of inertial data as an alternative to purely geometric INEs. In the literature, two types of the NN-based INS can be found; AI/ML enhanced inertial navigation, where specific noise components are identified from data and modeled by the NN [64], or full-data driven AI/ML navigation, where the whole INEs mechanization is substituted with the NN [65]. Besides that, the AI/ML algorithms have been in navigation system alignment [66] and in the detection of certain features, which can improve navigation performance, such as the zero-velocity [or the zero-velocity potential update (ZUPT)] [67] or movement pattern [68]. In [69], it was recognized that successful application of AI/ML to motion sensing and localization training data-set and public data-set was introduced. Similarly to the inertial sensors, the magnetometer measurements can be corrected using the AI/ML methods [70], [71].

3) *Electro-Optical Sensors*: EO sensors, e.g., cameras, radars, and LiDARs, and the need to process a huge amount of produced wide range data have significantly contributed to or even drive the development of the AI/ML techniques for data interpretation, object detection, and classification. Indeed, almost any computer vision system takes advantage of AI/ML algorithms [72]. The literature on the EO measurement (pre)processing is vast and out of the scope of this article; however, a survey can be found, e.g., in [4], [73], and [74]. Besides the object detection, the computer vision systems can also provide estimates of the position or orientation of their platform or another sensed object w.r.t. prerecorded map or a detected horizon [75], [76]. In this case, the computer vision system can be seen as an alternative sensor to inertial or GNSS-based sensors. Alternative solutions consider the joint use of visual and inertial measurements, for instance as in [77].

4) *Maps and Environment Models*: Modern navigation systems often rely on prerecorded maps (e.g., terrain, magnetic, gravity, or celestial), features, or images in different spectra. These data-based references can improve navigation system performance or even allow navigation in adverse or harsh environments (spoofed or jammed GNSS, reduced visibility, etc.). Therefore, similarly, research interest has focused on improving or enhancing the data-based references using tools of AI/ML. For example, in [78] and [79], an automated terrain feature identification by deep learning is discussed, and in [80] deep learning for a mapping of magnetic field distribution to topologies of different complexities is introduced.

B. System Models

Obtaining a model of the dynamics of a system of interest may proceed from the application of the first principles if the model is reasonably complex. Alternatively, a system identification procedure processing measured input–output data is used to find the model structure and/or parameters.

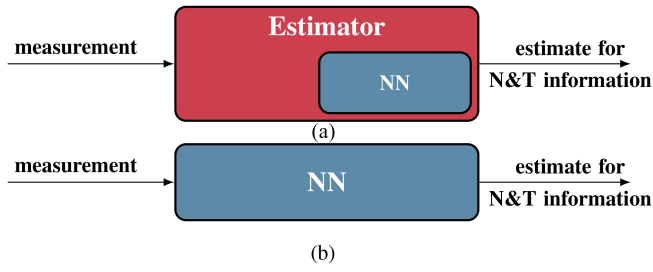


Fig. 3. Data-driven and hybrid-driven state estimation.
(a) Hybrid-driven state estimation. (b) Data-driven state estimation.

ML found its use in system identification more than two decades ago; see the excellent review focused on input–output models in [81] and [82]. N&T problems usually involve state-space models, for which identification by ML techniques is a recent topic; approaches utilizing ML in system modeling are addressed in detail in Section III.

C. Estimation and Data Fusion

ML is applied in state estimation mainly in the context of tracking. The dynamic of the tracked object is typically unknown, and the estimator makes use of its simple approximations, such as nearly constant velocity or acceleration models. Thus, data-driven approaches to state estimation utilizing NNs to improve the estimate quality are quite popular. The ML approaches can be classified into hybrid-driven (combining standard Bayesian estimation with NNs) and purely data-driven, utilizing end-to-end learning.

Some hybrid-driven approaches correct the estimates computed by standard model-based estimation algorithms. The algorithms in [83] and [84] use back-propagation NNs to correct position estimates. The algorithm proposed in [85] predicts nonlinear velocity and acceleration corrections to linear predicted states by an RNN, and the algorithm in [86] augments the error-state Kalman filter (KF) by a radial basis function NN to compensate for the lack of KF performance. These algorithms use NNs unaware of the system model and can be called loosely coupled hybrid-driven approaches, see Fig. 3(a).

In addition to providing estimate corrections, several hybrid-driven approaches directly estimate the state. In [87], two algorithms based on deep LSTM NNs [88] were proposed, which provide estimates either in two steps (time-update and measurement-update) or in a single step. Both algorithms are unaware of the system model, and the learning is based on estimation quality optimization. The algorithm in [89] uses an NN in the prediction step of the KF to provide not only the estimate but also the associated covariance matrix. In this case, the learning optimizes the negative log-likelihood of multivariate normal distribution.

Fully data-driven approaches to state estimation [see Fig. 3(b)] use plain data to learn mappings from observations to the states to avoid complications of the loosely coupled hybrid-driven methods. In [90], a large amount of data was simulated to achieve end-to-end learning. In [91], a new deep approach to Kalman filtering, which can be

learned directly in an end-to-end manner, was proposed, and in [92] all components of the process, i.e., data generator, sliding window, centralization strategy, and the learner have been proposed.

When several information sources, either at the signal level or the estimated level, are available, the prevalent way to obtain more consistent and accurate information is to employ a data fusion algorithm. ML automatically gaining deep relationships in data inputs may improve the overall performance of data fusion algorithms. Thus, the application of ML in data fusion has also become an endeavor of intensive research; see the reviews [3], [4], and [5]. We highlight that, when dealing with multiple sensing devices, it is paramount that proper transformations are performed, such that the measurements are represented in the same coordinate system. More information about this can be found in [93].

III. STATE-SPACE MODELS

As has been mentioned above, high-quality state-space models are the heart of many modern N&T systems as the state usually represents the N&T information, and the models describe its dynamics and its relation to the measurements obtained from the available sensors.

In N&T systems, we are typically concerned with models of the type

$$\dot{\mathbf{x}}_t = \tilde{\mathbf{f}}(\mathbf{x}_t, \mathbf{u}_t; \tilde{\boldsymbol{\lambda}}) + \mathbf{q}_t \quad (1)$$

$$\mathbf{y}_t = \mathbf{h}(\mathbf{x}_t, \mathbf{r}_t) \quad (2)$$

consisting of models of dynamics (1) and measurement (2). Here, $\mathbf{x}_t \in \mathbb{R}^{d_x}$ is the state vector, \mathbf{u}_t is a vector input, $\tilde{\mathbf{f}}$ and \mathbf{h} are the vector-valued state transition and measurement functions, respectively, and \mathbf{q}_t and \mathbf{r}_t are arbitrary zero-mean noise terms independent of \mathbf{x}_t . The measurement (2) describes the operation of the sensor, which provides some information about the state \mathbf{x}_t . In the sequel, we will concern with the dynamics (1) only as the measurement model is often given sufficiently accurate by the sensor manufacturer. In particular, we are interested in models where the transition function $\tilde{\mathbf{f}}$ is not fully representative of the true state dynamics due to either incapacity to explain time-varying dynamic scenarios or other unmodeled factors.

A. True Models

Any state-estimation algorithm requires a sufficiently accurate yet reasonably complex mathematical model of the underlying system. Unfortunately, often the “true” model, i.e., the system or data generator, is very complex and context-dependent and cannot be expressed in a finite-dimensional closed form. Instead, an approximate (and *reasonably* complex) PBM is constructed from the first-principles modeling.²

²Alternatively, a reasonably complex data-driven model found by system identification or by a combination of modeling and identification can be considered as well.



Fig. 4. General state-space structure dynamics.

To formalize and illustrate the problem, consider the “true” model (i.e., system or data generator) in the state-space form

$$\mathcal{M}^{\text{true}} : \quad \mathbf{x}_k = \mathbf{f}^{\text{true}}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}; \boldsymbol{\lambda}_{k-1}^{\text{true}}) + \mathbf{q}_{k-1}^{\text{true}} \quad (3)$$

where \mathbf{x}_k is the state at time instant t_k , $\mathbf{f}^{\text{true}}(\cdot)$ is unknown (and possibly complex) vector function, $\mathbf{q}_{k-1}^{\text{true}}$ is the state noise with PDFs $\mathbf{q}_{k-1}^{\text{true}} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q})$, and $\boldsymbol{\lambda}_{k-1}^{\text{true}}$ are model parameters (possibly time-varying) characterizing the context and influence of ambient environment.

