

COOPERATIVE SIMULTANEOUS LOCALIZATION AND TRACKING (COSLAT) WITH REDUCED COMPLEXITY AND COMMUNICATION

Florian Meyer¹, Franz Hlawatsch¹, and Henk Wymeersch²

¹Institute of Telecommunications, Vienna University of Technology, Vienna, Austria ({fmeyer, fhlawats}@nt.tuwien.ac.at)

²Department of Signals and Systems, Chalmers University of Technology, Gothenburg, Sweden (henkw@chalmers.se)

ABSTRACT

The recently introduced framework of *cooperative simultaneous localization and tracking* (CoSLAT) combines Bayesian cooperative agent self-localization with distributed target tracking. The original CoSLAT algorithm suffers from high computation and communication costs because it uses a particle-based message representation. Here, we propose an advanced *hybrid particle-based and parametric* message passing algorithm for CoSLAT in which both costs are significantly reduced. Simulation results show that the localization/tracking performance is not affected.

Index Terms—Distributed target tracking, cooperative localization, CoSLAT, nonparametric belief propagation, likelihood consensus.

1. INTRODUCTION

Contribution and relation to previous work. In decentralized agent networks, the tasks of *cooperative self-localization* (CSL) [1, 2] and *distributed target tracking* (DTT) [3] are closely related, since (i) to contribute to DTT, an agent needs to possess information about its own location, and (ii) the performance of CSL may be improved if the agents possess information about the location of a target. Therefore, devising *joint* CSL-DTT schemes is a promising approach.

In CSL, each cooperative agent (CA) measures its own location relative to neighboring CAs and estimates its own location by cooperating with other CAs [1, 2, 4–7]. In DTT, each CA acquires a measurement that is related to the state of a target, and it cooperatively estimates that state from the measurements of *all* CAs [3, 8–10]. *Simultaneous localization and tracking* (SLAT) [11–16] is a first attempt to combine self-localization and DTT. In SLAT, the CAs simultaneously track a target and localize themselves, however without using inter-CA distance measurements.

The recently introduced framework of *cooperative simultaneous localization and tracking* (CoSLAT) [17] provides a coherent combination of CSL and DTT. CoSLAT extends SLAT by using also inter-CA distance measurements. In [17], a distributed Bayesian message passing algorithm for CoSLAT was proposed. This algorithm integrates DTT in nonparametric belief propagation (NBP) based CSL [4, 18]. Its main advantage over previously proposed algorithms is a probabilistic information transfer between CSL and DTT, which allows CSL and DTT to support each other and thus can yield significant gains in both self-localization and target tracking performance [17]. However, just as NBP-based CSL [2, 4, 18], the algorithm has high computation and communication costs because it uses a particle-based message representation for self-localization.

Here, we propose an advanced CoSLAT algorithm that achieves significant savings in both communications and computation through the use of parametric inter-CA messages (as introduced in [18], al-

though the parameters are determined differently) and a simpler procedure to perform a message multiplication operation. In a two-dimensional (2D) setting, the communication cost is reduced by about an order of magnitude. Furthermore, the computational complexity scales only linearly with the number of particles, rather than quadratically as in the case of the algorithm of [17].

Paper outline. The system model is described in Section 2. In Section 3, the CoSLAT problem and the algorithm of [17] are reviewed. The proposed improved CoSLAT algorithm is developed in Section 4. Finally, simulation results are presented in Section 5.

2. SYSTEM MODEL

We consider a network of K CAs and one target agent, as shown in Fig. 1. The target agent is noncooperative, i.e., it does not take part in the CoSLAT process. All agents may be mobile. We index them by $k \in \mathcal{A} = \{0, \dots, K\}$, where $k = 0$ designates the target and $k \in \mathcal{A}_{\sim 0} \triangleq \mathcal{A} \setminus \{0\}$ the CAs. Typically, a small subset of static “anchor” CAs have perfect knowledge of their location. The *state* $\mathbf{x}_{k,n}$ of agent $k \in \mathcal{A}$ at time $n \in \{0, 1, \dots\}$ comprises the current location and, possibly, additional motion parameters [19]. The evolution of the states $\mathbf{x}_{k,n}$ is described by the state transition probability density functions (pdfs) $f(\mathbf{x}_{k,n} | \mathbf{x}_{k,n-1})$ and the prior pdf $f(\mathbf{x}_{k,0})$.

The communication and measurement topologies are characterized by (generally time-dependent) sets \mathcal{C}_n , $\mathcal{M}_{k,n}$, and \mathcal{T}_n . Specifically, two CAs $k, l \in \mathcal{A}_{\sim 0}$ are able to communicate with each other if $(k, l) \in \mathcal{C}_n \subseteq \mathcal{A}_{\sim 0} \times \mathcal{A}_{\sim 0}$. Here, \mathcal{C}_n is symmetric, i.e., if $(k, l) \in \mathcal{C}_n$, then $(l, k) \in \mathcal{C}_n$. Furthermore, CA $k \in \mathcal{A}_{\sim 0}$ acquires a measurement $y_{k,l;n}$ of its distance to CA $l \in \mathcal{A}_{\sim 0}$, with $(k, l) \in \mathcal{C}_n$, if $l \in \mathcal{M}_{k,n} \subseteq \mathcal{A}_{\sim 0} \setminus \{k\}$. Finally, CA $k \in \mathcal{A}_{\sim 0}$ acquires a measurement $y_{k,0;n}$ of its distance to the target, i.e., $0 \in \mathcal{M}_{k,n}$, if $k \in \mathcal{T}_n \subseteq \mathcal{A}_{\sim 0}$; thus, $\mathcal{T}_n \triangleq \{k \in \mathcal{A}_{\sim 0} | 0 \in \mathcal{M}_{k,n}\}$. An example of communication and measurement topologies is given in Fig. 1. We consider a 2D scenario; the extension to the 3D case is straightforward.

