(article starts on next page)

Variational Bayesian Expectation Maximization For Radar Map Estimation

Malin Lundgren, Lennart Svensson, Lars Hammarstrand

Abstract—For self-localization, a detailed and reliable map of the environment can be used to relate sensor data to static features with known locations. This paper presents a method for construction of detailed radar maps that describe the expected intensity of detections. Specifically, the measurements are modelled by an inhomogeneous Poisson process with a spatial intensity function given by the sum of a constant clutter level and an unnormalized Gaussian mixture. A substantial difficulty with radar mapping is the presence of data association uncertainties, i.e., the unknown associations between measurements and landmarks. In this paper, the association variables are introduced as hidden variables in a variational Bayesian expectation maximization (VBEM) framework, resulting in a computationally efficient mapping algorithm that enables a joint estimation of the number of landmarks and their parameters.

I. INTRODUCTION

Many automotive systems require a good understanding of the environment and the nearby traffic situation. To gain information about the surroundings, the vehicle is equipped with a set of onboard sensors, such as cameras, radars and internal sensors [1]. In recent years, a lot of research focus has shifted towards self-driving vehicles and that has given rise to new problems, which resemble those traditionally studied in robotics. For example, self-driving vehicles require detailed information about the world in order to localize itself and to plan its path. Consequently, there is a need for other types of environment descriptions than provided by traditional road maps.

In robotics, maps are often constructed as a robot navigates in an unknown environment using a set of onboard sensors. This problem is referred to as simultaneous localization and mapping (SLAM) and has received considerable attention in the literature [2], [3]. Examples of SLAM algorithms are EKF-SLAM, FastSLAM [4], LMB-SLAM [5] and Graph-SLAM [6], [7], where the latter has emerged as the algorithm of choice in many contexts. To enable self-driving vehicles in open traffic, the requirements on the localization performance are very strict. One way to obtain the needed accuracy is to construct maps for localization using data collected by sensors mounted on vehicles equipped with reference position systems

M. Lundgren was with the Department of Signals and Systems, Chalmers University of Technology, SE-412 96, Gothenburg, Sweden. She is now with the Active Safety and Chassis department at Volvo Car Corporation, Gothenburg. (E-mail: malin.lundgren@volvocars.com).

L. Svensson and L. Hammarstrand are with the Department of Signals and Systems, Chalmers University of Technology, SE-412 96, Gothenburg, Sweden. (E-mail: {lennart.svensson, lars.hammarstrand}@chalmers.se).

[8], [9]. Having access to accurate reference data, the position uncertainties become negligible and thus, in contrast to a SLAM problem, the focus can be strictly on map construction.

Radar has become a central sensor in many automotive applications, due to its ability to measure the distance, angle and relative speed to surrounding objects, and for its robustness to different weather conditions. The construction of radar maps has been studied in [10] and [11] where the probability hypothesis density (PHD) filter is utilized in order to estimate a map of the static environment. Common for many existing mapping and SLAM algorithms is that they rely on a point target assumption for the landmarks. However, when observing the nearby environment with a radar, many landmarks might give rise to multiple detections each scan and are thus better modelled as extended objects [12]–[16]. From these papers, it is clear that the unknown associations between objects and detections, i.e., the data association uncertainties, is a major challenge.

In this paper, the aim is to estimate a map of the world as seen by a radar, i.e., a map not solely consisting of point landmarks. The focus is on offline estimation and, in contrast to SLAM, we condition on the vehicle trajectory. The proposed radar mapping algorithm makes use of the Variational Bayesian Expectation Maximization (VBEM) algorithm [17] to jointly solve the mapping and the data association problems. Using VBEM we iteratively compute an approximate distribution over the map given the distribution over the data associations, and vice versa. This is fundamentally different compared to common SLAM algorithms that rely on a front-end processing step to estimate the data associations. Also, in contrast to LMB-SLAM [5] and the PHD filter solutions in [10], [11], the proposed algorithm does not rely on recursive approximations, but instead makes use of measurements from all times in order to determine the data association hypotheses and estimate the map parameters.

Each landmark in the map is modelled as an extended object that, similar as in [13], [14], is represented by a weighted Gaussian density. This model makes the mapping problem closely related to that of Gaussian mixture modelling which is a well-studied problem in the literature. For Gaussian mixtures it is common to employ iterative batch algorithms, such as Expectation Maximization (EM) [18] to deal with the data association uncertainties. The same concept is used in the probabilistic multiple hypotheses tracking (PMHT) algorithm [19], [20] which relies on EM to perform object tracking. A clear drawback with EM in these applications is that the number of parameters, i.e., the number of Gaussian components in a mixture or the number of tracked objects,

is assumed to be known. In VBEM on the other hand, the number of mixture components can be set large and, through optimization, the components that are not needed in order to model the observed mixture will get insignificant weights [21]. Thus, VBEM automatically handles the model selection problem.

Inspired by the Gaussian mixture literature, we derive radar mapping solutions using both VBEM and EM, where the latter is mainly used as a reference in the evaluation. Despite the similarities, there are important differences between Gaussian mixture modelling and the studied radar mapping problem. For example, we consider data collected by a sensor with a limited field of view, we get multiple detections at each discrete time instant and there are clutter measurements in the data set. Results on both a simulated example and on real data show that the proposed VBEM mapping algorithm enables joint estimation of the number of landmarks, their parameters and the clutter intensity.

The paper is organized as follows. In Section II, the considered problem is described and the models used for the data and for the map are presented. Section III provides a deeper discussion of the chosen models and the difficulties associated with the mapping problem. It also motivates the choice of methods. The derived solutions using EM and VBEM are presented in Section IV and V, respectively, together with short theoretical backgrounds of the methods. In Section VI, the algorithms are evaluated. Finally, in Section VII, the presented work is summarized together with conclusions.

II. PROBLEM FORMULATION

The objective in this paper is to find a method for estimating a map based on detections provided by radars mounted on a moving vehicle. To limit the problem to map estimation, the sensor pose is assumed known, e.g., given by an accurate reference system. The purpose of the map is not to describe the physical environment, or what kind of objects that are present, but the world as seen by the radar. In other words, we aim to describe a probabilistic radar sensor model as a function of the sensor pose. The remainder of this section describes the studied problem in more detail.

The measurements are modelled by an inhomogeneous Poisson process, which is characterized using a map in a north-east coordinate frame. Consequently, the number of detections at each time instant is given by a Poisson distribution and the location of measurements is described by an intensity function over the observed space. The spatial distribution of a measurement is assumed to be the sum of a constant clutter level and a Gaussian mixture, where each component in the mixture is referred to as a landmark. Based on these assumptions, each landmark in the map is modelled by a weight, ω_j, that corresponds to the expected number of detections generated by landmark j, a mean position $\boldsymbol{\mu}_j = [E_j, N_j]^T$ and a covariance matrix, $\boldsymbol{\Sigma}_j$, that define the distribution of the detections from the landmark. The clutter is described by the intensity parameter ρ, resulting in $\lambda_c = \rho V$ expected number of clutter detections at each scan with a sensor whose observation volume is V. Assuming that there are K

landmarks in the map, and using the above notation, the sought parameter vector $\boldsymbol{\theta}$ is

$$\boldsymbol{\theta} = [\boldsymbol{\mu}_1, ..., \boldsymbol{\mu}_K, \boldsymbol{\Sigma}_1, ..., \boldsymbol{\Sigma}_K, \omega_1, ..., \omega_K, \lambda_c]. \tag{1}$$

It is worth to emphasize that the number of landmarks K, and thus the number of parameters, is unknown and needs to be estimated based on data. A detailed mathematical description of the measurement model (the likelihood function) is given in Section III-A, but in the remainder of this section we elaborate on the underlying model assumptions and introduce some key notation.
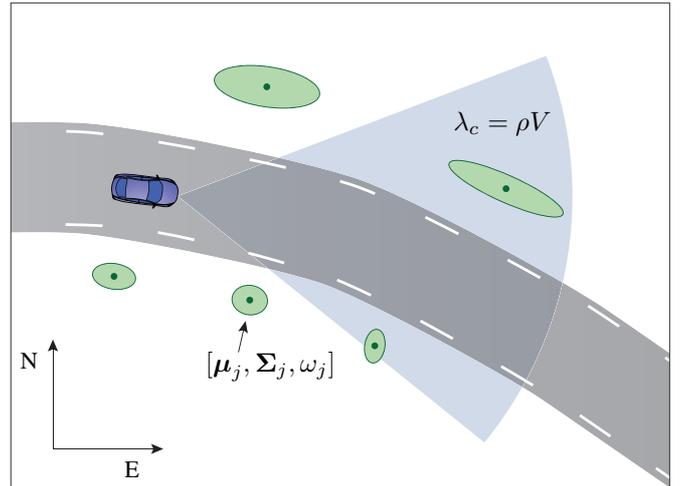
Figure 1: Overview of the considered problem.

What the radar observes depends on its position and heading angle, i.e., its pose, since the radar observations describe the relative distance between the sensor and a landmark. The sensor pose at time k is denoted by

$$\mathbf{x}_k = \begin{bmatrix} \mathbf{x}_k^{\text{p}} \\ x_k^{\varphi} \end{bmatrix}, \tag{2}$$

where $\mathbf{x}_k^{\text{p}} = [x_k^{\text{E}}, x_k^{\text{N}}]^T$ is the position of the sensor in the north-east coordinate frame, and x_k^{φ} is the heading angle. To limit the problem to map building, the position and heading angle of the sensor are assumed to be known at each time instant. The (known) trajectory from time 1 to M is denoted by $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_M]$. The sensor has a field of view described by a known function $f(\boldsymbol{\mu}_j, \mathbf{x}_k)$ which is defined such that the expected number of detections is $\alpha\omega_j$ for a landmark within the field of view, and zero otherwise. Consequently, we describe the field of view as a function of the vehicle position, $\mathbf{x}_k$, and the position of a landmark, $\boldsymbol{\mu}_j$, according to

$$f(\boldsymbol{\mu}_j, \mathbf{x}_k) = \begin{cases} \alpha, & \text{if } \boldsymbol{\mu}_j \text{ is within the field of view} \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

where $0 < \alpha \leq 1$. Similar to, e.g., [14], the field of view function is thus assumed to be constant over each Gaussian.

Each measurement from the environment is given as a range, r, and an angle, ϕ, relative to the radar sensor. A measurement may either originate from a landmark or from clutter and our final objective is to estimate the parameter vector in (1) from

many such observations. Suppose that the i:th measurement at time k is generated by landmark j, the observed range and angle are modelled as

$$\begin{bmatrix} r_k^i \\ \phi_k^i \end{bmatrix} = \begin{bmatrix} \|\boldsymbol{\mu}_j + \mathbf{d}_j - \mathbf{x}_k^{\mathrm{p}}\| \\ \angle \boldsymbol{\mu}_j + \mathbf{d}_j - \mathbf{x}_k^{\mathrm{p}} - x_k^{\varphi} \end{bmatrix} + \mathbf{w}_k, \qquad (4)$$

where $\mathbf{d}_j \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_j)$ captures the extension of the landmark and $\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_w)$ is the measurement noise. Since the sensor pose is known, the measurements can be transformed into the north-east coordinate frame, according to

$$\mathbf{y}_k^i = \begin{bmatrix} y_{k,\mathrm{E}}^i \\ y_{k,\mathrm{N}}^i \end{bmatrix} = \mathbf{x}_k^{\mathrm{p}} + r_k^i \begin{bmatrix} \cos\left(\phi_k^i + x_k^{\varphi}\right) \\ \sin(\phi_k^i + x_k^{\varphi}) \end{bmatrix}. \qquad (5)$$

The complete set of data, collected during M time steps and transformed into global coordinates, is denoted by $\mathbf{Y} = [\mathbf{Y}_1, \mathbf{Y}_2, \ldots, \mathbf{Y}_M]$, where $\mathbf{Y}_k = [y_k^1, \ldots, y_k^{n_k}]$. It is assumed that detections from moving objects can be filtered out due to their velocity such that the considered data set only contains detections from the static environment and clutter. Given the data, $\mathbf{Y}$, the aim in this paper is to estimate the parameters in (1). As discussed in the introduction, the main difficulty concerning this problem is the unknown data associations.

A. Notation

In the paper we use the following notation:

- For brevity, in some equations we drop the dependence of $\boldsymbol{\mu}_j$ and $\mathbf{x}_m$ in the notation for the measurement noise and use $\tilde{\mathbf{R}}_{jm} = \tilde{\mathbf{R}}(\boldsymbol{\mu}_j, \mathbf{x}_m)$.
- $\mathcal{N}(\mathbf{x}; \mathbf{m}, \mathbf{P})$ is the normal (or Gaussian) probability density function of a random vector $\mathbf{x} \in \mathbb{R}^D$ with mean $\mathbf{m}$ and covariance $\mathbf{P}$.
- $\mathcal{IW}(\mathbf{X}; \mathbf{S}, \nu)$ is the Inverse Wishart density function of a random matrix $\mathbf{X} \in \mathbb{S}_{++}^D$, i.e., a positive definite matrix with dimension $D \times D$. The parameters are a scaling matrix $\mathbf{S}$ and the degrees of freedom ν.
- $\mathcal{GAM}(x; a, b)$ is the Gamma probability density function of a scalar variable $x > 0$. The parameters $a > 0$ and $b > 0$ are called shape and rate parameters, respectively.
- $\psi(x)$ denotes the digamma function, defined as the derivative of the logarithm of the Gamma function $\Gamma(x)$.

