Cooperative localization with information-seeking control

COOPERATIVE LOCALIZATION WITH INFORMATION-SEEKING CONTROL

Florian Meyer[1], *Henk Wymeersch*[2], *and Franz Hlawatsch*[1]

[1] Institute of Telecommunications, Vienna University of Technology, Vienna, Austria ({fmeyer,fhlawats}@nt.tuwien.ac.at)
[2] Department of Signals and Systems, Chalmers University of Technology, Gothenburg, Sweden (henkw@chalmers.se)

ABSTRACT

We propose a Bayesian method for cooperative localization and control in mobile agent networks. Distributed, cooperative self-localization of each agent is supported by an information-seeking control of the movement of the agents. For cooperative localization, the SPAWN message passing scheme is used. Cooperative control is achieved by maximizing the negative joint posterior entropy of the agent states via a gradient ascent. The localization part of our method provides the control part with sample-based probabilistic information. Simulation results demonstrate intelligent behavior of the agents and excellent localization accuracy.

***Index Terms*—** Agent networks, distributed estimation, cooperative localization, information-seeking control, belief propagation.

1. INTRODUCTION

Location-aware mobile agent networks are important in many applications including target tracking, pollution source localization, agricultural and healthcare monitoring, and chemical plume tracking [1–9]. In cooperative localization, each mobile agent estimates its own position based on measurements relative to other agents and the exchange of information with other agents. This is a nonlinear and, for large networks, high-dimensional distributed estimation problem. In a Bayesian estimation context, efficient belief propagation (BP) message passing methods can be used [4, 10, 11]. This is possible because the factor graph [12] corresponding to the posterior probability density function (pdf) of the total state matches the network's communication and measurement topology.

In many location-aware scenarios, it is advantageous to control certain properties of the agent network, such as the agent positions or the measurement characteristics ("controlled sensing") [3, 5, 8, 13]. In particular, here we will address the problem of combining distributed estimation and distributed control in mobile agent networks. We will limit our discussion to *information-seeking* control, which seeks to maximize the joint information carried by the measurements of all agents about the total state to be estimated (in our case, all the agent positions). Possible measures of information include negative posterior entropy [14–17] and scalar-valued functions of the Fisher information matrix [18]. However, existing methods for combined estimation and information-seeking control [14–18] are limited to simple sequential Bayesian filtering problems and are not suited to cooperative localization schemes that use message passing techniques based on a possibly loopy factor graph. In addition, [18] uses a Kalman filtering framework and therefore is not suited to highly nonlinear, non-Gaussian problems.

Here, we propose a Bayesian framework and method for distributed, cooperative, sequential localization with distributed information-seeking position control. For distributed localization, following [10] and [4], we use the SPAWN (sum-product algorithm over a wireless network) message passing scheme and sample representations of probability distributions. For distributed control, we define a global (holistic) objective function as the negative joint posterior entropy of all the agent positions conditioned on all the measurements. This objective function is optimized jointly by all agents via a gradient ascent. The localization part of our method provides the control part with sample-based probabilistic information. Our method advances beyond [14–17] in that (i) it constitutes a more general information-seeking control framework that uses SPAWN for distributed sequential estimation of multiple time-varying states, and (ii) it includes estimation of the own (controlled) positions of the agents, thus enabling its use for cooperative localization.

The remainder of this paper is organized as follows. In Section 2, we describe the system model and formulate the joint localization and control problem. The distributed cooperative localization technique used in our method is reviewed in Section 3. In Section 4, the proposed information-seeking control technique is described. Finally, simulation results are presented in Section 5.

