December 2023 Volume 18 Number 2 ISSN 1557-6418

JAIF 23

Journal of Advances in Information Fusion

A semi-annual archival publication of the International Society of Information Fusion

Regular Papers Page

Guest Editorial:

Foreword to the Special Issue on Graph-Based Localization and Tracking

www.isif.org

ISIF

INTERNATIONAL SOCIETY OF INFORMATION FUSION

The International Society of Information Fusion (ISIF) is the premier professional society and global information resource for multidisciplinary approaches for theoretical and applied INFORMATION FUSION technologies. Technical areas of interest include target tracking, detection theory, applications for information fusion methods, image fusion, fusion systems architectures and management issues, classification, learning, data mining, Bayesian and reasoning methods.

Manuscripts are submitted at http://jaif.msubmit.net. If in doubt about the proper editorial area of a contribution, submit it under the unknown area.

Journal of Advances in Information Fusion (ISSN 1557-6418) is published semi-annually by the International Society of Information Fusion. The responsibility for the contents rests upon the authors and not upon ISIF, the Society, or its members. ISIF is a California Nonprofit Public Benefit Corporation at P.O. Box 4631, Mountain View, California 94040.

Guest Editorial: Foreword to the Special Issue on Graph-Based Localization and Tracking

Domenico Gaglione

Erik Leitinger

Jason L. Williams

Florian Meyer

The advent of graph-based processing in information fusion has offered a theoretical framework and a comprehensive toolkit for modeling the intricate statistical structures inherent in data fusion problems and has provided efficient and modular algorithmic solutions to high-dimensional problems, creating new benchmarks for performance, scalability, and flexibility. Graph-based methods have rapidly evolved to address the complex challenges of heterogeneous sensing environments, and today, they are at the forefront of pioneering solutions in applications as diverse as autonomous navigation, ocean sciences, asset tracking, future communications, and the burgeoning Internet of Things.

The papers featured in this special issue on *Graph-Based Localization and Tracking* underscore the versatility of graph-based approaches to overcome challenges posed by the non-Gaussian uncertainties of inexpensive, low-power sensing devices, often manifested as missed detections, false positives, and measurements of uncertain origin.

We commence with a paper on multipath-based simultaneous localization and mapping (SLAM), presenting a novel Bayesian particle-based sum-product algorithm (SPA) that can be interpreted as passing messages on a graphical model that adeptly fuses multiple measurements per virtual anchor, enhancing robustness in challenging indoor propagation environments.

Our journey into the underwater realm showcases a graph-based mapping algorithm implemented by autonomous underwater vehicles in mine countermeasure operations. The SPA's application in this domain demonstrates the potential of graph-based Bayesian inference for object detection and estimation in a challenging underwater environment.

We also delve into a comparative study between a traditional Joint Integrated Probabilistic Data Association filter incorporating target-provided measurements and a multitarget tracking approach derived using a probabilistic graphical model. This work provides critical insights into the performance trade-offs in scenarios with closely spaced targets and with targets executing sharp maneuvers.

Lastly, the special issue introduces an integrated learn-then-optimize framework for condition-based predictive maintenance scheduling. This fusion of deep learning and optimization underscores the transformative power of graph-based methods in predictive maintenance models, surpassing traditional methods in ensuring fleet availability and cost-effectiveness.

We invite our readers to immerse themselves in the insights provided by these studies, which shed light on the current state of graph-based localization and tracking and suggest avenues for future research.

Guest Editors:

Domenico Gaglione
Centre for Maritime Research and Experimentation
E-mail: domenico.gaglione@cmre.nato.int

Erik Leitinger
Graz University of Technology
E-mail: erik.leitinger@tugraz.at

Jason L. Williams
Commonwealth Scientific and Industrial Research Org.
E-mail: jason.williams@data61.csiro.au

Florian Meyer
University of California San Diego
E-mail: flmeyer@ucsd.edu

Multipath-Based SLAM for Non-Ideal Reflective Surfaces Exploiting Multiple-Measurement Data Association

LUKAS WIELANDNER
ALEXANDER VENUS
THOMAS WILDING
ERIK LEITINGER

Multipath-based simultaneous localization and mapping (MP-SLAM) is a promising approach to obtain position information of transmitters and receivers as well as information regarding the propagation environments in future mobile communication systems. Usually, specular reflections of the radio signals occurring at flat surfaces are modeled by virtual anchors (VAs) that are mirror images of the physical anchors (PAs). In existing methods for MP-SLAM, each VA is assumed to generate only a single measurement. However, due to imperfections of the measurement equipment such as noncalibrated antennas or model mismatch due to roughness of the reflective surfaces, there are potentially multiple multipath components (MPCs) that are associated with one single VA. In this paper, we introduce a Bayesian particle-based sum-product algorithm (SPA) for MP-SLAM that can cope with multiple-measurements being associated to a single VA. Furthermore, we introduce a novel statistical measurement model that is strongly related to the radio signal. It introduces additional dispersion parameters into the likelihood function to capture additional MPC-related measurements. We demonstrate that the proposed MP-SLAM method can robustly fuse multiple measurements per VA based on numerical simulations.

Manuscript received March 28, 2023; revised August 24, 2023; released for publication April 4, 2024.

Refereeing of this contribution was handled by Florian Meyer.

The authors are with the Signal Processing and Speech Communication Laboratory, Graz University of Technology, Graz, Austria (e-mail: lukas.wielandner@tugraz.at; a.venus@tugraz.at; thomas.wilding@tugraz.at; erik.leitinger@tugraz.at).

This work was supported in part by the Christian Doppler Research Association; the Austrian Federal Ministry for Digital and Economic Affairs; the National Foundation for Research, Technology, and Development; and the European Union's Horizon 2020 research and innovation programme under Grant 101013425 (Project "REINDEER").

1557-6418/2023/$17.00 © 2023 JAIF

I. INTRODUCTION

Multipath-based simultaneous localization and mapping (MP-SLAM) is a promising approach to obtain position information of transmitters and receivers as well as information regarding their propagation environments in future mobile communication systems. Usually, specular reflections of radio signals at flat surfaces are modeled by virtual anchors (VAs) that are mirror images of the physical anchors (PAs) [1]–[4]. The positions of these VAs are unknown. MP-SLAM algorithms can detect and localize VAs and jointly estimate the time-varying position of mobile agents [3]–[5]. The availability of VA location information makes it possible to leverage multiple propagation paths of radio signals for agent localization and can thus significantly improve localization accuracy and robustness. In nonideal scenarios with rough reflective surfaces [6], [7] and limitations in the measurement equipment, such as noncalibrated antennas [8], those standard methods are prone to fail since multiple measurements can originate from the same PA or VA. This shows the need for developing new methods to cope with these limitations.

A. State of the Art

The proposed algorithm follows the feature-based SLAM approach [9], [10], i.e., the map is represented by an unknown number of *features*, whose unknown positions are estimated in a sequential (time-recursive) manner. Existing MP-SLAM algorithms consider VAs [3], [4], [11]–[13] or master VAs (MVAs) [14]–[16] as features to be mapped. Most of these methods use estimated parameters related to multipath components (MPCs) contained in the radio signal, such as distances (which are proportional to delays), angle of arrivals (AOAs), or angle of departures (AODs) [17]. These parameters are estimated from the signal in a preprocessing stage [17]–[23] and are used as "measurements" available to the MP-SLAM algorithm. A complicating factor in feature-based SLAM is measurement origin uncertainty, i.e., the unknown association of measurements with features [3], [4], [11], [22], [24]. In particular, (i) it is not known which map feature was generated by which measurement, (ii) there are missed detections due to low signal-to-noise ratio (SNR) or occlusion of features, and (iii) there are false positive measurements due to clutter. Thus, an important aspect of MP-SLAM is *data association* between these measurements and the VAs or the MVAs. Probabilistic data association can increase the robustness and accuracy of MP-SLAM but introduce additional unknown parameters. State-of-the-art methods for MP-SLAM are Bayesian estimators that perform the sum-product algorithm (SPA) on a factor graph [3], [4], [11] to avoid the curse of dimensionality related to the high-dimensional estimation problems.

In these existing methods for MP-SLAM, each feature is assumed to generate only a single measurement [25], [26]. However, due to imperfections in the measurement equipment or model mismatch due to nonideal reflective surfaces (such as rough surfaces characterized by diffuse multipath [6], [7]), there are potentially multiple MPCs that need to be associated with a single feature (VAs or MVAs) to accurately represent the environment. This is related to the multiple-measurement-to-object data association in extended object tracking (EOT) [24], [27]–[29]. In EOT, the point object assumption is no longer valid; hence, one single object can potentially generate more than one measurement, resulting in a particularly challenging data association due to the large number of possible association events [28], [30], [31]. In [24], [29], an innovative approach to this multiple-measurements-to-object data association problem is presented. It is based on the framework of graphical models [32]. In particular, an SPA was proposed with computational complexity that scales only quadratically in the number of objects and the number of measurements, avoiding suboptimal clustering of spatially close measurements.

B. Contributions

In this paper, we introduce a Bayesian particle-based SPA for MP-SLAM that can cope with multiple-measurements associated with a single VA. The proposed method is based on a factor graph designed for scalable probabilistic multiple-measurement-to-feature association proposed in [24], [29]. We also introduce a novel statistical measurement model that is strongly related to the radio signal. It introduces additional dispersion parameters into the likelihood function to capture additional MPC-related measurements. The key contributions of this paper are as follows.

1) We introduce the multiple-measurement-to-feature data association proposed in [24] to MP-SLAM [3], [11].
2) We use this multiple-measurement data association to incorporate additional MPC-related measurements originating from nonideal effects such as rough reflective surfaces or noncalibrated antennas.
3) We introduce a novel likelihood function model that is augmented with dispersion parameters to capture these additional MPC-related measurements that are associated with a single VA.
4) We demonstrate based on synthetically generated measurements that the proposed MP-SLAM method robustly associates multiple measurements per VA and that it is able to significantly outperform state-of-the-art MP-SLAM methods [3], [11] in case additional MPC-related measurements occur.

This paper advances over the preliminary account of our method provided in the conference publication [33] by (i) presenting a detailed derivation of the factor graph, (ii) providing additional simulation results, and (iii) demonstrating performance advantages compared to the classical MP-SLAM [3], [11].

C. Notation

Random variables are displayed in sans serif, upright fonts; their realizations in serif, italic fonts. Vectors and matrices are denoted by bold lowercase and uppercase letters, respectively. For example, a random variable and its realization are denoted by x and x, respectively, and a random vector and its realization by $\mathbf{x}$ and $\boldsymbol{x}$, respectively. Furthermore, $\|\boldsymbol{x}\|$ and $\boldsymbol{x}^{\mathrm{T}}$ denote the Euclidean norm and the transpose of vector $\boldsymbol{x}$, respectively; $\propto$ indicates equality up to a normalization factor; $f(\boldsymbol{x})$ denotes the probability density function (PDF) of random vector $\mathbf{x}$ (this is a short notation for $f_{\mathbf{x}}(\boldsymbol{x})$); $f(\boldsymbol{x}|\boldsymbol{y})$ denotes the conditional PDF of random vector $\mathbf{x}$ conditioned on random vector $\mathbf{y}$ (this is a short notation for $f_{\mathbf{x}|\mathbf{y}}(\boldsymbol{x}|\boldsymbol{y})$). The cardinality of a set $\mathcal{X}$ is denoted as $|\mathcal{X}|$. $\delta(\cdot)$ denotes the Dirac delta function. Furthermore, $1_{\mathbb{A}}(\boldsymbol{x})$ denotes the indicator function, that is, $1_{\mathbb{A}}(\boldsymbol{x}) = 1$ if $\boldsymbol{x} \in \mathbb{A}$ and 0 otherwise, for $\mathbb{A}$ being an arbitrary set and $\mathbb{R}^{+}$ being the set of positive real numbers. Finally, δ_e denotes the indicator function of the event $e = 0$ (i.e., $\delta_e = 1$ if $e = 0$ and 0 otherwise). We define the following PDFs with respect to x: The Gaussian PDF is

$$f_{\mathrm{N}}(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-(x-\mu)^2}{2\sigma^2}}, \tag{1}$$

with mean μ and standard deviation σ [34]. The truncated Rician PDF is [35, Ch. 1.6.7]

$$f_{\mathrm{TRice}}(x; s, u, \lambda) = \frac{1}{Q_1(\frac{u}{s}, \frac{\lambda}{s})} \frac{x}{s^2} e^{\frac{-(x^2+u^2)}{2s^2}} I_0\left(\frac{x\,u}{s^2}\right) 1_{\mathbb{R}^{+}}(x-\lambda), \tag{2}$$

with noncentrality parameter u, scale parameter s, and truncation threshold λ. $I_0(\cdot)$ is the zeroth-order modified first-kind Bessel function and $Q_1(\cdot,\cdot)$ denotes the Marcum Q-function [34]. The truncated Rayleigh PDF is [35, Ch. 1.6.7]

$$f_{\mathrm{TRayl}}(x; s, \lambda) = \frac{x}{s^2} e^{\frac{-(x^2-\lambda^2)}{2s^2}} 1_{\mathbb{R}^{+}}(x-\lambda), \tag{3}$$

with scale parameter s and truncation threshold λ. This formula corresponds to the so-called Swerling I model [35]. The Gamma PDF is denoted as

$$\mathcal{G}(x; \alpha, \beta) = \frac{1}{\beta^{\alpha}\Gamma(\alpha)} x^{k-1} e^{-\frac{x}{\beta}}, \tag{4}$$

where α is the shape parameter, β is the scale parameter, and $\Gamma(\cdot)$ is the gamma-function. Finally, we define the uniform PDF $f_{\mathrm{U}}(x; a, b) = 1/(b-a)1_{[a,b]}(x)$.

II. GEOMETRICAL RELATIONS

At each time n, we consider a mobile agent at position $\boldsymbol{p}_n$ equipped with a single antenna and J base stations, called PAs, equipped with a single antenna and

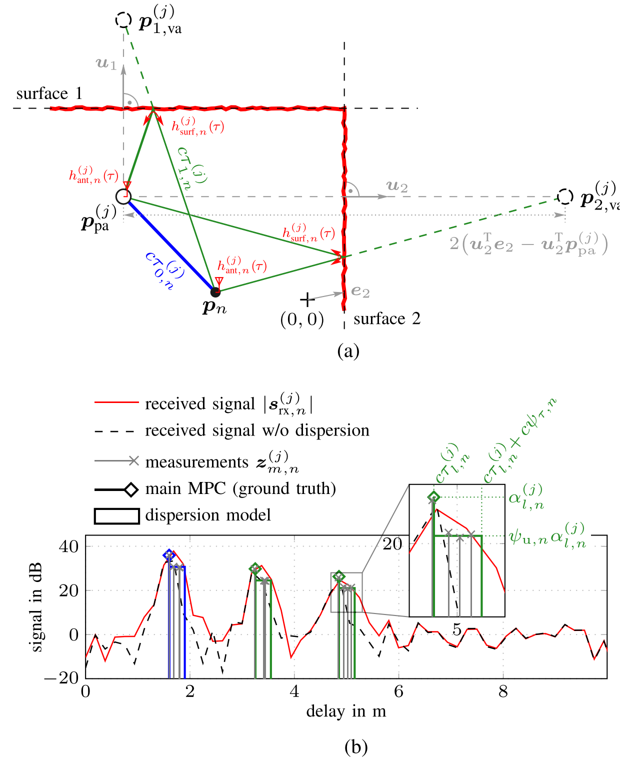


Figure 1. Exemplary indoor environment (a) and representative realization of a received signal (b). The floor plan in (a) includes an agent at position $\boldsymbol{p}_n$, a PA at position $\boldsymbol{p}_{\mathrm{pa}}^{(j)}$, and two VAs at positions $\boldsymbol{p}_{l,\mathrm{va}}^{(j)}$ for corresponding surfaces. The signal shown in (b) is received by PA at position $\boldsymbol{p}_{\mathrm{pa}}^{(j)}$. Nonideal antennas or reflective surfaces as indicated in (a) by generic impulse responses $h_{\mathrm{ant},n}^{(j)}(\tau)$ and $h_{\mathrm{surf},n}^{(j)}(\tau)$ lead to the received signal $\mathbf{s}_{\mathrm{rx},n}^{(j)}$ shown in (b) (c.f. received signal without dispersion). Resulting measurements (MPC parameter estimates) $\boldsymbol{z}_{m,n}^{(j)}$ are indicated in the received signal $\mathbf{s}_{\mathrm{rx},n}^{(j)}$ shown in (b) alongside the proposed dispersion model.

at known positions $\boldsymbol{p}_{\mathrm{pa}}^{(j)} = [p_{1,\mathrm{pa}}^{(j)}\ p_{2,\mathrm{pa}}^{(j)}]^{\mathrm{T}} \in \mathbb{R}^2$, $j \in \{1,\ldots,J\}$, where J is assumed to be known, in an environment described by reflective surfaces. Specular reflections of radio signals at flat surfaces are modeled by VAs that are mirror images of PAs. In particular, VA positions associated with single-bounce reflections are given by

$$\boldsymbol{p}_{l,\mathrm{va}}^{(j)} = \boldsymbol{p}_{\mathrm{pa}}^{(j)} + 2\big(\boldsymbol{u}_l^{\mathrm{T}}\boldsymbol{e}_l - \boldsymbol{u}_l^{\mathrm{T}}\boldsymbol{p}_{\mathrm{pa}}^{(j)}\big)\boldsymbol{u}_l, \tag{5}$$

where $\boldsymbol{u}_l$ is the normal vector of the according reflective surface, and $\boldsymbol{e}_l$ is an arbitrary point on this surface. The second summand in (5) represents the normal vector w.r.t. this reflective surface in direction $\boldsymbol{u}_l$ with the length of two times the distance between PA j at position $\boldsymbol{p}_{\mathrm{pa}}^{(j)}$ and the normal-point at the reflective surface, i.e., $2\big(\boldsymbol{u}_l^{\mathrm{T}}\boldsymbol{e}_l - \boldsymbol{u}_l^{\mathrm{T}}\boldsymbol{p}_{\mathrm{pa}}^{(j)}\big)$. An example is shown in Fig. 1(a). VA positions associated with multiple-bounce reflections are determined by applying (5) multiple times. The current number of *visible* VAs[1] within the scenario (associated with single-bounce and higher-order bounce reflections) is $L_n^{(j)}$ for each of the J PAs.

III. RADIO SIGNAL MODEL

At each time n, the mobile agent transmits a signal $s(t)$ from a single antenna, and each PA $j \in \{1,\ldots,J\}$ acts as a receiver having a single antenna. The received complex baseband signal at the jth PA is sampled N_{s} times with sampling frequency $f_{\mathrm{s}} = 1/T_{\mathrm{s}}$ yielding an observation period of $T = N_{\mathrm{s}}T_{\mathrm{s}}$. By stacking the samples, we obtain the discrete-time received signal vector

$$\mathbf{s}_{\mathrm{rx},n}^{(j)} = \sum_{l=1}^{L_n^{(j)}} \alpha_{l,n}^{(j)}\Bigg(\boldsymbol{s}\big(\tau_{l,n}^{(j)}\big) + \sum_{i=1}^{S_l^{(j)}} \beta_{l,i,n}^{(j)}\boldsymbol{s}\big(\tau_{l,n}^{(j)} + \nu_{l,i,n}^{(j)}\big)\Bigg) + \mathbf{w}_n^{(j)} \tag{6}$$

where $\boldsymbol{s}(\tau) \triangleq [s(-(N_{\mathrm{s}}-1)/2T_{\mathrm{s}}-\tau)\ \cdots\ s((N_{\mathrm{s}}-1)/2T_{\mathrm{s}}-\tau)]^{\mathrm{T}} \in \mathbb{C}^{N_{\mathrm{s}}\times 1}$ is the discrete-time transmit pulse. The first term contains the sum over the line-of-sight (LOS) component ($l=1$) and the $L_n^{(j)}-1$ specular MPCs (for $l \in \{2,\ldots,L_n^{(j)}\}$) termed main components. The lth main-component is characterized by its complex amplitude $\alpha_{l,n}^{(j)} \in \mathbb{C}$ and its delays $\tau_{l,n}^{(j)}$. The second term contains the sum over $S_l^{(j)}$ additional sub-components characterized by complex amplitudes $\alpha_{l,n}^{(j)}\beta_{l,i,n}^{(j)}$ and by (relative) delays $\tau_{l,n}^{(j)} + \nu_{l,i,n}^{(j)}$, where $\nu_{l,i,n}^{(j)}$ is the excess delay and $\beta_{l,i,n}^{(j)} \in \mathbb{R}$ is a relative dampening variable. The delays $\tau_{l,n}^{(j)}$ are proportional to the distances (ranges) between the agent and either the jth PA (for $l=1$) or the corresponding VAs (for $l \in \{2,\ldots,L_n^{(j)}\}$). That is $\tau_{1,n}^{(j)} = \|\boldsymbol{p}_n - \boldsymbol{p}_{\mathrm{pa}}^{(j)}\|/c$ and $\tau_{l,n}^{(j)} = \|\boldsymbol{p}_n - \boldsymbol{p}_{l,\mathrm{va}}^{(j)}\|/c$ for $l \in \{2,\ldots,L_n^{(j)}\}$, where c is the speed of light. The measurement noise vector $\mathbf{w}_n^{(j)} \in \mathbb{C}^{N_{\mathrm{s}}\times 1}$ is a zero-mean, circularly-symmetric complex Gaussian random vector with covariance matrix $\sigma^{(j)2}\boldsymbol{I}_{N_{\mathrm{s}}}$ and noise variance $\sigma^{(j)2} = N_0^{(j)}/T_{\mathrm{s}}$. The component SNR of MPC l is $\mathrm{SNR}_{l,n}^{(j)} = |\alpha_{l,n}^{(j)}|^2\|\boldsymbol{s}(\tau_{l,n}^{(j)})\|^2/\sigma^{(j)2}$. The component SNR of the subcomponents is given as $\mathrm{SNR}_{l,i,n}^{(j)} = \beta_{l,i,n}^{(j)2}\mathrm{SNR}_{l,n}^{(j)}$. The corresponding normalized amplitude is $u_{l,n}^{(j)} \triangleq \mathrm{SNR}_{l,n}^{(j)\frac{1}{2}}$ and $u_{l,i,n}^{(j)} \triangleq \mathrm{SNR}_{l,i,n}^{(j)\frac{1}{2}}$, respectively. Details about the signal model given in (6) are provided in Appendix A.

A. Signal Model Assumptions

To capture effects such as noncalibrated antennas [22, Section VII-C], the scattering from a user-body [36], [37], rural environments [38], [39] as well as nonideal reflective surfaces [6], we introduce the dispersion parameters $\psi_{\tau,l,n}^{(j)}$ and $\psi_{\mathrm{u},l,n}^{(j)}$. In this work, we assume the *fol-*

[1] A VA does not exist at time n, when the reflective surface corresponding to this VA is obstructed with respect to the agent.

lowing restrictions to this model: (i) the additional subcomponents with excess delays $\nu_{l,i,n}^{(j)} \in [0, \psi_{\tau,l,n}^{(j)}]$ after each MPC l have the same support, i.e., $\psi_{\tau,l,n}^{(j)} \triangleq \psi_{\tau,n}$ and (ii) the corresponding dampening variables are constant $\beta_{l,i,n}^{(j)} \triangleq \psi_{\mathrm{u},l,n}^{(j)}$ with the same value for each MPC l, i.e., $\psi_{\mathrm{u},l,n}^{(j)} \triangleq \psi_{\mathrm{u},n}$. This model can be applied to ultra-wideband systems with noncalibrated antennas [22, Section VII-C] that introduce delay dispersion or to environments containing moderate nonideal reflective surfaces [6], [7] that are approximately similar in behavior and do not change significantly over the explored area. An exemplary signal as well as the dispersion model is shown in Fig. 1(b).[2]

B. Parametric Channel Estimation

By applying at each time n, a channel estimation and detection algorithm (CEDA) [18]–[23] to the observed discrete signal vector $\boldsymbol{s}_{\mathrm{rx},n}^{(j)}$, one obtains, for each anchor j, a number of $M_n^{(j)}$ measurements denoted by $\boldsymbol{z}_{m,n}^{(j)}$ with $m \in \mathcal{M}_n^{(j)} \triangleq \{1, \ldots, M_n^{(j)}\}$. Each $\boldsymbol{z}_{m,n}^{(j)} = [z_{\tau m,n}^{(j)} \; z_{\mathrm{u}m,n}^{(j)}]^{\mathrm{T}}$ representing a potential MPC parameter estimate, contains a delay measurement $z_{\tau m,n}^{(j)} \in [0, \tau_{\max}]$ and a normalized amplitude measurement $z_{\mathrm{u}m,n}^{(j)} \in [\gamma, \infty)$, where γ is the detection threshold. The CEDA decomposes the signal $\mathbf{s}_{\mathrm{rx},n}^{(j)}$ into individual, decorrelated components according to (6), reducing the number of dimensions (as $M_n^{(j)}$ is usually much smaller than N_{s}). It thus compresses the information contained in $\mathbf{s}_{\mathrm{rx},n}^{(j)}$ into $\boldsymbol{z}_n^{(j)} = [\boldsymbol{z}_{1,n}^{(j)\mathrm{T}} \cdots \boldsymbol{z}_{M_n^{(j)},n}^{(j)\mathrm{T}}]^{\mathrm{T}}$. The stacked vector $\boldsymbol{z}_n = [\boldsymbol{z}_n^{(1)\mathrm{T}} \cdots \boldsymbol{z}_n^{(J)\mathrm{T}}]^{\mathrm{T}}$ is used by the proposed algorithm as a noisy measurement.

IV. SYSTEM MODEL

At each time n, the state $\mathbf{x}_n = [\mathbf{p}_n^{\mathrm{T}} \, \mathbf{v}_n^{\mathrm{T}}]^{\mathrm{T}}$ of the agent consists of its position $\mathbf{p}_n$ and velocity $\mathbf{v}_n$. We also introduce the augmented agent state $\tilde{\mathbf{x}}_n = [\mathbf{x}_n^{\mathrm{T}} \, \boldsymbol{\psi}_n^{\mathrm{T}}]^{\mathrm{T}}$ that contains the dispersion parameters $\boldsymbol{\psi}_n = [\psi_{\tau,n} \; \psi_{\mathrm{u},n}]^{\mathrm{T}}$. In line with [11], [22], [26], we account for the unknown number of VAs by introducing for each PA j potential VAs (PVAs) $k \in \mathcal{K}_n^{(j)} \triangleq \{1, \ldots, K_n^{(j)}\}$. The number of PVAs $K_n^{(j)}$ is the maximum possible number of VAs of PA j that produced measurements so far [26] (i.e., $K_n^{(j)}$ increases with time). The state of PVA (j, k) is denoted as $\mathbf{y}_{k,n}^{(j)} \triangleq [\mathbf{x}_{k,n}^{(j)\mathrm{T}} \; \mathrm{r}_{k,n}^{(j)}]^{\mathrm{T}}$ with $\mathbf{x}_{k,n}^{(j)} = [\mathbf{p}_{k,\mathrm{va}}^{(j)\mathrm{T}} \; \mathrm{u}_{k,n}^{(j)}]^{\mathrm{T}}$, which includes the normalized amplitude $\mathrm{u}_{k,n}^{(j)}$ [11], [22]. The existence/nonexistence of PVA k is modeled by the existence variable $\mathrm{r}_{k,n}^{(j)} \in \{0, 1\}$ in the sense that PVA k exists if and only if $r_{k,n}^{(j)} = 1$. The PVA state is considered formally also if PVA k is nonexistent, i.e., if $r_{k,n}^{(j)} = 0$.

Since a part of the PA state is unknown, we also consider the PA itself a PVA. Hence, we distinguish between the PVA $k = 1$ that explicitly represents the PA, which is *a priori* existent and has known and fixed position $\boldsymbol{p}_{1,\mathrm{va}}^{(j)} = \boldsymbol{p}_{\mathrm{pa}}^{(j)}$, and all other PVAs $k \in \{2, \ldots, K_n^{(j)}\}$ whose existence and position are *a priori* unknown. Note that the PVAs state representing the PA still considers the normalized amplitude $\mathrm{u}_{1,n}^{(j)}$ as well as the existence variable $\mathrm{r}_{1,n}^{(j)}$. The states $\mathbf{x}_{k,n}^{(j)\mathrm{T}}$ of nonexistent PVAs are obviously irrelevant. Therefore, all PDFs defined for PVA states, $f(\boldsymbol{y}_{k,n}) = f(\boldsymbol{x}_{k,n}, r_{k,n})$, are of the form $f(\boldsymbol{x}_{k,n}^{(j)}, 0) = f_{k,n}^{(j)} f_{\mathrm{d}}(\boldsymbol{x}_{k,n}^{(j)})$, where $f_{\mathrm{d}}(\boldsymbol{x}_{k,n}^{(j)})$ is an arbitrary "dummy" PDF and $f_{k,n}^{(j)} \in [0, 1]$ is a constant. We also define the stacked vectors $\mathbf{y}_n^{(j)} \triangleq \big[\mathbf{y}_{1,n}^{(j)\mathrm{T}} \cdots \mathbf{y}_{K_n^{(j)},n}^{(j)\mathrm{T}}\big]^{\mathrm{T}}$ and $\mathbf{y}_n \triangleq \big[\mathbf{y}_n^{(1)\mathrm{T}} \cdots \mathbf{y}_n^{(J)\mathrm{T}}\big]^{\mathrm{T}}$. Note that according to the model introduced in Section III, $\boldsymbol{\psi}_n$ is common for all PVAs. However, this model can be extended to individual dispersion parameters for each PVA (see [24]).

A. State Evolution

For each PVA with state $\mathbf{y}_{k,n-1}^{(j)}$ with $k \in \mathcal{K}_{n-1}^{(j)} \triangleq \{1, \ldots, K_{n-1}^{(j)}\}$ at time $n-1$ and PA j, there is one "legacy" PVA with state $\underline{\mathbf{y}}_{k,n}^{(j)} \triangleq \big[\underline{\mathbf{x}}_{k,n}^{(j)\mathrm{T}} \; \underline{\mathrm{r}}_{k,n}^{(j)}\big]^{\mathrm{T}}$ with $k \in \mathcal{K}_{n-1}^{(j)}$ at time n and PA j. We also define the joint states $\underline{\mathbf{y}}_n^{(j)} \triangleq \big[\underline{\mathbf{y}}_{1,n}^{(j)\mathrm{T}} \cdots \underline{\mathbf{y}}_{K_{n-1}^{(j)},n}^{(j)\mathrm{T}}\big]^{\mathrm{T}}$ and $\underline{\mathbf{y}}_n \triangleq \big[\underline{\mathbf{y}}_n^{(1)\mathrm{T}} \cdots \underline{\mathbf{y}}_n^{(J)\mathrm{T}}\big]^{\mathrm{T}}$. Assuming that the augmented agent state as well as the PVA states of all PAs evolve independently across k, n, and j, the joint state-transition PDF factorizes as [3], [26]

$$
f\big(\tilde{\boldsymbol{x}}_n, \underline{\boldsymbol{y}}_n | \tilde{\boldsymbol{x}}_{n-1}, \boldsymbol{y}_{n-1}\big) = f(\boldsymbol{x}_n | \boldsymbol{x}_{n-1}) f(\boldsymbol{\psi}_n | \boldsymbol{\psi}_{n-1}) \times \prod_{j=1}^{J} \prod_{k=1}^{K_{n-1}^{(j)}} f\big(\underline{\boldsymbol{y}}_{k,n}^{(j)} | \boldsymbol{y}_{k,n-1}^{(j)}\big), \quad (7)
$$

where $f\big(\underline{\boldsymbol{y}}_{k,n}^{(j)} | \boldsymbol{y}_{k,n-1}^{(j)}\big) \triangleq f\big(\underline{\boldsymbol{x}}_{k,n}^{(j)}, \underline{r}_{k,n}^{(j)} | \boldsymbol{x}_{k,n-1}^{(j)}, r_{k,n-1}^{(j)}\big)$ is the legacy PVA state-transition PDF. If PVA did not exist at time $n-1$, i.e., $r_{k,n-1}^{(j)} = 0$, it cannot exist as a legacy PVA at time n either. Thus,

$$
f\big(\underline{\boldsymbol{x}}_{k,n}^{(j)}, r_{k,n}^{(j)} | \boldsymbol{x}_{k,n-1}^{(j)}, 0\big) = \begin{cases} f_{\mathrm{d}}\big(\underline{\boldsymbol{x}}_{k,n}^{(j)}\big), & \underline{r}_{k,n}^{(j)} = 0 \\ 0, & \underline{r}_{k,n}^{(j)} = 1. \end{cases} \quad (8)
$$

If PVA existed at time $n-1$, i.e., $r_{k,n-1}^{(j)} = 1$, it either dies, i.e., $\underline{r}_{k,n}^{(j)} = 0$, or survives, i.e., $\underline{r}_{k,n}^{(j)} = 1$ with survival probability denoted as p_{s}. If it does survive, its new state $\underline{\mathbf{y}}_{k,n}^{(j)}$ is distributed according to the state-transition PDF $f\big(\underline{\boldsymbol{x}}_{k,n}^{(j)} | \boldsymbol{x}_{k,n-1}^{(j)}\big) \triangleq \delta\big(\underline{\boldsymbol{p}}_{k,\mathrm{va}}^{(j)} - \boldsymbol{p}_{k,\mathrm{va}}^{(j)}\big) f\big(u_{k,n}^{(j)} | u_{k,n-1}^{(j)}\big)$ [3], [11].

[2]Note that the proposed algorithm can be reformulated in line with [24] to the general case with individual delay supports $\psi_{\tau,l,n}^{(j)}$ and to more complex amplitudes distributions for $\beta_{l,i,n}^{(j)}$, especially when multiple-antenna systems provide multiple MPC parameters (delay, AOA, AOD) [4], [11], [16].

Thus,

$$f(\underline{\boldsymbol{x}}_{k,n}^{(j)}, \underline{r}_{k,n}^{(j)} | \boldsymbol{x}_{k,n-1}^{(j)}, 1) = \begin{cases} (1-p_{\mathrm{s}})\, f_{\mathrm{d}}(\underline{\boldsymbol{x}}_{k,n}^{(j)}), & \underline{r}_{k,n}^{(j)} = 0 \\ p_{\mathrm{s}}\, \delta(\underline{\boldsymbol{p}}_{k,\mathrm{va}}^{(j)} - \boldsymbol{p}_{k,\mathrm{va}}^{(j)})\, f(u_{k,n}^{(j)} | u_{k,n-1}^{(j)}), & \underline{r}_{k,n}^{(j)} = 1 \end{cases}. \quad (9)$$

The agent state $\mathbf{x}_n$ with state-transition PDF $f(\boldsymbol{x}_n|\boldsymbol{x}_{n-1})$ is assumed to evolve in time according to a two-dimensional, constant velocity and stochastic acceleration model [40] (linear movement) given as $\mathbf{x}_n = \boldsymbol{A}\,\mathbf{x}_{n-1} + \boldsymbol{B}\,\mathbf{w}_n$, with the acceleration process $\mathbf{w}_n$ being independent and identically distributed (i.i.d.) across n, zero mean, and Gaussian with covariance matrix $\sigma_{\mathrm{w}}^2 \boldsymbol{I}_2$, σ_{w} is the acceleration standard deviation, and $\boldsymbol{A} \in \mathbb{R}^{4\times 4}$ and $\boldsymbol{B} \in \mathbb{R}^{4\times 2}$ are defined according to [40, p. 273], with observation period ΔT. The state-transition PDFs of the dispersion parameter states $f(\boldsymbol{\psi}_n|\boldsymbol{\psi}_{n-1}) = f(\psi_{\tau,n}|\psi_{\tau,n-1}) f(\psi_{\mathrm{u},n}|\psi_{\mathrm{u},n-1})$ are assumed to evolve independently of each other across n. Since both dispersion parameters are strictly positive and independent, we model the individual state-transition PDFs by Gamma PDFs, given by $f(\psi_{\tau,n}|\psi_{\tau,n-1}) = \mathcal{G}(\psi_{\tau,n}; q_\tau, \psi_{\tau,n-1}/q_\tau)$ and $f(\psi_{\mathrm{u},n}|\psi_{\mathrm{u},n-1}) = \mathcal{G}(\psi_{\mathrm{u},n}; q_{\mathrm{u}}, \psi_{\mathrm{u},n-1}/q_{\mathrm{u}})$, respectively, where q_τ and q_{u} represent the respective state noise parameters [24], [27]. Note that a small q implies a large state transition uncertainty. The state-transition PDF of the normalized amplitude $\underline{u}_{k,n}^{(j)}$ is modeled by a truncated Rician PDF, i.e., $f(\underline{u}_{k,n}^{(j)}|u_{k,n-1}^{(j)}) = f_{\mathrm{TRice}}(\underline{u}_{k,n}^{(j)}; \sigma_{\mathrm{u},k}, u_{k,n-1}^{(j)}, 0)$ with state noise parameter $\sigma_{\mathrm{u},k}$. The truncated Rician PDF was found to be useful for the proposed amplitude model [22] [see (12) in Section IV-B].[3]

B. Measurement Model

At each time n and for each anchor j, the CEDA provides the currently observed measurement vector $\boldsymbol{z}_n^{(j)}$, with fixed $M_n^{(j)}$, according to Section III-B. Before the measurements are observed, they are random and represented by the vector $\mathbf{z}_{m,n}^{(j)} = [\mathrm{z}_{\tau m,n}^{(j)}\, \mathrm{z}_{\mathrm{u}m,n}^{(j)}]^{\mathrm{T}}$. In line with Section III-B, we define the nested random vectors $\mathbf{z}_n^{(j)} = [\mathbf{z}_{1,n}^{(j)\mathrm{T}} \cdots \mathbf{z}_{\mathsf{M}_n^{(j)},n}^{(j)\mathrm{T}}]^{\mathrm{T}}$, with length corresponding to the random number of measurements $\mathsf{M}_n^{(j)}$, and $\mathbf{z}_n = [\mathbf{z}_n^{(1)\mathrm{T}} \cdots \mathbf{z}_n^{(J)\mathrm{T}}]^{\mathrm{T}}$. The vector containing all numbers of measurements is defined as $\mathbf{M}_n = [\mathsf{M}_n^{(1)} \cdots \mathsf{M}_n^{(J)}]^{\mathrm{T}}$.

If PVA k exists ($r_{k,n}^{(j)} = 1$), it gives rise to a random number of measurements. The mean number of measurements per (existing) PVA is modeled by a Poisson point process with mean $\mu_{\mathrm{m}}(\boldsymbol{\psi}_n, u_{k,n}^{(j)})$. The individual measurements $\mathbf{z}_{m,n}^{(j)}$ are assumed to be conditionally independent, i.e., the joint PDF of all measurements factorizes as $f(\boldsymbol{z}_n^{(j)}|M_n^{(j)}, \boldsymbol{x}_n, \nu_{k,n}^{(j)}, \beta_{k,n}^{(j)}, \boldsymbol{x}_{k,n}^{(j)}) = \prod_{m=1}^{M_n^{(j)}} f(\boldsymbol{z}_{m,n}^{(j)}|\boldsymbol{x}_n, \nu_{k,n}^{(j)}, \beta_{k,n}^{(j)}, \boldsymbol{x}_{k,n}^{(j)})$.

If $\mathbf{z}_{m,n}^{(j)}$ is generated by a PVA, i.e., it corresponds to a main-component (LOS component or MPC), we assume that the single-measurement likelihood function $f(\boldsymbol{z}_{m,n}^{(j)}|\boldsymbol{x}_n, \nu_{k,n}^{(j)}, \beta_{k,n}^{(j)}, \boldsymbol{x}_{k,n}^{(j)})$ is conditionally independent across $\mathrm{z}_{\tau m,n}^{(j)}$ and $\mathrm{z}_{\mathrm{u}m,n}^{(j)}$. Thus, it factorizes as

$$f(\boldsymbol{z}_{m,n}^{(j)}|\boldsymbol{x}_n, \nu_{k,n}^{(j)}, \beta_{k,n}^{(j)}, \boldsymbol{x}_{k,n}^{(j)}) = f(z_{\tau m,n}^{(j)}|\boldsymbol{p}_n, \nu_{k,n}^{(j)}, \beta_{k,n}^{(j)}, \boldsymbol{x}_{k,n}^{(j)}) f(z_{\mathrm{u}m,n}^{(j)}|\beta_{k,n}^{(j)}, u_{k,n}^{(j)}). \quad (10)$$

The likelihood function of the corresponding delay measurement $\mathrm{z}_{\tau m,n}^{(j)}$ is given by

$$f(z_{\tau m,n}^{(j)}|\boldsymbol{p}_n, \nu_{k,n}^{(j)}, \beta_{k,n}^{(j)}, \boldsymbol{x}_{k,n}^{(j)}) = f_{\mathrm{N}}\Big(z_{\tau m,n}^{(j)};\ \tau(\boldsymbol{p}_{k,\mathrm{va}}^{(j)}, \boldsymbol{p}_n) + \nu_{k,n}^{(j)},\ \sigma_\tau^2(\beta_{k,n}^{(j)} u_{k,n}^{(j)})\Big), \quad (11)$$

with mean $\tau(\mathbf{p}_{k,\mathrm{va}}^{(j)}, \mathbf{p}_n) + \nu_{k,n}^{(j)}$ and variance $\sigma_\tau^2(\beta_{k,n}^{(j)} \mathrm{u}_{k,n}^{(j)})$, where $\tau(\mathbf{p}_{k,\mathrm{va}}^{(j)}, \mathbf{p}_n) = \|\mathbf{p}_n - \mathbf{p}_{k,\mathrm{va}}^{(j)}\|/c$. The standard deviation is determined from the Fisher information given by $\sigma_\tau^2(u) = c^2/(8\pi^2 \beta_{\mathrm{bw}}^2 u^2)$ with β_{bw} being the root-mean-squared bandwidth [42], [43] (see Section VI). The likelihood function of the corresponding normalized amplitude measurement $z_{\mathrm{u}m,n}^{(j)}$ is obtained as[4]

$$f(z_{\mathrm{u}m,n}^{(j)}|\beta_{k,n}^{(j)}, u_{k,n}^{(j)}) \triangleq f_{\mathrm{TRice}}(z_{\mathrm{u}m,n}^{(j)}; \sigma_{\mathrm{u}}(\beta_{k,n}^{(j)} u_{k,n}^{(j)}), \beta_{k,n}^{(j)} u_{k,n}^{(j)}, \gamma), \quad (12)$$

with scale parameter $\sigma_{\mathrm{u}}(\beta_{k,n}^{(j)} \mathrm{u}_{k,n}^{(j)})$, noncentrality parameter $\beta_{k,n}^{(j)} \mathrm{u}_{k,n}^{(j)}$, and detection threshold γ [22], [46]. The scale parameter is similarly determined from the Fisher information given by

$$\sigma_{\mathrm{u}}^2(u) = 1/2 + u/(4N_{\mathrm{s}}). \quad (13)$$

Note that this expression reduces to 1/2 if the additive white Gaussian noise (AWGN) variance $\sigma^{(j)2}$ is assumed to be known or N_{s} to grow indefinitely (see [22, Appendix D] for a detailed derivation). The probability of detection resulting from (12) is given by the Marcum Q-function, i.e., $p_{\mathrm{D}}(\beta_{k,n}^{(j)} \mathrm{u}_{k,n}^{(j)}) \triangleq Q_1(u/\sigma_{\mathrm{u}}(\beta_{k,n}^{(j)} u_{k,n}^{(j)}), \gamma/\sigma_{\mathrm{u}}(\beta_{k,n}^{(j)} u_{k,n}^{(j)}))$ [22], [47] (see Section I-C). Using the assumptions introduced in the Section III-A, the joint PDF of the dispersion variables can

[3] In [41], it is shown that for Swerling models I and III, a Gamma state-transition PDF represents a conjugate prior for making an analytical derivation possible.

[4] The proposed model describes the distribution of the amplitude estimates of the radio signal model given in (6) [22], [44]–[46].

be constructed as follows:

$$f(\nu_{k,n}^{(j)}, \beta_{k,n}^{(j)}|\boldsymbol{\psi}_n) = \frac{1}{2}\Big(\delta(\nu_{k,n}^{(j)})\,\delta(\beta_{k,n}^{(j)}-1) + f_{\mathrm{U}}(\nu_{k,n}^{(j)}; 0, \psi_{\tau,n})\delta(\beta_{k,n}^{(j)} - \psi_{\mathrm{u},n})\Big), \tag{14}$$

where the according delay dispersion random variable is given as $\mathsf{v}_{k,n}^{(j)} \sim f_{\mathrm{U}}(\nu_{k,n}^{(j)}; 0, \psi_{\tau,n})$ and the amplitude dispersion random variable is $\beta_{k,n}^{(j)} \sim \delta(\beta_{k,n}^{(j)} - \psi_{\mathrm{u},n})$. The PDF of a single measurement $\mathbf{z}_{m,n}^{(j)}$ can now be obtained by integrating out the dispersion variables as

$$\begin{aligned} f(\boldsymbol{z}_{m,n}^{(j)}|\tilde{\boldsymbol{x}}_n, \boldsymbol{x}_{k,n}^{(j)}) &= f(\boldsymbol{z}_{m,n}^{(j)}|\boldsymbol{x}_n, \boldsymbol{\psi}_n, \boldsymbol{x}_{k,n}^{(j)}) \\ &= \int f(\boldsymbol{z}_{m,n}^{(j)}|\boldsymbol{x}_n, \nu_{k,n}^{(j)}, \beta_{k,n}^{(j)}, \boldsymbol{x}_{k,n}^{(j)}) \\ &\qquad \times f(\nu_{k,n}^{(j)}, \beta_{k,n}^{(j)}|\boldsymbol{\psi}_n)\mathrm{d}\nu_{k,n}^{(j)}\,\mathrm{d}\beta_{k,n}^{(j)} \\ &= f(z_{\tau m,n}^{(j)}|\boldsymbol{p}_n, \boldsymbol{x}_{k,n}^{(j)}) f(z_{\mathrm{u}m,n}^{(j)}|u_{k,n}^{(j)}) \\ &\quad + f(z_{\tau m,n}^{(j)}|\boldsymbol{p}_n, \boldsymbol{\psi}_n, \boldsymbol{x}_{k,n}^{(j)}) f(z_{\mathrm{u}m,n}^{(j)}|u_{k,n}^{(j)}, \psi_{\mathrm{u},n}), \end{aligned} \tag{15}$$

with the main-component delay PDF

$$f(z_{\tau m,n}^{(j)}|\boldsymbol{p}_n, \boldsymbol{x}_{k,n}^{(j)}) = f_{\mathrm{N}}(z_{\tau m,n}^{(j)};\ \tau(\boldsymbol{p}_{k,\mathrm{va}}^{(j)}, \boldsymbol{p}_n),\ \sigma_\tau^2(u_{k,n}^{(j)})) \tag{16}$$

and the main-component amplitude PDF

$$f(z_{\mathrm{u}m,n}^{(j)}|u_{k,n}^{(j)}) = f_{\mathrm{TRice}}(z_{\mathrm{u}m,n}^{(j)}; \sigma_{\mathrm{u}}(u_{k,n}^{(j)}), u_{k,n}^{(j)}, \gamma), \tag{17}$$

as well as the additional subcomponent delay PDF

$$\begin{aligned} &f(z_{\tau m,n}^{(j)}|\boldsymbol{p}_n, \boldsymbol{\psi}_n, \boldsymbol{x}_{k,n}^{(j)}) \\ &= \frac{1}{\psi_{\tau,n}}\int_0^{\psi_{\tau,n}} f_{\mathrm{N}}\Big(z_{\tau m,n}^{(j)}; \tau(\boldsymbol{p}_{k,\mathrm{va}}^{(j)}, \boldsymbol{p}_n) + \nu_{k,n}^{(j)}, \sigma_\tau^2(\psi_{\mathrm{u},n}u_{k,n}^{(j)})\Big)\mathrm{d}\nu_{k,n}^{(j)} \\ &= \frac{1}{2\psi_{\tau,n}}\left(\mathrm{erf}\left(\frac{\tau(\boldsymbol{p}_{k,\mathrm{va}}^{(j)}, \boldsymbol{p}_n) + \psi_{\tau,n} - z_{\tau m,n}^{(j)}}{\sigma_\tau(\psi_{\mathrm{u},n}u_{k,n}^{(j)})\sqrt{2}}\right) - \mathrm{erf}\left(\frac{\tau(\boldsymbol{p}_{k,\mathrm{va}}^{(j)}, \boldsymbol{p}_n) - z_{\tau m,n}^{(j)}}{\sigma_\tau(\psi_{\mathrm{u},n}u_{k,n}^{(j)})\sqrt{2}}\right)\right) \end{aligned} \tag{18}$$

and the additional subcomponent amplitude PDF

$$f(z_{\mathrm{u}m,n}^{(j)}|\psi_{\mathrm{u},n}, u_{k,n}^{(j)}) = f_{\mathrm{TRice}}(z_{\mathrm{u}m,n}^{(j)}; \sigma_{\mathrm{u}}(\psi_{\mathrm{u},n}u_{k,n}^{(j)}), \psi_{\mathrm{u},n}u_{k,n}^{(j)}, \gamma). \tag{19}$$

The according probability of detection is given as $p_{\mathrm{D}}(\mathrm{u}_{k,n}^{(j)})$ for the main-component of each PVA or $p_{\mathrm{D}}(\psi_{\mathrm{u},n}\mathrm{u}_{k,n}^{(j)})$ for the additional subcomponents, respectively.

