Alternative EM Algorithms for Nonlinear State-space Models

Johan Wahlström*, Joakim Jalden‡, Isaac Skog†, Peter Händel‡
*Department of Computer Science, University of Oxford, Oxford, UK
†Department of Electrical Engineering, Linköping University, Linköping, Sweden
‡Department of Information Science and Engineering, KTH Royal Institute of Technology, Stockholm, Sweden
Email: johan.wahlstrom@cs.ox.ac.uk, jalden@kth.se, isaac.skog@liu.se, ph@kth.se

***Abstract*—The expectation-maximization algorithm is a commonly employed tool for system identification. However, for a large set of state-space models, the maximization step cannot be solved analytically. In these situations, a natural remedy is to make use of the expectation-maximization gradient algorithm, i.e., to replace the maximization step by a single iteration of Newton's method. We propose alternative expectation-maximization algorithms that replace the maximization step with a single iteration of some other well-known optimization method. These algorithms parallel the expectation-maximization gradient algorithm while relaxing the assumption of a concave objective function. The benefit of the proposed expectation-maximization algorithms is demonstrated with examples based on standard observation models in tracking and localization.**

***Index Terms*—Expectation-maximization, system identification, the Gauss-Newton method, Levenberg-Marquardt, trust region.**

I. INTRODUCTION

The problem of estimating the parameters of a nonlinear state-space model has received extensive study in the literature, and is of importance for applications within e.g., biomedicine [1], [2], neuroscience [3], and localization [4]. The main challenge of this problem is that the likelihood function generally is intractable. Although particle-based solutions have received plenty of attention [5], [6], they often come with limitations that require extensive workarounds. For example, standard particle-based methods for evaluating the likelihood function tend to be discontinuous in the parameter vector [7]. Similarly, the option of including the parameter in the state vector of a particle filter is problematic since the augmented process does not possess any forgetting property, and hence, the variance of the particle estimates is bound to diverge. While it is to some extent possible to bypass this problem by introducing artificial dynamics for the parameter vector, these methods require a significant amount of tuning [5]. Moreover, particle-based methods are generally not suitable for high-dimensional estimation problems since their performance degrades quickly with the dimension of the problem [8].

An alternative solution to the parameter identification problem is provided by the expectation-maximization (EM) algorithm [9]. The idea of the EM algorithm is to decompose an estimation problem into two steps: the expectation (E) step, which includes finding the posterior distribution of some hidden states given the current parameter estimate; and the maximization (M) step, where the parameter estimate is updated based on the posterior distribution computed in the E step. For many models, each of these two steps is more tractable than the original problem [10]. Generally, the EM algorithm is considered to be more numerically stable than gradient-based techniques, and tends to be favored by practitioners whenever it is applicable [5]. If the M step cannot be solved analytically, a convenient simplification is to perform an approximate maximization based on a single iteration of Newton's method. This is known as the EM gradient algorithm [11].

In this article, we address the problems that arise when the EM gradient algorithm is applied to a state-space model that is not guaranteed to give a concave objective function. Specifically, we propose EM algorithms that replace the standard M step with a single iteration of an alternative optimization method. For nonlinear state-space models with additive Gaussian noise, the M step can be reformulated as a stochastic nonlinear least-squares problem. This can be attacked using Gauss-Newton type methods. In addition, we explore solutions based on trust region and damped methods. To summarize, our main contribution lies in combining the idea of the EM gradient algorithm with optimization methods that are more generally applicable than Newton's method. In this way, we maintain the simplicity of the EM gradient algorithm while circumventing problems related to the shape of the objective function. Two numerical examples are used to benchmark the proposed algorithms against the EM gradient algorithm and standard filtering methods.

II. PROBLEM FORMULATION

Consider the state-space model

$$\mathbf{x}_{k+1} = \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}_k) + \mathbf{w}_k, \tag{1a}$$

$$\mathbf{y}_k = \mathbf{h}_{\boldsymbol{\theta}}(\mathbf{x}_k) + \boldsymbol{\epsilon}_k. \tag{1b}$$

Here, $\mathbf{x}_k \in \mathbb{R}^{N_x}$ denotes the state variable and $\mathbf{y}_k \in \mathbb{R}^{N_y}$ denotes the measurements. The noise processes $\mathbf{w}_k \in \mathbb{R}^{N_x}$ and $\boldsymbol{\epsilon}_k \in \mathbb{R}^{N_y}$ are assumed to be normally distributed and white with positive definite covariances $\mathbf{Q}$ and $\mathbf{R}$, respectively. Furthermore, $\boldsymbol{\theta} \in \mathbb{R}^{N_\theta}$ denotes a vector of unknown parameters that specifies the transition and measurement functions $\mathbf{f}_{\boldsymbol{\theta}}(\cdot)$ and $\mathbf{h}_{\boldsymbol{\theta}}(\cdot)$, respectively, and the initial state is distributed

according to some distribution $p(\mathbf{x}_0)$ that is independent of $\boldsymbol{\theta}$. Throughout the paper, the subindex k is used to denote quantities at sampling instance k. Nonlinear state-space models of the type described by (1) are abundant in engineering and signal processing and have been well-studied in the literature [12]–[16]. This paper is concerned with the problem of estimating the parameter vector $\boldsymbol{\theta}$ from a set of measurements $\mathbf{y}_{1:N} \triangleq \{\mathbf{y}_k\}_{k=1}^N$. The case where $\mathbf{Q}$ and $\mathbf{R}$ are included in the parameter vector $\boldsymbol{\theta}$ is left for future work (in this formulation, the maximization problem does not become a nonlinear least-squares problem when assuming additive Gaussian noise). Next, we briefly review variations of the EM algorithm and describe how these can be used to solve the stated problem. The review will both clarify the relation between the methods proposed in Section III and state-of-the-art EM algorithms, as well as provide the necessary background to the EM algorithm.

A. The Expectation-Maximization Algorithm

The EM algorithm attempts to solve the maximum likelihood problem

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} p_{\boldsymbol{\theta}}(\mathbf{y}_{1:N}) \tag{2}$$

by writing the likelihood as if the missing data $\mathbf{x}_{0:N} \triangleq \{\mathbf{x}_k\}_{k=0}^N$ were available. The log-likelihood of the complete data is then integrated with respect to the posterior distribution $p_{\boldsymbol{\theta}^{(i)}}(\mathbf{x}_{0:N}|\mathbf{y}_{1:N})$, where $\boldsymbol{\theta}^{(i)}$ denotes the current best estimate of $\boldsymbol{\theta}$. Thus, it is possible to alternate between updating the parameter estimate by maximizing the expected log-likelihood of the complete data, and using the new parameter estimate to update the posterior distribution of the missing data [9]. Formally, the E step consists of computing

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(i)}) \triangleq \mathbb{E}[\ln p_{\boldsymbol{\theta}}(\mathbf{x}_{0:N}, \mathbf{y}_{1:N})] \tag{3a}$$

where we let $\mathbb{E}[\cdot]$ denote the expectation with respect to $p_{\boldsymbol{\theta}^{(i)}}(\mathbf{x}_{0:N}|\mathbf{y}_{1:N})$. Similarly, the M step amounts to solving

$$\boldsymbol{\theta}^{(i+1)} = \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(i)}). \tag{3b}$$

These two steps are then repeated until convergence. As shown in [9], each iteration of the EM algorithm is guaranteed to either increase or maintain the likelihood, so that

$$p_{\boldsymbol{\theta}^{(i+1)}}(\mathbf{y}_{1:N}) \geq p_{\boldsymbol{\theta}^{(i)}}(\mathbf{y}_{1:N}). \tag{4}$$

In the remainder of this section, we describe variations of the EM algorithm.