III. MODEL PROPERTIES AND SOLUTION STRATEGIES

In this section we discuss difficulties associated with the considered problem, and how these have been handled in related work, mainly on the topic of Gaussian mixture models (GMM). More specifically, the focus is on data association uncertainties and how to deal with the unknown number of landmarks. We also present a detailed description of the used models and a derivation of key expressions needed for the solutions presented in the following sections.

When estimating some parameters $\boldsymbol{\theta}$ based on a set of data, $\mathbf{Y}$, a common strategy is to consider some optimality criterion. One aim can be to find the parameters that maximize the likelihood of the data

$$\boldsymbol{\theta}^{\mathrm{ML}} = \arg\max_{\boldsymbol{\theta}} \; \log p(\mathbf{Y}|\boldsymbol{\theta}). \qquad (6)$$

Or, using prior knowledge regarding $\boldsymbol{\theta}$, the aim can be to find the maximum a posteriori (MAP) or the minimum mean squared error (MMSE) estimate of the parameters

$$\boldsymbol{\theta}^{\mathrm{MAP}} = \arg\max_{\boldsymbol{\theta}} \; \log p(\boldsymbol{\theta}|\mathbf{Y}) \qquad (7)$$

$$\boldsymbol{\theta}^{\mathrm{MMSE}} = \mathbb{E}\{p(\boldsymbol{\theta}|\mathbf{Y})\} = \int \boldsymbol{\theta} p(\boldsymbol{\theta}|\mathbf{Y}) d\boldsymbol{\theta} \qquad (8)$$

where the posterior density, $p(\boldsymbol{\theta}|\mathbf{Y}) \propto p(\mathbf{Y}|\boldsymbol{\theta})p(\boldsymbol{\theta})$.

The main difficulty associated with the problem described in Section II, is that there are uncertainties in the data associations. That is, the source of measurements is unknown. The association uncertainty makes direct computation of the likelihood in (6) and thus the posterior density in (7) and (8) very hard. By introducing a variable $\mathbf{Z}$, that defines the data associations, the likelihood can be expressed as

$$p(\mathbf{Y}|\boldsymbol{\theta}) = \sum_{\mathbf{Z}} p(\mathbf{Y}, \mathbf{Z}|\boldsymbol{\theta}) = \sum_{\mathbf{Z}} p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\theta})p(\mathbf{Z}|\boldsymbol{\theta}), \quad (9)$$

where both $p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\theta})$ and $p(\mathbf{Z}|\boldsymbol{\theta})$ can be computed based on the used models. However, even when marginalizing over $\mathbf{Z}$, the likelihood is intractable since the number of association hypotheses grows fast as the number of measurements and potential sources increases.

There are different methods that can be used for handling the unknown data associations in a Gaussian mixture context. One approach is to employ a sampling-based method, such as Gibbs sampling [22], to generate samples from the joint density $p(\boldsymbol{\theta}, \mathbf{Z}|\mathbf{Y})$. Since sampling from the joint density often is hard, a Gibbs sampler constructs samples from the density by multiple draws from conditional distributions. More specifically, suppose that we want to draw N samples from the joint density $p(\tilde{\mathbf{X}})$, where $\tilde{\mathbf{X}} = [\tilde{\mathbf{x}}_1, \ldots, \tilde{\mathbf{x}}_n]$. Then, the i:th sample is denoted $\tilde{\mathbf{X}}^{(i)} = [\tilde{\mathbf{x}}_1^{(i)}, \ldots, \tilde{\mathbf{x}}_n^{(i)}]$, where each $\tilde{\mathbf{x}}_j^{(i)}$ is drawn from the conditional distribution $p(\tilde{\mathbf{x}}_j^{(i)}|\tilde{\mathbf{x}}_1^{(i)}, \ldots, \tilde{\mathbf{x}}_{j-1}^{(i)}, \tilde{\mathbf{x}}_{j+1}^{(i-1)}, \ldots, \tilde{\mathbf{x}}_n^{(i-1)})$. For problems involving a lot of data, i.e., many association variables, the method becomes impractical.

An efficient method that circumvents the difficulties associated with (9) is the EM algorithm. EM is an iterative method that in each iteration finds a point estimate of the parameters and uses that to compute the expectation of the associations. The algorithm utilizes the logarithm of the complete data likelihood, i.e., $\log p(\mathbf{Y}, \mathbf{Z}|\boldsymbol{\theta})$, which has a simple form for many common models. In VBEM, which is a generalized version of EM, both the associations and the parameters are treated as random variables. In order to obtain a tractable solution, the joint density is approximated by a factorization $p(\boldsymbol{\theta}, \mathbf{Z}|\mathbf{Y}) \approx q_{\theta}(\boldsymbol{\theta})q_z(\mathbf{Z})$ and the aim is to find densities $q_{\theta}(\boldsymbol{\theta})$ and $q_z(\mathbf{Z})$ that make the approximation as good as possible [17], [22]–[24]. Thus, in VBEM, the inference problem is transformed into an optimization problem. A major advantage with this formulation, compared to EM, is that the number of components in the Gaussian mixture does not need to be estimated before initialization. Instead, the number of components can be set much larger than the true number and the prior on the mixture weight are chosen such that most of its probability mass is close to zero. Then, through the

optimization, the components that are not needed in order to model the observed mixture will get negligible weights.

In this paper we derive solutions to the mapping problem in Section II using VBEM and EM. The motivation for using VBEM is the ability of the method to simultaneously estimate the number of components in a mixture and their parameters. In addition, the resulting algorithm is computationally efficient and straightforward to implement. The EM mapping algorithm is derived in order to get a reference method and for the possibility to incorporate parts of it into the VBEM solution. The remainder of this section considers model properties and approximations that enable the mapping solutions using EM and VBEM.

A. Factorization of the complete data likelihood

Both EM and VBEM make use of the logarithm of the complete data likelihood, $p(\mathbf{Y}, \mathbf{Z}|\boldsymbol{\theta})$, in order to estimate the parameters of interest. Since the logarithm of a product equals a sum of logarithms, the expression is simplified whenever the likelihood is a product of distributions, each with the property that its logarithm is a simple function. In this section, we show how the likelihood for the model in Section II can be factorized.

To facilitate the derivations, the data associations are described by a matrix $\mathbf{Z}_m$ whose elements z_{ij}^m are 1 if measurement i at time step m originates from component $j \in [1, K]$, and 0 otherwise. The association $z_{i0}^m = 1$ denotes the event that measurement i is clutter. It is worth remembering that the Poisson model assumption allows several measurements to originate from the same landmark but, ignoring possibly unresolved landmarks, a measurement can be generated by at most one landmark. Given data from M independent scans, the likelihood can be factorized according to

$$p(\mathbf{Y}, \mathbf{Z}|\boldsymbol{\theta}, \mathbf{X}) = \prod_{m=1}^{M} p(\mathbf{Y}_m, \mathbf{Z}_m|\boldsymbol{\theta}, \mathbf{x}_m). \qquad (10)$$

Introducing n_m as the number of associations at time step m, i.e., the number of measurements, each factor in (10) can be written as

$$\begin{aligned}
p(\mathbf{Y}_m, &\mathbf{Z}_m|\boldsymbol{\theta}, \mathbf{x}_m) = \\
&p(\mathbf{Y}_m|\mathbf{Z}_m, \boldsymbol{\theta}, \mathbf{x}_m)\Pr\{\mathbf{Z}_m|\boldsymbol{\theta}, \mathbf{x}_m\} = \\
&p(\mathbf{Y}_m|\mathbf{Z}_m, \boldsymbol{\theta}, \mathbf{x}_m)\Pr\{\mathbf{Z}_m, n_m|\boldsymbol{\theta}, \mathbf{x}_m\} = \\
&p(\mathbf{Y}_m|\mathbf{Z}_m, \boldsymbol{\theta}, \mathbf{x}_m)\Pr\{\mathbf{Z}_m|n_m, \boldsymbol{\theta}, \mathbf{x}_m\}\Pr\{n_m|\boldsymbol{\theta}, \mathbf{x}_m\}. \quad (11)
\end{aligned}$$

Since the data is generated from a Poisson process, the probability of receiving n measurements is

$$\Pr\{n|\boldsymbol{\theta}, \mathbf{x}_m\} = \frac{\lambda_m^n e^{-\lambda_m}}{n!}, \qquad (12)$$

where $\lambda_m = \lambda_c + \sum_r f(\boldsymbol{\mu}_r, \mathbf{x}_m)\omega_r$ is the expected number of measurements at time instant m. Using λ_m, we have

$$\Pr\{\mathbf{Z}_m|n_m, \boldsymbol{\theta}, \mathbf{x}_m\} = \prod_{i=1}^{n_m} \left(\frac{\lambda_c}{\lambda_m}\right)^{z_{i0}^m} \prod_{j=1}^{K} \left(\frac{f(\boldsymbol{\mu}_j, \mathbf{x}_m)\omega_j}{\lambda_m}\right)^{z_{ij}^m}. \qquad (13)$$

To express $p(\mathbf{Y}_m|\mathbf{Z}_m, \boldsymbol{\theta}, \mathbf{x}_m)$, we first recall that the available data set, $\mathbf{Y}$, consists of a large number of measurement where the noise contributions depend on the distances and angles between the sensor and the observed features. An approximation of the noise in the global coordinate fame is obtained by the first order Taylor expansion of the nonlinear model in (5), with respect to $\mathbf{d}_j$ and $\mathbf{w}_k$. That is

$$\mathbf{y}_k^i = \boldsymbol{\mu}_j + \mathbf{d}_j + \Gamma(\boldsymbol{\mu}_j, \mathbf{x}_k)\mathbf{w}_k, \qquad (14)$$

where

$$\Gamma(\boldsymbol{\mu}_j, \mathbf{x}_k) = \begin{bmatrix} \cos\left(\angle_{\boldsymbol{\mu}_j - \mathbf{x}_k^{\mathrm{p}}}\right) & -\|\boldsymbol{\mu}_j - \mathbf{x}_k^{\mathrm{p}}\| \sin\left(\angle_{\boldsymbol{\mu}_j - \mathbf{x}_k^{\mathrm{p}}}\right) \\ \sin\left(\angle_{\boldsymbol{\mu}_j - \mathbf{x}_k^{\mathrm{p}}}\right) & \|\boldsymbol{\mu}_j - \mathbf{x}_k^{\mathrm{p}}\| \cos\left(\angle_{\boldsymbol{\mu}_j - \mathbf{x}_k^{\mathrm{p}}}\right) \end{bmatrix}. \qquad (15)$$

This description provides a Gaussian approximation of the measurement noise in the global coordinate frame,

$$\mathbf{y}_k^i \sim \mathcal{N}\left(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j + \tilde{\mathbf{R}}(\boldsymbol{\mu}_j, \mathbf{x}_k)\right) \qquad (16)$$

where $\tilde{\mathbf{R}}(\boldsymbol{\mu}_j, \mathbf{x}_k) = \Gamma(\boldsymbol{\mu}_j, \mathbf{x}_k)\mathbf{R}_\omega \Gamma^T(\boldsymbol{\mu}_j, \mathbf{x}_k)$. Based on (16), we get

$$\begin{aligned}
p(\mathbf{Y}_m&|\mathbf{Z}_m, \boldsymbol{\theta}, \mathbf{x}_m) = \\
&\prod_{i=1}^{n_m} \left(\frac{1}{V}\right)^{z_{i0}^m} \prod_{j=1}^{K} \left(\mathcal{N}\left(\mathbf{y}_m^i; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j + \tilde{\mathbf{R}}(\boldsymbol{\mu}_j, \mathbf{x}_m)\right)\right)^{z_{ij}^m} \quad (17)
\end{aligned}$$

By combining the expressions in (12), (13) and (17), we obtain

$$\begin{aligned}
p(\mathbf{Y}_m, \mathbf{Z}_m|\boldsymbol{\theta}, \mathbf{x}_m) = \frac{1}{n_m!}\exp\left(-\left(\lambda_c + \sum_r f(\boldsymbol{\mu}_r, \mathbf{x}_m)\omega_r\right)\right) \\
\times \prod_{i=1}^{n_m} \left(\frac{\lambda_c}{V}\right)^{z_{i0}^m} \prod_{j=1}^{K} \left(f(\boldsymbol{\mu}_j, \mathbf{x}_m)\omega_j \mathcal{N}(\mathbf{y}_m^i; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j + \tilde{\mathbf{R}}_{jm})\right)^{z_{ij}^m}
\end{aligned}$$
$$(18)$$

where the factor $(\lambda_m)^{n_m} = (\lambda_c + \sum_r f(\boldsymbol{\mu}_r, \mathbf{x}_m)\omega_r)^{n_m}$ was eliminated by noting that z_{ij}^m is 1 exactly n_m times.

This factorized expression will be used as a starting point in the derivations of the EM mapping algorithm in Section IV and the VBEM counterpart in Section V.