It is also possible that a measurement $\mathbf{z}_{m,n}^{(j)}$ did not originate from any PVA (*false alarm*). False alarm measurements originating from the CEDA are assumed statistically independent of PVA states. They are modeled by a Poisson point process with mean μ_{fa} and PDF $f_{\mathrm{fa}}(\boldsymbol{z}_{m,n}^{(j)})$, which is assumed to factorize as $f_{\mathrm{fa}}(\boldsymbol{z}_{m,n}^{(j)}) = f_{\mathrm{fa}}(z_{\tau m,n}^{(j)}) f_{\mathrm{fa}}(z_{\mathrm{u}m,n}^{(j)})$. The false alarm PDF for a single delay measurement is assumed to be uniformly distributed as $f_{\mathrm{fa}}(z_{\tau m,n}^{(j)}) = f_{\mathrm{U}}(z_{\tau m,n}^{(j)}; 0, \tau_{\max})$. In correspondence to (12), the false alarm likelihood function of the normalized amplitude measurement is given as $f_{\mathrm{fa}}(z_{\mathrm{u}m,n}^{(j)}) \triangleq f_{\mathrm{TRayl}}(z_{\mathrm{u}m,n}^{(j)}; \sqrt{1/2}, \gamma)$ with the scale parameter, given as $\sqrt{1/2}$ and detection threshold γ.

Considering the measurement model for the normalized amplitudes in (12), the mean number of PVA-related measurements $\mu_{\mathrm{m}}(\tilde{\mathbf{x}}_n, \mathbf{x}_{k,n}^{(j)}) \triangleq \mu_{\mathrm{m}}(\boldsymbol{\psi}_n, \mathrm{u}_{k,n}^{(j)})$ is well approximated as

$$\mu_{\mathrm{m}}(\boldsymbol{\psi}_n, \mathrm{u}_{k,n}^{(j)}) = p_{\mathrm{D}}(\mathrm{u}_{k,n}^{(j)}) + \frac{N_{\mathrm{ny}}\,\psi_{\tau,n}}{c\,T_{\mathrm{s}}}\,p_{\mathrm{D}}(\psi_{\mathrm{u},n}\mathrm{u}_{k,n}^{(j)}) \tag{20}$$

The right-hand side fraction denotes the average number of additional subcomponents estimated by the CEDA at a detection threshold of $\gamma = 0\,\mathrm{dB}$, where we assume an average of N_{ny} components to be detected within one Nyquist sample. Accordingly, the mean number of false alarms is approximated as $\mu_{\mathrm{fa}} = N_{\mathrm{ny}}N_{\mathrm{s}}\,e^{-\gamma^2}$ with $e^{-\gamma^2} = \int_\gamma^\infty f_{\mathrm{fa}}(z_{\mathrm{u}m,n}^{(j)})\,\mathrm{d}z_{\mathrm{u}m,n}^{(j)}$ denoting the false alarm probability.

C. New PVAs

Newly detected PVAs, i.e., actual VAs that generate a measurement for the first time, are modeled by a Poisson point process with mean μ_{n} and PDF $f_{\mathrm{n}}(\overline{\boldsymbol{x}}_{m,n}^{(j)}|\tilde{\boldsymbol{x}}_n)$. Following [3], [26], newly detected VAs are represented by new PVA states $\overline{\mathbf{y}}_{m,n}^{(j)}, m \in \{1, \ldots, M_n^{(j)}\}$, where each new PVA state corresponds to a measurement $\mathbf{z}_{m,n}^{(j)}$; $\overline{r}_{m,n}^{(j)} = 1$ implies that measurement $\mathbf{z}_{m,n}^{(j)}$ was generated by a newly detected VA. Since newly detected VAs can potentially produce more than one measurement, we use the multiple-measurement-to-feature probabilistic data association and define this mapping as introduced in [24], [29]. We also introduce the joint states $\overline{\mathbf{y}}_n^{(j)} \triangleq [\overline{\mathbf{y}}_{1,n}^{(j)\mathrm{T}} \cdots \overline{\mathbf{y}}_{M_n^{(j)},n}^{(j)\mathrm{T}}]^{\mathrm{T}}$ and $\overline{\mathbf{y}}_n \triangleq [\overline{\mathbf{y}}_n^{(1)\mathrm{T}} \cdots \overline{\mathbf{y}}_n^{(J)\mathrm{T}}]^{\mathrm{T}}$. The vector of all PVAs at time n is given by $\mathbf{y}_n \triangleq [\underline{\mathbf{y}}_n^{\mathrm{T}}\ \overline{\mathbf{y}}_n^{\mathrm{T}}]^{\mathrm{T}}$. Note that the total number of PVAs per PA is given by $K_n^{(j)} = K_{n-1}^{(j)} + M_n^{(j)}$.

Since new PVAs are introduced as new measurements are available at each time, the number of PVAs grows indefinitely. Thus, for feasible methods, a suboptimal pruning step is employed that removes unlikely PVAs (see Section IV-F).

D. Association Vectors

For each PA, measurements $\boldsymbol{z}_{m,n}^{(j)}$ are subject to a data association uncertainty. It is not known which measurement $\boldsymbol{z}_{m,n}^{(j)}$ is associated with which PVA k, or if a measurement $\boldsymbol{z}_{m,n}^{(j)}$ did not originate from any PVA (*false alarm*) or if a PVA did not give rise to any measurement (*missed detection*). The associations between measurements $\boldsymbol{z}_{m,n}^{(j)}$ and the PVAs at time n is described by the binary PVA-orientated association variables with entries [24], [29]

$$\mathsf{a}_{km,n}^{(j)} \triangleq \begin{cases} 1, & \text{if measurement } m \text{ was generated by PVA } k \\ 0, & \text{otherwise}. \end{cases}$$

We distinguish between legacy and new PVA-associated variable vectors given, respectively, as $\underline{\mathbf{a}}_{k,n}^{(j)} \triangleq [\underline{\mathsf{a}}_{k1,n}^{(j)} \cdots \underline{\mathsf{a}}_{kM_n^{(j)},n}^{(j)}]^{\mathrm{T}}$ with $k \in \mathcal{K}_{n-1}^{(j)}$ and $\overline{\mathbf{a}}_{k,n}^{(j)} \triangleq [\overline{\mathsf{a}}_{k1,n}^{(j)} \cdots \overline{\mathsf{a}}_{kk,n}^{(j)}]^{\mathrm{T}}$ with $k \in \mathcal{M}_n^{(j)}$ and $\mathbf{a}_{k,n}^{(j)} \triangleq [\underline{\mathbf{a}}_{k,n}^{(j)\mathrm{T}} \cdots \overline{\mathbf{a}}_{k,n}^{(j)\mathrm{T}}]^{\mathrm{T}}$ [29]. We also define $\mathbf{a}_n^{(j)} \triangleq [\mathbf{a}_{1,n}^{(j)\mathrm{T}} \cdots \mathbf{a}_{K_n^{(j)},n}^{(j)\mathrm{T}}]^{\mathrm{T}}$ and $\mathbf{a}_n \triangleq [\mathbf{a}_n^{(1)\mathrm{T}} \cdots \mathbf{a}_n^{(J)\mathrm{T}}]^{\mathrm{T}}$. To reduce computational complexity, following [3], [25], [26], we use the redundant description of association variables, i.e., we introduce measurement-orientated association variable

$$\mathsf{b}_{m,n}^{(j)} \triangleq \begin{cases} k \in \{1, \ldots, K_n^{(j)}\}, & \text{if measurement } m \text{ was} \\ & \text{generated by PVA } k \\ 0, & \text{otherwise}, \end{cases}$$

and define the measurement-oriented association vector $\mathbf{b}_n^{(j)} = [\mathsf{b}_{1,n}^{(j)} \cdots \mathsf{b}_{M_n^{(j)},n}^{(j)}]$. We also define $\mathbf{b}_n \triangleq [\mathbf{b}_n^{(1)\mathrm{T}} \cdots \mathbf{b}_n^{(J)\mathrm{T}}]^{\mathrm{T}}$. Note that any data association event that can be expressed by both random vectors $\mathbf{a}_n$ and $\mathbf{b}_n$ is a valid event, i.e., any measurement can be generated by at most one PVA. This redundant representation of events makes it possible to develop scalable SPAs [3], [22], [25], [26].

E. Joint Posterior PDF

By using common assumptions [3], [22], [26], and for fixed and thus observed measurements $\boldsymbol{z}_{1:n}$, it can be shown that the joint posterior PDF of $\tilde{\mathbf{x}}_{1:n}$ ($\tilde{\mathbf{x}}_{1:n} \triangleq [\tilde{\mathbf{x}}_1^{\mathrm{T}} \cdots \tilde{\mathbf{x}}_n^{\mathrm{T}}]^{\mathrm{T}}$), $\mathbf{y}_{1:n}$, $\mathbf{a}_{1:n}$, and $\mathbf{b}_{1:n}$, conditioned on $\boldsymbol{z}_{1:n}$ for all time steps $n' \in \{1, \ldots, n\}$ is given by

$$\begin{aligned} & f(\tilde{\boldsymbol{x}}_{1:n}, \boldsymbol{y}_{1:n}, \boldsymbol{a}_{1:n}, \boldsymbol{b}_{1:n} | \boldsymbol{z}_{1:n}) \\ & \propto f(\boldsymbol{x}_1) f(\boldsymbol{\psi}_1) \Bigg(\prod_{j'=1}^{J} \prod_{k'=1}^{K_1^{(j')}} f(\underline{\boldsymbol{y}}_{k',1}^{(j')}) \Bigg) \\ & \times \prod_{n'=2}^{n} f(\boldsymbol{x}_{n'} | \boldsymbol{x}_{n'-1}) f(\boldsymbol{\psi}_{n'} | \boldsymbol{\psi}_{n'-1}) \\ & \times \prod_{j=1}^{J} \Bigg(\prod_{k=1}^{K_{n'-1}^{(j)}} g(\underline{\boldsymbol{y}}_{k,n'}^{(j)} | \boldsymbol{y}_{k,n'-1}^{(j)}, \tilde{\boldsymbol{x}}_{n'-1}) \\ & \times \prod_{m'=1}^{M_{n'}^{(j)}} q(\tilde{\boldsymbol{x}}_{n'}, \underline{\boldsymbol{y}}_{k,n'}^{(j)}, \underline{a}_{km',n'}^{(j)}; \boldsymbol{z}_{m',n'}^{(j)}) \underline{\Psi}(\underline{a}_{km',n'}^{(j)}, b_{m',n'}^{(j)}) \Bigg) \\ & \times \Bigg(\prod_{m=1}^{M_{n'}^{(j)}} v(\tilde{\boldsymbol{x}}_{n'}, \overline{\boldsymbol{y}}_{m,n'}^{(j)}, \overline{a}_{mm,n'}^{(j)}; \boldsymbol{z}_{m,n'}^{(j)}) \\ & \times \prod_{h=1}^{m-1} u(\tilde{\boldsymbol{x}}_{n'}, \overline{\boldsymbol{y}}_{m,n'}^{(j)}, \overline{a}_{mh,n'}^{(j)}; \boldsymbol{z}_{h,n'}^{(j)}) \overline{\Psi}(\overline{a}_{mh,n'}^{(j)}, b_{h,n'}^{(j)}) \Bigg), \end{aligned} \tag{21}$$

where $g(\underline{\boldsymbol{y}}_{k,n}^{(j)} | \underline{\boldsymbol{y}}_{k,n-1}^{(j)}, \tilde{\boldsymbol{x}}_{n-1})$, $q(\tilde{\boldsymbol{x}}_n, \underline{\boldsymbol{y}}_{k,n}^{(j)}, \underline{a}_{km,n}^{(j)}; \boldsymbol{z}_{m,n}^{(j)})$, $\Psi(a_{km,n}^{(j)}, b_{m,n}^{(j)})$, $u(\tilde{\boldsymbol{x}}_n, \overline{\boldsymbol{y}}_{k,n}^{(j)}, \overline{a}_{mh,n}^{(j)}; \boldsymbol{z}_{h,n}^{(j)})$ and $v(\tilde{\boldsymbol{x}}_n, \overline{\boldsymbol{y}}_{m,n}^{(j)}, \overline{a}_{mm,n}^{(j)}; \boldsymbol{z}_{m,n}^{(j)})$ are explained in what follows. The *pseudo state-transition function* is given by

$$\begin{aligned} & g(\underline{\boldsymbol{y}}_{k,n}^{(j)} | \boldsymbol{y}_{k,n-1}^{(j)}, \tilde{\boldsymbol{x}}_{n-1}) \\ & \triangleq \begin{cases} e^{-\mu_{\mathrm{m}}\left(\tilde{\boldsymbol{x}}_{n-1}, \underline{\boldsymbol{x}}_{k,n}^{(j)}\right)} f(\underline{\boldsymbol{x}}_{k,n}^{(j)}, 1 | \underline{\boldsymbol{x}}_{k,n-1}^{(j)}, \underline{r}_{k,n-1}^{(j)}), & \underline{r}_{k,n}^{(j)} = 1 \\ f(\underline{\boldsymbol{x}}_{k,n}^{(j)}, 0 | \underline{\boldsymbol{x}}_{k,n-1}^{(j)}, \underline{r}_{k,n-1}^{(j)}), & \underline{r}_{k,n}^{(j)} = 0, \end{cases} \end{aligned} \tag{22}$$

and the *pseudo prior distribution* as

$$f(\overline{\boldsymbol{y}}_{k,n}^{(j)} | \tilde{\boldsymbol{x}}_n) \triangleq \begin{cases} \mu_n f_n(\overline{\boldsymbol{x}}_{k,n}^{(j)} | \tilde{\boldsymbol{x}}_n) e^{-\mu_{\mathrm{m}}\left(\tilde{\boldsymbol{x}}_n, \overline{\boldsymbol{x}}_{k,n}^{(j)}\right)}, & \overline{r}_{k,n}^{(j)} = 1 \\ f_d(\overline{\boldsymbol{x}}_{k,n}^{(j)}), & \overline{r}_{k,n}^{(j)} = 0. \end{cases} \tag{23}$$

The *pseudo likelihood functions* related to legacy PVAs for $k \in \mathcal{K}_{n-1}^{(j)}$ $q(\tilde{\boldsymbol{x}}_n, \underline{\boldsymbol{y}}_{k,n}^{(j)}, \underline{a}_{km,n}^{(j)}; \boldsymbol{z}_{m,n}^{(j)}) = q(\tilde{\boldsymbol{x}}_n, \underline{\boldsymbol{x}}_{k,n}^{(j)}, \underline{r}_k^{(j)}, \underline{a}_{km,n}^{(j)}; \boldsymbol{z}_{m,n}^{(j)})$ is given by

$$\begin{aligned} & q(\tilde{\boldsymbol{x}}_n, \underline{\boldsymbol{x}}_{k,n}^{(j)}, 1, \underline{a}_{km,n}^{(j)}; \boldsymbol{z}_{m,n}^{(j)}) \\ & \triangleq \begin{cases} \dfrac{\mu_{\mathrm{m}}\left(\tilde{\boldsymbol{x}}_n, \underline{\boldsymbol{x}}_{k,n}^{(j)}\right) f(\boldsymbol{z}_{m,n}^{(j)} | \boldsymbol{p}_n, \boldsymbol{\psi}_n, \underline{\boldsymbol{x}}_{k,n}^{(j)})}{\mu_{\mathrm{fa}} f_{\mathrm{fa}}(\boldsymbol{z}_{m,n}^{(j)})}, & \underline{a}_{km,n}^{(j)} = 1 \\ 1, & \underline{a}_{km,n}^{(j)} = 0 \end{cases} \end{aligned} \tag{24}$$

and $q(\tilde{\boldsymbol{x}}_n, \underline{\boldsymbol{x}}_{k,n}^{(j)}, 0, \underline{a}_{km,n}^{(j)}; \boldsymbol{z}_{m,n}^{(j)}) \triangleq \delta_{\underline{a}_{km,n}^{(j)}}$. The *pseudo likelihood functions* related to a new PVA (with $k \in \mathcal{M}_n^{(j)} \backslash m$) is given as $u(\tilde{\boldsymbol{x}}_n, \overline{\boldsymbol{y}}_{k,n}^{(j)}, \overline{a}_{km,n}^{(j)}; \boldsymbol{z}_{m,n}^{(j)}) = u(\tilde{\boldsymbol{x}}_n, \overline{\boldsymbol{x}}_{k,n}^{(j)}, \overline{r}_k^{(j)}, \overline{a}_{km,n}^{(j)}; \boldsymbol{z}_{m,n}^{(j)})$ is given by

$$\begin{aligned} & u(\tilde{\boldsymbol{x}}_n, \overline{\boldsymbol{x}}_{k,n}^{(j)}, 1, \overline{a}_{km,n}^{(j)}; \boldsymbol{z}_{m,n}^{(j)}) \\ & \triangleq \begin{cases} \dfrac{f(\overline{\boldsymbol{y}}_{k,n}^{(j)} | \tilde{\boldsymbol{x}}_n) \mu_{\mathrm{m}}\left(\tilde{\boldsymbol{x}}_n, \overline{\boldsymbol{x}}_{k,n}^{(j)}\right) f(\boldsymbol{z}_{m,n}^{(j)} | \boldsymbol{p}_n, \boldsymbol{\psi}_n, \overline{\boldsymbol{x}}_{k,n}^{(j)})}{\mu_{\mathrm{fa}} f_{\mathrm{fa}}(\boldsymbol{z}_{m,n}^{(j)})}, & \overline{a}_{km,n}^{(j)} = 1 \\ 1, & \overline{a}_{km,n}^{(j)} = 0 \end{cases} \end{aligned} \tag{25}$$

and $u(\tilde{\boldsymbol{x}}_n, \overline{\boldsymbol{x}}_{k,n}^{(j)}, 0, \overline{a}_{km,n}^{(j)}; \boldsymbol{z}_{m,n}^{(j)}) \triangleq \delta_{\overline{a}_{km,n}^{(j)}}$, whereas for $k = m$ as $v(\tilde{\boldsymbol{x}}_n, \overline{\boldsymbol{y}}_m^{(j)}, \overline{a}_{mm,n}^{(j)}; \boldsymbol{z}_{m,n}^{(j)}) = v(\tilde{\boldsymbol{x}}_n, \overline{\boldsymbol{x}}_{m,n}^{(j)}, \overline{r}_{m,n}^{(j)}, \overline{a}_{mm,n}^{(j)}; \boldsymbol{z}_{m,n}^{(j)})$ is given by

$$\begin{aligned} & v(\tilde{\boldsymbol{x}}_n, \overline{\boldsymbol{x}}_{m,n}^{(j)}, 1, \overline{a}_{mm,n}^{(j)}; \boldsymbol{z}_{m,n}^{(j)}) \\ & \triangleq \begin{cases} \dfrac{f(\overline{\boldsymbol{y}}_{m,n}^{(j)} | \tilde{\boldsymbol{x}}_n) \mu_{\mathrm{m}}\left(\tilde{\boldsymbol{x}}_n, \overline{\boldsymbol{x}}_{m,n}^{(j)}\right) f\left(\boldsymbol{z}_{m,n}^{(j)} | \boldsymbol{p}_n, \boldsymbol{\psi}_n, \overline{\boldsymbol{x}}_{m,n}^{(j)}\right)}{\mu_{\mathrm{fa}} f_{\mathrm{fa}}(\boldsymbol{z}_{m,n}^{(j)})}, & \overline{a}_{mm,n}^{(j)} = 1 \\ 0, & \overline{a}_{mm,n}^{(j)} = 0 \end{cases} \end{aligned} \tag{26}$$

and $v(\tilde{\boldsymbol{x}}_n, \overline{\boldsymbol{x}}_{m,n}^{(j)}, 0, \overline{a}_{mm,n}^{(j)}; \boldsymbol{z}_{m,n}^{(j)}) \triangleq \delta_{\overline{a}_{mm,n}^{(j)}}$.

Finally, the binary *indicator functions* that check consistency for any pair $(a_{km,n}^{(j)}, b_{m,n}^{(j)})$ of PVA-oriented and measurement-oriented association variable at time n

are, respectively, given by

$$\underline{\Psi}(\underline{a}_{km,n}^{(j)}, b_{m,n}^{(j)})$$
$$\triangleq \begin{cases} 0, \ \underline{a}_{km,n}^{(j)} = 1, \ b_{m,n}^{(j)} \neq k \text{ or } \underline{a}_{km,n}^{(j)} = 0, \ b_{m,n}^{(j)} = k \\ 1, \ \text{else} \end{cases} \tag{27}$$

for $k \in \mathcal{K}_{n-1}^{(j)}$ and

$$\overline{\Psi}(\overline{a}_{km,n}^{(j)}, b_{m,n}^{(j)}) \triangleq \begin{cases} 0, & \overline{a}_{km,n}^{(j)} = 1, b_{m,n}^{(j)} \neq K_{n-1}^{(j)} + k \\ & \text{or } \overline{a}_{km,n}^{(j)} = 0, b_{m,n}^{(j)} = K_{n-1}^{(j)} + k \\ 1, & \text{else}. \end{cases} \tag{28}$$

for $k \in \mathcal{M}_n^{(j)}$. The factor graph representing the factorization (21) is shown in Fig. 2.

F. Detection of PVAs and State Estimation

We aim to estimate all states using all available measurements $\boldsymbol{z}_{1:n}$ from all PAs up to time n. In particular, we calculate estimates of the augmented agent state (containing the dispersion parameters) $\tilde{\mathbf{x}}_n$ by using the minimum mean-square error (MMSE) estimator [48, Ch. 4], i.e.,

$$\tilde{\boldsymbol{x}}_n^{\text{MMSE}} \triangleq \int \tilde{\boldsymbol{x}}_n f(\tilde{\boldsymbol{x}}_n | \boldsymbol{z}_{1:n}) \mathrm{d}\tilde{\boldsymbol{x}}_n, \tag{29}$$

where $\tilde{\boldsymbol{x}}_n^{\text{MMSE}} = [\boldsymbol{x}_n^{\text{MMSE T}} \boldsymbol{\psi}_n^{\text{MMSE T}}]^{\text{T}}$. The map of the environment is represented by reflective surfaces described by PVAs. Therefore, the state $\boldsymbol{x}_{k,n}^{(j)}$ of the detected PVAs $k \in \{1, \ldots, K_n^{(j)}\}$ must be estimated. This relies on the marginal posterior existence probabilities $p(r_{k,n}^{(j)} = 1 | \boldsymbol{z}_{1:n}) = \int f(\boldsymbol{x}_{k,n}^{(j)}, r_{k,n}^{(j)} = 1 | \boldsymbol{z}_{1:n}^{(j)}) \mathrm{d}\boldsymbol{x}_{k,n}^{(j)}$ and the marginal posterior PDFs $f(\boldsymbol{x}_{k,n}^{(j)} | r_{k,n}^{(j)} = 1, \boldsymbol{z}_{1:n}) = f(\boldsymbol{x}_{k,n}^{(j)}, r_{k,n}^{(j)} = 1 | \boldsymbol{z}_{1:n}) / p(r_{k,n}^{(j)} = 1 | \boldsymbol{z}_{1:n})$. A PVA k is declared to exist if $p(r_{k,n}^{(j)} = 1 | \boldsymbol{z}_{1:n}) > p_{\text{cf}}$, where p_{cf} is a confirmation threshold [48, Ch. 2]. To avoid that the number of PVA states grows indefinitely, PVA states with $p(r_{k,n}^{(j)} = 1 | \boldsymbol{z}_{1:n})$ below a threshold p_{pr} are removed from the state space ("pruned"). The number $\hat{K}_n^{(j)}$ of PVA states that are considered to exist is the estimate of the total number $L_n^{(j)}$ of VAs visible at time n. For existing PVAs, an estimate of its state $\mathbf{x}_{k,n}^{(j)}$ can again be calculated by the MMSE [48, Ch. 4]

$$\boldsymbol{x}_{k,n}^{(j)\text{MMSE}} \triangleq \int \boldsymbol{x}_{k,n}^{(j)} f(\boldsymbol{x}_{k,n}^{(j)} | r_{k,n}^{(j)} = 1, \boldsymbol{z}_{1:n}) \, \mathrm{d}\boldsymbol{x}_{k,n}^{(j)}. \tag{30}$$

The calculation of $f(\tilde{\boldsymbol{x}}_n | \boldsymbol{z}_{1:n})$, $p(r_{k,n} = 1 | \boldsymbol{z})$, and $f(\boldsymbol{x}_{k,n}^{(j)} | r_{k,n}^{(j)} = 1, \boldsymbol{z}_{1:n})$ from the joint posterior $f(\tilde{\boldsymbol{x}}_{1:n}, \boldsymbol{y}_{1:n}, \boldsymbol{a}_{1:n}, \boldsymbol{b}_{1:n} | \boldsymbol{z}_{1:n})$ by direct marginalization is not feasible. By performing sequential particle-based message passing (MP) using the SPA rules [3], [11], [46], [49]–[51] on the factor graph in Fig. 2, approximations ("beliefs") $b(\tilde{\boldsymbol{x}}_n)$ and $b(\boldsymbol{y}_{k,n}^{(j)})$ of the marginal posterior PDFs $f(\tilde{\boldsymbol{x}}_n | \boldsymbol{z}_{1:n})$, $p(r_{k,n}^{(j)} = 1 | \boldsymbol{z}_{1:n})$, and $f(\boldsymbol{x}_{k,n}^{(j)} | \, r_{k,n}^{(j)} = 1, \boldsymbol{z}_{1:n})$ can be obtained in an efficient way for the agent state as well as all legacy and new PVA states.

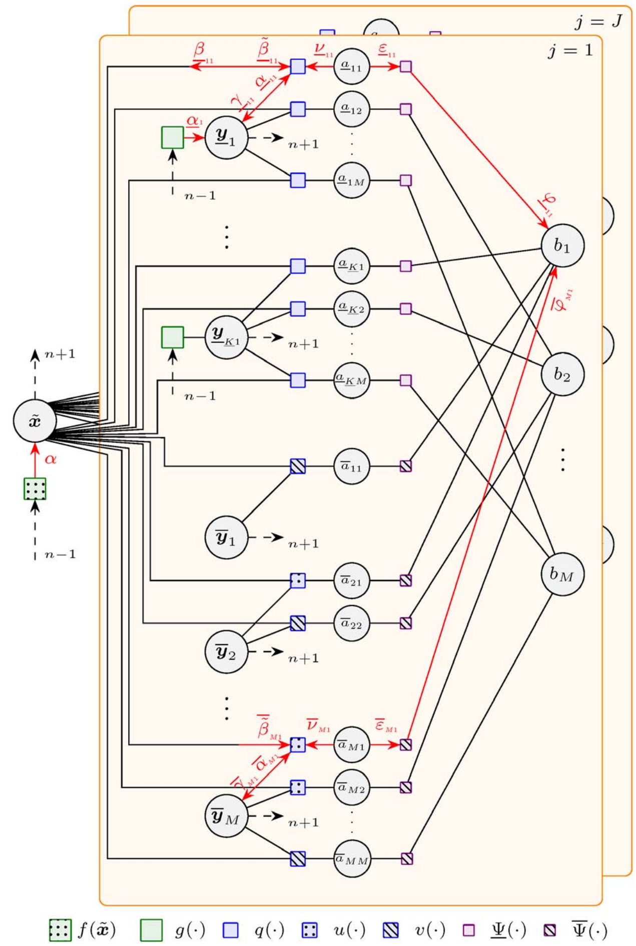


Figure 2. Factor graph for proposed algorithm. At MP iteration p, we use the following short hand notation: $f(\tilde{\boldsymbol{x}}) \triangleq f(\tilde{\boldsymbol{x}}_n | \tilde{\boldsymbol{x}}_{n-1})$, $g(\cdot)$, $q(\cdot)$, $u(\cdot)$, $v(\cdot)$, $\underline{\Psi}(\cdot)$ and $\overline{\Psi}(\cdot)$ corresponds to (22), (24), (25), (26), (27) and (28), respectively. Furthermore, we define $\alpha \triangleq \alpha(\tilde{\boldsymbol{x}}_n)$, $\underline{\alpha}_k \triangleq \alpha(\underline{\boldsymbol{x}}_{k,n}^{(j)}, \underline{r}_{k,n}^{(j)})$, $\underline{\alpha}_{kl} \triangleq \alpha_l(\underline{\boldsymbol{x}}_{k,n}^{(j)}, \underline{r}_{k,n}^{(j)})$, $\overline{\alpha}_{kl} \triangleq \alpha_l(\overline{\boldsymbol{x}}_{k,n}^{(j)}, \overline{r}_{k,n}^{(j)})$, $\underline{\varepsilon}_{kl} \triangleq \varepsilon(\underline{a}_{kl,n}^{(j)})$, $\overline{\varepsilon}_{kl} \triangleq \varepsilon(\overline{a}_{kl,n}^{(j)})$, $\underline{\gamma}_{kl} \triangleq \gamma_l(\underline{\boldsymbol{x}}_{k,n}^{(j)}, \underline{r}_{k,n}^{(j)})$, $\overline{\gamma}_{kl} \triangleq \gamma_l(\overline{\boldsymbol{x}}_{k,n}^{(j)}, \overline{r}_{k,n}^{(j)})$, $\underline{\nu}_{kl} \triangleq \underline{\nu}_{kl}(\underline{a}_{kl,n}^{(j)})$, $\overline{\nu}_{kl} \triangleq \overline{\nu}_{kl}(\overline{a}_{kl,n}^{(j)})$, $\underline{\varphi}_{kl} \triangleq \underline{\varphi}_{kl}(b_{l,n})$ and $\overline{\varphi}_{kl} \triangleq \overline{\varphi}_{kl}(b_{l,n})$. Due to our proposed scheduling, both $\tilde{\beta}_{kl}$ and $\overline{\tilde{\beta}}_{ml}$ are defined to be $\alpha(\tilde{\boldsymbol{x}}_n)$ according to (55). Furthermore, $\overline{\beta}_{ml} \triangleq 1$ and $\underline{\beta}_{kl} \triangleq \beta_{kl}^{(j)}(\tilde{\boldsymbol{x}}_n)$ since the augmented agent state is only updated with messages from legacy PVAs. The time evolution of the agent state and VAs is indicated with dashed arrows.

V. PROPOSED SPA

The factor graph in Fig. 2 has cycles, therefore we have to decide on a specific order of message computation [49], [52]. We use MP iteration with MP iteration $p \in \{1, \ldots, P\}$, where P is the maximum number of MP iterations. We choose the order according to the following rules: (i) messages are only sent forward in time; (ii) for each PA, messages are updated in parallel; (iii) along an edge connecting the augmented agent state variable node and a new PVA, messages are only sent from the former to the latter; (iv) the augmented agent state variable node is only updated at MP iteration P. The corresponding messages are shown in Fig. 2. Note that this scheduling is suboptimal since the extrinsic messages of the augmented agent state are neglected. This calculation order is solely chosen to reduce the computational demand. With these rules, the MP equations of the SPA [49] yield the following operations at each time step.

A. Prediction Step

A prediction step is performed for the augmented agent state and all legacy VAs $k \in \mathcal{K}_{n-1}^{(j)}$. It has the form of

$$\alpha(\tilde{\boldsymbol{x}}_n) = \int f(\tilde{\boldsymbol{x}}_n|\tilde{\boldsymbol{x}}_{n-1}) b(\tilde{\boldsymbol{x}}_{n-1}) \mathrm{d}\tilde{\boldsymbol{x}}_{n-1}, \tag{31}$$

$$\alpha(\underline{\boldsymbol{x}}_{k,n}^{(j)}, \underline{r}_{k,n}^{(j)}) = \sum_{r_{k,n-1}^{(j)} \in \{0,1\}} \iint g(\underline{\boldsymbol{x}}_{k,n}^{(j)}, \underline{r}_{k,n}^{(j)} | \boldsymbol{x}_{k,n-1}^{(j)}, r_{k,n-1}^{(j)}, \tilde{\boldsymbol{x}}_{n-1}) \times b(\boldsymbol{x}_{k,n-1}^{(j)}, r_{k,n-1}^{(j)}) b(\tilde{\boldsymbol{x}}_{n-1}) \mathrm{d}\boldsymbol{x}_{k,n-1}^{(j)} \mathrm{d}\tilde{\boldsymbol{x}}_{n-1} \tag{32}$$

with $b(\tilde{\boldsymbol{x}}_{n-1})$ and $b(\boldsymbol{x}_{k,n-1}^{(j)}, r_{k,n-1}^{(j)})$ denoting the beliefs of the augmented agent state and the legacy VA k calculated at the previous time step, respectively. The summation in (32), can be further written as

$$\alpha(\underline{\boldsymbol{x}}_{k,n}^{(j)}, \underline{r}_{k,n}^{(j)} = 1) = p_{\mathrm{s}} \iint e^{-\mu_m\left(\tilde{\boldsymbol{x}}_{n-1}, \underline{\mathbf{x}}_{k,n}^{(j)}\right)} f(\underline{\boldsymbol{x}}_{k,n}^{(j)}, 1 | \boldsymbol{x}_{k,n-1}^{(j)}, 1) \times b(\boldsymbol{x}_{k,n-1}^{(j)}, 1) b(\tilde{\boldsymbol{x}}_{k,n-1}^{(j)}) \mathrm{d}\boldsymbol{x}_{k,n-1}^{(j)} \mathrm{d}\tilde{\boldsymbol{x}}_{n-1}^{(j)} \tag{33}$$

and $\alpha(\underline{\boldsymbol{x}}_{k,n}^{(j)}, \underline{r}_{k,n}^{(j)} = 0) = \underline{\alpha}_k^{\mathrm{n},(j)} f_{\mathrm{d}}(\underline{\boldsymbol{x}}_{k,n}^{(j)})$ with

$$\begin{aligned}\underline{\alpha}_k^{\mathrm{n},(j)} &\triangleq \tilde{b}_{k,n-1} + (1-p_{\mathrm{s}}) \int b(\underline{\boldsymbol{x}}_{k,n-1}^{(j)}, 1) \mathrm{d}\underline{\boldsymbol{x}}_{k,n-1}^{(j)} \\ &= \tilde{b}_{k,n-1} + (1-p_{\mathrm{s}})(1-\tilde{b}_{k,n-1})\end{aligned} \tag{34}$$

where $\tilde{b}_{k,n-1} = \int b(\tilde{\boldsymbol{x}}_{k,n-1}^{(j)}, 0) \mathrm{d}\tilde{\boldsymbol{x}}_{k,n-1}^{(j)}$ approximates the probability of non-existence of legacy VA k.

B. Measurement Evaluation

The messages $\varepsilon^{[p]}(\overline{a}_{kl,n}^{(j)})$ sent from factor nodes $q(\tilde{\boldsymbol{x}}, \underline{\boldsymbol{y}}_{k,n}^{(j)}, \underline{a}_{kl,n}^{(j)}, \boldsymbol{z}_{l,n}^{(j)})$ to variable nodes $a_{kl,n}^{(j)}$ at MP iteration p with $k \in \{1, \ldots, K_{n-1}^{(j)}\}$ and $l \in \{1, \ldots, M_n^{(j)}\}$ are defined as

$$\varepsilon^{[p]}(\underline{a}_{kl,n}^{(j)}) = \iint \tilde{\beta}_{kl}^{[p]}(\tilde{\boldsymbol{x}}_n) \alpha_l^{[p]}(\underline{\boldsymbol{y}}_{k,n}^{(j)}) \times q(\tilde{\boldsymbol{x}}_n, \underline{\boldsymbol{y}}_{k,n}^{(j)}, \underline{a}_{kl,n}^{(j)}, \boldsymbol{z}_{l,n}^{(j)}) \tag{35}$$

The messages from factor nodes $u(\tilde{\boldsymbol{x}}_n, \overline{\boldsymbol{y}}_{k,n}^{(j)}, \overline{a}_{kl,n}^{(j)}, \boldsymbol{z}_{l,n}^{(j)})$ to variable nodes $\overline{a}_{kl,n}^{(j)}$ where $k \in \{1, \ldots, M_n^{(j)}\}$ and $l \in \{1, \ldots, M_n^{(j)}\} \backslash k$, are given as

$$\varepsilon^{[p]}(\overline{a}_{kl,n}^{(j)}) = \iint \tilde{\beta}_{kl}^{[p]}(\tilde{\boldsymbol{x}}_n) \alpha_l^{[p]}(\overline{\boldsymbol{y}}_{k,n}^{(j)}) \times u(\tilde{\boldsymbol{x}}_n, \overline{\boldsymbol{y}}_{k,n}^{(j)}, \overline{a}_{kl,n}^{(j)}, \boldsymbol{z}_{l,n}^{(j)}) \mathrm{d}\tilde{\boldsymbol{x}}_n \mathrm{d}\overline{\boldsymbol{y}}_{k,n}^{(j)} \tag{36}$$

and the messages from factor nodes $v(\tilde{\boldsymbol{x}}_n, \overline{\boldsymbol{y}}_{m,n}^{(j)}, \overline{a}_{mm,n}^{(j)}, \boldsymbol{z}_{m,n}^{(j)})$ to variable nodes $\overline{a}_{mm,n}^{(j)}$, $m \in \{1, \ldots, M_n^{(j)}\}$, are given as

$$\varepsilon^{[p]}(\overline{a}_{mm,n}^{(j)}) = \iint \tilde{\beta}_{mm}^{[p]}(\tilde{\boldsymbol{x}}_n) \alpha_m^{[p]}(\overline{\boldsymbol{y}}_{m,n}^{(j)}) \times v(\tilde{\boldsymbol{x}}_n, \overline{\boldsymbol{y}}_{m,n}^{(j)}, \overline{a}_{mm,n}^{(j)}, \boldsymbol{z}_{m,n}^{(j)}) \mathrm{d}\tilde{\boldsymbol{x}}_n \mathrm{d}\overline{\boldsymbol{y}}_{m,n}^{(j)}. \tag{37}$$

Note that $\alpha_l^{[p=1]}(\underline{\boldsymbol{y}}_{k,n}^{(j)}) \triangleq \alpha(\underline{\boldsymbol{x}}_{k,n}^{(j)}, \underline{r}_{k,n}^{(j)})$ and $\alpha_l^{[p=1]}(\overline{\boldsymbol{y}}_{k,n}^{(j)}) \triangleq 1$. For $p > 1, \alpha_l^{[p]}(\boldsymbol{y}_{k,n}^{(j)})$ is calculated according to Section V-E. Using (35), $\varepsilon^{[p]}(\underline{a}_{kl,n}^{[p](j)})$ is further investigated. For the messages containing information about legacy VAs, it results in

$$\varepsilon^{[p]}(\underline{a}_{kl,n}^{(j)} = 1) = \iint \tilde{\beta}_{kl}^{[p]}(\tilde{\boldsymbol{x}}_n) \alpha_l^{[p]}(\underline{\boldsymbol{x}}_{k,n}^{(j)}, \underline{r}_{k,n}^{(j)} = 1) \times \frac{\mu_{\mathrm{m}}(\tilde{\boldsymbol{x}}_n, \underline{\boldsymbol{x}}_{k,n}^{(j)}) f(\boldsymbol{z}_{l,n}^{(j)} | (\tilde{\boldsymbol{x}}_n, \underline{\boldsymbol{x}}_{k,n}^{(j)})}{\mu_{\mathrm{fa}} f_{\mathrm{fa}}(\boldsymbol{z}_{l,n}^{(j)})} \mathrm{d}\underline{\boldsymbol{x}}_{k,n}^{(j)} \mathrm{d}\tilde{\boldsymbol{x}}_n,$$

$$\varepsilon^{[p]}(\underline{a}_{kl,n}^{(j)} = 0) = \iint \tilde{\beta}_{kl}^{[p]}(\tilde{\boldsymbol{x}}_n) \Big(\alpha_l^{[p]}(\underline{\boldsymbol{x}}_{k,n}^{(j)}, \underline{r}_{k,n}^{(j)} = 1) + \alpha_l^{[p]}(\underline{\boldsymbol{x}}_{k,n}^{(j)}, \underline{r}_{k,n}^{(j)} = 0)\Big) \mathrm{d}\underline{\boldsymbol{x}}_{k,n}^{(j)} \mathrm{d}\tilde{\boldsymbol{x}}_n. \tag{38}$$

This can be further simplify by dividing both messages by $\varepsilon^{[p]}(\underline{a}_{kl,n}^{(j)} = 0)$. With an abuse of notation, it results in $\varepsilon^{[p]}(\underline{a}_{kl,n}^{(j)} = 0) = 1$.

The messages $\varepsilon^{[p]}(\overline{a}_{kl,n}^{(j)})$ can be obtained similarly by using (36) and (37), yielding

$$\varepsilon^{[p]}(\overline{a}_{kl,n}^{(j)} = 1) = \iint \tilde{\beta}_{kl}^{[p]}(\tilde{\boldsymbol{x}}_n) \alpha_l^{[p]}(\overline{\boldsymbol{x}}_{k,n}^{(j)}, \overline{r}_{k,n}^{(j)} = 1) \times \frac{f(\overline{\boldsymbol{x}}_{k,n}^{(j)} | \tilde{\boldsymbol{x}}_n) \mu_m(\tilde{\boldsymbol{x}}_n, \overline{\boldsymbol{x}}_{k,n}^{(j)}) f(\boldsymbol{z}_{l,n}^{(j)} | \tilde{\boldsymbol{x}}_n, \overline{\boldsymbol{x}}_{k,n}^{(j)})}{\mu_{\mathrm{fa}} f_{\mathrm{fa}}(\boldsymbol{z}_{l,n}^{(j)})} \mathrm{d}\overline{\boldsymbol{x}}_{k,n}^{(j)} \mathrm{d}\tilde{\boldsymbol{x}}_n \tag{39}$$

$$\varepsilon^{[p]}(\overline{a}_{kl,n}^{(j)} = 0) = \iint \tilde{\beta}_{kl}^{[p]}(\tilde{\boldsymbol{x}}_n) \Big(\alpha_l^{[p]}(\overline{\boldsymbol{x}}_{k,n}^{(j)}, \overline{r}_{k,n}^{(j)} = 1) + \alpha_l^{[p]}(\overline{\boldsymbol{x}}_{k,n}^{(j)}, \overline{r}_{k,n}^{(j)} = 0)\Big) \mathrm{d}\overline{\boldsymbol{x}}_{k,n}^{(j)} \mathrm{d}\tilde{\boldsymbol{x}}_n, \tag{40}$$

$$\varepsilon^{[p]}(\overline{a}_{mm,n}^{(j)} = 1) = \iint \tilde{\beta}_{mm}^{[p]}(\tilde{\boldsymbol{x}}_n) \alpha_m^{[p]}(\overline{\boldsymbol{x}}_{m,n}^{(j)}, \overline{r}_{m,n}^{(j)} = 1) \times \frac{f(\overline{\boldsymbol{x}}_{m,n}^{(j)} | \tilde{\boldsymbol{x}}_n) \mu_{\mathrm{m}}(\tilde{\boldsymbol{x}}_n, \overline{\boldsymbol{x}}_{m,n}^{(j)}) f(\boldsymbol{z}_{m,n}^{(j)} | \tilde{\boldsymbol{x}}_n, \overline{\boldsymbol{x}}_{m,n}^{(j)})}{\mu_{\mathrm{fa}} f_{\mathrm{fa}}(\boldsymbol{z}_{m,n}^{(j)})} \mathrm{d}\overline{\boldsymbol{x}}_{m,n}^{(j)} \mathrm{d}\tilde{\boldsymbol{x}}_n, \tag{41}$$

$$\varepsilon^{[p]}(\overline{a}^{(j)}_{mm,n}=0)=\iint \tilde{\beta}^{[p]}_{mm}(\tilde{\boldsymbol{x}}_n)\alpha^{[p]}_m(\overline{\boldsymbol{x}}^{(j)}_{m,n},\overline{r}^{(j)}_{m,n}=0)\mathrm{d}\overline{\boldsymbol{x}}^{(j)}_{m,n}\mathrm{d}\tilde{\boldsymbol{x}}_n. \tag{42}$$

The expressions can be simplified by dividing all messages by $\varepsilon(\overline{a}^{(j)}_{kl,n}=0)$. With an abuse of notation, it results in $\varepsilon(\overline{a}^{(j)}_{kl,n}=0)=1$ and

$$\varepsilon^{[p]}(\overline{a}^{(j)}_{mm,n}=0) = \frac{\iint \tilde{\beta}^{[p]}_{mm}(\tilde{\boldsymbol{x}}_n)\alpha^{[p]}_m(\overline{\boldsymbol{x}}^{(j)}_{m,n},0)\mathrm{d}\overline{\boldsymbol{x}}^{(j)}_{m,n}\mathrm{d}\tilde{\boldsymbol{x}}_n}{\iint \tilde{\beta}^{[p]}_{mm}(\tilde{\boldsymbol{x}}_n)\Big(\alpha^{[p]}_m(\overline{\boldsymbol{x}}^{(j)}_{m,n},1)+\alpha^{[p]}_m(\overline{\boldsymbol{x}}^{(j)}_{m,n},0)\Big)\mathrm{d}\overline{\boldsymbol{x}}^{(j)}_{m,n}\mathrm{d}\tilde{\boldsymbol{x}}_n} \tag{43}$$

C. Data Association

The messages $\varphi^{[p]}_{kl}(b^{(j)}_{l,n})$ sent from factor node $\Psi(a^{(j)}_{kl},b^{(j)}_{l})$ to variable node $b^{(j)}_{l,n}$ and the messages $\nu^{[p]}_{kl}(a^{(j)}_{kl,n})$ sent from factor node $\Psi(a^{(j)}_{kl},b^{(j)}_{l})$ to variable node $a^{(j)}_{kl,n}$ are calculated using the measurement evaluation messages in (35), (36) and (37). Details can be found in Appendix B.

D. Measurement Update for PVAs

Next, we determine the messages sent from factor node $q(\tilde{\boldsymbol{x}}_n,\underline{\boldsymbol{y}}^{(j)}_{k,n},\underline{a}^{(j)}_{kl},\boldsymbol{z}^{(j)}_{l,n})$ to variable node $\underline{\boldsymbol{y}}^{(j)}_{k,n}$ as

$$\gamma^{[p]}_l(\underline{\boldsymbol{y}}^{(j)}_{k,n})=\sum_{\underline{a}^{(j)}_{kl,n}\in\{0,1\}}\int q(\tilde{\boldsymbol{x}}_n,\underline{\boldsymbol{x}}^{(j)}_{k,n},\underline{r}^{(j)}_{k,n},\underline{a}^{(j)}_{kl,n},\boldsymbol{z}^{(j)}_{l,n}) \times \underline{\nu}^{[p]}_{kl}(\underline{a}^{(j)}_{kl,n})\mathrm{d}\tilde{\boldsymbol{x}}_n, \tag{44}$$

which results after marginalizing $\underline{a}^{(j)}_{kl,n}$ in

$$\gamma^{[p]}_l(\underline{\boldsymbol{x}}^{(j)}_{k,n},\underline{r}^{(j)}_{k}=1)=\int q(\tilde{\boldsymbol{x}}_n,\underline{\boldsymbol{x}}^{(j)}_{k,n},1,1,\boldsymbol{z}^{(j)}_{l,n})\underline{\nu}^{[p]}_{kl}(1)\mathrm{d}\tilde{\boldsymbol{x}}_n + \underline{\nu}^{[p]}_{kl}(0), \tag{45}$$

$$\gamma^{[p]}_l(\underline{\boldsymbol{x}}^{(j)}_{k,n},\underline{r}^{(j)}_{k}=0)=\underline{\nu}^{[p]}_{kl}(0). \tag{46}$$

The messages from factor node $u(\tilde{\boldsymbol{x}}_n,\overline{\boldsymbol{y}}^{(j)}_{k,n},\overline{a}^{(j)}_{kl},\boldsymbol{z}^{(j)}_{l,n})$ to variable node $\overline{\boldsymbol{y}}^{(j)}_{k,n}$ are given as

$$\gamma^{[p]}_l(\overline{\boldsymbol{y}}^{(j)}_{k,n})=\sum_{\overline{a}^{(j)}_{kl,n}\in\{0,1\}}\int u(\tilde{\boldsymbol{x}}_n,\overline{\boldsymbol{x}}^{(j)}_{k,n},\overline{r}^{(j)}_{k,n},\overline{a}^{(j)}_{kl,n},\boldsymbol{z}^{(j)}_{l,n}) \times \overline{\nu}^{[p]}_{kl}(\overline{a}^{(j)}_{kl,n})\mathrm{d}\tilde{\boldsymbol{x}}_n \tag{47}$$

which results after marginalizing $\overline{a}^{(j)}_{kl,n}$ in

$$\gamma^{[p]}_l(\overline{\boldsymbol{x}}^{(j)}_{k,n},\overline{r}^{(j)}_{k}=1)=\int u(\tilde{\boldsymbol{x}}_n,\overline{\boldsymbol{x}}^{(j)}_{k,n},1,1,\boldsymbol{z}^{(j)}_{l,n})\overline{\nu}^{[p]}_{kl}(1)\mathrm{d}\tilde{\boldsymbol{x}}_n + \overline{\nu}^{[p]}_{kl}(0) \tag{48}$$

$$\gamma^{[p]}_l(\overline{\boldsymbol{x}}^{(j)}_{k,n},\overline{r}^{(j)}_{k}=0)=\overline{\nu}^{[p]}_{kl}(0). \tag{49}$$

The message from factor node $v(\tilde{\boldsymbol{x}}_n,\overline{\boldsymbol{y}}^{(j)}_{m,n},\overline{a}^{(j)}_{mm},\boldsymbol{z}^{(j)}_{m,n})$ to variable node $\overline{\boldsymbol{y}}^{(j)}_{m,n}$ is given by

$$\gamma^{[p]}_m(\overline{\boldsymbol{y}}^{(j)}_{m,n})=\sum_{\overline{a}^{(j)}_{mm,n}\in\{0,1\}}\int v(\tilde{\boldsymbol{x}}_n,\overline{\boldsymbol{y}}^{(j)}_{m,n},\overline{a}^{(j)}_{mm,n},\boldsymbol{z}^{(j)}_{m,n}) \times \overline{\nu}^{[p]}_{mm}(\overline{a}^{(j)}_{mm,n})\mathrm{d}\tilde{\boldsymbol{x}}_n, \tag{50}$$

which results after marginalizing $\overline{a}^{(j)}_{mm,n}$ in

$$\gamma^{[p]}_m(\overline{\boldsymbol{x}}^{(j)}_{m,n},1)=\int v(\tilde{\boldsymbol{x}}_n,\overline{\boldsymbol{y}}^{(j)}_{m,n},\overline{a}^{(j)}_{mm,n},\boldsymbol{z}^{(j)}_{m,n})\overline{\nu}^{[p]}_{mm}(1)\mathrm{d}\tilde{\boldsymbol{x}}_n \tag{51}$$

$$\gamma^{[p]}_m(\overline{\boldsymbol{x}}^{(j)}_{m,n},0)=\overline{\nu}^{[p]}_{mm}(0). \tag{52}$$

The messages are initialized with $\gamma^{[p=1]}_{\ell}(\boldsymbol{y}^{(j)}_{k,n})=1$.