The distance measurements are modeled as

$$y_{k,l;n} = \|\tilde{\mathbf{x}}_{k,n} - \tilde{\mathbf{x}}_{l,n}\| + v_{k,l;n}, \quad (1)$$

where $\tilde{\mathbf{x}}_{k,n} \triangleq [x_{1,k,n} \ x_{2,k,n}]^T$ represents the location of agent $k \in \mathcal{A}$ (note that this is a part of the state $\mathbf{x}_{k,n}$). The measurement noise $v_{k,l;n}$

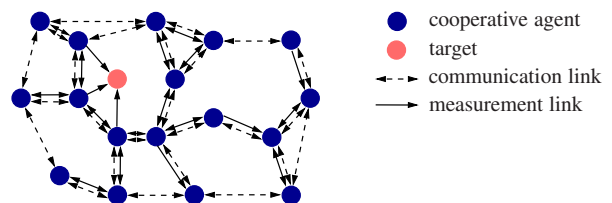


Fig. 1. Example of an agent network with a target.

This work was supported by the FWF under Grant S10603 and by the European Research Council under Grant No. 258418 (COOPNET).

is assumed Gaussian with a known variance σ_v^2 ; we also assume that $v_{k,l;n}$ and $v_{k',l',n'}$ are independent unless $(k, l, n) = (k', l', n')$.

3. REVIEW OF CoSLAT

In CoSLAT [17], at time n , each CA $k \in \mathcal{A}_{\sim 0}$ estimates both its own state $\mathbf{x}_{k,n}$ and the target state $\mathbf{x}_{0,n}$, using the measurements of the inter-CA distances and CA-target distances up to time n , i.e., $\mathcal{Y}_{1:n} \triangleq \{y_{k,l;n'}\}_{k \in \mathcal{A}_{\sim 0}, l \in \mathcal{M}_{k,n'}, n' \in \{1, \dots, n\}}$. The minimum mean square error (MMSE) estimator [20] of state $\mathbf{x}_{k,n}$, $k \in \mathcal{A}$ is given by

$$\hat{\mathbf{x}}_{k,n}^{\text{MMSE}} \triangleq \mathbb{E}\{\mathbf{x}_{k,n} | \mathcal{Y}_{1:n}\} = \int \mathbf{x}_{k,n} f(\mathbf{x}_{k,n} | \mathcal{Y}_{1:n}) d\mathbf{x}_{k,n}.$$

Here, the posterior pdf $f(\mathbf{x}_{k,n} | \mathcal{Y}_{1:n})$ needs to be computed from the joint posterior of the state set $\mathcal{X}_{0:n} \triangleq \{\mathbf{x}_{k,n'}\}_{k \in \mathcal{A}, n' \in \{0, \dots, n\}}$, $f(\mathcal{X}_{0:n} | \mathcal{Y}_{1:n})$, by marginalization.

3.1. Belief Propagation Message Passing

Using common assumptions [2], the joint posterior $f(\mathcal{X}_{0:n} | \mathcal{Y}_{1:n})$ factorizes according to [17, Equation 3], which corresponds to the factor graph [21] shown in Fig. 2. An efficient approximate marginalization of $f(\mathcal{X}_{0:n} | \mathcal{Y}_{1:n})$ can be achieved by applying a belief propagation [21] message passing scheme to this factor graph [17]. At each time n , P message passing iterations are performed. The approximate marginal posterior (AMP) of agent node $k \in \mathcal{A}$ at message passing iteration $p \in \{1, \dots, P\}$ is given by

$$b_{k,n}^{(p)}(\mathbf{x}_{k,n}) \propto \begin{cases} m_{\rightarrow n}(\mathbf{x}_{k,n}) \prod_{l \in \mathcal{M}_{k,n}} m_{l \rightarrow k}^{(p)}(\mathbf{x}_{k,n}), & k \in \mathcal{A}_{\sim 0} \\ m_{\rightarrow n}(\mathbf{x}_{0,n}) \prod_{l \in \mathcal{T}_n} m_{l \rightarrow 0}^{(p)}(\mathbf{x}_{0,n}), & k = 0, \end{cases} \quad (2)$$

where the ‘‘prediction message’’ $m_{\rightarrow n}(\mathbf{x}_{k,n})$ is calculated from the state transition pdf and the final iterated AMP at time $n-1$ as

$$m_{\rightarrow n}(\mathbf{x}_{k,n}) = \int f(\mathbf{x}_{k,n} | \mathbf{x}_{k,n-1}) b_{k,n-1}^{(P)}(\mathbf{x}_{k,n-1}) d\mathbf{x}_{k,n-1}, \quad (3)$$

and the ‘‘measurement messages’’ $m_{l \rightarrow k}^{(p)}(\mathbf{x}_{k,n})$ are calculated as

$$m_{l \rightarrow k}^{(p)}(\mathbf{x}_{k,n}) = \begin{cases} \int f(y_{k,l;n} | \mathbf{x}_{k,n}, \mathbf{x}_{l,n}) b_{l,n}^{(p-1)}(\mathbf{x}_{l,n}) d\mathbf{x}_{l,n}, & k \in \mathcal{A}_{\sim 0} \\ \int f(y_{l,0;n} | \mathbf{x}_{0,n}, \mathbf{x}_{l,n}) b_{l,n}^{(p-1)}(\mathbf{x}_{l,n}) d\mathbf{x}_{l,n}, & k = 0. \end{cases} \quad (4)$$