B. Extensions of the Expectation-Maximization Algorithm

There are many examples where either the E step or the M step does not have a closed-form solution [11]. In the former case, we first note that there are several different smoothers that can be used to approximate the posterior $p_{\boldsymbol{\theta}^{(i)}}(\mathbf{x}_{0:N}|\mathbf{y}_{1:N})$ [13]. Likewise, the expectations in $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(i)})$ can be approximated by averaging over samples from the posterior distribution $p_{\boldsymbol{\theta}^{(i)}}(\mathbf{x}_{0:N}|\mathbf{y}_{1:N})$ [17]. This is known as Monte Carlo EM (MCEM). In applications where maximization is cheaper than simulation, the algorithm can be made more efficient by reusing samples from previous iterations, so called stochastic approximation EM (SAEM) [18].

If there is no closed-form solution to the M step, it can be replaced by some method for identifying a parameter estimate $\boldsymbol{\theta}^{(i+1)}$ that satisfies

$$Q(\boldsymbol{\theta}^{(i+1)}, \boldsymbol{\theta}^{(i)}) \geq Q(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i)}). \tag{5}$$

The resulting algorithms are known as generalized EM (GEM) algorithms [9]. An example of a GEM algorithm is the expectation conditional maximization (ECM) algorithm. The ECM algorithm replaces the standard M step with a sequence of conditional maximization steps, each of which maximizes $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(i)})$, but with some vector function of $\boldsymbol{\theta}$ held fixed [19]. The strength of the ECM algorithm is that the conditional maximizations often have analytic solutions or at least are more simple to implement than the original M step [20]. To ensure convergence properties similar to those of the EM algorithm, the constraints are chosen so that the complete maximization resulting from a sequence of conditional maximization steps (performed in between two E steps) is over the full parameter space of $\boldsymbol{\theta}$. This is referred to as the space-filling condition. Although the ECM algorithm typically converges more slowly than the EM algorithm in terms of number of iterations, it can be much faster in total computer time [21, p. 159]. The following modifications of the ECM algorithm can be used to improve its convergence properties:

- The multicycle ECM (MCECM) algorithm adds E steps in between some of the conditional maximization steps [19]. Hence, each iteration is divided into different cycles, where each cycle consists of one E step followed by an ordered set of conditional maximizations.
- The expectation conditional maximization either (ECME) algorithm replaces $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(i)})$ with the actual likelihood function $p_{\boldsymbol{\theta}}(\mathbf{y}_{1:N})$ in some of the maximization steps. In [22], it was concluded that the ECME is nearly always faster than the EM and ECM algorithms in terms of number of iterations. In terms of total computer time, it can be faster by several orders of magnitude.
- The alternating expectation conditional maximization (AECM) algorithm allows the set of hidden states and the constraints to vary within and between iterations [20]. An iteration of the AECM algorithm is considered to consist of the minimal number of cycles (starting after the end of the previous iteration) that are needed to fulfill the space-filling condition. A table that clarifies how the EM, ECM, MCECM, and ECME algorithms can be seen as special cases of the AECM algorithm is provided in [20]. Although an AECM algorithm is not guaranteed to be a GEM algorithm [19], any iteration of an AECM algorithm is guaranteed to either increase or maintain the value of the likelihood function [20].

The drawbacks of the alternative M steps discussed in this subsection is that they are dependent on user design, and can require a significant amount of analytical and implementation effort [23]. Moreover, some of the maximization steps could

still need to be solved numerically, and there are no general guarantees on the relative performance of these algorithms and the standard EM algorithm. We conclude this section by considering a closed-form approximate M step that typically is more straightforward to implement.

C. The Expectation-Maximization Gradient Algorithm

For nonlinear state-space models with additive Gaussian noise, the derivatives of the intermediate function $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(i)})$ are readily available. It is then uncomplicated to apply the EM gradient algorithm, i.e., to replace the M step in the EM algorithm with a single iteration of Newton's method. Specifically, the EM gradient algorithm makes use of the second order Taylor expansion [11]

$$\begin{aligned} Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(i)}) &\approx \widehat{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{(i)}) \\ &\triangleq Q(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i)}) + \partial_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i)})(\boldsymbol{\theta} - \boldsymbol{\theta}^{(i)}) \\ &\quad + \tfrac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\theta}^{(i)})^{\intercal} \partial^2_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i)})(\boldsymbol{\theta} - \boldsymbol{\theta}^{(i)}) \end{aligned} \tag{6}$$

where $\partial_{\boldsymbol{\theta}}$ and $\partial^2_{\boldsymbol{\theta}}$ denote the Jacobian and Hessian with respect to $\boldsymbol{\theta}$, respectively. All derivatives are taken with respect to the first argument in $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(i)})$. The parameter update is then defined as

$$\begin{aligned} \boldsymbol{\theta}^{(i+1)} &= \arg\max_{\boldsymbol{\theta}} \widehat{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{(i)}) \\ &= \boldsymbol{\theta}^{(i)} - (\partial^2_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i)}))^{-1} (\partial_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i)}))^{\intercal}. \end{aligned} \tag{7}$$

The EM gradient algorithm is often favored for computational reasons since it only requires evaluations at $\boldsymbol{\theta} = \boldsymbol{\theta}^{(i)}$ and does not include any line search. At the same time, the EM gradient algorithm has local properties (convergence rate, monotonicity, etc.) that are similar to those of the EM algorithm [11]. A modification of the EM gradient algorithm was proposed in [24], which incorporated the idea of the ECME algorithm by replacing the Hessian of the objective function with a negative definite approximation of the Hessian of the likelihood function.

The Taylor expansion in (6) is typically performed under the assumption that the Hessian $\partial^2_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i)})$ is negative definite [11], [21], [24]. This assumption is not only employed when studying the convergence properties of Newton's method [25], but is also needed to ensure that the parameter updates are in ascent directions. However, as will be shown in Section IV, it is easy to find examples of real-world system identification problems, based on nonlinear state-space models of the type described in (1), where this assumption does not hold and where the Newton updates defined by (7) easily lead to convergence to a local minimum or divergence. Next, we consider several possible alternatives to the parameter update in (7). The resulting algorithms inherit the simplicity of the EM gradient algorithm, while being more suitable to models where the Hessian $\partial^2_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i)})$ is not guaranteed to be negative definite.

III. Alternative Approximate M Steps

This section considers the system identification problem discussed in Section II, and describes how the single iteration of Newton's method that is employed in the EM gradient algorithm can be replaced by a single iteration of some other optimization method. We consider approximate M steps based on the Gauss-Newton algorithm, trust region algorithms, and damped algorithms. The resulting EM algorithms are compatible with any methods for approximating the expectations and posteriors that are needed in the E step.

