Online Learning of Model Parameters and Object Classes in Extended Multiobject Tracking

Thomas John Bucco*, Günther Koliander†, Bernd Kreidl‡, and Franz Hlawatsch*

*Institute of Telecommunications, TU Wien, Vienna, Austria ({thomas.bucco, franz.hlawatsch}@tuwien.ac.at)
†Acoustics Research Institute, Austrian Academy of Sciences, Vienna, Austria (guenther.koliander@oeaw.ac.at)
‡Optronia GmbH, Innsbruck, Austria (berndkr@protonmail.com)

***Abstract*—Most multiobject tracking methods rely on a statistical model that involves unknown parameters. Here, we propose a Bayesian method for *class-aided online learning* of model parameters within extended multiobject tracking. We address the case where the extended objects belong to unknown object classes defined by unknown values of the model parameters. The proposed method learns the number of object classes, the class parameters, and the objects' class affiliations simultaneously with the tracking process, and the learned class and parameter information is leveraged for improved tracking. This is enabled by a parameter-dependent state-space model for extended multiobject tracking that incorporates a Dirichlet process prior, and by a related Gibbs sampler for online learning. Our simulation results demonstrate substantial gains in tracking performance due to class-aided online parameter learning.**

***Index Terms*—Extended multiobject tracking, extended multitarget tracking, tracking and classification, Dirichlet process, Bayesian nonparametrics, Gibbs sampler, clustering.**

I. Introduction

Extended multiobject tracking (EMT) aims at estimating the time-varying states of multiple extended objects—i.e., objects with a spatial extent—from sensor measurements [1]. EMT arises in many applications including autonomous driving [2], [3], biomedical analysis [4], and surveillance of maritime transport [5], underwater environments [6], and pollutants [7]. Most EMT methods require the specification of a statistical model. Usually, at least some of the model parameters are unknown and are chosen manually. Alternatively, they can be learned online during object state estimation; this approach has been used for EMT in [3], [8] and for classical multiobject tracking (i.e., considering point objects) in [9]–[11].

Here, we propose a new framework for *class-aided online parameter learning* within EMT. The underlying assumption is that the objects belong to object classes defined by specific values of the parameters. Practical examples of object classes include cars, bicycles, and pedestrians in a traffic scenario, or different types of cells in a biomedical scenario. The class structure aids parameter learning in that the parameters are estimated *jointly* for all objects belonging to the same class from all measurements generated by these objects. As we will demonstrate, a performance gain can be achieved even when the objects' class affiliations and the values of the class parameters are unknown.

Object classes have been previously considered within multiobject tracking. In particular, under the assumption that the number of object classes and the values of the class parameters are known, the classification of objects is addressed in the context of EMT in [12] and [13] and in the context of classical multiobject tracking in [14], [15]. In [16] and [17], object classification is considered in the EMT context under the assumption that the number of object classes is known and the motion model of each object class is known up to a random noise component with an unknown distribution that is identical for each class. In [18], the clustering of point objects into classes is considered; the number of object classes and the class parameter values are allowed to be unknown except for a known maximum number of classes, and the class parameters are estimated using a maximum likelihood approach.

In our method, the number of object classes, the class parameter values, and the objects' class affiliations are allowed to be unknown and are learned online, i.e., simultaneously with object state estimation. This online learning does not require any training data. Even during EMT operation, additional object classes are automatically created if new measurements do not conform to the currently existing classes. These desirable properties of our method are enabled by the use of a novel EMT version of the parameter-dependent state-space model presented in [19]. This model involves class parameters that are distributed according to a *Dirichlet process* (DP) prior [20, Ch. 4], which allows the number of object classes and the class parameter values to be unknown. The use of a DP in a state-space model for the purpose of clustering has been discussed, e.g., in [19], [21]–[25].

For parameter and class learning, we use a Bayesian inference framework and a Gibbs sampler [26], [27]. This learning process is combined with a conventional EMT method that is based on the architecture of [28] and uses an extended Kalman filter (EKF) for sequential state estimation [29] and heuristic techniques [28] to account for unknown object existence and measurement origin uncertainty. We will demonstrate experimentally that the proposed parameter learning technique leads to improved tracking performance. However, our goal is not to outperform state-of-the-art EMT methods but to introduce DP-based parameter learning within EMT and demonstrate a gain

This work was supported in part by the Austrian Science Fund (FWF) under grant P 32055-N31.

in tracking performance. We conjecture that the performance can be improved further by a complete integration of DP-based learning and EMT within an overall Bayesian model and inference method.

This paper is organized as follows. In Section II, we describe the basic EMT system model. In Section III, we present the object class structure and the DP prior. Section IV gives an overview of the proposed method. The Gibbs sampler for DP-based class and parameter learning is developed in Section V. Section VI describes the EMT part of the proposed method. Simulation results are presented in Section VII.

Notation: Random quantities are displayed in an upright (roman) sans serif font (e.g., x is a random scalar, $\mathbf{x}$ a random vector, and $\mathbf{X}$ a random matrix), and their realizations and other deterministic quantities in an italic serif font (e.g., x, $\boldsymbol{x}$, and $\boldsymbol{X}$). We define $[\boldsymbol{a}_i]_{i\in\mathcal{I}}$ to be the matrix with columns $\boldsymbol{a}_i$, $[\boldsymbol{A}_i]_{i\in\mathcal{I}}$ the matrix stacking the matrices $\boldsymbol{A}_i$ in a columnwise fashion, and $\mathrm{diag}(\cdot)$ the diagonal or block-diagonal matrix containing the arguments on its (block-)diagonal. The zero matrix and the all-ones matrix of size $l\times j$ are denoted as $\mathbf{0}_{l\times j}$ and $\mathbf{1}_{l\times j}$, respectively, and the identity matrix of size $j\times j$ as $\mathbf{I}_j$. The indicator function $\mathbb{1}(\cdot)$ is 1 if the argument is true and 0 otherwise. We use $p(\cdot)$ to denote the probability mass function (pmf) of a discrete random quantity and $f(\cdot)$ to denote the probability density function (pdf) of a continuous random quantity. Finally, $\mathcal{N}(\boldsymbol{x};\boldsymbol{\mu},\boldsymbol{\Sigma})$ denotes the pdf of a Gaussian random vector $\mathbf{x}$ with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$.

II. System Model

A. Object States and Motion Model

We consider $I\in\mathbb{N}_0$ objects within the total time interval $\{1,\ldots,K\}$. Object $i\in\{1,\ldots,I\}$ exists for time steps $k\in\mathcal{K}_i\triangleq[k_{\mathrm{S},i},k_{\mathrm{E},i}]\subseteq\{1,\ldots,K\}$ (here, subscripts S and E stand for "start" and "end," respectively). The number of objects I and the object existence intervals $\mathcal{K}_i$ are estimated by the method described in Section VI-A. The state of object i, $\mathbf{x}_{k,i}\triangleq[\mathsf{x}_{k,i,1}\ \mathsf{x}_{k,i,2}\ \dot{\mathsf{x}}_{k,i,1}\ \dot{\mathsf{x}}_{k,i,2}]^{\mathrm{T}}$, consists of the object's two-dimensional (2D) position and velocity for $k\in\mathcal{K}_i$ and is undefined for $k\notin\mathcal{K}_i$. (We note that our model and method can be easily extended to more than two dimensions.) The initial state $\mathbf{x}_{k_{\mathrm{S},i},i}$ of object i is zero-mean Gaussian with a known covariance matrix $\boldsymbol{\Sigma}_{\mathrm{S}}\in\mathbb{R}^{4\times4}$.

The object states evolve independently according to the nearly constant velocity model [30, Sec. 6.3.2], i.e.,

$$\mathbf{x}_{k,i}=\boldsymbol{F}\mathbf{x}_{k-1,i}+\mathbf{v}_{k,i},\quad k\in\mathcal{K}_i\backslash\{k_{\mathrm{S},i}\}. \tag{1}$$

Here, $\boldsymbol{F}\triangleq\begin{bmatrix}\mathbf{I}_2 & \Delta T\,\mathbf{I}_2\\ \mathbf{0}_{2\times2} & \mathbf{I}_2\end{bmatrix}$ with sampling period $\Delta T>0$, and the driving noise $\mathbf{v}_{k,i}$ has pdf $\mathcal{N}(\boldsymbol{v}_{k,i};\mathbf{0}_{4\times1},\mathbf{Q}_i)$ with a random covariance matrix

$$\mathbf{Q}_i\triangleq\mathrm{diag}(\mathsf{q}_{i,1}^2\mathbf{I}_2,\mathsf{q}_{i,2}^2\mathbf{I}_2).$$

The distribution of the driving noise variances $\mathsf{q}_{i,1}^2$ and $\mathsf{q}_{i,2}^2$ will be specified in Section III. Given $\boldsymbol{x}_{k-1,i}$ and $\boldsymbol{Q}_i$, $\mathbf{x}_{k,i}$ is conditionally independent across k and i and distributed according to

$$f(\boldsymbol{x}_{k,i}|\boldsymbol{x}_{k-1,i},\boldsymbol{Q}_i)=\mathcal{N}(\boldsymbol{x}_{k,i};\boldsymbol{F}\boldsymbol{x}_{k-1,i},\boldsymbol{Q}_i). \tag{2}$$

B. Measurement Model

Our measurement model is similar to that of [29]; however, differently from [29], we assume a Gaussian distribution of the object-generated measurements, we consider also measurements due to clutter, and we model the objects as rigid bodies whose extents are constant except for their orientation.

1) Object-generated Measurements: At each time $k\in\mathcal{K}_i$, we observe $\mathsf{m}_{k,i}\in\mathbb{N}_0$ measurements generated by object $i\in\{1,\ldots,I\}$. Here, $\mathsf{m}_{k,i}$ is Poisson distributed with a known rate $\lambda>0$, and different $\mathsf{m}_{k,i}$ are independent. For $k\notin\mathcal{K}_i$, we set $\mathsf{m}_{k,i}=0$.