As discussed in [17], this message passing scheme uses an approximation that avoids the costly calculation of extrinsic information. Numerical analysis showed that this approximation leads to slightly overconfident AMPs but does not degrade the estimation performance. Due to the measurement model in (1), $m_{l \rightarrow k}^{(p)}(\mathbf{x}_{k,n})$ depends only on the AMP of the location of agent l , $b_{l,n}^{(p-1)}(\tilde{\mathbf{x}}_{l,n})$, and is itself only a function of the location of agent k , $\tilde{\mathbf{x}}_{k,n}$. Hence, hereafter we will write $m_{l \rightarrow k}^{(p)}(\tilde{\mathbf{x}}_{k,n})$ instead of $m_{l \rightarrow k}^{(p)}(\mathbf{x}_{k,n})$. Messages are sent only forward in time, and iterative message passing is performed at each time step individually [2]. Therefore, $m_{\rightarrow n}(\mathbf{x}_{k,n})$ in (3) remains unchanged during the message passing iterations. Note that the message passing scheme (2)–(4) transfers probabilistic information between the CSL and DTT parts of the algorithm.

Direct calculation of (2)–(4) is still infeasible in general. An approximate NBP implementation [4, 18] was considered in [17]. This typically requires the transmission of several hundreds of particles

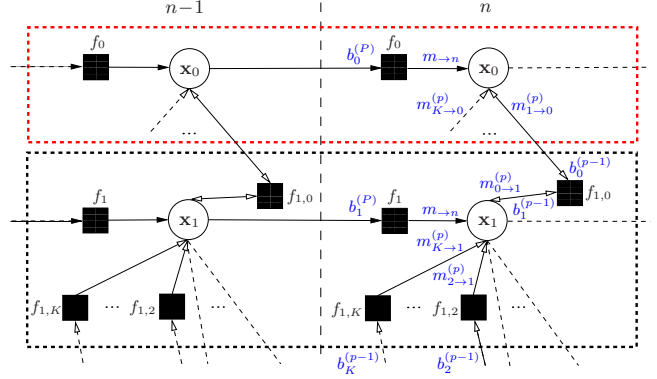


Fig. 2. A CoSLAT factor graph, shown for times $n-1$ and n . The top (red) dotted box corresponds to the DTT part (target $k=0$); the bottom (black) dotted box corresponds to the CSL part for CA $k=1$. Also shown are the messages and approximate marginal posteriors (AMPs) involved in calculating $b_{0,n}(\mathbf{x}_{0,n})$ and $b_{1,n}(\mathbf{x}_{1,n})$. All time indices are omitted for simplicity, and the following short notation is used: $f_k \triangleq f(\mathbf{x}_{k,n'} | \mathbf{x}_{k,n'-1})$ and $f_{k,l} \triangleq f(y_{k,l;n'} | \mathbf{x}_{k,n'}, \mathbf{x}_{l,n'})$, for $n' \in \{1, 2, \dots\}$. Edges with filled arrowheads depict messages or AMPs that are represented by *particles*; edges with non-filled arrowheads depict *parametric* measurement messages or AMPs.

between communicating CAs. Furthermore, NBP has a high computational complexity. This is mainly due to the message multiplication in (2), whose complexity scales as $\mathcal{O}(MJ^2)$, where J is the number of particles and M is the number of messages multiplied [18].

3.2. Distributed Implementation of CoSLAT

A distributed implementation of the NBP message passing scheme for CoSLAT is complicated by the fact that the product of measurement messages $\prod_{l \in \mathcal{T}_n} m_{l \rightarrow 0}^{(p)}(\tilde{\mathbf{x}}_{0,n})$ in (2) is not available at the CAs. In [17], this problem is solved by using the likelihood consensus scheme [8]. More specifically, at each CA $l \in \mathcal{T}_n$, the logarithm of $m_{l \rightarrow 0}^{(p)}(\tilde{\mathbf{x}}_{0,n})$ is approximated by a finite-order basis expansion:

$$\log m_{l \rightarrow 0}^{(p)}(\tilde{\mathbf{x}}_{0,n}) \approx \sum_{r=1}^R \beta_{l;n,r}^{(p)}(y_{l,0;n}) \varphi_r(\tilde{\mathbf{x}}_{0,n}). \quad (5)$$

Here, the ‘‘basis functions’’ $\varphi_r(\tilde{\mathbf{x}}_{0,n})$ do not depend on the specific CA l . The expansion coefficients $\beta_{l;n,r}^{(p)}(y_{l,0;n})$, $r \in \{1, \dots, R\}$ can be calculated locally at CA l by least squares fitting [22] using the location part of the particles representing $m_{\rightarrow n}(\mathbf{x}_{l,n})$ as reference points (cf. [8]). Furthermore, we formally set $\beta_{l;n,r}^{(p)}(y_{l,0;n}) = 0$ for all $r \in \{1, \dots, R\}$ if $l \notin \mathcal{T}_n$. The approximations (5) then entail the following approximation of the desired message product [17]:

$$\prod_{l \in \mathcal{T}_n} m_{l \rightarrow 0}^{(p)}(\tilde{\mathbf{x}}_{0,n}) \approx \exp\left(\sum_{r=1}^R B_{n,r}^{(p)} \varphi_r(\tilde{\mathbf{x}}_{0,n})\right),$$

with

$$B_{n,r}^{(p)} = \sum_{l \in \mathcal{T}_n} \beta_{l;n,r}^{(p)}(y_{l,0;n}) = \sum_{l \in \mathcal{A}_{\sim 0}} \beta_{l;n,r}^{(p)}(y_{l,0;n}). \quad (6)$$

The coefficients $B_{n,r}^{(p)}$ in (6) can be obtained at each CA by running R parallel instances of an average consensus algorithm or a gossip algorithm [23, 24] in the agent network. This requires only local communications between neighboring CAs.