A. The Gauss-Newton Method

The joint distribution of the state variables and the measurements in the model (1) can be decomposed according to

$$\begin{aligned} \ln p_{\boldsymbol{\theta}}(\mathbf{x}_{0:N}, \mathbf{y}_{1:N}) &= \ln p(\mathbf{x}_0) + \textstyle\sum_{k=1}^{N} \ln p_{\boldsymbol{\theta}}(\mathbf{x}_k | \mathbf{x}_{k-1}) \\ &\quad + \textstyle\sum_{k=1}^{N} \ln p_{\boldsymbol{\theta}}(\mathbf{y}_k | \mathbf{x}_k). \end{aligned} \tag{8}$$

This further means that

$$\begin{aligned} &Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(i)}) \\ &= \textstyle\sum_{k=1}^{N} \mathbb{E}[\mathcal{F}_{\boldsymbol{\theta}}(\mathbf{x}_{k-1}, \mathbf{x}_k)] + \sum_{k=1}^{N} \mathbb{E}[\mathcal{H}_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{y}_k)] \end{aligned} \tag{9}$$

with

$$\begin{aligned} &\mathcal{F}_{\boldsymbol{\theta}}(\mathbf{x}_{k-1}, \mathbf{x}_k) \\ &\triangleq -\tfrac{1}{2}(\mathbf{x}_k - \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}_{k-1}))^{\intercal} \mathbf{Q}^{-1} (\mathbf{x}_k - \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}_{k-1})) \end{aligned} \tag{10a}$$

and

$$\begin{aligned} &\mathcal{H}_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{y}_k) \\ &\triangleq -\tfrac{1}{2}(\mathbf{y}_k - \mathbf{h}_{\boldsymbol{\theta}}(\mathbf{x}_k))^{\intercal} \mathbf{R}^{-1} (\mathbf{y}_k - \mathbf{h}_{\boldsymbol{\theta}}(\mathbf{x}_k)), \end{aligned} \tag{10b}$$

where $\mathbf{Q}$ and $\mathbf{R}$ are the covariance matrices associated with model (1). Hence, (9) illustrates that the M step in (3b) can be cast as a stochastic nonlinear least-squares problem. As a result, the M step can be approximately solved by applying a single iteration of the Gauss-Newton algorithm. The Gauss-Newton algorithm can be seen as making the first order Taylor expansions

$$\mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}) \approx \mathbf{f}_{\boldsymbol{\theta}^{(i)}}(\mathbf{x}) + \partial_{\boldsymbol{\theta}} \mathbf{f}_{\boldsymbol{\theta}^{(i)}}(\mathbf{x})(\boldsymbol{\theta} - \boldsymbol{\theta}^{(i)}), \tag{11a}$$

$$\mathbf{h}_{\boldsymbol{\theta}}(\mathbf{x}) \approx \mathbf{h}_{\boldsymbol{\theta}^{(i)}}(\mathbf{x}) + \partial_{\boldsymbol{\theta}} \mathbf{h}_{\boldsymbol{\theta}^{(i)}}(\mathbf{x})(\boldsymbol{\theta} - \boldsymbol{\theta}^{(i)}), \tag{11b}$$

in $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(i)})$, which gives

$$\begin{aligned} Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(i)}) &\approx \widetilde{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{(i)}) \\ &\triangleq Q(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i)}) + \partial_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i)})(\boldsymbol{\theta} - \boldsymbol{\theta}^{(i)}) \\ &\quad + \tfrac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\theta}^{(i)})^{\intercal} \mathbf{H}(\boldsymbol{\theta}^{(i)})(\boldsymbol{\theta} - \boldsymbol{\theta}^{(i)}) \end{aligned} \tag{12}$$

where

$$\begin{aligned} &\mathbf{H}(\boldsymbol{\theta}^{(i)}) \\ &\triangleq \textstyle\sum_{k=1}^{N} \mathbb{E}[\mathcal{F}^H_{\boldsymbol{\theta}^{(i)}}(\mathbf{x}_{k-1})] + \sum_{k=1}^{N} \mathbb{E}[\mathcal{H}^H_{\boldsymbol{\theta}^{(i)}}(\mathbf{x}_k)], \end{aligned} \tag{13}$$

with

$$\mathcal{F}^H_{\boldsymbol{\theta}^{(i)}}(\mathbf{x}_k) \triangleq -\partial_{\boldsymbol{\theta}} \mathbf{f}_{\boldsymbol{\theta}^{(i)}}(\mathbf{x}_k)^{\intercal} \mathbf{Q}^{-1} \partial_{\boldsymbol{\theta}} \mathbf{f}_{\boldsymbol{\theta}^{(i)}}(\mathbf{x}_k), \tag{14a}$$

$$\mathcal{H}^H_{\boldsymbol{\theta}^{(i)}}(\mathbf{x}_k) \triangleq -\partial_{\boldsymbol{\theta}} \mathbf{h}_{\boldsymbol{\theta}^{(i)}}(\mathbf{x}_k)^{\intercal} \mathbf{R}^{-1} \partial_{\boldsymbol{\theta}} \mathbf{h}_{\boldsymbol{\theta}^{(i)}}(\mathbf{x}_k). \tag{14b}$$

Obviously, (13) and (14) make the assumption that the considered derivatives and expectations can be interchanged. The

Gauss-Newton update is then defined as

$$\begin{aligned}\boldsymbol{\theta}^{(i+1)} &= \arg\max_{\boldsymbol{\theta}} \widetilde{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{(i)}) \\ &= \boldsymbol{\theta}^{(i)} - \mathbf{H}(\boldsymbol{\theta}^{(i)})^{-1}(\partial_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i)}))^{\intercal}.\end{aligned} \tag{15}$$

One of the main benefits of the Gauss-Newton algorithm is that $\mathbf{H}(\boldsymbol{\theta}^{(i)})$ always is negative semidefinite and typically negative definite. Since $\mathbf{Q}$ and $\mathbf{R}$ are assumed to be positive definite, a sufficient condition for the latter is for example that either $\partial_{\boldsymbol{\theta}} \mathbf{f}_{\boldsymbol{\theta}^{(i)}}(\mathbf{x}_k)$ or $\partial_{\boldsymbol{\theta}} \mathbf{h}_{\boldsymbol{\theta}^{(i)}}(\mathbf{x}_k)$ have full rank. In this case, the Gauss-Newton update in (15) is always made in an ascent direction. However, it should be noted that the Gauss-Newton algorithm normally has linear convergence, as opposed to the quadratic convergence of the Newton method [26]. The extension of the parameter update in (15) to the case where $\mathbf{x}_0$ is a Gaussian with mean $\boldsymbol{\mu}_{\boldsymbol{\theta}}$ is straightforward.

When $\mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x})$ and $\mathbf{h}_{\boldsymbol{\theta}}(\mathbf{x})$ are linear in the parameter vector $\boldsymbol{\theta}$, the Gauss-Newton update in (15) is identical to the Newton update in (7). In this case, the maximization step becomes a linear least-squares problem that is solved analytically with one Newton or one Gauss-Newton iteration. Examples of such linear models include linearized inertial navigation systems where the parameters are considered to be the biases of the inertial sensors [27], autoregressive-moving-average (ARMA) processes [28], and the well-studied training model considered in [13]. However, in the general nonlinear case, the difference between the true Hessian employed in the EM gradient algorithm and the approximation used in (15) is

$$\begin{aligned}&\partial^2_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i)}) - \mathbf{H}(\boldsymbol{\theta}^{(i)}) \\ &= \textstyle\sum_{k=1}^{N} \mathbb{E}[\mathcal{F}^{\delta H}_{\boldsymbol{\theta}^{(i)}}(\mathbf{x}_{k-1}, \mathbf{x}_k)] + \sum_{k=1}^{N} \mathbb{E}[\mathcal{H}^{\delta H}_{\boldsymbol{\theta}^{(i)}}(\mathbf{x}_k, \mathbf{y}_k)]\end{aligned} \tag{16}$$