E. Extrinsic Information

For each legacy VA, the messages sent from variable node $\underline{\boldsymbol{y}}^{(j)}_{k,n}$ to factor nodes $q(\tilde{\boldsymbol{x}}_n,\underline{\boldsymbol{y}}^{(j)}_{k,n},\underline{a}^{(j)}_{kl,n};\boldsymbol{z}^{(j)}_{l,n})$ with $k\in\mathcal{K}^{(j)}_{n-1}, l\in\mathcal{M}^{(j)}_n$ at MP iteration $p+1$ are defined as

$$\alpha^{[p+1]}_l(\underline{\boldsymbol{y}}^{(j)}_{k,n})=\alpha(\underline{\boldsymbol{y}}^{(j)}_{k,n})\prod_{\substack{\ell=1\\ \ell\neq l}}^{M^{(j)}_n}\gamma^{[p]}_{\ell}(\underline{\boldsymbol{y}}^{(j)}_{k,n}). \tag{53}$$

For new VAs, a similar expression can be obtained for the messages from variable node $\overline{\boldsymbol{y}}^{(j)}_{m,n}$ to factor nodes $u(\tilde{\boldsymbol{x}}_n,\overline{\boldsymbol{y}}^{(j)}_{m,n},\overline{a}^{(j)}_{ml,n};\boldsymbol{z}^{(j)}_{l,n})$, and factor nodes $v(\tilde{\boldsymbol{x}}_n,\overline{\boldsymbol{y}}^{(j)}_{m,n},\overline{a}^{(j)}_{mm,n};\boldsymbol{z}^{(j)}_{m,n})$, i.e.,

$$\alpha^{[p+1]}_l(\overline{\boldsymbol{y}}^{(j)}_{m,n})=\alpha(\overline{\boldsymbol{y}}^{(j)}_{m,n})\prod_{\substack{\ell=1\\ \ell\neq l}}^{m}\gamma^{[p]}_{\ell}(\overline{\boldsymbol{y}}^{(j)}_{m,n}). \tag{54}$$

F. Measurement Update for Augmented Agent State

Due to the proposed scheduling, the augmented agent state is only updated by messages of legacy PVAs and only at the end of the iterative MP. This results in

$$\tilde{\beta}^{[p]}_{kl}(\tilde{\boldsymbol{x}}_n)=\alpha(\tilde{\boldsymbol{x}}_n), \tag{55}$$

$$\beta^{[p](j)}_{kl}(\tilde{\boldsymbol{x}}_n)=\sum_{\underline{a}^{(j)}_{kl,n}\in\{0,1\}}\sum_{\underline{r}^{(j)}_{k,n}\in\{0,1\}}\int \alpha^{[p]}_l(\underline{\boldsymbol{x}}^{(j)}_{k,n},\underline{r}^{(j)}_{k,n}) \times q(\tilde{\boldsymbol{x}}_n,\underline{\boldsymbol{x}}^{(j)}_{k,n},\underline{r}^{(j)}_{k,n},\underline{a}^{(j)}_{kl,n},\boldsymbol{z}^{(j)}_{l,n})\underline{\nu}^{[p]}_{kl}(\underline{a}^{(j)}_{kl,n})\mathrm{d}\underline{\boldsymbol{x}}^{(j)}_{k,n}, \tag{56}$$

which can be further simplified to

$$\beta^{[p](j)}_{kl}(\tilde{\boldsymbol{x}}_n)=\int \alpha^{[p]}_l(\boldsymbol{x}^{(j)}_{k,n},1)\Big(q(\tilde{\boldsymbol{x}}_n,\underline{\boldsymbol{x}}^{(j)}_{k,n},1,1,\boldsymbol{z}^{(j)}_{l,n})\underline{\nu}^{[p]}_{kl}(1) + \underline{\nu}^{[p]}_{kl}(0)\Big)\mathrm{d}\underline{\boldsymbol{x}}^{(j)}_{k,n}+\underline{\alpha}^{\mathrm{n},(j)}_{k}\underline{\nu}^{[p]}_{kl}(0). \tag{57}$$

G. Belief Calculation

Once all messages are available and $p = P$, the beliefs approximating the desired marginal posterior PDFs are obtained. The belief for the augmented agent state is given, up to a normalization factor, by

$$b(\tilde{\boldsymbol{x}}_n) \propto \alpha(\tilde{\boldsymbol{x}}_n) \prod_{j=1}^{J} \prod_{k=1}^{K_{n-1}^{(j)}} \prod_{m=1}^{M_n^{(j)}} \beta_{km}^{[P](j)}(\tilde{\boldsymbol{x}}_n), \tag{58}$$

where we only use messages from legacy VAs. This belief (after normalization) provides an approximation of the marginal posterior PDF $f(\tilde{\boldsymbol{x}}_n|\boldsymbol{z}_{1:n})$, and it is used instead of $f(\tilde{\boldsymbol{x}}_n|\boldsymbol{z}_{1:n})$ in (29). Furthermore, the beliefs of the legacy VAs $b(\underline{\boldsymbol{y}}_k^{(j)})$ and new VAs $b(\overline{\boldsymbol{y}}_k^{(j)})$ are given as

$$b(\underline{\boldsymbol{y}}_{k,n}^{(j)}) \propto \alpha(\underline{\boldsymbol{y}}_{k,n}^{(j)}) \prod_{l=1}^{M_n^{(j)}} \gamma_l^{[P]}(\underline{\boldsymbol{y}}_{k,n}^{(j)}), \tag{59}$$

$$b(\overline{\boldsymbol{y}}_{m,n}^{(j)}) \propto \alpha(\overline{\boldsymbol{y}}_{m,n}^{(j)}) \prod_{l=1}^{m} \gamma_l^{[P]}(\overline{\boldsymbol{y}}_{m,n}^{(j)}). \tag{60}$$

A computationally feasible approximate calculation of the various messages and beliefs can be based on the sequential Monte Carlo (particle-based) implementation approach introduced in [22], [26], [50].

VI. NUMERICAL RESULTS

The performance of the proposed algorithm (PROP) is validated and compared with the MP-SLAM algorithm from [3], [11], which assumes that each VA generates at most one measurement and that a measurement originates from at most one VA. The validation of the algorithms is based on synthetic measurements in two settings.

1) Experiment 1 in Section VI-B is based on measurements directly generated from the measurement model introduced in Section IV.
2) Experiment 2 in Section VI-C is based on measurements provided by a CEDA applied to radio signals that are generated with parameters according to the measurement model introduced in Section IV.

A. Simulation Scenario and Common Simulation Parameters

We consider an indoor scenario shown in Fig. 3. The scenario consists of two PAs at positions $\boldsymbol{p}_{\text{pa}}^{(1)} = [0.1\ 6]^{\text{T}}$ and $\boldsymbol{p}_{\text{pa}}^{(2)} = [0\ \ -0.2]^{\text{T}}$ and four reflective surfaces, i.e., four VAs per PA. The agent moves along a track which is observed for 300 time instances n with observation period $\Delta T = 1$ s. For simplicity, we restrict the simulations to single-bounce reflections. The distances of the main components are calculated based on the PA and the corresponding VA positions as well as agent positions (see

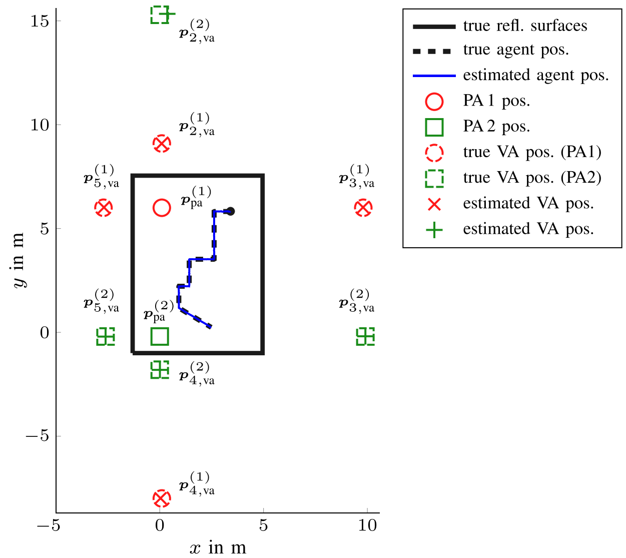


Figure 3. Considered scenario for performance evaluation in a rectangular room with two PAs, four reflective surfaces, and the corresponding VAs. The estimated agent track for a single realization is shown in blue.

Section III). Fig. 4 shows the distances of the main components versus time n. The signal SNR is set to 30 dB at an LOS distance of 1 m. The amplitudes of the main components (LOS component and the MPCs) are calculated using a free-space path loss model and an additional attenuation of 1 dB for each reflection at a flat surface. We use 20 000 particles. The particles for the initial agent state are drawn from a four-dimensional uniform distribution with center $\boldsymbol{x}_0 = [\boldsymbol{p}_0^{\text{T}}\ 0\ \ 0]^{\text{T}}$, where $\boldsymbol{p}_0$ is the starting position of the actual agent track, and the support of each position component about the respective center is given by [−0.1 m, 0.1 m] and of each velocity component is given by [−0.01 m/s, 0.01 m/s]. At time $n = 0$, the number of VAs is 0, i.e., no prior map information is available. The prior distribution for new PVA states $f_{\text{n}}(\overline{\boldsymbol{x}}_{m,n}^{(j)}|\tilde{\boldsymbol{x}}_n)$ is uniform on the square region given by [−15 m, 15 m] × [−15 m, 15 m] around the cen-

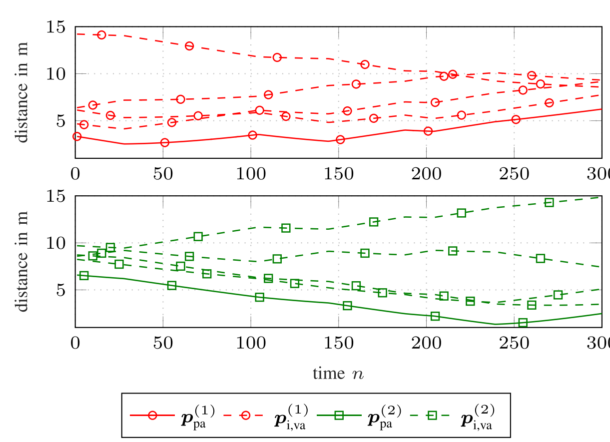


Figure 4. Distances of main components (between the PA positions as well as their corresponding VA positions and the agent positions) versus time n.

ter of the floor plan shown in Fig. 3, and the mean number of new PVAs at time n is $\mu_{\mathrm{n}} = 0.01$. The probability of survival is $p_{\mathrm{s}} = 0.999$. The confirmation threshold as well as the pruning threshold are given as $p_{\mathrm{cf}} = 0.5$ and $p_{\mathrm{pr}} = 10^{-3}$, respectively. For the sake of numerical stability, we introduce a small amount of regularization noise to the VA state $\boldsymbol{p}_{k,\mathrm{va}}$ at each time step n, i.e., $\underline{\boldsymbol{p}}_{k,\mathrm{va}}^{(j)} = \boldsymbol{p}_{k,\mathrm{va}}^{(j)} + \boldsymbol{\omega}_k$, where $\boldsymbol{\omega}_k$ is i.i.d. across k, zero-mean, and Gaussian with covariance matrix $\sigma_a^2 \mathbf{I}_2$ and $\sigma_a = 10^{-3}$ m. The state transition variances are set as $\sigma_{\mathrm{w}} = 10^{-3}\,\mathrm{m/s^2}$, $q_\tau = q_{\mathrm{u}} = 10^4$ [24], [27], and $\sigma_{\mathrm{u},k} = 0.05\, u_{k,n-1}^{(j)\,\mathrm{MMSE}}$. Note that for the normalized amplitude state we use a value proportional to the MMSE estimate of the previous time step $n{-}1$ as a heuristic. The dispersion parameters are set to fixed values over time n, i.e., $\psi_{\tau,n} = \psi_\tau = \psi_{\mathrm{d}}/c$ and $\psi_{\mathrm{u},n} = \psi_{\mathrm{u}}$.[5] The performance of the different methods discussed is measured in terms of the root-mean-squared error (RMSE) of the agent position and the dispersion parameters, as well as the optimal subpattern assignment (OSPA) error [53] of all VAs with with cutoff parameter and order set to 5 m and 2, respectively. The mean OSPA (MOSPA) errors and RMSEs of each unknown variable are obtained by averaging over all converged simulation runs. We declare a simulation run to be converged if $\{\forall n : \|\boldsymbol{p}_n - \boldsymbol{p}_n^{\mathrm{MMSE}}\| < d_{\mathrm{cv}}\ \mathrm{m}\}$, where d_{cv} is the convergence threshold.

B. Experiment 1: Measurement Model

We investigate PROP with four different dispersion parameter settings, given as ψ_{d}, which takes values of 0 m, 0.03 m, 0.15 m, and 0.3 m, and ψ_{u}, which is either set to 0 for $\psi_{\mathrm{d}} = 0$ m or 0.2 otherwise. Furthermore, we set $N_{\mathrm{ny}} = 4$. We performed 100 simulation runs. In each simulation run, we generated noisy measurements $\boldsymbol{z}_{m,n}^{(j)}$ according to the measurement model proposed in Section IV-B using the main components calculated as described in Section VI-A. In the case $\psi_{\mathrm{d}} = 0$ m, only main-component measurements are generated, which is equivalent to the system model in [11]. The detection threshold is given by $\gamma = 2.5$. For numerical stability, we reduced the root-mean-squared bandwidth β_{bw} for VAs by a factor of 4. The convergence threshold is set to $d_{\mathrm{cv}} = 0.2$.

Table I summarizes the number of converged runs (in percentage) as well as the mean number of detected VAs $\hat{K}$ (averaged over all simulation runs and time steps) for all investigated dispersion parameter settings. The results are summarized in Fig. 5. In particular, Fig. 5(a) shows the RMSE of the agent positions, Fig. 5(b) and (c) show the RMSE of the dispersion parameters, and Fig. 5(d)–(i) shows the MO-

[5]For better readability, we introduce ψ_{d} as a scaled version of ψ_τ.

Table I
Experiment 1: Convergence Rate and Mean Number of Estimated VAs for Different Algorithms and Dispersion Settings

	Setting	Convergence	$\hat{K}$
MP-SLAM	$\psi_{\mathrm{d}} = 0.00$ m	100%	4
	$\psi_{\mathrm{d}} = 0.03$ m	82%	9
	$\psi_{\mathrm{d}} = 0.15$ m	15%	16
	$\psi_{\mathrm{d}} = 0.30$ m	11%	30
PROP	$\psi_{\mathrm{d}} = 0.00$ m	100%	4
	$\psi_{\mathrm{d}} = 0.03$ m	100%	4
	$\psi_{\mathrm{d}} = 0.15$ m	100%	4
	$\psi_{\mathrm{d}} = 0.30$ m	96%	5

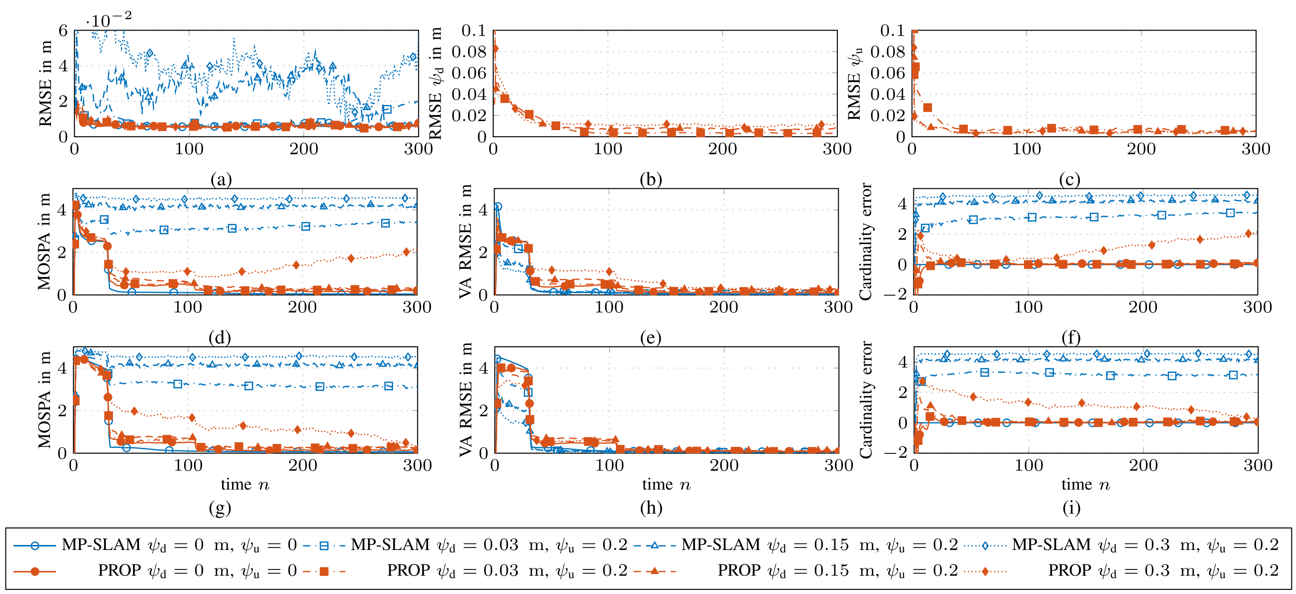


Figure 5. Experiment 1: Results for converged simulation runs. (a) shows the RMSE of the agent position over the whole track. (b) and (c) present the RMSE of the dispersion parameters. (d) and (g) present the map error in terms of the MOSPA for PA 1 and PA 2, respectively. (e) and (h) show the RMSE of the estimated VA positions for PA 1 and PA 2, respectively. (f) and (i) show the cardinality error of the estimated VAs for PA 1 and PA 2, respectively.

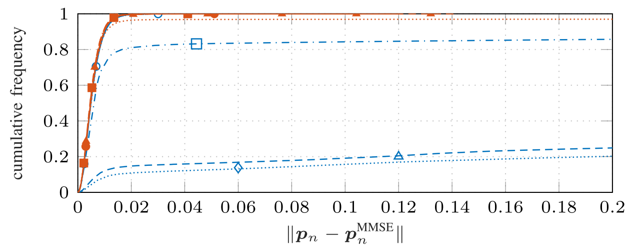


Figure 6. Experiment 1: Cumulative frequency of the deviation of the MMSE estimate of the agent position from the true agent position for all simulation runs and time instances. The legend is given in Fig. 5.

SPA error and its VA position error and mean cardinality error contributions for PA 1 and PA 2, respectively. The results in all figures are presented versus time n (and for all investigated dispersion parameter settings). Fig. 5(a) shows that the RMSE of the agent position of PROP is similar for all dispersion parameter settings. While PROP significantly outperforms MP-SLAM in terms of converged runs for dispersion parameter settings $\psi_{\mathrm{d}} > 0\,\mathrm{m}$, it shows slightly reduced performance for $\psi_{\mathrm{d}} = 0\,\mathrm{m}$. Additionally, Fig. 6 shows the cumulative frequencies of the individual agent errors, i.e., $\|\boldsymbol{p}_n - \boldsymbol{p}_n^{\mathrm{MMSE}}\|$ for all simulation runs and time instances. It can be observed that the MMSE positions of the agent of PROP show almost no large deviations, while the estimates of MP-SLAM exhibit large errors in many simulation runs. For dispersion parameter settings $\psi_{\mathrm{d}} > 0\,\mathrm{m}$, measurements of the subcomponents are available. Thus, as Fig. 5(b) and (c) show, the dispersion parameters are well estimated, as indicated by the small RMSEs. For the setting $\psi_{\mathrm{d}} = 0\,\mathrm{m}$, estimation of the dispersion parameters is not possible because there are no subcomponent measurements, i.e., there is only one measurement generated by each VA. However, as Fig. 5(a) shows, this does not affect the accuracy of the agent's position estimation.

The MOSPA errors (and their VA positions and the mean cardinality error contributions) of PROP, shown in Fig. 5(d) and (g), are very similar for all dispersion parameter settings. They slightly increase with an increased dispersion parameter ψ_{d}. Only for the setting $\psi_{\mathrm{d}} = 0.3\,\mathrm{m}$, PROP shows a larger cardinality error. This can be explained by looking at the distances from PA 1 and its corresponding VAs, as shown in Fig. 4. At the end of the agent track, many VAs show similar distances to the agent's position, making it difficult to resolve the individual components. For larger dispersion parameter ψ_{d}, this becomes even more challenging, leading to increased MOSPA errors. For PA 2 and the corresponding VAs, Fig. 4 shows that all components are well separated by their distances at the end of the agent track, which makes it easier for PROP to correctly estimate the number and positions of VAs. Unlike PROP, MP-SLAM completely fails to estimate the correct number of VAs for larger ψ_{d} (and ψ_{u}), resulting in a large cardinality error. This can be explained by the fact that MP-SLAM does not consider additional sub-components in the measurement and system model. We suspect that this estimation of additional spurious VAs is the reason for the large number of divergent simulation runs. As an example, Fig. 7 depicts the time evolution of the estimated distances (using the PA position, the estimated VA positions, and the estimated agent positions) with according component SNRs as well as the respective dispersion parameters for PA 1.

Table II
Experiment 2: Convergence Rate and Mean Number of Estimated VAs for Different Algorithms

Setting	Convergence	$\hat{K}$
MP-SLAM $\psi_{\mathrm{d}} = 0.30\,\mathrm{m}$	20%	7.5
PROP $\psi_{\mathrm{d}} = 0.30\,\mathrm{m}$	100%	3.7

C. Experiment 2: Radio Signals

In this section, we use a dispersion parameter setting of $\psi_{\mathrm{d}} = 0.3\,\mathrm{m}$ and $\psi_{\mathrm{u}} = 0.2$. The signal spectrum of the transmit pulse $s(t)$ has a root-raised-cosine shape with a roll-off factor of 0.6 and a 3 dB bandwidth of $B = 1\,\mathrm{GHz}$. The signal is critically sampled, i.e., $T_{\mathrm{s}} = 1/(1.6B)$, with a total number of $N_{\mathrm{s}} = 161$ samples, resulting in a maximum distance $d_{\max} = 60\,\mathrm{m}$. For the data generation, we use $N_{\mathrm{ny}} = 2$. We perform ten simulation runs. In each simulation run, we generate a received signal vector (see (6)) using the main components calculated as described in Section VI-A and uniformly distributed subcomponents (see (14)). To obtain the measurements, we use the CEDA in [19] with a detection threshold of $\gamma = 2$, i.e., corresponding to 6 dB [23]. For numerical stability, we reduced the root-mean-squared bandwidth β_{bw} for VAs by a factor of 4 and increased the factor $1/2$ in amplitude scale parameter in (13) to 4. The convergence threshold is $d_{\mathrm{cv}} = 2$.

Table II again summarizes the number of converged runs and the mean number of detected VAs. For PROP, none of the simulation runs diverged, but 80% of the MP-SLAMs simulation runs diverged, showing that PROP significantly outperforms MP-SLAM. The results shown in Fig. 8 follow a similar trend as the results shown in Fig. 5. The only significant difference is observed in the RMSE of the dispersion parameter ψ_{u}, which remains relatively large (see Fig. 8(c)). This is because the variance of the estimated normalized amplitudes provided by the CEDA is very large. This may be explained by two factors: (i) the CEDA also needs to estimate the noise variance, which is only approximately covered by the amplitude scale parameter given in (13) and (ii) the subcomponents are very close in the delay domain, resulting in strongly correlated amplitude estimates. The steps in Fig. 8(d) and (f) are due to crossings where the delays from two or more VAs to the agent are equal. Hence, one of the VAs is discarded, leading to an overall underestimated number of VAs.

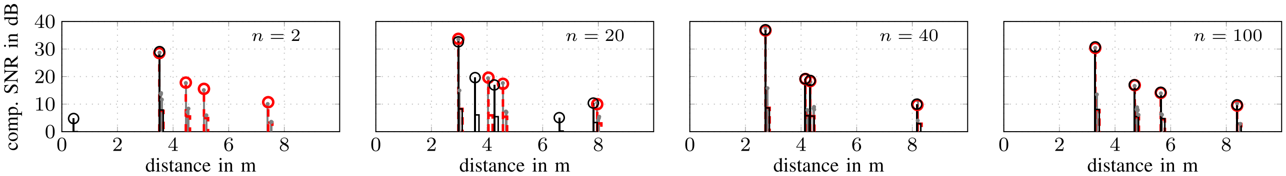


Figure 7. Estimated distances and dispersion parameters for PA 1 for a single simulation run are represented by dot markers and boxes, respectively. The true components and respective dispersion parameters are indicated in red. All measurements are indicated in gray. Estimated components and respective dispersion parameters are indicated in black.

VII. CONCLUSIONS

We have proposed a new MP-SLAM method that can cope with multiple measurements being generated by a single environment feature, i.e., a single VA. It is based on a novel statistical measurement model that is derived from the radio signal introducing dispersion parameters to MPCs. The resulting likelihood function model allows to capture the measurement spread originating from nonideal effects such as rough reflective surfaces or noncalibrated antennas. The performance results show that the proposed method is able to cope with multiple measurements being produced per VA and outperforms classical MP-SLAM in terms of the agent positioning error and the map MOSPA error. We show that multiple measurements get correctly associated with their corresponding VA, resulting in a correctly estimated number of VAs. Furthermore, the results indicate that the proposed algorithm generalizes to the classical MP-SLAM for a single measurement per VA. Possible directions for future research include the extension of individual dispersion parameters for each feature as well as incorporating multiple-measurements-to-feature data association into the MVA-based MP-SLAM method [46].

APPENDIX A
RADIO SIGNAL MODEL

In this section, we derive the radio signal model described in Section III. Usually, specular reflections of radio signals on flat surfaces are modeled by VAs that are mirror images of the PAs [1]–[4]. We start by defining the typical channel impulse response, given for time n and anchor j as

$$h_{\mathrm{c},n}^{(j)}(\tau) = \sum_{l=1}^{L_n^{(j)}} \alpha_{l,n}^{(j)} \delta\big(\tau - \tau_{l,n}^{(j)}\big). \tag{61}$$

The first summand describes the LOS component and the sum of $L_n^{(j)} - 1$ the specular MPCs with their corresponding complex amplitudes $\alpha_{l,n}^{(j)}$ and delays $\tau_{l,n}^{(j)}$, respectively. In nonideal radio channels, we observe rays to arrive as clusters [6], [7], [54], [55]. The reason for this observation is manifold. Typical examples are noncalibrated antennas, the scattering from a user-body as well as nonideal reflective surfaces. Fig. 1 visualizes these effects, introducing generic impulse responses $h_{\mathrm{ant},n}^{(j)}(\tau)$ and $h_{\mathrm{surf},n}^{(j)}(\tau)$. We propose to model the overall impulse response encompassing all considered dispersion effects

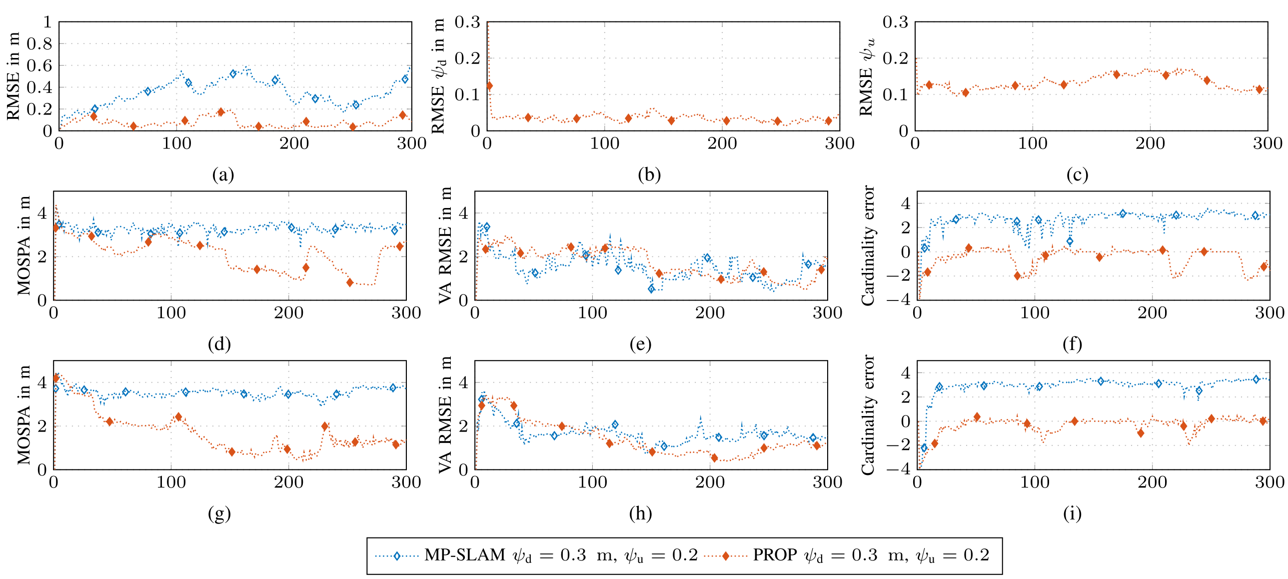


Figure 8. Experiment 2: Results for converged simulation runs based on estimates from CEDA. (a) shows the RMSE of the agent position over the whole track. (b) and (c) present the RMSE of the dispersion parameters. (d) and (g) present the map error in terms of the MOSPA for PA 1 and PA 2, respectively. (e) and (h) show the RMSE of the estimated VA positions for PA 1 and PA 2, respectively. (f) and (i) show the cardinality error of the estimated VAs for PA 1 and PA 2, respectively.

as

$$h_{\mathrm{d},n}^{(j)}(\tau) = \delta(\tau) + \sum_{i=1}^{S_l^{(j)}} \beta_{l,i,n}^{(j)} \delta(\tau - \nu_{l,i,n}^{(j)}), \tag{62}$$

where $\beta_{l,i,n}^{(j)} \in \mathbb{R}$ is a relative dampening variable and $\nu_{l,i,n}^{(j)}$ is the excess delay. The presented model denotes a marked Possion point process [55]. Its statistical properties, i.e, the distribution of $\nu_{l,i,n}^{(j)}$, $\beta_{l,i,n}^{(j)}$, and $S_l^{(j)}$, are discussed in Sections III and IV in detail. We obtain the complex baseband signal received at the jth anchor given by the convolution of $h_{\mathrm{d},n}^{(j)}(\tau)$ and $h_{\mathrm{c},n}^{(j)}(\tau)$ with the transmitted signal $s(t)$ as

$$\begin{aligned} \mathsf{s}_{\mathrm{rx},n}^{(j)} = \sum_{l=1}^{L_n^{(j)}} \alpha_{l,n}^{(j)} \Big(s(t - \tau_{l,n}^{(j)}) \\ + \sum_{i=1}^{S_l^{(j)}} \beta_{l,i,n}^{(j)} s(t - \tau_{l,n}^{(j)} - \nu_{l,i,n}^{(j)}) \Big) + \mathsf{n}_n^{(j)}(t). \end{aligned} \tag{63}$$

The second term $\mathsf{n}_n^{(j)}(t)$ represents an AWGN process with double-sided power spectral density $N_0^{(j)}/2$.

APPENDIX B DATA ASSOCIATION

This section contains the detailed derivation of the data association-related messages $\varphi_{kl}^{[p]}(b_{l,n}^{(j)})$ and $\nu_{kl}^{[p]}(a_{kl,n}^{(j)})$. Using the measurement evaluation messages in (35), (36), and (37), the messages $\underline{\varphi}_{kl}^{[p]}(b_{l,n}^{(j)})$ and $\overline{\varphi}_{ml}^{[p]}(b_{l,n}^{(j)})$ are calculated by

$$\underline{\varphi}_{kl}^{[p]}(b_{l,n}^{(j)}) = \sum_{\underline{a}_{kl,n}^{(j)} \in \{0,1\}} \varepsilon^{[p]}(\underline{a}_{kl,n}^{(j)}) \underline{\Psi}(\underline{a}_{kl,n}^{(j)}, b_{l,n}^{(j)}), \tag{64}$$

$$\overline{\varphi}_{ml}^{[p]}(b_{l,n}^{(j)}) = \sum_{\overline{a}_{ml,n}^{(j)} \in \{0,1\}} \varepsilon^{[p]}(\overline{a}_{ml,n}^{(j)}) \overline{\Psi}(\overline{a}_{ml,n}^{(j)}, b_{l,n}^{(j)}) \tag{65}$$

for $k \in \{1, \ldots, \underline{K}\}$ with $\underline{K} \triangleq K_{n-1}^{(j)}$ and $m, l \in \{1, \ldots, M_n^{(j)}\}$ and are sent from factor node $\underline{\Psi}(\underline{a}_{kl,n}^{(j)}, b_{l,n}^{(j)})$ and $\overline{\Psi}(\overline{a}_{ml,n}^{(j)}, b_{l,n}^{(j)})$ to variable node $b_{l,n}^{(j)}$, respectively. By making use of the indicator functions given in (27) and (28), respectively, (64) and (65) are also given as

$$\underline{\varphi}_{kl}^{[p]}(b_{l,n}^{(j)} = k) = \varepsilon^{[p]}(\underline{a}_{kl,n}^{(j)} = 1), \tag{66}$$

$$\underline{\varphi}_{kl}^{[p]}(b_{l,n}^{(j)} \neq k) = \varepsilon^{[p]}(\underline{a}_{kl,n}^{(j)} = 0), \tag{67}$$

$$\overline{\varphi}_{ml}^{[p]}(b_{l,n}^{(j)} = \underline{K} + m) = \varepsilon^{[p]}(\overline{a}_{ml,n}^{(j)} = 1), \tag{68}$$

$$\overline{\varphi}_{ml}^{[p]}(b_{l,n}^{(j)} \neq \underline{K} + m) = \varepsilon^{[p]}(\overline{a}_{ml,n}^{(j)} = 0). \tag{69}$$

The messages in (66)–(69) can be rewritten in the form of

$$\underline{\varphi}_{kl}^{[p]}(b_{l,n}^{(j)}) = \begin{cases} \frac{\varepsilon^{[p]}(\underline{a}_{kl,n}^{(j)} = 1)}{\varepsilon^{[p]}(\underline{a}_{kl,n}^{(j)} = 0)}, & b_{l,n}^{(j)} = k \\ 1, & b_{l,n}^{(j)} \neq k \end{cases} \tag{70}$$

$$\overline{\varphi}_{ml}^{[p]}(b_{l,n}^{(j)}) = \begin{cases} \frac{\varepsilon^{[p]}(\overline{a}_{ml,n}^{(j)} = 1)}{\varepsilon^{[p]}(\overline{a}_{ml,n}^{(j)} = 0)}, & b_{l,n}^{(j)} = \underline{K} + m \\ 1, & b_{l,n}^{(j)} \neq \underline{K} + m. \end{cases} \tag{71}$$

The messages $\underline{\nu}_{kl}^{[p]}(\underline{a}_{kl,n}^{(j)})$ and $\overline{\nu}_{ml}^{[p]}(\overline{a}_{ml,n}^{(j)})$ represent the messages from variable node $\underline{a}_{kl,n}^{(j)}$ to factor node $q(\tilde{\boldsymbol{x}}_n, \underline{\boldsymbol{y}}_{k,n}^{(j)}, \underline{a}_{kl,n}^{(j)}; \boldsymbol{z}_{l,n}^{(j)})$ and from variable node $\overline{a}_{ml,n}^{(j)}$ to factor node $u(\tilde{\boldsymbol{x}}_n, \overline{\boldsymbol{y}}_{m,n}^{(j)}, \overline{a}_{ml,n}^{(j)}; \boldsymbol{z}_{l,n}^{(j)})$, respectively. $\overline{\nu}_{mm}^{[p]}(\overline{a}_{mm,n}^{(j)})$ represents the messages from variable node $\overline{a}_{mm,n}^{(j)}$ to factor node $v(\tilde{\boldsymbol{x}}_n, \overline{\boldsymbol{y}}_{m,n}^{(j)}, \overline{a}_{mm,n}^{(j)}; \boldsymbol{z}_{m,n}^{(j)})$. They are defined as

$$\underline{\nu}_{kl}^{[p]}(\underline{a}_{kl,n}^{(j)}) = \sum_{b_{l,n}^{(j)}=0}^{K_n^{(j)}} \prod_{\substack{i=1 \\ i \neq k}}^{\underline{K}} \underline{\varphi}_{il}^{[p]}(b_{l,n}^{(j)}) \prod_{m=l}^{M_n^{(j)}} \overline{\varphi}_{ml}^{[p]}(b_{l,n}^{(j)}), \tag{72}$$

$$\overline{\nu}_{ml}^{[p]}(\overline{a}_{ml,n}^{(j)}) = \sum_{b_{l,n}^{(j)}=0}^{K_n^{(j)}} \prod_{i=1}^{\underline{K}} \underline{\varphi}_{il}^{[p]}(b_{l,n}^{(j)}) \prod_{\substack{h=l \\ h \neq m}}^{M_n^{(j)}} \overline{\varphi}_{hl}^{[p]}(b_{l,n}^{(j)}). \tag{73}$$

Using the results from (70) and (71), (72) and (73) are, respectively, rewritten as

$$\underline{\nu}_{kl}^{[p]}(\underline{a}_{kl,n}^{(j)} = 1) = \prod_{\substack{i=1 \\ i \neq k}}^{\underline{K}} \underline{\varphi}_{il}^{[p]}(b_{l,n}^{(j)} = k) \prod_{m=l}^{M_n^{(j)}} \overline{\varphi}_{ml}^{[p]}(b_{l,n}^{(j)} = \underline{K} + k), \tag{74}$$

$$\underline{\nu}_{kl}^{[p]}(\underline{a}_{kl,n}^{(j)} = 0) = \sum_{\substack{b_{l,n}^{(j)}=0 \\ b_{l,n}^{(j)} \notin \{k, \underline{K}+k\}}}^{K_n^{(j)}} \prod_{\substack{i=1 \\ i \neq k}}^{\underline{K}} \underline{\varphi}_{il}^{[p]}(b_{l,n}^{(j)}) \prod_{m=l}^{M_n^{(j)}} \overline{\varphi}_{ml}^{[p]}(b_{l,n}^{(j)}) \tag{75}$$

and

$$\overline{\nu}_{ml}^{[p]}(\overline{a}_{ml,n}^{(j)} = 1) = \prod_{i=1}^{\underline{K}} \underline{\varphi}_{il}^{[p]}(b_{l,n}^{(j)} = m) \prod_{\substack{h=l \\ h \neq m}}^{M_n^{(j)}} \overline{\varphi}_{hl}^{[p]}(b_{l,n}^{(j)} = \underline{K} + m), \tag{76}$$

$$\overline{\nu}_{ml}^{[p]}(\overline{a}_{ml,n}^{(j)} = 0) = \sum_{\substack{b_{l,n}^{(j)}=0 \\ b_{l,n}^{(j)} \notin \{m, \underline{K}+m\}}}^{K_n^{(j)}} \prod_{i=1}^{\underline{K}} \underline{\varphi}_{il}^{[p]}(b_{l,n}^{(j)}) \prod_{\substack{h=l \\ h \neq m}}^{M_n^{(j)}} \overline{\varphi}_{hl}^{[p]}(b_{l,n}^{(j)}). \tag{77}$$

Note that $\varphi_{kl}^{[p]}(b_{l,n}^{(j)} = 0) = 1$. By normalizing (74) by $\underline{\nu}_{kl}^{[p]}(\underline{a}_{kl,n}^{(j)} = 0)$ and (76) by $\overline{\nu}_{ml}^{[p]}(\overline{a}_{ml,n}^{(j)} = 0)$, equivalent expressions for (72) and (73) are given as

$$\underline{\nu}_{kl}^{[p]}(\underline{a}_{kl,n}^{(j)}) = \begin{cases} \dfrac{\prod_{\substack{i=1\\i\neq k}}^{K} \underline{\varphi}_{il}^{[p]}(b_{l,n}^{(j)}=k) \prod_{m=l}^{M} \overline{\varphi}_{ml}^{[p]}(b_{l,n}^{(j)}=\underline{K}+k)}{\sum_{\substack{b_{l,n}^{(j)}=0\\ b_{l,n}^{(j)}\notin\{k,\underline{K}+k\}}}^{K_n^{(j)}} \prod_{\substack{i=1\\i\neq k}}^{K} \underline{\varphi}_{il}^{[p]}(b_{l,n}^{(j)}) \prod_{m=l}^{M_n^{(j)}} \overline{\varphi}_{ml}^{[p]}(b_{l,n}^{(j)})}, & \underline{a}_{kl,n}^{(j)} = 1 \\ 1, & \underline{a}_{kl,n}^{(j)} = 0. \end{cases} \tag{78}$$

$$\overline{\nu}_{ml}^{[p]}(\overline{a}_{ml,n}^{(j)}) = \begin{cases} \dfrac{\prod_{i=1}^{K} \underline{\varphi}_{il}^{[p]}(b_{l,n}^{(j)}=m) \prod_{\substack{h=l\\h\neq m}}^{M_n^{(j)}} \overline{\varphi}_{hl}^{[p]}(b_{l,n}^{(j)}=\underline{K}+m)}{\sum_{\substack{b_{l,n}^{(j)}=0\\ b_{l,n}^{(j)}\notin\{m,\underline{K}+m\}}}^{K_n^{(j)}} \prod_{i=1}^{K} \underline{\varphi}_{il}^{[p]}(b_{l,n}^{(j)}) \prod_{\substack{h=l\\h\neq m}}^{M_n^{(j)}} \overline{\varphi}_{hl}^{[p]}(b_{l,n}^{(j)}=\underline{K}+m)}, & \overline{a}_{ml,n}^{(j)} = 1 \\ 1, & \overline{a}_{ml,n}^{(j)} = 0. \end{cases} \tag{79}$$

Finally, by calculating the explicit summations and multiplications in (78) and (79), it results in

$$\underline{\nu}_{kl}^{[p]}(\underline{a}_{kl,n}^{(j)}) = \begin{cases} \dfrac{1}{1+\sum_{\substack{i=1\\i\neq k}}^{K} \underline{\varphi}_{il}^{[p]}(b_{l,n}^{(j)}=i)+\sum_{m=l}^{M_n^{(j)}} \overline{\varphi}_{ml}^{[p]}(b_{l,n}^{(j)}=\underline{K}+m)}, & \underline{a}_{kl,n}^{(j)} = 1 \\ 1, & \underline{a}_{kl,n}^{(j)} = 0, \end{cases} \tag{80}$$

$$\overline{\nu}_{ml}^{[p]}(\overline{a}_{ml,n}^{(j)}) = \begin{cases} \dfrac{1}{1+\sum_{i=1}^{K} \underline{\varphi}_{il}^{[p]}(b_{l,n}^{(j)}=i)+\sum_{\substack{h=l\\h\neq m}}^{M_n^{(j)}} \overline{\varphi}_{hl}^{[p]}(b_{l,n}^{(j)}=\underline{K}+m)}, & \overline{a}_{ml,n}^{(j)} = 1 \\ 1, & \overline{a}_{ml,n}^{(j)} = 0. \end{cases} \tag{81}$$

REFERENCES

[1] E. Leitinger, P. Meissner, C. Rudisser, G. Dumphart, and K. Witrisal
"Evaluation of position-related information in multipath components for indoor positioning,"
IEEE J. Sel. Areas Commun., vol. 33, no. 11, pp. 2313–2328, Nov. 2015.

[2] K. Witrisal et al.
"High-accuracy localization for assisted living: 5G systems will turn multipath channels from foe to friend,"
IEEE Signal Process. Mag., vol. 33, no. 2, pp. 59–70, Mar. 2016.

[3] E. Leitinger, F. Meyer, F. Hlawatsch, K. Witrisal, F. Tufvesson, and M. Z. Win
"A belief propagation algorithm for multipath-based SLAM,"
IEEE Trans. Wireless Commun., vol. 18, no. 12, pp. 5613–5629, Dec. 2019.

[4] R. Mendrzik, F. Meyer, G. Bauch, and M. Z. Win
"Enabling situational awareness in millimeter wave massive MIMO systems,"
IEEE J. Sel. Topics Signal Process., vol. 13, no. 5, pp. 1196–1211, Sep. 2019.

[5] C. Gentner, T. Jost, W. Wang, S. Zhang, A. Dammann, and U. C. Fiebig
"Multipath assisted positioning with simultaneous localization and mapping,"
IEEE Trans. Wireless Commun., vol. 15, no. 9, pp. 6104–6117, Sep. 2016.

[6] J. Kulmer, F. Wen, N. Garcia, H. Wymeersch, and K. Witrisal
"Impact of rough surface scattering on stochastic multipath component models,"
in *Proc. IEEE Annu. Int. Symp. Pers., Indoor Mobile Radio Commun.*, 2018, 2018, pp. 1410–1416.

[7] F. Wen, J. Kulmer, K. Witrisal, and H. Wymeersch
"5G positioning and mapping with diffuse multipath,"
IEEE Trans. Wireless Commun., vol. 20, no. 2, pp. 1164–1174, 2021.

[8] R. Pöhlmann, S. Zhang, E. Staudinger, S. Caizzone, A. Dammann, and P. A. Hoeher
"Bayesian in-situ calibration of multiport antennas for DoA estimation: Theory and measurements,"
IEEE Access, vol. 10, pp. 37967–37983, 2022.

[9] H. Durrant-Whyte and T. Bailey
"Simultaneous localization and mapping: Part I,"
IEEE Robot. Autom. Mag., vol. 13, no. 2, pp. 99–110, Jun. 2006.

[10] M. Dissanayake, P. Newman, S. Clark, H. Durrant-Whyte, and M. Csorba
"A solution to the simultaneous localization and map building (SLAM) problem,"
IEEE Trans. Robot. Autom., vol. 17, no. 3, pp. 229–241, Jun. 2001.

[11] E. Leitinger, S. Grebien, and K. Witrisal
"Multipath-based SLAM exploiting AoA and amplitude information,"
in *Proc. IEEE Int. Conf. Commun. Workshops*, 2019, pp. 1–7.

[12] H. Kim, K. Granström, L. Gao, G. Battistelli, S. Kim, and H. Wymeersch
"5G mmWave cooperative positioning and mapping using multi-model PHD filter and map fusion,"
IEEE Trans. Wireless Commun., vol. 19, no. 6, pp. 3782–3795, Mar. 2020.

[13] H. Kim, K. Granstrom, L. Svensson, S. Kim, and H. Wymeersch
"PMBM-based SLAM filters in 5G mmWave vehicular networks,"
IEEE Trans. Veh. Technol., vol. 71, no. 8, pp. 8646–8661, Aug. 2022.