4. REDUCING COMPLEXITY AND COMMUNICATIONS

In the novel CoSLAT algorithm, a *parametric* representation of all AMPs is used. In this way, only the AMP parameters have to be transmitted between neighboring CAs (localization partners). The CAs then calculate parametric representations of the measurement messages. Using the importance sampling principle, the message multiplication in (2) can thus be done by evaluating the product of all available measurement messages at the predicted particles. Because of the parametric representations, kernel representations (as in [17]) are not needed; thereby, the complexity is reduced from $\mathcal{O}(MJ^2)$ to $\mathcal{O}(MJ)$ and the communication cost by an order of magnitude.

The measurement messages are represented using the annular parametric distributions introduced in [18]. However, whereas [18] employed sampling from the measurement messages and an optimization step, we determine the parameters directly from the AMP parameters of the localization partners and from the agent location estimate at the previous message passing iteration. Involving the previous agent location estimate also allows a more accurate determination of the width parameters of the measurement messages.

In the following, we present a more detailed description of the proposed CoSLAT algorithm.

4.1. Step 1: Extracting the AMP Parameters

Consider $b_{l,n}^{(p-1)}(\tilde{\mathbf{x}}_{l,n})$, i.e., the AMP of the location $\tilde{\mathbf{x}}_{l,n}$ of CA $l \in \mathcal{A}_{\sim 0}$ at message passing iteration $p-1$. This 2D function is either unimodal if the CA is well localized; bimodal with two modes if the CA is localized with ambiguity; or multimodal (e.g., annularly shaped) if the CA is poorly localized [2]. To reduce communications, we approximate $b_{l,n}^{(p-1)}(\tilde{\mathbf{x}}_{l,n})$ by a Gaussian $\mathcal{N}(\tilde{\boldsymbol{\mu}}_{l,n}, \tilde{\mathbf{C}}_{l,n})$ if it is unimodal and by a mixture of two Gaussians $\mathcal{N}(\tilde{\boldsymbol{\mu}}_{l,n}^{(1)}, \tilde{\mathbf{C}}_{l,n}^{(1)})$ and $\mathcal{N}(\tilde{\boldsymbol{\mu}}_{l,n}^{(2)}, \tilde{\mathbf{C}}_{l,n}^{(2)})$ with equal weights if it is bimodal. The means and covariances are then transmitted to the localization partners, which requires the transmission of $2+3=5$ real numbers in the unimodal case and of 10 real numbers in the bimodal case. If $b_{l,n}^{(p-1)}(\tilde{\mathbf{x}}_{l,n})$ is multimodal, no AMP parameters are transmitted, because a poorly localized CA cannot provide useful information to its partners.

We propose the following procedure for extracting the AMP parameters at CA l . First, CA l derives particles $\{\tilde{\mathbf{x}}_{l,n}^{(j)}\}_{j=1}^J$ representing $b_{l,n}^{(p-1)}(\tilde{\mathbf{x}}_{l,n})$ from the particles representing $b_{l,n}^{(p-1)}(\mathbf{x}_{l,n})$ by discarding the irrelevant entries in each particle vector (recall that $\tilde{\mathbf{x}}_{l,n}$ is a subvector of $\mathbf{x}_{l,n}$). Next, CA l uses a clustering algorithm such as K-means [25] to partition $\{\tilde{\mathbf{x}}_{l,n}^{(j)}\}_{j=1}^J$ into two disjoint subsets $\{\tilde{\mathbf{x}}_{l,n}^{(j)}\}_{j \in \mathcal{J}_1}$ and $\{\tilde{\mathbf{x}}_{l,n}^{(j)}\}_{j \in \mathcal{J}_2}$, and it calculates the Fisher linear discriminant [25] (denoted D) for that partition. Also, a mean and a covariance matrix are computed for each particle subset, i.e.,

$$\tilde{\boldsymbol{\mu}}_{l,n}^{(i)} = \frac{1}{|\mathcal{J}_i|} \sum_{j \in \mathcal{J}_i} \tilde{\mathbf{x}}_{l,n}^{(j)}, \quad \tilde{\mathbf{C}}_{l,n}^{(i)} = \frac{1}{|\mathcal{J}_i|} \sum_{j \in \mathcal{J}_i} \tilde{\mathbf{x}}_{l,n}^{(j)} \tilde{\mathbf{x}}_{l,n}^{(j)\text{T}} - \tilde{\boldsymbol{\mu}}_{l,n}^{(i)} \tilde{\boldsymbol{\mu}}_{l,n}^{(i)\text{T}},$$