B. Physics-Based Models

A simplified approximate PBM is then constructed in the form

$$\mathcal{M}^{\text{PBM}} : \quad \mathbf{x}_k = \mathbf{f}^{\text{PBM}}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}; \boldsymbol{\lambda}_{k-1}) + \mathbf{q}_{k-1}^{\text{PBM}} \quad (4)$$

where \mathbf{f}^{PBM} is *known* function and $\boldsymbol{\lambda}_{k-1}$ is an estimate of the “true” parameters $\boldsymbol{\lambda}_{k-1}^{\text{true}}$ with an error $\tilde{\boldsymbol{\lambda}}_{k-1}^{\text{true}} = \boldsymbol{\lambda}_{k-1}^{\text{true}} - \boldsymbol{\lambda}_{k-1}$. The parameter estimate $\boldsymbol{\lambda}_{k-1}$ can be found offline by system identification methods [94] or online by the joint or dual state estimation [95].

If not neglected in the state estimator design, the discrepancy of (i) the model structure [i.e., between $\mathbf{f}^{\text{PBM}}(\cdot)$ and $\mathbf{f}^{\text{true}}(\cdot)$] and (ii) the model parameters (i.e., between $\boldsymbol{\lambda}_{k-1}$ and $\boldsymbol{\lambda}_{k-1}^{\text{true}}$) is typically considered as “another *white*” component of the state noise resulting in inequality of covariance matrices $\text{cov}[\mathbf{q}_{k-1}^{\text{PBM}}] > \text{cov}[\mathbf{q}_{k-1}^{\text{true}}]$. The state-space model (4) with larger state noise (i.e., with a larger covariance matrix), then, inevitably leads to *less* accurate state estimates. More precisely, it is known that achievable estimation bounds are degraded when inaccurate or wrong models are used to derive optimal estimators, which in certain cases can jeopardize the estimation capabilities. This bound is called the misspecified Cramér–Rao Bound (MCRB) [96]. The model $\mathbf{f}^{\text{PBM}}(\cdot)$ is, then, often enhanced through data-driven methods, such that the mismatch between $\mathbf{f}^{\text{PBM}}(\cdot)$ and $\mathbf{f}^{\text{true}}(\cdot)$ is reduced and, consequently, the MCRB can be reduced.

C. Data-Driven Models

While ML appeared in system identification of input–output models a long time ago, see the excellent review in [81], most approaches using ML for system identification of state-space models have appeared only recently. They can be classified according to state-space structures embedding the NNs. In the general state-space structures (see Fig. 4), both functions in dynamics and measurement equations are replaced by a NN.

The data-driven model can be written in this case as

$$\mathcal{M}^{\text{NN}} : \quad \mathbf{x}_k = \mathbf{g}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}; \boldsymbol{\theta}^{\text{NN}}) + \mathbf{q}_{k-1}^{\text{NN}} \quad (5)$$

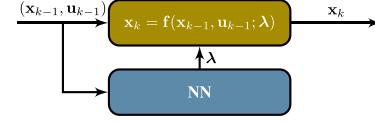


Fig. 5. Fixed model structure dynamics.

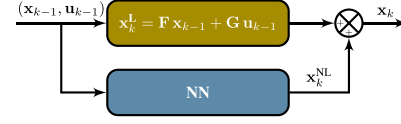


Fig. 6. Incremental state-space structure dynamics.

where $\mathbf{g}(\cdot)$ is a vector-valued function modeled as a NN being a function of the state and control, which is parameterized by the vector $\boldsymbol{\theta}^{\text{NN}}$ designed to minimize the discrepancy between (5) and the true model (3), i.e., designed so that $\text{cov}[\mathbf{q}_{k-1}^{\text{true}}] \approx \text{cov}[\mathbf{q}_{k-1}^{\text{NN}}]$.

Another approach to state-space model system identification is to consider a fixed model structure, such as a linear parameter-varying model and use the NNs to represent the matrices associated with Kalman equations [97], [98] (see Fig. 5).

A similar idea was elaborated in [99], where an incremental state-space structure has been proposed that separates \mathbf{f} into linear and nonlinear parts and uses a NN to represent the nonlinear part (see Fig. 6).

The main differences among the aforementioned models are in the way the available knowledge of the physics is considered. Namely, purely data driven models (Fig. 4) completely disregard the PBM. In Fig. 5, the PBM is assumed, with its parameters being learnt by a NN. Hence, the modeling adheres to the PBM structure. This corresponds to (adaptive) system identification. The model in Fig. 6 supplements the PBM approximation by a NN aimed at correcting the linear model in an *additive* manner. Here, no control is enforced on the NN, which may result in a negligible role of the linear model in the overall behavior. In summary, none of them fully exploits the potential of the PBM – often constructed by experts in the field – while exploiting the available data in a flexible manner.

D. AI Augmented PBMs

The pure PBM and general state-space structure dynamics using the NN are inherently tied with the abovementioned limitations. An attractive, yet not much explored, structure addressing the limitations augments the PBM by a NN to model the discrepancy between the PBM and the true system dynamics [100]. Due to embracing the full PBM, this structure may have great potential for state estimation as it efficiently combines the knowledge of some parts of the model while using NNs to represent the unknown or uncertain parts.

The proposed concept of the AI-augmented PBMs (APBMs) aims at the compensation of the true and approximate PBM structure and parameter discrepancy by a

TABLE I
Summary of pros/cons for the Various Modeling Approaches Discussed in Section III

	PBM	NN	NN-param	NN-increm	APBM
Pros	Accurate predictions under nominal conditions; Explainability, Great interpretability.	Potential to model complex phenomena; Efficient offline learning schemes.	Ability to model complex parameter correlations; Explainability, Interpretability; Efficient offline learning schemes.	Ability to complement physics-based knowledge; Limited explainability.	Potential to model complex phenomena under non-nominal conditions; Accurate performance under nominal conditions; Explainability.
Cons	Limited adaptability; Prone to modeling mismatches under non-nominal conditions; Expert knowledge required.	Vasts amounts of data required; Supervised training typically needed; No explainability.	Large amounts of data required; Limited adaptability to changing conditions.	Moderate amounts of data required; Moderate adaptability to non nominal conditions;	Moderate amounts of data required; Possibility for unsupervised learning.

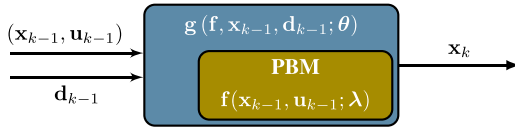


Fig. 7. Augmented PBM dynamics.

data-driven AI component processing additional data³ (hard and soft), as illustrated in Fig. 7. The resulting state-space model can be written in the following structure:

$\mathcal{M}^{\text{APBM}}$:

$$\mathbf{x}_k = \mathbf{g}(\mathbf{f}^{\text{PBM}}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}; \lambda), \mathbf{x}_{k-1}, \mathbf{d}_{k-1}; \theta) + \mathbf{q}_{k-1}^{\text{APBM}} \quad (6)$$

where \mathbf{d}_{k-1} represents the additional data, $\mathbf{g}(\cdot)$ is a vector-valued NN-based function that is aware of the PBM \mathbf{f}^{PBM} and is parameterized by the vector θ designed to minimize the discrepancy between (6) and the true model (3), i.e., designed so that⁴

$$\text{cov}[\mathbf{q}_{k-1}^{\text{true}}] \approx \text{cov}[\mathbf{q}_{k-1}^{\text{APBM}}]. \quad (7)$$

Compared with the standard AI-based solutions, APBMs can be used even without available training data (and without training), in which case the physics will be driving the model. Once data is available, the NN is trained (online or offline), and the APBM is used instead, leading to performance improvement. Note that the APBM approach also allows data-based tuning of the PBM parameter vector λ , which typically has to be (somehow) specified by the designer or the user.

The APBM framework takes advantage of combining both model- and AI-based approaches, thus enabling

- 1) natural incorporation of the physical insights of the system, allowing specification of the model as close

to reality as possible or specification of ranges of sought parameters, and

- 2) leveraging available data for model improvement, leading to a more accurate model and, subsequently, to more accurate inference results.