where

$$\begin{aligned}&[\mathcal{F}^{\delta H}_{\boldsymbol{\theta}^{(i)}}(\mathbf{x}_{k-1}, \mathbf{x}_k)]_{n,:} \\ &\triangleq (\mathbf{x}_k - \mathbf{f}_{\boldsymbol{\theta}^{(i)}}(\mathbf{x}_{k-1}))^{\intercal} \mathbf{Q}^{-1} \partial_{[\boldsymbol{\theta}]_n} \partial_{\boldsymbol{\theta}} \mathbf{f}_{\boldsymbol{\theta}^{(i)}}(\mathbf{x}_{k-1}),\end{aligned} \tag{17a}$$

$$\begin{aligned}&[\mathcal{H}^{\delta H}_{\boldsymbol{\theta}^{(i)}}(\mathbf{x}_k, \mathbf{y}_k)]_{n,:} \\ &\triangleq (\mathbf{y}_k - \mathbf{h}_{\boldsymbol{\theta}^{(i)}}(\mathbf{x}_k))^{\intercal} \mathbf{R}^{-1} \partial_{[\boldsymbol{\theta}]_n} \partial_{\boldsymbol{\theta}} \mathbf{h}_{\boldsymbol{\theta}^{(i)}}(\mathbf{x}_k),\end{aligned} \tag{17b}$$

for any $n = 1, \ldots, N_\theta$, with $[\mathbf{A}]_{n,:}$ denoting the nth row of $\mathbf{A}$ and $[\mathbf{a}]_n$ denoting the nth element of $\mathbf{a}$. Furthermore, note that $\mathbb{E}_{p_{\boldsymbol{\theta}}(\mathbf{x}_{0:N}, \mathbf{y}_{1:N})}[\mathbf{x}_k - \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}_{k-1})] = \mathbf{0}_{N_x,1}$ and $\mathbb{E}_{p_{\boldsymbol{\theta}}(\mathbf{x}_{0:N}, \mathbf{y}_{1:N})}[\mathbf{y}_k - \mathbf{h}_{\boldsymbol{\theta}}(\mathbf{x}_k)] = \mathbf{0}_{N_y,1}$ for any $k = 1, \ldots, N$, where $\boldsymbol{\theta}$ is the parameter that was used to generate the state variables and the measurements, and $\mathbf{0}_{\ell_1,\ell_2}$ denotes a zero matrix of dimension $\ell_1 \times \ell_2$. As a result, we would expect that $\partial^2_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i)}) \approx \mathbf{H}(\boldsymbol{\theta}^{(i)})$ when $\boldsymbol{\theta}^{(i)} \approx \boldsymbol{\theta}$, the second derivatives of $\mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}_k)$ and $\mathbf{h}_{\boldsymbol{\theta}}(\mathbf{x}_k)$ are sufficiently small, and N is sufficiently large. In other words, under these conditions, the Hessian $\partial^2_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i)})$ will tend to be negative definite and the EM gradient algorithm will typically be an adequate alternative. Further, whenever all second derivatives of $\mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}_k)$ and $\mathbf{h}_{\boldsymbol{\theta}}(\mathbf{x}_k)$ are independent of $\mathbf{x}_k$, it holds that $\mathbb{E}_{p_{\boldsymbol{\theta}}(\mathbf{y}_{1:N})}[\partial^2_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}, \boldsymbol{\theta}) - \mathbf{H}(\boldsymbol{\theta})] = \mathbf{0}_{N_\theta}$, where $\mathbf{0}_\ell$ denotes a zero matrix of dimension $\ell \times \ell$.

B. Trust Region Methods

The idea of trust region methods is to constrain the updated parameter estimate to be in a region where the approximation of the objective function can be trusted [29]. Applying this idea to the approximations of $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(i)})$ defined in (6) and (12), we obtain

$$\boldsymbol{\theta}^{(i+1)} = \arg\max_{\|\boldsymbol{\theta} - \boldsymbol{\theta}^{(i)}\| \leq d^{(i)}} \widehat{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{(i)}) \tag{18a}$$

and

$$\boldsymbol{\theta}^{(i+1)} = \arg\max_{\|\boldsymbol{\theta} - \boldsymbol{\theta}^{(i)}\| \leq d^{(i)}} \widetilde{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{(i)}), \tag{18b}$$

respectively. Here, $\|\cdot\|$ denotes the Euclidean norm and $d^{(i)}$ is known as the radius of the trust region. As discussed in [30], the solutions to the maximization problems in (18a) and (18b) can be written as

$$\begin{aligned}&\boldsymbol{\theta}^{(i+1)} \\ &= \boldsymbol{\theta}^{(i)} - (\partial^2_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i)}) - \lambda \mathbf{I}_{N_\theta})^{-1} (\partial_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i)}))^{\intercal}\end{aligned} \tag{19a}$$

and

$$\begin{aligned}&\boldsymbol{\theta}^{(i+1)} \\ &= \boldsymbol{\theta}^{(i)} - (\mathbf{H}(\boldsymbol{\theta}^{(i)}) - \lambda \mathbf{I}_{N_\theta})^{-1} (\partial_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i)}))^{\intercal},\end{aligned} \tag{19b}$$

respectively, for some damping parameter $\lambda \geq 0$. Here, $\mathbf{I}_\ell$ denotes the identity matrix of dimension ℓ. Obviously, if the constraint is inactive it holds that $\lambda = 0$ and we recover the parameter updates in (7) and (15).

The radius should be continuously updated based on the fit of the function approximation. For the trust region Newton step in (18a), we first compute the so-called gain ratio

$$\rho(\boldsymbol{\theta}^{(i+1)}, \boldsymbol{\theta}^{(i)}) \triangleq \frac{Q(\boldsymbol{\theta}^{(i+1)}, \boldsymbol{\theta}^{(i)}) - Q(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i)})}{\widehat{Q}(\boldsymbol{\theta}^{(i+1)}, \boldsymbol{\theta}^{(i)}) - \widehat{Q}(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i)})}. \tag{20}$$

The following standard strategy is then applied: If $\rho(\boldsymbol{\theta}^{(i+1)}, \boldsymbol{\theta}^{(i)}) < 0.25$, we set $d^{(i+1)} = d^{(i)}/2$. If $\rho(\boldsymbol{\theta}^{(i+1)}, \boldsymbol{\theta}^{(i)}) > 0.75$, we set $d^{(i+1)} = \max(d^{(i)}, 3\|\boldsymbol{\theta}^{(i+1)} - \boldsymbol{\theta}^{(i)}\|)$ [26]. The analogous updates are made for the trust region Gauss-Newton step in (18b). We emphasize that $Q(\boldsymbol{\theta}^{(i+1)}, \boldsymbol{\theta}^{(i+1)})$ is generally not equal to $Q(\boldsymbol{\theta}^{(i+1)}, \boldsymbol{\theta}^{(i)})$, and hence, the function values needed to compute the gain ratio $\rho(\boldsymbol{\theta}^{(i+1)}, \boldsymbol{\theta}^{(i)})$ cannot be reused in the computation of $\rho(\boldsymbol{\theta}^{(i+2)}, \boldsymbol{\theta}^{(i+1)})$ (as would have been the case when performing multiple trust region iterations with the same objective function).