for $i \in \{1, 2\}$. If D is above a threshold T and $\|\tilde{\boldsymbol{\mu}}_{l,n}^{(1)} - \tilde{\boldsymbol{\mu}}_{l,n}^{(2)}\| > 4\sigma_v$, the clustering result is accepted and, thus, the bimodal Gaussian mixture model is adopted for $b_{l,n}^{(p-1)}(\tilde{\mathbf{x}}_{l,n})$. Otherwise, the clustering is rejected and a single mean $\tilde{\boldsymbol{\mu}}_{l,n}$ and covariance matrix $\tilde{\mathbf{C}}_{l,n}$ are determined from the total particle set $\{\tilde{\mathbf{x}}_{l,n}^{(j)}\}_{j=1}^J$. Then, if $(\tilde{\mathbf{C}}_{l,n})_{1,1} + (\tilde{\mathbf{C}}_{l,n})_{2,2} < 10\sigma_v^2$, the unimodal Gaussian model $\mathcal{N}(\tilde{\boldsymbol{\mu}}_{l,n}, \tilde{\mathbf{C}}_{l,n})$ is adopted, otherwise $b_{l,n}^{(p-1)}(\tilde{\mathbf{x}}_{l,n})$ is considered multimodal. For $l=0$ (the target), CA k calculates $b_{0,n}^{(p-1)}(\mathbf{x}_{0,n})$ via the likelihood con-

sensus scheme described in Section 3.2 and extracts corresponding parameters $\tilde{\boldsymbol{\mu}}_{0,n}, \tilde{\mathbf{C}}_{0,n}$ based on the unimodal Gaussian model.

4.2. Step 2: Calculating the Parametric Measurement Messages

After all AMP parameters have been transmitted, each CA $k \in \mathcal{A}_{\sim 0}$ knows (approximate representations of) the AMPs $b_{l,n}^{(p-1)}(\tilde{\mathbf{x}}_{l,n})$ of its localization partners $l \in \mathcal{M}_{k,n}$. CA k next calculates a particle representation of its own AMP $b_{k,n}^{(p)}(\mathbf{x}_{k,n})$ by implementing (2)–(4) as described in Section 4.3. Because this requires closed-form expressions of the measurement messages $m_{l \rightarrow k}^{(p)}(\tilde{\mathbf{x}}_{k,n})$, $l \in \mathcal{M}_{k,n}$, we use the parametric message representations introduced in [18]. More specifically, if $b_{l,n}^{(p-1)}(\tilde{\mathbf{x}}_{l,n})$ is unimodal, we set

$$m_{l \rightarrow k}^{(p)}(\tilde{\mathbf{x}}_{k,n}) \propto \exp\left(-\frac{(y_{l,k;n} - \|\tilde{\mathbf{x}}_{k,n} - \tilde{\boldsymbol{\mu}}_{l,n}\|)^2}{2r_{l,k;n}}\right). \quad (7)$$

This is an annulus about $\tilde{\boldsymbol{\mu}}_{l,n}$ with nominal radius $y_{l,k;n}$; the radial width about the nominal radius is determined by $r_{l,k;n}$. If $b_{l,n}^{(p-1)}(\tilde{\mathbf{x}}_{l,n})$ is bimodal, we set $m_{l \rightarrow k}^{(p)}(\tilde{\mathbf{x}}_{k,n})$ equal to the sum of two annuli that are located about $\tilde{\boldsymbol{\mu}}_{l,n}^{(1)}$ and $\tilde{\boldsymbol{\mu}}_{l,n}^{(2)}$ and have equal nominal radius $y_{l,k;n}$ and possibly different width parameters $r_{l,k;n}^{(1)}$ and $r_{l,k;n}^{(2)}$. Finally, if $b_{l,n}^{(p-1)}(\tilde{\mathbf{x}}_{l,n})$ is multimodal, CA k ignores localization partner l by setting $m_{l \rightarrow k}^{(p)}(\tilde{\mathbf{x}}_{k,n})$ to a constant value.

It remains to determine the width parameter(s). Let us first consider the unimodal representation (7). If agent l is an anchor CA, we have $b_{l,n}^{(p-1)}(\tilde{\mathbf{x}}_{l,n}) = \delta(\tilde{\mathbf{x}}_{l,n} - \tilde{\boldsymbol{\mu}}_{l,n})$, and the message (7) with $r_{l,k;n} = \sigma_v^2$ is exactly equal to (4). Otherwise, let $\rho_{k,l;n}(\tilde{\mathbf{x}}_{l,n}) \triangleq \|\hat{\tilde{\mathbf{x}}}_{k,n}^{(p-1)} - \tilde{\mathbf{x}}_{l,n}\|$ be the distance of the estimate of $\tilde{\mathbf{x}}_{k,n}$ at message passing iteration $p-1$, denoted $\hat{\tilde{\mathbf{x}}}_{k,n}^{(p-1)}$, from $\tilde{\mathbf{x}}_{l,n}$. A good approximation of (4) is then obtained by choosing

$$r_{l,k;n} = \mathbf{h}_{k,l;n}^{\text{T}} \tilde{\mathbf{C}}_{l,n} \mathbf{h}_{k,l;n} + \sigma_v^2, \quad (8)$$