Essentially, the use of APBMs ensures that nonlinear complex models can be learned, whereas the well-understood physics of the nominal models are still enforced.

E. Comparison of Modeling Approaches

Different modeling approaches involve different trade-offs. Table I provides a comparison of the alternatives detailed earlier, namely, the use of PBM from Section III-B; data-driven models as in Fig. 4, whereby a NN model is used without knowledge of the physics of the problem; data-driven models as in Fig. 5, where an NN model is used to characterize the parameters of the PBM. Referred to as NN-param in the table; data-driven models as in Fig. 6, where the NN component is used to incrementally assist the physics-based component. Referred to as NN-increm in the table; and the APBM architecture in Fig. 7, where physics-based and data-driven are fully interlaced. The comparison tackles important aspects of the models, such as their validity in nonnominal conditions (that is, conditions under which they were initially learned), their explainability ability, and the need for supervised learning.

IV. ILLUSTRATIVE EXAMPLES

To further compare the different modeling approaches, we considered two illustrative examples: 1) a synthetic experiment considering a radar target tracking problem with nonGaussian measurement noise, which enabled the use of ground truth for error metrics benchmarking; and 2) a vehicular car tracking application, where real-data measurements were used. This experiment assisted in showing the performance of physics-based and APBM approaches under nonGaussian data and varying object dynamics.

³The additional data represent the data that are available and provide some information about the state, but that are not used by the PBM for various reasons.

⁴We suppose that the modeling uncertainty is summarized through the model covariance

A. Radar Target Tracking

To test the discussed approach, we consider a 2-D target tracking application with additive constant velocity [101] and sinusoidal [102] terms. Synthetic measurements from two collocated sensors measuring received signal strength and bearings were considered

$$\mathbf{y}_k = \mathbf{h}(\mathbf{p}_k^s, \mathbf{p}_k) + \mathbf{r}_k = \begin{pmatrix} 10 \log_{10} \left(\frac{\Psi_0}{\|\mathbf{p}_k^s - \mathbf{p}_k\|^\alpha} \right) \\ \angle(\mathbf{p}_k^s, \mathbf{p}_k) \end{pmatrix} + \mathbf{r}_k \quad (8)$$

with \mathbf{p}_k^s being the position of the sensors, \mathbf{p}_k the unknown position of the target, $10 \log_{10}(\Psi_0) = 30$ dBm, $\alpha = 2.2$ the path loss exponent, $\angle(\mathbf{p}_k^s, \mathbf{p}_k)$ denoting the angle between locations \mathbf{p}_k^s and \mathbf{p}_k in radians, and $\mathbf{r}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k)$, $\mathbf{R}_k = \text{diag}(0.1, 0.1)$, the measurement noise. In our experiments, we assume that the exact sensors positions are unknown [103], [104], but their estimates $\hat{\mathbf{p}}_k^s = \mathbf{p}_k^s + \mathbf{r}_k^s$ are provided by a GNSS system with \mathbf{r}_k^s representing the GNSS positioning error $\mathbf{r}_k^s \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k^s)$, $\mathbf{R}_k^s = \mathbf{I}$. Consequently, the measurement model (8) contains both additive and nonadditive noise components \mathbf{r}_k and \mathbf{r}_k^s , respectively. Thus, the measurement model in (8) can be rewritten as $\mathbf{y}_k = \mathbf{h}(\hat{\mathbf{p}}_k^s - \mathbf{r}_k^s, \mathbf{p}_k) + \mathbf{r}_k$.

For the state-estimation task, the nonadditive component was transformed into an additive form leading to the following model for the measurement equation $\mathbf{y}_k = \mathbf{h}(\hat{\mathbf{p}}_k^s, \mathbf{p}_k) + \mathbf{r}_k + \check{\mathbf{r}}_k^s$, where $\check{\mathbf{r}}_k^s$ represents the approximate additive effect of the nonadditive component \mathbf{r}_k^s on the measurement. Its moments, i.e., the mean and the covariance matrix, were obtained by the statistical linearization at the current estimate and the nonadditive noise component mean value [105].

Regarding the target's dynamical model we consider two scenarios. In the first scenario (S1) the dynamics of the target is simulated from a model composed of two parts, a *constant* velocity (CV) part and a *variable* turn-rate (VT) part. We refer to this model as CVVT

$$\mathcal{M}_{S1}^{\text{true}} : \quad \begin{aligned} \mathbf{x}_k &= (\mathbf{F} + \mathbf{G}_{k-1})\mathbf{x}_{k-1} + \mathbf{M}\mathbf{q}_{k-1} \\ \Omega_k &= \Omega_{k-1} + v_{k-1} \end{aligned} \quad (9)$$

with

$$\mathbf{F} = \begin{pmatrix} 1 & T_s & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T_s \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad \mathbf{M} = \begin{pmatrix} T_s^2/2 & 0 \\ T_s & 0 \\ 0 & T_s^2/2 \\ 0 & T_s \end{pmatrix}$$

$$\mathbf{G}_{k-1} = \begin{pmatrix} 0 & \frac{\sin \Omega_{k-1} T_s}{T_s} & 0 & -\frac{1 - \cos \Omega_{k-1} T_s}{\Omega_{k-1}} \\ 0 & \cos \Omega_{k-1} T_s & 0 & -\sin \Omega_{k-1} T_s \\ 0 & \frac{1 - \cos \Omega_{k-1} T_s}{\Omega_{k-1}} & 0 & \frac{\sin \Omega_{k-1} T_s}{\Omega_{k-1}} \\ 0 & \sin \Omega_{k-1} T_s & 0 & \cos \Omega_{k-1} T_s \end{pmatrix}$$

$\mathbf{x}_k = (x_k, \dot{x}_k, y_k, \dot{y}_k)^\top$ being a state vector, composed of the 2-D position ($\mathbf{p}_k = (x_k, y_k)^\top$) and velocity ($\dot{\mathbf{p}}_k = (\dot{x}_k, \dot{y}_k)^\top$) of the target, and Ω_k being an angle state. In our experiments we generate trajectories with the following parameters: $T_s = 1$ s is the sampling period and $\mathbf{q}_k \sim \mathcal{N}(\mathbf{0}, 0.01 \cdot \mathbf{I})$ is the process noise that models the acceleration as a random term. $v_k \sim \mathcal{N}(0, 10^{-5})$ is the process noise for the angle Ω_k .

The true trajectory was initialized at $\mathbf{x}_0 = (50, 0, 50, 0)^\top$ and $\Omega_0 = 0.05\pi$.

In the second scenario (S2) we consider a near CV model

$$\mathcal{M}_{S2}^{\text{true}} : \quad \mathcal{M}_{S1}^{\text{true}}(\mathbf{G}_{k-1} = \mathbf{0}) \quad \forall k \in \mathbb{Z}_+. \quad (10)$$

We performed Monte Carlo (MC) experiments where 100 trajectories were generated for each scenario for $T = 1000$ s, with true sensor position $\hat{\mathbf{p}}_k^s = (0, 0)^\top$.

We performed state estimation using four approaches. Two PBMs (Section III-B), CVVT and CV; a completely data-driven (Section III-C) using NN; and the APBM (Section III-D) composed of a CV term augmented with a NN. While CVVT and CV are described by (9) and (10), respectively, the NN model and APBM are given by

$$\mathcal{M}^{\text{NN}} : \quad \mathbf{x}_k = \gamma(\mathbf{x}_{k-1}; \boldsymbol{\omega}) + \mathbf{M}\mathbf{q}_{k-1} \quad (11)$$

and

$$\begin{aligned} \mathcal{M}^{\text{APBM}} : \quad \mathbf{x}_k &= g(f(\mathbf{x}_{k-1}), \mathbf{x}_{k-1}; \boldsymbol{\theta}) \\ &= \phi_0 \mathbf{F}\mathbf{x}_{k-1} + \phi_1 \gamma(\mathbf{x}_{k-1}; \boldsymbol{\omega}) + \mathbf{M}\mathbf{q}_{k-1} \end{aligned} \quad (12)$$

where $g(\cdot)$ is the APBM parameterized by $\boldsymbol{\theta} = \{\phi_0, \phi_1, \boldsymbol{\omega}\}$, $\gamma(\cdot)$ is a NN parameterized by $\boldsymbol{\omega}$ with one hidden layer with five neurons and ReLu activation function, an output layer with four neurons and linear activation, leading to a total of $42 + 2$ parameters including bias terms.