2. SYSTEM MODEL AND PROBLEM FORMULATION

We consider a network of mobile agents $l \in \mathcal{A} \subset \mathbb{N}$. Fig. 1 depicts the overall system model and corresponding signal processing system relative to an arbitrary agent l. The *state* $\mathbf{x}_l^{(n)}$ of agent l at discrete time $n \in \{0, 1, \ldots\}$ consists of the agent's 2D position, i.e., $\mathbf{x}_l^{(n)} \triangleq \big[x_{l,1}^{(n)}, x_{l,2}^{(n)}\big]^{\mathrm{T}}$. The agent states evolve independently according to [19]

$$\mathbf{x}_l^{(n)} = \mathbf{x}_l^{(n-1)} + T_0\, \mathbf{u}_l^{(n)} + \sqrt{T_0}\, \mathbf{q}_l^{(n)}, \quad n = 1, 2, \ldots . \qquad (1)$$

Here, T_0 is the sampling interval, $\mathbf{u}_l^{(n)} \in \mathbb{R}^2$ is a controlled and hence deterministic velocity, which is constrained as $\|\mathbf{u}_l^{(n)}\| \le u_l^{\max}$,

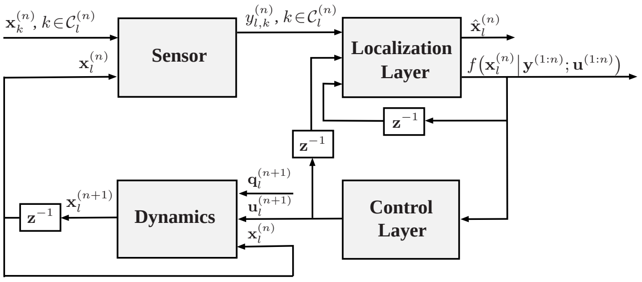

Fig. 1. Block diagram of the overall system model and signal processing system for an arbitrary agent $l \in \mathcal{A}$.

This work was supported by the Austrian Science Fund (FWF) under Grants S10603 and P27370 and by the European Commission under ERC Grant No. 258418 (COOPNET) and the Newcom# Network of Excellence in Wireless Communications.

and $\mathbf{q}_l^{(n)} \in \mathbb{R}^2$ is driving noise [20] that is independent across l and n. For correct interpretation of $\mathbf{u}_l^{(n)}$ within (1), it is assumed that the agents know the orientation of the global reference frame. The statistical relation between $\mathbf{x}_l^{(n-1)}$ and $\mathbf{x}_l^{(n)}$ as defined by (1) can also be described by the *state-transition pdf* $f\big(\mathbf{x}_l^{(n)} \big| \mathbf{x}_l^{(n-1)}; \mathbf{u}_l^{(n)}\big)$.

The communication and measurement topology of the network is described by neighborhood sets $\mathcal{C}_l^{(n)} \subseteq \mathcal{A} \backslash \{l\}$ as follows. Agent l communicates with agent k and acquires a measurement $y_{l,k}^{(n)}$ relative to agent k if $k \in \mathcal{C}_l^{(n)}$. This relation is symmetric, i.e., $k \in \mathcal{C}_l^{(n)}$ implies $l \in \mathcal{C}_k^{(n)}$. We consider noisy distance measurements

$$y_{l,k}^{(n)} = \|\mathbf{x}_l^{(n)} - \mathbf{x}_k^{(n)}\| + v_{l,k}^{(n)}, \quad k \in \mathcal{C}_l^{(n)}, \tag{2}$$

where $v_{l,k}^{(n)}$ is measurement noise that is assumed independent across l, k, and n. The statistical relation between $y_{l,k}^{(n)}$ and the involved states $\mathbf{x}_l^{(n)}$ and $\mathbf{x}_k^{(n)}$ is also described by the *local likelihood function* $f\big(y_{l,k}^{(n)} \big| \mathbf{x}_l^{(n)}, \mathbf{x}_k^{(n)}\big)$. In what follows, we denote by $\mathbf{x}^{(n)} \triangleq \big[\mathbf{x}_l^{(n)}\big]_{l \in \mathcal{A}}$, $\mathbf{u}^{(n)} \triangleq \big[\mathbf{u}_l^{(n)}\big]_{l \in \mathcal{A}}$, and $\mathbf{y}^{(n)} \triangleq \big[y_{l,k}^{(n)}\big]_{l \in \mathcal{A},\, k \in \mathcal{C}_l^{(n)}}$ the vectors of, respectively, all states, control vectors, and measurements at time n. Furthermore, we set $\mathbf{x}^{(1:n)} \triangleq \big[\mathbf{x}^{(1)\mathrm{T}}, \ldots, \mathbf{x}^{(n)\mathrm{T}}\big]^{\mathrm{T}}$, $\mathbf{u}^{(1:n)} \triangleq \big[\mathbf{u}^{(1)\mathrm{T}}, \ldots, \mathbf{u}^{(n)\mathrm{T}}\big]^{\mathrm{T}}$, and $\mathbf{y}^{(1:n)} \triangleq \big[\mathbf{y}^{(1)\mathrm{T}}, \ldots, \mathbf{y}^{(n)\mathrm{T}}\big]^{\mathrm{T}}$. We note that our framework and method can be extended to many other state-evolution and measurement models. For example, also the state-evolution model (1) may be nonlinear [21].