[14] E. Leitinger and F. Meyer
"Data fusion for multipath-based SLAM,"
in *Proc. 54th Asilomar Conf. Signals, Syst., Comput.*, 2020, pp. 934–939.

[15] E. Leitinger, B. Teague, W. Zhang, M. Liang, and F. Meyer
"Data fusion for radio frequency SLAM with robust sampling,"
in *Proc. 25th Int. Conf. Inf. Fusion*, 2022, pp. 1–6.

[16] E. Leitinger, A. Venus, B. Teague, and F. Meyer
"Data fusion for multipath-based SLAM: Combining information from multiple propagation paths,"
IEEE Trans. Signal Process., to be published. doi: 10.1109/TSP.2023.3310360.

[17] A. Richter
"Estimation of radio channel parameters: Models and algorithms," Ph.D. dissertation, Ilmenau University of Technology, Ilmenau, Germany, 2005.

[18] D. Shutin, W. Wang, and T. Jost
"Incremental sparse Bayesian learning for parameter estimation of superimposed signals,"
in *Proc. Int. Conf. Sampling Theory Appl.*, 2013, pp. 6–9.

[19] T. L. Hansen, M. A. Badiu, B. H. Fleury, and B. D. Rao
"A sparse Bayesian learning algorithm with dictionary parameter estimation,"

in *Proc. IEEE 8th Sensor Array Multichannel Signal Process. Workshop*, 2014, pp. 385–388.

[20] M. A. Badiu, T. L. Hansen, and B. H. Fleury "Variational Bayesian inference of line spectra," *IEEE Trans. Signal Process.*, vol. 65, no. 9, pp. 2247–2261, May 2017.

[21] T. L. Hansen, B. H. Fleury, and B. D. Rao "Superfast line spectral estimation," *IEEE Trans. Signal Process.*, vol. 66, no. 10, pp. 2511–2526, May 2018.

[22] X. Li, E. Leitinger, A. Venus, and F. Tufvesson "Sequential detection and estimation of multipath channel parameters using belief propagation," *IEEE Trans. Wireless Commun.*, vol. 21, no. 10, pp. 8385–8402, Apr. 2022.

[23] S. Grebien, E. Leitinger, B. H. Fleury, and K. Witrisal "Super-resolution channel estimation including the dense multipath component — A sparse variational Bayesian approach," 2023, *arXiv:2308.01702*.

[24] F. Meyer and J. L. Williams "Scalable detection and tracking of geometric extended objects," *IEEE Trans. Signal Process.*, vol. 69, pp. 6283–6298, Oct. 2021.

[25] J. Williams and R. Lau "Approximate evaluation of marginal association probabilities with belief propagation," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 50, no. 4, pp. 2942–2959, Oct. 2014.

[26] F. Meyer et al. "Message passing algorithms for scalable multitarget tracking," *Proc. IEEE*, vol. 106, no. 2, pp. 221–259, Feb. 2018.

[27] J. W. Koch "Bayesian approach to extended object and cluster tracking using random matrices," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 44, no. 3, pp. 1042–1059, Jul. 2008.

[28] K. Granström, M. Fatemi, and L. Svensson "Poisson multi-Bernoulli mixture conjugate prior for multiple extended target filtering," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 56, no. 1, pp. 208–225, Jun. 2020.

[29] F. Meyer and J. L. Williams "Scalable detection and tracking of extended objects," in *Proc. Int. Conf. Acoust., Speech Signal Process. 2020*, 2020, pp. 8916–8920.

[30] K. Granström, C. Lundquist, and O. Orguner "Extended target tracking using a Gaussian-mixture PHD filter," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 48, no. 4, pp. 3268–3286, Oct. 2012.

[31] K. Granström and M. Baum "Extended object tracking: Introduction, overview and applications," *J. Adv. Inf. Fusion*, vol. 12, Dec. 2017.

[32] D. Koller and N. Friedmann *Probabilistic Graphical Models: Principles and Techniques*. Cambridge, MA, USA: MIT Press, 2009.

[33] L. Wielandner, A. Venus, T. Wilding, and E. Leitinger "Multipath-based SLAM with multiple-measurement data association," in *Proc. 26th Int. Conf. Inf. Fusion*, 2023, pp. 1–8.

[34] S. M. Kay *Fundamentals of Statistical Signal Processing: Detection Theory*. Upper Saddle River, NJ, USA: Prentice-Hall, 1998.

[35] Y. Bar-Shalom, P. K. Willett, and X. Tian *Tracking and Data Fusion: A Handbook of Algorithms*. Storrs, CT, USA: Yaakov Bar-Shalom, 2011.

[36] T. Wilding, E. Leitinger, U. Mühlmann, and K. Witrisal "Modeling human body influence on UWB channels," in *Proc. IEEE 31st Annu. Int. Symp. Personal, Indoor Mobile Radio Commun.*, 2020, pp. 1–6.

[37] T. Wilding, E. Leitinger, U. Muehlmann, and K. Witrisal "Statistical modeling of the human body as an extended antenna," in *Proc. 15th Eur. Conf. Antennas Propag.*, 2021, pp. 1–5.

[38] F. M. Schubert, B. H. Fleury, P. Robertson, R. Prieto-Cerdeirai, A. Steingass, and A. Lehner "Modeling of multipath propagation components caused by trees and forests," in *Proc. Fourth Eur. Conf. Antennas Propag.*, 2010, pp. 1–5.

[39] F. M. Schubert, B. H. Fleury, R. Prieto-Cerdeira, A. Steingass, and A. Lehner "A rural channel model for satellite navigation applications," in *Proc. 6th Eur. Conf. Antennas*, 2012, pp. 2431–2435.

[40] Y. Bar-Shalom, T. Kirubarajan, and X.-R. Li *Estimation with Applications to Tracking and Navigation*. New York, NY, USA: Wiley., 2002.

[41] M. Mertens, M. Ulmke, and W. Koch "Ground target tracking with RCS estimation based on signal strength measurements," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 52, no. 1, pp. 205–220, Feb. 2016.

[42] K. Witrisal, E. Leitinger, S. Hinteregger, and P. Meissner "Bandwidth scaling and diversity gain for ranging and positioning in dense multipath channels," vol. 5, no. 4, pp. 396–399, May 2016.

[43] T. Wilding, S. Grebien, E. Leitinger, U. Mühlmann, and K. Witrisal "Single-anchor, multipath-assisted indoor positioning with aliased antenna arrays," in *Proc. 52nd Asilomar Conf. Signals, Syst., Comput.*, 2018, pp. 525–531.

[44] A. Lepoutre, O. Rabaste, and F. Le Gland "Exploiting amplitude spatial coherence for multi-target particle filter in track-before-detect," in *Proc. 16th Int. Conf. Inf. Fusion*, 2013, 2013, pp. 319–326.

[45] A. Lepoutre, O. Rabaste, and F. Le Gland "Multitarget likelihood computation for track-before-detect applications with amplitude fluctuations of type Swerling 0, 1, and 3" *IEEE Trans. Aerosp. Electron. Syst.*, vol. 52, no. 3, pp. 1089–1107, Jun. 2016.

[46] A. Venus, E. Leitinger, S. Tertinek, and K. Witrisal "A graph-based algorithm for robust sequential localization exploiting multipath for obstructed-los-bias mitigation," *IEEE Trans. Wireless Commun.*, to be published, doi: 10.1109/TWC.2023.3285530.

[47] D. Lerro and Y. Bar-Shalom "Automated tracking with target amplitude information," in *Proc. Amer. Control Conf.*, 1990, pp. 2875–2880.

[48] H. V. Poor *An Introduction to Signal Detection and Estimation*, 2nd ed. New York, NY, USA: Springer, 1994.

[49] F. Kschischang, B. Frey, and H.-A Loeliger "Factor graphs and the sum-product algorithm," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.

[50] F. Meyer, O. Hlinka, H. Wymeersch, E. Riegler, and F. Hlawatsch "Distributed localization and tracking of mobile networks including noncooperative objects," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 2, no. 1, pp. 57–71, Mar. 2016.

[51] F. Meyer, P. Braca, P. Willett, and F. Hlawatsch

"A scalable algorithm for tracking an unknown number of targets using multiple sensors," *IEEE Trans. Signal Process.*, vol. 65, no. 13, pp. 3478–3493, Jul. 2017.

[52] H.-A. Loeliger "An introduction to factor graphs," *IEEE Signal Process. Mag.*, vol. 21, no. 1, pp. 28–41, Feb. 2004.

[53] D. Schuhmacher, B.-T Vo, and B.-N. Vo "A consistent metric for performance evaluation of multi-object filters," *IEEE Trans. Signal Process.*, vol. 56, no. 8, pp. 3447–3457, Aug. 2008.

[54] A. Saleh and R. Valenzuela "A statistical model for indoor multipath propagation," *IEEE J. Sel. Areas Commun.*, vol. 5, no. 2, pp. 128–137, Feb. 1987.

[55] T. Pedersen "Modeling of path arrival rate for in-room radio channels with directive antennas," *IEEE Trans. Antennas Propag.*, vol. 66, no. 9, pp. 4791–4805, 2018.

Lukas Wielandner (S'20) received the Dipl.-Ing. (M.Sc.) degree in technical physics from Graz University of Technology, Graz, Austria, in 2018. He received the Ph.D. degree in electrical engineering at the Signal Processing and Speech Communication Laboratory (SPSC), Graz University of Technology, Graz, Austria, in 2022. His research interests include localization and navigation, estimation/detection theory, inference on graphs, and iterative message passing algorithms.

Alexander Venus (S'20) received the B.Sc. and Dipl.-Ing. (M.Sc.) degrees (with highest honors) in biomedical engineering and information and communication engineering from the Graz University of Technology, Austria, in 2012 and 2015, respectively. He was a Research and Development Engineer at Anton Paar GmbH, Graz, from 2014 to 2019. He is currently a Project Assistant at the Graz University of Technology, where he is pursuing his Ph.D. degree.

His research interests include radio-based localization and navigation, statistical signal processing, estimation/detection theory, machine learning, and error bounds.

Thomas Wilding (S'17) received the B.Sc. and Dipl.-Ing. (M.Sc.) degrees in audio and electrical engineering from the University of Music and Performing Arts Graz, Graz, Austria, in 2013 and 2016, respectively, and the Ph.D. from Graz University of Technology, Graz, Austria, in 2022. He is currently a Post-doctoral Researcher at the Graz University of Technology, Graz, working on positioning, sensing, and environment learning in wireless systems.

His research interests include radio localization and navigation, graphical models, and data fusion.

Erik Leitinger (Member) received the M.Sc. and Ph.D. degrees (with highest honors) in electrical engineering from the Graz University of Technology, Graz, Austria, in 2012 and 2016, respectively. From 2016 to 2018, he was a Postdoctoral Researcher at the Department of Electrical and Information Technology, Lund University, Lund, Sweden. He is currently a University Assistant at the Graz University of Technology, Graz, Austria. He served as Cochair of the special session "Synergistic Radar Signal Processing and Tracking" at the IEEE Radar Conference in 2021. He is Coorganizer of the special issue "Graph-Based Localization and Tracking" in the *Journal of Advances in Information Fusion (JAIF)*. Dr. Leitinger received an Award of Excellence from the Federal Ministry of Science, Research and Economy (BMWFW) for his Ph.D. Thesis. He is an Erwin Schrödinger Fellow. His research interests include inference on graphs, localization and navigation, machine learning, multiagent systems, stochastic modeling and estimation of radio channels, and estimation/detection theory.

Autonomous Mapping of Underwater Objects With the Sum–Product Algorithm

DOMENICO GAGLIONE
GIOVANNI SOLDI
PAOLO BRACA

Mapping of underwater objects is usually conducted with autonomous underwater vehicles (AUVs). A standard approach in mine countermeasure (MCM) operations is to perform a two-phase reconnaissance: in the first phase, a survey mission is carried out to detect and classify the objects; in the second phase, objects are reacquired to confirm the actual presence of mines. The data acquired during this multiphase mission greatly depends on the accuracy of the AUV's navigation system. This paper proposes a graph-based mapping algorithm that takes into account the unknown AUV position, as well as the output of the classification process, and uses the sum–product algorithm (SPA) to obtain a principled and intuitive approximation of the Bayesian inference needed for object detection and estimation. The SPA-based mapping algorithm is derived in detail, and its performance is evaluated in a simulated MCM scenario.

Manuscript received March 1, 2023; revised June 27, 2023; released for publication April 4, 2024

Refereeing of this contribution was handled by Erik Leitinger.

This work was supported by the NATO Allied Command Transformation (ACT) under the DKOE project.

This paper was previously presented in part at Global Oceans 2020, Singapore–US Gulf Coast, October 2020.

D. Gaglione, G. Soldi, and P. Braca are with the NATO Centre for Maritime Research and Experimentation (CMRE), 19126 La Spezia, Italy (e-mail: domenico.gaglione@cmre.nato.int, giovanni.soldi@cmre.nato.int, paolo.braca@cmre.nato.int).

1557-6418/2023/$17.00 © 2023 JAIF

I. INTRODUCTION

Underwater mapping and surveying arise in a variety of applications, from environmental assessments and inspections [1]–[3] to marine archaeology [4] and mine countermeasure (MCM) operations [5].

These tasks are generally conducted with unmanned underwater vehicles, such as remotely operated vehicles (ROVs) and autonomous underwater vehicles (AUVs) [6]. ROVs are connected to a support ship or marine platform and are operated from above the water's surface. The cable provides power and high-speed communications, allowing the operator to guide the vehicle while receiving sensor data in quasi-real time. Even though the cable can extend over several kilometers, this can limit the maximum range of operations. AUVs, instead, are untethered and preprogrammed to perform a specific task with little or no operator interaction, and they are usually designed for long-range, high-endurance missions.

When performing mapping and surveying with AUVs, navigation information is of paramount importance. Indeed, the quality and value of the data acquired during a mission greatly depend on the accuracy of the vehicle's navigation system. The unavailability of global positioning system (GPS) technologies in the underwater environment, limited by the heavy attenuation of radio frequency signals, requires the use of other methods for vehicle localization, e.g., acoustic or inertial navigation [7]. Acoustic navigation is based on the use of external references (i.e., acoustic beacons) at known positions that provide navigation aids to the unmanned vehicle, such as relative range and bearing; however, the deployment of such acoustic beacons might be inconvenient or even unfeasible in some scenarios. Inertial navigation systems (INSs), instead, calculate the instantaneous position and orientation of the vehicle using high-frequency data from an inertial measurement unit (IMU) available on board. A typical IMU includes accelerometers and gyroscopes, and the INS provides position and orientation information by integrating the values measured by these devices. However, because of this integration, the inherent errors in the accelerometers and gyroscopes accumulate over time, resulting in position and orientation errors that increase over time [8], [9]. A performance measure for an INS is given by the inertial drift rate in position that, for current high-quality commercial INSs, is of several kilometers per hour [6]. Advanced techniques, e.g., aiding the INS with a Doppler velocity log, a pressure sensor, and magnetometers, can reduce this drift to less than 0.5% of the AUV's traveled distance [10].

The use of AUVs is widely acknowledged as beneficial in MCM scenarios since they allow to operate from a distance in safe conditions. A classic approach to minehunting is to perform a two-phase reconnaissance. During the first phase, a survey mission is carried out using an AUV equipped with a synthetic aperture sonar

(SAS) capable of detecting and classifying mine-like objects (MLOs); the AUV generally follows a *lawnmower* pattern so as to ensure full coverage of the designed area, and the classification is performed by means of automatic target recognition (ATR) techniques [11], [12]. Once potential MLOs are located, the second phase is committed to the reacquisition of the contacts with an AUV equipped with a lower-range but higher-resolution sensor [13]. This second phase is more effective as the position uncertainty of the objects detected during the first phase is limited and their classification is accurate. Therefore, the algorithm responsible for building a comprehensive map of the underwater objects—both MLOs and non-MLOs—needs to account for the inherent uncertainty of the AUV position, as well as process the ATR classification output.

The mapping algorithm can exploit the position of the detected objects to sequentially refine the estimate of the AUV position, particularly when the same area happens to be surveyed multiple times. This approach is known in the robotics literature as simultaneous localization and mapping (SLAM), and has been applied also in the underwater domain. However, due to the lack of underwater features suitable as anchor points, underwater robotic mapping has primarily been focused on structured, man-made, or confined underwater environments [14], [15]. Recently, a probability hypothesis density (PHD)-based underwater mapping algorithm has been proposed [16], [17]. The PHD filter is an example of set-type tracking algorithm in which object states and measurements are represented by random finite sets (RFSs), a formulation that is particularly convenient for addressing situations with a varying number of objects to locate, object (dis)appearance and spawning, the presence of clutter and association uncertainty, false alarms, and missed detections. However, the methods in [16], [17] do not account for the uncertain AUV position. Nevertheless, RFS-based approaches that jointly estimate the AUV position and the (mobile) object states have been presented for other applications, such as autonomous driving [18].

This paper proposes and describes a Bayesian mapping algorithm based on an emerging approach to information fusion. This approach relies on a factor graph representation of the statistical model of the underwater mapping problem—including the uncertain AUV position—and on the sum–product algorithm (SPA) to efficiently obtain a principled and intuitive approximation of the Bayesian inference needed for object detection and estimation [19], [20]. Parts of this work were presented in our conference publication [21]. This paper differs from that publication in that it extends the formulation to account for the unknown AUV position; it presents detailed derivations of the joint posterior distribution; and it presents the SPA messages in a complete and detailed manner. Note that the statistical formulation and the factor graph described in this paper are similar to those presented in [22] for multipath-based indoor SLAM and in [23] for cooperative localization and tracking using a network of sensing agents. The main difference between the current work and those cited papers is in the application: specifically, when conducting underwater mapping with an SAS sensor, not all the objects are observable at all times. Moreover, the statistical formulation herein presented integrates the output of the ATR, enabling discrimination among different object types.

The remainder of this article is organized as follows. The basic notation and nomenclature are described in the next subsection. Section II describes the problem at hand and outlines the system model. The stochastic formulation is given in Section III, while the proposed method is detailed in Section IV. Results obtained in a simulated MCM scenario are shown in Section V, and Section VI concludes the paper.

A. Notation

Throughout this paper, column vectors are denoted by boldface lower-case letters (e.g., $\boldsymbol{a}$) and matrices by boldface upper-case letters (e.g., $\boldsymbol{A}$). $\mathbf{I}$ denotes the identity matrix and $\mathbf{1}$ denotes the column vector of all ones, with the size determined by the subscript or from the context. The transpose of a matrix $\boldsymbol{A}$ is written as $\boldsymbol{A}^{\mathrm{T}}$. We write $\mathrm{diag}(a_1, \ldots, a_n)$ for an $n \times n$ diagonal matrix with diagonal entries $a_1, \ldots, a_n$. Moreover, given a sequence $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n$, the column vector stacking all the elements of the sequence is denoted as $\boldsymbol{a}_{1:n} = [\boldsymbol{a}_1^{\mathrm{T}}, \ldots, \boldsymbol{a}_n^{\mathrm{T}}]^{\mathrm{T}}$. The Euclidean norm of vector $\boldsymbol{a}$ is denoted by $\|\boldsymbol{a}\|$. For a two-dimensional (2D) vector $\boldsymbol{a}$, $\angle\boldsymbol{a}$ is the angle defined counterclockwise and such that $\angle\boldsymbol{a} = 0$ for $\boldsymbol{a} = [1, 0]^{\mathrm{T}}$. The symbol $\propto$ denotes equality up to a constant factor. Sets are denoted by calligraphic letters (e.g., $\mathcal{A}$), the Dirac delta function is denoted by $\delta(\cdot)$, and the Kronecker delta is denoted by $\delta_{a,b}$, and is equal to 1 if $a = b$, and 0 otherwise. Finally, we denote the probability mass function (pmf) of a discrete random variable or vector by $p(\cdot)$ and the probability density function (pdf) of a continuous random variable or vector by $f(\cdot)$; the latter notation will also be used for a mixed pdf/pmf of both continuous and discrete random variables or vectors.

II. PROBLEM DESCRIPTION AND SYSTEM MODEL

A. AUV State, Navigation Data, and ATR Detections

Let $\boldsymbol{s}_t$ represent the AUV state at time step $t = 1, 2, \ldots$, whose evolution is given by the following kinematic model:

$$\boldsymbol{s}_t = \boldsymbol{\varepsilon}(\boldsymbol{s}_{t-1}, \boldsymbol{u}_t), \tag{1}$$

where $\boldsymbol{u}_t$ is a driving process noise independent across t. The AUV is equipped with an on-board device, e.g., an INS, that provides at time t a noisy observation of the AUV state $\boldsymbol{s}_t$; this observation, referred to as navigation

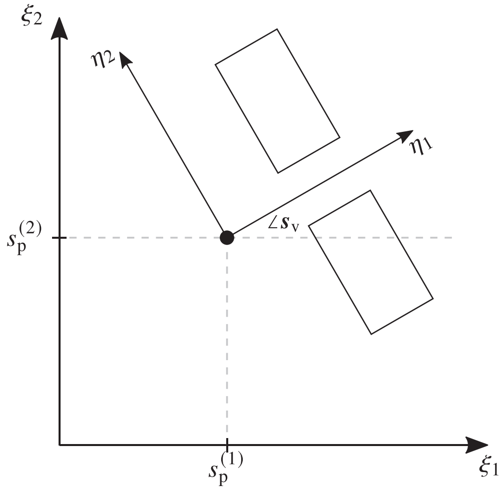


Figure 1. Illustration of the scenario (note that the time index t is omitted). The global coordinate system is (ξ_1, ξ_2). The position of the AUV (black dot) is $\boldsymbol{s}_{\mathrm{p}} = [s_{\mathrm{p}}^{(1)}, s_{\mathrm{p}}^{(2)}]^{\mathrm{T}}$, and its forward direction is given by the angle $\angle \boldsymbol{s}_{\mathrm{v}}$. The local coordinate system (η_1, η_2) is defined by the position and heading of the AUV.

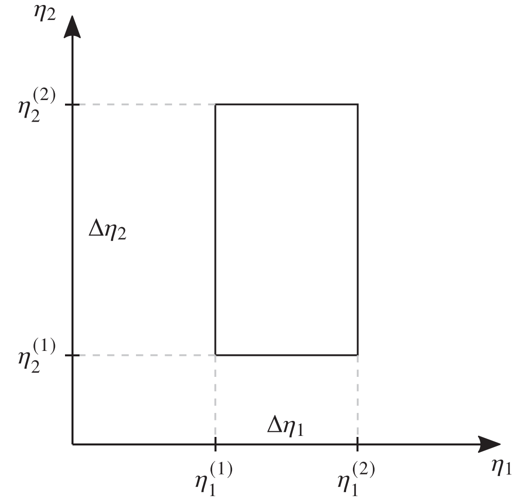


Figure 2. Geometry of the port side SAS tile in local coordinates: $(\eta_1^{(1)}, \eta_2^{(1)})$ is the location of the corner of the tile closest to the origin, i.e., the position of the AUV; $\Delta\eta_1 = \eta_1^{(2)} - \eta_1^{(1)}$ is the tile extent along η_1; $\Delta\eta_2 = \eta_2^{(2)} - \eta_2^{(1)}$ is the tile extent along η_2.

data, is modeled as

$$\boldsymbol{g}_t = \boldsymbol{\gamma}_{\mathrm{A}}(\boldsymbol{s}_t, \boldsymbol{v}_t), \tag{2}$$

where $\boldsymbol{v}_t$ is an observation process noise independent across t. To facilitate the description that follows, we consider the state $\boldsymbol{s}_t$ to be composed of the AUV's position $\boldsymbol{s}_{t,\mathrm{p}}$ and velocity $\boldsymbol{s}_{t,\mathrm{v}}$ in Cartesian coordinate, i.e., $\boldsymbol{s}_t = \big[\boldsymbol{s}_{t,\mathrm{p}}^{\mathrm{T}}, \boldsymbol{s}_{t,\mathrm{v}}^{\mathrm{T}}\big]^{\mathrm{T}}$. Nevertheless, the derivation of the proposed algorithm is general enough to accommodate a different definition of $\boldsymbol{s}_t$ that may also include additional kinematic parameters, e.g., the AUV turn rate. As illustrated in Fig. 1, the state $\boldsymbol{s}_t$ defines the AUV local coordinate system (η_1, η_2), whose origin is $\boldsymbol{s}_{t,\mathrm{p}}$ and that is rotated of an angle $\angle \boldsymbol{s}_{t,\mathrm{v}}$ in a counterclockwise direction with respect to the global coordinate system (ξ_1, ξ_2). Note that a generic point $\boldsymbol{\rho}_\xi$ in global coordinates can be converted into local coordinates as $\boldsymbol{\rho}_\eta = \boldsymbol{\theta}(\boldsymbol{\rho}_\xi; \boldsymbol{s}_t) \triangleq \boldsymbol{R}(\angle \boldsymbol{s}_{t,\mathrm{v}})\big[\boldsymbol{\rho}_\xi - \boldsymbol{s}_{t,\mathrm{p}}\big]$, where $\boldsymbol{R}(\cdot)$ is a clockwise rotation matrix.

The AUV is equipped with an SAS, a high-resolution sonar that generates acoustic images of the bottom. Such images—or SAS tiles—come in pairs, covering both port and starboard sides of the AUV, but having a coverage gap beneath. Figure 2 shows the geometry of the port side SAS tile in local coordinates (the starboard side tile is obtained by mirroring the port side tile on the η_1-axis). The SAS images are processed by an ATR algorithm that detects and classifies the features of interest, providing the location within the tile (i.e., in local coordinates) of each detection and the probabilities of such detection of being generated by an object of class $c \in \{1, \ldots, C\}$, where C is the total number of classes. Specifically, the number of detections (or measurements) extracted by the ATR algorithm from the SAS tiles at time t is m_t. The location of the mth measurement in local coordinates is represented by the vector $\boldsymbol{\ell}_{m,t} = \big[\ell_{m,t}^{(1)}, \ell_{m,t}^{(2)}\big]^{\mathrm{T}}$. The probability of the mth measurement of being generated by an object of class $c \in \{1, \ldots, C\}$ is referred to as $\pi_{m,t}^{(c)}$, and the sum of these probabilities is 1, i.e., $\sum_{j=1}^{C} \pi_{m,t}^{(j)} = 1$. Since any one of these C probabilities can be derived from the other $C - 1$, the ATR algorithm actually provides the vector $\boldsymbol{\pi}_{m,t} \triangleq \big[\pi_{m,t}^{(1)}, \ldots, \pi_{m,t}^{(C-1)}\big]^{\mathrm{T}}$, i.e., the vector stacking all the probabilities but $\pi_{m,t}^{(C)}$. Indeed, $\pi_{m,t}^{(C)}$ can then be calculated from $\boldsymbol{\pi}_{m,t}$ as $\pi_{m,t}^{(C)} = 1 - \mathbf{1}_{C-1}^{\mathrm{T}} \boldsymbol{\pi}_{m,t}$. Concretely, if the ATR algorithm distinguished between MLOs and non-MLOs, then $C = 2$, and $\pi_{m,t}^{(1)}$ and $\pi_{m,t}^{(2)} = 1 - \pi_{m,t}^{(1)}$ would be, respectively, the probability that the mth measurement is generated by an MLO, and the probability that the mth measurement is generated by a non-MLO. Unlike the approaches presented in [24], [25], here the ATR algorithm does not distinguish between object- and clutter-generated measurements. For convenience, we define the vector of the mth measurement at time t as $\boldsymbol{z}_{m,t} \triangleq [\boldsymbol{\ell}_{m,t}^{\mathrm{T}}, \boldsymbol{\pi}_{m,t}^{\mathrm{T}}]^{\mathrm{T}}$, and the vector of all the measurements extracted at time t as $\boldsymbol{z}_t \triangleq [\boldsymbol{z}_{1,t}^{\mathrm{T}}, \ldots, \boldsymbol{z}_{m_t,t}^{\mathrm{T}}]^{\mathrm{T}}$.

B. Potential Object States

As done in [23], [26], we account for a time-varying unknown number of objects by introducing the concept of *potential object* (PO). The number of POs at time t is k_t; the existence of PO $k \in \{1, \ldots, k_t\}$ at time t is indicated by the binary variable $r_{k,t} \in \{0, 1\}$, i.e., $r_{k,t} = 1$ if the PO exists and $r_{k,t} = 0$ otherwise. Position and class of PO k are denoted by $\boldsymbol{x}_{k,t}$ and $\tau_{k,t}$, respectively, and are formally considered also if $r_{k,t} = 0$. We combine the position, class, and existence variables of PO k into the state vector $\boldsymbol{y}_{k,t} \triangleq [\boldsymbol{x}_{k,t}^{\mathrm{T}}, \tau_{k,t}, r_{k,t}]^{\mathrm{T}}$, and define the joint vector of all the POs at time t as $\boldsymbol{y}_t \triangleq [\boldsymbol{y}_{1,t}^{\mathrm{T}}, \ldots, \boldsymbol{y}_{k_t,t}^{\mathrm{T}}]^{\mathrm{T}}$. We observe that the position $\boldsymbol{x}_{k,t}$ and class $\tau_{k,t}$ of any nonexisting PO (i.e., for which $r_{k,t} = 0$) are obviously irrelevant; thus, all the pdfs defined for the PO states, i.e., $f(\boldsymbol{y}_{k,t}) = f(\boldsymbol{x}_{k,t}, \tau_{k,t}, r_{k,t})$, are such that

$$f(\boldsymbol{x}_{k,t}, \tau_{k,t}, r_{k,t} = 0) = f_{k,t}\, f_{\mathrm{D}}(\boldsymbol{x}_{k,t}, \tau_{k,t}),$$

where $f_{k,t} \in [0, 1]$ is a constant and $f_{\mathrm{D}}(\boldsymbol{x}_{k,t}, \tau_{k,t})$ is an arbitrary *dummy* pdf.

Each PO at time t is either a *new* PO or a *legacy* PO. New POs model those objects that are detected for the first time by the ATR algorithm at time t. Each new PO corresponds to a measurement $\boldsymbol{z}_{m,t}$; therefore, the number of new POs at time t is m_t. The state of a new PO is denoted by $\overline{\boldsymbol{y}}_{m,t} \triangleq [\overline{\boldsymbol{x}}_{m,t}^{\mathrm{T}}, \overline{\tau}_{m,t}, \overline{r}_{m,t}]^{\mathrm{T}}$, $m \in \{1, \ldots, m_t\}$, and $\overline{r}_{m,t} = 1$ thus means that measurement m was generated by an object that was never detected before, namely, a *newly detected* object; we define the joint state vector of all new POs introduced at time t as $\overline{\boldsymbol{y}}_t \triangleq [\overline{\boldsymbol{y}}_{1,t}^{\mathrm{T}}, \ldots, \overline{\boldsymbol{y}}_{m_t,t}^{\mathrm{T}}]^{\mathrm{T}}$. A legacy PO is a PO that has already been introduced at any previous time $t' < t$. We indicate with $\underline{\boldsymbol{y}}_{k,t} \triangleq [\underline{\boldsymbol{x}}_{k,t}^{\mathrm{T}}, \underline{\tau}_{k,t}, \underline{r}_{k,t}]^{\mathrm{T}}$ the state of legacy PO $k \in \{1, \ldots, k_{t-1}\}$, and with $\underline{\boldsymbol{y}}_t \triangleq [\underline{\boldsymbol{y}}_{1,t}^{\mathrm{T}}, \ldots, \underline{\boldsymbol{y}}_{k_{t-1},t}^{\mathrm{T}}]^{\mathrm{T}}$ the joint legacy PO state vector. The k_{t-1} legacy POs and the m_t new POs form the set of $k_t = k_{t-1} + m_t$ POs at time t, i.e., $\boldsymbol{y}_t \triangleq [\underline{\boldsymbol{y}}_t^{\mathrm{T}}, \overline{\boldsymbol{y}}_t^{\mathrm{T}}]$, which will then become legacy POs at time $t + 1$. Note that using this mechanism, the number of POs grows indefinitely over time. To keep a tractable number of POs, a suboptimal pruning step is performed once all the measurements at time t are processed; details are provided in Section IV-F.

The joint PO state $\boldsymbol{y}_t$ evolves over time according to a first-order Markov model, and each PO state vector $\boldsymbol{y}_{k,t}$ evolves independently [23], [26]. Moreover, recalling that for each PO at time $t - 1$, there is one legacy PO at time t, the joint PO state transition pdf is

$$f(\underline{\boldsymbol{y}}_t|\boldsymbol{y}_{t-1}) = \prod_{k=1}^{k_{t-1}} f(\underline{\boldsymbol{y}}_{k,t}|\boldsymbol{y}_{k,t-1}). \tag{3}$$

Note that the number of POs at time $t = 0$ is zero, i.e., $k_0 = 0$; therefore, for $t = 1$, the transition pdf in (3) is the result of an empty product, that is, $f(\underline{\boldsymbol{y}}_1|\boldsymbol{y}_0) = 1$. Furthermore, assuming that given the position and existence of PO k at time $t-1$ (i.e., $\boldsymbol{x}_{k,t-1}$ and $r_{k,t-1}$), the position and existence of legacy PO k at time t (i.e., $\underline{\boldsymbol{x}}_{k,t}$ and $\underline{r}_{k,t}$) are conditionally independent of the PO class at time $t - 1$ and the legacy PO class at time t (i.e., $\tau_{k,t-1}$ and $\underline{\tau}_{k,t}$); and that given $\tau_{k,t-1}$, the legacy PO class $\underline{\tau}_{k,t}$ is conditionally independent of $\boldsymbol{x}_{k,t-1}$ and $r_{k,t-1}$, we obtain

$$\begin{aligned} f(\underline{\boldsymbol{y}}_{k,t}|\boldsymbol{y}_{k,t-1}) &= f(\underline{\boldsymbol{x}}_{k,t}, \underline{\tau}_{k,t}, \underline{r}_{k,t}|\boldsymbol{x}_{k,t-1}, \tau_{k,t-1}, r_{k,t-1}) \\ &= f(\underline{\boldsymbol{x}}_{k,t}, \underline{r}_{k,t}|\underline{\tau}_{k,t}, \boldsymbol{x}_{k,t-1}, \tau_{k,t-1}, r_{k,t-1}) \\ &\quad \times p(\underline{\tau}_{k,t}|\boldsymbol{x}_{k,t-1}, \tau_{k,t-1}, r_{k,t-1}) \\ &= f(\underline{\boldsymbol{x}}_{k,t}, \underline{r}_{k,t}|\boldsymbol{x}_{k,t-1}, r_{k,t-1})\, p(\underline{\tau}_{k,t}|\tau_{k,t-1}). \end{aligned} \tag{4}$$

Since in the considered scenario the objects, hence the POs, are stationary (i.e., they cannot leave the surveilled area), the pdf $f(\underline{\boldsymbol{x}}_{k,t}, \underline{r}_{k,t}|\boldsymbol{x}_{k,t-1}, r_{k,t-1})$ is defined as follows: if PO k does not exist at time $t-1$, i.e., if $r_{k,t-1} = 0$, then it cannot exist as legacy PO at time t; if it does exist at time $t - 1$, i.e., if $r_{k,t-1} = 1$, then it exists as legacy PO at time t and its position $\underline{\boldsymbol{x}}_{k,t}$ is distributed according to the transition pdf $f(\underline{\boldsymbol{x}}_{k,t}|\boldsymbol{x}_{k,t-1}) = \delta(\underline{\boldsymbol{x}}_{k,t} - \boldsymbol{x}_{k,t-1})$, that is,

$$\begin{aligned} &f(\underline{\boldsymbol{x}}_{k,t}, \underline{r}_{k,t}|\boldsymbol{x}_{k,t-1}, r_{k,t-1}) \\ &\quad = \begin{cases} (1 - \underline{r}_{k,t}) f_{\mathrm{D}}(\underline{\boldsymbol{x}}_{k,t}), & r_{k,t-1} = 0, \\ \underline{r}_{k,t}\, f(\underline{\boldsymbol{x}}_{k,t}|\boldsymbol{x}_{k,t-1}), & r_{k,t-1} = 1, \end{cases} \end{aligned} \tag{5}$$

where $f_{\mathrm{D}}(\underline{\boldsymbol{x}}_{k,t}) = \sum_{\underline{\tau}_{k,t}=1}^{C} f_{\mathrm{D}}(\underline{\boldsymbol{x}}_{k,t}, \underline{\tau}_{k,t})$. Additionally, given that the class of an object cannot change over time, the pmf $p(\underline{\tau}_{k,t}|\tau_{k,t-1}) = \delta_{\underline{\tau}_{k,t}, \tau_{k,t-1}}$.

C. ATR Measurement Model

The probability that PO k is detected by the ATR algorithm at time t, i.e., that PO k generates a measurement $\boldsymbol{z}_{m,t}$, is function of the PO position $\boldsymbol{x}_{k,t}$ and class $\tau_{k,t}$, as well as of the AUV state $\boldsymbol{s}_t$, and is denoted by $P_{\mathrm{d}}(\boldsymbol{x}_{k,t}, \tau_{k,t}, \boldsymbol{s}_t)$. As an example, the probability of detection could be nonzero only inside the SAS tiles, i.e.,

$$\begin{aligned} &P_{\mathrm{d}}(\boldsymbol{x}_{k,t}, \tau_{k,t}, \boldsymbol{s}_t) \\ &\quad \triangleq \begin{cases} p_{\mathrm{d}}(\tau_{k,t}) & \text{if } \boldsymbol{\theta}(\boldsymbol{x}_{k,t}; \boldsymbol{s}_t) \text{ is within the SAS tiles,} \\ 0 & \text{otherwise,} \end{cases} \end{aligned}$$

where $p_{\mathrm{d}}(\tau_{k,t})$ is a class-dependent probability of detection. Alternatively, $P_{\mathrm{d}}(\boldsymbol{x}_{k,t}, \tau_{k,t}, \boldsymbol{s}_t)$ could also account for some environmental characteristics, such as the bottom type [27].

The statistical dependency of a PO-generated measurement $\boldsymbol{z}_{m,t}$ on the PO position $\boldsymbol{x}_{k,t}$ and class $\tau_{k,t}$, and on the AUV state $\boldsymbol{s}_t$, is described by the likelihood function $f(\boldsymbol{z}_{m,t}|\boldsymbol{x}_{k,t}, \tau_{k,t}, \boldsymbol{s}_t) = f(\boldsymbol{\ell}_{m,t}, \boldsymbol{\pi}_{m,t}|\boldsymbol{x}_{k,t}, \tau_{k,t}, \boldsymbol{s}_t)$. Following [24] and assuming that $\boldsymbol{\ell}_{m,t}$ is conditionally independent of $\boldsymbol{\pi}_{m,t}$ and $\tau_{k,t}$ given $\boldsymbol{x}_{k,t}$ and $\boldsymbol{s}_t$, and that $\boldsymbol{\pi}_{m,t}$ is conditionally independent of $\boldsymbol{x}_{k,t}$ and $\boldsymbol{s}_t$ given $\tau_{k,t}$, the likelihood function can be factorized as

$$\begin{aligned} f(\boldsymbol{z}_{m,t}|\boldsymbol{x}_{k,t}, \tau_{k,t}, \boldsymbol{s}_t) &= f(\boldsymbol{\ell}_{m,t}, \boldsymbol{\pi}_{m,t}|\boldsymbol{x}_{k,t}, \tau_{k,t}, \boldsymbol{s}_t) \\ &= f(\boldsymbol{\ell}_{m,t}|\boldsymbol{\pi}_{m,t}, \boldsymbol{x}_{k,t}, \tau_{k,t}, \boldsymbol{s}_t)\, f(\boldsymbol{\pi}_{m,t}|\boldsymbol{x}_{k,t}, \tau_{k,t}, \boldsymbol{s}_t) \\ &= f(\boldsymbol{\ell}_{m,t}|\boldsymbol{x}_{k,t}, \boldsymbol{s}_t)\, f(\boldsymbol{\pi}_{m,t}|\tau_{k,t}). \end{aligned} \tag{6}$$

The first factor in (6), i.e., $f(\boldsymbol{\ell}_{m,t}|\boldsymbol{x}_{k,t}, \boldsymbol{s}_t)$, is determined by the ATR measurement model, defined as

$$\boldsymbol{\ell}_{m,t} = \boldsymbol{\gamma}_{\mathrm{O}}\big(\boldsymbol{\theta}(\boldsymbol{x}_{k,t}; \boldsymbol{s}_t), \boldsymbol{\omega}_{m,t}\big),$$

and by the statistics of the ATR measurement noise $\boldsymbol{\omega}_{m,t}$, assumed independent across m and t. The second factor in (6), i.e., $f(\boldsymbol{\pi}_{m,t}|\tau_{k,t})$, is modeled according to a Dirichlet distribution with vector parameter $\boldsymbol{\alpha}_{\tau_{k,t}} \triangleq$

$\left[\alpha^{(1)}_{\tau_{k,t}}, \ldots, \alpha^{(C)}_{\tau_{k,t}}\right]^{\mathrm{T}}$, that is,

$$f(\boldsymbol{\pi}_{m,t}|\tau_{k,t}=c) = \frac{1}{B(\boldsymbol{\alpha}_c)} \prod_{j=1}^{C-1} \pi_{m,t}^{(j)\left(\alpha_c^{(j)}-1\right)} \times \left(1-\mathbf{1}_{C-1}^{\mathrm{T}}\boldsymbol{\pi}_{m,t}\right)^{\left(\alpha_c^{(C)}-1\right)}, \tag{7}$$

where $B(\cdot)$ is the multivariate beta function.

A clutter-generated measurement (i.e., a false-alarm) is statistically described by the pdf $f_{\mathrm{FA}}(\boldsymbol{z}_{m,t}) = f_{\mathrm{FA}}(\boldsymbol{\ell}_{m,t}, \boldsymbol{\pi}_{m,t})$ that, assuming the independence between $\boldsymbol{\ell}_{m,t}$ and $\boldsymbol{\pi}_{m,t}$, can be factorized as $f_{\mathrm{FA}}(\boldsymbol{z}_{m,t}) = f_0(\boldsymbol{\ell}_{m,t})g_0(\boldsymbol{\pi}_{m,t})$. The number of clutter-generated measurements at each time t within both the port side and starboard side tiles is assumed Poisson distributed with mean μ_0.

D. Data Association

The measurements $\boldsymbol{z}_{m,t}$, $m \in \{1, \ldots, m_t\}$, have unknown origins, namely, it is unknown if a given measurement is generated from clutter or from a PO, and from which PO. Here, we consider the *point-object assumption*, stating that, at each time t, a measurement $\boldsymbol{z}_{m,t}$ originates either from a legacy PO, or from a new PO, or from clutter, and it cannot originate from more than one source (legacy POs, new POs, or clutter) simultaneously. Conversely, each PO (either legacy or new) can generate at most one measurement at time t [28]. Following [23], [26], the association between the k_{t-1} legacy POs, m_t new POs, and m_t measurements can be mathematically described by introducing: (i) the set $\mathcal{N}_t$ of measurements generated by newly detected objects at time t, that is, $\mathcal{N}_t \triangleq \left\{m \in \{1, \ldots, m_t\} : \bar{r}_{m,t} = 1\right\}$; (ii) the legacy PO-oriented association vector $\boldsymbol{a}_t \triangleq [a_{1,t}, \ldots, a_{k_{t-1},t}]^{\mathrm{T}}$; and (iii) the measurement-oriented association vector $\boldsymbol{b}_t \triangleq [b_{1,t}, \ldots, b_{m_t,t}]^{\mathrm{T}}$. Specifically, $a_{k,t}$ is defined as $m \in \{1, \ldots, m_t\}$ if legacy PO k generates measurement m, and as 0 if legacy PO k does not generate any measurement. Similarly, $b_{m,t}$ is defined as $k \in \{1, \ldots, k_{t-1}\}$ if measurement m originates from legacy PO k and as 0 if measurement m does not originate from any legacy PO. Note that $b_{m,t} = 0$ implies that measurement m either is clutter-generated or originates from a newly detected object. Then, the point-object assumption can be expressed by the indicator function $\Phi(\boldsymbol{a}_t, \boldsymbol{b}_t)$, defined as [23]

$$\Phi(\boldsymbol{a}_t, \boldsymbol{b}_t) \triangleq \Psi(\boldsymbol{a}_t, \boldsymbol{b}_t) \prod_{m \in \mathcal{N}_t} \Gamma(b_{m,t}), \tag{8}$$

where

$$\Gamma\left(b_{m,t}\right) \triangleq \begin{cases} 0 & b_{m,t} \in \{1, \ldots, k_{t-1}\}, \\ 1 & b_{m,t} = 0, \end{cases} \tag{9}$$

and

$$\Psi(\boldsymbol{a}_t, \boldsymbol{b}_t) \triangleq \prod_{k=1}^{k_{t-1}} \prod_{m=1}^{m_t} \psi(a_{k,t}, b_{m,t}), \tag{10}$$

with

$$\psi(a_{k,t}, b_{m,t}) \triangleq \begin{cases} 0 & a_{k,t} = m \text{ and } b_{m,t} \neq k\,, \\ & \quad \text{or } a_{k,t} \neq m \text{ and } b_{m,t} = k\,, \\ 1 & \text{otherwise}\,. \end{cases}$$

Note that, since the product in (8) is over the set $\mathcal{N}_t$, the indicator function $\Phi(\boldsymbol{a}_t, \boldsymbol{b}_t)$ formally depends also on the new PO existence variables $\bar{r}_{m,t}, m \in \{1, \ldots, m_t\}$. Expression (8) can be easily explained as follows: valid associations described by $\boldsymbol{a}_t$, $\boldsymbol{b}_t$, and the new PO existence variables $\bar{r}_{m,t}, m \in \{1, \ldots, m_t\}$, are those for which $\Phi(\boldsymbol{a}_t, \boldsymbol{b}_t) = 1$; and we note that $\Psi(\boldsymbol{a}_t, \boldsymbol{b}_t)$ is 0 if a measurement is associated with two or more legacy POs (and, vice versa, if a legacy PO is associated with two or more measurements), and 1 otherwise; and that the product over $m \in \mathcal{N}_t$ of $\Gamma(b_{m,t})$ is 0 if any measurement generated by a new PO is also associated with a legacy PO, and 1 otherwise.

III. STOCHASTIC PROBLEM FORMULATION

A. Joint Posterior pdf

The objective of the mapping of underwater objects is to determine if a PO exists and estimate its position and class given all AUV navigation data and all measurements extracted by the ATR algorithm up to time t, i.e., given $\boldsymbol{g}_{1:t}$ and $\boldsymbol{z}_{1:t}$. In the Bayesian framework here described, this essentially consists in evaluating for each PO $k \in \{1, \ldots, k_t\}$ the posterior marginal pmf $p(r_{k,t}|\boldsymbol{g}_{1:t}, \boldsymbol{z}_{1:t})$, used for existence declaration,[1] and the conditional marginal pdf $f(\boldsymbol{x}_{k,t}|r_{k,t} = 1, \boldsymbol{g}_{1:t}, \boldsymbol{z}_{1:t})$ and pmf $p(\tau_{k,t}|r_{k,t} = 1, \boldsymbol{g}_{1:t}, \boldsymbol{z}_{1:t})$, used for position and class estimation, respectively. These marginal posterior distributions can be calculated by simple elementary operations—including marginalization—from the joint posterior distribution $f(\boldsymbol{y}_{0:t}, \boldsymbol{s}_{0:t}, \boldsymbol{a}_{1:t}, \boldsymbol{b}_{1:t}|\boldsymbol{g}_{1:t}, \boldsymbol{z}_{1:t}) = f(\boldsymbol{x}_{0:t}, \boldsymbol{\tau}_{0:t}, \boldsymbol{r}_{0:t}, \boldsymbol{s}_{0:t}, \boldsymbol{a}_{1:t}, \boldsymbol{b}_{1:t}|\boldsymbol{g}_{1:t}, \boldsymbol{z}_{1:t})$. Here, the sequence of AUV states $\boldsymbol{s}_{0:t}$ is considered as nuisance parameters to be marginalized out, where $\boldsymbol{s}_0$ is the state at time $t = 0$ whose prior distribution $f(\boldsymbol{s}_0)$ is known; $\boldsymbol{y}_0$ is introduced for mathematical convenience, since at time $t = 0$ the number of POs is zero, i.e., $k_0 = 0$. This joint posterior pdf can be factorized as (details are provided in the Appendix)

$$\begin{aligned} f\left(\boldsymbol{y}_{0:t}, \boldsymbol{s}_{0:t}, \boldsymbol{a}_{1:t}, \boldsymbol{b}_{1:t}\middle|\boldsymbol{g}_{1:t}, \boldsymbol{z}_{1:t}\right) &\propto f(\boldsymbol{s}_0) f(\boldsymbol{y}_0) \\ &\times \prod_{t'=1}^{t} f\left(\underline{\boldsymbol{y}}_{t'}\middle|\boldsymbol{y}_{t'-1}\right) f\left(\boldsymbol{s}_{t'}\middle|\boldsymbol{s}_{t'-1}\right) f\left(\boldsymbol{g}_{t'}\middle|\boldsymbol{s}_{t'}\right) \\ &\times f\left(\overline{\boldsymbol{y}}_{t'}, \boldsymbol{a}_{t'}, \boldsymbol{b}_{t'}, m_{t'}\middle|\underline{\boldsymbol{y}}_{t'}, \boldsymbol{s}_{t'}\right) f\left(\boldsymbol{z}_{t'}\middle|\boldsymbol{y}_{t'}, \boldsymbol{s}_{t'}, \boldsymbol{a}_{t'}, m_{t'}\right), \end{aligned} \tag{11}$$

[1]The existence of PO k is confirmed if $p(r_{k,t} = 1|\boldsymbol{g}_{1:t}, \boldsymbol{z}_{1:t})$ is above an existence threshold E_{th} [29, Ch. 2].

where $f(\underline{\boldsymbol{y}}_t|\boldsymbol{y}_{t-1})$ is defined in (3), $f(\boldsymbol{s}_t|\boldsymbol{s}_{t-1})$ derives from the kinematic model in (1), and $f(\boldsymbol{g}_t|\boldsymbol{s}_t)$ is the likelihood determined by the navigation data model in (2). Following the derivations in [26], next we provide expressions for the prior data association pdf $f(\overline{\boldsymbol{y}}_t, \boldsymbol{a}_t, \boldsymbol{b}_t, m_t|\underline{\boldsymbol{y}}_t, \boldsymbol{s}_t)$ and the measurement likelihood $f(\boldsymbol{z}_t|\boldsymbol{y}_t, \boldsymbol{s}_t, \boldsymbol{a}_t, m_t)$.