C. Damped Methods

Instead of setting a strict limit on the distance $\|\boldsymbol{\theta}^{(i+1)} - \boldsymbol{\theta}^{(i)}\|$, damped methods add the corresponding penalty term to the objective function. When replacing the standard M step with a single iteration of the damped Newton method or the damped Gauss-Newton method (also known as the the

TABLE I
RELATION BETWEEN METHODS FOR M STEP.

	Objective function	
Maximization	$\widehat{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{(i)})$ (Newton's method)	$\widetilde{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{(i)})$ (The Gauss-Newton method)
Standard	Section II-C, equation (7)	Section III-A, equation (15)
Trust region	Section III-B, equation (18a)	Section III-B, equation (18b)
Damped	Section III-C, equation (21a)	Section III-C, equation (21b)

Levenberg-Marquardt algorithm), we obtain

$$\begin{aligned}&\boldsymbol{\theta}^{(i+1)}\\ &= \arg\max_{\boldsymbol{\theta}} \widehat{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{(i)}) - \tfrac{1}{2}\lambda^{(i)}\|\boldsymbol{\theta} - \boldsymbol{\theta}^{(i)}\|^2 \qquad (21a)\\ &= \boldsymbol{\theta}^{(i)} - (\partial^2_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i)}) - \lambda^{(i)}\mathbf{I}_{N_\theta})^{-1}(\partial_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i)}))^\intercal,\end{aligned}$$

and

$$\begin{aligned}&\boldsymbol{\theta}^{(i+1)}\\ &= \arg\max_{\boldsymbol{\theta}} \widetilde{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{(i)}) - \tfrac{1}{2}\lambda^{(i)}\|\boldsymbol{\theta} - \boldsymbol{\theta}^{(i)}\|^2 \qquad (21b)\\ &= \boldsymbol{\theta}^{(i)} - (\mathbf{H}(\boldsymbol{\theta}^{(i)}) - \lambda^{(i)}\mathbf{I}_{N_\theta})^{-1}(\partial_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i)}))^\intercal,\end{aligned}$$

respectively.

Generally, the damping parameter $\lambda^{(i)}$ should be decreased as the parameter estimate approaches the solution and more trust can be put in the approximation of $Q(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i)})$ [31]. We will follow the well-established updating scheme proposed in [32] which makes use of a scaling factor $\nu^{(i)}$. Hence, after obtaining a new parameter estimate $\boldsymbol{\theta}^{(i+1)}$ from (21a) or (21b), we compute the corresponding gain ratio $\rho(\boldsymbol{\theta}^{(i+1)}, \boldsymbol{\theta}^{(i)})$. There are then two possible cases. If $\rho(\boldsymbol{\theta}^{(i+1)}, \boldsymbol{\theta}^{(i)}) > 0$, the new parameter estimate $\boldsymbol{\theta}^{(i+1)}$ is accepted, the damping parameter is updated according to $\lambda^{(i+1)} = \lambda^{(i)} \max(1/3, 1 - (2\rho(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i+1)}) - 1)^3)$, and we reinitialize the scaling factor as $\nu_{i+1} = 2$. If $\rho(\boldsymbol{\theta}^{(i+1)}, \boldsymbol{\theta}^{(i)}) \leq 0$, the parameter estimate is rejected, and we instead set $\boldsymbol{\theta}^{(i+1)} = \boldsymbol{\theta}^{(i)}$, $\lambda^{(i+1)} = \nu^{(i)} \cdot \lambda^{(i)}$, and $\nu^{(i+1)} = 2 \cdot \nu^{(i)}$. The initial scale factor is set to $\nu^{(0)} = 2$. Although equations (19) and (21) demonstrate the close relation between trust region and damped methods, there is no simple formula that describes the connection between the trust region radius and the damping parameter that gives the same parameter update [26].

Last, we note that while the M steps involving $\widetilde{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{(i)})$ are dependent on the assumption of additive Gaussian noise, which enables the formulation in (9), the M steps based on $\widehat{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{(i)})$ could be employed also under more general assumptions on the noise terms (assuming that the expressions for $\partial_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i)})$ and $\partial^2_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i)})$ are modified accordingly). The relation between the M step in the EM gradient algorithm and the five alternative M steps proposed in this section is summarized in Table I.

IV. EXAMPLES

To evaluate the efficiency of the discussed M steps, this section considers two models. Both have measurement functions that are highly nonlinear in the parameter vector, and hence, they do not permit closed-form solutions to the M step in the standard EM algorithm. EM algorithms based on the six different algorithms in Table I are compared with estimators based on extended Kalman filters (EKF) and unscented Kalman filters (UKF) [33] where the state vector is extended with the sought parameter vector. Neither the filters nor the EM algorithms use any a priori knowledge of $\boldsymbol{\theta}$ (this can be incorporated into the EM algorithms as described in [21, p. 26]). We begin by giving some details on the implementation.

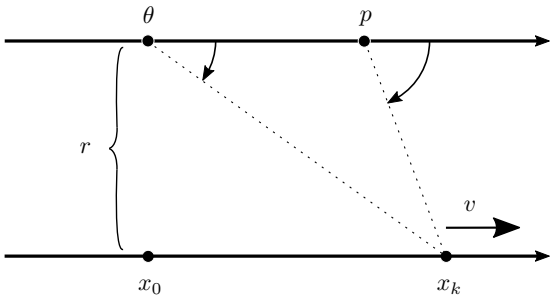

Fig. 1. The bearings-only tracking scenario with the target at x_k and the tracking platforms at the unknown and known positions θ and p, respectively.

A. Implementation Details

In the E step, we chose to approximate the posteriors $p_{\boldsymbol{\theta}^{(i)}}(\mathbf{x}_k | \mathbf{y}_{1:N})$ and $p_{\boldsymbol{\theta}^{(i)}}(\mathbf{x}_k, \mathbf{x}_{k+1} | \mathbf{y}_{1:N})$ by means of an extended Kalman smoother [34]. The expectations in $\partial_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i)})$, $\partial^2_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i)})$, and $\mathbf{H}(\boldsymbol{\theta}^{(i)})$ were approximated by applying the cubature transform [35], and using the means and covariances given by the smoother. All in all, this meant that we could perform a complete EM iteration without the need to resort to Monte Carlo methods.

The EM algorithms were iterated until $\|\boldsymbol{\theta}^{(i+1)} - \boldsymbol{\theta}^{(i)}\| < 10^{-4}$, at which point the final estimate was $\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}^{(i+1)}$. An EM algorithm was considered to have diverged when reaching its hundredth iteration or when $\|\boldsymbol{\theta}\| > 10^3$. Runs leading to divergence were excluded from computations of root-mean-square errors (RMSEs). Further, the radius and the damping parameter were initialized as $d^{(0)} = 0.2$ and $\lambda^{(0)} = 1$, respectively, and each marker in the studied plots was obtained as the result of 10^4 simulations of the full sequence of state vectors and measurements.