The unknown target tracking scenario with access only to noise measurements of position states makes the supervised (in terms of pairs on inputs and outputs of the transition function) training for NN model and APBM unfeasible. For this reason, similarly to [100], we opted for an online training approach by augmenting the state space with model parameters and using cubature rules to solve the recursive Bayesian equations under Gaussianity assumptions. Specifically, we used the cubature Kalman filter approach [102]. To control NN contribution in APBMs and make purely NN-transitions more stable we leveraged the augmented likelihood model discussed in [100], which effectively introduces a regularization to model parameters that can be controlled by a hyperparameter λ . Here, we set $\lambda_{\text{APBM}} = \lambda_{\text{NN}} = 0.05$ so that a number of MC realizations diverging for both models becomes comparable with the CVVT for S1 and CV for S2. For all methods $\hat{\mathbf{x}}_0$ was drawn from a Gaussian distribution with mean \mathbf{x}_0 and covariance $\text{diag}(0.1, 0.01, 0.1, 0.01)$. For the CVVT we initialized $\hat{\Omega}_0$ drawing from $\mathcal{N}(\Omega_0, 10^{-5})$.

A fragment of one realization of the true trajectory (under S1) and the estimated trajectories for the four models can be seen in Fig. 8. The root-mean-square-error (RMSE) over the time and MC simulations for the two scenarios S1 (left plots) and S2 (right plots) can be found in Fig. 9. The top plots illustrate the RMSE in the form of the box-plots (across all MC realizations) and the bottom figures visualize the RMSE results using the cumulative distribution function (CDF) for a single realization.

It can be seen that in Scenario S1, with a dominant turning rate (TR) component, the CVVT leads to the best results, and the CV results in quite large estimation errors.

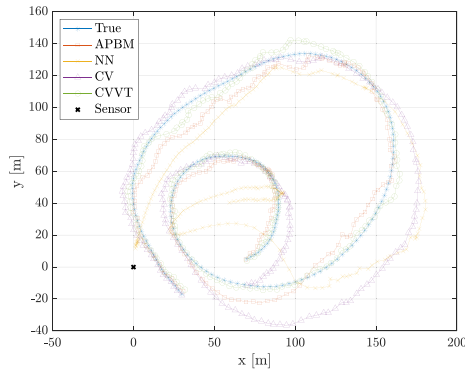


Fig. 8. True path for scenario S1 and estimated trajectories using the CV, CVVT, NN, and APBM models.

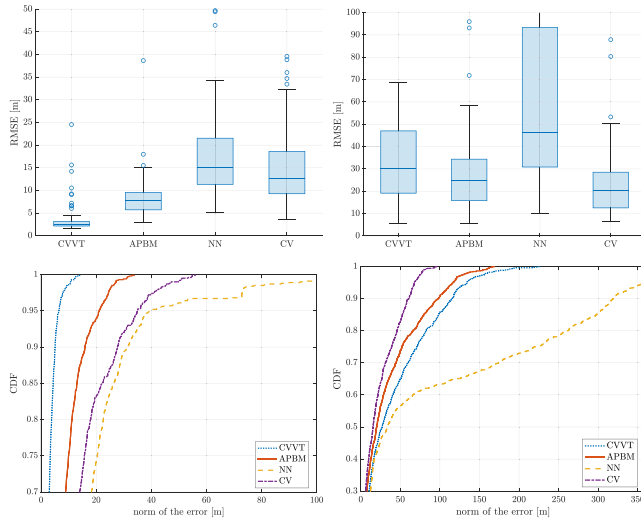


Fig. 9. RMSE box-plots over 100 Monte Carlo simulations and CDFs of single realizations for trajectories generated under S1 (left) and S2 (right). λ was set so that the numbers of divergent were comparable with the true models. For S1 (left plots) we had 0, 0, 2, and 0 divergent runs for CVVT, APBM, NN, and APBM, respectively. For S2 the number of divergent runs was 2, 2, 5, and 0.

Although the APBM is based on the CV model, the NN-based component is able to compensate to some extent for the discrepancy between the true model and APBM physics component. Examining the results for Scenario S2, we can see that the CVVT leads to poor performance due to the model mismatch whereas CV leads to the best performance. We highlight that the CV model and the APBM provide very similar performance, which are far superior to the ones achieved with the CVVT and NN models. Comparing the plots for S1 and S2, the APBM seems to offer good estimation performance with the minimized effort on a priori selection of the PBM model. On the other hand, the NN-based model provides the least accurate estimates. This is also expected since NNs are trained online without any prior information regarding the underlying dynamics. Code is also available (upon paper acceptance) and the interested reader is encouraged to use it to explore different parameters and models.

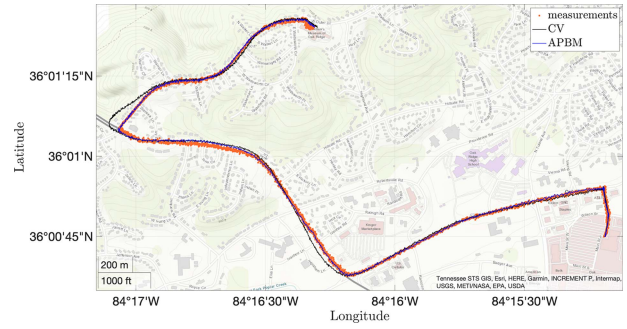


Fig. 10. GPS-based car (starting from the right) tracking. Noisy measurements are depicted by orange dots whereas estimated trajectories obtained using CV and APBM are depicted by solid lines. The APBM provides more accurate trajectory estimates, particularly during sharp turns of the vehicle whereas the CV model has a longer convergence time due to the assumed inertia.

In general, the results clearly indicate that online learning trajectory models are challenging and leveraging physical knowledge can dramatically improve model performance.

B. Car Tracking From Real GPS Data

To compare the APBM and its physics-based counterpart model, CV, on real data we considered a vehicular tracking problem where noisy GNSS positions are available whereas the dynamics of the receiver are not accurately known a priori. To that aim, we used a GNSS dataset of digitized RF signals named the Oak Ridge Spoofing and Interference Test Battery (OAKBAT) [106], recorded with the bandwidth of 5 MHz. The dataset contains digitized recordings of the GPS L1 C/A signal. Besides the six spoofing sets, OAKBAT contains two spoofing-free baseline sets: static and dynamic. For the experiment, we used the spoofing-free dynamic set. The sampled RF signal was processed using the GNSS-software-defined radio (SDR) [107], an open-source GNSS-SDR receiver. The software was used for the acquisition; tracking; and position, velocity, and time (PVT) computation steps. The position component of the PVT was then used as the noisy measurement for the vehicular tracking experiment, where states contained both position and velocities. The dataset used is representative of nonGaussian, possibly biased observations, given the vehicle travels a multipath-rich environment.

Since the underlying model is, in fact, unknown, we considered only CV and APBM models as described in (10) and (12), respectively. In this experiment, covariances and parameters were set to $\mathbf{R} = 10 \cdot \mathbf{I}$, $\mathbf{Q} = 10^{-3} \cdot \mathbf{I}$, $T_s = 0.02$ s, and $\lambda_{\text{APBM}} = 10^8$.

Fig. 10 depicts the measurements (light orange dots) and the estimated trajectories provided by CV (black) and APBM models (blue). It is clear that during turns the APBM compensated for the overconfidence of the CV on its dynamics, providing a more accurate estimation of the trajectory. Despite the fact that there is no ground truth for this model and, thus, providing a quantitative performance metric is challenging we can observe that the trajectory obtained with

the APBM is confined to the road limits for a substantially longer time than the CV, especially during sharp turns. This real data experiment shows that the flexibility offered by APBMs can outperform purely PBMs, when these are not accurately representing the dynamics of the state.

V. CHALLENGES AND FUTURE RESEARCH

A. Explainability, Quantification, and Control of NN Contribution

When referring to APBMs and other hybrid model variants, a fundamental point for the explainability of the model relates to the ability to quantify and control the NN contribution to the overall model. While in [100], an augmented likelihood model was used for controlling the NN contribution to the overall model, the quantification is still an open problem. Furthermore, the augmented likelihood approach leads to a tremendous increase in the computational cost since it increases the dimension of the measurements with the number of model parameters.