At each time n, the following two tasks are to be performed: (i) Each agent $l \in \mathcal{A}$ estimates its own state (position) $\mathbf{x}_l^{(n)}$ from prior information and $\mathbf{y}^{(1:n)}$, i.e., all past and present measurements in the entire network. (ii) The position of each agent is controlled such that the negative joint posterior entropy of all states in the network at the next time, conditioned on all measurements in the network at the next time, is maximized.

In the next two sections, we will present a distributed, recursive method for these tasks. Our method consists of a *localization layer* and a *control layer*, as shown in Fig. 1. In the localization layer, agent l computes an approximation of the marginal posterior pdf of $\mathbf{x}_l^{(n)}$ given all the past and present measurements and control vectors in the entire network, and a corresponding position estimate $\hat{\mathbf{x}}_l^{(n)}$. In the control layer, agent l uses the approximate marginal posteriors computed in the localization layer to calculate a quasi-optimal control vector $\mathbf{u}_l^{(n+1)}$.

3. LOCALIZATION LAYER

Bayesian estimation of the position $\mathbf{x}_l^{(n)}$ of agent $l \in \mathcal{A}$ from $\mathbf{y}^{(1:n)}$ is based on the posterior $f\big(\mathbf{x}_l^{(n)} \big| \mathbf{y}^{(1:n)}; \mathbf{u}^{(1:n)}\big)$, which is a marginal of the joint posterior $f\big(\mathbf{x}^{(1:n)} \big| \mathbf{y}^{(1:n)}; \mathbf{u}^{(1:n)}\big)$, i.e.,

$$f\big(\mathbf{x}_l^{(n)} \big| \mathbf{y}^{(1:n)}; \mathbf{u}^{(1:n)}\big) = \int f\big(\mathbf{x}^{(1:n)} \big| \mathbf{y}^{(1:n)}; \mathbf{u}^{(1:n)}\big)\, \mathrm{d}\mathbf{x}_{\sim l,n}^{(1:n)}. \tag{3}$$

Here, $\mathbf{x}_{\sim l,n}^{(1:n)}$ is $\mathbf{x}^{(1:n)}$ with $\mathbf{x}_l^{(n)}$ removed. The minimum mean-square error (MMSE) estimator [22] of $\mathbf{x}_l^{(n)}$ is then obtained as

$$\hat{\mathbf{x}}_{l,\mathrm{MMSE}}^{(n)} \triangleq \int \mathbf{x}_l^{(n)} f\big(\mathbf{x}_l^{(n)} \big| \mathbf{y}^{(1:n)}; \mathbf{u}^{(1:n)}\big)\, \mathrm{d}\mathbf{x}_l^{(n)}, \quad l \in \mathcal{A}. \tag{4}$$

Unfortunately, straightforward evaluation of (3) and (4) is computationally infeasible. However, using Bayes' rule and common assumptions [4], the joint posterior can be factorized as

$$\begin{aligned} &f\big(\mathbf{x}^{(1:n)} \big| \mathbf{y}^{(1:n)}; \mathbf{u}^{(1:n)}\big) \\ &\quad \propto \prod_{l \in \mathcal{A}} f\big(\mathbf{x}_l^{(0)}\big) \prod_{n'=1}^{n} f\big(\mathbf{x}_l^{(n')} \big| \mathbf{x}_l^{(n'-1)}; \mathbf{u}_l^{(n')}\big) \\ &\qquad\qquad \times \prod_{k \in \mathcal{C}_l^{(n')}} f\big(y_{l,k}^{(n')} \big| \mathbf{x}_l^{(n')}, \mathbf{x}_k^{(n')}\big). \end{aligned} \tag{5}$$