B. Prior Data Association pdf

By considering the point-object assumption (cf. Section II-D), and assuming that positions and classes of legacy POs and new POs at time t are independent, the pdf $f(\overline{\boldsymbol{y}}_t, \boldsymbol{a}_t, \boldsymbol{b}_t, m_t|\underline{\boldsymbol{y}}_t, \boldsymbol{s}_t)$ can be expressed as

$$f(\overline{\boldsymbol{y}}_t, \boldsymbol{a}_t, \boldsymbol{b}_t, m_t|\underline{\boldsymbol{y}}_t, \boldsymbol{s}_t) \propto \Psi(\boldsymbol{a}_t, \boldsymbol{b}_t) \times \prod_{k=1}^{k_{t-1}} q_1(\underline{\boldsymbol{y}}_{k,t}, a_{k,t}, \boldsymbol{s}_t; m_t) \prod_{m=1}^{m_t} h_1(\overline{\boldsymbol{y}}_{m,t}, b_{m,t}). \quad (12)$$

The derivation of this pdf closely follows the derivation of the pdf in [26, eq. (60)] and is thus omitted. The proportionality is due to a constant factor that only depends on the number of measurements m_t, the indicator function $\Psi(\boldsymbol{a}_t, \boldsymbol{b}_t)$ is defined in (10), and the functions $q_1(\cdot)$ and $h_1(\cdot)$—representing the contributions to the prior data association pdf of the legacy and new POs, respectively—are provided in the following. The function $q_1(\underline{\boldsymbol{y}}_{k,t}, a_{k,t}, \boldsymbol{s}_t; m_t) = q_1(\underline{\boldsymbol{x}}_{k,t}, \underline{\tau}_{k,t}, \underline{r}_{k,t}, a_{k,t}, \boldsymbol{s}_t; m_t)$ is defined for $\underline{r}_{k,t} = 1$ as

$$q_1(\underline{\boldsymbol{x}}_{k,t}, \underline{\tau}_{k,t}, \underline{r}_{k,t}=1, a_{k,t}, \boldsymbol{s}_t; m_t) \triangleq \begin{cases} \dfrac{P_{\mathrm{d}}(\underline{\boldsymbol{x}}_{k,t}, \underline{\tau}_{k,t}, \boldsymbol{s}_t)}{\mu_0} & a_{k,t} \in \{1, \ldots, m_t\}, \\ 1 - P_{\mathrm{d}}(\underline{\boldsymbol{x}}_{k,t}, \underline{\tau}_{k,t}, \boldsymbol{s}_t) & a_{k,t} = 0, \end{cases} \quad (13)$$

and for $\underline{r}_{k,t} = 0$ as

$$q_1(\underline{\boldsymbol{x}}_{k,t}, \underline{\tau}_{k,t}, \underline{r}_{k,t}=0, a_{k,t}, \boldsymbol{s}_t; m_t) \triangleq \delta_{a_{k,t},0}. \quad (14)$$

The function $h_1(\overline{\boldsymbol{y}}_{m,t}, b_{m,t}) = h_1(\overline{\boldsymbol{x}}_{m,t}, \overline{\tau}_{m,t}, \overline{r}_{m,t}, b_{m,t})$ is defined for $\overline{r}_{m,t} = 1$ as

$$h_1(\overline{\boldsymbol{x}}_{m,t}, \overline{\tau}_{m,t}, \overline{r}_{m,t}=1, b_{m,t}) \triangleq \Gamma(b_{m,t}) \frac{\mu_{\mathrm{N}}}{\mu_0} f_{\mathrm{N}}(\overline{\boldsymbol{x}}_{m,t}, \overline{\tau}_{m,t}) = \begin{cases} 0 & b_{m,t} \in \{1, \ldots, k_{t-1}\}, \\ \dfrac{\mu_{\mathrm{N}}}{\mu_0} f_{\mathrm{N}}(\overline{\boldsymbol{x}}_{m,t}, \overline{\tau}_{m,t}) & b_{m,t} = 0, \end{cases} \quad (15)$$

and for $\overline{r}_{m,t} = 0$ as

$$h_1(\overline{\boldsymbol{x}}_{m,t}, \overline{\tau}_{m,t}, \overline{r}_{m,t}=0, b_{m,t}) \triangleq f_{\mathrm{D}}(\overline{\boldsymbol{x}}_{m,t}, \overline{\tau}_{m,t}). \quad (16)$$

Here, μ_{N} is the mean number of newly detected object at each time t (assumed Poisson distributed) and $f_{\mathrm{N}}(\overline{\boldsymbol{x}}_{m,t}, \overline{\tau}_{m,t})$ is the prior distribution of position and class of a new PO that, assuming the independence between $\overline{\boldsymbol{x}}_{m,t}$ and $\overline{\tau}_{m,t}$, can be factorized as $f_{\mathrm{N}}(\overline{\boldsymbol{x}}_{m,t}, \overline{\tau}_{m,t}) = f_{\mathrm{n}}(\overline{\boldsymbol{x}}_{m,t}) f_{\mathrm{n}}(\overline{\tau}_{m,t})$. Note that the function $h_1(\cdot)$ incorporates the indicator function $\Gamma(\cdot)$ defined in (9), and that the combined use in (12) of the functions $\Psi(\boldsymbol{a}_t, \boldsymbol{b}_t)$ and $h_1(\cdot)$ describes the point-object assumption as done by the indicator function $\Phi(\cdot)$ defined in (8).

C. ATR Measurements Likelihood

By considering the point-object assumption (cf. Section II-D) and assuming that PO-generated measurements and clutter-generated measurements are independent, the measurement likelihood $f(\boldsymbol{z}_t|\boldsymbol{y}_t, \boldsymbol{s}_t, \boldsymbol{a}_t, m_t) = f(\boldsymbol{z}_t|\overline{\boldsymbol{y}}_t, \underline{\boldsymbol{y}}_t, \boldsymbol{s}_t, \boldsymbol{a}_t, m_t)$ can be expressed as

$$f(\boldsymbol{z}_t|\overline{\boldsymbol{y}}_t, \underline{\boldsymbol{y}}_t, \boldsymbol{s}_t, \boldsymbol{a}_t, m_t) \propto \prod_{k=1}^{k_{t-1}} q_2(\underline{\boldsymbol{y}}_{k,t}, a_{k,t}, \boldsymbol{s}_t; \boldsymbol{z}_t) \times \prod_{m=1}^{m_t} h_2(\overline{\boldsymbol{y}}_{m,t}, \boldsymbol{s}_t; \boldsymbol{z}_{m,t}). \quad (17)$$

The derivation of the likelihood in (17) closely follows the derivation of the likelihood in [26, eq. (64)] and is thus omitted. The proportionality is due to a constant factor that only depends on the measurement vector $\boldsymbol{z}_t$, and the functions $q_2(\cdot)$ and $h_2(\cdot)$—embedding the measurement likelihoods related to the legacy and new POs, respectively—are provided in the following. The function $q_2(\underline{\boldsymbol{y}}_{k,t}, a_{k,t}, \boldsymbol{s}_t; \boldsymbol{z}_t) = q_2(\underline{\boldsymbol{x}}_{k,t}, \underline{\tau}_{k,t}, \underline{r}_{k,t}, a_{k,t}, \boldsymbol{s}_t; \boldsymbol{z}_t)$ is defined for $\underline{r}_{k,t} = 1$ as

$$q_2(\underline{\boldsymbol{x}}_{k,t}, \underline{\tau}_{k,t}, \underline{r}_{k,t}=1, a_{k,t}, \boldsymbol{s}_t; \boldsymbol{z}_t) \triangleq \begin{cases} \dfrac{f(\boldsymbol{z}_{m,t}|\underline{\boldsymbol{x}}_{k,t}, \underline{\tau}_{k,t}, \boldsymbol{s}_t)}{f_{\mathrm{FA}}(\boldsymbol{z}_{m,t})} & a_{k,t} \in \{1, \ldots, m_t\}, \\ 1 & a_{k,t} = 0, \end{cases} \quad (18)$$

and for $\underline{r}_{k,t} = 0$ as

$$q_2(\underline{\boldsymbol{x}}_{k,t}, \underline{\tau}_{k,t}, \underline{r}_{k,t}=0, a_{k,t}, \boldsymbol{s}_t; \boldsymbol{z}_t) \triangleq 1\,. \quad (19)$$

The function $h_2(\overline{\boldsymbol{y}}_{m,t}, \boldsymbol{s}_t; \boldsymbol{z}_{m,t}) = h_2(\overline{\boldsymbol{x}}_{m,t}, \overline{\tau}_{m,t}, \overline{r}_{m,t}, \boldsymbol{s}_t; \boldsymbol{z}_{m,t})$ is defined as

$$h_2(\overline{\boldsymbol{x}}_{m,t}, \overline{\tau}_{m,t}, \overline{r}_{m,t}, \boldsymbol{s}_t; \boldsymbol{z}_{m,t}) \triangleq \begin{cases} \dfrac{f(\boldsymbol{z}_{m,t}|\overline{\boldsymbol{x}}_{m,t}, \overline{\tau}_{m,t}, \boldsymbol{s}_t)}{f_{\mathrm{FA}}(\boldsymbol{z}_{m,t})} & \overline{r}_{m,t} = 1\,, \\ 1 & \overline{r}_{m,t} = 0\,. \end{cases} \quad (20)$$

IV. PROPOSED METHOD

A. Factor Graph and Message Scheduling

The final factorization of the joint posterior pdf $f(\boldsymbol{y}_{0:t}, \boldsymbol{s}_{0:t}, \boldsymbol{a}_{1:t}, \boldsymbol{b}_{1:t}|\boldsymbol{g}_{1:t}, \boldsymbol{z}_{1:t})$—obtained by inserting (10)

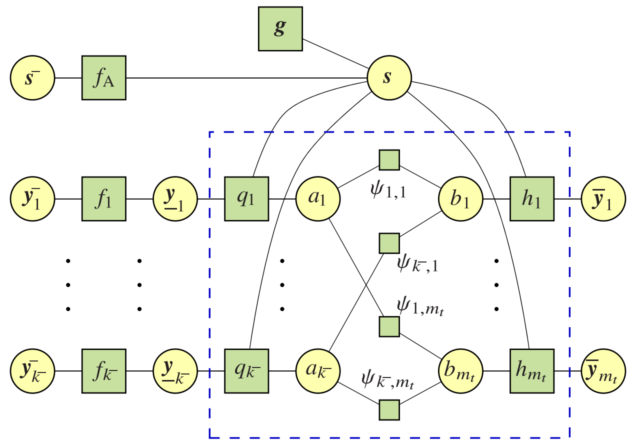


Figure 3. Factor graph representing the factorization in (21) of the joint posterior pdf $f(\boldsymbol{y}_{0:t}, \boldsymbol{s}_{0:t}, \boldsymbol{a}_{1:t}, \boldsymbol{b}_{1:t}|\boldsymbol{g}_{1:t}, \boldsymbol{z}_{1:t})$ for a single time step. Solid circles and squares represent variable nodes and factor nodes, respectively. The following short notations are used: $\boldsymbol{s}^{-} \triangleq \boldsymbol{s}_{t-1}$; $\boldsymbol{s} \triangleq \boldsymbol{s}_t$; $\boldsymbol{y}_k^{-} \triangleq \boldsymbol{y}_{k,t-1}$; $k^{-} \triangleq k_{t-1}$; $\underline{\boldsymbol{y}}_k \triangleq \underline{\boldsymbol{y}}_{k,t}$; $\overline{\boldsymbol{y}}_m \triangleq \overline{\boldsymbol{y}}_{m,t}$; $f_{\mathrm{A}} \triangleq f(\boldsymbol{s}_t|\boldsymbol{s}_{t-1})$; $f_k \triangleq f(\underline{\boldsymbol{y}}_{k,t}|\boldsymbol{y}_{k,t-1})$; $\boldsymbol{g} \triangleq f(\boldsymbol{g}_t|\boldsymbol{s}_t)$; $a_k \triangleq a_{k,t}$; $b_m \triangleq b_{m,t}$; $q_k \triangleq q(\underline{\boldsymbol{y}}_{k,t}, a_{k,t}, \boldsymbol{s}_t; \boldsymbol{z}_t)$; $h_m \triangleq h(\overline{\boldsymbol{y}}_{m,t}, b_{m,t}, \boldsymbol{s}_t; \boldsymbol{z}_{m,t})$; $\psi_{k,m} \triangleq \psi(a_{k,t}, b_{m,t})$.

into (12), and (3), (12), and (17) into (11)—is

$$
\begin{aligned}
&f\big(\boldsymbol{y}_{0:t}, \boldsymbol{s}_{0:t}, \boldsymbol{a}_{1:t}, \boldsymbol{b}_{1:t}\big|\boldsymbol{g}_{1:t}, \boldsymbol{z}_{1:t}\big) \\
&\quad\propto f\big(\boldsymbol{s}_0\big) f\big(\boldsymbol{y}_0\big) \prod_{t'=1}^{t} f\big(\boldsymbol{s}_{t'}\big|\boldsymbol{s}_{t'-1}\big) f\big(\boldsymbol{g}_{t'}\big|\boldsymbol{s}_{t'}\big) \\
&\quad\times \left[\prod_{k=1}^{k_{t'-1}} f\big(\underline{\boldsymbol{y}}_{k,t'}\big|\boldsymbol{y}_{k,t'-1}\big)\, q\big(\underline{\boldsymbol{y}}_{k,t'}, a_{k,t'}, \boldsymbol{s}_{t'}; \boldsymbol{z}_{t'}\big)\right. \\
&\quad\times \left.\prod_{m=1}^{m_{t'}} \psi\big(a_{k,t'}, b_{m,t'}\big)\right] \prod_{m'=1}^{m_{t'}} h\big(\overline{\boldsymbol{y}}_{m',t'}, b_{m',t'}, \boldsymbol{s}_{t'}; \boldsymbol{z}_{m',t'}\big),
\end{aligned} \tag{21}
$$

where $q(\cdot) \triangleq q_1(\cdot)q_2(\cdot)$ and $h(\cdot) \triangleq h_1(\cdot)h_2(\cdot)$. Direct marginalization of this joint posterior pdf for the computation of the marginal posterior pdfs/pmfs mentioned in Section III-A is generally unfeasible in reasonable time, as it requires high-dimensional integration and summation. Approximations at time t of these marginal pdfs/pmfs—called *beliefs* and referred to as $\tilde{f}_t(\cdot)$—can be efficiently obtained by applying the SPA on a factor graph [19], [20], carefully devised from the factorization in (21).

Such factor graph, illustrated for a single time step in Fig. 3, contains loops: an inner loop involving the data association variables $a_{k,t}$ and $b_{m,t}$, and an outer loop involving the AUV state $\boldsymbol{s}_t$ and factor nodes $q(\cdot)$ and $h(\cdot)$. Therefore, a scheduling of the messages is defined based on the following rules: (i) messages are not sent backward in time; (ii) iterative message passing is only performed for the data association, i.e., for the inner loop. More specifically, at each time t, the *inbound* messages—from outside to inside the blue dashed rectangle—are computed first. These messages represent the prediction of the legacy PO states and AUV states, and are computed assuming that all the *outbound* messages—from inside to outside the blue dashed rectangle—are equal to one. The inbound messages are then employed within the inner loop for data association. When all the iterations of the inner loop are performed, the outbound messages are calculated and eventually used to compute the beliefs of the PO states. Next, we provide expressions of the messages combining the scheduling rules stated above and the generic SPA rules provided in [19]. The messages are all denoted by $\zeta_{\alpha\to\beta}(\cdot)$, where α and β are, respectively, the origin and destination nodes of the message. Moreover, we assume that the beliefs are normalized, i.e., $\int \tilde{f}_t(\alpha)\mathrm{d}\alpha = 1$.

B. Inbound Messages

The inbound messages from variable node "$\boldsymbol{s}$" to factor node "q_k", i.e., $\zeta_{\boldsymbol{s}\to q_k}(\boldsymbol{s}_t)$, and from variable node "$\boldsymbol{s}$" to factor node "h_m", i.e., $\zeta_{\boldsymbol{s}\to h_m}(\boldsymbol{s}_t)$, represent the prediction of the AUV state and its refinement with navigation data. Recalling that the inbound messages are computed assuming that the outbound messages are all equal to one, the expressions of the messages $\zeta_{\boldsymbol{s}\to q_k}(\boldsymbol{s}_t)$ and $\zeta_{\boldsymbol{s}\to h_m}(\boldsymbol{s}_t)$ coincide; for them, we use the common notation $\zeta_{\boldsymbol{s}}(\boldsymbol{s}_t)$, that is,

$$
\begin{aligned}
\zeta_{\boldsymbol{s}}(\boldsymbol{s}_t) &\triangleq \zeta_{\boldsymbol{s}\to q_k}(\boldsymbol{s}_t) = \zeta_{\boldsymbol{s}\to h_m}(\boldsymbol{s}_t) \\
&= f(\boldsymbol{g}_t|\boldsymbol{s}_t) \int \tilde{f}_{t-1}(\boldsymbol{s}_{t-1}) f(\boldsymbol{s}_t|\boldsymbol{s}_{t-1})\, \mathrm{d}\boldsymbol{s}_{t-1},
\end{aligned} \tag{22}
$$

where $\tilde{f}_{t-1}(\cdot)$ is the belief computed at previous time $t-1$, whose expression is later provided in Section IV-E. For convenience, we also introduce the following constant:

$$
\zeta_{\boldsymbol{s}}^{0} = \int \zeta_{\boldsymbol{s}}(\boldsymbol{s}_t)\, \mathrm{d}\boldsymbol{s}_t. \tag{23}
$$

The inbound message from variable node "$\underline{\boldsymbol{y}}_k$" to factor node "$q_k$", representing the prediction of the legacy PO k, is computed as follows:

$$
\begin{aligned}
&\zeta_{\underline{\boldsymbol{y}}_k\to q_k}\big(\underline{\boldsymbol{y}}_{k,t}\big) = \zeta_{\underline{\boldsymbol{y}}_k\to q_k}\big(\underline{\boldsymbol{x}}_{k,t}, \underline{\tau}_{k,t}, \underline{r}_{k,t}\big) \\
&\quad= \sum_{r_{k,t-1}=0}^{1} \sum_{\tau_{k,t-1}=1}^{C} \int \tilde{f}_{t-1}\big(\boldsymbol{x}_{k,t-1}, \tau_{k,t-1}, r_{k,t-1}\big) \\
&\qquad\times f\big(\underline{\boldsymbol{x}}_{k,t}, \underline{\tau}_{k,t}, \underline{r}_{k,t}\big|\boldsymbol{x}_{k,t-1}, \tau_{k,t-1}, r_{k,t-1}\big)\, \mathrm{d}\boldsymbol{x}_{k,t-1}.
\end{aligned}
$$

Note that, since the belief $\tilde{f}_{t-1}(\cdot)$ is normalized, the message $\zeta_{\underline{\boldsymbol{y}}_k\to q_k}(\underline{\boldsymbol{x}}_{k,t}, \underline{\tau}_{k,t}, \underline{r}_{k,t})$ is also normalized, i.e.,

$$
\sum_{\underline{r}_{k,t}=0}^{1} \sum_{\underline{\tau}_{k,t}=1}^{C} \int \zeta_{\underline{\boldsymbol{y}}_k\to q_k}(\underline{\boldsymbol{x}}_{k,t}, \underline{\tau}_{k,t}, \underline{r}_{k,t})\mathrm{d}\underline{\boldsymbol{x}}_{k,t} = 1. \tag{24}
$$

Furthermore, according to the definitions (4)–(5), and recalling that the POs are stationary and that their class does not change over time, the message

$\zeta_{\underline{\mathbf{y}}_k \to q_k}(\underline{\mathbf{x}}_{k,t}, \underline{\tau}_{k,t}, \underline{r}_{k,t})$ for $\underline{r}_{k,t} = 1$ becomes

$$\zeta_{\underline{\mathbf{y}}_k \to q_k}\big(\underline{\mathbf{x}}_{k,t}, \underline{\tau}_{k,t}, \underline{r}_{k,t} = 1\big) = \tilde{f}_{t-1}\big(\underline{\mathbf{x}}_{k,t}, \underline{\tau}_{k,t}, \underline{r}_{k,t} = 1\big). \tag{25}$$

Finally, the inbound message from variable node "$\overline{\mathbf{y}}_m$" to factor node "h_m" is equal to one, i.e.,

$$\zeta_{\overline{\mathbf{y}}_m \to h_m}(\overline{\mathbf{y}}_{m,t}) = \zeta_{\overline{\mathbf{y}}_m \to h_m}(\overline{\mathbf{x}}_{m,t}, \overline{\tau}_{m,t}, \overline{r}_{m,t}) = 1, \tag{26}$$

since "$\overline{\mathbf{y}}_m$" is a leaf variable node of the factor graph.

C. SPA-Based Data Association

The SPA-based data association is an iterative procedure that allows to compute accurate approximations of the marginal posterior data association pmfs, i.e., $p(a_{k,t}|\mathbf{g}_{1:t}, \mathbf{z}_{1:t})$ and $p(b_{m,t}|\mathbf{g}_{1:t}, \mathbf{z}_{1:t})$ [30]. Practically, the SPA-based data association step converts the messages $\zeta_{q_k \to a_k}(a_{k,t})$ and $\zeta_{h_m \to b_m}(b_{m,t})$, into the messages $\zeta_{a_k \to q_k}(a_{k,t})$ and $\zeta_{b_m \to h_m}(b_{m,t})$. Expressions of the latter messages are provided in [26, Sec. IX-A3], whereas details of the messages $\zeta_{q_k \to a_k}(a_{k,t})$ and $\zeta_{h_m \to b_m}(b_{m,t})$ are given below.

The message from factor node "q_k" to variable node "a_k" is computed as

$$\zeta_{q_k \to a_k}(a_{k,t}) = \sum_{\underline{r}_{k,t}=0}^{1} \sum_{\underline{\tau}_{k,t}=1}^{C} \iint \zeta_{\underline{\mathbf{y}}_k \to q_k}\big(\underline{\mathbf{x}}_{k,t}, \underline{\tau}_{k,t}, \underline{r}_{k,t}\big) \times \zeta_{\mathbf{s}}(\mathbf{s}_t) q\big(\underline{\mathbf{x}}_{k,t}, \underline{\tau}_{k,t}, \underline{r}_{k,t}, a_{k,t}, \mathbf{s}_t; \mathbf{z}_t\big) \mathrm{d}\underline{\mathbf{x}}_{k,t} \mathrm{d}\mathbf{s}_t.$$

Using definitions (13)–(14) and (18)–(19), constant (23), condition (24), and message (25), we obtain for $a_{k,t} = 0$

$$\zeta_{q_k \to a_k}(a_{k,t} = 0) = \zeta_{\mathbf{s}}^0 - \iint \zeta_{\mathbf{s}}(\mathbf{s}_t) \Bigg[\sum_{\underline{\tau}_{k,t}=1}^{C} P_{\mathrm{d}}\big(\underline{\mathbf{x}}_{k,t}, \underline{\tau}_{k,t}, \mathbf{s}_t\big) \times \tilde{f}_{t-1}\big(\underline{\mathbf{x}}_{k,t}, \underline{\tau}_{k,t}, 1\big) \Bigg] \mathrm{d}\underline{\mathbf{x}}_{k,t} \mathrm{d}\mathbf{s}_t,$$

and for $a_{k,t} \in \{1, \ldots, m_t\}$

$$\zeta_{q_k \to a_k}(a_{k,t} = m) = \frac{1}{\mu_0 f_{\mathrm{FA}}(\mathbf{z}_{m,t})} \iint \zeta_{\mathbf{s}}(\mathbf{s}_t) \Bigg[\sum_{\underline{\tau}_{k,t}=1}^{C} P_{\mathrm{d}}\big(\underline{\mathbf{x}}_{k,t}, \underline{\tau}_{k,t}, \mathbf{s}_t\big) \times \tilde{f}_{t-1}\big(\underline{\mathbf{x}}_{k,t}, \underline{\tau}_{k,t}, 1\big) f\big(\mathbf{z}_{m,t} | \underline{\mathbf{x}}_{k,t}, \underline{\tau}_{k,t}, \mathbf{s}_t\big) \Bigg] \mathrm{d}\underline{\mathbf{x}}_{k,t} \mathrm{d}\mathbf{s}_t.$$

Similarly, the message from factor node "h_m" to variable node "b_m" is computed as

$$\zeta_{h_m \to b_m}(b_{m,t}) = \sum_{\overline{r}_{m,t}=0}^{1} \sum_{\overline{\tau}_{m,t}=1}^{C} \iint \zeta_{\overline{\mathbf{y}}_m \to h_m}\big(\overline{\mathbf{x}}_{m,t}, \overline{\tau}_{m,t}, \overline{r}_{m,t}\big) \times \zeta_{\mathbf{s}}(\mathbf{s}_t) h\big(\overline{\mathbf{x}}_{m,t}, \overline{\tau}_{m,t}, \overline{r}_{m,t}, b_{m,t}, \mathbf{s}_t; \mathbf{z}_{m,t}\big) \mathrm{d}\overline{\mathbf{x}}_{m,t} \mathrm{d}\mathbf{s}_t.$$

Using definitions (15)–(16) and (20), constant (23), and message (26), we obtain for $b_{m,t} = 0$

$$\zeta_{h_m \to b_m}(b_{m,t} = 0) = \zeta_{\mathbf{s}}^0 + \frac{\mu_{\mathrm{N}}}{\mu_0 f_{\mathrm{FA}}(\mathbf{z}_{m,t})} \iint \zeta_{\mathbf{s}}(\mathbf{s}_t) \Bigg[\sum_{\overline{\tau}_{m,t}=1}^{C} f_{\mathrm{N}}\big(\overline{\mathbf{x}}_{m,t}, \overline{\tau}_{m,t}\big) \times f\big(\mathbf{z}_{m,t} | \overline{\mathbf{x}}_{m,t}, \overline{\tau}_{m,t}, \mathbf{s}_t\big) \Bigg] \mathrm{d}\overline{\mathbf{x}}_{m,t} \mathrm{d}\mathbf{s}_t,$$

and $\zeta_{h_m \to b_m}(b_{m,t}) = \zeta_{\mathbf{s}}^0$ for $b_{m,t} \in \{1, \ldots, k_{t-1}\}$.

D. Outbound Messages

Once the iterations of the inner loop for data association are performed, and the messages $\zeta_{a_k \to q_k}(a_{k,t})$ and $\zeta_{b_m \to h_m}(b_{m,t})$ are available, the outbound messages are computed and eventually used to obtain the updated beliefs. The outbound message from factor node "q_k" to variable node "$\mathbf{s}$", representing the contribution of the legacy PO k to the inference of the AUV state $\mathbf{s}_t$, is computed as follows:

$$\zeta_{q_k \to \mathbf{s}}(\mathbf{s}_t) = \sum_{\underline{r}_{k,t}=0}^{1} \sum_{\underline{\tau}_{k,t}=1}^{C} \sum_{a_{k,t}=0}^{m_t} \int \zeta_{\underline{\mathbf{y}}_k \to q_k}\big(\underline{\mathbf{x}}_{k,t}, \underline{\tau}_{k,t}, \underline{r}_{k,t}\big) \times \zeta_{a_k \to q_k}(a_{k,t}) q\big(\underline{\mathbf{x}}_{k,t}, \underline{\tau}_{k,t}, \underline{r}_{k,t}, a_{k,t}, \mathbf{s}_t; \mathbf{z}_t\big) \mathrm{d}\underline{\mathbf{x}}_{k,t}.$$

As before, using definitions (13)–(14) and (18)–(19), condition (24), and message (25), the message $\zeta_{q_k \to \mathbf{s}}(\mathbf{s}_t)$ can be rewritten as

$$\zeta_{q_k \to \mathbf{s}}(\mathbf{s}_t) = \zeta_{a_k \to q_k}(a_{k,t} = 0) - \sum_{\underline{\tau}_{k,t}=1}^{C} \int P_{\mathrm{d}}\big(\underline{\mathbf{x}}_{k,t}, \underline{\tau}_{k,t}, \mathbf{s}_t\big) \times \tilde{f}_{t-1}\big(\underline{\mathbf{x}}_{k,t}, \underline{\tau}_{k,t}, 1\big) \mathfrak{f}\big(\underline{\mathbf{x}}_{k,t}, \underline{\tau}_{k,t}, \mathbf{s}_t; \mathbf{z}_t\big) \mathrm{d}\underline{\mathbf{x}}_{k,t},$$

where

$$\mathfrak{f}\big(\underline{\mathbf{x}}_{k,t}, \underline{\tau}_{k,t}, \mathbf{s}_t; \mathbf{z}_t\big) \triangleq \zeta_{a_k \to q_k}(a_{k,t} = 0) - \frac{1}{\mu_0} \sum_{m=1}^{m_t} \zeta_{a_k \to q_k}(a_{k,t} = m) \frac{f\big(\mathbf{z}_{m,t} | \underline{\mathbf{x}}_{k,t}, \underline{\tau}_{k,t}, \mathbf{s}_t\big)}{f_{\mathrm{FA}}(\mathbf{z}_{m,t})}.$$

Similarly, the outbound message from factor node "h_m" to variable node "$\mathbf{s}$", representing the contribution of the new PO m to the inference of the AUV state $\mathbf{s}_t$, is computed as follows:

$$\zeta_{h_m \to \mathbf{s}}(\mathbf{s}_t) = \sum_{\overline{r}_{m,t}=0}^{1} \sum_{\overline{\tau}_{m,t}=1}^{C} \sum_{b_{m,t}=0}^{k_{t-1}} \int \zeta_{\overline{\mathbf{y}}_m \to h_m}\big(\overline{\mathbf{x}}_{m,t}, \overline{\tau}_{m,t}, \overline{r}_{m,t}\big) \times \zeta_{b_m \to h_m}(b_{m,t}) h\big(\overline{\mathbf{x}}_{m,t}, \overline{\tau}_{m,t}, \overline{r}_{m,t}, b_{m,t}, \mathbf{s}_t; \mathbf{z}_{m,t}\big) \mathrm{d}\overline{\mathbf{x}}_{m,t}.$$

Using definitions (15)–(16) and (20), and message (26), the message $\zeta_{h_m \to \boldsymbol{s}}(\boldsymbol{s}_t)$ can be rewritten as

$$\zeta_{h_m \to \boldsymbol{s}}(\boldsymbol{s}_t) = \sum_{k=1}^{k_{t-1}} \zeta_{b_m \to h_m}(b_{m,t} = k) + \zeta_{b_m \to h_m}(b_{m,t} = 0) \Bigg[1 + \frac{\mu_{\mathrm{N}}}{\mu_0 f_{\mathrm{FA}}(\boldsymbol{z}_{m,t})} \times \sum_{\overline{\tau}_{m,t}=1}^{C} \int f_{\mathrm{N}}(\overline{\boldsymbol{x}}_{m,t}, \overline{\tau}_{m,t}) f(\boldsymbol{z}_{m,t} | \overline{\boldsymbol{x}}_{m,t}, \overline{\tau}_{m,t}, \boldsymbol{s}_t) \, \mathrm{d}\overline{\boldsymbol{x}}_{m,t} \Bigg].$$

Finally, the outbound messages from factor node "q_k" to variable node "$\underline{\boldsymbol{y}}_k$", and from factor node "h_m" to variable node "$\overline{\boldsymbol{y}}_m$", are computed as, respectively,

$$\zeta_{q_k \to \underline{\boldsymbol{y}}_k}(\underline{\boldsymbol{y}}_{k,t}) = \int \zeta_{\boldsymbol{s}}(\boldsymbol{s}_t) \prod_{\substack{k'=1 \\ k' \neq k}}^{k_{t-1}} \zeta_{q_{k'} \to \boldsymbol{s}}(\boldsymbol{s}_t) \prod_{m=1}^{m_t} \zeta_{h_m \to \boldsymbol{s}}(\boldsymbol{s}_t) \times \left[\sum_{a_{k,t}=0}^{m_t} \zeta_{a_k \to q_k}(a_{k,t}) q(\underline{\boldsymbol{y}}_{k,t}, a_{k,t}, \boldsymbol{s}_t; \boldsymbol{z}_t) \right] \mathrm{d}\boldsymbol{s}_t$$

and

$$\zeta_{h_m \to \overline{\boldsymbol{y}}_m}(\overline{\boldsymbol{y}}_{m,t}) = \int \zeta_{\boldsymbol{s}}(\boldsymbol{s}_t) \prod_{k=1}^{k_{t-1}} \zeta_{q_k \to \boldsymbol{s}}(\boldsymbol{s}_t) \prod_{\substack{m'=1 \\ m' \neq m}}^{m_t} \zeta_{h_{m'} \to \boldsymbol{s}}(\boldsymbol{s}_t) \times \left[\sum_{b_{m,t}=0}^{k_{t-1}} \zeta_{b_m \to h_m}(b_{m,t}) h(\overline{\boldsymbol{y}}_{m,t}, b_{m,t}, \boldsymbol{s}_t; \boldsymbol{z}_{m,t}) \right] \mathrm{d}\boldsymbol{s}_t.$$

E. Beliefs Computation

The final step of the proposed algorithm regards the computation of the beliefs at current time t of the legacy PO states, i.e., $\tilde{f}_t(\underline{\boldsymbol{y}}_{k,t})$, $k \in \{1, \ldots, k_{t-1}\}$, and the new PO states, i.e., $\tilde{f}_t(\overline{\boldsymbol{y}}_{m,t})$, $m \in \{1, \ldots, m_t\}$. The belief $\tilde{f}_t(\underline{\boldsymbol{y}}_{k,t})$ is computed—up to a constant factor—as the product of the messages that are passed (in opposite directions) over the edge connecting variable node "$\underline{\boldsymbol{y}}_k$" and factor node "$q_k$" [19], that is,

$$\tilde{f}_t(\underline{\boldsymbol{y}}_{k,t}) \propto \zeta_{q_k \to \underline{\boldsymbol{y}}_k}(\underline{\boldsymbol{y}}_{k,t}) \, \zeta_{\underline{\boldsymbol{y}}_k \to q_k}(\underline{\boldsymbol{y}}_{k,t}).$$

The constant factor (not reported) ensures that the belief normalizes to 1. Similarly, the belief $\tilde{f}_t(\overline{\boldsymbol{y}}_{m,t})$ is computed as the product of the messages that are passed (in opposite directions) over the edge connecting variable node "$\overline{\boldsymbol{y}}_m$" and factor node "$h_m$", that is,

$$\tilde{f}_t(\overline{\boldsymbol{y}}_{m,t}) \propto \zeta_{h_m \to \overline{\boldsymbol{y}}_m}(\overline{\boldsymbol{y}}_{m,t}) \, \zeta_{\overline{\boldsymbol{y}}_m \to h_m}(\overline{\boldsymbol{y}}_{m,t}) = \zeta_{h_m \to \overline{\boldsymbol{y}}_m}(\overline{\boldsymbol{y}}_{m,t}).$$

Eventually, the belief of the AUV state $\tilde{f}_t(\boldsymbol{s}_t)$ is also computed as it is needed for the computation of the message (22) at the next step $t+1$; this belief is calculated as the product of all the messages directed toward variable node "$\boldsymbol{s}$" [19], that is,

$$\tilde{f}_t(\boldsymbol{s}_t) \propto \zeta_{\boldsymbol{s}}(\boldsymbol{s}_t) \prod_{k=1}^{k_{t-1}} \zeta_{q_k \to \boldsymbol{s}}(\boldsymbol{s}_t) \prod_{m=1}^{m_t} \zeta_{h_m \to \boldsymbol{s}}(\boldsymbol{s}_t).$$

F. Implementation Details

The proposed SPA-based algorithm for autonomous mapping of underwater objects is implemented following a particle-based approach [31] that scales quadratically with the number of particles and the number of legacy POs, and scales linearly with the number of measurements and the number of iterations of the data association loop. As mentioned in Section II-B, in order to keep a tractable number of POs over time, a pruning step is performed. Specifically, any PO $k \in \{1, \ldots, k_t\}$ whose posterior probability of existence, i.e., $p(r_{k,t} = 1 | \boldsymbol{g}_{1:t}, \boldsymbol{z}_{1:t})$, is below a given threshold P_{th}, is removed and is not carried over to the next time step $t+1$. Moreover, to avoid PO particle impoverishment, especially due to the stationarity of the considered objects, a simple roughening strategy is employed [32].

V. SIMULATION RESULTS

A. Scenario Description

Performance of the proposed SPA-based algorithm for autonomous mapping of underwater objects is evaluated in a typical MCM mission. An AUV is programmed to survey an area of 0.25 km^2 with 5 MLOs and 45 non-MLOs by following a lawnmower pattern. The scenario is illustrated in Fig. 4, with the AUV trajectory shown as an orange solid line, and MLOs and non-MLOs as, respectively, red diamonds and green dots. The AUV state includes position and velocity in Cartesian coordinate, as well as the turn rate ν_t, i.e., $\boldsymbol{s}_t = [\boldsymbol{s}_{t,\mathrm{p}}^{\mathrm{T}}, \boldsymbol{s}_{t,\mathrm{v}}^{\mathrm{T}}, \nu_t]^{\mathrm{T}}$. The AUV kinematic model employed in the proposed algorithm and used to evaluate the pdf $f(\boldsymbol{s}_t | \boldsymbol{s}_{t-1})$ is the nearly constant turn model, that is (cf. eq. (1)),

$$\boldsymbol{s}_t = \boldsymbol{\varepsilon}(\boldsymbol{s}_{t-1}, \boldsymbol{u}_t) = \boldsymbol{F}(\boldsymbol{s}_{t-1}) + \boldsymbol{G}(\boldsymbol{s}_{t-1}) \boldsymbol{u}_t, \qquad (27)$$

where $\boldsymbol{F}(\cdot)$ and $\boldsymbol{G}(\cdot)$ are defined in [33, eqs. (6) and (7a)], and $\boldsymbol{u}_t$ is a 2D zero-mean Gaussian process noise whose covariance is $\mathrm{diag}(\sigma_{\mathrm{lin}}^2, \sigma_{\mathrm{ang}}^2)$.

The AUV moves at 1.5 m/s, produces an SAS image every $T = 33$ s, and completes the survey in approximately 2.8 h, i.e., in 305 time steps. The dimensions of the SAS tile along and across the direction of travel are, respectively, $\Delta\eta_1 = 50$ m and $\Delta\eta_2 = 110$ m, and its position with respect to the AUV is defined by $\eta_1^{(1)} = 0$ and $\eta_2^{(1)} = 20$ m (see Fig. 2). The ATR algorithm detects objects within SAS tiles with probability $p_{\mathrm{d}}(\tau_{k,t}) = p_{\mathrm{d}} = 0.9$, and distinguishes among $C = 2$ classes, i.e., MLOs ($c = 1$) and non-MLOs ($c = 2$); the vector parameters $\boldsymbol{\alpha}_1$ and $\boldsymbol{\alpha}_2$ used for simulating the probabilities

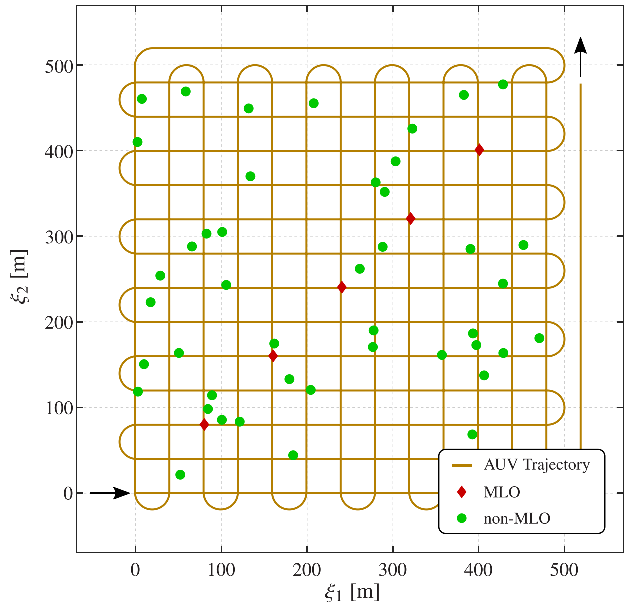


Figure 4. Illustration of the simulated scenario. The orange solid line represents the trajectory of the AUV, with the arrows indicating the beginning and the end of the trajectory. MLOs and non-MLOs are depicted as, respectively, red diamonds and green dots.

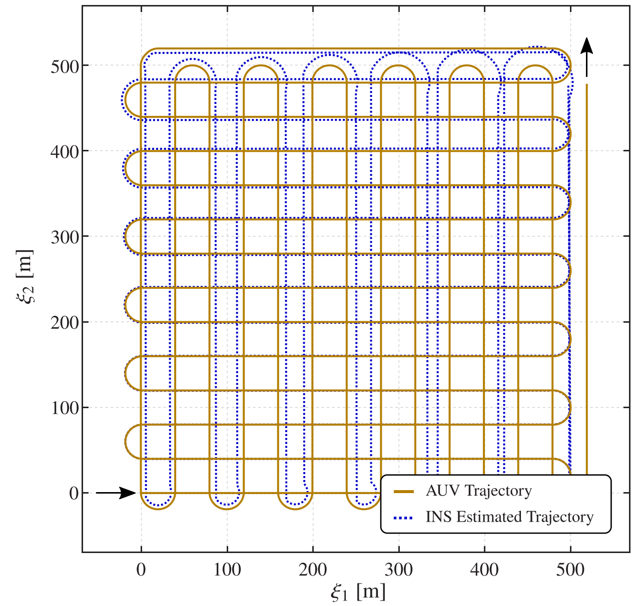


Figure 5. Illustration of the simulated scenario. The orange solid line represents the trajectory of the AUV, and the blue dotted line represents the trajectory estimated by the INS following the model in (28)–(30) with $\lambda_0 = \pi/4$ and $\epsilon_\mathrm{p} = 7.5$ m/h^2. The arrows indicate the beginning and the end of the trajectory.

$\pi_{m,t}^{(1)}$ as well as for the evaluation of the pdf $f(\pi_{m,t}^{(1)}|\tau_{k,t})$ in (7) are set to $\boldsymbol{\alpha}_1 = [6, 2]^\mathrm{T}$ and $\boldsymbol{\alpha}_2 = [2, 6]^\mathrm{T}$, respectively. The object-generated measurements are simulated according to the following model, that is,

$$\begin{aligned}\boldsymbol{\ell}_{m,t} &= \boldsymbol{\gamma}_\mathrm{O}\big(\boldsymbol{\theta}(\boldsymbol{x}_{k,t}; \boldsymbol{s}_t), \boldsymbol{\omega}_{m,t}\big)\\ &= \boldsymbol{R}(\angle \boldsymbol{s}_{t,\mathrm{v}})\big[\boldsymbol{x}_{k,t} - \boldsymbol{s}_{t,\mathrm{p}}\big] + \boldsymbol{\omega}_{m,t},\end{aligned}$$

where $\boldsymbol{\omega}_{m,t}$ is a 2D zero-mean Gaussian process noise with covariance $\sigma_\omega^2\mathbf{I}$ with $\sigma_\omega = 1$ m. The same model is used to evaluate the likelihood $f(\boldsymbol{\ell}_{m,t}|\boldsymbol{x}_{k,t}, \boldsymbol{s}_t)$. The mean number of clutter-generated measurements within both the port side and starboard side tiles is $\mu_0 = 0.1$.

The AUV's INS provides navigation data $\boldsymbol{g}_t \in \mathbb{R}^3$—that includes 2D position and heading—generated according to the following model:

$$\boldsymbol{g}_t = \big[\boldsymbol{s}_{t,\mathrm{p}}^\mathrm{T}, \angle \boldsymbol{s}_{t,\mathrm{v}}\big]^\mathrm{T} + \boldsymbol{d}_t, \tag{28}$$

where $\boldsymbol{d}_t$ is a 3D component that emulates the INS drift. For this analysis, the INS error model described in [34] is employed. Specifically, the position error has mean and variance that accumulate, respectively, quadratically and cubically over time [34, Table III]; for the heading error, instead, both mean and variance accumulate linearly over time [34, Table II]. Therefore, $\boldsymbol{d}_t$ is modeled as a 3D Gaussian process with time-varying mean

$$\frac{1}{2}\begin{bmatrix}\epsilon_\mathrm{p}\cos(\angle \boldsymbol{s}_{t,\mathrm{v}} + \lambda_0)\big(tT\big)^2\\ \epsilon_\mathrm{p}\sin(\angle \boldsymbol{s}_{t,\mathrm{v}} + \lambda_0)\big(tT\big)^2\\ (-1)^\iota 2\epsilon_\mathrm{h}\big(tT\big)\end{bmatrix}, \tag{29}$$

and time-varying covariance

$$\begin{aligned}\frac{1}{3}\mathrm{diag}\Big(&\varsigma_\mathrm{p}^2\big(tT\big)^3\cos^2(\angle \boldsymbol{s}_{t,\mathrm{v}} + \lambda_0), \ldots\\ &\varsigma_\mathrm{p}^2\big(tT\big)^3\sin^2(\angle \boldsymbol{s}_{t,\mathrm{v}} + \lambda_0), \ldots\\ &3\varsigma_\mathrm{h}^2\big(tT\big)\Big),\end{aligned} \tag{30}$$

where $\lambda_0 \in (0, 2\pi)$ and $\iota \in \{0, 1\}$. Parameters ϵ_p and ϵ_h drive, respectively, the quadratic growth of the position drift, and the linear growth of the heading drift, whereas ς_p^2 and ς_h^2 control, respectively, the cubic growth of the position error variance—for each Cartesian coordinate—and the linear growth of the heading error variance. Note that the INS drift is related to the AUV local reference system, which explains the use of $\angle \boldsymbol{s}_{t,\mathrm{v}}$ in (29)–(30); finally, ι is used to select a positive or negative heading drift, and λ_0 is used to balance the position drift over the two Cartesian coordinates. As an example, with $\lambda_0 = 0$ the AUV's position estimated by the INS is *advanced* with respect to the true AUV's position; with $\lambda_0 = \pi$, instead, the AUV's position estimated by the INS is *delayed* with respect to the true AUV's position. Figure 5 shows an example with $\lambda_0 = \pi/4$ and $\epsilon_\mathrm{p} = 7.5$ m/h^2: as time goes by, the INS provides AUV position estimates that are quadratically farther from the actual trajectory.

The INS error model defined by (29)–(30) is assumed partly unknown when running the proposed algorithm. Specifically, we consider unknown the mean (29) as well as the angle λ_0, wheres we assume known the cubic growth law of the position error variance and the linear growth law of the heading error variance. Therefore, the navigation data model used to evaluate the likelihood $f(\boldsymbol{g}_t|\boldsymbol{s}_t)$ is set to (cf. eq. (2)),

$$\boldsymbol{g}_t = \boldsymbol{\gamma}_{\mathrm{A}}(\boldsymbol{s}_t, \boldsymbol{v}_t) = \left[\boldsymbol{s}_{t,\mathrm{p}}^{\mathrm{T}}, \angle\boldsymbol{s}_{t,\mathrm{v}}\right]^{\mathrm{T}} + \boldsymbol{v}_t,$$

where $\boldsymbol{v}_t$ is a 3D zero-mean Gaussian process noise with time-varying covariance $\mathrm{diag}(\sigma_{\mathrm{p},t}^2, \sigma_{\mathrm{p},t}^2, \sigma_{\mathrm{h},t}^2)$, where

$$\sigma_{\mathrm{p},t}^2 = \frac{1}{3}\sigma_{\mathrm{p}}^2(tT)^3 \quad \text{and} \quad \sigma_{\mathrm{h},t}^2 = \sigma_{\mathrm{h}}^2(tT)$$

with $\sigma_{\mathrm{p}} = 3\,\varsigma_{\mathrm{p}}$ and $\sigma_{\mathrm{h}} = 3\,\varsigma_{\mathrm{h}}$.