B. Theoretical Guarantees

Although NNs and hybrid models, such as the APBM can lead to improved performances in many scenarios, the lack of convergence guarantees, bounds on their approximation capabilities, and guarantees over the observability of states, especially under online scenarios, make their applications risky under critical safety applications. Therefore, there is a need for sound theoretical work discussing and addressing these points.

While there are several observability and identifiability analysis frameworks in the literature [108], [109], [110], only a few results exist when NNs are included in the models [111], [112]. The inclusion of NNs leads to clear observability issues since different solutions (e.g., permutations of parameters and layers can lead to the same effective model. However, how this impacts the observability of states with NN components as part of the model is still a plain field.

Models are, in essence, misspecified mathematical approximations of complex underlying real-world phenomena. Although bounds on the variance of estimators, such as the Cramér-Rao bound (CRB), exist under misspecified models [96], [113], [114], Bayesian versions of MCRB have only been addressed recently [115] and in a nonrecursive fashion and assuming very simplistic models.

C. Identification of Noise Parameters

The combination of PBM and NN-based models in hybrid models calls for the specification of the state noise parameters, such as the mean and covariance matrix. While many methods for noise parameter estimation have been proposed [116], [117], classified into correlation, covariance matching, likelihood, and Bayesian methods, most of them are limited to linear models and batch processing of the data. The inclusion of NNs inevitably leads to (highly) nonlinear models, which is the first limiting factor. Online scenarios then require a recursive update of the noise parameter estimates, and a simple extending the state with the

parameters may make the problem intractable as the state also contains the NN parameters. Identification of noise parameters in hybrid models is, thus, another challenging issue tied to the APBM.

D. Computational Aspects

One of the advantages of NN models is often the capability to generate approximations of complex phenomena at relatively low computational costs. In online scenarios, though, the need for constant adaptation of model parameters introduces a tremendous computational burden. This is even more problematic when performing recursive Bayesian inference where nonlinearities require integral approximations via either linearization [105], sampling [102], [118], [119], or other approximations, such as variational methods [120], [121]. Although variational approximations leverage the machinery of NN toolboxes and can be trained using stochastic gradient descent, they are slow to converge, trained over a time window, and, thus, are still far from ideal for real-time applications. In nononline/real-time settings, time-series modeling via NNs requires back-propagation through time, which can lead to further issues, such as vanishing gradients (especially for long sequences), and the need for accurate initial state estimations when deployed.

VI. CONCLUSION

This article discusses the importance of modeling dynamics in state estimation, and in particular in the context of N&T systems. When purely physics-based modeling cannot be realistically used, a current trend is to leverage data-driven models. This article presented various alternatives to employ data-driven methods as an alternative or jointly with PBMs, where different tradeoffs were identified. Among the approaches, augmented PBMs are a powerful modeling tool that accounts for physics-informed components, which provide model explainability, while leveraging data-driven ML models that assist in learning complex behaviors. Two illustrative examples, both with synthetic and real dataset, were discussed to better understand those modeling choices and their implications in the state estimation performance.

REFERENCES

- [1] J. Dampf et al., "More than we ever dreamed possible: Processor technology for GNSS software receivers in the year 2015," *Inside GNSS*, vol. 10, no. 4, pp. 62–72, Jul. 2015.
- [2] Y. Li et al., "Inertial sensing meets machine learning: Opportunity or challenge?," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 8, pp. 9995–10011, Aug. 2022.
- [3] T. Meng, X. Jing, Z. Yan, and W. Pedrycz, "A survey on machine learning for data fusion," *Inf. Fusion*, vol. 57, pp. 115–129, 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1566253519303902>
- [4] A. Wrabel, R. Graef, and T. Brosch, "A survey of artificial intelligence approaches for target surveillance with radar sensors," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 36, no. 7, pp. 26–43, Jul. 2021.
- [5] E. Blasch et al., "Machine learning/artificial intelligence for sensor data fusion—opportunities and challenges," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 36, no. 7, pp. 80–93, Jul. 2021.
- [6] B. Azari et al., "Automated deep learning-based wide-band receiver," *Comput. Netw.*, vol. 218, 2022, Art. no. 109367.