The measurements generated by an existing object i, $\mathbf{z}_{k,i}^{(n)}=\big[\mathsf{z}_{k,i,1}^{(n)}\ \mathsf{z}_{k,i,2}^{(n)}\big]^{\mathrm{T}}$ with $n\in\{1,\ldots,\mathsf{m}_{k,i}\}$, are modeled as noisy versions of the object position according to

$$\mathbf{z}_{k,i}^{(n)}=\boldsymbol{H}\mathbf{x}_{k,i}+\boldsymbol{R}(\mathbf{x}_{k,i})\mathbf{u}_{k,i}^{(n)}. \tag{3}$$

Here, $\boldsymbol{H}\triangleq[\mathbf{I}_2\ \mathbf{0}_{2\times2}]$ so that $\boldsymbol{H}\mathbf{x}_{k,i}=[\mathsf{x}_{k,i,1}\ \mathsf{x}_{k,i,2}]^{\mathrm{T}}$ is the object position, and $\boldsymbol{R}(\mathbf{x}_{k,i})\mathbf{u}_{k,i}^{(n)}$ describes the measurement noise according to an elliptical model of the object's physical extent. (We note that this model is practically useful and admits a simple parameterization; however, our approach of class-aided EMT can be readily generalized to other parametric models of the object extent.) The two factors of $\boldsymbol{R}(\mathbf{x}_{k,i})\mathbf{u}_{k,i}^{(n)}$ are given as follows:

- The random noise vector $\mathbf{u}_{k,i}^{(n)}$ is zero-mean Gaussian with a random covariance matrix

  $$\mathbf{D}_i\triangleq\mathrm{diag}(\mathsf{d}_{i,1}^2,\mathsf{d}_{i,2}^2).$$

  It is furthermore conditionally independent, given $\mathbf{D}_i$, across k, n, and i. The distribution of the noise variances $\mathsf{d}_{i,1}^2$ and $\mathsf{d}_{i,2}^2$ will be specified in Section III. We note that the measurement noise standard deviations $\mathsf{d}_{i,1}>0$ and $\mathsf{d}_{i,2}>0$ are here interpreted as the lengths of the half-axes of the object extent ellipse, and thus the distribution of the measurement noise $\mathbf{u}_{k,i}^{(n)}$ matches the elliptical object extent. Accordingly, although $\mathbf{D}_i$ is involved in the measurement model (3), we will consider it a parameter of object i and call it the "object extent matrix."
- The matrix $\boldsymbol{R}(\mathbf{x}_{k,i})\triangleq\begin{bmatrix}\cos\phi(\mathbf{x}_{k,i}) & -\sin\phi(\mathbf{x}_{k,i})\\ \sin\phi(\mathbf{x}_{k,i}) & \cos\phi(\mathbf{x}_{k,i})\end{bmatrix}$ rotates the vector $\mathbf{u}_{k,i}^{(n)}$ by the angle $\phi(\mathbf{x}_{k,i})\triangleq\tan^{-1}\big(\frac{\dot{\mathsf{x}}_{k,i,2}}{\dot{\mathsf{x}}_{k,i,1}}\big)$. This causes the first half-axis of the object extent ellipse, which determines the distribution of $\mathbf{u}_{k,i}^{(n)}$ via $\mathbf{D}_i$, to be aligned with the object's current direction of movement as defined by the velocity vector $[\dot{\mathsf{x}}_{k,i,1}\ \dot{\mathsf{x}}_{k,i,2}]^{\mathrm{T}}$. Thus, the 2D random vector $\boldsymbol{R}(\mathbf{x}_{k,i})\mathbf{u}_{k,i}^{(n)}$ is oriented along the object's current direction of movement.

Because of the nonlinear mappings $\boldsymbol{R}(\mathbf{x}_{k,i})$ and $\phi(\mathbf{x}_{k,i})$, the measurement model (3) is nonlinear. Given $\boldsymbol{x}_{k,i}$ and $\boldsymbol{D}_i$, and assuming $m_{k,i}>0$, $\mathbf{z}_{k,i}^{(n)}$ is conditionally independent across

k, i, and n and distributed according to

$$f\big(\boldsymbol{z}_{k,i}^{(n)}\big|\,\boldsymbol{x}_{k,i},\boldsymbol{D}_i\big)=\mathcal{N}\big(\boldsymbol{z}_{k,i}^{(n)};\boldsymbol{H}\boldsymbol{x}_{k,i},\boldsymbol{R}(\boldsymbol{x}_{k,i})\boldsymbol{D}_i\boldsymbol{R}(\boldsymbol{x}_{k,i})^{\mathrm{T}}\big).$$

We denote by $\mathbf{Z}_{k,i}\triangleq\big[\mathbf{z}_{k,i}^{(n)}\big]_{n=1}^{\mathsf{m}_{k,i}}$ the matrix whose columns are all the measurements generated by object i. Conditioned on $m_{k,i}$, $\boldsymbol{x}_{k,i}$, and $\boldsymbol{D}_i$, $\mathbf{Z}_{k,i}$ is distributed according to

$$\begin{aligned}&f(\boldsymbol{Z}_{k,i}|\,m_{k,i},\boldsymbol{x}_{k,i},\boldsymbol{D}_i)\\&\quad=\prod_{n=1}^{m_{k,i}}\mathcal{N}\big(\boldsymbol{z}_{k,i}^{(n)};\boldsymbol{H}\boldsymbol{x}_{k,i},\boldsymbol{R}(\boldsymbol{x}_{k,i})\boldsymbol{D}_i\boldsymbol{R}(\boldsymbol{x}_{k,i})^{\mathrm{T}}\big).\end{aligned}\tag{4}$$

2) Measurements Due to Clutter: At each time k, we also observe $\mathsf{m}_{k,0}\in\mathbb{N}_0$ measurements due to clutter. Here, $\mathsf{m}_{k,0}$ is Poisson distributed with a known rate $\lambda_0>0$, and different $\mathsf{m}_{k,0}$ are independent. The measurements due to clutter, denoted $\mathbf{z}_{k,0}^{(n)}$, are independent across $n=1,\dots,\mathsf{m}_{k,0}$ and distributed according to some pdf $f_{\text{clutter}}\big(\boldsymbol{z}_{k,0}^{(n)}\big)$. We also define $\mathbf{Z}_{k,0}\triangleq\big[\mathbf{z}_{k,0}^{(n)}\big]_{n=1}^{\mathsf{m}_{k,0}}$. We note that the total number of measurements observed at time k, both object-generated and due to clutter, is given by $\mathsf{M}_k=\sum_{i=0}^{I}\mathsf{m}_{k,i}$.

3) Data Association: The number I of objects, the object existence intervals $\mathcal{K}_i$, and the origin i of each measurement $\mathbf{z}_{k,i}^{(n)}$—which is either a specific object $i\in\{1,\dots,I\}$ or clutter ($i=0$)—are unknown. This information is estimated by a separate algorithm, described in Section VI-A, that performs object existence detection and data association (DA). In this context, it will be convenient to define the "DA-aware" measurement matrix $\mathbf{Z}_{k,0:I}\triangleq[\mathbf{Z}_{k,i}]_{i=0}^{I}$ that collects all measurements at time k in the order of increasing origin index $i=0,1,\dots,I$. Since the measurement origins are *a priori* unknown, we cannot consider $\mathbf{Z}_{k,0:I}$ as the observed data. On the other hand, we denote by $\mathbf{z}_k^{(j)}$ with $j=1,\dots,\mathsf{M}_k$ the measurements in arbitrary order, without reference to the measurement origin i, and we define the "observed" measurement matrix $\mathbf{Z}_k\triangleq\big[\mathbf{z}_k^{(j)}\big]_{j=1}^{\mathsf{M}_k}$. Each column of $\mathbf{Z}_k$ is equal to a column of $\mathbf{Z}_{k,0:I}$ via a permutation $\pi_k(\cdot)\colon\mathbb{R}^{2\times\mathsf{M}_k}\to\mathbb{R}^{2\times\mathsf{M}_k}$ that reorders the columns of $\mathbf{Z}_{k,0:I}$. Thus, DA—i.e., associating each measurement $\mathbf{z}_k^{(j)}$ to an origin i—amounts to inferring $\pi_k(\cdot)$. This task will be considered in Section VI-A.

III. Object Classes and Class Parameters

Our class-aided learning approach to EMT relies on the notion that the objects belong to classes defined by certain random object parameters. We define these class-defining object parameters to consist of the object extent parameters $\mathsf{d}_{i,1}^2,\mathsf{d}_{i,2}^2$ and the driving noise variances $\mathsf{q}_{i,1}^2,\mathsf{q}_{i,2}^2$, which were introduced in Sections II-B1 and II-A, respectively. These parameters are combined into the vector

$$\mathbf{p}_i\triangleq\big[\mathsf{d}_{i,1}^2\;\;\mathsf{d}_{i,2}^2\;\;\mathsf{q}_{i,1}^2\;\;\mathsf{q}_{i,2}^2\big]^{\mathrm{T}},$$

which will be termed the "class-defining object parameter." Each object class is defined by a unique value of $\mathbf{p}_i$.

A. Dirichlet Process Prior

A probabilistic class structure of the objects can be obtained by using a Dirichlet process (DP) prior [20, Ch. 4] for the class-defining object parameters $\mathbf{p}_i$. The DP prior can be defined by a procedure for randomly generating parameters $\mathbf{p}_i$ for an infinite sequence of objects $i=1,2,\dots$ [31]. The generated $\mathbf{p}_i$ are dependent because *subsets of them can be equal*. These subsets constitute the object classes in the sense that all objects i with equal $\mathbf{p}_i$ belong to the same class.

The generation procedure consists of the following three steps. First, we generate an infinite number of random *tentative class parameters*

$$\mathbf{p}_c^*\triangleq\big[\mathsf{d}_{c,1}^{*2}\;\;\mathsf{d}_{c,2}^{*2}\;\;\mathsf{q}_{c,1}^{*2}\;\;\mathsf{q}_{c,2}^{*2}\big]^{\mathrm{T}},$$

which are indexed by the *class index* $c\in\mathbb{N}$. Here, $\mathbf{p}_c^*$ is the object parameter within class c. We call the $\mathbf{p}_c^*$ *tentative* class parameters because at this point it is not yet clear if class c is active in the sense that any objects i belong to it, i.e., any actual object parameters $\mathbf{p}_i$ equal $\mathbf{p}_c^*$. By definition, the $\mathbf{p}_c^*$ are independent across c and identically distributed according to a *base pdf* f_{G}, i.e.,

$$\mathbf{p}_1^*,\mathbf{p}_2^*,\dots\overset{\text{iid}}{\sim}f_{\mathrm{G}}(\boldsymbol{p}_c^*).\tag{5}$$

The base pdf will be specified in Section III-C.

The second step, which is parallel to the first step, is to generate an infinite sequence of random *weights* $\mathsf{B}_c\in(0,1)$, $c\in\mathbb{N}$. Here, B_c is interpreted as the (random) prior probability that a given object i belongs to class c in that $\mathbf{p}_i=\mathbf{p}_c^*$, which implies that $\sum_{c=1}^{\infty}\mathsf{B}_c=1$ (almost surely). This property can be ensured by a recursive generation of the B_c that corresponds to successively breaking off parts of lengths B_c from a stick of length 1 and is thus known as the stick-breaking process [31]. The stick-breaking process involves a "concentration" parameter α. The probability distribution of the infinite random sequence of pairs $(\mathbf{p}_c^*,\mathsf{B}_c)_{c\in\mathbb{N}}$ is said to be a *DP with base pdf* f_{G} *and concentration* α.