B. Bearings-Only Tracking

Consider the setting of a bearings-only tracking problem where a target is traveling in a known direction in two dimensions with the mean speed v. The bearing of the target is measured relative to two stationary tracking platforms positioned along a line that is parallel to the path of the target with a perpendicular distance of r. One of the platforms is known to be located at p as measured along the line, and the other is located at the unknown position θ. Denoting the position of the target by x, the transition and measurement functions can be written as [35]

$$f(x_k) = x_k + v, \tag{22a}$$

$$\mathbf{h}_\theta(x_k) = \begin{bmatrix} \arctan2(r, x_k - p) \\ \arctan2(r, x_k - \theta) \end{bmatrix}. \tag{22b}$$

For the simulations, we let $r = 1$, $p = 1$, $\theta = 0$, $v = 0.5$, $\mathbf{Q} = (0.1)^2$, $\mathbf{R} = \sigma_r^2 \mathbf{I}_2$, and $N = 15$. Further, the initial state was $x_0 = 0$ with an initial uncertainty of $P_0 = 0$, and the initial estimate of θ was simulated from a uniform distribution over $(\theta - \delta\theta, \theta + \delta\theta)$ where $\delta\theta = 7.5$. The scenario is illustrated in Fig. 1.

Fig. 2 (a) now displays the RMSEs obtained when estimating θ from the set of measurements $\mathbf{y}_{1:N}$ while varying σ_r. Here, the EM algorithms have been abbreviated based on whether the M step was implemented as a single iteration of Newton's method (N), the Gauss-Newton method (GN), the trust region Newton method (TRN), the trust region Gauss-Newton method (TRGN), the damped Newton method (DN), or the damped Gauss-Newton method (DGN). As can be seen from Fig. 2 (a), neither the EKF nor the UKF can manage the nonlinearities of the system, while all the EM algorithms display practically equivalent RMSEs. However, Fig. 2 (b) reveals that the EM gradient algorithm and the EM algorithm using damped Newton steps diverge in approximately 70 [%] of the runs at small measurement errors, and we also see some divergence when using Gauss-Newton steps. No divergence is experienced with the EM algorithms employing trust region methods or the EM algorithm using damped Gauss-Newton steps.

C. Log-distance Path Loss Model

In our second example, we consider a two-dimensional localization problem with a receiver positioned at $\mathbf{x}_k$, two transmitters at the known positions $\mathbf{p}_1$ and $\mathbf{p}_2$, and one transmitter at the unknown position $\boldsymbol{\theta}$. Further, we assume the availability of received signal strength (RSS) measurements that can be modeled according to the log-distance path loss model. Assuming that the receiver has the mean speed $\mathbf{v}$, the corresponding transition and measurement functions can be written as [36], [37]

$$\mathbf{f}(\mathbf{x}_k) = \mathbf{x}_k + \mathbf{v}, \tag{23a}$$

$$\mathbf{h}_{\boldsymbol{\theta}}(\mathbf{x}_k) = -c \cdot 10 \begin{bmatrix} \log_{10}(\|\mathbf{x}_k - \mathbf{p}_1\|^2) \\ \log_{10}(\|\mathbf{x}_k - \mathbf{p}_2\|^2) \\ \log_{10}(\|\mathbf{x}_k - \boldsymbol{\theta}\|^2) \end{bmatrix}. \tag{23b}$$

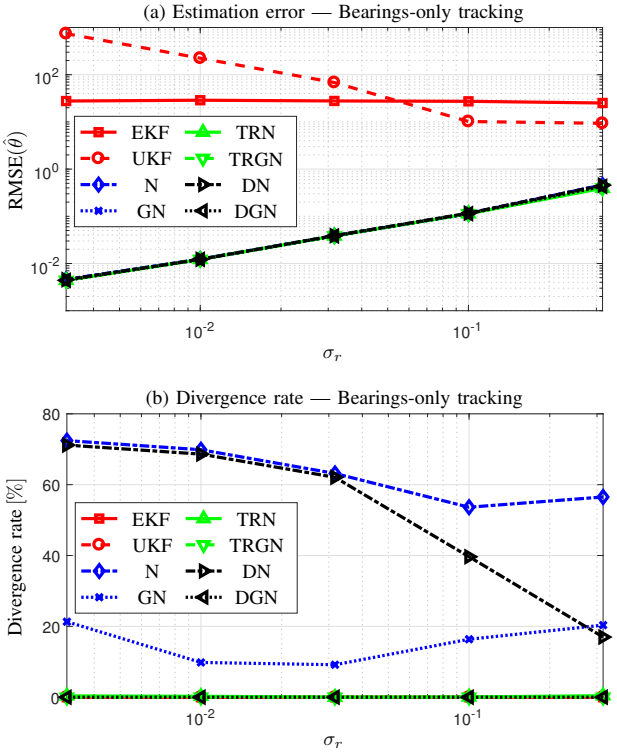

Fig. 2. (a) Errors and (b) divergence rates of the system identification algorithms in the bearings-only tracking scenario.

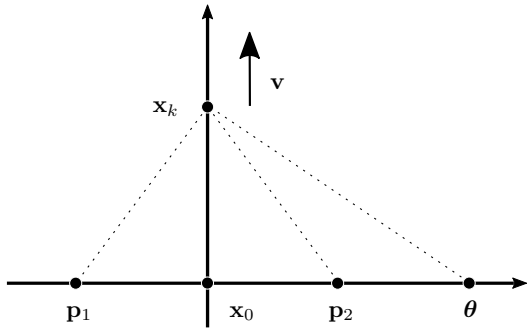

Fig. 3. The localization problem using the log-distance path loss model with a receiver positioned at $\mathbf{x}_k$, two transmitters at the known positions $\mathbf{p}_1$ and $\mathbf{p}_2$, and one transmitter at the unknown position $\boldsymbol{\theta}$.

For the simulations, we used $c = 1/10 \cdot \ln 10$, $\mathbf{p}_1 = [-1\ 0]^\intercal$, $\mathbf{p}_2 = [1\ 0]^\intercal$, $\boldsymbol{\theta} = [2\ 0]^\intercal$, $\mathbf{v} = [0\ 0.5]^\intercal$, $\mathbf{Q} = (0.1)^2\,\mathbf{I}_2$, $\mathbf{R} = \sigma_r^2\,\mathbf{I}_3$, and $N = 15$. Further, the initial state was $\mathbf{x}_0 = [0\ 0]^\intercal$ with an initial uncertainty of $\mathbf{P}_0 = \mathbf{0}_2$, and the initial estimate of $\boldsymbol{\theta}$ was simulated from a uniform distribution over $([\boldsymbol{\theta}]_1 - \delta\theta, [\boldsymbol{\theta}]_1 + \delta\theta) \times ([\boldsymbol{\theta}]_2 - \delta\theta, [\boldsymbol{\theta}]_2 + \delta\theta)$ where $\delta\theta = 1$. The scenario is illustrated in Fig. 3.