B. Prior for the parameters

To compute the posterior distribution of the parameters, we need a prior density $p(\boldsymbol{\theta})$. It is assumed that the parameters are independent before we observe any data, implying that

$$p(\boldsymbol{\theta}) = p(\lambda_c) \prod_{j=1}^{K} p(\omega_j) \prod_{j=1}^{K} p(\boldsymbol{\mu}_j) \prod_{j=1}^{K} p(\boldsymbol{\Sigma}_j), \qquad (19)$$

where the individual priors are specified according to

$$\begin{aligned}
\omega_j &\sim \mathcal{GAM}(a_0, b_0) \\
\boldsymbol{\mu}_j &\sim \mathcal{U} \\
\boldsymbol{\Sigma}_j &\sim \mathcal{IW}(\mathbf{S}_0, \nu_0) \\
\lambda_c &\sim \mathcal{GAM}(c_0, d_0).
\end{aligned}$$

The reason for using a uniform prior over the map is that no information regarding the landmark positions is available a priori we receive the observations. The Inverse-Wishart prior

for $\boldsymbol{\Sigma}_j$ is chosen since it is conjugate to a Gaussian likelihood. Similarly, the choice of a Gamma prior for $\omega_j \geq 0$ and $\lambda_c \geq 0$ is motivated by the fact that it is a conjugate prior to the Poisson distribution that is used to model the number of detections.

C. Approximations

In each iteration of the proposed algorithms, the field of view function, $f(\boldsymbol{\mu}_j, \mathbf{x}_m)$, and the measurement noise covariance, $\tilde{\mathbf{R}}(\boldsymbol{\mu}_j, \mathbf{x}_m)$, are assumed to be constants based on the latest estimate of the landmark means. More specifically, at iteration $t + 1$, it is assumed that $f(\boldsymbol{\mu}_j, \mathbf{x}_m) \approx f(\boldsymbol{\mu}_j^{(t)}, \mathbf{x}_m)$ and that $\tilde{\mathbf{R}}(\boldsymbol{\mu}_j, \mathbf{x}_m) \approx \tilde{\mathbf{R}}(\boldsymbol{\mu}_j^{(t)}, \mathbf{x}_m)$. The argument for this simplification is that both the field of view function and the noise covariance are much less informative regarding the parameters of the observed landmarks than the measurements. In addition, as the algorithm starts to converge, the estimated landmarks move very little in each iteration, implying that $\boldsymbol{\mu}_j^{(t+1)} \approx \boldsymbol{\mu}_j^{(t)}$.

IV. EM FOR ESTIMATION OF A POISSON PROCESS

Often in the EM algorithm, the aim is to find the parameter vector $\boldsymbol{\theta}$ that maximizes the likelihood $p(\mathbf{Y}|\boldsymbol{\theta})$. However, since we have defined a prior for the parameter vector, in this paper EM is employed to find a MAP estimate of $\boldsymbol{\theta}$. Consequently, the objective in this section is to perform the optimization

$$\hat{\boldsymbol{\theta}} = \arg \max_{\boldsymbol{\theta}} \ \log p(\mathbf{Y}|\boldsymbol{\theta}) + \log p(\boldsymbol{\theta}). \tag{20}$$

As discussed in Section III-A, a direct computation of the likelihood, $p(\mathbf{Y}|\boldsymbol{\theta})$, is difficult when the data associations are unknown. Instead, the maximization in EM is performed by iteratively finding a better parameter vector according to

$$\boldsymbol{\theta}^{(t+1)} = \arg \max_{\boldsymbol{\theta}} \ Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) + \log p(\boldsymbol{\theta}), \tag{21}$$

where $\boldsymbol{\theta}^{(t)}$ denotes the parameters from iteration t, and $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ is referred to as the auxiliary function that is maximized in the ML setting of EM. Given the sensor positions $\mathbf{X}$, Q is defined as

$$Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) = \mathbb{E}_Z\{\log p(\mathbf{Y}, \mathbf{Z}|\boldsymbol{\theta}, \mathbf{X})|\mathbf{Y}, \boldsymbol{\theta}^{(t)}, \mathbf{X}\}. \tag{22}$$

In (22), $\mathbf{Z}$ is called the hidden variable (or missing data) which is a variable that, if it was known, would make the problem simple. As stated in Section III, $\mathbf{Z}$ defines the associations between the data and the potential sources of the measurements.

Each iteration of the EM algorithm consists of two steps, called the expectation (E) step and the maximization (M) step. To find a MAP estimate of $\boldsymbol{\theta}$, these steps are performed according to (22) and (21):

E: Determine the expectation of the log-likelihood with respect to $\mathbf{Z}$.

M: Maximize $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) + \log p(\boldsymbol{\theta})$ with respect to $\boldsymbol{\theta}$.

For the model presented in Section II, and further discussed in Section III-A, the auxiliary function is found as the expectation with respect to $\mathbf{Z}$ of the logarithm of (10). That is,

$$\begin{aligned}
Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) &= \mathbb{E}_Z\{\log p(\mathbf{Y}, \mathbf{Z}|\boldsymbol{\theta}, \mathbf{X})|\mathbf{Y}, \boldsymbol{\theta}^{(t)}, \mathbf{X}\} \\
&= -\sum_{m=1}^{M} \lambda_c - \sum_{m=1}^{M}\sum_{j=1}^{K} f(\boldsymbol{\mu}_j, \mathbf{x}_m)\omega_j - \sum_{m=1}^{M} \log(n_m!) \\
&\quad + \sum_{m=1}^{M}\sum_{i=1}^{n_m} \Big[\mathbb{E}_Z\{z_{i0}^m|\mathbf{Y}, \boldsymbol{\theta}^{(t)}, \mathbf{X}\}(\log \lambda_c - \log V) \\
&\quad + \sum_{j=1}^{K} \mathbb{E}_Z\{z_{ij}^m|\mathbf{Y}, \boldsymbol{\theta}^{(t)}, \mathbf{X}\}\Big(\log f(\boldsymbol{\mu}_j, \mathbf{x}_m) \\
&\quad + \log \omega_j + \log \mathcal{N}\big(\mathbf{y}_m^i; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j + \tilde{\mathbf{R}}(\boldsymbol{\mu}_j, \mathbf{x}_m)\big)\Big)\Big]. \tag{23}
\end{aligned}$$

Given the prior in (19),

$$\begin{aligned}
\log p(\boldsymbol{\theta}) &= \log p(\lambda_c) \\
&\quad + \sum_{j=1}^{K} \Big(\log p(\omega_j) + \log p(\boldsymbol{\mu}_j) + \log p(\boldsymbol{\Sigma}_j)\Big). \tag{24}
\end{aligned}$$

In (23) and (24), if not sooner, the benefit of the factorization in Section III-A becomes clear since the expression to work with becomes a sum where the association variables enter linearly. We now present the computations involved in the E and M steps using (23) and (24).

A. E step

In the E step of the algorithm, the expectation of the log-likelihood, with respect to the hidden variables, is computed. Due to the form of the expression for the log-likelihood, where z_{ij}^m enters linearly, it is enough to compute the expected value of each association. More specifically, it is enough to compute

$$\mathbb{E}\{z_{ij}^m|\mathbf{Y}, \boldsymbol{\theta}^{(t)}, \mathbf{X}\} = \Pr\left\{z_{ij}^m = 1|\mathbf{y}_m^i, \boldsymbol{\theta}^{(t)}, \mathbf{x}_m\right\}. \tag{25}$$

For $j \neq 0$ this probability can be expressed as

$$\begin{aligned}
&\Pr\left\{z_{ij}^m = 1|\mathbf{y}_m^i, \boldsymbol{\theta}^{(t)}, \mathbf{x}_m\right\} \propto \\
&\mathcal{N}\big(\mathbf{y}_m^i; \boldsymbol{\mu}_j^{(t)}, \boldsymbol{\Sigma}_j^{(t)} + \tilde{\mathbf{R}}(\boldsymbol{\mu}_j^{(t)}, \mathbf{x}_m)\big) f(\boldsymbol{\mu}_j^{(t)}, \mathbf{x}_m)\omega_j^{(t)} \tag{26}
\end{aligned}$$

where $[\boldsymbol{\mu}_r^{(t)}, \boldsymbol{\Sigma}_r^{(t)}, \omega_r^{(t)}]$ and $\lambda_c^{(t)}$ are given by the parameter vector from the last iteration, $\boldsymbol{\theta}^{(t)}$. Similarly, for $j = 0$, i.e., when the measurement is clutter,

$$\Pr\left\{z_{i0}^m = 1|\mathbf{y}_m^i, \boldsymbol{\theta}^{(t)}, \mathbf{x}_m\right\} \propto \lambda_c^{(t)}/V. \tag{27}$$

The normalization of the these probabilities follows from the assumption that each measurement either originates from a landmark, or is due to clutter. Thus, the association probabilities sum to one according to $\sum_j \Pr\left\{z_{ij}^m = 1|\mathbf{Y}_m, \boldsymbol{\theta}^{(t)}, \mathbf{x}_m\right\} = 1$.

B. M step

In the M step, the aim is to find the parameter vector $\boldsymbol{\theta}^{(t+1)}$ that maximizes the auxiliary function. For ω_j, $\boldsymbol{\mu}_j$ and λ_c, the sought values are obtained using the straightforward approach of setting the first order derivative to zero, i.e.,

$$\frac{\partial}{\partial \boldsymbol{\theta}_j}\Big(Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) + \log p(\boldsymbol{\theta})\Big) = 0 \tag{28}$$

and solve for $\boldsymbol{\theta}_j$. This way, the new parameter values for ω_j, $\boldsymbol{\mu}_j$ and λ_c are given by:

$$\omega_j^{(t+1)} = \frac{(a_0 - 1) + \sum_{m=1}^{M}\sum_{i=1}^{n_M}\mathbb{E}_Z\{z_{ij}^m\}}{b_0 + \sum_{m=1}^{M} f(\boldsymbol{\mu}_j, \mathbf{x}_m)} \tag{29}$$

$$\boldsymbol{\mu}_j^{(t+1)} = \left(\sum_{m=1}^{M}\sum_{i=1}^{n_m} \pi_{ij}^m \Big(\boldsymbol{\Sigma}_j + \tilde{\mathbf{R}}(\boldsymbol{\mu}_j, \mathbf{x}_m)\Big)^{-1}\right)^{-1}$$
$$\times \sum_{m=1}^{M}\sum_{i=1}^{n_m} \pi_{ij}^m \Big(\boldsymbol{\Sigma}_j + \tilde{\mathbf{R}}(\boldsymbol{\mu}_j, \mathbf{x}_m)\Big)^{-1} \mathbf{y}_m^i \tag{30}$$

$$\lambda_c^{(t+1)} = \frac{(c_0 - 1) + \sum_{m=1}^{M}\sum_{i=1}^{n_m}\mathbb{E}_Z\{z_{i0}^m\}}{d_0 + M} \tag{31}$$

where $\mathbb{E}_Z\{z_{ij}^m\}$ is a shorthand notation for the expectation $\mathbb{E}_Z\{z_{ij}^m|\mathbf{Y}, \boldsymbol{\theta}^{(t)}, \mathbf{X}\}$. At this point, it is worth recalling the approximations in Section III-C. These approximations greatly simplify the derivative with respect to $\boldsymbol{\mu}_j$ since they imply that the derivative of $f(\boldsymbol{\mu}_j, \mathbf{x}_m) \approx f(\boldsymbol{\mu}_j^{(t)}, \mathbf{x}_m)$ and of $\hat{\mathbf{R}}(\boldsymbol{\mu}_j, \mathbf{x}_m) \approx \hat{\mathbf{R}}(\boldsymbol{\mu}_j^{(t)}, \mathbf{x}_m)$ are zero.

For the covariances, $\boldsymbol{\Sigma}_j$, the derivative of $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ is of such form that directly solving for $\boldsymbol{\Sigma}_j$ is not possible. Instead we propose that the maximization is performed by a gradient-based optimization approach. To enable a computation of the gradient with respect to $\boldsymbol{\Sigma}_j$, the covariance matrix is parameterized using the Cholesky factorization

$$\boldsymbol{\Sigma}_j = \mathbf{L}_j \mathbf{L}_j^T, \tag{32}$$

where $\mathbf{L}_j$ is a lower triangular matrix. Since $\boldsymbol{\Sigma}_j$ is a 2×2 matrix, $\mathbf{L}_j$ contains 3 parameters that are to be found in the optimization. Gathering the terms in (23) and (24) that depend on $\mathbf{L}_j$, and denoting the expression by $f(\mathbf{L}_j)$, the optimization problem can be formulated as

$$\hat{\mathbf{L}}_j = \arg\max_{\mathbf{L}_j} f(\mathbf{L}_j)$$
$$= \arg\max_{\mathbf{L}_j} \sum_{m=1}^{M}\sum_{i=1}^{n_m} \mathbb{E}\{z_{ij}^m\}\Big(\frac{-1}{2}\log|\mathbf{L}_j\mathbf{L}_j^T + \tilde{\mathbf{R}}(\boldsymbol{\mu}_j, \mathbf{x}_m)|$$
$$- \frac{1}{2}(\mathbf{y}_m^i - \boldsymbol{\mu}_j)^T(\mathbf{L}_j\mathbf{L}_j^T + \tilde{\mathbf{R}}(\boldsymbol{\mu}_j, \mathbf{x}_m))^{-1}(\mathbf{y}_m^i - \boldsymbol{\mu}_j)\Big)$$
$$- \frac{1}{2}\Big[(\nu_0 + D + 1)\log|\mathbf{L}_j\mathbf{L}_j^T| + \mathrm{Tr}\Big(\mathbf{S}_0(\mathbf{L}_j\mathbf{L}_j^T)^{-1}\Big)\Big]. \tag{33}$$

The gradient of $f(\mathbf{L}_j)$, needed in the optimization, is

$$\frac{\partial f(\mathbf{L}_j)}{\partial \mathbf{L}_j} = \sum_{m=1}^{M}\sum_{i=1}^{n_m} \mathbb{E}\{z_{ij}^m\}(\mathbf{L}_j\mathbf{L}_j^T + \tilde{\mathbf{R}}_{jm})^{-1}$$
$$\times [(\mathbf{y}_m^i - \boldsymbol{\mu}_j)(\mathbf{y}_m^i - \boldsymbol{\mu}_j)^T - \mathbf{L}_j\mathbf{L}_j^T - \tilde{\mathbf{R}}_{jm}]$$
$$\times (\mathbf{L}_j\mathbf{L}_j^T + \tilde{\mathbf{R}}_{jm})^{-1}\mathbf{L}_j$$
$$+ (\mathbf{L}_j\mathbf{L}_j^T)^{-1}\Big(\mathbf{S}_0 - (\nu_0 + D + 1)\mathbf{L}_j\mathbf{L}_j^T\Big)(\mathbf{L}_j\mathbf{L}_j^T)^{-1}\mathbf{L}_j. \tag{34}$$

The derivation of (34) is outlined in Appendix A and the resulting EM mapping algorithm is described using pseudocode in Algorithm 1.