At this point, we generated a realization $(\boldsymbol{p}_c^*,B_c)_{c\in\mathbb{N}}$ of the random sequence $(\mathbf{p}_c^*,\mathsf{B}_c)_{c\in\mathbb{N}}$. The final step is to generate the sequence of object parameters $\mathbf{p}_i$, $i=1,2,\dots$ by drawing each $\mathbf{p}_i$ conditionally independently, given $(\boldsymbol{p}_c^*,B_c)_{c\in\mathbb{N}}$, from the set of tentative class parameters $\{\boldsymbol{p}_c^*\}_{c\in\mathbb{N}}$ such that

$$\mathbf{p}_i=\boldsymbol{p}_c^*\;\;\text{with probability }B_c.\tag{6}$$

Note that the $\mathbf{p}_i$ are conditionally independent given $(\boldsymbol{p}_c^*,B_c)_{c\in\mathbb{N}}$ but not unconditionally independent.

By combining the DP prior for the object parameters $\mathbf{p}_i$ with the system model of Section II, we obtain our overall statistical model for EMT. This model is visualized in Fig. 1.

B. Class Structure and Clustering

According to (6), two or more object parameters $\mathbf{p}_i$ may be equal to the same class parameter $\mathbf{p}_c^*$, which means that the corresponding objects i belong to the same object class c. Let $\mathsf{c}_i\in\mathbb{N}$ denote the (random) class index c of object i. We then trivially have

$$\mathbf{p}_i=\mathbf{p}_{\mathsf{c}_i}^*,\quad\text{for }i=1,2,\dots.\tag{7}$$

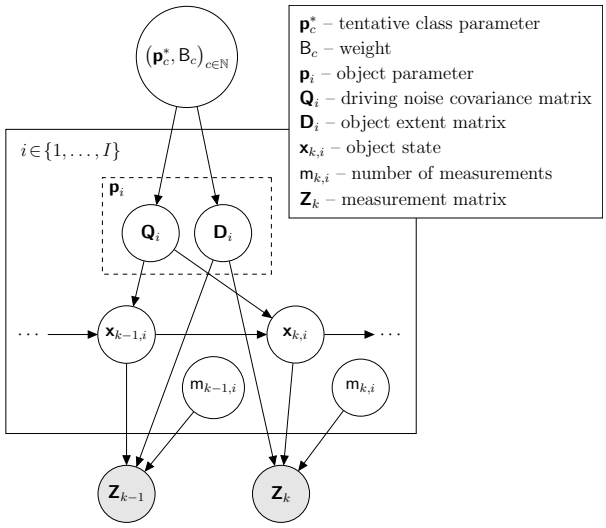

Fig. 1. The proposed statistical model for class-aided EMT. Random variables displayed in shaded disks are observed. The large rectangle indicates a repetition of the nodes and edges contained therein.

Hence, as claimed earlier, the DP prior induces a probabilistic class structure of the objects. Note that the class affiliation (class index c_i) of object i does not change with time k.

Estimating the unknown object class indices c_i, along with the class parameters $\mathbf{p}_c^*$, from the observed measurements amounts to a *clustering* of the objects i. As a result of this clustering, ideally, all the measurements that are generated by all the objects i belonging to the same class c are used for estimating the *common* object parameter $\mathbf{p}_i = \mathbf{p}_c^*$. This improves the accuracy of estimating the class parameters $\mathbf{p}_c^*$ and, in turn, the object parameters $\mathbf{p}_i = \mathbf{p}_{\mathsf{c}_i}^*$.

C. Choice of the Base pdf

According to (5), the tentative class parameters $\mathbf{p}_c^* = \left[\mathsf{d}_{c,1}^{*2}\ \ \mathsf{d}_{c,2}^{*2}\ \ \mathsf{q}_{c,1}^{*2}\ \ \mathsf{q}_{c,2}^{*2}\right]^{\mathrm{T}}$ are drawn independently from the base pdf $f_{\mathrm{G}}(\boldsymbol{p}_c^*)$. We choose a factorizing base pdf

$$f_{\mathrm{G}}(\boldsymbol{p}_c^*) = f\big(d_{c,1}^{*2}\big) f\big(d_{c,2}^{*2}\big) f\big(q_{c,1}^{*2}\big) f\big(q_{c,2}^{*2}\big),$$

corresponding to *a priori* independent parameter components $d_{c,1}^{*2}$ etc. These components have inverse gamma pdfs with known hyperparameters, i.e., $f\big(d_{c,1}^{*2}\big) = \Gamma^{-1}\big(d_{c,1}^{*2}; a_{\mathrm{d},1}, b_{\mathrm{d},1}\big)$, $f\big(d_{c,2}^{*2}\big) = \Gamma^{-1}\big(d_{c,2}^{*2}; a_{\mathrm{d},2}, b_{\mathrm{d},2}\big)$, $f\big(q_{c,1}^{*2}\big) = \Gamma^{-1}\big(q_{c,1}^{*2}; a_{\mathrm{q},1}, b_{\mathrm{q},1}\big)$, and $f\big(q_{c,2}^{*2}\big) = \Gamma^{-1}\big(q_{c,2}^{*2}; a_{\mathrm{q},2}, b_{\mathrm{q},2}\big)$. These prior pdfs are conjugate priors for the associated likelihood functions implied by the model of Section II, since the inverse gamma distribution is the conjugate prior for a Gaussian likelihood function with unknown variance (cf. (2) and (4)).

D. Sufficient Statistic

For a given time $k \in \{1,\ldots,K\}$, let us combine the object states $\mathbf{x}_{k',i}$ and the DA-aware object-generated measurements $\mathbf{Z}_{k',i}$ for all times $k' \le k$ into the matrices $\mathbf{X}_{1:k,i} \triangleq [\mathbf{x}_{k',i}]_{k'=1}^{k}$ and $\mathbf{Z}_{1:k,i} \triangleq [\mathbf{Z}_{k',i}]_{k'=1}^{k}$, respectively. The information about the $\mathbf{p}_i$ and c_i provided by the $\mathbf{X}_{1:k,i}$ and $\mathbf{Z}_{1:k,i}$ can be summarized in a lower-dimensional representation. Concretely, based on our statistical model as formulated in Sections II and III, it can be shown that

$$f(\boldsymbol{p}_i|\boldsymbol{X}_{1:k,i}, \boldsymbol{Z}_{1:k,i}) = f(\boldsymbol{p}_i|\boldsymbol{s}_{k,i}), \quad k \in \mathcal{K}_i,$$

where $\mathbf{s}_{k,i}$ is a *sufficient statistic* random vector that will be specified presently. Thus, the influence of $\mathbf{X}_{1:k,i}$ and $\mathbf{Z}_{1:k,i}$ on $\mathbf{p}_i$ is summarized by $\mathbf{s}_{k,i}$. Similarly, defining $\mathbf{c} \triangleq [\mathsf{c}_1 \cdots \mathsf{c}_I]^{\mathrm{T}}$, $\mathbf{X}_{1:k} \triangleq [\mathbf{X}_{1:k,i}]_{i=1}^{I}$, $\mathbf{Z}_{1:k} \triangleq [\mathbf{Z}_{1:k,i}]_{i=1}^{I}$, and $\mathbf{S}_k \triangleq [\mathbf{s}_{k,i}]_{i=1}^{I}$, one can show

$$p(\boldsymbol{c}|\boldsymbol{X}_{1:k}, \boldsymbol{Z}_{1:k}) = p(\boldsymbol{c}|\boldsymbol{S}_k)$$

and

$$f(\boldsymbol{p}_c^*|\boldsymbol{c}, \boldsymbol{X}_{1:k}, \boldsymbol{Z}_{1:k}) = f(\boldsymbol{p}_c^*|\boldsymbol{c}, \boldsymbol{S}_k).$$

The size of the sufficient statistic matrix $\boldsymbol{S}_k$ does not grow with time k, differently from $\boldsymbol{X}_{1:k}$ and $\boldsymbol{Z}_{1:k}$. Moreover, $\boldsymbol{S}_k$ can be calculated efficiently from $\boldsymbol{X}_k$ and $\boldsymbol{Z}_k$. Using $\boldsymbol{S}_k$ instead of $\boldsymbol{X}_{1:k}$ and $\boldsymbol{Z}_{1:k}$ results in a large reduction of computational complexity (see Section V).

The sufficient statistic is given as $\mathbf{s}_{k,i} = [\mathsf{s}_{k,i,1} \cdots \mathsf{s}_{k,i,5}]^{\mathrm{T}}$ with $\mathsf{s}_{k,i,1} = \sum_{k'=k_{\mathrm{S},i}}^{k} \mathsf{m}_{k',i}$, $\mathsf{s}_{k,i,2} = \sum_{k'=k_{\mathrm{S},i}}^{k} \sum_{n=1}^{\mathsf{m}_{k',i}} \mathsf{u}_{k',i,1}^{(n)2}$, $\mathsf{s}_{k,i,3} = \sum_{k'=k_{\mathrm{S},i}}^{k} \sum_{n=1}^{\mathsf{m}_{k',i}} \mathsf{u}_{k',i,2}^{(n)2}$, $\mathsf{s}_{k,i,4} = \frac{1}{2}\sum_{k'=k_{\mathrm{S},i}+1}^{k} (\mathsf{v}_{k',i,1}^2 + \mathsf{v}_{k',i,2}^2)$, and $\mathsf{s}_{k,i,5} = \frac{1}{2}\sum_{k'=k_{\mathrm{S},i}+1}^{k} (\mathsf{v}_{k',i,3}^2 + \mathsf{v}_{k',i,4}^2)$, for $i = 1,\ldots,I$ and $k \in \mathcal{K}_i$. Here, $\mathsf{u}_{k,i,1}^{(n)}$ and $\mathsf{u}_{k,i,2}^{(n)}$ are the components of $\mathbf{u}_{k,i}^{(n)} = \boldsymbol{R}(\mathbf{x}_{k,i})^{-1}\big(\mathbf{z}_{k,i}^{(n)} - \boldsymbol{H}\mathbf{x}_{k,i}\big)$ (cf. (3); note that $\boldsymbol{R}(\mathbf{x}_{k,i})^{-1} = \boldsymbol{R}(\mathbf{x}_{k,i})^{\mathrm{T}}$), and $\mathsf{v}_{k,i,1}$ etc. are the components of $\mathbf{v}_{k,i} = \mathbf{x}_{k,i} - \boldsymbol{F}\mathbf{x}_{k-1,i}$ (cf. (1)). According to these expressions, $\mathbf{s}_{k,i}$ can be calculated recursively as