The RMSEs associated with $[\boldsymbol{\theta}]_1$ and $[\boldsymbol{\theta}]_2$ are displayed in Figs. 4 (a) and (b), respectively. All the alternative M steps can be seen to lead to lower RMSEs than the standard EM gradient algorithm using Newton updates. While the RMSE is only modestly decreased with damped Newton updates, the performance gain is substantial with both trust region updates and Gauss-Newton updates. At the lowest noise levels, the

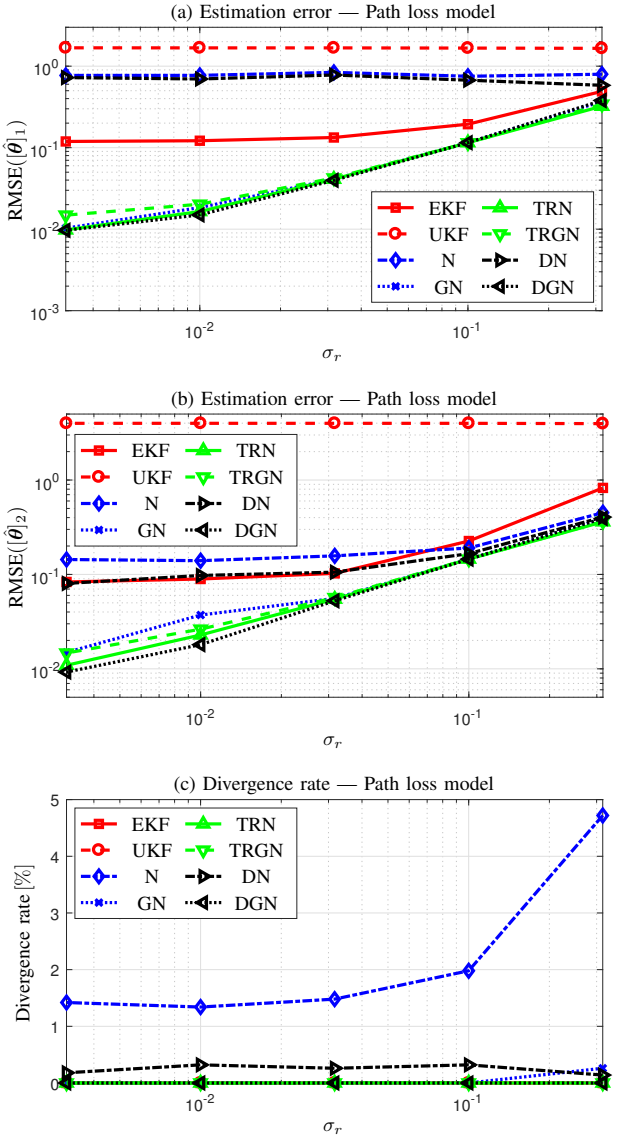

Fig. 4. (a), (b) Errors and (c) divergence rates of the system identification algorithms in the path loss scenario.

best performance is achieved with M steps based on a damped Gauss-Newton step, which can be seen to give an RMSE that is about one to two order of magnitudes smaller than that of the original EM gradient algorithm. The poor performance of the UKF can be improved by incorporating a prior on $\boldsymbol{\theta}$, thereby giving RMSEs that are similar to those of the EKF. However, there was no prior that enabled the UKF to provide better estimates than the EM algorithms with trust region or Gauss-Newton steps[1]. Although there is far less divergence than in the bearings-only tracking scenario considered earlier, Fig. 4 (c) shows that some divergence is still experienced with Newton, damped Newton, and Gauss-Newton updates.

Despite the large RMSEs displayed by the EM algorithms

[1]Upon closer study of the filter, the high RMSEs of the UKF seem to be related to inadequate variance estimates. As previously noted in e.g., [38], there are nonlinear functions for which the Taylor transformation in the EKF provide better variance estimates than the unscented transform.

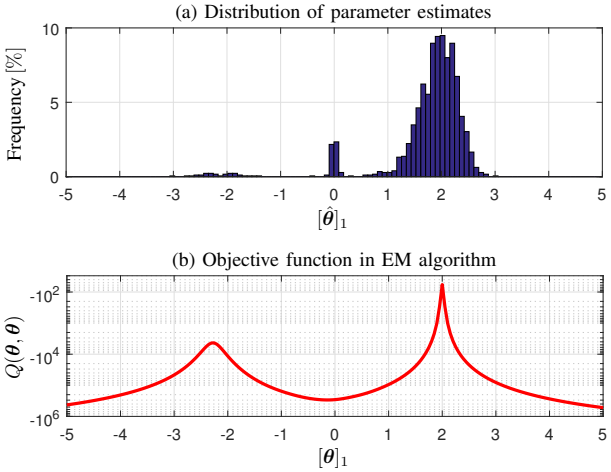

Fig. 5. (a) The distribution of the estimates $[\hat{\boldsymbol{\theta}}]_1$ produced by the EM gradient algorithm in the path loss model scenario with $\sigma_r = 10^{-0.5}$; (b) The objective function $Q(\boldsymbol{\theta}, \boldsymbol{\theta})$ as dependent on $[\boldsymbol{\theta}]_1$ with $[\boldsymbol{\theta}]_2 = 0$, in a randomly chosen simulation of the log-distance path loss model.

with Newton and damped Newton updates in Fig. 4 (a), the estimates provided by these algorithms are in most runs acceptable. This is illustrated in Fig. 5 (a), which shows the distribution of the estimates $[\hat{\boldsymbol{\theta}}]_1$ produced by the EM gradient algorithm at $\sigma_r = 10^{-0.5}$. Most of the estimates can be seen to be concentrated around the true value 2. However, there are also several estimates concentrated around the values -2 and 0. The reason for this can be understood by studying Fig. 5 (b), which displays the objective function $Q(\boldsymbol{\theta}, \boldsymbol{\theta})$ in a randomly chosen simulation with the second element of the parameter vector held fixed at its true value $[\boldsymbol{\theta}]_2 = 0$. This function has local maximums and minimums at approximately $[\boldsymbol{\theta}]_1 = -2$ and $[\boldsymbol{\theta}]_1 = 0$, respectively, which explains why the EM gradient algorithm produced estimates close to these values. The shape of the plot in Fig. 5 (b) can easily be explained based on the geometry in Fig. 3.

V. CONCLUSIONS

This paper examined EM algorithms for estimating the parameters of nonlinear state-space models with additive Gaussian noise. When the M step cannot be solved by analytical means, the EM gradient algorithm provides an approximate solution that is straightforward to implement. However, the solution is based on a second-order Taylor expansion that makes the assumption of a negative definite Hessian. To expand the applicability of the algorithm, five alternative M steps were derived by following the methodology of the EM gradient algorithm and applying modifications based on combinations of the Gauss-Newton method, trust region methods, and damped methods. The resulting EM algorithms were benchmarked in an experimental study using observation models associated with measurements of bearing and received signal strength. Simulations indicated that the proposed M steps compare favorably to the standard Newton step in the EM gradient algorithm both in terms of estimation accuracy and in terms of convergence properties. The lowest RMSEs were achieved by the EM algorithms where the M step

employed trust region steps or damped Gauss-Newton steps (the Levenberg-Marquard method). Given the close connection between trust region and damped methods, the corresponding differences in performance can be expected to be associated with the chosen strategies for updating the trust region radius and the damping parameter. In summary, the proposed EM algorithms are attractive alternatives to the EM gradient algorithm when estimating the parameters of nonlinear state-space models with nonlinearities in the parameter vector. As such, they have the potential to enable stable and computationally efficient estimators for applications within e.g., biomedicine and localization.

ACKNOWLEDGEMENTS

This research is financially supported by the Swedish Foundation for Strategic Research (SSF) via the project *ASSEMBLE*.

REFERENCES

[1] D. T. Westwick, "Nonlinear system identification in biomedical engineering: Techniques, applications and challenges," in *Proc. IFAC Symp. Syst. Identification and Syst. Parameter Estimation*, Newcastle, Australia, Mar. 2006, pp. 128–130.

[2] J. Lu, Z. Yang, K. Okkelberg, and M. Ghovanloo, "Joint magnetic calibration and localization based on expectation maximization for tongue tracking," *Accepted in IEEE Trans. Biomed. Eng.*

[3] G. B. Stanley, *Neural Engineering*. Springer, 2005, ch. Neural System Identification, pp. 367–388.