- [7] *Software Considerations in Airborne Systems and Equipment Certification*, Standard No. DO-178C, Jan. 2012.
- [8] S. Jacklin, "Certification of safety-critical software under DO-178 C and DO-278 A," in *Proc. Infotech@ Aerosp. AIAA*, 2012, Art. no. 2473.
- [9] EASA, "Artificial intelligence roadmap: A human-centric approach to AI in aviation," European Union Aviation Safety Agency (EASA), Cologne, Germany, Tech. Rep. 1.0, Feb. 2020.
- [10] FAA, "Identification of artificial intelligence (AI) applications for maintenance, monitoring, and control of airway facilities," Federal Aviation Administration (FAA), Atlantic City Airport, NJ, USA, Tech. Rep. DOT-FAA-CT-TN92-41-1, May 1994.
- [11] E. Mazor, A. Averbuch, Y. B.-Shalom, and J. Dayan, "Interacting multiple model methods in target tracking: A survey," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 34, no. 1, pp. 103–123, Jan. 1998.
- [12] J. Duník, S. K. Biswas, A. G. Dempster, T. Pany, and P. Closas, "State estimation methods in navigation: Overview and application," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 35, no. 12, pp. 16–31, Dec. 2020.
- [13] D. Dardari, E. Falletti, and M. Luise, *Satellite and Terrestrial Radio Positioning Techniques: A Signal Processing Perspective*. Orlando, FL, USA: Academic Press, 2011.
- [14] M. G. Amin, P. Closas, A. Broumandan, and J. L. Volakis, "Vulnerabilities, threats, and authentication in satellite-based navigation systems [scanning the issue]," *Proc. IEEE*, vol. 104, no. 6, pp. 1169–1173, Jun. 2016.
- [15] D. Dardari, P. Closas, and P. M. Djurić, "Indoor tracking: Theory, methods, and technologies," *IEEE Trans. Veh. Technol.*, vol. 64, no. 4, pp. 1263–1278, Apr. 2015.
- [16] Z. M. Kassas, P. Closas, and J. Gross, "Navigation systems panel report navigation systems for autonomous and semi-autonomous vehicles: Current trends and future challenges," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 34, no. 5, pp. 82–84, May 2019.
- [17] Y. J. Morton, F. van Diggelen, J. J. Spilker Jr, B. W. Parkinson, S. Lo, and G. Gao, *Position, Navigation, and Timing Technologies in the 21st Century: Integrated Satellite Navigation, Sensor Systems, and Civil Applications*. Hoboken, NJ, USA: Wiley, 2021.
- [18] J. Proakis and M. Salehi, *Digital Communications*, 5th ed. New York, NY, USA: McGraw-Hill, 2001.
- [19] A. Siemuri, H. Kuusniemi, M. S. Elmusrati, P. Välisuo, and A. Shamsuzzoha, "Machine learning utilization in GNSS—use cases, challenges and future applications," in *Proc. Int. Conf. Localization GNSS*, 2021, pp. 1–6.
- [20] R. M. Ferre, A. de la Fuente, and E. S. Lohan, "Jammer classification in GNSS bands via machine learning algorithms," *Sensors*, vol. 19, no. 22, 2019, Art. no. 4841.
- [21] A. Louis and M. Raimondi, "Neural network based evil waveforms detection," in *Proc. 33rd Int. Tech. Meeting Satell. Division Inst. Navigation*, 2020, pp. 1984–1989.
- [22] P. Wu, H. Calatrava, T. Imbiriba, and P. Closas, "Jammer classification with federated learning," in *Proc. IEEE/ION Position, Location Navigation Symp.*, 2023, pp. 228–234.
- [23] A. Nardin, T. Imbiriba, and P. Closas, "Jamming source localization using augmented physics-based model," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2023, pp. 1–5.
- [24] M. Mosavi and F. Shafiee, "Narrowband interference suppression for GPS navigation using neural networks," *GPS Solutions*, vol. 20, no. 3, pp. 341–351, 2016.
- [25] M. R. Manesh, J. Kenney, W. C. Hu, V. K. Devabhaktuni, and N. Kaabouch, "Detection of GPS spoofing attacks on unmanned aerial systems," in *Proc. IEEE 16th Annu. Consum. Commun. Netw. Conf.*, 2019, pp. 1–6.
- [26] P. B.-Darian, H. Li, P. Wu, and P. Closas, "Deep neural network approach to detect GNSS spoofing attacks," in *Proc. 33rd Int. Tech. Meeting Satell. Division Inst. Navigation*, 2020, pp. 3241–3252.
- [27] S. Tohidi and M. R. Mosavi, "Effective detection of GNSS spoofing attack using a multi-layer perceptron neural network classifier trained by PSO," in *Proc. 25th Int. Comput. Conf., Comput. Soc.*, 2020, pp. 1–5.
- [28] R. C.-Palomino, A. Bhattacharya, G. Bovet, and D. Giustiniano, "Short: LSTM-based GNSS spoofing detection using low-cost spectrum sensors," in *Proc. IEEE 21st Int. Symp. "World Wireless, Mobile Multimedia Netw."*, 2020, pp. 273–276.
- [29] S. Semanjski, A. Muls, I. Semanjski, and W. De Wilde, "Use and validation of supervised machine learning approach for detection of GNSS signal spoofing," in *Proc. Int. Conf. Localization GNSS*, 2019, pp. 1–6.
- [30] Y. Quan, L. Lau, G. W. Roberts, X. Meng, and C. Zhang, "Convolutional neural network based multipath detection method for static and kinematic GPS high precision positioning," *Remote Sens.*, vol. 10, no. 12, 2018, Art. no. 2052.
- [31] R. Sun, L.-T. Hsu, D. Xue, G. Zhang, and W. Y. Ochieng, "GPS signal reception classification using adaptive neuro-fuzzy inference system," *J. Navigation*, vol. 72, no. 3, pp. 685–701, 2019.
- [32] L.-T. Hsu, "GNSS multipath detection using a machine learning approach," in *Proc. IEEE 20th Int. Conf. Intell. Transp. Syst.*, 2017, pp. 1–6.
- [33] B. Guermah, H. E. Ghazi, T. Sadiki, and H. Guermah, "A robust GNSS LOS/multipath signal classifier based on the fusion of information and machine learning for intelligent transportation systems," in *Proc. IEEE Int. Conf. Technol. Manage., Operations Decis.*, 2018, pp. 94–100.
- [34] G. Zhang, P. Xu, H. Xu, and L.-T. Hsu, "Prediction on the urban GNSS measurement uncertainty based on deep learning networks with long short-term memory," *IEEE Sensors J.*, vol. 21, no. 18, pp. 20563–20577, Sep. 2021.
- [35] N. I. Ziedan, "Optimized position estimation in multipath environments using machine learning," in *Proc. 34th Int. Tech. Meeting Satell. Division Inst. Navigation*, 2021, pp. 3437–3451.
- [36] G. Caparra, P. Zoccarato, and F. Melman, "Machine learning correction for improved PVT accuracy," in *Proc. 34th Int. Tech. Meeting Satell. Division Inst. Navigation*, 2021, pp. 3392–3401.
- [37] P. B.-Darian and P. Closas, "Deep neural network approach to GNSS signal acquisition," in *Proc. IEEE/ION Position, Location Navigation Symp.*, 2020, pp. 1214–1223.
- [38] P. B.-Darian, H. Li, P. Wu, and P. Closas, "Deep learning of GNSS acquisition," *Sensors*, vol. 23, no. 3, 2023, Art. no. 1566.
- [39] M. Orabi, J. Khalife, A. A. Abdallah, Z. M. Kassas, and S. S. Saab, "A machine learning approach for GPS code phase estimation in multipath environments," in *Proc. IEEE/ION Position, Location Navigation Symp.*, 2020, pp. 1224–1229.
- [40] H. Li, P. B.-Darian, P. Wu, and P. Closas, "Deep neural network correlators for GNSS multipath mitigation," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 59, no. 2, pp. 1249–1259, Apr. 2023.
- [41] C. Savas and F. Dovis, "Multipath detection based on k-means clustering," in *Proc. 32nd Int. Tech. Meeting Satell. Division Inst. Navigation*, 2019, pp. 3801–3811.
- [42] T. Suzuki, K. Kusama, and Y. Amano, "NLOS multipath detection using convolutional neural network," in *Proc. 33rd Int. Tech. Meeting Satell. Division Inst. Navigation*, 2020, pp. 2989–3000.
- [43] Q. Liu, Z. Huang, and J. Wang, "Indoor non-line-of-sight and multipath detection using deep learning approach," *GPS Solutions*, vol. 23, no. 3, pp. 1–14, 2019.
- [44] E. Munin, A. Blais, and N. Couellan, "Convolutional neural network for multipath detection in GNSS receivers," in *Proc. Int. Conf. Artif. Intell. Data Analytics Air Transp.*, 2020, pp. 1–10.
- [45] H. Li, P. B.-Darian, P. Wu, and P. Closas, "Deep learning of GNSS signal correlation," in *Proc. 33rd Int. Tech. Meeting Satell. Division Inst. Navigation*, 2020, pp. 2836–2847.
- [46] D. Brum et al., "A proposed earthquake warning system based on ionospheric anomalies derived from GNSS measurements and artificial neural networks," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2019, pp. 9295–9298.
- [47] M. Alshaye, F. Alawwad, and I. Elshafiey, "Hurricane tracking using multi-GNSS-R and deep learning," in *Proc. 3rd Int. Conf. Comput. Appl. Inf. Secur.*, 2020, pp. 1–4.