$$\mathbf{s}_{k,i} = \mathbf{s}_{k-1,i} + \Delta\mathbf{s}_{k,i}, \quad k = k_{\mathrm{S},i}+1,\ldots,k_{\mathrm{E},i}, \qquad (8)$$

where the components of $\Delta\mathbf{s}_{k,i}$ are $\Delta\mathsf{s}_{k,i,1} = \mathsf{m}_{k,i}$, $\Delta\mathsf{s}_{k,i,2} = \sum_{n=1}^{\mathsf{m}_{k,i}} \mathsf{u}_{k,i,1}^{(n)2}$, $\Delta\mathsf{s}_{k,i,3} = \sum_{n=1}^{\mathsf{m}_{k,i}} \mathsf{u}_{k,i,2}^{(n)2}$, $\Delta\mathsf{s}_{k,i,4} = \frac{1}{2}(\mathsf{v}_{k,i,1}^2 + \mathsf{v}_{k,i,2}^2)$, and $\Delta\mathsf{s}_{k,i,5} = \frac{1}{2}(\mathsf{v}_{k,i,3}^2 + \mathsf{v}_{k,i,4}^2)$. This recursion is initialized with $\mathsf{s}_{k_{\mathrm{S},i},i,1} = \mathsf{m}_{k_{\mathrm{S},i},i}$, $s_{k_{\mathrm{S},i},i,2} = \sum_{n=1}^{\mathsf{m}_{k_{\mathrm{S},i},i}} \mathsf{u}_{k_{\mathrm{S},i},i,1}^{(n)2}$, $\mathsf{s}_{k_{\mathrm{S},i},i,3} = \sum_{n=1}^{\mathsf{m}_{k_{\mathrm{S},i},i}} \mathsf{u}_{k_{\mathrm{S},i},i,2}^{(n)2}$, $\mathsf{s}_{k_{\mathrm{S},i},i,4} = \frac{1}{2}(\mathsf{v}_{k_{\mathrm{S},i},i,1}^2 + \mathsf{v}_{k_{\mathrm{S},i},i,2}^2)$, and $\mathsf{s}_{k_{\mathrm{S},i},i,5} = \frac{1}{2}(\mathsf{v}_{k_{\mathrm{S},i},i,3}^2 + \mathsf{v}_{k_{\mathrm{S},i},i,4}^2)$. We formally extend $\mathbf{s}_{k,i}$ to $k \notin \mathcal{K}_i$ by setting $\mathbf{s}_{k,i} = \mathbf{0}_{5\times 1}$ for $k = 1,\ldots,k_{\mathrm{S},i}-1$ and $\mathbf{s}_{k,i} = \mathbf{s}_{k_{\mathrm{E},i},i}$ for $k = k_{\mathrm{E},i}+1,\ldots,K$.

IV. Method Overview

The proposed EMT method consists of three stages that are performed at each time k: first, an algorithm based on [28] that performs DA and object existence detection and will be briefly termed the *DAOD algorithm*; second, a state estimation algorithm based on the EKF method of [29]; and third, a DP-based learning algorithm that uses the Gibbs sampler for DPs proposed in [26], [27] to learn the objects' class indices and parameters. This three-stage architecture is depicted in Fig. 2. It is suboptimal from a Bayesian inference viewpoint but still able to leverage the latent object class structure for improved EMT. The first two stages together constitute a simple conventional EMT method, which is sufficient to demonstrate the improvement in EMT performance due to class-based parameter learning.

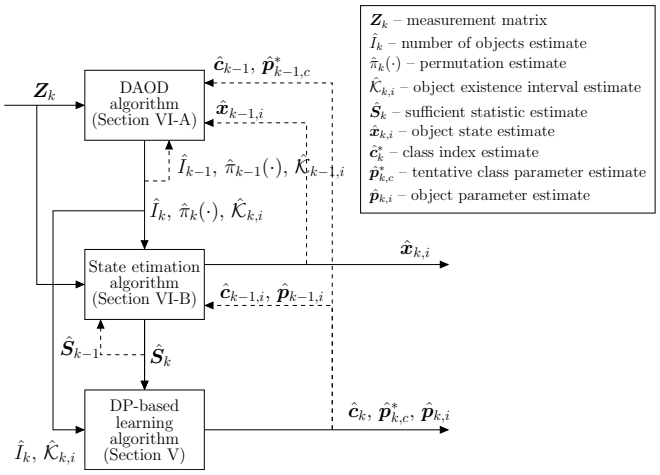

Fig. 2. Block diagram summarizing the proposed EMT architecture.

At time $k \in \{1,\dots,K\}$, the DAOD algorithm uses the observed measurement matrix $\boldsymbol{Z}_k$ and, except for $k=1$, the previous (from time $k-1$) results of the DAOD, state estimation, and DP-based learning algorithms to calculate an estimate $\hat{I}_k$ of the number I of objects, an estimate $\hat{\pi}_k(\cdot)$ of the permutation $\pi_k(\cdot)$, and estimates $\hat{\mathcal{K}}_{k,i}$, $i=1,\dots,\hat{I}_k$ of the object existence intervals $\mathcal{K}_i$. Next, the state estimation algorithm uses $\boldsymbol{Z}_k$, the results of the DAOD algorithm, and, except for $k=1$, the previous object parameter estimates $\hat{\boldsymbol{p}}_{k-1,i}$, $i=1,\dots,\hat{I}_{k-1}$ and the previous sufficient statistic estimate $\hat{\boldsymbol{S}}_{k-1}=[\,\hat{\boldsymbol{s}}_{k-1,i}]_{i=1}^{\hat{I}_{k-1}}$ to calculate object state estimates $\hat{\boldsymbol{x}}_{k,i}$, $i=1,\dots,\hat{I}_k$ and a sufficient statistic estimate $\hat{\boldsymbol{S}}_k=[\,\hat{\boldsymbol{s}}_{k,i}]_{i=1}^{\hat{I}_k}$. Finally, the DP-based learning algorithm uses $\hat{I}_k$, $\hat{\mathcal{K}}_{k,i}$, and $\hat{\boldsymbol{S}}_k$ to calculate an estimate $\hat{\boldsymbol{c}}_k$ of the class index vector $\boldsymbol{c}$, estimates $\hat{\boldsymbol{p}}^*_{k,c}$ of the class parameters $\boldsymbol{p}^*_c$, and estimates $\hat{\boldsymbol{p}}_{k,i}$ of the object parameters $\boldsymbol{p}_i$, $i=1,\dots,\hat{I}_k$. These three stages will be further described in Sections VI-A, VI-B, and V.

The DP-based learning algorithm jointly estimates the (identical) parameters $\boldsymbol{p}_i$ of all objects i belonging to class c from all measurements generated by these objects. The resulting improvement in parameter estimation leads to improved results of DA and state estimation, which implies an improved EMT performance. Conversely, improved results of DA and state estimation lead to improved parameter and class estimation. This interdependency of parameter and class learning, DA, and state estimation is the motivation behind the proposed combination of a conventional EMT method and DP-based parameter and class learning (or DP-based clustering).

V. DP-based Learning Algorithm

The DP-based learning algorithm clusters the objects in that it simultaneously estimates the objects' class indices $\mathsf{c}=[\mathsf{c}_1\cdots\mathsf{c}_I]^{\mathrm{T}}$, the class parameters p^*_c, and the object parameters $\mathsf{P}\triangleq[\mathsf{p}_i]_{i=1}^{I}$. The algorithm uses the number of objects I, object existence intervals $\mathcal{K}_i$, and sufficient statistics $\boldsymbol{S}_k$. In the overall EMT method, these quantities are replaced by estimates $\hat{I}_k$, $\hat{\mathcal{K}}_{k,i}$, and $\hat{\boldsymbol{S}}_k$ that are calculated by the DAOD algorithm ($\hat{I}_k$ and $\hat{\mathcal{K}}_{k,i}$) and by the state estimation algorithm ($\hat{\boldsymbol{S}}_k$), as described in Sections VI-A and VI-B, respectively.

A. Gibbs Sampler

We adopt a Bayesian inference framework. Since the posterior distributions are intractable, we use the Gibbs sampler for DPs proposed in [26], [27]. At each time k, the Gibbs sampler performs L sampling steps. At sampling step $\ell\in\{1,\dots,L\}$, it first draws a class index vector sample $\boldsymbol{c}^{(\ell)}=\big[c_1^{(\ell)}\cdots c_I^{(\ell)}\big]^{\mathrm{T}}$. (Note that $\boldsymbol{c}^{(\ell)}$ and other samples depend on k, but we omit the subscript k for notational simplicity.) Next, it draws a class parameter matrix sample $\boldsymbol{P}^{*(\ell)}_{\boldsymbol{c}^{(\ell)}}\triangleq\big[\boldsymbol{p}^{*(\ell)}_c\big]_{c\in\mathcal{C}^{(\ell)}}$, where $\mathcal{C}^{(\ell)}\triangleq\big\{c_1^{(\ell)},\dots,c_I^{(\ell)}\big\}$ is the set of class index samples. We remark that while certain $c_i^{(\ell)}$ for different i may be equal, $\boldsymbol{P}^{*(\ell)}_{\boldsymbol{c}^{(\ell)}}$ contains the corresponding column $\boldsymbol{p}^{*(\ell)}_{c_i^{(\ell)}}$ only once. From $\boldsymbol{P}^{*(\ell)}_{\boldsymbol{c}^{(\ell)}}$, an object parameter matrix sample $\boldsymbol{P}^{(\ell)}\triangleq\big[\boldsymbol{p}_i^{(\ell)}\big]_{i=1}^{I}$ is then obtained by invoking the identity $\mathsf{p}_i=\mathsf{p}^*_{\mathsf{c}_i}$ in (7) and accordingly choosing the columns of $\boldsymbol{P}^{(\ell)}$ as

$$\boldsymbol{p}_i^{(\ell)}=\boldsymbol{p}^{*(\ell)}_{c_i^{(\ell)}}. \tag{9}$$

It can be shown [32] that the samples $\boldsymbol{c}^{(\ell)}$ and $\boldsymbol{P}^{(\ell)}$ are asymptotically (for $L\to\infty$) distributed according to, respectively, the conditional posterior pmf $p(\boldsymbol{c}|\boldsymbol{X}_{1:k},\boldsymbol{Z}_{1:k})=p(\boldsymbol{c}|\boldsymbol{S}_k)$ and the conditional posterior pdf $f(\boldsymbol{P}|\boldsymbol{X}_{1:k},\boldsymbol{Z}_{1:k})=f(\boldsymbol{P}|\boldsymbol{S}_k)$.