Remark The INS usually provides its output much faster than $T = 33$ s. For this analysis, we assume that navigation data are available every second; this means that between any two time steps $t-1$ and t, the INS provides 32 navigation outputs. Therefore, between any two time steps, prediction and update of the AUV state are performed every second by only using the INS output; this is equivalent to run the SPA-based algorithm on a factor graph only composed by the variable nodes "$\boldsymbol{s}^-$" and "$\boldsymbol{s}$", and the factor nodes "f_{A}" and "$\boldsymbol{g}$" (see Fig. 3). Then, when both the SAS image and navigation data are available at time t, the full SPA-based algorithm described in Section IV is run. Finally, since the AUV state prediction is performed every second, the parameters σ_{lin} and σ_{ang} defining the process noise $\boldsymbol{u}_t$ in (27) and used to evaluate the pdf $f(\boldsymbol{s}_t|\boldsymbol{s}_{t-1})$ are set to $\sigma_{\mathrm{lin}} = 0.5$ m/s^2 and $\sigma_{\mathrm{ang}} = 20$ deg/s^2.

B. Results

The results obtained with the *proposed* SPA-based algorithm for autonomous mapping of underwater objects are compared with three alternative SPA-based algorithms; these alternative algorithms all assume that the AUV state at time t is known. The first one is *clairvoyant* in that it knows the "true" AUV state at time t; this is clearly used as a benchmark, since this information is not available in practice. The second alternative algorithm, referred to as *INS-plain*, considers the navigation data $\boldsymbol{g}_t$ as AUV state at current time t, with no further processing. The third alternative algorithm exploits the navigation data provided by the INS every second to sequentially infer the AUV state at time t by means of a particle filter, thus called *INS-filter*. This differs from the proposed algorithm in that it does not exploit the information readily available on the detected objects to refine the AUV state estimate, and, on the other hand, it uses only the estimated AUV position—and not its belief—to make inference about the existence and location of the objects. The results shown hereafter are averaged over 100 Monte Carlo runs; each run differs for the positions of the non-MLOs that are uniformly located in the area of interest, and for the values of $\lambda_0 \in (0, 2\pi)$ and $\iota \in \{0, 1\}$ used in (29)–(30) for the generation of the INS drifts. Moreover, 1000 particles are used to describe the SPA beliefs and messages, and the prior distributions $f_{\mathrm{n}}(\overline{\boldsymbol{x}}_{m,t})$ and $f_{\mathrm{n}}(\overline{\tau}_{m,t})$ related to the position and class of a new PO are, respectively, uniform over the SAS tile, and

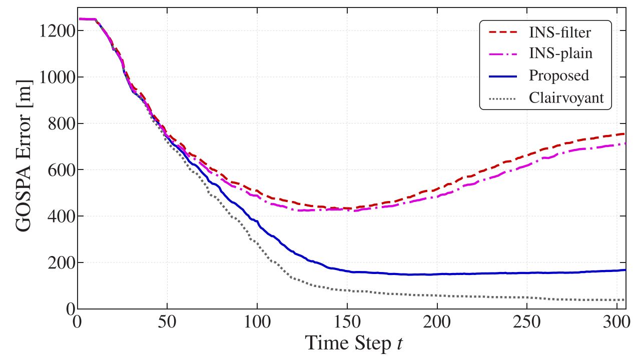


Figure 6. GOSPA errors obtained with the proposed algorithm and with three alternative algorithms, namely, clairvoyant, INS-plain, and INS-filter.

equal to $f_{\mathrm{n}}(\overline{\tau}_{m,t}) = 0.5$ for $\overline{\tau}_{m,t} \in \{1, 2\}$. The remaining parameters are set to $\epsilon_{\mathrm{p}} = 7.5$ m/h^2, $\epsilon_{\mathrm{h}} = 1$ deg/h, $\varsigma_{\mathrm{p}} = 7.5$ m/h$^{\frac{3}{2}}$, $\varsigma_{\mathrm{h}} = 0.6$ deg/h$^{\frac{1}{2}}$, $\mu_{\mathrm{N}} = 10^{-3}$, $P_{\mathrm{th}} = 10^{-4}$, and, unless otherwise stated, $E_{\mathrm{th}} = 0.8$.

The performance of the different algorithms is compared in Fig. 6 in terms of the Euclidean distance-based generalized optimal sub-pattern assignment (GOSPA) error [35], with cut-off parameter 50 m and order 1. The GOSPA metric accounts for localization errors for correctly confirmed objects, errors for missed objects, and false objects (i.e., confirmed POs not corresponding to any actual object). As expected, the clairvoyant algorithm presents the lowest GOSPA error, since it assumes a perfect knowledge of the AUV state at each time t; the error is decreasing because more objects are observed and detected as the AUV surveys the area of interest. The proposed algorithm, the INS-plain, and the INS-filter approaches have GOSPA errors very similar to those obtained with the clairvoyant algorithm at the beginning of the mission; indeed, up to time step 50, the impact of the drift is limited. As the mission proceeds, we observe that the proposed algorithm clearly outperforms the INS-plain and INS-filter approaches, demonstrating the benefit of including the inference of the AUV state within the objects detection/estimation procedure. These results are confirmed by those reported in Fig. 7, which shows the number of correctly detected objects—those for which the distance between the estimated and true position is lower than 20 m—versus the number of false objects at the end of the mission. These curves are obtained by varying the existence threshold[2] E_{th}, and demonstrate the capability of the proposed algorithm to account for the uncertain AUV state and correctly detect a higher number of objects compared to the INS-plain and INS-filter approaches. Finally, Fig. 8 shows the cardinality—i.e., the number of detected objects, including potential false objects—obtained with the proposed algorithm and the three alternative approaches

[2]For the generation of the curves in Fig. 7, 20 values of the existence threshold E_{th} are selected evenly distributed on a log scale between a minimum value of 0.02 and a maximum value of 0.99.

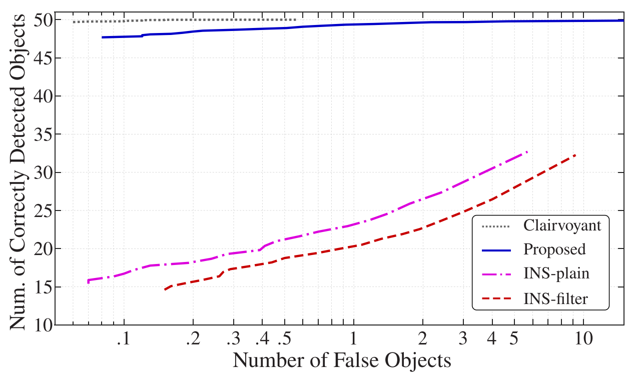


Figure 7. Number of correctly detected objects versus the number of false objects at the end of the mission obtained with the proposed algorithm and with three alternative algorithms, namely, clairvoyant, INS-plain, and INS-filter.

over time, and compares them with the true cardinality,[3] demonstrating the effectiveness of the proposed algorithm. The capability of the clairvoyant and proposed algorithms to correctly detect a higher number of objects clearly reflects into their ability to correctly classify them. Indeed, the clairvoyant and proposed algorithms reach an overall classification accuracy of 99%, compared to 47% and 45% obtained with, respectively, the INS-plain approach and the INS-filter approach. Lastly, as concerns the AUV position estimate, the proposed algorithm provides a time-averaged root-mean-square error (RMSE) of 10.9 m, while the INS-filter approach provides a time-averaged RMSE of 11.5 m. This moderate improvement of the proposed algorithm compared to the INS-filter approach is likely due to the fact that not all the objects are observable at all times; therefore, at each time t, the proposed algorithm only relies on a small set of detected objects to estimate the AUV position.

[3]The true cardinality is time-varying since more objects are observed as the AUV surveys the area of interest.

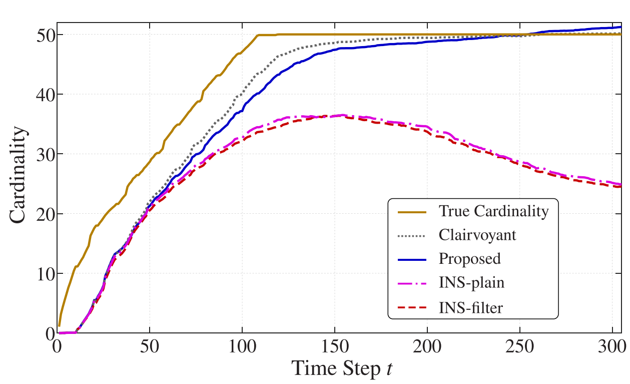


Figure 8. Cardinality over time obtained with the proposed algorithm and with three alternative algorithms, namely, clairvoyant, INS-plain, and INS-filter.

VI. CONCLUSIONS

Mapping of underwater objects is usually conducted with AUVs. A classic example is MCM operations, in which AUVs allow to operate from a distance in safe conditions. Independently of the type of application, the quality and value of the data acquired by an AUV are significantly influenced by the accuracy of its position. Because of the unavailability of GPS technologies below the sea surface, an AUV generally relies on an INS that calculates the position and heading of the vehicle by integrating values measured by accelerometers and gyroscopes available on-board. However, because of this integration, the inherent errors of these devices accumulate over time, resulting in position and orientation errors that increase over time.

This paper has proposed and described a Bayesian graph-based mapping algorithm that accounts for the inherent uncertainty of the AUV position. Exploiting the SPA, the proposed technique obtains a principled and intuitive approximation of the Bayesian inference needed for underwater object detection and estimation. The effectiveness of the proposed algorithm has been demonstrated in a simulated MCM scenario, which has shown the benefit of including the inference of the AUV position within the object detection/estimation procedure.

APPENDIX

Here, we derive the factorization in (11) of the joint posterior pdf $f(\boldsymbol{y}_{0:t}, \boldsymbol{s}_{0:t}, \boldsymbol{a}_{1:t}, \boldsymbol{b}_{1:t} | \boldsymbol{g}_{1:t}, \boldsymbol{z}_{1:t})$. Since the measurements $\boldsymbol{z}_{1:t}$ are observed, hence known, the joint vector of numbers of measurements $\boldsymbol{m}_{1:t}$ is also known, that is, $f(\boldsymbol{y}_{0:t}, \boldsymbol{s}_{0:t}, \boldsymbol{a}_{1:t}, \boldsymbol{b}_{1:t} | \boldsymbol{g}_{1:t}, \boldsymbol{z}_{1:t}) = f(\boldsymbol{y}_{0:t}, \boldsymbol{s}_{0:t}, \boldsymbol{a}_{1:t}, \boldsymbol{b}_{1:t} | \boldsymbol{g}_{1:t}, \boldsymbol{z}_{1:t}, \boldsymbol{m}_{1:t})$. Then, assuming that all the variables—joint PO state, AUV state, data association variables, navigation data, and measurements—at time t are conditionally independent of the past variables given the joint PO state and AUV state at previous time $t-1$, we obtain the factorization in (32). Recalling from Section II-B that $\boldsymbol{y}_t$ is the vector stacking the k_{t-1} legacy PO states and the m_t new PO states at time t, that is, $\boldsymbol{y}_t = [\underline{\boldsymbol{y}}_t^{\mathrm{T}}, \overline{\boldsymbol{y}}_t^{\mathrm{T}}]^{\mathrm{T}}$, each factor $f(\boldsymbol{y}_t, \boldsymbol{s}_t, \boldsymbol{a}_t, \boldsymbol{b}_t, \boldsymbol{g}_t, \boldsymbol{z}_t, m_t | \boldsymbol{y}_{t-1}, \boldsymbol{s}_{t-1})$ of the product in (32) can be further expressed as

$$\begin{aligned}
&f(\boldsymbol{y}_t, \boldsymbol{s}_t, \boldsymbol{a}_t, \boldsymbol{b}_t, \boldsymbol{g}_t, \boldsymbol{z}_t, m_t | \boldsymbol{y}_{t-1}, \boldsymbol{s}_{t-1}) \\
&\quad = f(\overline{\boldsymbol{y}}_t, \boldsymbol{a}_t, \boldsymbol{b}_t, \boldsymbol{g}_t, \boldsymbol{z}_t, m_t | \underline{\boldsymbol{y}}_t, \boldsymbol{s}_t, \boldsymbol{y}_{t-1}, \boldsymbol{s}_{t-1}) \\
&\qquad\quad \times f(\underline{\boldsymbol{y}}_t, \boldsymbol{s}_t | \boldsymbol{y}_{t-1}, \boldsymbol{s}_{t-1}) \\
&\quad = f(\overline{\boldsymbol{y}}_t, \boldsymbol{a}_t, \boldsymbol{b}_t, \boldsymbol{g}_t, \boldsymbol{z}_t, m_t | \underline{\boldsymbol{y}}_t, \boldsymbol{s}_t) f(\underline{\boldsymbol{y}}_t | \boldsymbol{y}_{t-1}) f(\boldsymbol{s}_t | \boldsymbol{s}_{t-1}) \\
&\quad = f(\overline{\boldsymbol{y}}_t, \boldsymbol{a}_t, \boldsymbol{b}_t, \boldsymbol{z}_t, m_t | \underline{\boldsymbol{y}}_t, \boldsymbol{s}_t) f(\boldsymbol{g}_t | \boldsymbol{s}_t) \\
&\qquad\quad \times f(\underline{\boldsymbol{y}}_t | \boldsymbol{y}_{t-1}) f(\boldsymbol{s}_t | \boldsymbol{s}_{t-1}),
\end{aligned} \tag{31}$$

where we assumed that PO states and AUV states evolve independently, and that navigation data and measurements are conditionally independent given the AUV

$$
\begin{aligned}
&f\left(\boldsymbol{y}_{0:t}, \boldsymbol{s}_{0:t}, \boldsymbol{a}_{1:t}, \boldsymbol{b}_{1:t} \middle| \boldsymbol{g}_{1:t}, \boldsymbol{z}_{1:t}, \boldsymbol{m}_{1:t}\right) \\
&\quad \propto f\left(\boldsymbol{y}_{0:t}, \boldsymbol{s}_{0:t}, \boldsymbol{a}_{1:t}, \boldsymbol{b}_{1:t}, \boldsymbol{g}_{1:t}, \boldsymbol{z}_{1:t}, \boldsymbol{m}_{1:t}\right) \\
&\quad = f\left(\boldsymbol{y}_t, \boldsymbol{s}_t, \boldsymbol{a}_t, \boldsymbol{b}_t, \boldsymbol{g}_t, \boldsymbol{z}_t, m_t \middle| \boldsymbol{y}_{0:t-1}, \boldsymbol{s}_{0:t-1}, \boldsymbol{a}_{1:t-1}, \boldsymbol{b}_{1:t-1}, \boldsymbol{g}_{1:t-1}, \boldsymbol{z}_{1:t-1}, \boldsymbol{m}_{1:t-1}\right) \\
&\qquad \times f\left(\boldsymbol{y}_{0:t-1}, \boldsymbol{s}_{0:t-1}, \boldsymbol{a}_{1:t-1}, \boldsymbol{b}_{1:t-1}, \boldsymbol{g}_{1:t-1}, \boldsymbol{z}_{1:t-1}, \boldsymbol{m}_{1:t-1}\right) \\
&\quad = f\left(\boldsymbol{y}_t, \boldsymbol{s}_t, \boldsymbol{a}_t, \boldsymbol{b}_t, \boldsymbol{g}_t, \boldsymbol{z}_t, \boldsymbol{m}_t \middle| \boldsymbol{y}_{t-1}, \boldsymbol{s}_{t-1}\right) \times f\left(\boldsymbol{y}_{0:t-1}, \boldsymbol{s}_{0:t-1}, \boldsymbol{a}_{1:t-1}, \boldsymbol{b}_{1:t-1}, \boldsymbol{g}_{1:t-1}, \boldsymbol{z}_{1:t-1}, \boldsymbol{m}_{1:t-1}\right) \\
&\quad = f(\boldsymbol{s}_0) f(\boldsymbol{y}_0) \prod_{t'=1}^{t} f\left(\boldsymbol{y}_{t'}, \boldsymbol{s}_{t'}, \boldsymbol{a}_{t'}, \boldsymbol{b}_{t'}, \boldsymbol{g}_{t'}, \boldsymbol{z}_{t'}, m_{t'} \middle| \boldsymbol{y}_{t'-1}, \boldsymbol{s}_{t'-1}\right)
\end{aligned} \tag{32}
$$

state $\boldsymbol{s}_t$. Then, observing that the description of the data association given by $\boldsymbol{a}_t$ and $\boldsymbol{b}_t$ is redundant once m_t is observed—indeed, $\boldsymbol{a}_t$ can be derived from $\boldsymbol{b}_t$, and vice versa, when m_t is known [26] —, each factor $f(\overline{\boldsymbol{y}}_t, \boldsymbol{a}_t, \boldsymbol{b}_t, z_t, m_t | \underline{\boldsymbol{y}}_t, \boldsymbol{s}_t)$ can be further expressed as

$$
\begin{aligned}
&f(\overline{\boldsymbol{y}}_t, \boldsymbol{a}_t, \boldsymbol{b}_t, z_t, m_t | \underline{\boldsymbol{y}}_t, \boldsymbol{s}_t) \\
&\quad = f(z_t | \boldsymbol{y}_t, \boldsymbol{a}_t, \boldsymbol{b}_t, m_t, \boldsymbol{s}_t) f(\overline{\boldsymbol{y}}_t, \boldsymbol{a}_t, \boldsymbol{b}_t, m_t | \underline{\boldsymbol{y}}_t, \boldsymbol{s}_t) \\
&\quad = f(z_t | \boldsymbol{y}_t, \boldsymbol{a}_t, m_t, \boldsymbol{s}_t) f(\overline{\boldsymbol{y}}_t, \boldsymbol{a}_t, \boldsymbol{b}_t, m_t | \underline{\boldsymbol{y}}_t, \boldsymbol{s}_t).
\end{aligned} \tag{33}
$$

Eventually, by inserting (33) into (31) and (31) into (32), we obtain the factorization in (11).

REFERENCES

[1] S. Mukhopadhyay, C. Wang, M. Patterson, M. Malisoff, and F. Zhang
"Collaborative autonomous surveys in marine environments affected by oil spills," *Cooperative Robots and Sensor Networks*, Anis Koubâa and Abdelmajid Khelil, Eds. Heidelberg, Germany: Springer, 2014, pp. 87–113.

[2] M. Jacobi
"Autonomous inspection of underwater structures," in *Robot. Auton. Syst.*, vol. 67, pp. 80–86, May 2015.

[3] M. Ho, S. El-Borgi, D. Patil, and G. Song
"Inspection and monitoring systems subsea pipelines: A review paper,"
Struct. Health Monit., vol. 19, no. 2, pp. 606–645, Mar. 2020.

[4] N. Gracias et al.
"Mapping the Moon: Using a lightweight AUV to survey the site of the 17th century ship 'La Lune' in
Proc. IEEE OCEANS, 2013, pp. 1–8.

[5] S. Sariel, T. Balch, and N. Erdogan
"Naval mine countermeasure missions,"
IEEE Robot. Autom. Mag., vol. 15, no. 1, pp. 45–52, Mar. 2008.

[6] M. R. Dhanak and N. I. Xiros
Handbook of Ocean Engineering. Heidelberg, Germany: Springer, 2016.

[7] J. Melo and A. Matos
"Survey on advances on terrain based navigation for autonomous underwater vehicles,"
Ocean Eng., vol. 139, pp. 250–264, Jul. 2017.

[8] Ø. Hegrenæs and O. Hallingstad
"Model-aided INS with sea current estimation for robust underwater navigation,"
IEEE J. Ocean. Eng., vol. 36, no. 2, pp. 316–337, Apr. 2011.

[9] J. Bao, D. Li, X. Qiao, and T. Rauschenbach
"Integrated navigation for autonomous underwater vehicles in aquaculture: A review,"
Inf. Process. Agric., vol. 7, no. 1, pp. 139–151, Mar. 2020.

[10] P. Liu, B. Wang, Z. Deng, and M. Fu
"INS/DVL/PS tightly coupled underwater navigation method with limited DVL measurements,"
IEEE Sens. J., vol. 18, no. 7, pp. 2994–3002, Apr. 2018.

[11] D. P. William
"Fast target detection in synthetic aperture sonar imagery: A new algorithm and large-scale performance analysis,"
IEEE J. Ocean. Eng., vol. 40, no. 1, pp. 71–92, Jan. 2015.

[12] S. Hożyń
"A review of underwater mine detection and classification in sonar imagery," *Electronics*, vol. 10, no. 23, Jan. 2021.

[13] N. Palomeras, T. Furfaro, D. P. Williams, M. Carreras, and S. Dugelay
"Automatic target recognition for mine countermeasure missions using forward-looking sonar data,"
IEEE J. Ocean. Eng., vol. 47, no. 1, pp. 141–161, Jan. 2022.

[14] A. Mallios, P. Ridao, D. Ribas, and E. Hernández
"Scan matching SLAM in underwater environments,"
Auton. Robot., vol. 36, no. 3, pp. 181–198, Mar. 2014.

[15] N. Fairfield, G. Kantor, D. Jonak, and D. Wettergreen
"Autonomous exploration and mapping of flooded sinkholes,"
Int. J. Robot. Res., vol. 29, no. 6, pp. 748–774, Sep. 2009.

[16] T. Fabbri, F. Di Corato, D. Fenucci, D. Meucci, and A. Caiti
"Multiple target tracking in seabed surveys using the GM-PHD filter,"
in *Proc. IEEE OCEANS*, 2015, pp. 1–6.

[17] J. Melo and S. Dugelay
"AUV mapping of underwater targets,"
in *Proc. IEEE OCEANS*, 2019, pp. 1–6.

[18] M. Fröhle, C. Lindberg, K. Granström, and H. Wymeersch
"Multisensor Poisson multi-Bernoulli filter for joint target-sensor state tracking,"
IEEE Trans. Intell. Veh., vol. 4, no. 4, pp. 609–621, Dec. 2019.

[19] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger
"Factor graphs and the sum-product algorithm,"
IEEE Trans. Inf. Theory, vol. 47, no. 2, pp. 498–519, Feb. 2001.

[20] H.-A. Loeliger, J. Dauwels, J. Hu, S. Korl, L. Ping, and F. R. Kschischang
"The factor graph approach to model-based signal processing,"
Proc. IEEE, vol. 95, no. 6, pp. 1295–1322, Jun. 2007.

[21] J. Melo, D. Gaglione, G. Soldi, P. Braca, and S. Dugelay
"A belief propagation based algorithm for autonomous mapping of underwater targets,"
in *Proc. IEEE OCEANS*, 2020, pp. 1–7.

[22] E. Leitinger, F. Meyer, F. Hlawatsch, K. Witrisal, F. Tufvesson, and M. Z. Win
"A belief propagation algorithm for multipath-based SLAM,"
IEEE Trans. Wireless Commun., vol. 18, no. 12, pp. 5613–5629, Dec. 2019.

[23] M. Brambilla et al.
"Cooperative localization and multitarget tracking in agent networks with the sum-product algorithm,"
IEEE Open J. Signal Process., vol. 3, pp. 169–195, Mar. 2022.

[24] G. Soldi et al.
"Underwater tracking based on the sum-product algorithm enhanced by a neural network detections classifier," *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2020, pp. 5460–5464.

[25] D. Gaglione, G. Soldi, P. Braca, G. De Magistris, F. Meyer, and F. Hlawatsch
"Classification-aided multitarget tracking using the sum-product algorithm,"
in *IEEE Signal Process. Lett.*, vol. 27, pp. 1710–1714, Sep. 2020.

[26] F. Meyer, T. Kropfreiter, J. L. Williams, R. Lau, F. Hlawatsch, P. Braca, and M. Z. Win
"Message passing algorithms for scalable multitarget tracking,"
Proc. IEEE, vol. 106, no. 2, pp. 221–259, Feb. 2018.

[27] O. Daniell, Y. Petillot, S. Reed, J. Vazquez, and A. Frau
"Reducing false alarms in automated target recognition using local sea-floor characteristics,"
in *Proc. Sensor Signal Process. Defence*, 2014, pp. 1–5.

[28] Y. Bar-Shalom, P. K. Willett, and X. Tian
Tracking and Data Fusion: A Handbook of Algorithms.
Storrs, CT, USA: YBS Publishing, 2011.

[29] H. V. Poor
An Introduction to Signal Detection and Estimation, 2nd ed.
New York, NY, USA: Springer, 1994.

[30] J. L. Williams and R. Lau
"Approximate evaluation of marginal association probabilities with belief propagation,"
IEEE Trans. Aerosp. Electron. Syst., vol. 50, no. 4, pp. 2942–2959, Oct. 2014.

[31] F. Meyer, P. Braca, P. Willett, and F. Hlawatsch
"A scalable algorithm for tracking an unknown number of targets using multiple sensors,"
IEEE Trans. Signal Process., vol. 65, no. 13, pp. 3478–3493, Jul. 2017.

[32] T. Li, M. Bolic, and P. M. Djuric
"Resampling methods for particle filtering: Classification, implementation, and strategies,"
IEEE Signal Process. Mag., vol. 32, no. 3, pp. 70–86, May 2015.

[33] M. Roth, G. Hendeby, and F. Gustafsson
"EKF/UKF maneuvering target tracking using coordinated turn models with polar/Cartesian velocity,"
in *Proc. 17th Int. Conf. Inf. Fusion*, 2014, pp. 1–8.

[34] O. J. Woodman
"An introduction to inertial navigation," University of Cambridge, Computer Laboratory, Cambridge, UK, Tech. Rep. UCAM-CL-TR-696, Aug. 2007.

[35] A. S. Rahmathullah, Á. F. García-Fernández, and L. Svensson
"Generalized optimal sub-pattern assignment metric,"
in *Proc. 20th Int. Conf. Inf. Fusion*, 2017, pp. 1–8.

Domenico Gaglione (Member, IEEE) received the B.Sc. and M.Sc. degrees (summa cum laude) in telecommunications engineering from Università degli Studi di Napoli Federico II, Naples, Italy, in 2011 and 2013, respectively, and the Ph.D. degree from the Department of Electronic and Electrical Engineering, University of Strathclyde, Glasgow, U.K., in 2017. He is currently a Research Scientist with NATO STO Centre for Maritime Research and Experimentation (CMRE), La Spezia, Italy. His research interests include statistical signal processing with emphasis on state estimation, data fusion, and multisensor multitarget tracking. He was the recipient of the Best Student Paper Award at the 2015 IEEE International Radar Conference (RadarCon), Arlington, VA, USA, and of the NATO STO Scientific Achievement Award in 2020, as a Member of the Data Knowledge and Operational Effectiveness (DKOE) Research Group at CMRE for Advances in Artificial Intelligence and Information Fusion for Maritime Situational Awareness.

Giovanni Soldi received the master's degree in applied mathematics from the University of Milan, Milan, Italy, in 2011, and the Ph.D. degree in signal processing from Télécom ParisTech, Paris, France, in 2016. Since 2016, he has been a Scientist with the Centre for Maritime Research and Experimentation (CMRE), La Spezia, Italy. His research interests include statistical signal processing with a focus on state estimation, data fusion, and multisensor-multitarget tracking techniques. He was the recipient of the NATO STO Scientific Achievement Award, in 2020, as a Member of the Data Knowledge and Operational Effectiveness (DKOE) Research Group, CMRE for Advances in Artificial Intelligence and Information Fusion for Maritime Situational Awareness.

Paolo Braca (Senior Member, IEEE) received the Laurea degree (summa cum laude) in electronic engineering and the Ph.D. degree (Hons.) in information engineering from the University of Salerno, Fisciano, Italy, in 2006 and 2010, respectively. In 2009, he was a Visiting Scholar with the Department of Electronics and Communication Engineering, University of Connecticut, Storrs, CT, USA. From 2010 to 2011, he was a Postdoctoral Associate with the University of Salerno. In 2011, he joined the NATO STO CMRE, where he is currently a Senior Scientist and a Project Manager. He leaded a number of research projects funded by the European Commission, by the U.S. Office of Naval Research, and other institutions. He conducts research in the general area of statistical signal processing with emphasis on detection and estimation theory, wireless sensor network, multi-agent algorithms, target tracking and data fusion, adaptation and learning over graphs, radar (sonar) signal processing, and machine learning. He has coauthored more than 200 publications in international scientific journals, conference proceedings, and NATO technical reports. He was awarded with the National Scientific Qualification to function as an Associate Professor and a Full Professor with Italian Universities, in 2017 and 2018, respectively. He is with the technical committee of the major international conferences in the field of signal processing and data fusion. He was the recipient of the Best Student Paper Award (first runner-up) at FUSION conference in 2009, and NATO STO Scientific Achievement Award (SAA) 2017 for its contribution to the "Development and Demonstration of Networked Autonomous ASW". His coauthored paper was the recipient of the Best Paper Award (first runner-up) at the SSPD conference in 2019. He was also the recipient of the NATO STO SAA, in 2020, as a Team Leader for the Advances in Artificial Intelligence and Information Fusion for Maritime Situational Awareness. His coauthored article was the recipient of the Young Scientist Contest Award at the Signal Processing Symposium 2021. He is an Associate Editor for the T-AES, ISIF-JAIF, and IET-RSN. In 2017, he was a Lead Guest Editor of the Special Issue "Sonar Multi-Sensor Applications and Techniques" in IET-RSN. He was an Associate Editor for the SP-M, T-SP, and EURASIP-JASP.

Comments on "Variations of Joint Integrated Data Association with Radar and Target-Provided Measurements"

DOMENIC GAGLIONE
PAOLO BRACA
GIOVANNI SOLDI
FLORIAN MEYER
AUDUN G. HEM
EDMUND F. BREKKE
FRANZ HLAWATSCH

Recently, a method for including target-provided measurements within a joint integrated probabilistic data association (JIPDA) filter was presented and compared with a belief propagation (BP)-based multitarget tracking method. While the JIPDA-based approach uses multiple kinematic models within an interacting multiple model framework, the BP-based approach uses only a single kinematic model. Here, we present and analyze the results of similar experiments conducted on both simulated and real data. Our results show that the JIPDA-based method tends to outperform the BP-based method when the targets are well-separated and perform sharp maneuvers, whereas the BP-based method outperforms the JIPDA-based method when the targets are closely spaced.

Manuscript received December 4, 2023; revised February 29, 2024; released for publication April 4, 2024

Refereeing of this contribution was handled by Jason L. Williams.

This work was supported in part by the NATO Allied Command Transformation (ACT) under the DKOE project, in part by the Austrian Science Fund (FWF) under grant P 32055-N31, in part by the Research Council of Norway through Project 295033 Autosit, in part by DNV, in part by Kongsberg, and in part by Maritime Robotics.

D. Gaglione, P. Braca, and G. Soldi are with the Research Department, NATO Centre for Maritime Research and Experimentation (CMRE), 19126 La Spezia, Italy (e-mail: domenico.gaglione@cmre.nato.int; paolo.braca@cmre.nato.int; giovanni.soldi@cmre.nato.int).

F. Meyer is with the Department of Electrical and Computer Engineering and the Scripps Institution of Oceanography, University of California San Diego, La Jolla, CA 92093, USA (e-mail: flmeyer@ucsd.edu).

A. G. Hem and E. F. Brekke are with the Department of Engineering Cybernetics, Norwegian University of Science and Technology, 7034 Trondheim, Norway (e-mail: audun.g.hem@ntnu.no; edmund.brekke@ntnu.no).

F. Hlawatsch is with the Institute of Telecommunications, TU Wien, 1040 Vienna, Austria (e-mail: franz.hlawatsch@tuwien.ac.at).

1557-6418/2024/$17.00 © 2023 JAIF

I. INTRODUCTION

A. Background

In a recent publication [1], three methods for including target-provided measurements in a joint integrated probabilistic data association (JIPDA) framework were proposed. The framework considered in [1], referred to as VIMMJIPDA filter, combines interacting multiple models (IMM) and a visibility state within the well-established JIPDA filter [2]. The IMM concept, first introduced in [3], allows the use of multiple kinematic models for the tracking of maneuvering targets, while the visibility state indicates whether the tracked target is visible to the sensor or not. A target-provided measurement is an observation produced by a target and made available to the tracking method. This observation usually includes kinematic information, e.g., the target's position and velocity, and additional information such as a unique code identifying the target. The target obtains its own position and velocity through an onboard device, generally a global navigation satellite system (GNSS) transponder, and transmits this information, as well as any other relevant information, to neighboring targets and to a central fusion node. Examples of such systems are the automatic identification system (AIS) for maritime surveillance and vessel collision avoidance [4] and the automatic dependent surveillance broadcast (ADS-B) system for air traffic control [5].

These target-dependent reporting systems differ from classical perception sensors such as radar, lidar, and cameras in several aspects. Firstly, the measurements they produce are asynchronous because they are provided by the targets themselves, and each target can transmit its messages at any time. Secondly, a target-provided measurement cannot be a false alarm because it is not the result of a detection process.[1] Several attempts have been made to fuse target-provided measurements and observations produced by perception sensors. One common approach is to consider the reporting system and the perception sensor as stand-alone assets, and accordingly estimate two separate sets of tracks, which are later fused to form a single set of estimated tracks. This approach, which is known as track-level fusion, has some performance limitations compared to measurement-level fusion techniques [6].

The methods proposed in [1] follow a measurement-level fusion approach and are based on the VIMMJIPDA tracking method. Specifically, three different methods for handling the target-provided measurements are proposed. One of them processes the

[1]Nevertheless, target-dependent reporting systems like the AIS can be subject to intentional reporting of false information. However, this is not taken into account in [1] nor in the remainder of this paper.

measurements as they arrive, i.e., sequentially; the others collect the measurements and process them at fixed times. The validity of these approaches is demonstrated both in a simulated maritime scenario and with real data acquired as part of the Autosea project conducted by the Norwegian University of Science and Technology [7], and the performance of the proposed methods is compared with that of the belief propagation (BP)-based tracking method with target-provided measurement fusion capabilities presented in [8], [9]. The setup of both the simulated scenario and the real experiment consists of a single radar sensor and the AIS. It is observed that the particle filter (PF) implementation of the BP-based tracking method (referred to as the BP-PF+AIS method) performs worse than the VIMMJIPDA-based methods and, in some cases, even worse than a radar-only method, i.e., a method that uses only the radar measurements.

B. Contribution

The implementation of the BP-PF+AIS method is not publicly available, which led the authors of [1] to use their own implementation. In this paper, we study the performance of the original implementation of the BP-PF+AIS method used in [8], [9] for a simulated scenario similar to the one described in [1, Sec. VIII-A], as well as on the real dataset provided by the Autosea project [7]. Additionally, we consider the simulated scenario described in [9, Sec. VI-A]. The performance obtained with the original implementation of the BP-PF+AIS method is compared with that obtained with the original VIMMJIPDA method using only the radar measurements [2] and with the sequential method proposed in [1] (to be referred to as VIMMJIPDA+AIS), for which code is available in [10]. We note that the BP-PF+AIS method described in [8], [9] does not use multiple kinematic models. However, a BP-based tracking method using multiple kinematic models that conforms to the general IMM approach is presented in [11]. Therefore, we also evaluate and compare the performance of the BP-PF+AIS method described in [8], [9], properly extended to exploit multiple kinematic models as proposed in [11]; we refer to this version as BP-PF+AIS+IMM method. We will demonstrate that while the BP-PF+AIS and BP-PF+AIS+IMM methods have performance advantages in the case of closely spaced targets, the VIMMJIPDA+AIS method performs better when the targets are well-separated and when they perform sharp maneuvers.

The remainder of this paper is organized as follows: Section II provides a brief description of the VIMMJIPDA, VIMMJIPDA+AIS, BP-PF+AIS, and BP-PF+AIS+IMM methods. Section III presents the results of an experimental comparison of these methods conducted on two simulated scenarios, while in Section IV the performance is compared on a real dataset. Concluding remarks are provided in Section V.

II. BRIEF DESCRIPTION OF THE COMPARED METHODS

The VIMMJIPDA method, derived in [2] as a special case of the Poisson multi-Bernoulli filter, is a variation of the JIPDA filter for multitarget tracking that includes multiple kinematic models and a visibility state, and uses hypothesis enumeration to model the target-measurement data association. Specifically, a single-linkage clustering strategy is used to group targets that share measurements. Then, for groups with less than four targets or less than two measurements, brute-force hypothesis enumeration is performed, whereas Murty's algorithm [12] with a maximum of eight hypotheses is used for all other groups. The VIMMJIPDA+AIS method proposed in [1] builds upon [2] and incorporates target-provided measurements. One important technical detail that enables this is to model target birth as a marked Poisson point process, where the marks are constituted by the unique codes identifying the targets.

The BP-based multitarget tracking methods are described in [13] and references therein. The principle behind these methods is to exploit the statistical independence of certain random variables describing the tracking problem, and to represent these independence relationships by means of a factor graph. Then, using a message-passing algorithm—i.e., the sum-product algorithm—on this factor graph enables an intuitive and computationally efficient approximation of the Bayesian inference needed for target detection and estimation. Fundamental for the derivation of these methods is to properly model and formulate the target-measurement data association. An iterative BP-based algorithm for data association with remarkable performance in terms of convergence and accuracy was proposed in [14]. A common approach to implementing BP-based tracking algorithms for general nonlinear/non-Gaussian kinematic and measurement models is to resort to a PF, as described in [15].

Building upon [13]–[15], a suite of BP-PF methods has been developed recently. The BP-PF+AIS method proposed in [9] extends the previous works to incorporate heterogeneous data. This method fuses sensor measurements and target-provided measurements, e.g., AIS data, by establishing an appropriate likelihood for target-provided measurements and a statistical model for data association. A self-tuning BP-PF method that continuously adapts to time-varying system models is proposed in [11]. This method infers and adapts to an unknown detection probability of the sensors and employs multiple kinematic models in line with the IMM framework. Similar to a construction kit system, BP-based algorithm parts can be combined in a modular manner to achieve desired functionalities and properties. For example, the BP-PF+AIS+IMM method, which is used for comparison in this paper, combines the IMM framework proposed in [11] with the ability

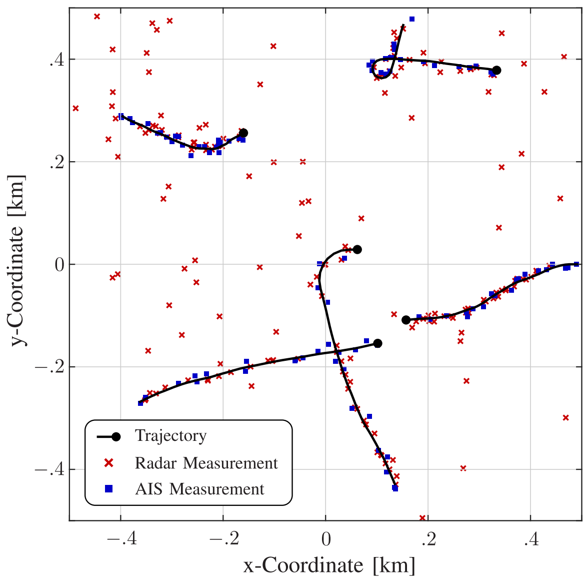


Figure 1. A realization of the simulated scenario considered in [1, Sec. VIII-A], with $P_\mathrm{D} = 0.5$. The black dots indicate the final positions of the trajectories.

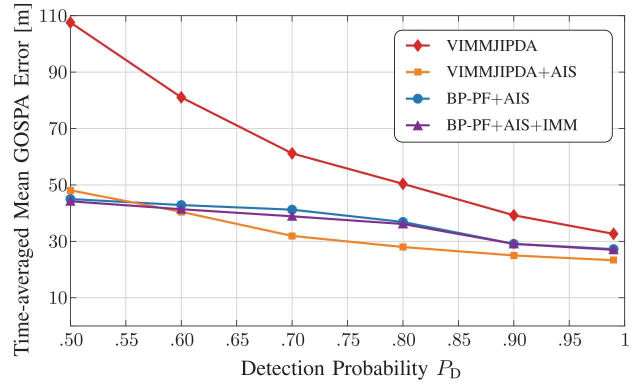


Figure 2. Time-averaged mean GOSPA error versus detection probability P_D of the radar sensor for the simulated scenario considered in [1, Sec. VIII-A].

to fuse sensor measurements and target-provided measurements as established in [9].

III. SIMULATION RESULTS

In this section, we present simulation results for the scenarios considered in [1, Sec. VIII-A] and [9, Sec. VI-A].

A. Scenario Considered in [1]

The simulated scenario considered in [1, Sec. VIII-A] employs a single radar sensor located at $[0, 0]^\mathrm{T}$ that surveys a circular area of radius 500 m with a time scan duration of 2.5 s. Five targets appear at the edge of that area, three at time $t = 0$ s and two at time $t = 10$ s, initially moving with a velocity of 3.75 m/s. The trajectories of the targets are generated according to a nearly constant velocity (NCV) kinematic model [16, Sec. 6.2.2] with driving noise variance set to 0.1^2 m^2 s^{-3}, and with occasional maneuvers according to a coordinated turn (CT) kinematic model [16, Sec. 4.2.2]. The radar detects a target with probability P_D and generates range-bearing measurements; the measurement noise is a two-dimensional (2D) zero-mean Gaussian random vector with covariance $\mathrm{diag}(8^2\,\mathrm{m}^2, 1^2\,\mathrm{deg}^2)$. The number of false alarms is Poisson distributed with mean 2. All targets provide AIS measurements containing their unique identifying code as well as their 2D Cartesian position and velocity. The number of AIS measurements provided by a target during each time scan is Poisson distributed with mean 0.5. The AIS measurement noise for position and velocity is a 4D zero-mean Gaussian random vector with covariance $\mathrm{diag}(3^2\,\mathrm{m}^2, 3^2\,\mathrm{m}^2, 0.1^2\,\mathrm{m}^2/\mathrm{s}^2, 0.1^2\,\mathrm{m}^2/\mathrm{s}^2)$. Figure 1 shows a realization of the scenario with the trajectories of the five targets, the 2D position component of the AIS measurements, and the radar measurements generated with $P_\mathrm{D} = 0.5$.

In Figures 2 and 3, we demonstrate and compare the performance of the radar-only method (i.e., VIMMJIPDA [2]), the sequential method proposed in [1] (i.e., VIMMJIPDA+AIS), the original implementation of the BP-PF+AIS method [8], [9], and the BP-PF+AIS+IMM method. The performance of these methods is measured by the mean generalized optimal sub-pattern assignment (GOSPA) error [17] of order 2 with a cutoff parameter 200 m, averaged over 100 simulation runs. The mean GOSPA error accounts for localization errors for correctly confirmed targets as well as for errors due to missed and false targets. For the VIMMJIPDA and VIMMJIPDA+AIS methods, we use the parameters reported in [1, Tab. III]. Where applicable, the same parameters are also used for the BP-PF+AIS and BP-PF+AIS+IMM methods (e.g., the survival probability), while parameters specifically related to the BP-based methods (e.g., the number of potential targets) are set as in [9]. The VIMMJIPDA and VIMMJIPDA+AIS methods use three models to characterize the kinematics of the targets, namely, two NCV models with different driving noise variances and one CT model. The BP-PF+AIS method uses a single NCV model; therefore, to account for potential maneuvers, the driving noise variance of the NCV model for the BP-PF+AIS method is set to 0.8^2 m^2 s^{-3}. Finally, the BP-PF+AIS+IMM method uses two NCV models with driving noise variance 0.05^2 m^2 s^{-3} and 0.8^2 m^2 s^{-3}. Differently from the NCV model, the CT model does not allow a simple closed-form calculation of the likelihood for the target-provided measurements specified in the supplementary material of [9]. Developing a tractable implementation of this likelihood is outside the scope of this paper, and for this reason, the BP-PF+AIS+IMM method does not employ a CT model.

Figure 2 shows the time-averaged mean GOSPA error when the detection probability P_D of the radar sensor is varied from 0.50 to 0.99. It can be seen that the VIMMJIPDA+AIS method performs better than

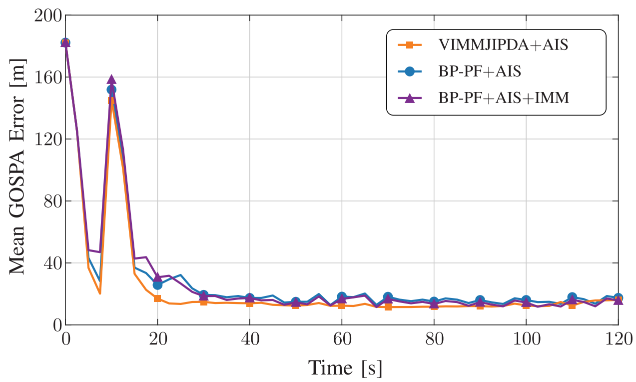


Figure 3. Mean GOSPA error versus time for the simulated scenario considered in [1, Sec. VIII-A] with $P_{\mathrm{D}} = 0.9$.

both the BP-PF+AIS and BP-PF+AIS+IMM methods. Furthermore, the use of multiple NCV models within the BP-PF+AIS method offers only a marginal improvement. The difference in performance between the VIMMJIPDA+AIS method and the BP-PF+AIS and BP-PF+AIS+IMM methods can be explained by the fact that the VIMMJIPDA+AIS method uses also a CT model to better track maneuvering targets, and also by the fact that the BP-PF+AIS and BP-PF+AIS+IMM methods create a larger number of false tracks. However, differently from the results reported in [1], the time-averaged mean GOSPA error of the BP-PF+AIS method is lower than that of the VIMMJIPDA method.

Figure 3 shows the mean GOSPA error versus time for $P_{\mathrm{D}} = 0.9$. Again differently from the results reported in [1], both the VIMMJIPDA+AIS method and the BP-PF+AIS and BP-PF+AIS+IMM methods correctly initialize the targets, as is demonstrated by their similar mean GOSPA errors at times $t = 0$ s and $t = 10$ s, i.e., when the targets appear. The slightly lower mean GOSPA error of the VIMMJIPDA+AIS method relative to the BP-PF+AIS and BP-PF+AIS+IMM methods can again be explained by the fact that the VIMMJIPDA+AIS method uses an additional CT model that allows it to maintain track continuity when targets maneuver and by the larger number of false tracks created by the BP-PF+AIS and BP-PF+AIS+IMM methods. This is confirmed by Table I, which reports the individual costs constituting the mean GOSPA error (averaged over time), i.e., the localization cost for correctly confirmed targets and the costs for missed and false targets. The larger number of false tracks created by the BP-PF+AIS and BP-PF+AIS+IMM methods is mainly due to the use of the heuristic described in [15] to model the generation of new targets, which was later superseded by the fully Bayesian BP-based tracking method proposed in [13].

Finally, Table II presents a comparison between the average computation times per time scan for all the methods. This comparison shows that the BP-PF+AIS method is the fastest method, even faster than the original VIMMJIPDA method that does not process the target-provided measurements. However, definite conclusions cannot be drawn from this analysis, given the different implementations, the different number of kinematic models used, and the different programming languages employed.

Table I
Time-Averaged Individual Costs Constituting the Mean GOSPA Error (in Meter) for the Simulated Scenario Considered in [1, Sec. VIII-A] with $P_{\mathrm{D}} = 0.9$.

	Localization	False	Missed
VIMMJIPDA+AIS	12.3	**0.3**	**12.7**
BP-PF+AIS	12.6	0.8	16.9
BP-PF+AIS+IMM	**11.2**	1.6	17.5

Bold font highlights the lowest value in each column.

Table II
Average Computation Times (in Second) per Time Scan for the Simulated Scenario Considered in [1, Sec. VIII-A].

	P_{D}					
	0.50	0.60	0.70	0.80	0.90	0.99
VIMMJIPDA	0.32	0.31	0.28	0.28	0.25	0.26
VIMMJIPDA+AIS	0.79	0.70	0.58	0.57	0.45	0.46
BP-PF+AIS	**0.20**	**0.19**	**0.20**	**0.21**	**0.21**	**0.21**
BP-PF+AIS+IMM	0.55	0.53	0.56	0.56	0.56	0.56

Bold font highlights the lowest value in each column.