Algorithm 1 Mapping solution using EM

Require: Data, $\mathbf{Y} = [\mathbf{Y}_1, \ldots, \mathbf{Y}_M]$
 Sensor poses, $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_M]$
 Number of clusters, K
1: t = 0
2: Initialize $\boldsymbol{\theta}^{(t)}$
3: **while** Not converged **do**
4:   **E step:**
5:   **for** $m = 1 : M$ **do**
6:     **for** $i = 1 : n_m$ **do**
7:       Compute $\mathbb{E}_Z\{z_{ij}^m\} = \Pr\left\{z_{ij}^m = 1|\boldsymbol{\theta}^{(t)}, \mathbf{Y}_m, \mathbf{x}_m\right\}$, for $j = 1 \ldots K$, according to (26)
8:       Compute $\mathbb{E}_Z\{z_{i0}^m\} = \Pr\left\{z_{i0}^m = 1|\boldsymbol{\theta}^{(t)}, \mathbf{Y}_m, \mathbf{x}_m\right\}$, according to (27)
9:     **end for**
10: **end for**
11: **M step:**
12: Compute $\omega_j^{(t+1)}$, for $j = 1 \ldots K$, according to (29)
13: Compute $\boldsymbol{\mu}_j^{(t+1)}$, for $j = 1 \ldots K$, according to (30)
14: Compute $\lambda_c^{(t+1)}$, according to (31)
15: Perform a gradient-based optimization for $\hat{\mathbf{L}}_j$, $j = 1 \ldots K$, using the objective function in (33) and the gradient in (34).
16: Compute $\boldsymbol{\Sigma}_j^{(t+1)} = \hat{\mathbf{L}}_j\hat{\mathbf{L}}_j^T$, for $j = 1 \ldots K$
17: Set t = t+1
18: **end while**

C. Negligible measurement noise

Depending on the environment and the used sensors, the measurement noise might be very small compared to the extensions of the landmarks. In these cases, the measurement noise is negligible and the landmark covariance is assumed to alone describe the spread of the detections. Then, the expression for $\boldsymbol{\mu}_j^{(t+1)}$ simplifies to

$$\boldsymbol{\mu}_j^{(t+1)} = \sum_{m=1}^{M}\sum_{i=1}^{n_m}\mathbb{E}_Z\{z_{ij}^m\}\mathbf{y}_m^i\left(\sum_{m=1}^{M}\sum_{i=1}^{n_m}\mathbb{E}_Z\{z_{ij}^m\}\right)^{-1}. \tag{35}$$

It also makes it possible to find an analytical expression for $\boldsymbol{\Sigma}_j^{(t+1)}$ using the Inverse-Wishart prior for the covariance

matrix:

$$\boldsymbol{\Sigma}_j^{(t+1)} = \left(\mathbf{S}_0 + \sum_{m=1}^{M} \sum_{i=1}^{n_m} \mathbb{E}_Z\{z_{ij}^m\}(\mathbf{y}_m^i - \boldsymbol{\mu}_j)(\mathbf{y}_m^i - \boldsymbol{\mu}_j)^T \right)$$
$$\times \left((\nu_0 + D + 1) + \sum_{m=1}^{M} \sum_{i=1}^{n_m} \mathbb{E}_Z\{z_{ij}^m\} \right)^{-1}. \tag{36}$$

The resulting mapping algorithm is obtained by replacing the computations of $\boldsymbol{\mu}_j^{(t+1)}$ and $\boldsymbol{\Sigma}_j^{(t+1)}$ in Algorithm 1 with the expressions in (35) and (36).

V. VBEM FOR ESTIMATION OF A POISSON PROCESS

Similarly to the EM approach, when applying VBEM to the mapping problem, the unknown data associations need to be considered while estimating the parameters of the map. We employ the mean field approximation [17] and thus assume that the joint posterior density of the parameters, $\boldsymbol{\theta}$, and the associations, $\mathbf{Z}$, can be factorized according to

$$p(\boldsymbol{\theta}, \mathbf{Z}|\mathbf{Y}, \mathbf{X}) \approx q_\theta(\boldsymbol{\theta})q_z(\mathbf{Z}). \tag{37}$$

We then wish to find distributions $q_\theta(\boldsymbol{\theta})$ and $q_z(\mathbf{Z})$ to make the approximation as good as possible in the "exclusive" Kullback-Leibler sense [25], [26]. More specifically, the aim is to minimize

$$\mathrm{KL}(q_\theta(\boldsymbol{\theta})q_z(\mathbf{Z})||p(\boldsymbol{\theta}, \mathbf{Z}|\mathbf{Y}, \mathbf{X})) =$$
$$\int q_\theta(\boldsymbol{\theta})q_z(\mathbf{Z}) \log \frac{q_\theta(\boldsymbol{\theta})q_z(\mathbf{Z})}{p(\boldsymbol{\theta}, \mathbf{Z}|\mathbf{Y}, \mathbf{X})} d\boldsymbol{\theta}d\mathbf{Z}. \tag{38}$$

The optimization is performed iteratively by minimizing (38) with respect to $q_z(\mathbf{Z})$ while keeping $q_\theta(\boldsymbol{\theta})$ fixed, and then repeating the procedure in order to find a new $q_\theta(\boldsymbol{\theta})$ based on the current $q_z(\mathbf{Z})$. One can show that the best fit for the distribution of the associations is [17]

$$q_z(\mathbf{Z}) \propto \exp\left(\int q_\theta(\boldsymbol{\theta}) \log p(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\theta}|\mathbf{X})d\boldsymbol{\theta} \right)$$
$$= \exp\left(\mathbb{E}_\theta\{\log p(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\theta}|\mathbf{X})\} \right). \tag{39}$$

Similarly, the distribution over the parameters can be found as

$$q_\theta(\boldsymbol{\theta}) \propto \exp\left(\int q_z(\mathbf{Z}) \log p(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\theta}|\mathbf{X})d\mathbf{Z} \right)$$
$$= \exp\left(\mathbb{E}_Z\{\log p(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\theta}|\mathbf{X})\} \right). \tag{40}$$

The joint density in (39) and (40) can be expresses as

$$p(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\theta}|\mathbf{X}) = p(\mathbf{Y}, \mathbf{Z}|\boldsymbol{\theta}, \mathbf{X})p(\boldsymbol{\theta}), \tag{41}$$

where $p(\mathbf{Y}, \mathbf{Z}|\boldsymbol{\theta}, \mathbf{X})$ is factorized according to Section III-A. Each iteration of the VBEM algorithm thus consists of the computations of (39) and (40).

A. Approximations

A key property of the VBEM framework is that it provides tractable solutions for models that belong to the conjugate-exponential family [23], [27]. These models satisfy two conditions; the complete data likelihood belongs to the exponential family while the prior for the parameters is conjugate to the likelihood. In this paper, this holds for λ_c and ω_j. However, the expressions for the means and the covariances are more complicated due to the presence of the measurement noise.

Considering a Normal-Inverse-Wishart prior for the landmark mean and covariance

$$p(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j) = \mathcal{NIW}(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j; \mathbf{m}_0, \kappa_0, \mathbf{S}_0, \nu_0)$$
$$\triangleq \mathcal{N}(\boldsymbol{\mu}_j; \mathbf{m}_0, \kappa_0^{-1}\boldsymbol{\Sigma}_j)\mathcal{IW}(\boldsymbol{\Sigma}_j; \mathbf{S}_0, \nu_0)$$
$$\propto |\boldsymbol{\Sigma}_j|^{-(\nu_0+D+2)/2}\exp\left(-\frac{1}{2}\mathrm{Tr}\left(\mathbf{S}_0\boldsymbol{\Sigma}_j^{-1}\right) \right)$$
$$\exp\left(-\frac{\kappa_0}{2}(\boldsymbol{\mu}_j - \mathbf{m}_0)^T\boldsymbol{\Sigma}_j^{-1}(\boldsymbol{\mu}_j - \mathbf{m}_0) \right). \tag{42}$$

The measurement model is Gaussian, but with a varying co-variance $\boldsymbol{\Sigma}_j + \tilde{\mathbf{R}}(\boldsymbol{\mu}_j, \mathbf{x}_m)$, depending on from which position the landmark is observed. Using the measurement model (16), the single-measurement likelihood is

$$p(\mathbf{y}_m^i|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j) = \mathcal{N}\left(\mathbf{y}_m^i; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j + \tilde{\mathbf{R}}(\boldsymbol{\mu}_j, \mathbf{x}_m) \right)$$
$$\propto |\boldsymbol{\Sigma}_j + \tilde{\mathbf{R}}(\boldsymbol{\mu}_j, \mathbf{x}_m)|^{-1/2}$$
$$\times \exp\left(-\frac{1}{2}(\mathbf{y}_m^i - \boldsymbol{\mu}_j)^T\left(\boldsymbol{\Sigma}_j + \tilde{\mathbf{R}}(\boldsymbol{\mu}_j, \mathbf{x}_m)\right)^{-1}(\mathbf{y}_m^i - \boldsymbol{\mu}_j) \right). \tag{43}$$

Hence, if $\tilde{\mathbf{R}}(\boldsymbol{\mu}_j, \mathbf{x}_m) \neq \mathbf{0}$, the resulting posterior density, $p(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j|\mathbf{y}_m^i) \propto p(\mathbf{y}_m^i|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)p(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$, is no longer a Normal-Inverse-Wishart distribution. Instead the functional form of the distribution changes between iterations implying that the model does not belong to the conjugate-exponential family. To overcome this issue, there are approximations that can be used.

In this paper, we employ two strategies for dealing with the non-conjugate-exponential model. First, we constrain the variational distribution of $\boldsymbol{\Sigma}_j$ to $q_\Sigma(\boldsymbol{\Sigma}_j) = \delta(\boldsymbol{\Sigma}_j - \hat{\boldsymbol{\Sigma}}_j)$ implying that the uncertainties in the covariance estimate is ignored and the computation of $\hat{\boldsymbol{\Sigma}}_j$ is performed according to EM [28]. Second, for $\boldsymbol{\mu}_j$ we adopt the Laplace approximation as suggested in [29]. That is, defining $f(\boldsymbol{\mu}_j) = \log q_\mu(\boldsymbol{\mu}_j)$, the second order Taylor expansion of f is

$$f(\boldsymbol{\mu}_j) \approx f(\hat{\boldsymbol{\mu}}_j) + \nabla f(\hat{\boldsymbol{\mu}}_j)(\boldsymbol{\mu}_j - \hat{\boldsymbol{\mu}}_j)$$
$$+ \frac{1}{2}(\boldsymbol{\mu}_j - \hat{\boldsymbol{\mu}}_j)^T\nabla^2 f(\hat{\boldsymbol{\mu}}_j)(\boldsymbol{\mu}_j - \hat{\boldsymbol{\mu}}_j), \tag{44}$$

where $\nabla f(\hat{\boldsymbol{\mu}}_j)$ denotes the gradient of f, evaluated at $\boldsymbol{\mu}_j = \hat{\boldsymbol{\mu}}_j$. Similarly, $\nabla^2 f(\hat{\boldsymbol{\mu}}_j)$ is the Hessian of f. By setting $\hat{\boldsymbol{\mu}}_j$ equal to the MAP estimate of $\boldsymbol{\mu}_j$, $\nabla f(\hat{\boldsymbol{\mu}}) = 0$. Thus,

$$f(\boldsymbol{\mu}_j) \approx f(\hat{\boldsymbol{\mu}}_j) + \frac{1}{2}(\boldsymbol{\mu}_j - \hat{\boldsymbol{\mu}}_j)^T\nabla^2 f(\hat{\boldsymbol{\mu}}_j)(\boldsymbol{\mu}_j - \hat{\boldsymbol{\mu}}_j) \tag{45}$$

and the variational distribution of $\boldsymbol{\mu}_j$ is

$$q_\mu(\boldsymbol{\mu}_j) = e^{f(\boldsymbol{\mu}_j)} \approx \mathcal{N}\left(\boldsymbol{\mu}_j; \mathbf{m}_j, \mathbf{P}_j \right), \tag{46}$$

where the mean and covariance are given by

$$\mathbf{m}_j = \hat{\boldsymbol{\mu}}_j \tag{47}$$
$$\mathbf{P}_j = -[\nabla^2 f(\hat{\boldsymbol{\mu}}_j)]^{-1}. \tag{48}$$

Using this approximation, the resulting variational distribution of the mean turns out to be a Gaussian density even if no Gaussian assumption is made.