At sampling step ℓ, the Gibbs sampler has available the sufficient statistic $\boldsymbol{S}_k$ and the results of the previous sampling step $\ell-1$, which are $\boldsymbol{c}^{(\ell-1)}=\big[c_1^{(\ell-1)}\cdots c_I^{(\ell-1)}\big]^{\mathrm{T}}$ and $\boldsymbol{P}^{*(\ell-1)}_{\boldsymbol{c}^{(\ell-1)}}=\big[\boldsymbol{p}^{*(\ell-1)}_c\big]_{c\in\mathcal{C}^{(\ell-1)}}$ with $\mathcal{C}^{(\ell-1)}=\big\{c_1^{(\ell-1)},\dots,c_I^{(\ell-1)}\big\}$. For initialization, we set $\boldsymbol{c}^{(0)}=[1\cdots I]^{\mathrm{T}}$, i.e., each object is assigned an individual class. Furthermore, the columns $\boldsymbol{p}^{*(0)}_c$ of $\boldsymbol{P}^{*(0)}_{\boldsymbol{c}^{(0)}}$ are chosen identically as $\boldsymbol{p}^{*(0)}_c=\mathbb{E}[\mathsf{p}^*_c]=\big[\mathbb{E}[\mathsf{d}^{*2}_{c,1}]\ \mathbb{E}[\mathsf{d}^{*2}_{c,2}]\ \mathbb{E}[\mathsf{q}^{*2}_{c,1}]\ \mathbb{E}[\mathsf{q}^{*2}_{c,2}]\big]^{\mathrm{T}}$, for $c\in\mathcal{C}^{(0)}=\{c_1^{(0)},\dots,c_I^{(0)}\}=\{1,\dots,I\}$. Here, according to Section III-C, $\mathbb{E}[\mathsf{d}^{*2}_{c,\nu}]=b_{\mathrm{d},\nu}/(a_{\mathrm{d},\nu}-1)$ and $\mathbb{E}[\mathsf{q}^{*2}_{c,\nu}]=b_{\mathrm{q},\nu}/(a_{\mathrm{q},\nu}-1)$, for $\nu=1,2$.

From the samples $\boldsymbol{c}^{(\ell)}$ and $\boldsymbol{P}^{(\ell)}$ for $\ell=1,\dots,L$, an approximation of the minimum mean squared error estimator [33] of P is calculated as the respective sample mean, and an approximation of the maximum a posteriori estimator (or, more precisely, detector) [33] of c is calculated as the respective sample mode, i.e.,

$$\hat{\boldsymbol{P}}=\frac{1}{L}\sum_{\ell=1}^{L}\boldsymbol{P}^{(\ell)},\qquad \hat{\boldsymbol{c}}=\operatorname{argmax}_{\boldsymbol{c}\in\{\boldsymbol{c}^{(\ell)}\}_{\ell=1}^{L}}L(\boldsymbol{c}). \tag{10}$$

Here, $L(\boldsymbol{c})$ is the number of samples $\boldsymbol{c}^{(\ell)}$ that equal $\boldsymbol{c}$.

The principle of the Gibbs sampler is to draw the sample of a random variable from its *full conditional pdf/pmf*, which is the conditional pdf/pmf given all the other random variables—to be termed the *condition variables*—and the observed measurements. For the condition variables, the most recent samples are substituted, i.e., from sampling step ℓ or, if not yet available, $\ell-1$. In what follows, we describe the Gibbs sampler for a specific, fixed sampling step $\ell\in\{1,\dots,L\}$.

B. Sampling the Class Indices

When drawing the class index sample $c_i^{(\ell)}$ for object i, the condition variable samples in the full conditional pmf are (i) the most recent samples of the class indices $\mathsf{c}_{i'}$ for all $i' \in \{1,\dots,I\}\setminus\{i\}$, i.e., $c_{i'}^{(\ell)}$ for $i'=1,\dots,i-1$ and $c_{i'}^{(\ell-1)}$ for $i'=i+1,\dots,I$, which will be combined into the vector $\boldsymbol{c}_{\neg i}^{(\ell,\ell-1)} \triangleq \big[c_1^{(\ell)}\cdots c_{i-1}^{(\ell)}\, c_{i+1}^{(\ell-1)}\cdots c_I^{(\ell-1)}\big]^{\mathrm{T}}$; and (ii) the class parameter samples $\boldsymbol{p}_c^{*(\ell-1)}$ for $c \in \mathcal{C}_{\neg i}^{(\ell,\ell-1)} \triangleq \big\{c_1^{(\ell)},\dots,c_{i-1}^{(\ell)}, c_{i+1}^{(\ell-1)},\dots,c_I^{(\ell-1)}\big\}$. Here, for a given $c \in \mathcal{C}_{\neg i}^{(\ell,\ell-1)}$, $\boldsymbol{p}_c^{*(\ell-1)}$ is available from the previous sampling step $\ell-1$ if c is also an element of $\mathcal{C}^{(\ell-1)}$; in the opposite case, i.e., if $c \notin \mathcal{C}^{(\ell-1)}$, $\boldsymbol{p}_c^{*(\ell-1)}$ is not available and must be newly generated, which is done as explained at the end of the present subsection and at the end of Section V-C. It will be convenient to use the shorthands $\mathsf{c}_{\neg i} \triangleq \big[\mathsf{c}_1\cdots\mathsf{c}_{i-1}\,\mathsf{c}_{i+1}\cdots\mathsf{c}_I\big]$, $\mathbf{p}^*_{\mathsf{c}_{\neg i}} \triangleq \big[\mathbf{p}_c^*\big]_{c\in\{\mathsf{c}_1,\dots,\mathsf{c}_{i-1},\mathsf{c}_{i+1},\dots,\mathsf{c}_I\}}$, and $\boldsymbol{P}^{*(\ell-1)}_{\boldsymbol{c}_{\neg i}^{(\ell,\ell-1)}} \triangleq \big[\boldsymbol{p}_c^{*(\ell-1)}\big]_{c\in\mathcal{C}_{\neg i}^{(\ell,\ell-1)}}$.

When sampling c_i, there are two cases: (i) c_i equals the class index of one of the other objects, i.e., $\mathsf{c}_i \in \mathcal{C}_{\neg i}^{(\ell,\ell-1)}$, and (ii) $\mathsf{c}_i \notin \mathcal{C}_{\neg i}^{(\ell,\ell-1)}$, i.e., a new class is created for object i. Therefore, we sample c_i from a discrete probability distribution $P_i^{(\ell)}(c)$ that is defined as follows. For $c \in \mathcal{C}_{\neg i}^{(\ell,\ell-1)}$ (case (i)), $P_i^{(\ell)}(c) \triangleq \Pr\big[\mathsf{c}_i = c \,\big|\, \mathbf{c}_{\neg i} = \boldsymbol{c}_{\neg i}^{(\ell,\ell-1)}, \mathbf{P}^*_{\mathbf{c}_{\neg i}} = \boldsymbol{P}^{*(\ell-1)}_{\boldsymbol{c}_{\neg i}^{(\ell,\ell-1)}}, \mathbf{S}_k = \boldsymbol{S}_k\big]$, which can be shown to be given by

$$\begin{aligned} P_i^{(\ell)}(c) = C\,\frac{1}{\alpha+I-1}\,N_c\big(\boldsymbol{c}_{\neg i}^{(\ell,\ell-1)}\big) \\ \times \mathcal{N}\big([\sqrt{s_{k,i,2}}\ \sqrt{s_{k,i,3}}]^{\mathrm{T}}; \mathbf{0}_{2\times1}, \boldsymbol{D}_c^{*(\ell-1)}\big) \\ \times \mathcal{N}\big([\sqrt{s_{k,i,4}}\ \sqrt{s_{k,i,5}}]^{\mathrm{T}}; \mathbf{0}_{2\times1}, \breve{\boldsymbol{Q}}_c^{*(\ell-1)}\big). \end{aligned} \tag{11}$$

Here, C is a constant, $N_c\big(\boldsymbol{c}_{\neg i}^{(\ell,\ell-1)}\big)$ is the number of elements of $\boldsymbol{c}_{\neg i}^{(\ell,\ell-1)}$ that equal c, $\boldsymbol{D}_c^{*(\ell-1)} \triangleq \mathrm{diag}\big((d_{c,1}^{*2})^{(\ell-1)}, (d_{c,2}^{*2})^{(\ell-1)}\big)$, and $\breve{\boldsymbol{Q}}_c^{*(\ell-1)} \triangleq \mathrm{diag}\big((q_{c,1}^{*2})^{(\ell-1)}, (q_{c,2}^{*2})^{(\ell-1)}\big)$.

For $c \notin \mathcal{C}_{\neg i}^{(\ell,\ell-1)}$ (case (ii)), we create a new value of c, denoted $c_{i,\mathrm{new}}^{(\ell)}$. Accordingly, $P_i^{(\ell)}(c)$ is equal to $P_i^{(\ell)}\big(c_{i,\mathrm{new}}^{(\ell)}\big) \triangleq \Pr\big[\mathsf{c}_i \notin \mathcal{C}_{\neg i}^{(\ell,\ell-1)} \,\big|\, \mathbf{c}_{\neg i} = \boldsymbol{c}_{\neg i}^{(\ell,\ell-1)}, \mathbf{P}^*_{\mathbf{c}_{\neg i}} = \boldsymbol{P}^{*(\ell-1)}_{\boldsymbol{c}_{\neg i}^{(\ell,\ell-1)}}, \mathbf{S}_k = \boldsymbol{S}_k\big]$, which can be shown to be given by

$$\begin{aligned} P_i^{(\ell)}\big(c_{i,\mathrm{new}}^{(\ell)}\big) = C\,\frac{\alpha}{\alpha+I-1}\,\mathcal{N}\big([\sqrt{s_{k,i,2}}\ \sqrt{s_{k,i,3}}]^{\mathrm{T}}; \mathbf{0}_{2\times1}, \tilde{\boldsymbol{D}}\big) \\ \times \mathcal{N}\big([\sqrt{s_{k,i,4}}\ \sqrt{s_{k,i,5}}]^{\mathrm{T}}; \mathbf{0}_{2\times1}, \tilde{\boldsymbol{Q}}\big)\,\frac{f_{\mathrm{G}}(\tilde{\boldsymbol{p}})}{f(\tilde{\boldsymbol{p}}\,|\,\boldsymbol{s}_{k,i})}. \end{aligned}$$