B. Scenario Considered in [9]

Next, we present results for a simulated scenario that is similar to the one considered in [9, Sec. VI-A]. Our scenario consists of nine targets that are moving with a constant velocity of 4 m/s. The starting points of the target trajectories are equally spaced on a circle with center $[0, 0]^{\mathrm{T}}$ and a radius of 4 km. The target trajectories and the radar sensor are depicted in Figure 4. Unlike the scenario considered in the previous subsection, here the trajectories are deterministic—thus, they are equal for all simulation runs—and approximately cross each other

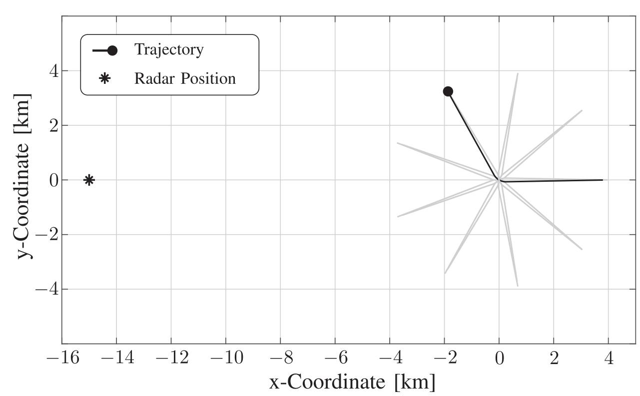


Figure 4. Simulated scenario considered in [9, Sec. VI-A]. The star marks the position of the radar sensor, and the dot indicates the final position of the highlighted trajectory. The other trajectories are rotated (by multiples of 40°) versions of the highlighted one.

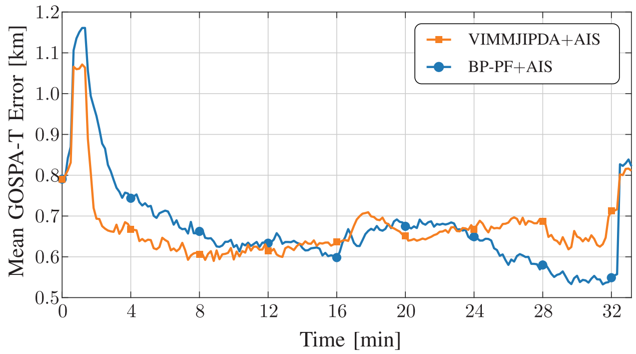


Figure 5. Mean GOSPA-T error versus time for the simulated scenario considered in [9, Sec. VI-A].

in $[0, 0]^{\mathrm{T}}$. Five targets appear at $t = 0$ s and do not disappear, and the other four targets appear at $t = 40$ s and disappear at about $t = 32$ min. Six randomly selected targets provide AIS measurements between $t = 1.5$ min and $t = 31.5$ min. The number of AIS measurements provided by a target during each time scan is Poisson distributed with mean 0.5 for three of the six targets and mean 1 for the other three targets. The time scan duration is set to 10 s. The AIS measurement noise is modeled as before. The radar detects a target with probability $P_{\mathrm{D}} = 0.5$, and it generates range-bearing measurements with a 2D zero-mean Gaussian measurement noise with covariance $\mathrm{diag}(250^2\ \mathrm{m}^2, 2.56^2\ \mathrm{deg}^2)$. The number of false alarms is Poisson distributed with mean 2.

For this scenario, both the BP-PF+AIS and VIMMJIPDA+AIS methods use a single NCV model with driving noise variance set to $0.15^2\ \mathrm{m}^2\,\mathrm{s}^{-3}$. The parameters for the BP-PF+AIS method are set as in [9]. For the VIMMJIPDA+AIS method, we use the parameters reported in [1, Tab. III] with the exception of the clutter density set to $1.7 \times 10^{-9}\ \mathrm{m}^{-2}$, the unknown target rate set to $10^{-10}\ \mathrm{m}^{-2}$, and the parameters related to the radar measurement noise, that is, the range measurement variance set to $250^2\ \mathrm{m}^2$ and the bearing measurement variance set to $2.56^2\ \mathrm{deg}^2$.

As previously done in [9], we compare the VIMMJIPDA+AIS and BP-PF+AIS methods in terms of the mean GOSPA error for trajectories (GOSPA-T) [18] of order 2 and with a cutoff parameter of 500 m, averaged over 100 simulation runs. Compared to the GOSPA error, the GOSPA-T error additionally accounts for track switches by adding a switching penalty of 125 m. One can see in Figure 5 that the VIMMJIPDA+AIS method outperforms the BP-PF+AIS method during approximately the first half of the simulation, that is, where the targets are well-separated. As the targets get closer, the difference between the GOSPA-T errors of the two methods becomes less significant. From minute 24, after the targets crossed their paths, the BP-PF+AIS method outperforms the VIMMJIPDA+AIS method. This is due to the inability of the VIMMJIPDA+AIS method to continue tracking some of the targets after they crossed their paths, as demonstrated by the higher time-averaged missed cost component of the mean GOSPA-T error shown in Table III. On the other hand, the time-averaged localization and false costs of the VIMMJIPDA+AIS method are lower than those of the BP-PF+AIS method. In terms of average computation time, the BP-PF+AIS method is faster than the VIMMJIPDA+AIS method: it requires 0.61 s to process each time scan, whereas the VIMMJIPDA+AIS method requires 0.81 s.

Table III
Time-Averaged Individual Costs Constituting the Mean GOSPA-T Error (in Meter) for the Simulated Scenario Considered in [9].

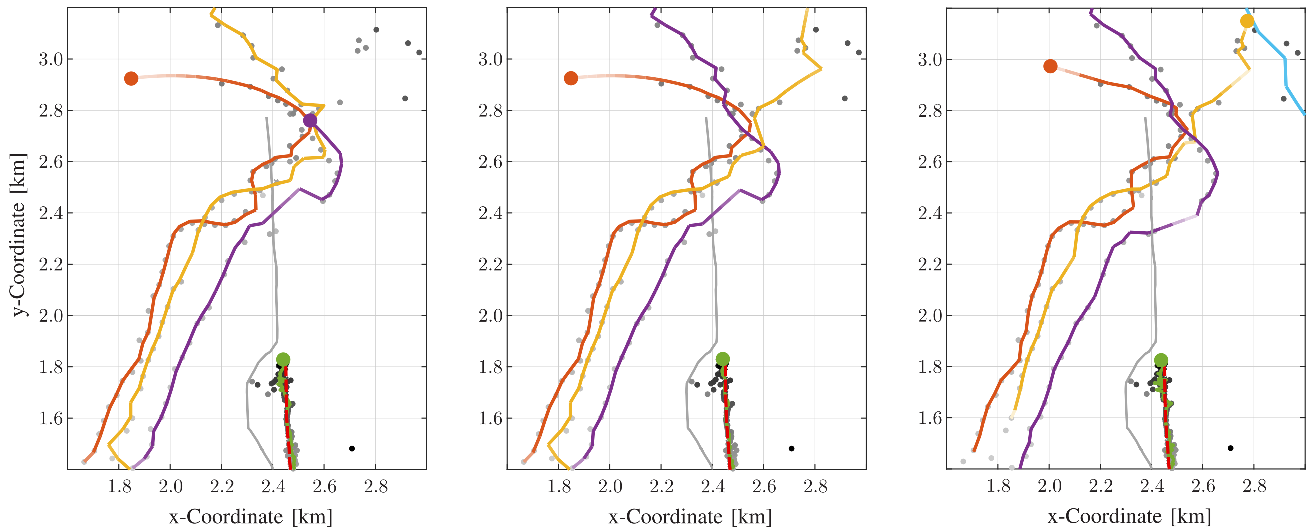
	Localization	False	Missed	Switch
VIMMJIPDA+AIS	**249.9**	**138.3**	532.3	**9.5**
BP-PF+AIS	325.0	271.7	**394.8**	10.5

Bold font highlights the lowest value in each column.

IV. RESULTS FOR REAL DATA

Finally, we assess and compare the performance of the VIMMJIPDA, VIMMJIPDA+AIS, and BP-PF+AIS methods for a real dataset that was acquired as part of the Autosea project [7]. The scenario now consists of a radar sensor mounted onboard a semi-autonomous surface craft and four unknown targets: a 30-m-long slow-moving vessel consistently providing AIS measurements and three fast-moving rigid-hull inflatable boats (RHIBs), one of which provides a single AIS measurement. The VIMMJIPDA and VIMMJIPDA+AIS methods employ three kinematic models as in [1]—two NCV models and one CT model—and use the parameters reported in [1, Tab. III]. The BP-PF+AIS method uses a single NCV model with driving noise variance set to $1.7^2\ \mathrm{m}^2\,\mathrm{s}^{-3}$, which is higher than the driving noise variances used for the VIMMJIPDA and VIMMJIPDA+AIS methods, and also noticeably higher than the driving noise variance used for the BP-PF+AIS method in [1]. Results obtained with the BP-PF+AIS+IMM method using two NCV models are not reported because they are equivalent to those obtained with the BP-PF+AIS method.

Figure 6 shows the trajectories estimated by the three methods as colored solid lines. The semiautonomous surface craft is sailing from north to south, and its trajectory, depicted as a gray solid line, is known. The unknown targets are traveling from south to north. The ground-truth trajectory of the slow-moving vessel, obtained by connecting its AIS measurements, is also depicted as a red dashed line; the ground-truth trajectories of the three fast-moving RHIBs are not available. Differently from the results reported in [1], Figure 6 shows that the BP-PF+AIS method performs better than the VIMMJIPDA method, which loses track of one of the three RHIBs when their paths cross, and



(a) VIMMJIPDA method (b) VIMMJIPDA+AIS method (c) BP-PF+AIS method

Figure 6. Trajectories estimated by (a) the VIMMJIPDA method, (b) the VIMMJIPDA+AIS method, and (c) the BP-PF+AIS method using a real dataset acquired as part of the Autosea project [7]. The estimated trajectories are depicted in orange, yellow, purple, and green, with their final positions indicated by large dots. The transparency of the tracks is related to their existence probability: lighter (darker) colors correspond to lower (higher) existence probabilities. The red dashed line indicates the ground-truth trajectory of the slow-moving vessel. The gray line represents the known trajectory of the radar sensor. The gray/black dots and crosses indicate the radar and AIS measurements, respectively; the measurements become darker as time passes by. The blue line in the top-right corner of the rightmost panel is a false track created by the BP-PF+AIS method.

performs almost identically to the VIMMJIPDA+AIS method. Despite using only a single NCV model, the BP-PF+AIS method is able to estimate the trajectories of all the targets with high accuracy. The drawbacks of using a higher driving noise variance than the driving noise variances used for the VIMMJIPDA and VIMMJIPDA+AIS methods and for the BP-PF+AIS method in [1] are manifested by the fact that the estimated trajectory for the slow-moving vessel exhibits abrupt changes of direction, and that a false track is created in the top-right corner of the considered area.

V. CONCLUSION

Recently, an extension of the VIMMJIPDA method that is able to include target-provided measurements was proposed in [1]. The effectiveness of this approach was validated in [1] through a comparison with the BP-PF+AIS method presented in [8], [9], whose code is not publicly available. In this paper, we presented the results of an experimental comparison using the implementation of the BP-PF+AIS method originally used in [8], [9], as well as the BP-PF+AIS+IMM method from [11]. Simulation results showed that the VIMMJIPDA+AIS method outperforms the BP-PF+AIS and BP-PF+AIS+IMM methods when the targets are well-separated, whereas the BP-PF+AIS and BP-PF+AIS+IMM methods have performance advantages in the case of closely spaced targets. The reason why the VIMMJIPDA+AIS method performs worse in the latter case is likely the limited performance of the data association scheme based on Murty's algorithm, which struggles when targets are closely spaced. Improvements to the VIMMJIPDA+AIS method can be obtained by resorting to the variational approximation method presented in [19]. However, due to its use of a CT kinematic model, the VIMMJIPDA+AIS method generally provides more accurate estimates when the targets perform sharp maneuvers. On the other hand, the BP-based data association algorithm used within the BP-PF+AIS and BP-PF+AIS+IMM methods tends to produce better results in challenging tracking environments with tighter target spacings [14]. Finally, results obtained with a real dataset showed that the BP-PF+AIS method using a single NCV kinematic model whose driving noise parameter is sufficiently high can track the agile RHIBs with performance comparable to that obtained with the VIMMJIPDA +AIS method.

REFERENCES

[1] A. G. Hem and E. F. Brekke
"Variations of joint integrated data association with radar and target-provided measurements,"
J. Adv. Inf. Fusion, vol. 17, no. 2, pp. 97–115, Dec. 2022.

[2] E. F. Brekke, A. G. Hem, and L.-C. N. Tokle
"Multitarget tracking with multiple models and visibility: Derivation and verification on maritime radar data,"
IEEE J. Ocean. Eng., vol. 46, no. 4, pp. 1272–1287, Oct. 2021.

[3] H. A. P. Blom and Y. Bar-Shalom
"The interacting multiple model algorithm for systems with Markovian switching coefficients,"
IEEE Trans. Autom. Control, vol. 33, no. 8, pp. 780–783, Aug. 1988.

[4] E. Tu, G. Zhang, L. Rachmawati, E. Rajabally, and G.-B. Huang "Exploiting AIS data for intelligent maritime navigation: A comprehensive survey from data to methodology," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 5, pp. 1559–1582, May 2018.

[5] M. Strohmeier, M. Schafer, V. Lenders, and I. Martinovic "Realities and challenges of NextGen air traffic management: The case of ADS-B," *IEEE Commun. Mag.*, vol. 52, no. 5, pp. 111–118, May 2014.

[6] H. Chen, T. Kirubarajan, and Y. Bar-Shalom "Performance limits of track-to-track fusion versus centralized estimation: Theory and application," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 39, no. 2, pp. 386–400, Apr. 2003.

[7] E. F. Brekke *et al.*, "The Autosea project: Developing closed-loop target tracking and collision avoidance systems," *J. Phys. Conf. Ser.*, vol. 1357, pp. 012020, Oct. 2019.

[8] D. Gaglione, P. Braca, and G. Soldi "Belief propagation based AIS/radar data fusion for multi–target tracking," in *Proc. FUSION-18*, Cambridge, UK, 2018, pp. 2143–2150.

[9] D. Gaglione, P. Braca, G. Soldi, F. Meyer, F. Hlawatsch, and M. Z. Win "Fusion of sensor measurements and target-provided information in multitarget tracking," *IEEE Trans. Signal Process.*, vol. 70, pp. 322–336, Dec. 2021.

[10] Online code repository: Variations of joint integrated data association with radar and target-provided measurements. [Online]. Available: https://doi.org/10.24433/CO.4125751.v2

[11] G. Soldi, F. Meyer, P. Braca, and F. Hlawatsch "Self-tuning algorithms for multisensor-multitarget tracking using belief propagation," *IEEE Trans. Signal Process.*, vol. 67, no. 15, pp. 3922–3937, Aug. 2019.

[12] K. G. Murty "An algorithm for ranking all the assignments in order of increasing cost," *Oper. Res.*, vol. 16, no. 3, pp. 682–687, Jun. 1968.

[13] F. Meyer *et al.*, "Message passing algorithms for scalable multitarget tracking," *Proc. IEEE*, vol. 106, no. 2, pp. 221–259, Feb. 2018.

[14] J. Williams and R. Lau "Approximate evaluation of marginal association probabilities with belief propagation," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 50, no. 4, pp. 2942–2959, Oct. 2014.

[15] F. Meyer, P. Braca, F. Hlawatsch, and P. Willett "A scalable algorithm for tracking an unknown number of targets using multiple sensors," *IEEE Trans. Signal Process.*, vol. 65, no. 13, pp. 3478–3493, Jul. 2017.

[16] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan *Estimation with Applications to Tracking and Navigation*. New York, NY, USA: Wiley, 2001.

[17] A. S. Rahmathullah, Á. F. García-Fernández, and L. Svensson "Generalized optimal sub-pattern assignment metric," in *Proc. FUSION-17*, Xi'an, China, 2017, pp. 1–8.

[18] Á. F. García-Fernández, A. S. Rahmathullah, and L. Svensson "A metric on the space of finite sets of trajectories for evaluation of multi-target tracking algorithms," *IEEE Trans. Signal Process.*, vol. 68, pp. 3917–3928, Jun. 2020.

[19] J. Williams "An efficient, variational approximation of the best fitting multi-Bernoulli filter," *IEEE Trans. Signal Process.*, vol. 63, no. 1, pp. 258–273, Jan. 2015.

Domenico Gaglione (Member, IEEE) received the B.Sc. and M.Sc. degrees (summa cum laude) in telecommunications engineering from Università degli Studi di Napoli Federico II, Naples, Italy, in 2011 and 2013, respectively, and the Ph.D. degree from the Department of Electronic and Electrical Engineering, University of Strathclyde, Glasgow, UK, in 2017. He is currently a Research Scientist with NATO STO Centre for Maritime Research and Experimentation (CMRE), La Spezia, Italy. His research interests include statistical signal processing with emphasis on state estimation, data fusion, and multisensor multitarget tracking.

Paolo Braca (Senior Member, IEEE) received the Laurea degree (summa cum laude) in electronic engineering and the Ph.D. degree (Hons.) in information engineering from the University of Salerno, Fisciano, Italy, in 2006 and 2010, respectively. In 2009, he was a Visiting Scholar with the Department of Electronics and Communication Engineering, University of Connecticut, Storrs, CT, USA. From 2010 to 2011, he was a Postdoctoral Associate with the University of Salerno. In 2011, he joined the NATO STO CMRE, where he is currently a Senior Scientist and a Project Manager. He conducts research in the general area of statistical signal processing with emphasis on detection and estimation theory, wireless sensor network, multi-agent algorithms, target tracking and data fusion, adaptation and learning over graphs, radar (sonar) signal processing, and machine learning. He has coauthored more than 200 publications in international scientific journals, conference proceedings, and NATO technical reports.

Giovanni Soldi received the master's degree in applied mathematics from the University of Milan, Milan, Italy, in 2011, and the Ph.D. degree in signal processing from Télécom ParisTech, Paris, France, in 2016. Since 2016, he has been a Scientist with the Centre for Maritime Research and Experimentation (CMRE), La Spezia, Italy. His research interests include statistical signal processing with a focus on state estimation, data fusion, and multisensor-multitarget tracking techniques.

Florian Meyer (Member, IEEE) received the M.Sc. and Ph.D. degrees (with highest honors) in electrical engineering from TU Wien, Vienna, Austria in 2011 and 2015, respectively. He is an Assistant Professor with the University of California San Diego, La Jolla, CA, jointly between the Scripps Institution of Oceanography and the Electrical and Computer Engineering Department. From 2017 to 2019, he was a Postdoctoral Fellow and Associate with the Laboratory for Information & Decision Systems at the Massachusetts Institute of Technology, Cambridge, MA, and from 2016 to 2017, he was a Research Scientist with the NATO Centre for Maritime Research and Experimentation, La Spezia, Italy. Prof Meyer's research interests include statistical signal processing, high-dimensional and nonlinear estimation, inference on graphs, machine perception, and graph neural networks.

Audun Gullikstad Hem received his M.Sc. in Engineering Cybernetics from the Norwegian University of Science and Technology (NTNU) in 2021, and is currently a Ph.D. student at the Department of Engineering Cybernetics at NTNU. He specializes in target tracking for autonomous surface vehicles, and is currently working in the Autosit project.

Edmund Brekke (Senior Member, IEEE) received the M.Sc. degree in industrial mathematics in 2005, and the Ph.D. degree in engineering cybernetics in 2010, both from the Norwegian University of Science and Technology (NTNU), Trondheim, Norway. From 2010 to 2014, he worked with the Acoustic Research Laboratory (ARL), NUS in Singapore as a postdoctoral Research Fellow. In 2014 he rejoined NTNU and the Department of Engineering Cybernetics, and is currently a Professor in sensor fusion. Brekke's research interests are in the area of sensor fusion and situational awareness, with a particular focus on multi-target tracking and maritime surface autonomy.

Franz Hlawatsch received the Diplom-Ingenieur, Dr. techn., and Univ.-Dozent (habilitation) degrees in electrical engineering/signal processing from TU Wien, Vienna, Austria, in 1983, 1988, and 1996, respectively. Since 1983, he has been with the Institute of Telecommunications, TU Wien, where he is currently an Associate Professor. During 1991–1992, as a recipient of an Erwin Schrödinger Fellowship, he spent a sabbatical year with the Department of Electrical Engineering, University of Rhode Island, Kingston, RI, USA. In 1999, 2000, and 2001, he held one-month Visiting Professor positions with INP/ENSEEIHT, Toulouse, France and IRCCyN, Nantes, France. He (co)authored a book, three review papers that appeared in the IEEE Signal Processing Magazine, about 260 refereed scientific papers and book chapters, and three patents. He coedited three books. His research interests include statistical signal and data processing methods and their application to inference and learning problems.

Data Fusion for Optimal Condition-Based Aircraft Fleet Maintenance With Predictive Analytics

ZHENGYANG FAN
KUO-CHU CHANG
RAN JI
GENSHE CHEN

Maintaining and deploying a fleet of aircraft with limited resources and various mission requirements is both immensely challenging and of primary importance. Traditional preventive maintenance methods are static and inflexible and are not equipped to consider the complex dynamics of aircraft (e.g., wear and age), which may lead to low fleet availability and high maintenance costs. In this paper, we propose an integrated learn-then-optimize framework for condition-based predictive maintenance scheduling to support daily flight and maintenance planning by fusing data from multiple onboard sensors. The paradigm first predicts the remaining useful life for components of aircraft by using deep learning techniques, then models the fleet-level optimization as a constrained mixed-integer programming problem that captures different failure modes of aircraft and the available maintenance facilities. We also propose valid inequalities to improve the computational efficiency of the optimization model. Finally, we conduct a series of simulated experiments to validate the performance of the proposed predictive maintenance model. The numerical results show that the predictive maintenance model outperforms the traditional preventive maintenance model with respect to the mission accomplishment rate, aircraft availability rate, and cost effectiveness.

Manuscript received October 26, 2022; revised November 5, 2023; released for publication April 4, 2024

Refereeing of this contribution was handled by Ramona Georgescu.

Z. Fan, K.-C. Chang, and R. Ji are with the Department of Systems Engineering and Operations Research, George Mason University, Fairfax, VA, 22030, USA (e-mail: zfan3@gmu.edu; kchang@gmu.edu; rji2@gmu.edu).

G. Chen is with the Intelligent Fusion Technology Inc., Germantown, MD, 20874, USA (e-mail: gchen@intfusiontech.com).

This work was supported by the Naval Air Warfare Center under Contract no. N68335-20-C-0306. Any opinions, findings, and conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of NAVAIR.

1557-6418/23/$17.00 © 2023 JAIF

I. INTRODUCTION AND RELATED WORK

Flight and maintenance planning (FMP) for military aircraft aims to identify optimal scheduling for a given fleet by (1) determining which aircraft are available to fly and for what duration, and (2) if and when to conduct maintenance on grounded aircraft. FMP aims to accomplish these mission tasks efficiently while also keeping operational costs at a minimum. FMP plays a critical role in guaranteeing the safety and reliability (e.g., fleet readiness rate and mission accomplishment rate) of military or commercial airlines.

Modern surface and aviation systems are designed with an ever-increasing level of automation and advanced machinery that include state-of-the-art sensors that monitor vital aircraft, ship, and auxiliary system functions. New tools and technologies are needed to augment current onboard condition monitoring and maintenance processes, improve system availability, increase operational readiness, and reduce life cycle costs. Along with the development of Industrial 4.0, which integrates sensors, software, and intelligent control to improve industrial processes, aircraft maintenance is transitioning from more traditional processes of corrective and preventive maintenance to a data-driven, predictive maintenance paradigm. While there have been significant strides made in utilizing machine learning (ML) and augmented intelligence (AI) for predictive maintenance, there is still a need to develop new tools that can produce more efficient and accurate condition-based predictive maintenance (CBPM) decisions [39]. Predictive maintenance involves analyzing machine data collected from various monitoring sensors, such as thermal, acoustic, vibration, pressure, and temperature, to generate meaningful insights about the machine's state, including failure classification, remaining useful life (RUL), and time to failure (TTF) [37]. The ultimate objective is to schedule proactive maintenance more accurately, enhance readiness, and improve efficiency in the logistics and supply chain. Such capability becomes especially crucial for mission-critical systems, ensuring sustained combat operations and readiness while minimizing costs and unplanned downtime.

The predictive maintenance approach fuses the data from on-board sensors to monitor the health condition of aircraft components to predict RUL prognostics of system and identify anomalous behavior, and thus turn equipment sensor data into meaningful, actionable insights for proactive maintenance in the anticipation of failure [39]. There are two main challenges in RUL-based predictive maintenance and flight scheduling problems. The first challenge is how to accurately predict the RUL prognostics for system components by exploiting the data from multiple sensors. The second challenge is the integration of RUL prognostics into FMP, considering the workforce capacity (e.g., availability of workstations and technicians to repair the components), flight mission requirements (e.g., type and number of aircraft required to conduct different missions), and system reliability requirements.

For RUL prediction, most existing studies fall into three categories [33]: statistical-based models, conventional ML models, and deep learning models. Statistical-based models are built by fitting a probabilistic model to data by assuming that the degradation of system components over time can be characterized via an appropriate parametric function or a specified stochastic process model. For example, the Wiener process has been successfully employed to capture the degradation of components in bridge beams [31], thrust ball bearings [30], and micro-electron mechanical systems [35]. Gaussian process regression model is used to predict RUL prognostics of battery health [23] and slow-speed bearings [1]. The conventional ML algorithms, such as support vector machines, tree-based methods, and neural networks, have been extensively used in predictive maintenance in the past decades. For example, [28] developed a support vector regression model with a multi-class solver to identify various faulty patterns in rotating machines. Reference [17] constructed a random forest regression model to predict the RUL of spur gears. More recently, with the expansion of big data techniques, the popularity of deep learning algorithms for predictive maintenance has noticeably increased. Reference [12] developed recurrent neural networks (RNNs) for RUL prediction of bearings. Reference [34] designed a double convolutional neural network (CNN) architecture to predict RUL using time-series vibration signals. Reference [38] employed long short-term memory (LSTM) RNN to learn the long-term dependencies among degraded capacities and predict the RUL of lithium-ion batteries. We refer interested readers to [2] and [4] for a comprehensive survey of ML approaches, and to [27], particularly for deep learning approaches in predictive maintenance.

Numerous efforts have also been devoted to military aircraft fleet scheduling optimization. From the military perspective, one major concern in this problem is operational readiness [21], [24]. Reference [24] formulated a mixed-integer programming (MIP) model to maximize fleet availability under skilled workforce constraints. The model admits a network flow interpretation and can be solved efficiently by the branch-and-bound method. Reference [14] proposed a multiobjective MIP model to maximize fleet availability. To further improve the computational efficiency, [16] and [15] extended the work of [3] and developed heuristic algorithms to solve large-scale problem instances. Instead of directly maximizing fleet availability, [19] and [3] minimized the maximum number of aircraft in maintenance to be greater than the number of available maintenance spaces over the planning horizon. [21] introduced an MIP model for long-term planning of military aircraft by considering type-D heavy maintenance. Another major challenge for the FMP problem is computational efficiency. Some heuristic algorithms have been proposed to solve large-scale problem instances, for example, [8], [15], [16]. To obtain exact solutions with computational efficiency, [9] proposed an iterative algorithm that cuts off infeasible relaxation solutions via special valid inequalities. Reference [10] modify the classical ϵ-constraint method to solve a biobjective quadratic program. More recently, [22] used ML models to predict the characteristics of optimal solutions, and added these characteristics to the original formulations to shrink solution space.

Though RUL prediction and maintenance scheduling optimization have received a considerable amount of attention from their own domains, very few studies have developed RUL prognostics and subsequently integrated the predicted RUL into maintenance scheduling. Reference [20] built an LSTM neural network to predict multiclass RUL prognostics for turbofan engines of aircraft, which are used subsequently to order and manage replacement spare parts. Reference [6] developed a particle filtering algorithm for RUL prediction of aircraft cooling units, and the predicted RUL was then passed to a linear programming optimization model to optimally schedule a fleet of aircraft maintenance considering spare parts. More recently, in the work of [18], the authors considered the maintenance of aircraft brakes using a threshold-based maintenance policy, i.e., once the predicted RUL falls below a user-defined threshold, the brake is replaced. They solved a multi-objective scheduling optimization model that seeks a trade-off between the minimization of flight delays, the number of unscheduled maintenance tasks, and the total number of maintenance tasks. Using predicted RUL as model coefficients, [32] proposed a multiobjective genetic algorithm for maintenance scheduling for a vehicle fleet by minimizing total cost, workload, and the expected number of failures and total changes in maintenance schedule. Most recently, [7] studied the alarm-based maintenance planning with imperfect RUL predictions for a fleet of vehicles by considering estimation uncertainties.

In this work, we propose an integrated learn-then-optimize framework for flight and maintenance scheduling with RUL predictions to maximize fleet-level operational availability and minimize costs. The proposed framework first employs advanced analytics from multiple onboard sensory data to draw meaningful insights to predict machine states and proactively schedule maintenance and flights to minimize costs and unplanned downtime. More specifically, we first develop a bidirectional long short-term memory (biLSTM) deep learning model to combine the time-series monitoring data for predicting the RUL of aircraft system components, and subsequently incorporate the RUL into an optimization model to determine the optimal fleet-level maintenance policies and flight scheduling by considering practical constraints such as workforce capacity and mission requirements. To the best of our knowledge, this paper represents the first study that explicitly considers three elements, i.e., (1) deep learning with multisensor data for RUL prediction, (2) predictive maintenance scheduling, and (3) flight mission planning into an integrated learn-then-optimize framework. Our work differs from previous research in the sense that previous studies only focused on RUL prediction and predictive maintenance scheduling, while our proposed framework also takes into account the flight mission planning decision, workforce capacity constraints (e.g., different types and number of technicians required to conduct maintenance for different components of an aircraft system), and mission requirements (e.g., type and number of aircraft required to conduct specific missions), which are of particular importance to military applications.

The remainder of the paper is structured as follows. In Section II, we discuss the problem description and learn-then-optimize framework. Section III briefly introduces bidirectional LSTM and discusses the RUL prediction using multiple commercial modular aero-propulsion system simulation (C-MAPSS) engine sensory data. Section IV presents the MIP formulation for the FMP optimization model, followed by a set of valid inequalities to boost the computational speed. Various numerical experiments are conducted in Section V to demonstrate the superiority of the proposed predictive maintenance model over the traditional preventive maintenance method. In Section VI, we conclude our work.

II. PROBLEM DESCRIPTION

Heavy equipment maintenance facilities such as aircraft service centers face the challenge of maximizing readiness while minimizing the costs of various maintenance tasks, subject to the availability constraints of specialty technicians, workstations, spare inventory, and mission requirements. FMP require making decisions about which aircraft should perform a mission and which aircraft should enter maintenance, while optimizing efficiency with limited resources, including technical workforce and maintenance workstations. Traditional maintenance scheduling is done via preventive maintenance, i.e., a fixed schedule to maintain each aircraft periodically to prevent failure. The proposed approach is to replace the preventive maintenance with a predictive maintenance strategy where the machinery conditions (such as RUL) are considered in order to proactively schedule the maintenance. The proposed approach is designed within a learn-then-optimize framework, in which we first apply deep learning to analyze the time-series monitoring sensory data for predicting equipment RUL and subsequently incorporate it into an optimization formulation to determine optimal fleet-level maintenance policies. In this section, we briefly introduce the elements of the proposed learn-then-optimize framework, covering the description of FMP problem setting, multicomponent aircraft system, and workforce capacity consideration.

A. Problem Setting for FMP

We consider the problem of scheduling a set of predictive maintenance tasks with a given number of available maintenance stations and workforce capacity constraints. For each type of maintenance task, the skills required and the number of technicians with the skills needed to work on the task are assumed to be known (similar setting used in [24]). The tasks are to be performed at the available workstations by the needed number of technicians with a specific skill or multiple skills. All of the skilled technicians required for the set of tasks to be performed must be available in the given time frame.

The operational goal is to maximize the readiness of a fleet of aircraft with a certain mission requirement given the constraints. The key is to ensure sufficient availability of aircraft to meet the operational demands, which refer to all the flight activities (namely, waves or sorties) that are planned in a given period (say, 24 h). A mission flight of a single aircraft is called a sortie. More than one aircraft flying together is called a wave or mission.

Predictive planning is assumed to be executed daily based on the estimated RUL of each component for the aircraft. The 24-h planning horizon is assumed to begin at 6:00 pm (see [24]). The objective is to determine the schedule (i.e., the sequence in which the maintenance tasks are executed) with the maximum availability-to-cost ratio. Note that the optimal solution to this problem will yield a maintenance schedule in which (a) all mission tasks in the set will be performed as much as possible, (b) the available workforce and workstations will be utilized in an efficient manner, and (c) the minimum possible time to perform maintenance will be determined. Hence, a solution to this problem is critical to reducing maintenance downtime and maximizing fleet operational readiness.

B. Multicomponent Aircraft System and Workforce Capacity

In this study, we consider the FMP problem with a fleet of aircraft, where each aircraft has a system of multiple repairable or replaceable components. Each aircraft is assumed to be of a specific type that can perform designated mission tasks. Each component of aircraft is assumed to fail independently of other components. Repair or replacement maintenance is scheduled based not only on the estimated RUL of this component in anticipation of a failure but also on the task's durational requirements. For example, suppose that the threshold value for triggering the replacement is when RUL reduces to 3 h. If a component has 4 h of RUL, then typically it will not be scheduled for maintenance and will be allowed to take on a task. However, if a mission task requires the flying duration to be 5 h, then this aircraft cannot fulfill this requirement and must either undergo maintenance to replace the component or be assigned to fly other missions. In this study, we assume each aircraft has a multicomponent system, each component is independent from the others, and each component requires different technician specialties for repair or replacement. Generally, while flying is underway (and also immediately before and after), technicians are divided into three groups/specialties to perform the following activities:

- Trade 1: Weapons and armament electrical (WP)
- Trade 2: Airframe mechanical, airframe electrical and propulsion (AF)
- Trade 3: Avionics/electronics (AV)

During actual operations, each type of failure/maintenance requires several types/trades of technicians for service. For example, engine failure may require two AF technicians and three AV technicians to perform the repair, and radar failure may require one WP technician and one AV technician for repair. The explicit incorporation of mission requirements (e.g., type of aircraft and flying duration) as well as the workforce capacity (e.g., number and types of specialty technicians) increases the practicability and complexity of the FMP problem significantly.

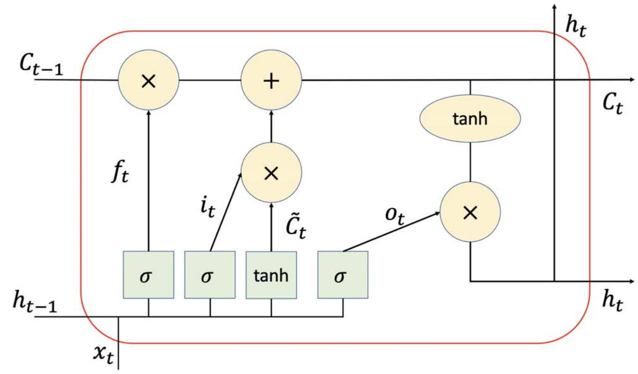


Figure 1. A vanilla LSTM cell [29].

In the cell, the functional relationships for each component are given as follows:

$$
\begin{aligned}
f_t &= \sigma\left(W_f \cdot x_t + R_f \cdot h_{t-1} + b_f\right)\\
i_t &= \sigma\left(W_i \cdot x_t + R_i \cdot h_{t-1} + b_i\right)\\
o_t &= \sigma\left(W_o \cdot x_t + R_o \cdot h_{t-1} + b_o\right)\\
\tilde{C}_t &= \varphi\left(W_c \cdot x_t + R_c \cdot h_{t-1} + b_c\right)\\
C_t &= f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t\\
h_t &= o_t \cdot \phi\left(C_t\right),
\end{aligned}
\tag{1}
$$

where f_t, i_t, and o_t stand for forget gate, input gate, and output gate, respectively. Forget gate removes historical information from C_{t-1}; input and output gates control to update and output which part of information. σ, φ, and ϕ are nonlinear activation functions.

In order to obtain smooth states estimation of LSTM networks, bidirectional LSTM was proposed in [26], where a backward path is added to smooth out the prediction. Bidirectional LSTM thus can utilize the information in both forward and backward directions, which makes it suitable for intermediate prediction. Figure 2 describes the architecture of bidirectional LSTM.

Formulas for each component in backward path are almost identical to the forward LSTM model, which are

III. DEEP LEARNING FOR RUL PREDICTION

The accurate prediction of RUL prognostics provides important input to the scheduling optimization model. In this section, we will showcase our study on how to use LSTM deep learning approaches to predict RUL using a C-MAPSS engine degradation dataset [25].

A. LSTM RNN

LSTM is one of the most widely used RNN architecture in time sequence ML modeling. It uses gates to control information flow in the recurrent computations and is excellent at holding long-term memories. Its gating mechanisms are ideally suitable for modeling the machinery degradation process [37]. The original version of LSTM was proposed in [13]. The popular Vanilla LSTM was introduced in [11], where forget gate was added to the LSTM architecture to improve the model. Figure 1 shows the vanilla LSTM cell [29]. LSTM memory cells consist of different neural networks, which are called gates. Gates are used to track the interactions between the memory units, and to decide which data should be remembered or forgotten during the training process. The input gate and output gate determine if the state memory cells can be modified by the input signal. The forget gate controls whether or not to forget the previous status of the signal.

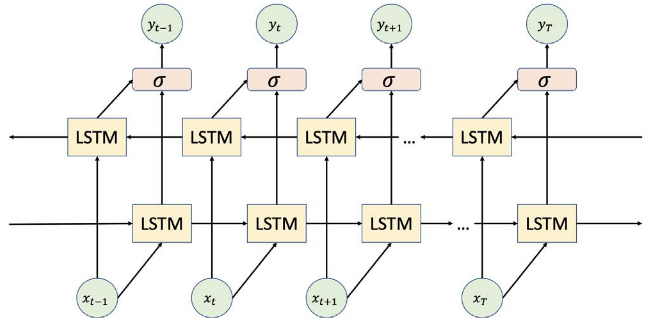


Figure 2. Bidirectional LSTM architecture.

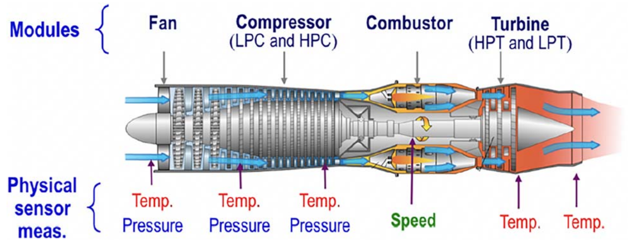


Figure 3. Turbofan engine model [37].

given below:

$$
\begin{aligned}
f'_t &= \sigma\left(W'_f \cdot x'_t + R'_f \cdot h'_{t-1} + b'_f\right)\\
i'_t &= \sigma\left(W'_i \cdot x'_t + R'_i \cdot h'_{t-1} + b'_i\right)\\
o'_t &= \sigma\left(W'_o \cdot x'_t + R'_o \cdot h_{t-1} + b'_o\right)\\
\tilde{C}'_t &= \varphi\left(W'_c \cdot x'_t + R'_c \cdot h'_{t-1} + b'_c\right)\\
C'_t &= f'_t \cdot C'_{t-1} + i'_t \cdot \tilde{C}'_t\\
h'_t &= o'_t \cdot \phi\left(C'_t\right),
\end{aligned}
\tag{2}
$$

where f'_t, i'_t, and o'_t denote forget gate, input gate, and output gate, respectively (analogous to f_t, i_t and o_t as in LSTM model).

Table I
C-MAPSS Monitoring Sensor Data

Symbol	*Description*	*Units*
T2	Total temperature at fan inlet	R
T24	Total temperature at LPC outlet	R
T30	Total temperature at HPC outlet	R
T50	Total temperature at LPT outlet	R
P2	Pressure at fan inlet	psia
P15	Total pressure in bypass-duct	psia
P30	Total pressure at HPC outlet	psia
Nf	Physical fan speed	rpm
Ne	Physical core speed	rpm
epr	Engine pressure ratio (P50/P2)	–
Ps30	Static pressure at HPC outlet	psia
phi	Ratio of fuel flow to Ps30	pps/psi
NRf	Corrected fan speed	rpm
NRe	Corrected core speed	rpm
BPR	Bypass ratio	–
farB	Burner fuel-air ratio	–
htBleed	Bleed Enthalpy	–
Nf-dmd	Demanded fan speed	rpm
PCNfR-dmd	Demanded corrected fan speed	rpm
W31	HPT coolant bleed	lbm/s
W32	LPT coolant bleed	lbm/s

B. RUL Prognostics Prediction for Aircraft Engines

C-MAPSS is a tool to simulate performance of the turbofan engine, which is built under MATLAB and Simulink environment (see [25] for details). As shown

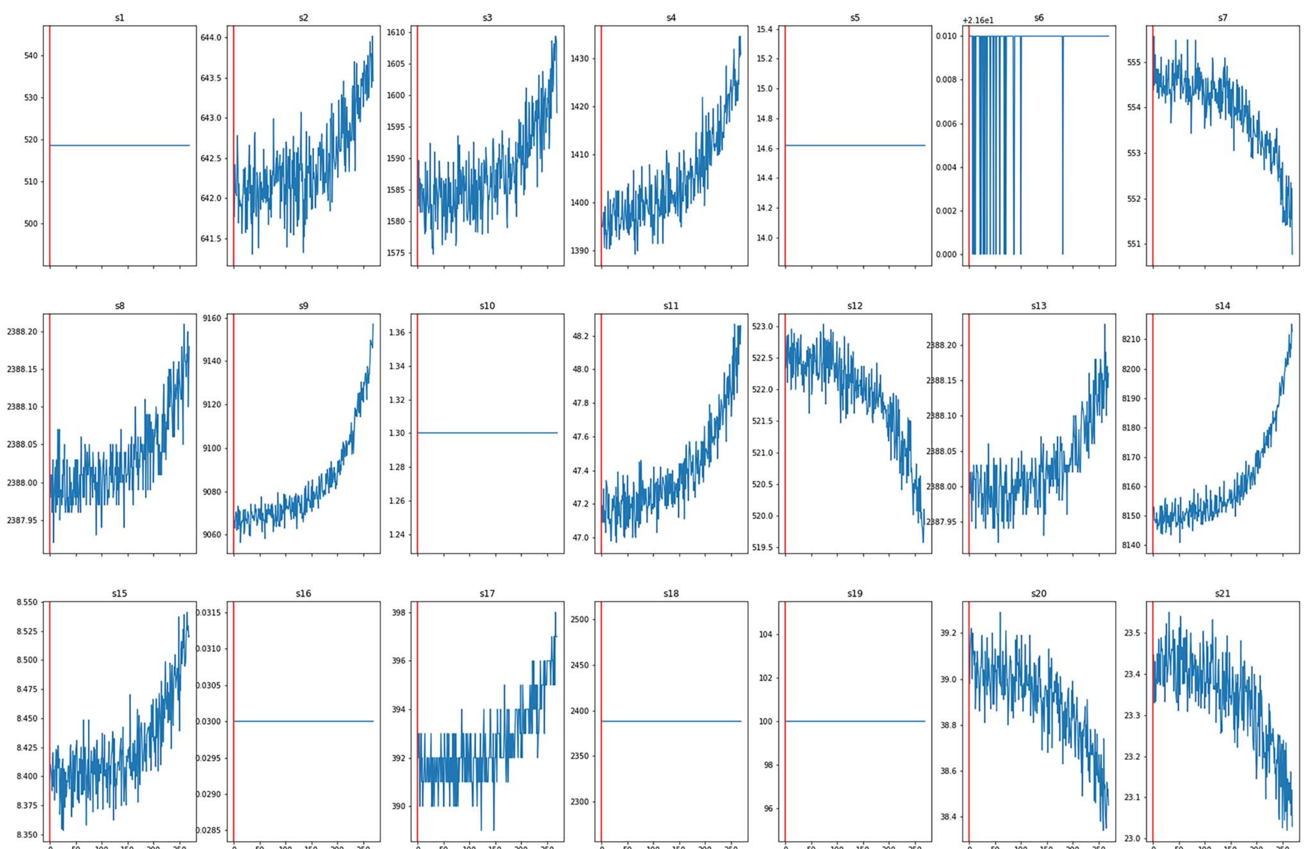


Figure 4. Run-to-failure sensor data examples.

in Fig. 3 (adapted from [37]), a turbofan engine typically consists of five modules: fan, low-pressure turbine (LPT), high-pressure turbine (HPT), low-pressure compressor (LPC) and high-pressure compressor (HPC).

The C-MAPSS dataset [25] includes hundreds of engine profiles with 21 onboard sensors monitoring the engine's health status (see Table I for descriptions of the sensors).

Three operation condition indicators (altitude, Mach numbers, and throttle resolver angle) are included as part of the observations as well. As an example, the C-MAPSS FD001 data set includes 200 engine profiles, 100 of which constitute a training set, where the historical run-to-failure measurement records are included. Figure 4 shows a run-to-failure data trajectories example from the 21 sensors under a specific operating condition. In the remaining testing set, sensor measurements are only recorded up to an early stage, and the goal is to predict the remaining engine life.

C. LSTM Training and Testing

In order to train the LSTM model and predict RUL given the temporal degradation process in time-series data, we reshape input sensors' readings into blocks of size (30, 24), that is, a series of 30 consecutive sensor measurements (21 sensors) together with the operating condition indicator (3 indicators). For each input data block, the output is the corresponding remaining life at the last cycle in the time-series. To train a bidirectional LSTM model, the objective function is defined as the empirical mean squared error. Here, we select a two layer bidirectional LSTM with 145 neurons for each forward and backward layer in the network architecture. In the learning process, we apply the stochastic gradient descent (SGD) method with batch size 50 and learning rate of 0.0015. We test the RUL prediction performance with the C-MAPSS data. For example, Fig. 5 shows the comparison of true RUL and predicted RUL for testing dataset FD001. The root mean squared error (RMSE) obtained after testing 100 engines was found to be 15.16 operational cycles. This signifies that the average prediction error for the RUL of the engines is 15.16 operational cycles. Given that these results align with the best state-of-the-art algorithms' performance [36], this RMSE value is deemed a positive indicator of the predictive accuracy of our model. By fusing the 21 sensory time-series data with the bidirectional LSTM, the resulting RUL predictions are good indications of engine health condition and will be used to determine the optimal maintenance schedule.

IV. PREDICTIVE MAINTENANCE AND MISSION SCHEDULING OPTIMIZATION

In this section, we first present our MIP formulation for FMP optimization model in Section IV-A, then in Section IV-B, we present two sets of valid inequalities to further improve the computational efficiency.

A. Model Formulation

We introduce the notations used in the optimization model in Table II.

With these notations, the objective of the optimization model as defined in (3) is to maximize the normalized availability. We define normalized availability as the ratio of the weighted sum of available aircraft over the total maintenance cost over the planning horizon,

$$\max \quad \frac{\sum_{t=1}^{T}\sum_{k=1}^{K} M_t^k \cdot \left(\sum_{i=1}^{A_k}\left(1 - z_{it}^k\right)\right)}{\sum_{t=1}^{T}\sum_{k=1}^{K}\sum_{i=1}^{A_k}\sum_{f=1}^{F} c_f^k x_{itf}^k}. \qquad (3)$$

The weight values indicate the relative importance of mission in a particular operational cycle; thus, higher weights are assigned to more critical missions. The denominator represents the total maintenance cost in the planning horizon. The optimal maintenance schedules

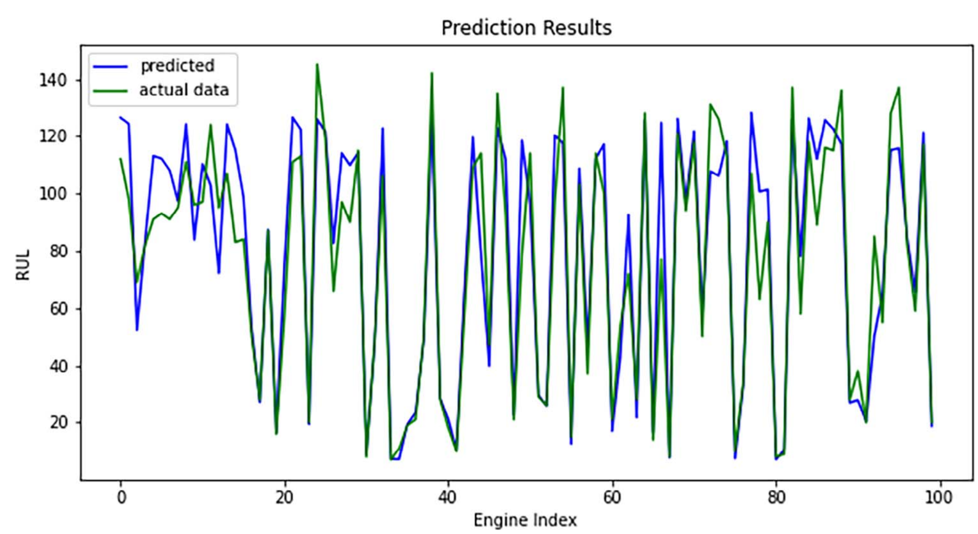


Figure 5. RUL prediction results on FD001 dataset.