B. Resulting algorithm

Each iteration in the VBEM algorithm consists of the computation of $q_z(\mathbf{Z})$ and $q_\theta(\boldsymbol{\theta})$ according to (39) and (40), respectively. In order to compute $q_z(\mathbf{Z})$, we introduce π_{ij}^m as the probability that measurement $\mathbf{y}_m^i$ was generated by landmark $j \neq 0$, or is clutter if $j = 0$. Then, the distribution of $\mathbf{Z}$ can be written as

$$q_z(\mathbf{Z}) = \prod_{m=1}^{M} \prod_{i=1}^{n_m} (\pi_{i0}^m)^{z_{i0}^m} \prod_{j=1}^{K} (\pi_{ij}^m)^{z_{ij}^m}. \tag{49}$$

Since each measurement $\mathbf{y}_m^i$ either originates from one of the landmarks or is a clutter detection, the probabilities π_{ij}^m must satisfy

$$\pi_{i0}^m + \sum_{j=1}^{K} \pi_{ij}^m = 1. \tag{50}$$

To be able to compute $q_z(\mathbf{Z})$ according to (39), a distribution $q_\theta(\boldsymbol{\theta})$ over the parameters is required. Using the approximations in Section V-A, $q_\theta(\boldsymbol{\theta}) = q_\lambda(\lambda_c) \prod_{j=1}^{K} q_\omega(\omega_j) q_\mu(\boldsymbol{\mu}_j) q_\Sigma(\boldsymbol{\Sigma}_j)$ where

$$q_\omega(\omega_j) = \mathcal{GAM}(\omega_j; a_j, b_j) \tag{51}$$
$$q_\mu(\boldsymbol{\mu}_j) = \mathcal{N}(\boldsymbol{\mu}_j; \mathbf{m}_j, \mathbf{P}_j) \tag{52}$$
$$q_\Sigma(\boldsymbol{\Sigma}_j) = \delta(\boldsymbol{\Sigma}_j - \hat{\boldsymbol{\Sigma}}_j) \tag{53}$$
$$q_\lambda(\lambda_c) = \mathcal{GAM}(\lambda_c; c, d). \tag{54}$$

Based on these models, in Appendix B it is shown that the association probabilities are

$$\pi_{i0}^m \propto \exp\big(\psi(c)\big)\big(Vd\big)^{-1} \tag{55}$$

$$\pi_{ij}^m \propto \frac{b_j^{-1}|\hat{\boldsymbol{\Sigma}}_j + \tilde{\mathbf{R}}_{jm}|^{-1/2}}{(2\pi)^{D/2}}$$
$$\times \exp\left(\psi(a_j) - \frac{1}{2}\mathrm{Tr}\big((\hat{\boldsymbol{\Sigma}}_j + \tilde{\mathbf{R}}_{jm})^{-1}\mathbf{P}_j\big)\right)$$
$$\times \exp\left(-\frac{1}{2}(\mathbf{y}_m^i - \mathbf{m}_j)^T\big(\hat{\boldsymbol{\Sigma}}_j + \tilde{\mathbf{R}}_{jm}\big)^{-1}(\mathbf{y}_m^i - \mathbf{m}_j)\right) \tag{56}$$

where, similar as before, D is the dimension of $\boldsymbol{\mu}_j$.

Based on the computed $q_z(\mathbf{Z})$, Appendix B derives $q_\theta(\boldsymbol{\theta})$ based on (40). The variational distributions of the parameters are given by (51) – (54) where the hyperparameters for ω_j and λ_j are updated according to

$$a_j = a_0 + \sum_{m=1}^{M} \sum_{i=1}^{n_m} \pi_{ij}^m \tag{57}$$

$$b_j = b_0 + \sum_{m=1}^{M} f(\boldsymbol{\mu}_j, \mathbf{x}_m) \tag{58}$$

$$c = c_0 + \sum_{m=1}^{M} \sum_{i=1}^{n_m} \pi_{i0}^m \tag{59}$$

$$d = d_0 + M, \tag{60}$$

where a_0 and b_0 are chosen such that $\mathcal{GAM}(\omega_j; a_0, b_0)$ has most its mass close to zero, indicating that the weights are

likely to be very small. The mean and covariance of the distribution $\mathcal{N}(\boldsymbol{\mu}_j; \mathbf{m}_j, \mathbf{P}_j)$ are:

$$\mathbf{m}_j = \left(\sum_{m=1}^{M} \sum_{i=1}^{n_m} \pi_{ij}^m\big(\boldsymbol{\Sigma}_j + \tilde{\mathbf{R}}(\boldsymbol{\mu}_j, \mathbf{x}_m)\big)^{-1}\right)^{-1}$$
$$\times \sum_{m=1}^{M} \sum_{i=1}^{n_m} \pi_{ij}^m\big(\boldsymbol{\Sigma}_j + \tilde{\mathbf{R}}(\boldsymbol{\mu}_j, \mathbf{x}_m)\big)^{-1}\mathbf{y}_m^i \tag{61}$$

$$\mathbf{P}_j = \left(\sum_{m=1}^{M} \sum_{i=1}^{n_m} \pi_{ij}^m\big(\boldsymbol{\Sigma}_j + \tilde{\mathbf{R}}(\boldsymbol{\mu}_j, \mathbf{x}_m)\big)^{-1}\right)^{-1}. \tag{62}$$

It is worth noting that the expression in (61) is identical to that in (30), i.e., the mean in the EM mapping solution. The derived EM algorithm will be useful also for $\boldsymbol{\Sigma}_j$ since the variational distribution of the covariance matrix is constrained to $q_\Sigma(\boldsymbol{\Sigma}_j) = \delta(\boldsymbol{\Sigma}_j - \hat{\boldsymbol{\Sigma}}_j)$. Hence, the MAP estimate $\hat{\boldsymbol{\Sigma}}_j$ can be found using the same gradient-based approach as in (32) – (34).

For clarity, a summary of the resulting mapping algorithm is presented in Algorithm 2. The complexity of the algorithm is bilinear in the number of landmarks, K, and the total number of measurements, $N = \sum_{m=1}^{M} n_m$, i.e., $\mathcal{O}(KN)$. However, an advantage with the method is that a large mapping problem can be divided into several local, and thus smaller, mapping problems, something that will reduce the complexity.

Algorithm 2 Mapping solution using VBEM

Require: Data, $\mathbf{Y} = [\mathbf{Y}_1, \ldots, \mathbf{Y}_M]$
        Sensor poses, $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_M]$
1: Set an upper bound on the number of clusters, K
2: Set priors on the parameters
    • $\omega_j \sim \mathcal{GAM}(\omega_j; a_0, b_0)$, for $j = 1, \ldots, K$
    • $\boldsymbol{\mu}_j \sim \mathcal{N}(\boldsymbol{\mu}_j; \mathbf{m}_0^j, \kappa_0^{-1}\boldsymbol{\Sigma}_j)$, for $j = 1, \ldots, K$
      where $\mathbf{m}_0^j \sim \mathcal{U}$
    • $\boldsymbol{\Sigma}_j \sim \mathcal{IW}(\boldsymbol{\Sigma}_j; \mathbf{S}_0, \nu_0)$, for $j = 1, \ldots, K$
    • $\lambda_c \sim \mathcal{GAM}(\lambda_c; c_0, d_0)$
3: **while** Not converged **do**
4:   **Compute** $q_z(\mathbf{Z})$:
5:   **for** $m = 1 : M$ **do**
6:     **for** $i = 1 : n_m$ **do**
7:       Compute π_{i0}^m according to (55)
8:       Compute π_{ij}^m, for $j = 1 \ldots K$, according to (56)
9:     **end for**
10:   **end for**
11:   Normalize the association probabilities, (50).
12:   **Compute** $q_\theta(\boldsymbol{\theta})$:
13:   Update a_j and b_j, for $j = 1, \ldots, K$,
      according to (57) and (58)
14:   Update c and d, according to (59) and (60)
15:   Update $\mathbf{m}_j$ and $\mathbf{P}_j$, for $j = 1 \ldots K$,
      according to (61) and (62)
16:   Perform a gradient-based optimization for $\hat{\mathbf{L}}_j$, $j = 1 \ldots K$, using the objective function in (33) and the gradient in (34).
17:   Compute $\hat{\boldsymbol{\Sigma}}_j = \hat{\mathbf{L}}_j \hat{\mathbf{L}}_j^T$, for $j = 1 \ldots K$
18: **end while**

C. Negligible measurement noise

As discussed in Section IV-C, there are situations when the measurement noise can be considered negligible. There might also be scenarios where the Laplace and Dirac approximations introduce errors that are considered worse than the achieved benefits. By ignoring the measurement noise, the model belongs to the conjugate-exponential family and, as shown in appendix C, the joint variational distribution of $\boldsymbol{\mu}_j$ and $\boldsymbol{\Sigma}_j$ is Normal-Inverse-Wishart. That is,

$$q_{\mu,\Sigma}(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j) = \mathcal{NIW}(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j; \mathbf{m}_j, \kappa_j, \mathbf{S}_j, \nu_j), \qquad (63)$$

where the hyperparameters are given by

$$\mathbf{m}_j = \sum_{m=1}^{M} \sum_{i=1}^{n_m} \pi_{ij}^m \mathbf{y}_m^i \left(\sum_{m=1}^{M} \sum_{i=1}^{n_m} \pi_{ij}^m \right)^{-1} \qquad (64)$$

$$\kappa_j = \sum_{m=1}^{M} \sum_{i=1}^{n_m} \pi_{ij}^m \qquad (65)$$

$$\mathbf{S}_j = \mathbf{S}_0 + \sum_{m=1}^{M} \sum_{i=1}^{n_m} \pi_{ij}^m \left(\mathbf{y}_m^i (\mathbf{y}_m^i)^T - \mathbf{m}_j \mathbf{m}_j^T \right) \qquad (66)$$

$$\nu_j = \nu_0 + 1 + \sum_{m=1}^{M} \sum_{i=1}^{n_m} \pi_{ij}^m. \qquad (67)$$

With this description of the landmarks, the probabilities for the measurement to landmark associations change and are given as

$$\pi_{ij}^m \propto \frac{b_j^{-1} |\mathbf{S}_j|^{-1/2}}{2(2\pi)^{D/2}}$$
$$\times \exp\left(\psi(a_j)\right) \exp\left(\frac{-1}{2}\nu_j(\mathbf{y}_m^i - \mathbf{m}_j)^T \mathbf{S}_j^{-1}(\mathbf{y}_m^i - \mathbf{m}_j)\right)$$
$$\times \exp\left(\sum_{k=1}^{D} \psi\left(\frac{\nu_j - D - k}{2}\right)\right) \exp\left(\frac{-D}{\kappa_j}\right). \qquad (68)$$

The probability that a measurement is clutter is still given as $\pi_{i0}^m \propto \exp(\psi(c))(Vd)^{-1}$, but it is worth noticing that the normalization changes due to the change in π_{ij}^m.

VI. EVALUATION

In this section, the proposed VBEM mapping algorithm is evaluated on two scenarios. First, we consider a simulated scenario where the sensor observes the environment while traveling two laps on a track. Second, we apply the mapping algorithm on a set of real radar data. As a reference, the performance is compared to that of an iterative extended object PHD mapping algorithm [13], [30] and the EM mapping algorithm derived in Section IV.

A. Implementation details

Here we give a short summary of the chosen settings for the methods in the evaluation.

1) VBEM:
The VBEM algorithm is initiated with K = 300 landmarks. The prior landmark means, $\mathbf{m}_0$, are drawn from a uniform distribution over the map while the other parameters are set to $\mathbf{S}_0 = 10 \cdot \mathbf{I}_{2\times 2}$, $\nu_0 = 5$, $a_0 = 0.1$, $b_0 = 0.2$, $c_0 = 0.05$ and $d_0 = 0.1$. These values result in a priori expectations/variances $\mathbb{E}\{\boldsymbol{\Sigma}_j\} = 5 \cdot \mathbf{I}_{2\times 2}$, $\mathbb{E}\{\omega_j\} = 0.5$, $\text{Var}\{\omega_j\} = 2.5$, $\mathbb{E}\{\lambda_c\} = 0.5$ and $\text{Var}\{\lambda_c\} = 5$. Note that the parameters a_0 and b_0 are chosen such that the weights are likely to be very small.

2) EM:
Since the EM algorithm is not able to estimate the number of landmarks, we assume that some other algorithm provides EM with a perfect estimate of K such that the algorithm can be initiated with K=20, i.e. the true number of landmarks, for the simulated scenario. Clearly, this is an idealized and slightly unrealistic assumption but we argue that it is interesting to compare the VBEM solution with what should be the best possible version of EM. The prior parameter values are set to the same values as for VBEM.