Here, C is the same constant as in (11), which is chosen such that $\sum_{c\in\mathcal{C}_{\neg i}^{(\ell,\ell-1)}} P_i^{(\ell)}(c) + P_i^{(\ell)}\big(c_{i,\mathrm{new}}^{(\ell)}\big) = 1$; furthermore, $\tilde{\boldsymbol{D}} \triangleq \mathrm{diag}\big(\tilde{d}_1^2, \tilde{d}_2^2\big)$, $\tilde{\boldsymbol{Q}} \triangleq \mathrm{diag}\big(\tilde{q}_1^2, \tilde{q}_2^2\big)$, $\tilde{\boldsymbol{p}} \triangleq \big[\tilde{d}_1^2\ \tilde{d}_2^2\ \tilde{q}_1^2\ \tilde{q}_2^2\big]^{\mathrm{T}}$, where $\tilde{d}_1^2$ etc. are arbitrary except that $f_{\mathrm{G}}(\tilde{\boldsymbol{p}}) \neq 0$, and

$$\begin{aligned} f(\tilde{\boldsymbol{p}}\,|\,\boldsymbol{s}_{k,i}) = \Gamma^{-1}\big(\tilde{d}_1^2; a'_{\mathrm{d},i,1}, b'_{\mathrm{d},i,1}\big)\,\Gamma^{-1}\big(\tilde{d}_2^2; a'_{\mathrm{d},i,2}, b'_{\mathrm{d},i,2}\big) \\ \times \Gamma^{-1}\big(\tilde{q}_1^2; a'_{\mathrm{q},i,1}, b'_{\mathrm{q},i,1}\big)\,\Gamma^{-1}\big(\tilde{q}_2^2; a'_{\mathrm{q},i,2}, b'_{\mathrm{q},i,2}\big), \end{aligned}$$

with $a'_{\mathrm{d},i,\nu} = a_{\mathrm{d},\nu} + \frac{1}{2}\,s_{k,i,1}$, $b'_{\mathrm{d},i,\nu} = b_{\mathrm{d},\nu} + \frac{1}{2}\,s_{k,i,\nu+1}$, $a'_{\mathrm{q},i,\nu} = a_{\mathrm{q},\nu} + k_{\mathrm{E},i} - k_{\mathrm{S},i} + 1$, and $b'_{\mathrm{q},i,\nu} = b_{\mathrm{q},\nu} + s_{k,i,\nu+3}$, for $\nu = 1, 2$.

The new class index value $c_{i,\mathrm{new}}^{(\ell)}$ is chosen as the smallest integer that is larger than any previously sampled c value, i.e,

$$c_{i,\mathrm{new}}^{(\ell)} \triangleq \max\big\{c_{\max}^{(\ell-1)}, \max_{i'\in\{1,\dots,i-1\}} c_{i'}^{(\ell)}\big\} + 1, \tag{12}$$

with $c_{\max}^{(\ell-1)} \triangleq \max_{\ell'\in\{1,\dots,\ell-1\}, i\in\{1,\dots,I\}} c_i^{(\ell')}$. If $c_i^{(\ell)} = c_{i,\mathrm{new}}^{(\ell)}$, then a "previous" sample $\boldsymbol{p}^{*(\ell-1)}_{c_{i,\mathrm{new}}^{(\ell)}}$ of $\mathbf{p}^*_{c_{i,\mathrm{new}}^{(\ell)}}$ must be generated. This sample will be used when sampling the class indices of the successive objects $i' > i$, i.e., it is used in $P_{i'}^{(\ell)}\big(c_{i',\mathrm{new}}^{(\ell)}\big) = \Pr\big[\mathsf{c}_{i'} \notin \mathcal{C}_{\neg i'}^{(\ell,\ell-1)} \,\big|\, \mathbf{c}_{\neg i'} = \boldsymbol{c}_{\neg i'}^{(\ell,\ell-1)}, \mathbf{P}^*_{\mathbf{c}_{\neg i'}} = \boldsymbol{P}^{*(\ell-1)}_{\boldsymbol{c}_{\neg i'}^{(\ell,\ell-1)}}, \mathbf{S}_k = \boldsymbol{S}_k\big]$ as a column of $\boldsymbol{P}^{*(\ell-1)}_{\boldsymbol{c}_{\neg i'}^{(\ell,\ell-1)}}$. The full conditional pdf from which $\boldsymbol{p}^{*(\ell-1)}_{c_{i,\mathrm{new}}^{(\ell)}}$ is drawn will be considered in the next subsection.

C. Sampling the Class/Object Parameters

After drawing the class index vector sample $\boldsymbol{c}^{(\ell)} = \big[c_1^{(\ell)}\cdots c_I^{(\ell)}\big]^{\mathrm{T}}$, the second task of the Gibbs sampler at sampling step ℓ is to draw class parameter samples $\boldsymbol{p}_c^{*(\ell)}$ for all $c \in \mathcal{C}^{(\ell)} = \big\{c_1^{(\ell)},\dots,c_I^{(\ell)}\big\}$, or equivalently to draw $\boldsymbol{P}^{*(\ell)}_{\boldsymbol{c}^{(\ell)}} = \big[\boldsymbol{p}_c^{*(\ell)}\big]_{c\in\mathcal{C}^{(\ell)}}$. When drawing $\boldsymbol{p}_c^{*(\ell)}$ for a specific $c \in \mathcal{C}^{(\ell)}$, the condition variable samples are $\boldsymbol{c}^{(\ell)}$ and the most recent samples of $\mathbf{p}^*_{c'}$ for all $c' \in \mathcal{C}^{(\ell)}\setminus\{c\}$, i.e., $\boldsymbol{p}_{c'}^{*(\ell)}$ or $\boldsymbol{p}_{c'}^{*(\ell-1)}$, which are combined into the matrix $\boldsymbol{P}_{\neg c}^{*(\ell,\ell-1)}$. Accordingly, $\boldsymbol{p}_c^{*(\ell)}$ is drawn from the full conditional pdf $f_{\mathbf{p}^*_c|\mathbf{c},\mathbf{P}^*_{\neg c},\mathbf{S}_k}\big(\boldsymbol{p}_c^*\,\big|\,\boldsymbol{c}^{(\ell)}, \boldsymbol{P}_{\neg c}^{*(\ell,\ell-1)}, \boldsymbol{S}_k\big)$, where $\mathbf{P}^*_{\neg c} \triangleq \big[\mathbf{p}^*_{c'}\big]_{c'\in\{c_1,\dots,c_I\}\setminus\{c\}}$. This pdf can be shown to be given by

$$\begin{aligned} f(\boldsymbol{p}_c^*\,|\,\boldsymbol{c}, \boldsymbol{P}^*_{\neg c}, \boldsymbol{S}_k) \\ = \Gamma^{-1}\big(d_{c,1}^{*2}; a''_{\mathrm{d},c,1}, b''_{\mathrm{d},c,1}\big)\,\Gamma^{-1}\big(d_{c,2}^{*2}; a''_{\mathrm{d},c,2}, b''_{\mathrm{d},c,2}\big) \\ \times \Gamma^{-1}\big(q_{c,1}^{*2}; a''_{\mathrm{q},c,1}, b''_{\mathrm{q},c,1}\big)\,\Gamma^{-1}\big(q_{c,2}^{*2}; a''_{\mathrm{q},c,2}, b''_{\mathrm{q},c,2}\big), \end{aligned}$$

with $a''_{\mathrm{d},c,\nu} = a_{\mathrm{d},\nu} + \frac{1}{2}\sum_{i:c_i=c} s_{k,i,1}$, $b''_{\mathrm{d},c,\nu} = b_{\mathrm{d},\nu} + \frac{1}{2}\sum_{i:c_i=c} s_{k,i,\nu+1}$, $a''_{\mathrm{q},c,\nu} = a_{\mathrm{q},\nu} + K(c) - N(c)$, and $b''_{\mathrm{q},c,\nu} = b_{\mathrm{q},\nu} + \sum_{i:c_i=c} s_{k,i,\nu+3}$, for $\nu = 1, 2$. Here, $K(c)$ is the number of time steps k at which at least one existing object belongs to class c, and $N(c)$ is the number of c_i that equal c.

In addition, as mentioned in Section V-B, sampling $\mathbf{p}_c^*$ is also necessary during the sampling of the c_i if $c_i^{(\ell)} = c_{i,\mathrm{new}}^{(\ell)}$: in that case, a "previous" sample $\boldsymbol{p}^{*(\ell-1)}_{c_{i,\mathrm{new}}^{(\ell)}}$ is drawn from the same full conditional pdf, but with $\boldsymbol{P}_{\neg c}^{*(\ell,\ell-1)}$ replaced by $\boldsymbol{P}^{*(\ell-1)}_{\boldsymbol{c}_{\neg i}^{(\ell,\ell-1)}}$ because all parameter samples for $c' \neq c$ have already been obtained at this point.

VI. Data Association, Object Existence Detection, and Object State Estimation

A. DAOD Algorithm

The DAOD algorithm performs DA and object existence detection, i.e., it calculates estimates $\hat{I}_k$, $\hat{\pi}_k(\cdot)$, and $\hat{\mathcal{K}}_{k,i}$ for $i = 1,\dots,\hat{I}_k$, as summarized in Section IV. It consists of the DA and object existence detection parts of the multiobject tracking method described in [28] with the following modifications. First, we consider 2D measurements. Second, we add a clustering task to account for the fact that an object can generate multiple measurements. This clustering task is solved using the expectation-maximization (EM) clustering method [34] with the number of clusters equal to $\hat{I}_{k-1}$ (in the DA

part of the DAOD algorithm) or κ (in the object existence detection part). Here, $\hat{I}_{k-1}$ is upper bounded by a prespecified parameter $I_{\max}$ and κ is determined by a model selection stage that performs EM-based clustering for $\kappa' = 1, \ldots, \kappa_{\max}$ clusters and then selects the optimal value of κ' according to the Akaike information criterion [35], [36].

The DAOD algorithm requires the parameters $d^2_{i,1}$, $d^2_{i,2}$, $q^2_{i,1}$, and $q^2_{i,2}$, for $i = 1, \ldots, I_{\max}$, which will be referred to as the *DAOD parameters*. Except for $k = 1$, the DAOD parameters are determined for $i = 1, \ldots, \hat{I}_{k-1}$ as $d^2_{i,1} = \hat{d}^2_{k-1,i,1}$, $d^2_{i,2} = \hat{d}^2_{k-1,i,2}$, $q^2_{i,1} = \hat{q}^2_{k-1,i,1}$, and $q^2_{i,2} = \hat{q}^2_{k-1,i,2}$. Here, $\hat{I}_{k-1}$ was previously calculated by the DAOD algorithm, and $\hat{d}^2_{k-1,i,1}$ etc. were previously calculated by the DP-based learning algorithm (see (9), (10), and Section V-C). If $\hat{I}_{k-1} < I_{\max}$ (recall that $\hat{I}_{k-1}$ is upper bounded by $I_{\max}$), then the remaining DAOD parameters for $i = \hat{I}_{k-1}+1, \ldots, I_{\max}$ are determined using the following procedure based on the *Chinese restaurant process* (CRP), which is closely related to the DP [37].