where $\mathbf{h}_{k,l;n}$ is the gradient of $\rho_{k,l;n}(\tilde{\mathbf{x}}_{l,n})$ evaluated at $\tilde{\boldsymbol{\mu}}_{l,n}$ [7]. This result for $r_{l,k;n}$ is obtained via a linear approximation of the “reduced” measurement equation $y'_{k,l;n} = \rho_{k,l;n}(\tilde{\mathbf{x}}_{l,n}) + v_{k,l;n}$ around $\tilde{\boldsymbol{\mu}}_{l,n}$ [7]. More specifically, $r_{l,k;n}$ is the variance of $\rho_{k,l;n}(\tilde{\boldsymbol{\mu}}_{l,n}) + \mathbf{h}_{k,l;n}^{\text{T}}(\tilde{\mathbf{x}}_{l,n} - \tilde{\boldsymbol{\mu}}_{l,n}) + v_{k,l;n}$. Note that now the radial width of the annular message $m_{l \rightarrow k}^{(p)}(\tilde{\mathbf{x}}_{k,n})$ in (7) is influenced by both the uncertainty in the l th CA location, expressed by $\mathbf{h}_{k,l;n}^{\text{T}} \tilde{\mathbf{C}}_{l,n} \mathbf{h}_{k,l;n}$, and the measurement variance σ_v^2 .

For the bimodal representation, we choose the two width parameters as in (8), i.e., $r_{l,k;n}^{(i)} = \mathbf{h}_{k,l;n}^{(i)\text{T}} \tilde{\mathbf{C}}_{l,n}^{(i)} \mathbf{h}_{k,l;n}^{(i)} + \sigma_v^2$ for $i \in \{1, 2\}$, where $\mathbf{h}_{k,l;n}^{(i)}$ is the gradient of $\rho_{k,l;n}(\tilde{\mathbf{x}}_{l,n})$ evaluated at $\tilde{\boldsymbol{\mu}}_{l,n}^{(i)}$.

4.3. Step 3: Updating the AMPs

With all messages $m_{l \rightarrow k}^{(p)}(\tilde{\mathbf{x}}_{k,n})$, $l \in \mathcal{M}_{k,n}$ determined, an approximation of the functional form of $\prod_{l \in \mathcal{M}_{k,n}} m_{l \rightarrow k}^{(p)}(\tilde{\mathbf{x}}_{k,n})$ is available at CA $k \in \mathcal{A}_{\sim 0}$. Thus, CA k is able to calculate a particle representation of its updated AMP $b_{k,n}^{(p)}(\mathbf{x}_{k,n})$ according to (2). This is done by means of importance sampling [26], using the prediction message $m_{\rightarrow n}(\mathbf{x}_{k,n})$ as proposal density: particles $\{\mathbf{x}_{k,n}^{(j)}\}_{j=1}^J$ are drawn from $m_{\rightarrow n}(\mathbf{x}_{k,n})$, and associated weights are obtained as $w_{k,n}^{(j)} = \prod_{l \in \mathcal{M}_{k,n}} m_{l \rightarrow k}^{(p)}(\tilde{\mathbf{x}}_{k,n}^{(j)})$. This is followed by a resampling step [26] to obtain equally weighted particles.

Comments: The complexity of the overall algorithm scales as $\mathcal{O}(MJ)$. If the prior of CA k (at time $n=0$) is uniform, very non-informative, or difficult to sample from, it may be preferable to use

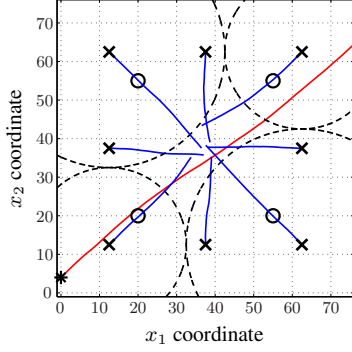


Fig. 3. Network topology used for the simulations, with example realizations of the target and CA trajectories. Initial mobile CA locations are indicated by crosses, anchor locations by circles, and the initial target location by a star. The big dashed circles indicate the measurement regions of the CAs initially located near the corners.

one of the measurement messages $m_{l \rightarrow k}^{(p)}(\tilde{\mathbf{x}}_{k,1})$ as proposal density at time $n=1$. The drawing of particles from $m_{\rightarrow n}(\mathbf{x}_{k,n})$ in (3) using particles representing $b_{k,n-1}^{(p)}(\mathbf{x}_{k,n-1})$ and the drawing of particles from $m_{l \rightarrow k}^{(p)}(\tilde{\mathbf{x}}_{k,n})$ in (4) using particles representing $b_{l,n}^{(p-1)}(\mathbf{x}_{l,n})$ are described in [18].

5. SIMULATION RESULTS

We consider a network of $K=12$ CAs and one target as depicted in Fig. 3. Eight CAs are mobile and four CAs are anchors (i.e., static CAs with perfect location information modeled via Dirac-shaped priors). Each CA moves within a field of size 75×75 , has a communication range of 50, and attempts to localize itself and the target. The measurement regions of the four CAs initially located near the corners are indicated in Fig. 3 by dashed circles. The measurement regions of the other eight CAs cover the entire field.