- [48] Q. Yan and W. Huang, "Sea ice sensing from GNSS-R data using convolutional neural networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 10, pp. 1510–1514, Oct. 2018.
- [49] Q. Yuan et al., "Deep learning in environmental remote sensing: Achievements and challenges," *Remote Sens. Environ.*, vol. 241, 2020, Art. no. 111716.
- [50] Y. L. Liu, Y. Morton, and Y. J. Jiao, "Application of machine learning to characterization of GPS L1 ionospheric amplitude scintillation," in *Proc. IEEE/ION Position, Location Navigation Symp.*, 2018, pp. 1159–1166.
- [51] N. Linty, A. Farasin, A. Favenza, and F. Dovis, "Detection of GNSS ionospheric scintillations based on machine learning decision tree," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 55, no. 1, pp. 303–317, Feb. 2019.
- [52] L. Mallika et al., "Machine learning algorithm to forecast ionospheric time delays using global navigation satellite system observations," *Acta Astronautica*, vol. 173, pp. 221–231, 2020.
- [53] R. O. Perez, "Using tensorflow-based neural network to estimate GNSS single frequency ionospheric delay (IONONet)," *Adv. Space Res.*, vol. 63, no. 5, pp. 1607–1618, 2019.
- [54] Second Workshop on Machine Learning and the Physical Sciences (NeurIPS 2019), Vancouver Canada, 2019.
- [55] M. O. Selbesoglu, "Prediction of tropospheric wet delay by an artificial neural network model based on meteorological and GNSS data," *Eng. Sci. Technol., Int. J.*, vol. 23, no. 5, pp. 967–972, 2020.
- [56] J. V.-Valls, N. Linty, P. Closas, F. Dovis, and J. T. Curran, "Survey on signal processing for GNSS under ionospheric scintillation: Detection, monitoring, and mitigation," *J. Inst. Navigation*, vol. 67, no. 3, pp. 511–536, 2020.
- [57] P. D. Groves, *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems*. Norwood, MA, USA: Artech House, 2008.
- [58] R. M. Rogers, *Applied Mathematics in Integrated Navigation Systems*, 2nd ed. Amer. Inst. Aeronaut. Astronaut., Inc. Alexander Bell Drive, Reston, VA, 2003.
- [59] P. Picerno, M. Iosa, C. D'Souza, M. G. Benedetti, S. Paolucci, and G. Morone, "Wearable inertial sensors for human movement analysis: A five-year update," *Expert Rev. Med. Devices*, vol. 18, no. sup1, pp. 79–94, 2021.
- [60] F. Nobre and C. Heckman, "Learning to calibrate: Reinforcement learning for guided calibration of visual-inertial rigs," *Int. J. Robot. Res.*, vol. 38, no. 12/13, pp. 1388–1402, 2019.
- [61] C. Xiyuan, "Modeling random gyro drift by time series neural networks and by traditional method," in *Proc. Int. Conf. Neural Netw. Signal Process.*, 2003, pp. 810–813.
- [62] S. Chong et al., "Temperature drift modeling of MEMS gyroscope based on genetic-Elman neural network," *Mech. Syst. Signal Process.*, vol. 72/73, pp. 897–905, 2016. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0888327015005002>
- [63] S. Wang, W. Zhu, Y. Shen, J. Ren, H. Gu, and X. Wei, "Temperature compensation for mems resonant accelerometer based on genetic algorithm optimized backpropagation neural network," *Sensors Actuators A: Phys.*, vol. 316, 2020, Art. no. 112393.
- [64] E. Pukhov and H. I. Cohen, "Novel approach to improve performance of inertial navigation system via neural network," in *Proc. IEEE/ION Position, Location Navigation Symp.*, 2020, pp. 746–754.
- [65] B.-S. Lin, I.-J. Lee, S.-P. Wang, J.-L. Chen, and B.-S. Lin, "Residual neural network and long short-term memory-based algorithm for estimating the motion trajectory of inertial measurement units," *IEEE Sensors J.*, vol. 22, no. 7, pp. 6910–6919, Apr. 2022.
- [66] K.-W. Chiang, Y.-W. Huang, and X. Niu, "Rapid and accurate ins alignment for land applications," *Surv. Rev.*, vol. 42, no. 317, pp. 279–291, 2010.
- [67] Y. Kone, N. Zhu, V. Renaudin, and M. Ortiz, "Machine learning-based zero-velocity detection for inertial pedestrian navigation," *IEEE Sensors J.*, vol. 20, no. 20, pp. 12343–12353, Oct. 2020.
- [68] B. Hu, P. Dixon, J. Jacobs, J. Dennerlein, and J. Schiffman, "Machine learning algorithms based on signals from a single wearable inertial sensor can detect surface- and age-related differences in walking," *J. Biomech.*, vol. 71, pp. 37–42, 2018.
- [69] C. Chen, P. Zhao, C. X. Lu, W. Wang, A. Markham, and N. Trigoni, "Deep-learning-based pedestrian inertial navigation: Methods, data set, and on-device inference," *IEEE Internet Things J.*, vol. 7, no. 5, pp. 4431–4441, May 2020.
- [70] T. Siddique and M. S. Mahmud, "Real-time machine learning enabled low-cost magnetometer system," in *Proc. IEEE Sensors*, 2022, pp. 1–4.
- [71] A. Gnadt, *Machine Learning-Enhanced Magnetic Calibration for Airborne Magnetic Anomaly Navigation*, Amer. Inst. Aeronautics Astronaut., San Diego, CA, Jan. 3–7, 2022.
- [72] C. Kanellakis and G. Nikolakopoulos, "Survey on computer vision for UAVs: Current developments and trends," *J. Intell. Robot. Syst.*, vol. 87, pp. 141–168, 2017.
- [73] L. Zhou, H. Yu, Y. Lan, and M. Xing, "Artificial intelligence in interferometric synthetic aperture radar phase unwrapping: A review," *IEEE Geosci. Remote Sens. Mag.*, vol. 9, no. 2, pp. 10–28, Jun. 2021.
- [74] S. S. Vayghan, M. Salmani, N. Ghasemkhani, B. Pradhan, and A. Alamri, "Artificial intelligence techniques in extracting building and tree footprints using aerial imagery and LiDAR data," *Geocarto Int.*, vol. 37, no. 10, pp. 2967–2995, 2022.
- [75] A. Cherian, J. Andersh, V. Morellas, N. Papanikolopoulos, and B. Mettler, "Autonomous altitude estimation of a UAV using a single onboard camera," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2009, pp. 3900–3905.
- [76] B. Grelsson, "Vision-based localization and attitude estimation methods in natural environments," Ph.D. dissertation, Linköping Univ., Linköping, Sweden, 2019.
- [77] T. Qin, P. Li, and S. Shen, "VINS-Mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Trans. Robot.*, vol. 34, no. 4, pp. 1004–1020, Aug. 2018.
- [78] W. Li and C.-Y. Hsu, "Automated terrain feature identification from remote sensing imagery: A deep learning approach," *Int. J. Geographical Inf. Sci.*, vol. 34, no. 4, pp. 637–660, 2020.
- [79] C. Gevaert, C. Persello, F. Nex, and G. Vosselman, "A deep learning approach to DTM extraction from imagery using rule-based training labels," *ISPRS J. Photogrammetry Remote Sens.*, vol. 142, pp. 106–123, 2018.
- [80] A. Khan, V. Ghorbanian, and D. Lowther, "Deep learning for magnetic field estimation," *IEEE Trans. Magn.*, vol. 55, no. 6, Jun. 2019, Art. no. 7202304.
- [81] A. Chiuso and G. Pillonetto, "System identification: A machine learning perspective," *Annu. Rev. Control, Robot., Auton. Syst.*, vol. 2, no. 1, pp. 281–304, 2019.
- [82] L. Ljung, C. Andersson, K. Tiels, and T. B. Schön, "Deep learning and system identification," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 1175–1181, 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2405896320317353>
- [83] L. Chin, "Application of neural networks in target tracking data fusion," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 30, no. 1, pp. 281–287, Jan. 1994.
- [84] V. Vaidehi, N. Chitra, C. Krishnan, and M. Chokkalingam, "Neural network aided Kalman filtering for multitarget tracking applications," in *Proc. IEEE Radar Conf. Radar Next Millennium*, 1999, pp. 160–165.
- [85] M. Owen and A. Stubberud, "NEKF IMM tracking algorithm," *Proc. SPIE*, vol. 5204, pp. 223–233, 2003.
- [86] N. Shaukat, A. Ali, M. Iqbal, M. Moinuddin, and P. Otero, "Multi-sensor fusion for underwater vehicle localization by augmentation of RBF neural network and error-state Kalman filter," *Sensors*, vol. 21, 2021, Art. no. 1149.
- [87] C. Gao, J. Yan, S. Zhou, P. Varshney, and H. Liu, "Long short-term memory-based deep recurrent neural networks for target tracking," *Inf. Sci.*, vol. 502, pp. 279–296, 2019.
- [88] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