First, we generate class index values c_i. For $i = 1, \ldots, \hat{I}_{k-1}$, we set c_i equal to the respective estimate $\hat{c}_{k-1,i}$ that was previously calculated by the DP-based learning algorithm (see (10)), and for $i = \hat{I}_{k-1}+1, \ldots, I_{\max}$, we recursively sample c_i from the CRP-related conditional pmf [37]

$$p(c_i|c_1, \ldots, c_{i-1}) = \frac{1}{\alpha + i - 1} \Bigg(\alpha \mathbb{1}(c_i = c_{i,\text{new}}) + \sum_{c=1}^{c_{i-1,\max}} N_c(c_1, \ldots, c_{i-1}) \mathbb{1}(c_i = c) \Bigg).$$

Here, the $c_{i'}$ for $i' = 1, \ldots, i-1$ were sampled previously or were set equal to $\hat{c}_{k-1,i'}$; $c_{i-1,\max} \triangleq \max_{i' \in \{1,\ldots,i-1\}} c_{i'}$; $c_{i,\text{new}} \triangleq c_{i-1,\max} + 1$; and $N_c(c_1, \ldots, c_{i-1})$ is the number of $c_{i'}$ with $i' \in \{1, \ldots, i-1\}$ that are equal to c.

The DAOD parameters $d^2_{i,1}$, $d^2_{i,2}$, $q^2_{i,1}$, and $q^2_{i,2}$ for $i = \hat{I}_{k-1}+1, \ldots, I_{\max}$ are now calculated differently for the two cases $c_i \in \{c_1, \ldots, c_{\hat{I}_{k-1}}\}$ (i.e., c_i is a class for which the DP-based learning algorithm already calculated parameter estimates $\hat{d}^{*2}_{k,c_i,1}$, $\hat{d}^{*2}_{k,c_i,2}$, $\hat{q}^{*2}_{k,c_i,1}$, and $\hat{q}^{*2}_{k,c_i,2}$) and $c_i \notin \{c_1, \ldots, c_{\hat{I}_{k-1}}\}$. In the first case, we set the DAOD parameters equal to the mentioned parameter estimates. In the second case, i.e., $c_i \notin \{c_1, \ldots, c_{\hat{I}_{k-1}}\}$, there are two subcases: if $c_i = c_{i,\text{new}} = c_{i-1,\max} + 1$, we sample the DAOD parameters from the prior pdfs specified in Section III-C, and if $c_i \in \{\hat{c}_{k-1,\max}+1, \ldots, c_{i-1,\max}\}$ with $\hat{c}_{k-1,\max} \triangleq \max_{i \in \{1,\ldots,\hat{I}_{k-1}\}} \hat{c}_{k-1,i}$—i.e., c_i is a class for which we already sampled the DAOD parameters in the former subcase for some $i' < i$—we set the DAOD parameters equal to those samples.

For initialization at the first time step $k = 1$, $\hat{I}_{k-1} = \hat{I}_0$ is set to 0, and $\hat{c}_{k-1,i} = \hat{c}_{0,i}$ is generated for all $i = 1, \ldots, I_{\max}$ by means of the CRP-based sampling procedure, which is initialized with $\hat{c}_{0,i} = 1$.

B. State Estimation Algorithm

The state estimation algorithm—the second stage of the proposed EMT method according to Section IV—calculates state estimates $\hat{\boldsymbol{X}}_k$ by means of the EKF method of [29], which is modified in that it uses the object extent parameters provided by the DP-based learning stage (the respective elements of $\hat{\boldsymbol{P}}$ in (10)). In addition, it calculates sufficient statistic estimates $\hat{\boldsymbol{S}}_k$ according to (8).

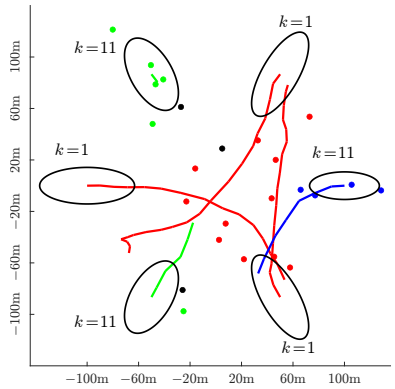

Fig. 3. Example of a simulated EMT scenario observed during 20 time steps. The trajectories of six elliptical objects—each belonging to one of three classes—are drawn in blue, red, and green according to their class affiliations, i.e., all objects belonging to the same class are drawn in the same color. Object-generated measurements are drawn as dots using the color corresponding to the object's class affiliation, and measurements due to clutter are drawn as black dots. We only display the measurements present at time $k = 11$. The initial position of each object i is indicated by an ellipse with half-axes $d_{i,1}$ and $d_{i,2}$ and labeled by the time k at which the object appears.

VII. Experimental Performance Evaluation

A. Simulation Setting

We simulated an EMT scenario during $K = 100$ time steps. Three new objects appear every 10 time steps with initial positions regularly spaced on a circle of radius 100 m, and move with an initial velocity of 10 m/s toward the center of the circle. At each time step k, a previously existing object continues to exist with probability 0.92. The system model is as described in Sections II and III with the following model hyperparameters (which are known by all methods): $\alpha = 0.5$, $a_{\mathrm{d},1} = a_{\mathrm{d},2} = 12$, $b_{\mathrm{d},1} = 11000$, $b_{\mathrm{d},2} = 5500$, $a_{\mathrm{q},1} = a_{\mathrm{q},2} = 4$, $b_{\mathrm{q},1} = 300$, $b_{\mathrm{q},2} = 30$, $\lambda = 4$, and $\lambda_0 = 5$. This results in a mean number of object classes equal to 2.64 with mean extent parameters $\mathbb{E}[\mathsf{d}^{*2}_{c,1}] = 1000\,\mathrm{m}^2$ and $\mathbb{E}[\mathsf{d}^{*2}_{c,2}] = 500\,\mathrm{m}^2$ and mean driving noise variances $\mathbb{E}[\mathsf{q}^{*2}_{c,1}] = 100\,\mathrm{m}^2/\mathrm{s}^2$ and $\mathbb{E}[\mathsf{q}^{*2}_{c,2}] = 10\,\mathrm{m}^2/\mathrm{s}^4$. The sampling period is $\Delta T = 1\,\mathrm{s}$. The initial state covariance matrix is $\boldsymbol{\Sigma}_{\mathrm{S}} = \mathrm{diag}(100^2, 100^2, 10^2, 10^2)$. The clutter pdf is uniform. Fig. 3 depicts an example of a simulated EMT scenario during 20 time steps.

We compare the proposed EMT method performing DP-based object clustering and parameter learning as described in Sections IV–VI (dubbed DP-EMT) with two reference methods that use alternative parameter learning techniques: a baseline EMT method that does not perform clustering (dubbed B-EMT) and an EMT method that performs EM-based clustering (dubbed EM-EMT). More specifically, B-EMT equals the proposed EMT method except that it does not assume a class structure, i.e., the parameters of each object i are estimated using only the measurements assigned to object

TABLE I
NRMSE OF THE OBJECT PARAMETER ESTIMATES.

	DP-EMT	B-EMT	EM-EMT
NRMSE$(\hat{d}^2_{k,i,1})$	0.46	7.07	0.93
NRMSE$(\hat{d}^2_{k,i,2})$	0.70	9.74	0.86
NRMSE$(\hat{q}^2_{k,i,1})$	4.67	9.89	0.91
NRMSE$(\hat{q}^2_{k,i,2})$	0.83	7.58	1.13

i rather than the measurements of all objects belonging to the same class as object i. EM-EMT, on the other hand, performs parameter learning by means of the technique proposed in [18], which uses the EM algorithm to cluster the objects and estimate the parameters of each object class. To adapt [18] to our EMT scenario, we replaced in our overall EMT method the DP-based learning algorithm by the EM-based learning algorithm of [18]. Just as our method (DP-EMT), EM-EMT learns the object classes and parameters without knowing the number of classes or the objects' class affiliations beforehand. However, it uses a maximum likelihood framework whereas DP-EMT uses a Bayesian framework. Because use of the initialization procedure specified in [18] resulted in poor tracking performance of EM-EMT in our challenging tracking scenario, we improved on that procedure by heuristically incorporating the prior mean of $\mathbf{p}_i$ in the initialization of EM-EMT.

All three EMT methods use the DAOD algorithm with $I_{\max} = 60$ and $\kappa_{\max} = 3$ for DA and object existence detection (see Section VI-A). DP-EMT and B-EMT use $L = 80$ sampling steps for inference, preceded by a burn-in period comprising 20 sampling steps. EM-EMT requires the specification of a maximum number of object classes, which we set to 4. (We observed this choice to result in the best tracking performance, even better than choosing the integer closest to the mean number of classes, which is 3.)

B. Results

We first evaluate the parameter learning performance of the three methods. Table I shows the normalized root mean squared error (NRMSE) of the parameter estimates $\hat{d}^2_{k,i,1}$, $\hat{d}^2_{k,i,2}$, $\hat{q}^2_{k,i,1}$, and $\hat{q}^2_{k,i,2}$ obtained with each method. Here, e.g., NRMSE$(\hat{d}^2_{k,i,1})$ is calculated by averaging $\left(\hat{d}^2_{k,i,1} - d^2_{i,1}\right)^2$ over all times k, all objects i, and all datasets, then taking the square root of the result, and finally dividing by $\mathbb{E}[\mathsf{d}^{*2}_{c,1}]$. It can be seen that DP-EMT significantly outperforms both B-EMT and EM-EMT except for $\hat{q}^2_{k,i,1}$, i.e., the position-related driving noise variance, where it is outperformed by EM-EMT. We also see that B-EMT—which does not perform object clustering—exhibits the worst parameter learning performance. We can conclude from these results that object clustering tends to strongly improve the parameter learning performance, and the DP approach (DP-EMT) typically outperforms the maximum likelihood approach (EM-EMT).

Next, we evaluate the performance of object class learning exhibited by DP-EMT and EM-EMT. We do so by considering the number of detected object classes at the final time $K = 100$, averaged over all datasets, which we observed to be 2.13 for DP-EMT and 1 for EM-EMT. Comparing with the true average number of object classes, which we observed to be 2.64, we conclude that EM-EMT tends to underestimate the number of object classes whereas the estimate of DP-EMT is quite close to the true number. We also observed in our simulations that EM-EMT estimates the number of object classes more accurately during the first 30 time steps, but the estimate tends toward 1 as more measurements are observed. A possible explanation is that in EM-EMT, object class detection relies on the Bayesian information criterion [36], whose penalty for new object classes grows with the number of measurements.