3) PHD:
To consider extended landmarks, we use the PHD algorithm in [13] but constrain the objects to be static. In addition, the prediction of a static map and the effect of the limited field of view in the update step follow the mapping ideas in [30]. The PHD algorithm runs with a known clutter intensity and hence, λ_c does not need to be estimated. The intensity function is represented by a Gaussian mixture where the maximum number of components is set to $N_{\max} = 300$. To keep the number of components below $N_{\max}$, a merge and prune step is included in the algorithm. In the filter, the probability of survival is $P_s = 1$ and the probability of detection is $P_d = 0.95$. The birth intensity is modelled by a Gaussian component with weight $\omega_b = 0.5$ and the mean at the center of the field of view. The extension parameters, including a and b, are set to the same values as for the VBEM and EM algorithms.

B. Simulated data

In the simulated scenario, the sensor travels two laps on the track depicted in Figure 2. There are $K = 20$ landmarks along
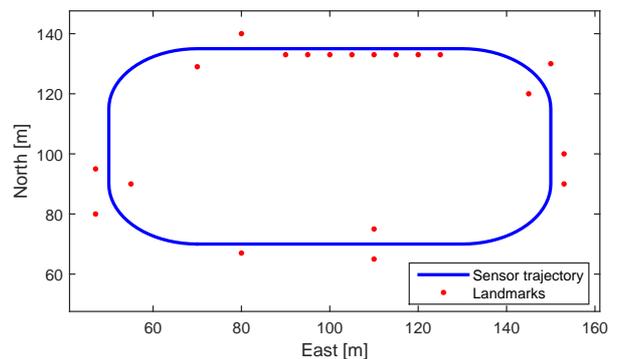
Figure 2: The simulated track and the positions of the landmarks.

the track, each described by a position, a covariance matrix and a weight. In this example, $\omega_j \in [0.5, 2]$. There is also a uniform clutter intensity resulting in $\lambda_c = \rho V$ expected clutter

measurements at each time instant. To start with, $\lambda_c = 2$. The data is generated assuming an ideal field of view with a range of 60 m and angle $\pm$ 30 degrees, such that

$$f(\boldsymbol{\mu}_j, \mathbf{x}_k) = \begin{cases} 1, & \text{if } \boldsymbol{\mu}_j \text{ is within the field of view} \\ 0, & \text{otherwise.} \end{cases} \quad (69)$$

The measurement noise covariance is set to $\mathbf{R}_\omega = \text{diag}([\sigma_r^2, \sigma_\phi^2])$ where $\sigma_r = 0.3$ m and $\sigma_\phi = 3°$.

1) Results: One advantage with VBEM for this problem is the possibility to initiate the algorithm with a number of landmarks that is known to be much larger than the true number and, as seen in Figure 3, most of the weights go towards zero over time. This is a consequence from the optimization, where the Gaussian components not needed to describe the joint density will get negligible weights and thus can be removed from the map. The number of iterations until convergence differs between the solutions, but for illustration, 30 iterations are executed for both EM and VBEM.
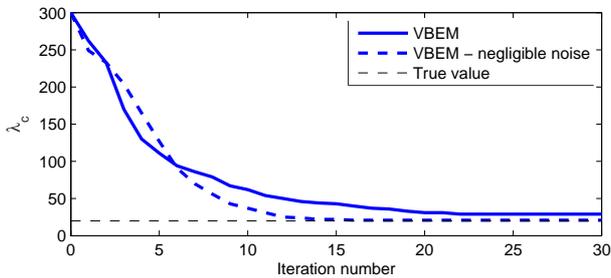
Figure 3: The number of landmarks with a weight $\hat{\omega}_j = a_j/b_j > 0.01$.

To evaluate the ability to estimate all the properties of the landmarks, i.e., position, weight and covariance, we use the squared L2 norm, sometimes also referred to as the integral squared error. For two functions $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$, the squared L2 norm is defined as

$$\|f_1 - f_2\|_2^2 = \int \left(f_1(\mathbf{x}) - f_2(\mathbf{x}) \right)^2 d\mathbf{x}. \quad (70)$$

In [31], it is shown that for Gaussian mixtures, (70) has a simple and exact form that can be used here. In addition, we study how accurately the different methods estimate the clutter intensity. The resulting squared L2 norm and the estimated λ_c are depicted in Figures 4 and 5, respectively.

To see how the estimation of the landmark parameters is affected by the clutter level, we run the algorithms with $\lambda_c = [1, 10, 20, 35, 50]$. The squared L2 norm after 30 iterations, as a function of the increasing clutter intensity, is illustrated in Figure 6.

Comparing the two VBEM solutions, incorporation of the measurement noise results in a better description, in squared L2 sense, of the measurement intensity. However, by neglecting the noise, the estimated number of landmarks is more accurate. Moreover, despite using the same models as VBEM and being initialized with the true number of landmarks, the EM algorithms do not perform as well as their VBEM counterparts. It is worth noting that a considerable weakness with the PHD mapping algorithm is that it may declare a
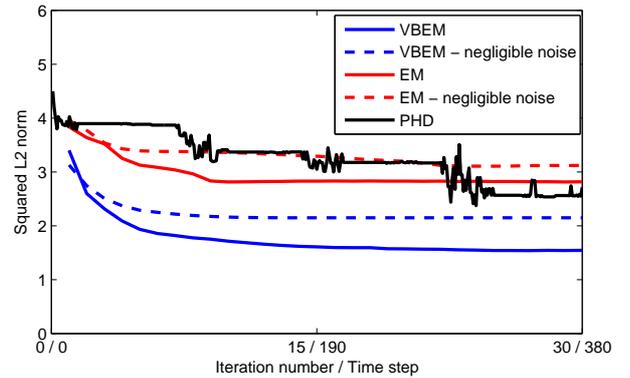
Figure 4: The squared L2 norm as a function of iteration number for EM and VBEM, and as a function of time step for PHD.
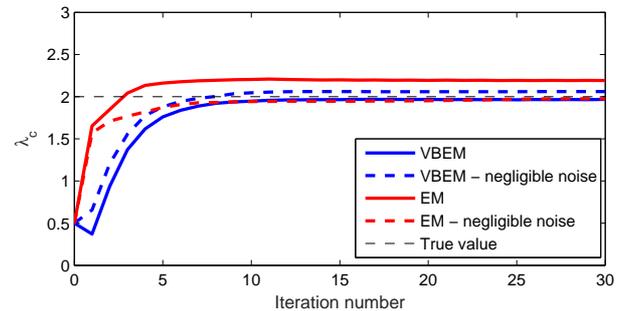
Figure 5: The expected number of clutter detections for the VBEM and EM algorithms. The true $\lambda_c = 2$.

landmark as lost (non-existing) if it is not detected at a few time instances. This is not an issue in the batch solutions where all data is considered in each iteration. Finally, by increasing the clutter intensity, we show that the proposed VBEM method manage to extract the landmarks even in dense clutter.

C. Real radar data

In this section, we apply the VBEM, EM and PHD mapping algorithms to a set of real radar data from a short road segment. Figure 7 shows an intensity plot of the collected data, together with the sensor trajectory, in a north-east coordinate frame.

Since the clutter intensity, needed in the PHD algorithm, and the number of landmarks, needed in the EM algorithm, are unknown when using real data, we use the estimated λ_c and K provided by the VBEM algorithm. In reality, there is no ground truth available showing what a radar map of an area should look like and consequently, when running a mapping algorithm on real data, it is not possible to perform evaluation in the same manner as for the simulated example. Instead, we evaluate the map regarding how well it describes data by computing the log likelihood. The log likelihoods for the three methods are shown in Figure 8, where it is clear that the proposed VBEM algorithm provides the best description of data in this scenario.

For illustration, Figure 9 and 10 show examples of the maps obtained using VBEM and PHD, respectively. Comparing the
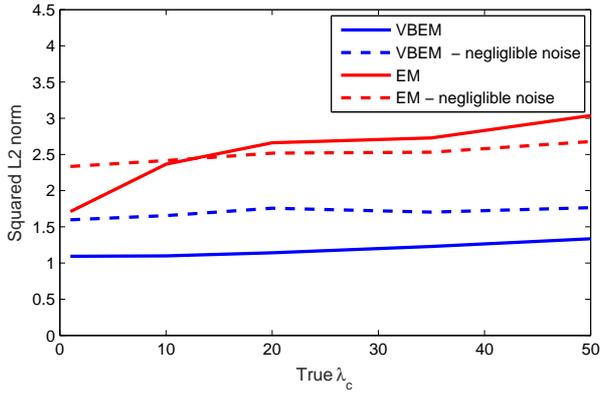
Figure 6: The squared L2 norm after 30 iterations as a function of the expected number of clutter detection at each time instant.
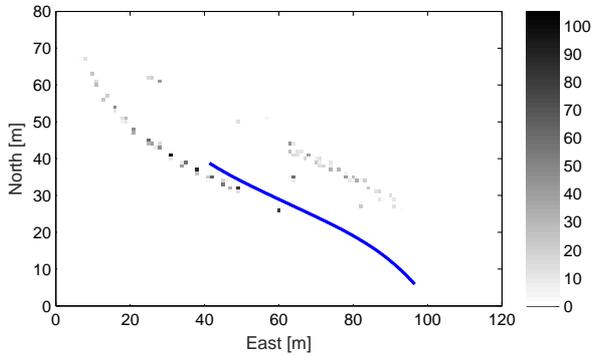
Figure 7: The intensity of radar detections together with the sensor trajectory. The intensity describes the number of detections in the data set within an area of 1 m^2.

figures we note that some landmarks in the VBEM map have been clustered in the PHD map, resulting in a less detailed map.
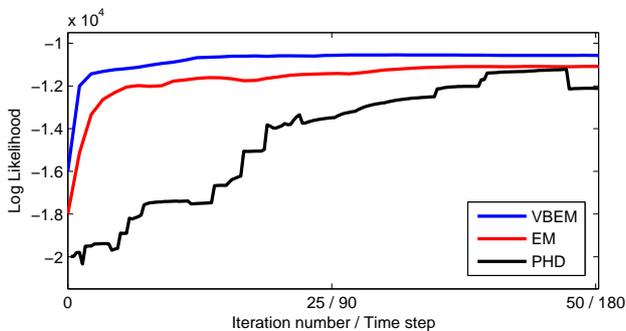
Figure 8: The log likelihood as a function of iteration number for VBEM and EM, and as a function of time step for PHD.
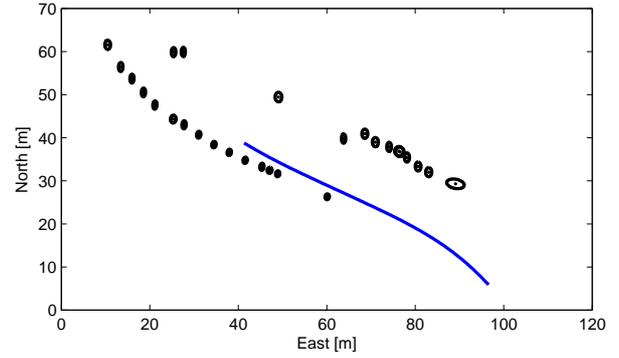
Figure 9: The resulting map using VBEM.
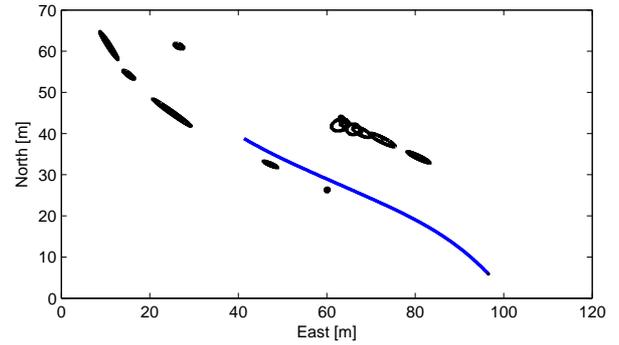
Figure 10: The resulting map using PHD.

VII. CONCLUSIONS

In this paper we present a method for estimating radar maps. The proposed map consists of a constant clutter level and an unknown number of landmarks. Each landmark is modelled by a weighted Gaussian component with a mean position, a covariance matrix that captures the extension and a weight corresponding to the expected number of measurements from the landmark.

We promote a solution that estimates the model parameters using VBEM. The result is an efficient batch solution that enables joint estimation of the number of landmarks, their parameters and the clutter intensity. In the evaluation, the proposed VBEM algorithm is shown to provide a better description of the world compared to the corresponding EM algorithm (with given number of landmarks) and a PHD mapping algorithm (with given clutter intensity).

APPENDIX

In the appendix the derivations of the expressions needed to perform mapping using EM and VBEM are summarized.

A. Derivation of EM solution

This section provides details on the derivations needed for the gradient descent performed to find the landmark covariances in the EM mapping algorithm. The procedure follows from [32].