- [89] S. Jung, I. Schlangen, and A. Charlish, "A mnemonic Kalman filter for non-linear systems with extensive temporal dependencies," *IEEE Signal Process. Lett.*, vol. 27, pp. 1005–1009, 2020.
- [90] K. Thormann, F. Sigges, and M. Baum, "Learning an object tracker with a random forest and simulated measurements," in *Proc. 20th Int. Conf. Inf. Fusion*, 2017, pp. 1–4.
- [91] P. Becker, H. Pandya, G. Gebhardt, C. Zhao, C. J. Taylor, and G. Neumann, "Recurrent Kalman networks: Factorized inference in high-dimensional deep feature spaces," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 544–552.
- [92] B. Zhai, W. Yi, M. Li, H. Ju, and L. Kong, "Data-driven XGBoost-based filter for target tracking," *J. Eng.*, no. 20, pp. 6683–6687, 2019.
- [93] P. D. Groves, "Principles of GNSS, inertial, and multisensor integrated navigation systems, [Book review]," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 30, no. 2, pp. 26–27, Feb. 2015.
- [94] L. Ljung, *System Identification: Theory for the User*. Upper Saddle River, NJ, USA: Prentice-Hall, 1999.
- [95] S. Haykin, *Kalman Filtering and Neural Networks*, vol. 47. Hoboken, NJ, USA: Wiley, 2004.
- [96] S. Fortunati, F. Gini, M. S. Greco, and C. D. Richmond, "Performance bounds for parameter estimation under misspecified models: Fundamental findings and applications," *IEEE Signal Process. Mag.*, vol. 34, no. 6, pp. 142–157, Nov. 2017.
- [97] Y. Bao, J. M. Velni, A. Basina, and M. Shahbakhti, "Identification of state-space linear parameter-varying models using artificial neural networks," *IFAC-PapersOnLine*, vol. 53, pp. 5286–5291, 2020.
- [98] C. Zhao, L. Sun, Z. Yan, G. Neumann, T. Duckett, and R. Stolkin, "Learning Kalman network: A deep monocular visual odometry for on-road driving," *Robot. Auton. Syst.*, vol. 121, 2019, Art. no. 103234.
- [99] M. Schoukens, "Improved initialization of state-space artificial neural networks," in *Proc. Eur. Control Conf.*, Delft, The Netherlands, 2021, pp. 1913–1918, doi: [10.23919/ECC54610.2021.9655207](https://doi.org/10.23919/ECC54610.2021.9655207).
- [100] T. Imbiriba, A. Demirkaya, J. Duník, O. Straka, D. Erdoğan, and P. Closas, "Hybrid neural network augmented physics-based models for nonlinear filtering," in *Proc. 25th Int. Conf. Inf. Fusion*, 2022, pp. 1–6.
- [101] P. Closas and C. F. Prades, "Particle filtering with adaptive number of particles," in *Proc. Aerosp. Conf.*, 2011, pp. 1–7.
- [102] I. Arasaratnam and S. Haykin, "Cubature Kalman filters," *IEEE Trans. Autom. Control*, vol. 54, no. 6, pp. 1254–1269, Jun. 2009.
- [103] M. S. Arulampalam, B. Ristic, N. Gordon, and T. Mansell, "Bearings-only tracking of manoeuvring targets using particle filters," *EURASIP J. Adv. Signal Process.*, vol. 2004, pp. 1–15, 2004.
- [104] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Boston, MA, USA: Artech House, 2004.
- [105] S. Särkkä, *Bayesian Filtering and Smoothing*. New York, NY, USA: Cambridge Univ. Press, 2013.
- [106] A. Albright, S. Powers, J. Bonior, and F. Combs, "Oak ridge spoofing and interference test battery (OAKBAT)-Pure tones," Oak Ridge National Lab (ORNL), Oak Ridge, TN, US, Tech. Rep. 10.13139/ORNLNCCS/1665888, 2020.
- [107] C. F. Prades, J. Arribas, P. Closas, C. Avilés, and L. Esteve, "GNSS-SDR: An open source tool for researchers and developers," in *Proc. 24th Int. Tech. Meeting Satell. Division Inst. Navigation*, 2011, pp. 780–794.
- [108] A. F. Villaverde, A. Barreiro, and A. Papachristodoulou, "Structural identifiability of dynamic systems biology models," *PLoS Comput. Biol.*, vol. 12, no. 10, 2016, Art. no. e1005153.
- [109] A. F. Villaverde et al., "Observability and structural identifiability of nonlinear biological systems," *Complexity*, vol. 2019, pp. 1–12, 2019.
- [110] L. A. Aguirre, L. L. Portes, and C. Letellier, "Structural, dynamical and symbolic observability: From dynamical systems to networks," *PLoS One*, vol. 13, no. 10, 2018, Art. no. e0206180.
- [111] F. Albertini and E. D. Sontag, "State observability in recurrent neural networks," *Syst. Control Lett.*, vol. 22, no. 4, pp. 235–244, 1994.
- [112] M. T. Angulo, A. Aparicio, and C. H. Moog, "Structural accessibility and structural observability of nonlinear networked systems," *IEEE Trans. Netw. Sci. Eng.*, vol. 7, no. 3, pp. 1656–1666, Jul.–Sep. 2020.
- [113] C. D. Richmond, "On constraints in parameter estimation and model misspecification," in *Proc. 21st Int. Conf. Inf. Fusion*, 2018, pp. 1080–1085.
- [114] C. D. Richmond and L. L. Horowitz, "Parameter bounds on estimation accuracy under model misspecification," *IEEE Trans. Signal Process.*, vol. 63, no. 9, pp. 2263–2278, May 2015.
- [115] S. Tang, T. Imbiriba, and P. Closas, "On parametric misspecified Bayesian Cramer-Rao bound: An application to linear Gaussian systems," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2023, pp. 1–5.
- [116] J. Duník, O. Straka, O. Kost, and J. Havlík, "Noise covariance matrices in state-space models: A survey and comparison—Part I," *Int. J. Adaptive Control Signal Process.*, vol. 31, pp. 1505–1543, 2017.
- [117] O. Kost, J. Duník, and O. Straka, "Measurement difference method: A universal tool for noise identification," *IEEE Trans. Autom. Control*, vol. 68, no. 3, pp. 1792–1799, Mar. 2023.
- [118] E. A. Wan, R. V. der Merwe, and A. T. Nelson, "Dual estimation and the unscented transformation," in *Proc. Neural Inf. Process. Syst.*, 2000, pp. 666–672.
- [119] F. Gustafsson, "Particle filter theory and practice with positioning applications," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 25, no. 7, pp. 53–82, Jul. 2010.
- [120] R. Krishnan, U. Shalit, and D. Sontag, "Deep Kalman filters," 2015, *arXiv:1511.05121*.
- [121] R. Krishnan, U. Shalit, and D. Sontag, "Structured inference networks for nonlinear state space models," in *Proc. AAAI Conf. Artif. Intell.*, 2017, pp. 2101–2109.



Tales Imbiriba (Member, IEEE) received his doctorate degree in electrical engineering from the Department of Electrical Engineering (DEE), the Federal University of Santa Catarina (UFSC), Florianópolis, Brazil, in 2016.

He was a Postdoctoral Researcher with the DEE–UFSC (2017–2019) and with the ECE Department of the NU (2019–2021). He is currently an Associate Research Professor with the ECE Department and a Senior Research Scientist with the Institute for Experiential AI, both with the Northeastern University (NU), Boston, MA, USA. His research interests include audio and image processing, pattern recognition, kernel methods, adaptive filtering, and Bayesian inference.



Ondřej Straka (Member, IEEE) received the master's degree in cybernetics and control engineering and a Ph.D. degree in cybernetics from the University of West Bohemia, Pilsen, Czech Republic, in 1998 and 2004, respectively.

He is currently 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), and NTIS–New Technologies for the Information Society. He has participated in a number of

projects of fundamental research and in several projects of applied research. He was involved in the development of several software frameworks for nonlinear state estimation and system identification. He has authored or coauthored more than 140 journal and conference papers. His research interests include local and global nonlinear state estimation methods, system identification, performance evaluation, and fault detection.

Dr. Straka was the recipient of the Werner von Siemens Excellence Award in 2014 for the most important result in basic research. He has been actively involved in organizing committees of a number of conferences such as ACD (2015), MFI (2020–2022), or FUSION (2016 and 2023).



Jindřich Duník (Senior Member, IEEE) received the Ing. (M.Sc.) and Ph.D. degrees in automatic control in 2003 and 2008, respectively, both from the University of West Bohemia (UWB), Pilsen, Czech Republic.

He is an Associate Professor with the Department of Cybernetics, UWB, focusing on state estimation and system identification, and a Senior Scientist with Honeywell International, Aerospace Advanced Technology Europe. He has authored or coauthored more than 80 technical papers (both journal and conference) and patents devoted to nonlinear filtering, system identification, and navigation. His research interests include inertial and satellite-based navigation systems and integrity monitoring methods.

Dr. Duník was the recipient of Honeywell Technology Achievement Awards, Werner von Siemens Excellence Award, and IEEE AESS Harry Rowe Mimno Award. He is a Member of AESS Navigation System Panel.



Pau Closas (Senior Member, IEEE) received the M.S. and Ph.D. degrees in electrical engineering from Universitat Politècnica de Catalunya (UPC), Barcelona, Spain, in 2003 and 2009, respectively, and an M.S. degree in advanced maths and mathematical engineering from UPC, in 2014.

He is an Associate Professor in electrical and computer engineering with Northeastern University, Boston, MA, USA. His research interests include statistical signal processing, stochastic

filtering, robust filtering, and machine learning, with applications to positioning and localization systems.

Dr. Closas was the recipient of the EURASIP Best Ph.D. Thesis Award in 2014, the 9th Duran Farell Award for Technology Research, the 2016 ION's Early Achievements Award, 2019 NSF CAREER Award, and the IEEE AESS Harry Rowe Mimno Award in 2022. He volunteered in editorial roles (e.g., *Navigation*, Proc. IEEE, IEEE TRANSACTIONS VEHICULAR TECHNOLOGY, and IEEE SIGNAL PROCESSING MAGAZINE), and has been actively involved in organizing committees of a number of conferences, such as EUSIPCO (2011, 2019–2022), IEEE SSP'16, IEEE/ION PLANS (2020, 2023), or IEEE ICASSP'20.