The states of the mobile CAs and of the target consist of location and velocity, i.e., $\mathbf{x}_{k,n} = [x_{1,k,n} \ x_{2,k,n} \ \dot{x}_{1,k,n} \ \dot{x}_{2,k,n}]^T$. Each mobile CA starts moving only when it is sufficiently localized in the sense that the trace of its location covariance matrix is below $5\sigma_v^2$. The CA then attempts to reach the center of the scene, $\tilde{\mathbf{x}}_c = [37.5 \ 37.5]^T$, in 75 time steps; therefore, the mobile CA trajectories are generated using a Dirac-shaped prior located at $\boldsymbol{\mu}_{k,0} = [x_{1,k,0} \ x_{2,k,0} \ (\tilde{x}_{1,c} - x_{1,k,0})/75 \ (\tilde{x}_{2,c} - x_{2,k,0})/75]^T$, where $x_{1,k,0}$ and $x_{2,k,0}$ are chosen as shown in Fig. 3. However, for our simulations of the algorithms, we assumed a location prior that is uniform on $[-500, 500] \times [-500, 500]$ and—after the CA is sufficiently localized—a Gaussian velocity prior with mean $[(\tilde{x}_{1,c} - \hat{x}_{1,k,n'})/75 \ (\tilde{x}_{2,c} - \hat{x}_{2,k,n'})/75]^T$ and covariance matrix $\text{diag}\{0.001, 0.001\}$. Here, $\hat{\mathbf{x}}_{k,n'}$ is the location estimate at the time n' at which the k th CA is sufficiently localized for the first time. The prior of the target state is Gaussian with mean $\boldsymbol{\mu}_{0,0} = [0 \ 5 \ 0.4 \ 0.4]^T$ and covariance matrix $\mathbf{C}_{0,0} = \text{diag}\{1, 1, 0.001, 0.001\}$. The mobile CAs and the target evolve independently according to $\mathbf{x}_{k,n} = \mathbf{G}\mathbf{x}_{k,n-1} + \mathbf{W}\mathbf{u}_{k,n}$, $n = 1, 2, \dots$ [19], where the matrices $\mathbf{G} \in \mathbb{R}^{4 \times 4}$ and $\mathbf{W} \in \mathbb{R}^{4 \times 2}$ are chosen as in [8] and the driving noise vectors $\mathbf{u}_{k,n} \in \mathbb{R}^2$ are Gaussian, i.e., $\mathbf{u}_{k,n} \sim \mathcal{N}(\mathbf{0}, \sigma_u^2 \mathbf{I})$, with variance $\sigma_u^2 = 0.00005$ for the mobile CAs and $\sigma_u^2 = 0.001$ for the target; furthermore, $\mathbf{u}_{k,n}$ and $\mathbf{u}_{k',n'}$ are independent unless $(k,n) = (k',n')$. The observation noise variance is $\sigma_v^2 = 2$. We performed 100 simulation runs.

We compare the performance of the proposed CoSLAT algorithm

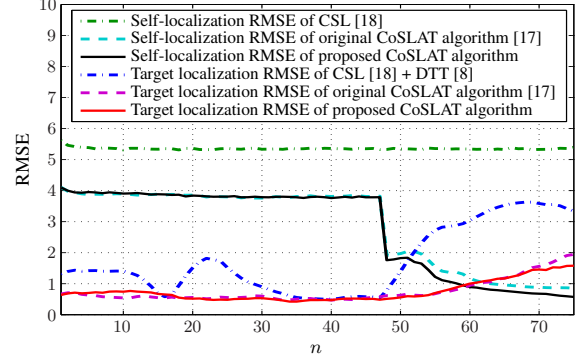


Fig. 4. Average root-mean-square errors (RMSE) of self-localization and target localization versus time n .

with that of the original CoSLAT algorithm in [17]. In addition, we consider a second reference method that separately performs CSL by means of NBP as described in [18] and DTT by means of the likelihood consensus-based distributed particle filter presented in [8]; the latter uses the CA location estimates provided by CSL. In all three methods, the likelihood consensus scheme uses an average consensus [23] with six iterations, and the basis expansion is a third-order polynomial approximation [8], resulting in an expansion order of $R=16$. The NBP scheme performs $P=3$ message passing iterations. The number of particles used by both NBP and the distributed particle filter is $J=500$. The threshold in the proposed algorithm is $T=40$; this value was observed to lead to a reliable clustering of particles for a bimodal AMP.

Fig. 4 shows the simulated root-mean-square self-localization and target localization errors for $n = 1, \dots, 75$. These errors were determined by averaging over all CAs and all simulation runs. It is seen that the proposed algorithm performs equally well as the algorithm of [17], both with respect to self-localization and target tracking; thus, the substantial savings in complexity and communication are not offset by a performance loss. Furthermore, the self-localization error of both CoSLAT algorithms is seen to be significantly smaller than that of the second reference method. This is because with pure CSL, the lower-left and upper-right CAs do not have enough partners for accurate self-localization, whereas with CoSLAT, these CAs can use their measured distance to the target to calculate the message from the target node, $m_{0 \rightarrow k}^{(p)}(\tilde{\mathbf{x}}_{k,n})$, which is exploited for improved self-localization. Additionally, also the target tracking error of both CoSLAT algorithms is significantly smaller than that of the second reference method for almost all times. This is because with separate CSL and DTT, the poor self-localization of the lower-left and upper-right CAs results in a degraded target tracking performance.

6. CONCLUSION

We have proposed a novel hybrid parametric/nonparametric message passing algorithm for cooperative simultaneous localization and tracking (CoSLAT). This algorithm uses parametric representations of the measurement messages and of the inter-agent messages involved in the approximate marginal posteriors (AMPs). The parameters of the measurement messages are determined directly from the AMP parameters of the localization partners and from the iterated location estimate. Compared to the original CoSLAT method, the proposed algorithm achieves an order-of-magnitude reduction of communications (in a two-dimensional setting) and a substantial reduction of computational complexity, without a loss in performance.