The gradient of (33) is found by computing the gradients of $\log |\mathbf{L}_j \mathbf{L}_j^T + \tilde{\mathbf{R}}_{jm}|$ and $(\boldsymbol{\mu}_j - \mathbf{y}_m^i)^T (\mathbf{L}_j \mathbf{L}_j^T + \tilde{\mathbf{R}}_{jm})^{-1} (\boldsymbol{\mu}_j - \mathbf{y}_m^i)$. First, we factorize the measurement noise covariance in the same manner as $\boldsymbol{\Sigma}_j$, that is, $\tilde{\mathbf{R}}_{jm} = \mathbf{L}_{R(m)} \mathbf{L}_{R(m)}^T$, and introduce

$$\tilde{\mathbf{X}} = [\mathbf{L}_j \quad \mathbf{L}_{R(m)}]^T, \quad \mathbf{C} = (\mathbf{y}_m^i - \boldsymbol{\mu}_j)(\mathbf{y}_m^i - \boldsymbol{\mu}_j)^T. \quad (71)$$

Given these matrices, and setting $\mathbf{A}$ to the identity matrix, the following relations hold

$$\mathrm{Tr}\left((\tilde{\mathbf{X}}^T \mathbf{A} \tilde{\mathbf{X}})^{-1} \mathbf{C} \right) = (\mathbf{y}_m^i - \boldsymbol{\mu}_j)^T (\boldsymbol{\Sigma}_j + \tilde{\mathbf{R}})^{-1} (\mathbf{y}_m^i - \boldsymbol{\mu}_j)$$
$$\log |\tilde{\mathbf{X}}^T \mathbf{A} \tilde{\mathbf{X}}| = \log |\boldsymbol{\Sigma}_j + \tilde{\mathbf{R}}_{jm}|$$
$$(72)$$

For the left hand sides of (72), the derivatives with respect to the matrix $\mathbf{X}$ can can found in tables. Then, since for a $m \times n$ matrix, the derivative of a scalar is

$$\frac{\partial f(\tilde{\mathbf{X}})}{\partial \tilde{\mathbf{X}}} = \begin{bmatrix} \frac{\partial f}{\partial \tilde{\mathbf{X}}_{11}} & \cdots & \frac{\partial f}{\partial \tilde{\mathbf{X}}_{1n}} \\ \vdots & \ddots & \vdots \\ \frac{\partial f}{\partial \tilde{\mathbf{X}}_{m1}} & \cdots & \frac{\partial f}{\partial \tilde{\mathbf{X}}_{mn}} \end{bmatrix}, \quad (73)$$

the derivatives with respect to $\mathbf{L}_j$ can be found as a part of the derivative with respect to $\tilde{\mathbf{X}}$.

B. Derivation of VBEM solution

This section presents the details of the derivations of the VBEM mapping solution. The derivation starts with the variational distribution $q_z(\mathbf{Z})$ and continues with $q_\theta(\boldsymbol{\theta})$.

1) Computation of $q_z(\mathbf{Z})$: To find the distribution over the association variables, we compute the expectation of $\log p(\boldsymbol{\theta}, \mathbf{Z}, \mathbf{Y})$ with respect to $\boldsymbol{\theta}$. That is,

$$\log q_z(\mathbf{Z}) = \mathbb{E}_{\boldsymbol{\theta}} \left\{ \log p(\boldsymbol{\theta}, \mathbf{Z}, \mathbf{Y}) \right\}$$
$$= \mathbb{E}_{\boldsymbol{\theta}} \left\{ \sum_{m=1}^{M} \log p(\mathbf{Y}_m | \mathbf{Z}_m, \boldsymbol{\theta}) + \log \Pr \{ \mathbf{Z}_m | n_m, \boldsymbol{\theta} \} \right\} + C$$
$$= \sum_{m=1}^{M} \sum_{i=1}^{n_m} z_{i0}^m \left(\mathbb{E}_\lambda \{ \log \lambda_c \} - \log V \right)$$
$$- \sum_{j=1}^{K} z_{ij}^m \left(\frac{D}{2} \log 2\pi + \frac{1}{2} \mathbb{E}_\Sigma \left\{ \log |\boldsymbol{\Sigma}_j + \tilde{\mathbf{R}}| \right\} - \mathbb{E}_\omega \{ \log \omega_j \}$$
$$+ \frac{1}{2} \mathbb{E}_{\mu, \Sigma} \left\{ (\mathbf{y}_m^i - \boldsymbol{\mu}_j)^T (\boldsymbol{\Sigma}_j + \tilde{\mathbf{R}})^{-1} (\mathbf{y}_m^i - \boldsymbol{\mu}_j) \right\} \right) + C$$
$$(74)$$

where all the terms that not include $\mathbf{Z}$ are gathered in the constant C.

To be able to compute the probabilities π_{ij}^m, and thus the distribution of the associations, we need to compute the expectations involved in (74). First, considering that the variational distribution of $\boldsymbol{\Sigma}_j$ is $q_\Sigma(\boldsymbol{\Sigma}_j) = \delta(\boldsymbol{\Sigma}_j - \mathbf{S}_j)$ and that $\int f(x) \delta(x - \hat{x}) dx = f(\hat{x})$:

$$\mathbb{E}_\Sigma \left\{ \log |\boldsymbol{\Sigma}_j + \tilde{\mathbf{R}}(\boldsymbol{\mu}_j, \mathbf{x}_m)| \right\} = \log |\mathbf{S}_j + \tilde{\mathbf{R}}(\boldsymbol{\mu}_j, \mathbf{x}_m)|. \quad (75)$$

The expectation of the logarithm of $\omega \sim \mathcal{GAM}(a, b)$ is [24]

$$\mathbb{E}_\omega \{ \log \omega \} = \psi(a) - \log b, \quad (76)$$

where ψ denotes the digamma function. This expression is applicable to both ω_j and λ_c. Finally, we wish to find the expectation of

$$\mathbb{E}_{\mu, \Sigma} \left\{ (\mathbf{y}_m^i - \boldsymbol{\mu}_j)^T \left(\boldsymbol{\Sigma}_j + \tilde{\mathbf{R}}(\boldsymbol{\mu}_j, \mathbf{x}_m) \right)^{-1} (\mathbf{y}_m^i - \boldsymbol{\mu}_j) \right\}. \quad (77)$$

To perform the derivations, we use the following relations

$$\mathbf{x}^T \mathbf{A} \mathbf{x} = \mathrm{Tr}(\mathbf{x}^T \mathbf{A} \mathbf{x})$$
$$\mathrm{Tr}(\mathbf{x}^T \mathbf{A} \mathbf{x}) = \mathrm{Tr}(\mathbf{A} \mathbf{x} \mathbf{x}^T)$$
$$\mathrm{Tr}(\mathbf{A} + \mathbf{B}) = \mathrm{Tr}(\mathbf{A}) + \mathrm{Tr}(\mathbf{B}).$$

Using these rules, one can rewrite (77) as

$$\mathrm{Tr}\left(\int \left(\hat{\boldsymbol{\Sigma}}_j + \tilde{\mathbf{R}}(\boldsymbol{\mu}_j, \mathbf{x}_m) \right)^{-1} \right.$$
$$\left. \int (\mathbf{y}_m^i - \boldsymbol{\mu}_j)(\mathbf{y}_m^i - \boldsymbol{\mu}_j)^T q(\boldsymbol{\mu}_j) d\boldsymbol{\mu}_j q(\boldsymbol{\Sigma}_j) d\boldsymbol{\Sigma}_j \right)$$
$$(78)$$

Taking a look at the integral over $\boldsymbol{\mu}_j \sim \mathcal{N}(\boldsymbol{\mu}_j; \mathbf{m}_j, \mathbf{P}_j)$:

$$\int (\mathbf{y}_m^i - \boldsymbol{\mu}_j)(\mathbf{y}_m^i - \boldsymbol{\mu}_j)^T q(\boldsymbol{\mu}_j) d\boldsymbol{\mu}_j$$
$$= \mathbb{E}_\mu \left\{ (\mathbf{y}_m^i - \boldsymbol{\mu}_j)(\mathbf{y}_m^i - \boldsymbol{\mu}_j)^T \right\}$$
$$= \mathbb{E}_\mu \left\{ \mathbf{y}_m^i (\mathbf{y}_m^i)^T - \mathbf{y}_m^i \boldsymbol{\mu}_j^T - \boldsymbol{\mu}_j (\mathbf{y}_m^i)^T + \boldsymbol{\mu}_j \boldsymbol{\mu}_j^T \right\}$$
$$= \mathbf{y}_m^i (\mathbf{y}_m^i)^T - \mathbf{y}_m^i \mathbf{m}_j^T - \mathbf{m}_j (\mathbf{y}_m^i)^T + (\mathbf{P}_j + \mathbf{m}_j \mathbf{m}_j^T)$$
$$= (\mathbf{y}_m^i - \mathbf{m}_j)(\mathbf{y}_m^i - \mathbf{m}_j)^T + \mathbf{P}_j$$
$$(79)$$

Inserting (79) into (78), and again making use of that $\int f(x) \delta(x - \hat{x}) dx = f(\hat{x})$, the sought expectation is found as

$$\mathbb{E}_{\mu, \Sigma} \left\{ (\mathbf{y}_m^i - \boldsymbol{\mu}_j)^T \left(\boldsymbol{\Sigma}_j + \tilde{\mathbf{R}}(\boldsymbol{\mu}_j, \mathbf{x}_m) \right)^{-1} (\mathbf{y}_m^i - \boldsymbol{\mu}_j) \right\}$$
$$= \mathrm{Tr}\left(\hat{\boldsymbol{\Sigma}}_j^{-1} \left[(\mathbf{y}_m^i - \mathbf{m}_j)(\mathbf{y}_m^i - \mathbf{m}_j)^T + \mathbf{P}_j \right] \right)$$
$$= (\mathbf{y}_m^i - \mathbf{m}_j)^T \left(\hat{\boldsymbol{\Sigma}}_j + \tilde{\mathbf{R}}(\boldsymbol{\mu}_j, \mathbf{x}_m) \right)^{-1} (\mathbf{y}_m^i - \mathbf{m}_j)$$
$$+ \mathrm{Tr}\left(\hat{\boldsymbol{\Sigma}}_j^{-1} \mathbf{P}_j \right)$$
$$(80)$$

2) Computation of $q_\theta(\boldsymbol{\theta})$: Given $q_z(\mathbf{Z})$, we now want to find the distribution of the parameters $\boldsymbol{\theta}$. Starting from the method of variational EM and including the factorization of the joint density

$$\log q_\theta(\boldsymbol{\theta}) = \mathbb{E}_\mathbf{Z} \left\{ \log p(\boldsymbol{\theta}, \mathbf{Z}, \mathbf{Y}) \right\}$$
$$= \mathbb{E}_\mathbf{Z} \left\{ \sum_{m=1}^{M} \log p(\mathbf{Y}_m | \mathbf{Z}_m, \boldsymbol{\theta}) + \log \Pr \{ \mathbf{Z}_m | n_m, \boldsymbol{\theta} \} + \log p(\boldsymbol{\theta}) \right\}$$
$$= \mathbb{E}_\mathbf{Z} \left\{ \sum_{m=1}^{M} \log p(\mathbf{Y}_m | \mathbf{Z}_m, \boldsymbol{\theta}) + \log \Pr \{ \mathbf{Z}_m | n_m, \boldsymbol{\theta} \} + \log p(\lambda_c) \right.$$
$$\left. + \sum_{j=1}^{K} \log p(\omega_j) + \sum_{j=1}^{K} \log p(\boldsymbol{\mu}_j) + \sum_{j=1}^{K} \log p(\boldsymbol{\Sigma}_j) \right\}. \quad (81)$$

Starting with the clutter parameter, we consider all terms that depend on λ_c:

$$\log q_\lambda(\lambda_c) = \sum_{m=1}^{M}\left(\sum_{i=1}^{n_m}\pi_{i0}^m \log \lambda_c - \lambda_c\right) + \log p(\lambda_c) + C$$
$$= \log \lambda_c \left(c_0 - 1 + \sum_{m=1}^{M}\sum_{i=1}^{n_m}\pi_{i0}^m\right) - (M + d_0)\lambda_c + C. \tag{82}$$

Ignoring the constant term, the density can be found as

$$q_\lambda(\lambda_c) \propto \lambda_c^{\left(c_0 - 1 + \sum_{m=1}^{M}\sum_{i=1}^{n_m}\pi_{i0}^m\right)} e^{-(M+d_0)\lambda_c} \tag{83}$$

which can be identified as a Gamma distribution with parameters according to

$$q_\lambda(\lambda_c) = \mathcal{GAM}\left(\lambda_c; c_0 + \sum_{m=1}^{M}\sum_{i=1}^{n_m}\pi_{i0}^m, d_0 + M\right) \tag{84}$$

The Gamma prior for the weights implies that similar derivations provide us with $q(\omega_j)$.

To find $q_\mu(\boldsymbol{\mu}_j)$ we gather the terms in $\log q_\mu(\boldsymbol{\theta})$ that include $\boldsymbol{\mu}_j$. That is,

$$\log q_\mu(\boldsymbol{\mu}_j) = C + \\
\sum_{m=1}^{M}\sum_{i=1}^{n_m}\pi_{ij}^m\left(-\frac{1}{2}(\mathbf{y}_m^i - \boldsymbol{\mu}_j)^T\left(\boldsymbol{\Sigma}_j + \tilde{\mathbf{R}}_{jm}\right)^{-1}(\mathbf{y}_m^i - \boldsymbol{\mu}_j)\right). \tag{85}$$

Here, the terms where $\boldsymbol{\mu}_j$ enters as a part of the field of view function or the noise covariance are ignored due to the assumptions in Section III-C. According to the Laplace approximation in (46),

$$q_\mu(\boldsymbol{\mu}_j) \approx \mathcal{N}(\boldsymbol{\mu}_j; \hat{\boldsymbol{\mu}}_j, -\nabla^2[\log q(\hat{\boldsymbol{\mu}}_j)]^{-1}) \tag{86}$$

where $\hat{\boldsymbol{\mu}}_j$ is the MAP estimate of $\boldsymbol{\mu}_j$ that is obtained using EM. The covariance $\mathbf{P}_j = -\nabla^2[\log q_\mu(\hat{\boldsymbol{\mu}}_j)]^{-1}$ is found by deriving (85) twice with respect to $\boldsymbol{\mu}_j$.