Table II
Notation

Sets:	
T	Planning horizon ($t = 1, 2, ..., T$)
K	Type of aircraft ($k = 1, 2, ..., K$)
F	Type of failure mode/component ($f = 1, 2, ..., F$)
R	Type of technicians ($r = 1, 2, ..., R$)
Parameters:	
S	Number of the maintenance station
A_k	Number of type-k aircraft
M_t^k	Number of required aircraft of type k conducting missions at time t
D_t^k	Mission duration of aircraft type k at time t
I_{fr}^k	Number of technicians of trade r required to rectify the failure mode f for aircraft type k
E_{fr}^k	Maintenance time of trade r required to rectify the failure mode f for aircraft type k
λ_r	Number of technicians of trade r available at the initial period of the planning horizon
r_{if}^k	Initial RUL for component f of aircraft i of type k
l_{tf}^k	Duration of mission for component f of type k aircraft at time t
c_f^k	Fixed charge cost for repairing component/failure mode f for one aircraft of type k
E_{i0fr}^k	Remaining maintenance time for trade r to repair aircraft i of type k
$r_{max,f}^k$	RUL for component f of aircraft type k after maintenance (i.e., the maximum RUL)
Decision Variables:	
x_{itf}^k	$= \begin{cases} 1 & \text{if aircraft } i \text{ of type } k \text{ enters maintenance station in time } t \text{ to rectify failure mode } f \\ 0 & \text{otherwise} \end{cases}$
z_{itfr}^k	$= \begin{cases} 1 & \text{if aircraft } i \text{ of type } k \text{ is in maintenance in time } t \text{ to rectify failure mode } f \text{ by trade } r \\ 0 & \text{otherwise} \end{cases}$
z_{itf}^k	$= \begin{cases} 1 & \text{if aircraft } i \text{ of type } k \text{ is in maintenance in time } t \text{ to rectify failure mode } f \\ 0 & \text{otherwise} \end{cases}$
z_{it}^k	$= \begin{cases} 1 & \text{if aircraft } i \text{ of type } k \text{ is in maintenance in time } t \\ 0 & \text{otherwise} \end{cases}$
y_{it}^k	$= \begin{cases} 1 & \text{if aircraft } i \text{ of type } k \text{ conducts mission in time } t \\ 0 & \text{otherwise} \end{cases}$
r_{itf}^k	RUL of component/failure mode f for aircraft i of type k in time t

are to be obtained subject to a number of constraints, as shown below.

Mission Requirement Constraint

$$\sum_{i=1}^{A_k} y_{it}^k = M_t^k \qquad t \in T, k \in K. \quad (4)$$

Constraint (4) enforces the demands of mission requirements to be satisfied for aircraft type k in each period t. That is, the sum of all assigned missions needs to satisfy the mission requirement.

RUL Dynamic Constraints

$$r_{i1f}^k = r_{if}^k \qquad \forall i \in I, k \in K, f \in F, \quad (5)$$

$$r_{i,t+1,f}^k \leq r_{itf}^k - y_{it}^k \cdot l_{tf}^k + r_{\mathrm{max,f}}^k \cdot x_{itf}^k \qquad t = 1, ..., T-1, k \in K, i \in I, f \in F, \quad (6)$$

$$r_{i,t+1,f}^k \geq r_{itf}^k - y_{it}^k \cdot l_{tf}^k \qquad t = 1, ..., T, k \in K, i \in I, f \in F, \quad (7)$$

$$r_{i,t+1,f}^k \leq r_{\mathrm{max,f}}^k \qquad t = 1, ..., T-1, k \in K, i \in I, f \in F, \quad (8)$$

$$r_{i,t+1,f}^k \geq r_{\mathrm{max,f}}^k \cdot x_{itf}^k \qquad t = 1, ..., T-1, k \in K, i \in I, f \in F, \quad (9)$$

Constraints (5)–(9) model the behavior of RUL in failure mode f for aircraft i of type k at time t. In other words, these constraints model the dynamics of components' RUL when a mission assignment or maintenance activity is conducted. Note that our assumption is that different components of an aircraft may exhibit different rates of degradation when a mission is conducted, resulting in varying reductions in the remaining operational cycles for each component, thus leading to different durations of missions for different components. For instance, when an aircraft undertakes a mission with a duration of 2 h, its weapon-related components may reduce their RUL by one operational cycle, while electrical and propulsion components may reduce their RUL by two operational cycles. However, in our numerical experiment section, we've simplified this by assuming a global reduction of RUL for all components, implying that all components experience the same RUL reduction.

Maintenance State Dynamic Constraints

$$x_{itf}^k \le z_{i,t+y,f,r}^k \qquad t = E_{i0fr}^k + 1, \dots, T - E_{fr}^k, \quad y = 1, \dots, E_{fr}^k, i \in I, k \in K, r \in R, \tag{10}$$

$$m \cdot x_{itf}^k \ge z_{i,t+y,f,r}^k \qquad t = E_{i0fr}^k + 1, \dots, T - E_{fr}^k, \quad y = 1, \dots, E_{fr}^k, i \in I, k \in K, r \in R, \tag{11}$$

$$x_{itf}^k + x_{i,t+y,f}^k \le 1 \qquad t = E_{i0fr}^k + 1, \dots, T - E_{fr}^k, \quad y = 1, \dots, E_{fr}^k, i \in I, k \in K, r \in R, \tag{12}$$

$$x_{itf}^k + x_{i,t+y,f}^k \le 1 \qquad t = T - E_{fr}^k + 1, \dots, T, \quad y = 1, \dots, T - t, i \in I, k \in K, r \in R, \tag{13}$$

$$m \cdot x_{itf}^k \ge z_{i,t+y,fr}^k \qquad t = T - E_{fr}^k + 1, \dots, T, \quad y = 1, \dots, T - t, i \in I, k \in K, r \in R, \tag{14}$$

$$x_{itf}^k + x_{i,t+y,f}^k \le 1 \qquad t = T - E_{fr}^k + 1, \dots, T, \quad y = 1, \dots, T - t, i \in I, k \in K, r \in R, \tag{15}$$

$$z_{itfr}^k = 1 \qquad t = 1, \dots, E_{i0fr}^k, i \in I, \quad k \in K, r \in R, f \in F, \tag{16}$$

$$x_{itf}^k + z_{itfr}^k \le 1 \qquad t = 1, \dots, E_{i0fr}^k, i \in I, \quad k \in K, r \in R, f \in F, \tag{17}$$

$$z_{itf}^k \ge z_{itfr}^k \qquad t = 1, \dots, E_{i0fr}^k, i \in I, k \in K, r \in R, f \in F, \tag{18}$$

$$z_{itf}^k \le \sum_{r=1}^{R} z_{itfr}^k \qquad t \in T, k \in K, i \in I, f \in F, \tag{19}$$

$$z_{it}^k \ge z_{itf}^k \qquad t \in T, k \in K, i \in I, f \in F, \tag{20}$$

$$z_{it}^k \le \sum_{f=1}^{F} z_{itf}^k \qquad t \in T, k \in K, i \in I, \tag{21}$$

$$z_{it}^k + y_{it}^k \le 1 \qquad t \in T, k \in K, i \in I. \tag{22}$$

The maintenance state dynamic constraints are a set of logic constraints: When an aircraft performs its maintenance activity, it will remain in the maintenance state and cannot participate in other activities, such as flight missions. Constraints (10)–(17) model relationship between x_{itf}^k and z_{itfr}^k for different time segments. Specifically, constraints (16) and (17) enforce that for aircraft that do not complete maintenance during the previous planning horizon (the previous day), they will stay in maintenance and will not be available until $\max_r E_{i0fr}^k$. Notice that constraints (10)–(17) consist of two parts: The first part includes constraints (16) and (17), which are meaningful when there are unfinished maintenance left from the previous day, e.g., $E_{i0fr}^k > 0$. These constraints model the dynamics for the periods until the leftover maintenance is completed. The second part, constraints (10)–(15), models the maintenance state dynamics after the completion of unfinished maintenance from the previous day. Constraints (18)–(21) impose relationships among z_{itfr}^k, z_{itf}^k, z_{it}^k. For example, if, for some r, $z_{itfr}^k = 1$, then, constraint (18) will ensure $z_{itf}^k = 1$, and constraint (20) will force $z_{it}^k = 1$. That means, if some trade r is working on an aircraft, then this aircraft is in maintenance at time t. Constraint (22) states that if an aircraft is in maintenance, it is not available for any missions at that time.

Resource and Mission Conflict Constraints

$$\sum_{k=1}^{K} \sum_{i=1}^{A_k} z_{it}^k \le S \qquad t \in T, \tag{23}$$

$$\sum_{k=1}^{K} \sum_{i=1}^{A_k} \sum_{f=1}^{F} z_{itfr}^k \cdot I_{fr}^k \le \lambda_r \qquad t \in T, r \in R, \tag{24}$$

$$y_{i,t+y}^k + y_{it}^k \le 1 \qquad t \in T, k \in K, i \in I, y = 1, \dots, D_t^k, \tag{25}$$

$$y_{it}^k + x_{i,t+y,f}^k \le 1 \qquad t \in T, k \in K, i \in I, y = 1, \dots, D_t^k. \tag{26}$$

Constraints (23) and (24) are resource constraints: Constraint (23) requires that one can at most maintain S (the number of maintenance stations we have) aircraft simultaneously. Constraint (24) describes that the required number of maintenance technicians cannot exceed the number of available technicians. Constraint (25) enforces that an aircraft can only conduct one mission at a time. Constraint (26) prevents an aircraft that is in maintenance from conducting a mission.

B. Reformulation and Valid Inequality

1) Reformulation of Objective Function: Notice that the objective function (3) is highly non-linear. We convert it to a linear objective function via epigraphic reformulation, namely,

$$\max \quad \lambda \tag{27}$$

with a new constraint involving bilinear terms

$$\lambda \sum_{t=1}^{T} \sum_{k=1}^{K} \sum_{i=1}^{A_k} \sum_{f=1}^{F} c_f^k \cdot x_{itf}^k \le \sum_{t=1}^{T} \sum_{k=1}^{K} M_t^k \cdot \left(\sum_{i=1}^{A_k} \left(1 - z_{it}^k\right) \right).$$

Since bilinear programming problems are notoriously difficult to solve in practice, by introducing several auxiliary variables and a number of constraints, we further linearize the above bilinear constraint via McCormick inequalities:

$$w_{ift}^{k} \le \lambda \qquad t = 1, ..., T, k \in K, i \in I, f \in F, \tag{28}$$

$$w_{ift}^{k} \le m \cdot x_{ift}^{k} \qquad t = 1, ..., T, k \in K, i \in I, f \in F, \tag{29}$$

$$w_{ift}^{k} \ge -m \cdot \left(1 - x_{ift}^{k}\right) + \lambda \qquad t = 1, ..., T, k \in K, i \in I, f \in F, \tag{30}$$

$$\sum_{t=1}^{T}\sum_{k=1}^{K} M_t^k \cdot \left(\sum_{i=1}^{A_k}\left(1 - z_{it}^k\right)\right) \ge \sum_{t=1}^{T}\sum_{k=1}^{K}\sum_{i=1}^{A_k}\sum_{f=1}^{F} c_f^k w_{itf}^k. \tag{31}$$

Finally, with the transformed objective function (27) and the constraints (4)–(26) and (28)–(31), we obtain a mixed-integer linear programming formulation for our CBPM scheduling problem, which can be solved efficiently by using commercial optimization solvers like CPLEX or GUROBI. Note that in reality, some of the desired missions may not be fulfilled even when all aircraft are in good condition. We therefore convert the mission requirement constraint (4) into an inequality constraint by allowing part of the missions to not be completed.

2) Valid Inequalities: Although the MIP formulation proposed in Section III-A can be directly solved by using commercial solvers such as GUROBI and CPLEX, it can be time-consuming for some parameter instances. Based on some preliminary computational experiments, we noticed that solvers may not obtain optimal integer solutions within a reasonable amount of time when mission durations are small and mission requirements are dense (always have mission during the day shift). We therefore developed two classes of valid inequalities to improve computational efficiency. In order to derive valid inequalities, we further assume that each component of each aircraft only needs to be maintained at most once in the planning horizon and that it is available for mission assignment after the maintenance. This assumption is not restrictive since we are planning over a daily horizon, and we mainly focus on maintenance types such as line maintenance or line-replaceable units (LRUs) replacement. Before we formally describe the valid inequalities for solving the proposed MIP model, we first review $0-1$ knapsack set and cover inequality from integer programming.

Given $b > 0$ and $a_j > 0$ for $j \in N := \{1, 2, \ldots, n\}$, the $0 - 1$ knapsack set is defined as

$$K := \left\{ x \in \{0, 1\}^n : \sum_{j=1}^{n} a_j x_j \le b \right\}.$$

A set $C \subseteq N$ is called a cover if $\sum_{j\in C} a_j > b$, and a cover inequality corresponding to the cover C is given by

$$\sum_{j\in C} x_j \le |C| - 1, \tag{32}$$

where $|C|$ denotes the cardinality of set C. The cover inequality (32) is a valid inequality for set K. Readers are refereed to [5] for more details about knapsack inequality.

Proposition 1. *For each component of each aircraft of each aircraft type, the inequality*

$$\sum_{t\in C} y_{it}^k \le |C| - 1 + \sum_{t=1}^{c_0-1} x_{itf}^k \tag{33}$$

is valid for original MIP formulation described by constraints (4)–(26), where C is the cover of the following knapsack constraint:

$$\sum_{t=1}^{24} D_t^k y_{it}^k \le r_{if}^k, \tag{34}$$

and c_0 denotes the smallest element of cover C.

Proof. We will consider two cases:

Case-1. $\sum_{t=1}^{c_0-1} x_{itf}^k = 0$: In this case, ith aircraft of type k aircraft will not be maintained during time $t = 1$ to time $t = c_0 - 1$. The inequality is reduced to cover inequality

$$\sum_{t\in C} y_{it}^k \le |C| - 1,$$

which is valid since we cannot let the aircraft conduct all missions at time $t \in C$ without maintaining the aircraft to increase its RUL r_{if}^k.

Case-2. $\sum_{t=1}^{c_0-1} x_{itf}^k = 1$: In this case, the aircraft is maintained at some time point between time $t = 1$ and time $t = c_0 - 1$. Then the inequality reduces to

$$\sum_{t\in C} y_{it}^k \le |C|,$$

which is redundant since, by our assumption, the aircraft can conduct any mission after the maintenance. □

Notice that we cannot identify all possible cover inequalities in general, and a separation algorithm is needed to find violated cover inequalities. In our case, however, we can directly identify all cover inequalities by enumeration since we only have 24 variables in knapsack constraint

$$\sum_{t=1}^{24} D_t^k y_{it}^k \le r_{if}^k.$$

Also, the aircraft are usually scheduled during the day shift (starts at 6:00 am and ends at 6:00 pm) [24]; we thus only need to identify all possible cover C of knapsack constraint

$$\sum_{t=13}^{24} D_t^k y_{it}^k \leq r_{if}^k.$$

This is because on the night shift, the mission duration parameter D_t^k will be 0. Therefore, for each component of each aircraft (i.e., for each i, k, and f), we only need to enumerate $2^{12} = 4096$ cover inequalities. However, when the planning horizon is longer (e.g., more than 24 h), enumerating all cover inequalities can be extremely time-consuming since the number of required inequalities is exponential in the length of the planning horizon. In our case, according to our numerical experiment, it will take less than 30 s to find all covers when we have a fleet of 12 aircraft with a planning horizon of 1 day.

The valid inequalities presented in Proposition 1 are usually not strong, hence insufficient to solve large-scale practical problems. Our next result shows a way to strengthen the valid inequalities in Proposition 1.

Proposition 2. *For each component of each aircraft of each aircraft type, the inequality*

$$\sum_{t\in C} y_{it}^k + \sum_{t\in C_1} y_{it}^k \leq |C| - 1 + (1 + |C_1|) \sum_{t=1}^{c_0-1} x_{itf}^k$$

is valid for original MIP formulation described by constraints (4)–(26), where C is the cover of the following knapsack constraint

$$\sum_{t=1}^{24} D_t^k y_{it}^k \leq r_{if}^k,$$

and c_0 denotes the smallest element of cover C. C_1 is defined as

$$C_1 := \{t' : c_1 + 1 \leq t' \leq 24, D_t^k \geq \max_{t\in C} D_t^k\},$$

where c_1 denotes the largest element of cover C.

Proof. Similar to the proof of Proposition 1, we will consider two cases:

Case-1. $\sum_{t=1}^{c_0-1} x_{itf}^k = 0$: In this case, ith aircraft of type k aircraft will not be maintained during time $t = 1$ to time $t = c_0 - 1$. The inequality is reduced to cover inequality

$$\sum_{t\in C} y_{it}^k + \sum_{t\in C_1} y_{it}^k \leq |C| - 1,$$

which is an extended cover inequality of cover inequality

$$\sum_{t\in C} y_{it}^k \leq |C| - 1,$$

hence valid for original formulations.

Case-2. $\sum_{t=1}^{c_0-1} x_{itf}^k = 1$: In this case, the aircraft is maintained at some time point between time $t = 1$ and time $t = c_0 - 1$. Then the inequality reduces to

$$\sum_{t\in C} y_{it}^k + \sum_{t\in C_1} y_{it}^k \leq |C| + |C_1|,$$

which is redundant, hence valid for original formulations. □

V. NUMERICAL EXPERIMENT

To validate the proposed model and solution method, we conduct a series of numerical experiments to illustrate the model performance and computational efficiency. In Section V-A, we explain the experimental designs and parameter settings for the computational tests. Section V-B shows the performance of the proposed predictive maintenance model compared to the traditional preventive maintenance model. Section V-C evaluates the impact of two classes of valid inequalities (introduced in Section IV-B) on computational efficiency. The experiments are conducted on a computer with an Intel Core i7 4-core 2.7 GHz processor and 16 GB of RAM. All mathematical programs are coded in Python 3 language, and all problem instances are solved by GUROBI 9.1 under Python API.

A. Experimental Design and Parameters Setting

Before we provide a formal presentation and interpretation of our numerical results, we will first introduce the workflow of the learn-then-optimize framework that we used to generate the results. Given a set of onboard sensor readings, we first predict RUL for different components of an aircraft by fusing the data with the bidirectional LSTM neural network model (introduced in Section III). The values of the initial estimated RUL in Table V are obtained based on these predictions. The predicted RUL is then used as input parameters for the optimization model (proposed in Section IV). Together with other parameters such as mission and maintenance requirements, we run our optimization model and finally obtain the scheduling decisions.

The parameter settings for our experiment are given as follows: Planning horizon $T = 24$ h (since we are planning daily operations), types of aircraft $K = 2$, with 6 aircraft for each type, i.e., a total of 12 aircraft. For each type of aircraft, there are three different failure modes to be considered, e.g., WP, AF, and AV. Each failure mode requires several technicians to repair it. The number of maintenance stations is set to be three, so no more than three aircraft can be maintained at the same time. The number of available technicians for each trade (i.e., trades 1, 2, and 3) at the beginning of the planning horizon is set to be 10.

For the experiment scenario, Table III summarizes the mission type (i.e., the number and type of aircraft M_t^k and mission during time D_t^k) for a 24-h horizon. In this example, all mission types in the first 12 h (6:00 pm to 6:00 am) are the same, during which there are no

Table III
Mission Requirements for Number and Type of Aircraft M_t^k and During Time D_t^k

t	1–12	13	14	15	16	17	18	19	20	21	22	23	24
M_t^1	0	1	1	1	2	2	2	1	1	1	2	2	2
M_t^2	0	1	1	1	2	2	2	1	1	1	2	2	2
D_t^1	0	1	2	1	3	1	2	1	2	1	1	2	1
D_t^2	0	2	1	2	1	2	2	2	1	2	2	2	1

missions to be conducted. The mission types can be, for example, (1) surveillance or (2) escort for various time periods. Different mission types require different numbers of aircraft and duration.

Table IV shows the maintenance time and number of technicians of trade r required to rectify failure mode f for aircraft type k. To initiate the experiment, we specify the initial health conditions, which are represented by the estimated RUL from deep learning models, for all aircraft. Table V displays the estimated RUL (in number of operational cycles) for component f of type k aircraft i at the beginning of the planning horizon. To take into account the maintenance time, Table VI specifies the required maintenance time for component f of type k aircraft at time t. In this example, every type of mission consumes five units of time (i.e., five operational cycles) for each component of each type of aircraft. Table VI displays the maintenance cost of repairing component/failure mode f of aircraft type k and the restored RUL after completing the maintenance/repair.

B. Assessment of Model Performance

We next evaluate the scheduling performance of our proposed predictive maintenance model, compared to the traditional preventive maintenance model as a benchmark. To validate the model's performance, we conduct the tests over a 72-h horizon by assuming that the mission requirements are the same for these three days. We also assume that the RUL inputs for the second and third days are calculated based on aircraft usage and maintenance during the previous day. For example, without any maintenance, the initial RUL for the second day is the initial RUL for the first day minus the total mission duration of the first day.

Table IV
Maintenance Duration E_{rf}^k and Number of Technicians I_{rf}^k of Trade r Required to Rectify Failure Mode f for Aircraft Type k

	$f=1$			$f=2$			$f=3$		
I_{rf}^k	$r=1$	$r=2$	$r=3$	$r=1$	$r=2$	$r=3$	$r=1$	$r=2$	$r=3$
$k=1$	1	1	0	1	0	1	0	1	1
$k=2$	1	2	1	1	0	1	0	1	1
	$f=1$			$f=2$			$f=3$		
E_{rf}^k	$r=1$	$r=2$	$r=3$	$r=1$	$r=2$	$r=3$	$r=1$	$r=2$	$r=3$
$k=1$	2	2	2	2	2	2	2	2	2
$k=2$	2	2	2	2	2	2	2	2	2

Table V
Initial Estimated RUL r_{if}^k of Component f for i^{th} Aircraft of Type k

	$i=1$			$i=2$			$i=3$		
r_{if}^k	$f=1$	$f=2$	$f=3$	$f=1$	$f=2$	$f=3$	$f=1$	$f=2$	$f=3$
$k=1$	12	12	25	12	12	25	13	25	7
$k=2$	12	25	25	22	20	3	25	25	5
	$i=4$			$i=5$			$i=6$		
r_{if}^k	$f=1$	$f=2$	$f=3$	$f=1$	$f=2$	$f=3$	$f=1$	$f=2$	$f=3$
$k=1$	4	8	15	3	25	20	3	10	12
$k=2$	4	5	25	3	25	20	3	25	25

Figure 6 shows the resulting optimal maintenance schedule and flight planning decisions obtained based on the mission requirements in the example scenario over a three-day planning period. In the figure, the red and green bar represent maintenance activity and mission assignment activity, respectively. The length of the bars stands for the maintenance/mission duration. The locations for maintenance activity (red bar) indicate which component is maintained. For example, for the fifth type 2 aircraft, it will maintain the first component (bottom of row 5-2) on the second day, while for the third type 1 aircraft, it will maintain the third component (top of row 3-1) on the third day. The numbers under rows MissionReq-1 and MissionReq-2 represent the number of required aircraft for each type of aircraft at a specific time slot. We can observe that all mission requirements are satisfied at all times, and there is no time conflict between maintenance and flight decisions. For example, at $t = 16$, the commander requires two type 1 and two type 2 aircraft to conduct a mission. We can see that the optimal scheduler assigns numbers 5 and 6 type 1 aircraft and numbers 5 and 6 type 2 aircraft to conduct the mission. We can interpret the decisions in terms of aircraft number 5 of type 1 as an example. It is assigned to conduct missions at time $t = 16$, and 24 in the first day. Then it is scheduled to maintain/repair component 1 at $t = 8$ in the second day, which takes 2 h to complete. After completing the repair, this aircraft is available again and can be assigned to conduct new missions. As shown in the figure, the aircraft is assigned to conduct missions at time $t = 14, 16, 19, 20, 22,$ and 23.

Table VI
Maintenance Cost c_f^k for Component f of Type k Aircraft and Restored RUL $r_{\max,f}^k$ for Component f of Type k.

c_f^k	$f=1$	$f=2$	$f=3$
$k=1$	100	90	85
$k=2$	80	120	110
$r_{\max,f}^k$	$f=1$	$f=2$	$f=3$
$k=1$	50	70	90
$k=2$	50	80	100

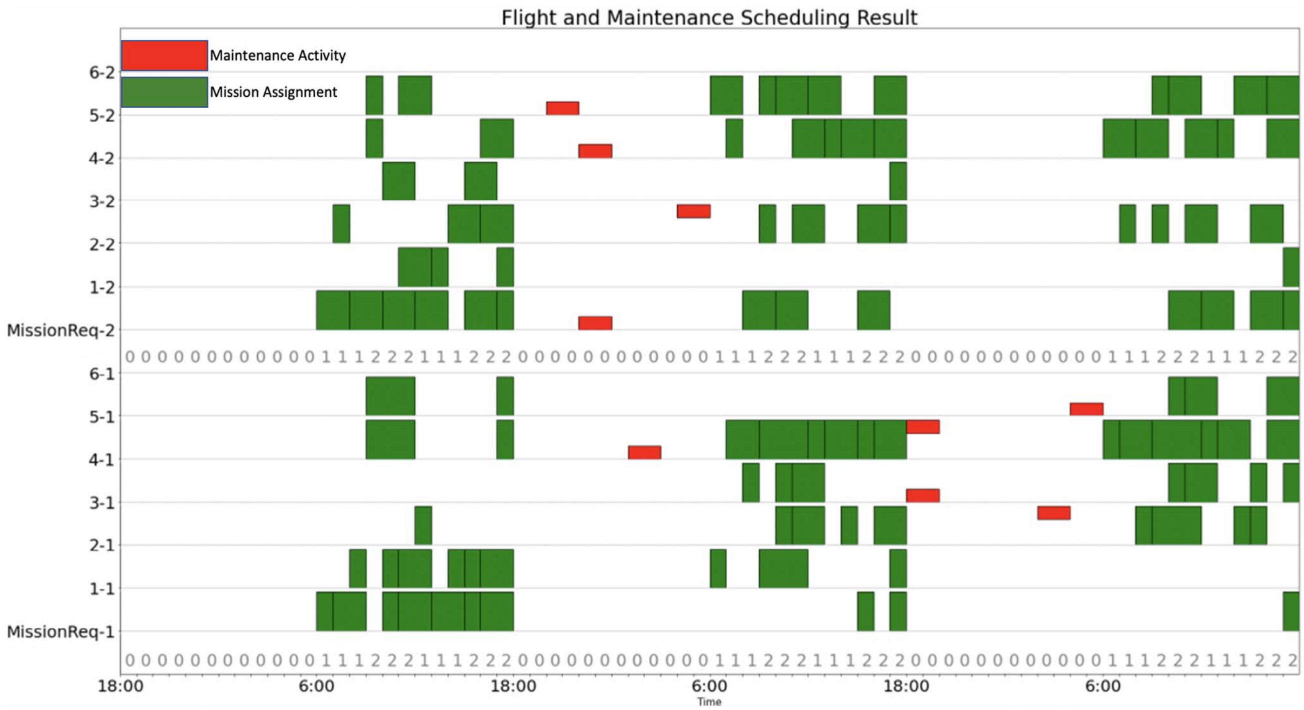


Figure 6. Flight and maintenance scheduling results.

To further demonstrate the benefits of the proposed methodology, we compare the performance of our predictive maintenance approach with that of traditional preventive maintenance methods. In preventive maintenance, an aircraft is maintained after a fixed number of operation cycles. Predictive maintenance, on the other hand, schedules the maintenance only when needed based on the predicted RUL, as described previously. In addition, in practical operations, some random failures may occur even when an aircraft is in good condition. We thus use a Bernoulli random variable, whose success probability is proportional to the cumulative usage of a component, to model these random failures. A success of the Bernoulli random variable indicates the failure of a component in our experiments. The Bernoulli random variable is realized during the preflight and postflight checks. Thus, an aircraft cannot conduct missions if a random failure occurs (or is detected) during the preflight check, and the mission is tagged as incomplete. If a random failure occurs, the aircraft needs to undergo unscheduled maintenance when maintenance resources are available; otherwise, the aircraft will be put into a waiting queue and will be unavailable to conduct any missions.

We compare the performance of predictive and preventive maintenance models over a 240-day horizon in terms of mission accomplishment rate and aircraft availability rate. The mission accomplishment rate is defined as the percentage of desired missions that are completed on time, while the aircraft availability rate is defined as the percentage of aircraft that are ready and available for mission assignments. We also evaluate the impact of the number of maintenance stations (S) and the duration of maintenance time (E^k_{fr}) on the scheduling decisions and model performance. Please note that the testing scenario involved maintenance durations ranging from 6 to 15, which differs from the specific parameters used to generate Fig. 6. This intentional difference was introduced to assess the robustness of the scheduling decisions when confronted with varying maintenance durations.

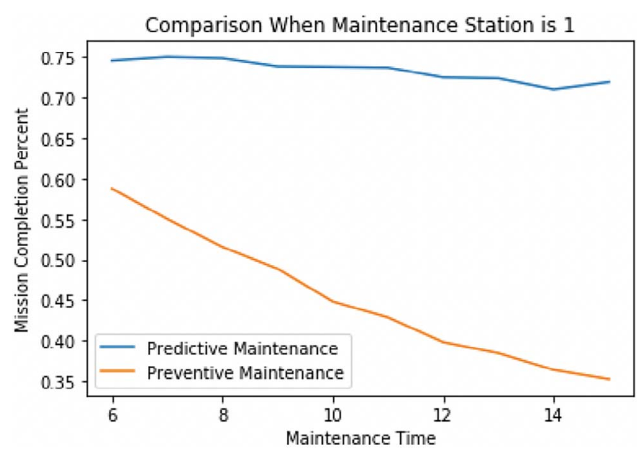


Figure 7. Mission completion rate with $S = 1$.

With one maintenance station ($S = 1$), Figs. 7 and 8 show the comparisons between predictive maintenance and preventive maintenance models across different maintenance durations in terms of mission accomplishment rate and aircraft availability rate, respectively. Figs. 9 and 10 illustrate the comparisons with two available maintenance stations ($S = 2$). Two important insights can be derived from the figures. First, the proposed predictive maintenance model dominates the

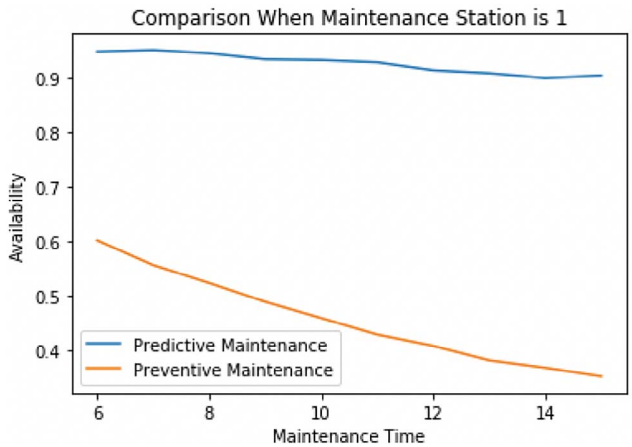


Figure 8. Aircraft availability with $S = 1$.

classical preventive maintenance model by obtaining significantly higher values in both mission accomplishment rate and aircraft availability rate. For example, when $S = 1$, the mission accomplishment rate of the predictive model ranges from 70% to 75%, while the highest value of the preventive model is only 59% and decreases dramatically as maintenance time increases. Similarly, the predictive model leads to an aircraft availability rate consistently over 90%, while the preventive model can only maintain 60% when maintenance time is short. The availability rate of the preventive model drops rapidly when maintenance duration increases. This observation leads to the second insight that the predictive maintenance model performs consistently well across different maintenance times, while the preventive model is very sensitive to required maintenance time and its performance drops significantly when maintenance time increases.

C. Impact of Valid Inequality on Computational Efficiency

In our proposed framework and numerical experiments, the scheduling optimization problem is supposed

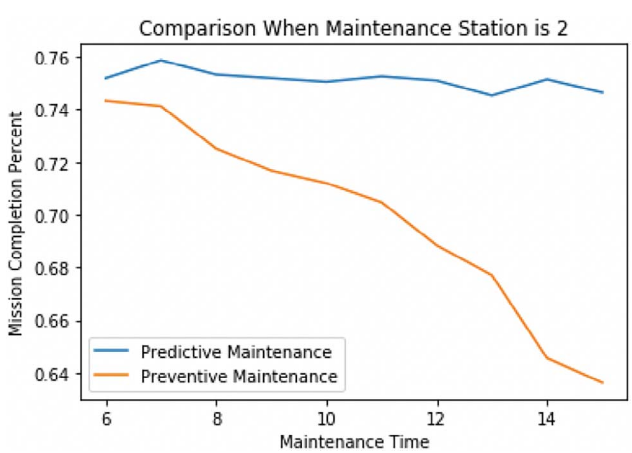


Figure 9. Mission completion rate with $S = 2$.

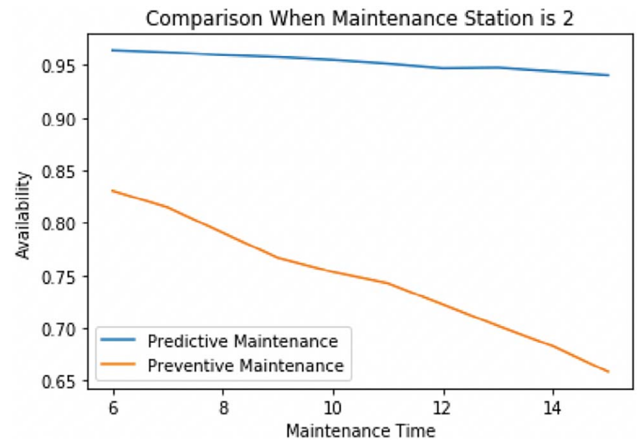


Figure 10. Aircraft availability with $S = 2$.

to be implemented every 24 h. Computational efficiency thus plays an important role in the effective implementation of real-world applications. In this section, we will demonstrate the computational efficiency of the proposed solution approaches and the impact of two classes of valid inequalities.

We adopt one problem instance using the same parameter settings specified in Section V-A. We also randomly generated four other problem instances to demonstrate the performance of the proposed valid inequalities, in which the parameters M_t^k and D_t^k are generated according to discrete uniform distributions $U(1, 3)$ and $U(1, 3)$, respectively, to emulate small mission durations and dense mission requirements. For each problem instance, we solve it five times, aiming to avoid the possibility that the improvement is due to some random procedure within the GUROBI optimizer. We set the running time limit to 10 h (36,000 CPU seconds). We report the average solution time over the five implementations in Table VII. The first column in Table VII stands for the problem instance index. The second and third columns represent the solution time without any valid inequalities and solution time with valid inequalities proposed in Propositions 1 and 2. The results clearly indicate that the proposed valid inequalities can improve the computational speed by several orders of magnitude. For cases 1 and 5, the original formulation cannot solve the problem to exactness within 10 h. However, by incorporating the valid inequalities, the average solution time reduces to 30.0 and 39.0 s, leading to an efficiency improvement of several orders of magnitude.

Table VII
Solution Time of Formulation Without and With Valid Inequalities

Instance	Original formulation	With valid inequalities
1	36,000	30.0
2	282.2	17.0
3	850.8	16.8
4	36,000	426.4
5	36,000	39.0

For case 4, the solution time with valid inequalities is 426.4 s, which is still approximately 80 times faster than the original formulation. For cases 2 and 3, which can be solved quickly by the original formulation within 282.2 and 850.8 s, the valid inequalities can still improve the computational speed significantly and obtain the solutions within 17.0 and 16.8 s. In sum, the two classes of valid inequalities proposed can efficiently improve the computational speed and make the proposed predictive maintenance models more easily applied in practical settings in a timely manner.

VI. CONCLUSION

In this study, we propose an integrated learn-then-optimize framework for CBPM scheduling and flight mission planning. The bidirectional LSTM deep learning techniques are used to combine data from multiple sensors to predict the RUL values of a multicomponent aircraft. With the predicted RUL values, a MIP formulation is then proposed to maximize the fleet availability rate subject to different mission requirements and trade types. Two classes of valid inequalities are proposed to further improve the computational efficiency of the model. The proposed predictive maintenance method significantly outperforms the traditional preventive maintenance method in a hypothetical but somewhat realistic scenario.

REFERENCES

[1] S. A. Aye and P. Heyns "An integrated Gaussian process regression for prediction of remaining useful life of slow speed bearings based on acoustic emission," *Mech. Syst. Signal Process*, vol. 84, pp. 485–498, 2017.

[2] T. P. Carvalho, F. A. Soares, R. Vita, R. d. P. Francisco, J. P. Basto, and S. G. Alcalá "A systematic literature review of machine learning methods applied to predictive maintenance," *Comput. Ind. Eng.*, vol. 137, 2019, Art. no. pp. 106024.

[3] P. Cho "Optimal scheduling of fighter aircraft maintenance." M.S. thesis, Sloan School of Management, Massachusetts Institute of Technology, Cambridge, MA, USA, 2011.

[4] Z. M. Çınar, A. A. Nuhu, Q. Zeeshan, O. Korhan, M. Asmael, and B. Safaei "Machine learning in predictive maintenance towards sustainable smart manufacturing in industry 4.0." *Sustainability*, vol. 12, no. 19, 2020, Art. no. 8211.

[5] M. Conforti, G. Cornuéjols, and G. Zambelli *Integer Programming*, vol. 271. Berlin, Germany: Springer, 2014.

[6] I. de Pater and M. Mitici "Predictive maintenance for multi-component systems of repairables with remaining-useful-life prognostics and a limited stock of spare components," *Rel. Eng. Syst. Saf.*, vol. 214, 2021, Art. no. 107761.

[7] I. de Pater, A. Reijns, and M. Mitici "Alarm-based predictive maintenance scheduling for aircraft engines with imperfect remaining useful life prognostics," *Rel. Eng. Syst. Saf.*, vol. 221, 2022, Art. no. 108341.

[8] Q. Feng, X. Bi, X. Zhao, Y. Chen, and B. Sun "Heuristic hybrid game approach for fleet condition-based maintenace planning," *Rel. Eng. Syst. Saf.*, vol. 157, pp. 166–176, 2017.

[9] A. Gavranis and G. Kozanidis "An exact solution algorithm for maximizing the fleet availability of a unit of aircraft subject to flight and maintenance requirements," *Eur. J. Oper. Res.*, vol. 242, pp. 631–643, 2015.

[10] A. Gavranis and G. Kozanidis "Mixed integer bi-objective quadratic programming for maximum-value minimum-variability fleet availability of a unit of mission aircraft," *Comput. Ind. Eng.*, vol. 10, pp. 13–29, 2017.

[11] F. Gers, J. Schmidhuber, and F. Cummins "Learning to forget: Continual prediction with LSTM," *Neural Comput.*, vol. 12, pp. 2451–2471, 2000.

[12] L. Guo, N. Li, F. Jia, Y. Lei, and J. Lin "A recurrent neural network based health indicator for remaining useful life prediction of bearings," *Neurocomputing*, vol. 240, pp. 98–109, 2017.

[13] S. Hochreiter and J. Schmidhuber "Long short-term memory," *Neural Comput.*, vol. 9, pp. 1735–1780, 1997.

[14] G. Kozanidis "A multiobjective model for maximizing fleet availability under the presence of filight and maintenance requirement," *J. Adv. Transp.*, vol. 43, pp. 155–182, 2009.

[15] G. Kozanidis, A. Gavranis, and G. Liberopoulos "Heurisitics for flight and maintenance planning of mission aircraft," *Ann. Operations Res.*, vol. 221, pp. 211–238, 2014.

[16] G. Kozanidis, G. Liberopoulos, and C. Pitsilkas "Flight and maintenance planning of military aircraft for maximum fleet availability," *Mil. Operations Res.*, vol. 15, pp. 53–73, 2010.

[17] P. Kundu, A. K. Darpe, and M. S. Kulkarni "An ensemble decision tree methodology for remaining useful life prediction of spur gears under natural pitting progression," *Struct. Health Monit.*, vol. 19, no. 3, pp. 854–872, 2020.

[18] J. Lee and M. Mitici "Multi-objective design of aircraft maintenance using Gaussian process learning and adaptive sampling," *Rel. Eng. Syst. Saf.*, vol. 218, 2022, Art. no. 108123.

[19] Z. Li, J. Guo, and R. Zhou "Maintenance scheduling optimization based on reliability and prognostics information," *Annu. Rel. Maintainability Symp.*, 2016, pp. 15.

[20] K. T. Nguyen and K. Medjaher "A new dynamic predictive maintenance framework using deep learning for failure prognostics," *Rel. Eng. Syst. Saf.*, vol. 188, pp. 251–262, 2019.

[21] F. Peschiera, O. Battaia, A. Hait, and N. Dupin "Long term planning of military aircraft flight and maintenance operations," 2020, arXiv: 2001.09856.

[22] F. Peschiera, R. Dell, J. Royset, A. Hait, N. Dupin, and O. Battaia "A novel solution approach with ML-based pseudo-cuts for the flight and maintenance planning problem," *OR Spectr.*, vol. 43, 2021, pp. 635–664.

[23] R. R. Richardson, M. A. Osborne, and D. A. Howey "Gaussian process regression for forecasting battery state of health," *J. Power Sources*, vol. 357, pp. 209–219, 2017.

[24] N. Safaei, D. Nanjevic, and A. K. Jardine "Workforce-constrained maintenance scheduling for military aircraft fleet: A case study," *Ann. Operations Res.*, vol. 186, pp. 295–316, 2011.

[25] A. Saxena, K. Goebel, D. Simon, and N. Eklund
"Damage propagation modeling for aircraft engine run-to-failure simulation,"
Proc. IEEE Int. Conf. Prognostics Health Mangement, 2008, pp. 1–9.

[26] M. Schuster and K. Paliwal
"Bidirectional recurrent neural networks,"
IEEE Trans. Signal Proess., vol. 45, no. 11, pp. 2674–2681, Nov. 1997.

[27] O. Serradilla, E. Zugasti, J. Rodriguez, and U. Zurutuza
"Deep learning models for predictive maintenance: A survey, comparison, challenges and prospects," Appl. Intell., vol. 52, pp. 1093410964, 2022

[28] C. Shen, D. Wang, F. Kong, and W. T. Peter
"Fault diagnosis of rotating machinery based on the statistical parameters of wavelet packet paving and a generic support vector regressive classifier,"
Measurement, vol. 46, no. 4, pp. 1551–1564, 2013.

[29] S. Varsamopoulos, K. Bertels, and C. G. Almudever
"Comparing neural network based decoders for the surface code,"
IEEE Transactions on Computers, vol. 69, no. 2, pp. 300–311, 2019.

[30] H. Wang, X. Ma, and Y. Zhao
"An improved Wiener process model with adaptive drift and diffusion for online remaining useful life prediction,"
Mech. Syst. Signal Process, vol. 127, pp. 370–387, 2019.

[31] X. Wang
"Wiener processes with random effects for degradation data,"
J. Multivariate Anal., vol. 101, no. 2, pp. 340–351, 2010.

[32] Y. Wang, S. Limmer, D. Van Nguyen, M. Olhofer, T. Bäck, and M. Emmerich
"Optimizing the maintenance schedule for a vehicle fleet: A simulation-based case study,"
Eng. Optim., vol. 54, no. 7, pp. 1258–1271, 2022.

[33] Y. Wen, M. F. Rahman, H. Xu, and T.-L. B. Tseng
"Recent advances and trends of predictive maintenance from data-driven machine prognostics perspective,"
Measurement, vol. 187, 2022, Art. no. 110276.

[34] B. Yang, R. Liu, and E. Zio
"Remaining useful life prediction based on a double-convolutional neural network architecture,"
IEEE Trans. Ind. Electron., vol. 66, no. 12, 9521–9530, Dec. 2019.

[35] Z.-S. Ye, Y. Shen, and M. Xie
"Degradation-based burn-in with preventive maintenance,"
Eur. J. Oper. Res., vol. 221, no. 2, 360–367, 2012.

[36] J. Zhang, P. Wang, R. Yan, and R. X. Gao
"Deep learning for improved system remaining life prediction,"
Procedia CIRP, vol. 72, pp. 1033–1038, 2018.

[37] J. Zhang, P. Wang, R. Yan, and R. X. Gao
"Long short-term memory for machine remaining life prediction,"
J. Manuf. Syst., vol. 48, pp. 78–86, 2018.

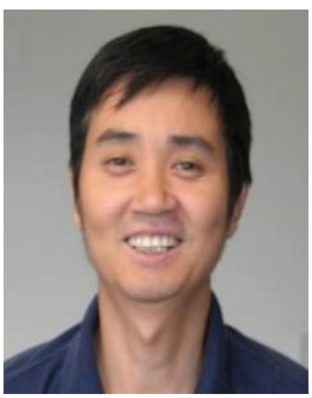
[38] Y. Zhang, R. Xiong, H. He, and M. G. Pecht
"Long short-term memory recurrent neural network for remaining useful life prediction of lithium-ion batteries,"
IEEE Trans. Veh. Technol., vol. 67, no. 7, pp. 5695–5705, Jul. 2018.

[39] T. Zonta, C. André da Costa, R. da Rosa Righi, M. José de Lima, E. Silveira da Trindade, and G. P. Li
"Predictive maintenance in the industry 4.0: A systematic literature review,"
Comput. Ind. Eng., vol. 150, 2020, Art. no. 106889.

Zhengyang Fan received the M.S. degree in statistical science from George Mason University, Fairfax, VA, USA, in 2017 and he is currently a Ph.D. candidate in the Department of Systems Engineering and Operations, at George Mason University, Fairfax, VA, USA. His current research interests focus on explainable AI, stochastic programming, and distributionally robust optimization.

Kuo-Chu Chang received the M.S. and Ph.D. degrees in electrical engineering from the University of Connecticut, Storrs, CT, USA, in 1983 and 1986, respectively. He is currently a Professor with the Department of Systems Engineering and Operations Research, George Mason University, Fairfax, VA, USA. His research interests include estimation theory, data fusion, Bayesian inference, machine learning, and financial engineering. He has more than 35 years of industrial and academic experience and has published more than 300 papers in the areas of multitarget tracking, distributed sensor fusion, Bayesian networks technologies, and quantitative finance. As an IEEE Fellow, his research has been strongly supported by ONR, AFOSR, NSF, DARPA, and ARO.

Ran Ji is an Assistant Professor with the Department of System Engineering and Operations Research, George Mason University, Fairfax, VA, USA. His research focuses on stochastic optimization, distributionally robust optimization, chance-constrained programming and uncertainty quantification, and data-driven decision-making. Specifically, some of his research goals include tackling optimization problems under uncertainty via mathematical programming. His application areas of interest include financial risk modeling, portfolio optimization, supply chain risk analytics, inventory and production planning, statistical learning, and data mining.



Genshe Chen received the B.S. and M.S. degrees in electrical engineering and the Ph.D. degree in aerospace engineering, in 1989, 1991, and 1994, respectively, all from Northwestern Polytechnical University, Xian, China. He is currently the Chief Technological Officer (CTO) of Intelligent Fusion Technology, Inc., Germantown, MD, USA, where he directs research and development activities for the Government Services and Commercial Solutions group. He was the CTO of DCM Research Resources LLC, Germantown, MD, USA, and the program manager for networks, systems, and control at Intelligent Automation, Inc. His research interests include space resiliency, space situational awareness, cognitive radio for satellite communications, cooperative control and optimization for military operations, multi-INT data fusion, guidance, navigation, and control, electronic warfare, game theory, and human-cyber-physical systems.

INTERNATIONAL SOCIETY OF INFORMATION FUSION

ISIF Website: http://www.isif.org

SOCIETY VISION

The International Society of Information Fusion (ISIF) is the premier professional society and global information resource for multidisciplinary approaches for theoretical and applied information fusion technologies.

SOCIETY MISSION

Advocate

To advance the profession of fusion technologies, propose approaches for solving real-world problems, recognize emerging technologies, and foster the transfer of information.

Serve

To serve its members and engineering, business, and scientific communities by providing high-quality information, educational products, and services.

Communicate

To create international communication forums and hold international conferences in countries that provide for interaction of members of fusion communities with each other, with those in other disciplines, and with those in industry and academia.

Educate

To promote undergraduate and graduate education related to information fusion technologies at universities around the world. Sponsor educational courses and tutorials at conferences.

Integrate

Integrate ideas from various approaches for information fusion, and look for common threads and themes– look for overall principles, rather than a multitude of point solutions. Serve as the central focus for coordinating the activities of world-wide information fusion related societies or organizations. Serve as a professional liaison to industry, academia, and government.

Disseminate

To propagate the ideas for integrated approaches to information fusion so that others can build on them in both industry and academia.

Call for Papers

The Journal of Advances in Information Fusion (JAIF) seeks original contributions in the technical areas of research related to information fusion. Authors are encouraged to submit their manuscripts for peer review http://isif.org/journal.

Call for Reviewers

The success of JAIF and its value to the research community is strongly dependent on the quality of its peer review process. Researchers in the technical areas related to information fusion are encouraged to register as a reviewer for JAIF at http://jaif.msubmit.net. Potential reviewers should notify via email the appropriate editors of their offer to serve as a reviewer.

Journal of Advances in Information Fusion Volume 18 Number 2 December 2023 ISSN 1557-6418