7. REFERENCES

- [1] N. Patwari, J. N. Ash, S. Kyperountas, A. O. Hero III, R. L. Moses, and N. S. Correal, "Locating the nodes: Cooperative localization in wireless sensor networks," *IEEE Signal Process. Mag.*, vol. 22, pp. 54–69, Jul. 2005.
- [2] H. Wymeersch, J. Lien, and M. Z. Win, "Cooperative localization in wireless networks," *Proc. IEEE*, vol. 97, pp. 427–450, Feb. 2009.
- [3] J. Liu, M. Chu, and J. Reich, "Multitarget tracking in distributed sensor networks," *IEEE Signal Process. Mag.*, vol. 24, pp. 36–46, May 2007.
- [4] A. T. Ihler, J. W. Fisher, R. L. Moses, and A. S. Willsky, "Nonparametric belief propagation for self-localization of sensor networks," *IEEE J. Sel. Areas Comm.*, vol. 23, pp. 809–819, Apr. 2005.
- [5] C. Pedersen, T. Pedersen, and B. H. Fleury, "A variational message passing algorithm for sensor self-localization in wireless networks," in *Proc. IEEE ISIT-11*, Saint Petersburg, Russia, pp. 2158–2162, Aug. 2011.
- [6] M. A. Caceres, F. Penna, H. Wymeersch, and R. Garello, "Hybrid cooperative positioning based on distributed belief propagation," *IEEE J. Sel. Areas Comm.*, vol. 29, pp. 1948–1958, Dec. 2011.
- [7] T. Sathyan and M. Hedley, "Fast and accurate cooperative tracking in wireless networks," *IEEE Trans. Mobile Comp.*, 2012, to appear.
- [8] O. Hlinka, O. Slučiak, F. Hlawatsch, P. M. Djurić, and M. Rupp, "Likelihood consensus and its application to distributed particle filtering," *IEEE Trans. Signal Process.*, vol. 60, pp. 4334–4349, Aug. 2012.
- [9] S. Farahmand, S. I. Roumeliotis, and G. B. Giannakis, "Set-membership constrained particle filter: Distributed adaptation for sensor networks," *IEEE Trans. Signal Process.*, vol. 59, pp. 4122–4138, Sep. 2011.
- [10] V. Savic, H. Wymeersch, and S. Zazo, "Belief consensus algorithms for fast distributed target tracking in wireless sensor networks," 2012. arXiv:1202.5261[cs.DC].
- [11] C. Taylor, A. Rahimi, J. Bachrach, H. Shrobe, and A. Grue, "Simultaneous localization, calibration, and tracking in an ad hoc sensor network," in *Proc. IPSN-06*, Nashville, TN, pp. 27–33, Apr. 2006.
- [12] S. Funiak, C. Guestrin, M. Paskin, and R. Sukthankar, "Distributed localization of networked cameras," in *Proc. IPSN-06*, Nashville, TN, pp. 34–42, Apr. 2006.
- [13] A. F. Garcia-Fernandez, M. R. Morelande, and J. Grajal, "Multitarget simultaneous localization and mapping of a sensor network," *IEEE Trans. Signal Process.*, vol. 59, pp. 4544–4558, Oct. 2011.
- [14] P. Oguz-Ekim, J. Gomes, J. Xavier, and P. Oliveira, "ML-based sensor network localization and tracking: Batch and time-recursive approaches," in *Proc. EUSIPCO-09*, Glasgow, Scotland, Aug. 2009.
- [15] X. Chen, A. Edelstein, Y. Li, M. Coates, M. G. Rabbat, and A. Men, "Sequential Monte Carlo for simultaneous passive device-free tracking and sensor localization using received signal strength measurements," in *Proc. IPSN-11*, Chicago, IL, pp. 342–353, Apr. 2011.
- [16] N. Kantas, S. Singh, and A. Doucet, "Distributed maximum likelihood for simultaneous self-localization and tracking in sensor networks," *IEEE Trans. Signal Process.*, vol. 60, pp. 5038–5047, Oct. 2012.
- [17] F. Meyer, E. Riegler, O. Hlinka, and F. Hlawatsch, "Simultaneous distributed sensor self-localization and target tracking using belief propagation and likelihood consensus," in *Proc. Asilomar Conf. Sig., Syst., Comput.*, Pacific Grove, CA, Nov. 2012. arXiv:1211.6988[cs.NI].
- [18] J. Lien, J. Ferner, W. Srichavengsup, H. Wymeersch, and M. Z. Win, "A comparison of parametric and sample-based message representation in cooperative localization," *Int. J. Navig. Observ.*, 2012.
- [19] X. R. Li and V. P. Jilkov, "Survey of maneuvering target tracking. Part I: Dynamic models," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 39, pp. 1333–1364, Oct. 2003.
- [20] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Upper Saddle River, NJ: Prentice-Hall, 1993.
- [21] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inf. Theory*, vol. 47, pp. 498–519, Feb. 2001.
- [22] Å. Björck, *Numerical Methods for Least Squares Problems*. Philadelphia, PA: SIAM, 1996.
- [23] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. Autom. Contr.*, vol. 49, pp. 1520–1533, Sep. 2004.
- [24] A. G. Dimakis, S. Kar, J. M. F. Moura, M. G. Rabbat, and A. Scaglione, "Gossip algorithms for distributed signal processing," *Proc. IEEE*, vol. 98, pp. 1847–1864, Nov. 2010.
- [25] G. Gan, C. Ma, and J. Wu, *Data Clustering: Theory, Algorithms, and Applications*. Philadelphia, PA: SIAM, 2007.
- [26] A. Doucet, N. De Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*. New York, NY: Springer, 2001.