C. VBEM mapping without noise

Describing the mean and covariance by a Normal-Inverse-Wishart distribution, $\boldsymbol{\mu}_j$ and $\boldsymbol{\Sigma}_j$ are dependent. Consequently, the derivations of $q_{\mu|\Sigma}(\boldsymbol{\mu}_j|\boldsymbol{\Sigma}_j)$ and $q_\Sigma(\boldsymbol{\Sigma}_j)$ start from the joint distribution,

$$\log q_{\mu,\Sigma}(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j) = \sum_{m=1}^{M}\sum_{i=1}^{n_m}\mathbb{E}\{z_{ij}^m\}\left(-\frac{1}{2}\log|\boldsymbol{\Sigma}_j| \right. \tag{87}$$
$$\left. -\frac{1}{2}(\mathbf{y}_m^i - \boldsymbol{\mu}_j)^T\boldsymbol{\Sigma}_j^{-1}(\mathbf{y}_m^i - \boldsymbol{\mu}_j)\right) + \log p(\boldsymbol{\Sigma}_j) + C.$$

Starting with the means, since $\log q_{\mu,\Sigma}(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j) = \log q_{\mu|\Sigma}(\boldsymbol{\mu}_j|\boldsymbol{\Sigma}_j) + \log q_\Sigma(\boldsymbol{\Sigma}_j)$, the distribution $q_{\mu|\Sigma}(\boldsymbol{\mu}_j|\boldsymbol{\Sigma}_j)$ is found by collecting the terms in (87) that depend on $\boldsymbol{\mu}_j$.

$$\log q_{\mu|\Sigma}(\boldsymbol{\mu}_j|\boldsymbol{\Sigma}_j) = \\
= \sum_{m=1}^{M}\sum_{i=1}^{n_m}\mathbb{E}\{z_{ij}^m\}\left(-\frac{1}{2}(\mathbf{y}_m^i - \boldsymbol{\mu}_j)^T\boldsymbol{\Sigma}_j^{-1}(\mathbf{y}_m^i - \boldsymbol{\mu}_j)\right) + C, \tag{88}$$

which can be shown to be

$$q_{\mu|\Sigma}(\boldsymbol{\mu}_j|\boldsymbol{\Sigma}_j) = \mathcal{N}\left(\boldsymbol{\mu}_j; \mathbf{m}_j, \kappa_j^{-1}\boldsymbol{\Sigma}_j\right), \tag{89}$$

where

$$\mathbf{m}_j = \sum_{m=1}^{M}\sum_{i=1}^{n_m}\mathbb{E}\{z_{ij}^m\}\mathbf{y}_m^i\left(\sum_{m=1}^{M}\sum_{i=1}^{n_m}\mathbb{E}\{z_{ij}^m\}\right)^{-1} \tag{90}$$

$$\kappa_j = \sum_{m=1}^{M}\sum_{i=1}^{n_m}\mathbb{E}\{z_{ij}^m\}. \tag{91}$$

For $\boldsymbol{\Sigma}_j$, the variational distribution is found by computing $\log q_\Sigma(\boldsymbol{\Sigma}_j) = \log q_{\mu,\Sigma}(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j) - \log q_{\mu|\Sigma}(\boldsymbol{\mu}_j|\boldsymbol{\Sigma}_j)$ using that $\mathbf{x}^T\mathbf{A}\mathbf{x} = \text{Tr}(\mathbf{x}^T\mathbf{A}\mathbf{x}) = \text{Tr}(\mathbf{x}\mathbf{x}^T\mathbf{A})$ and the expressions in (87) and (89):

$$\log q_\Sigma(\boldsymbol{\Sigma}_j) = \log q_{\mu,\Sigma}(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j) - \log q_{\mu|\Sigma}(\boldsymbol{\mu}_j|\boldsymbol{\Sigma}_j) \\
= \log|\boldsymbol{\Sigma}_j|\left(-\frac{\nu_0 + D + \sum_{m=1}^{M}\sum_{i=1}^{n_m}\mathbb{E}\{z_{ij}^m\}}{2}\right) \\
-\frac{1}{2}\text{Tr}\left(\left(\mathbf{S}_0 - \kappa_j(\boldsymbol{\mu}_j - \mathbf{m}_j)(\boldsymbol{\mu}_j - \mathbf{m}_j)^T \right.\right. \\
\left.\left. + \sum_{m=1}^{M}\sum_{i=1}^{n_m}\mathbb{E}\{z_{ij}^m\}(\mathbf{y}_m^i - \boldsymbol{\mu}_j)(\mathbf{y}_m^i - \boldsymbol{\mu}_j)^T\right)\boldsymbol{\Sigma}_j^{-1}\right) + C, \tag{92}$$

implying that

$$q_\Sigma(\boldsymbol{\Sigma}_j) = \mathcal{IW}\left(\boldsymbol{\Sigma}_j; \mathbf{S}_j, \nu_j\right), \tag{93}$$

with parameters given by

$$\mathbf{S}_j = \mathbf{S}_0 + \sum_{m=1}^{M}\sum_{i=1}^{n_m}\mathbb{E}\{z_{ij}^m\}\left(\mathbf{y}_m^i(\mathbf{y}_m^i)^T - \mathbf{m}_j\mathbf{m}_j^T\right) \tag{94}$$

$$\nu_j = \nu_0 + 1 + \sum_{m=1}^{M}\sum_{i=1}^{n_m}\mathbb{E}\{z_{ij}^m\}. \tag{95}$$

REFERENCES

[1] W. Fleming, "New Automotive Sensors – A Review," *IEEE Sensors Journal*, vol. 8, no. 11, pp. 1900–1921, November 2008.

[2] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: part I," *IEEE Robotics Automation Magazine,*, vol. 13, no. 2, pp. 99–110, June 2006.

[3] T. Bailey and H. Durrant-Whyte, "Simultaneous localization and mapping (SLAM): part II," *IEEE Robotics Automation Magazine,*, vol. 13, no. 3, pp. 108–117, September 2006.

[4] M. Montemerlo, S. Thrun, D. Koller, B. Wegbreit *et al.*, "FastSLAM: A factored solution to the simultaneous localization and mapping problem," in *Proceedings of the AAAI National Conference on Artificial Intelligence*, 2002, pp. 593–598.

[5] H. Deusch, S. Reuter, and K. Dietmayer, "The Labeled Multi-Bernoulli SLAM Filter," *IEEE Signal Processing Letters*, vol. 22, no. 10, pp. 1561–1565, Oct 2015.

[6] S. Thrun and M. Montemerlo, "The Graph SLAM Algorithm With Applications to Large-Scale Mapping of Urban Structures," *The International Journal of Robotics Research*, vol. 25, no. 5-6, pp. 403–429, 2006.

[7] G. Grisetti, R. Kummerle, C. Stachniss, and W. Burgard, "A Tutorial on Graph-Based SLAM," *IEEE Intelligent Transportation Systems Magazine*, vol. 2, no. 4, pp. 31–43, winter 2010.

[8] J. Ziegler *et al.*, "Making Bertha Drive – An Autonomous Journey on a Historic Route," *IEEE Intelligent Transportation Systems Magazine*, vol. 6, no. 2, pp. 8–20, Summer 2014.

[9] M. Lundgren, E. Stenborg, L. Svensson, and L. Hammarstrand, "Vehicle Self-Localization Using Off-the-Shelf Sensors And a Detailed Map," in *IEEE Intelligent Vehicles Symposium Proceedings*, June 2014, pp. 522–528.

[10] M. Adams, J. Mullane, E. Jose, and B.-N. Vo, *Robotic Navigation and Mapping with Radar*. Artech House, 2012.

[11] C. Lundquist, L. Hammarstrand, and F. Gustafsson, "Road Intensity Based Mapping Using Radar Measurements With a Probability Hypothesis Density Filter," *IEEE Transactions on Signal Processing*, vol. 59, no. 4, pp. 1397–1408, April 2011.

[12] J. Koch, "Bayesian Approach to Extended Object and Cluster Tracking Using Random Matrices," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 44, no. 3, pp. 1042–1059, July 2008.

[13] K. Granström, C. Lundquist, and O. Orguner, "Extended Target Tracking using a Gaussian-Mixture PHD Filter," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 4, pp. 3268–3286, October 2012.

[14] C. Lundquist, K. Granström, and U. Orguner, "An Extended Target CPHD Filter and a Gamma Gaussian Inverse Wishart Implementation." *Journal of Selected Topics in Signal Processing*, vol. 7, no. 3, pp. 472–483, 2013.

[15] L. Hammarstrand, M. Lundgren, and L. Svensson, "Adaptive Radar Sensor Model For Tracking Structured Extended Objects," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 3, pp. 1975–1995, 2012.

[16] L. Hammarstrand, L. Svensson, F. Sandblom, and J. Sörstedt, "Extended Object Tracking Using a Radar Resolution Model," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 3, pp. 2371–2386, 2012.

[17] M. J. Beal, "Variational Algorithms For Approximate Bayesian Inference," Ph.D. dissertation, University of London, 2003.

[18] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum Likelihood From Incomplete Data Via the EM Algorithm," *Journal of the Royal Statistical Society, Series B*, vol. 39, no. 1, pp. 1–38, 1977.

[19] R. L. Streit and T. E. Luginbuhl, "Probabilistic Multi-Hypothesis Tracking," NUWC-NPT/10/428, Naval Undersea Warefare Center Division, Tech. Rep., 1995.

[20] M. Wieneke and W. Koch, "Probabilistic Tracking of Multiple Extended Targets Using Random Matrices," in *SPIE Proceedings*, vol. 7698, 2010.

[21] A. Corduneanu and C. M. Bishop, "Variational Bayesian model selection for mixture distributions," in *Artificial intelligence and Statistics*. Morgan Kaufmann Waltham, MA, 2001, pp. 27–34.

[22] K. P. Murphy, *Machine Learning: a Probabilistic Perspective*. MIT press, 2012.

[23] J. M. Winn and C. M. Bishop, "Variational Message Passing," in *Journal of Machine Learning Research*, 2005, pp. 661–694.

[24] C. M. Bishop, *Pattern recognition and machine learning*. Springer, New York, 2006, vol. 1.

[25] S. Kullback and R. A. Leibler, "On Information and Sufficiency," *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[26] T. Minka *et al.*, "Divergence measures and message passing," Technical report, Microsoft Research, Tech. Rep., 2005.

[27] Z. Ghahramani and M. J. Beal, "Propagation Algorithms For Variational Bayesian Learning," *Advances in neural information processing systems*, pp. 507–513, 2001.

[28] P. Liang, S. Petrov, M. I. Jordan, and D. Klein, "The Infinite PCFG Using Hierarchical Dirichlet Processes." in *EMNLP-CoNLL*, 2007, pp. 688–697.

[29] C. Wang and D. M. Blei, "Variational Inference in Nonconjugate Models," *The Journal of Machine Learning Research*, vol. 14, no. 1, pp. 1005–1031, 2013.

[30] J. Mullane, B.-N. Vo, M. Adams, and B.-T. Vo, "A Random-Finite-Set Approach to Bayesian SLAM," *IEEE Transactions on Robotics*, vol. 27, no. 2, pp. 268–282, April 2011.

[31] D. Crouse, P. Willett, K. Pattipati, and L. Svensson, "A look at Gaussian mixture reduction algorithms," in *Proceedings of the 14th International Conference on Information Fusion (FUSION)*, July 2011, pp. 1–8.

[32] J. L. Williams, "Gaussian mixture reduction for tracking multiple maneuvering targets in clutter," Master's thesis, Air Force Institute of Technology, March 2003.

Malin Lundgren received the M.Sc degree in electrical engineering from Chalmers University of Technology, Gothenburg, Sweden, in 2004.

After a couple of years in the automotive industry she returned to Chalmers to work as a Ph.D. student within the areas of filtering, sensor fusion, localization and mapping, primarily for automotive applications.

Since mid 2015, Malin is with the sensor fusion group at Volvo Car Corporation, Gothenburg.

Lennart Svensson Lennart Svensson was born in Älvängen, Sweden in 1976. He received the M.S. degree in electrical engineering in 1999 and the Ph.D. degree in 2004, both from Chalmers University of Technology, Gothenburg, Sweden.

He is currently Associate Professor at the Signal Processing group, again at Chalmers University of Technology. His main research interests include machine learning and Bayesian inference in general, and nonlinear filtering and tracking in particular.

Lars Hammarstrand was born in Landvetter, Sweden in 1979. He received his M.Sc. and Ph.D. degree in electrical engineering from Chalmers University of Technology, Gothenburg, Sweden, in 2004 and 2010, respectively.

Between 2004 and 2011, he was with the Active Safety and Chassis Department at Volvo Car Corporation, Gothenburg, conducting research on tracking and sensor fusion methods for active safety systems. Currently, Lars is a Assistant Professor at the Signal Processing group at Chalmers University of Technology where his main research interests are in the fields of estimation, sensor fusion, self-localization and mapping, especially with application to self-driving vehicles.