[4] L. P. Perera, P. Oliveira, and C. G. Soares, "System identification of vessel steering with unstructured uncertainties by persistent excitation maneuvers," *IEEE J. Oceanic Eng.*, vol. 41, no. 3, pp. 515–528, Jul. 2016.

[5] N. Kantas, A. Doucet, S. S. Singh, J. Maciejowski, and N. Chopin, "On particle methods for parameter estimation in state-space models," *Statistical Sci.*, vol. 30, no. 3, pp. 328–351, Aug. 2015.

[6] T. B. Schön, F. Lindsten, J. Dahlin, J. Wågberg, C. A. Naesseth, A. Svensson, and L. Dai, "Sequential Monte Carlo methods for system identification," in *Proc. IFAC Symp. Syst. Identification*, Beijing, China, Oct. 2015, pp. 775–786.

[7] S. Malik and M. K. Pitt, "Particle filters for continuous likelihood evaluation and maximisation," *J. Econometrics*, vol. 165, no. 2, pp. 190–209, Dec. 2011.

[8] C. Snyder, T. Bengtsson, P. Bickel, and J. Anderson, "Obstacles to high-dimensional particle filtering," *Monthly Weather Rev.*, vol. 136, no. 12, pp. 4629–4640, Dec. 2008.

[9] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Royal Statistical Soc., Series B*, vol. 39, no. 1, pp. 1–38, 1977.

[10] A. Dembo and O. Zeitouni, "Parameter estimation of partially observed continuous time stochastic processes via the EM algorithm," *Stochastic Process. and their Appl.*, vol. 23, no. 1, pp. 91–113, Oct. 1986.

[11] K. Lange, "A gradient algorithm locally equivalent to the EM algorithm," *J. Royal Statistical Soc., Series B*, vol. 57, no. 2, pp. 425–437, 1995.

[12] Z. Ghahramani and S. T. Roweis, "Learning nonlinear dynamical systems using an EM algorithm," in *Proc. Int. Conf. Advances in Neural Inf. Process. Syst. II*, Denver, CO, Nov. 1999, pp. 431–437.

[13] T. B. Schön, A. Wills., and B. Ninness., "Maximum likelihood nonlinear system estimation," in *Proc. IFAC Symp. Syst. Identification*, Newcastle, Australia, Mar. 2006, pp. 1003–1008.

[14] E. Liitiäinen, N. Reyhani, and A. Lendasse, "EM-algorithm for training of state-space models with application to time series prediction," in *Proc. European Symp. Artificial Neural Netw.*, Bruges, Belgium, Apr. 2006, pp. 137–142.

[15] E. Özkan, C. Fritsche, and F. Gustafsson, "Online EM algorithm for joint state and mixture measurement noise estimation," in *Proc. Int. Conf. Inf. Fusion*, Singapore, Singapore, Jul. 2012, pp. 1935–1940.

[16] J. Kokkala, A. Solin, and S. Särkkä, "Expectation maximization based parameter estimation by sigma-point and particle smoothing," in *Proc. Int. Conf. Inf. Fusion*, Salamanca, Spain, Jul. 2014.

[17] G. C. G. Wei and M. A. Tanner, "A Monte Carlo implementation of the EM algorithm and the poor man's data augmentation algorithms," *J. American Statistical Association*, vol. 85, no. 411, pp. 699–704, Sep. 1990.

[18] B. Delyon, M. Lavielle, and E. Moulines, "Convergence of a stochastic approximation version of the EM algorithm," *Ann. Statist.*, vol. 27, no. 1, pp. 94–128, Mar. 1999.

[19] X.-L. Meng and D. B. Rubin, "Maximum likelihood estimation via the ECM algorithm: A general framework," *Biometrika*, vol. 80, no. 2, pp. 267–278, Jun. 1993.

[20] X.-L. Meng and D. V. Dyk, "The EM algorithm — an old folk-song sung to a fast new tune," *Series B — Statistical Methodology*, vol. 59, no. 3, pp. 511–567, Jun. 1997.

[21] G. J. McLachlan and T. Krishnan, *The EM Algorithm and Extensions*. Wiley, 2008.

[22] C. Liu and D. B. Rubin, "The ECME algorithm: A simple extension of EM and ECM with faster monotone convergence," *Biometrika*, vol. 81, no. 4, pp. 633–648, Dec. 1994.

[23] D. A. van Dyk and X.-L. Meng, "On the orderings and groupings of conditional maximizations within ECM-type algorithms," *J. Comput. and Graphical Statistics*, vol. 6, no. 2, pp. 202–223, Jun. 1997.

[24] K. Lange, "A quasi-Newton acceleration of the EM algorithm," *Statistica Sinica*, vol. 5, pp. 1–18, 1995.

[25] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2009.

[26] K. Madsen, H. Nielsen, and O. Tingleff, "Methods for non-linear least squares problems," Technical University of Denmark, Tech. Rep., Apr. 2004.

[27] J. Wahlström, I. Skog, and P. Händel, "IMU alignment for smartphone-based automotive navigation," in *Proc. IEEE Int. Conf. Inf. Fusion*, Washington, DC, Jul. 2015, pp. 1437–1443.

[28] K. Metaxoglou and A. Smith, "Maximum likelihood estimation of VARMA models using a state-space EM algorithm," *J. Time Series Anal.*, vol. 28, no. 5, pp. 666–685, Feb. 2007.

[29] D. C. Sorensen, "Newton's method with a model trust region modification," *SIAM J. Numer. Anal.,*, vol. 19, no. 2, pp. 409–426, Apr. 1982.

[30] J. Nocedal and S. J. Wright, *Numerical Optimization*. Springer, 2006.

[31] H. P. Gavin, "The Levenberg-Marquardt method for nonlinear least squares curve-fitting problems," Duke University, Tech. Rep., Mar. 2017.

[32] H. B. Nielsen, "Damping parameter in Marquardt's method," Technical University of Denmark, Tech. Rep. IMM-REP-1999-05.

[33] D. Simon, "Kalman filtering with state constraints: a survey of linear and nonlinear algorithms," *IET Control Theory Appl.*, vol. 4, no. 8, pp. 1303–1318, Aug. 2010.

[34] S. Särkkä, *Bayesian Filtering and Smoothing*. Cambridge University Press, Sep. 2013.

[35] F. Gustafsson and G. Hendeby, "Some relations between extended and unscented Kalman filters," *IEEE Trans. Signal Process.*, vol. 60, no. 2, pp. 545–555, Feb. 2012.

[36] H. Nurminen, J. Talvitie, S. Ali-Löytty, P. Müller, E. S. Lohan, R. Piché, and M. Renfors, "Statistical path loss parameter estimation and positioning using RSS measurements in indoor wireless networks," in *Proc. IEEE Int. Conf. Indoor Positioning Indoor Navigation*, Sydney, Australia, Nov. 2012.

[37] Y. Zhao, F. Yin, F. Gunnarsson, M. Amirijoo, E. Özkan, and F. Gustafsson, "Particle filtering for positioning based on proximity reports," in *Proc. IEEE Int. Conf. Inf. Fusion*, Washington, DC, Jul. 2015, pp. 1046–1052.

[38] M. Rhudy, Y. Gu, and M. R. Napolitano, "An analytical approach for comparing linearization methods in EKF and UKF," *Int. J. Advanced Robot. Syst.*, vol. 10, no. 4, Apr. 2013.