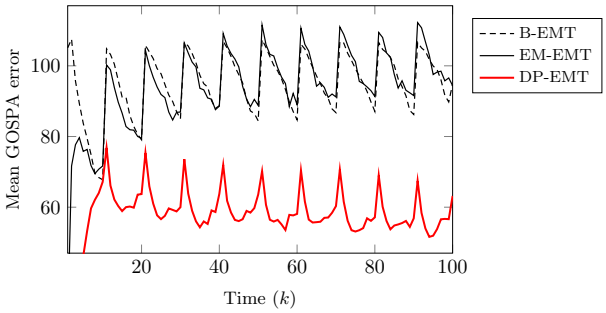

Fig. 4. Mean GOSPA error versus time.

Finally, we evaluate the tracking performance of the methods. Our performance metric is the Euclidean distance-based mean generalized optimal subpattern assignment (GOSPA) error [38] with cutoff parameter 120, order 1, and parameter α set to 2 (not to be confused with the DP's concentration parameter α). Fig. 4 shows the mean GOSPA error, obtained by averaging over 400 datasets, as a function of time. The periodic peaks are due to the appearance of new objects. It is seen that the tracking performance of DP-EMT is consistently better than that of both B-EMT and EM-EMT. Furthermore, the tracking performance of B-EMT, which does not use object clustering, is initially (during the first 40 time steps, and most notably during the first 10 time steps) worse than that of EM-EMT. This can be explained by the fact that only few measurements are available during this initial period, which causes the parameter learning in B-EMT to be inaccurate and, in turn, results in poor tracking performance. Because DP-EMT and EM-EMT exhibit better tracking performance during the same period, we conclude that object clustering and class-aided parameter learning—as performed by DP-EMT and EM-EMT—are most beneficial when few measurements are available. On the other hand, for later times this performance gain of EM-EMT over B-EMT disappears until EM-EMT even performs partly worse than B-EMT. This seems to be related to the fact—mentioned earlier—that the detection of object classes by EM-EMT becomes less reliable with time. In contrast, DP-EMT exhibits a consistent performance gain over B-EMT (and also over EM-EMT).

VIII. CONCLUSION

We proposed a class-aided EMT method for scenarios where the objects belong to classes that are distinguished by the

values of certain statistical model parameters. The number of object classes, the class parameter values, and the objects' class affiliations are allowed to be unknown. Our method combines a conventional EMT algorithm with a Bayesian technique for object clustering and parameter learning. This technique uses a parameter-dependent state-space model with a DP prior for the unknown object parameters as well as a Gibbs sampler for Bayesian inference. Simulation results demonstrated significant performance gains regarding parameter estimation, object clustering, and object tracking.

In our future work, we will extend our framework to achieve a full integration of DP-based parameter and class learning, DA, object existence detection, and state estimation within a joint Bayesian model. We conjecture that this will yield substantial additional performance gains.

ACKNOWLEDGMENT

We would like to thank the anonymous reviewers for constructive comments that led to an improved presentation.

REFERENCES

[1] K. Granström, M. Baum, and S. Reuter, "Extended object tracking: Introduction, overview and applications," *J. Adv. Inf. Fusion*, vol. 12, no. 2, pp. 139–174, Dec. 2017.

[2] M. Michaelis, P. Berthold, D. Meissner, and H.-J. Wuensche, "Heterogeneous multi-sensor fusion for extended objects in automotive scenarios using Gaussian processes and a GMPHD-filter," in *Proc. SDF-17*, Bonn, Germany, Oct. 2017, pp. 1–6.

[3] 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.

[4] I. Smal, K. Draegestein, N. Galjart, W. Niessen, and E. Meijering, "Particle filtering for multiple object tracking in dynamic fluorescence microscopy images: Application to microtubule growth analysis," *IEEE Trans. Medical Imaging*, vol. 27, no. 6, pp. 789–804, June 2008.

[5] K. Granström, A. Natale, P. Braca, G. Ludeno, and F. Serafino, "Gamma Gaussian inverse Wishart probability hypothesis density for extended target tracking using X-band marine radar data," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 12, pp. 6617–6631, Dec. 2015.

[6] G. Ferri, A. Munafò, A. Tesei, P. Braca, F. Meyer, K. Pelekanakis, R. Petroccia, J. Alves, C. Strode, and K. LePage, "Cooperative robotic networks for underwater surveillance: An overview," *IET Radar Sonar Navig.*, vol. 11, no. 12, pp. 1740–1761, Dec. 2017.

[7] F. Septier, A. Carmi, and S. Godsill, "Tracking of multiple contaminant clouds," in *Proc. FUSION-09*, Seattle, WA, USA, July 2009, pp. 1280–1287.

[8] X. Yang and Q. Jiao, "Variational approximation for adaptive extended target tracking in clutter with random matrix," *IEEE Trans. Veh. Technol.*, vol. 72, no. 10, pp. 12 639–12 652, Oct. 2023.

[9] 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.

[10] G. Papa, P. Braca, S. Horn, S. Marano, V. Matta, and P. Willett, "Multisensor adaptive Bayesian tracking under time-varying target detection probability," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 52, no. 5, pp. 2193–2209, 2016.

[11] K. G. Jamieson, M. R. Gupta, and D. W. Krout, "Sequential Bayesian estimation of the probability of detection for tracking," in *Proc. FUSION-09*, Seattle, WA, USA, July 2009, pp. 641–648.

[12] L. Wang and R. Zhan, "Joint detection, tracking and classification of multiple maneuvering star-convex extended targets," *IEEE Sensors J.*, vol. 24, no. 4, pp. 5004–5024, Feb. 2024.

[13] Y. Li, P. Wei, G. Li, Y. Chen, L. Gao, and H. Zhang, "Joint detection, tracking and classification of multiple extended objects based on the JDTC-GIW-MeMBer filter," *Signal Processing*, vol. 178, p. 107800, 2021.

[14] S. Challa and G. W. Pulford, "Joint target tracking and classification using radar and ESM sensors," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 37, no. 3, pp. 1039–1055, 2001.

[15] J. Liang, M. Li, Z. Jing, and H. Pan, "Multi-target joint detection, tracking and classification based on marginal GLMB filter and belief function theory," *Sensors*, vol. 20, no. 15, 2020.

[16] C. Magnant, A. Giremus, E. Grivel, L. Ratton, and B. Joseph, "Joint tracking and classification based on kinematic and target extent measurements," in *Proc. FUSION-15*, Washington, DC, USA, July 2015, pp. 1748–1755.

[17] C. Magnant, S. Kemkemian, and L. Zimmer, "Joint tracking and classification for extended targets in maritime surveillance," in *Proc. IEEE Radar Conf.*, Oklahoma City, OK, USA, Apr. 2018, pp. 1117–1122.

[18] X. He, R. Tharmarasa, T. Kirubarajan, A. Jousselme, and P. Valin, "Joint class identification and target classification using multiple HMMs," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 50, no. 2, pp. 1269–1282, 2014.

[19] A. Lin, Y. Zhang, J. Heng, S. A. Allsop, K. M. Tye, P. E. Jacob, and D. Ba, "Clustering time series with nonlinear dynamics: A Bayesian non-parametric and particle-based approach," in *Proc. AISTATS-19*, Naha, Okinawa, Japan, Apr. 2019, pp. 2476–2484.

[20] S. Ghosal and A. Van der Vaart, *Fundamentals of Nonparametric Bayesian Inference*. Cambridge University Press, 2017.

[21] L. E. Nieto-Barajas and A. Contreras-Cristán, "A Bayesian nonparametric approach for time series clustering," *Bayesian Anal.*, vol. 9, no. 1, pp. 147–170, March 2014.

[22] S. Chiappa and D. Barber, "Dirichlet mixtures of Bayesian linear Gaussian state-space models: A variational approach," Max Planck Institute for Biological Cybernetics, Tübingen, Germany, Tech. Rep. 161, June 2007.

[23] N. Bathaee and H. Sheikhzadeh, "Non-parametric Bayesian inference for continuous density hidden Markov mixture model," *Statist. Methodol.*, vol. 33, pp. 256–275, 2016.

[24] K. P. Lennox, D. B. Dahl, M. Vannucci, R. Day, and J. W. Tsai, "A Dirichlet process mixture of hidden Markov models for protein structure prediction," *Ann. Appl. Statist.*, vol. 4, no. 2, p. 916, 2010.

[25] V. Bastani, L. Marcenaro, and C. Regazzoni, "Unsupervised trajectory pattern classification using hierarchical Dirichlet process mixture hidden Markov model," in *Proc. IEEE MLSP-14*, Reims, France, Sept. 2014, pp. 1–6.

[26] S. N. MacEachern, "Estimating normal means with a conjugate style Dirichlet process prior," *Commun. Stat. Simul. Comput.*, vol. 23, no. 3, pp. 727–741, 1994.

[27] M. West, P. Müller, and M. D. Escobar, "Hierarchical priors and mixture models, with application in regression and density estimation," in *Aspects of Uncertainty: A Tribute to D.V. Lindley*, P. Freeman, Ed. New York, NY, USA: Wiley, 1994, pp. 363–386.

[28] X. Weng, J. Wang, D. Held, and K. Kitani, "3D multi-object tracking: A baseline and new evaluation metrics," in *Proc. IEEE/RSJ IROS-20*, Las Vegas, NV, USA, Oct. 2020, pp. 10 359–10 366.

[29] S. Yang and M. Baum, "Extended Kalman filter for extended object tracking," in *Proc. IEEE ICASSP-17*, New Orleans, LA, USA, March 2017, pp. 4386–4390.

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

[31] J. Sethuraman, "A constructive definition of Dirichlet priors," *Stat. Sin.*, vol. 4, no. 2, pp. 639–650, 1994.

[32] C. P. Robert and G. Casella, *Monte Carlo Statistical Methods*. New York, NY, USA: Springer, 1999.

[33] G. E. Box and G. C. Tiao, *Bayesian Inference in Statistical Analysis*. New York, NY, USA: Wiley, 1992.

[34] G. J. McLachlan and T. Krishnan, *The EM Algorithm and Extensions*. Hoboken, NJ, USA: Wiley, 2007.

[35] H. Akaike, "Information theory and an extension of the maximum likelihood principle," in *Breakthroughs in Statistics*, S. Kotz and N. Johnson, Eds. Springer, 1992, pp. 610–624.

[36] M. R. Forster, "Key concepts in model selection: Performance and generalizability," *J. Math. Psych.*, vol. 44, no. 1, pp. 205–231, 2000.

[37] J. Pitman, *Combinatorial Stochastic Processes*. Berlin, Germany: Springer, 2006.

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