An approximation of the marginal posterior $f\big(\mathbf{x}_l^{(n)} \big| \mathbf{y}^{(1:n)}; \mathbf{u}^{(1:n)}\big)$ can then be obtained by executing sample-based SPAWN message passing [4, 23] on the factor graph corresponding to (5). As a result, samples $\big\{\mathbf{x}_l^{(n,j)}\big\}_{j=1}^{J}$ approximating $f\big(\mathbf{x}_l^{(n)} \big| \mathbf{y}^{(1:n)}; \mathbf{u}^{(1:n)}\big)$ and a corresponding approximation of (4),

$$\hat{\mathbf{x}}_l^{(n)} = \frac{1}{J} \sum_{j=1}^{J} \mathbf{x}_l^{(n,j)},$$

are available at agent l. A more detailed description of the method used in the localization layer is presented in [4, 21, 23].

4. CONTROL LAYER

4.1. Objective Function and Controller

According to our definition in Section 2, the vector comprising all measurements at the next time is $\mathbf{y}^{(n+1)} = \big[y_{l,k}^{(n+1)}\big]_{l \in \mathcal{A},\, k \in \mathcal{C}_l^{(n+1)}}$. However, to develop the controller, we formally replace in this definition $\mathcal{C}_l^{(n+1)}$ by $\mathcal{C}_l^{(n)}$ since at the current time n, the sets $\mathcal{C}_l^{(n+1)}$ are not yet known. Then, each agent $l \in \mathcal{A}$ calculates its next control variable $\mathbf{u}_l^{(n+1)}$ such that the global information about the next total state $\mathbf{x}^{(n+1)}$ given $\mathbf{y}^{(1:n+1)}$ is maximized. This information can be quantified by the negative conditional differential entropy [24, Chap. 8] of $\mathbf{x}^{(n+1)}$ given $\mathbf{y}^{(n+1)}$, with $\mathbf{y}^{(1:n)}$ included as an additional condition that has already been observed and is thus fixed:

$$\begin{aligned} &-h\big(\mathsf{x}^{(n+1)} \big| \mathsf{y}^{(n+1)}; \mathbf{y}^{(1:n)}, \mathbf{u}^{(1:n+1)}\big) \\ &\quad = \iint f\big(\mathbf{x}^{(n+1)}, \mathbf{y}^{(n+1)} \big| \mathbf{y}^{(1:n)}; \mathbf{u}^{(1:n+1)}\big) \\ &\qquad \times \log f\big(\mathbf{x}^{(n+1)} \big| \mathbf{y}^{(n+1)}, \mathbf{y}^{(1:n)}; \mathbf{u}^{(1:n+1)}\big)\, \mathrm{d}\mathbf{x}^{(n+1)} \mathrm{d}\mathbf{y}^{(n+1)}, \end{aligned} \tag{6}$$

where $\log$ denotes the natural logarithm. Note that we use a sans serif font for $\mathsf{x}^{(n+1)}$ and $\mathsf{y}^{(n+1)}$ in $h\big(\mathsf{x}^{(n+1)} \big| \mathsf{y}^{(n+1)}; \mathbf{y}^{(1:n)}, \mathbf{u}^{(1:n+1)}\big)$ in order to indicate that $h\big(\mathsf{x}^{(n+1)} \big| \mathsf{y}^{(n+1)}; \mathbf{y}^{(1:n)}, \mathbf{u}^{(1:n+1)}\big)$ depends on the *random vectors* $\mathsf{x}^{(n+1)}$ and $\mathsf{y}^{(n+1)}$, i.e., on their joint distribution but not on their values. Furthermore note that, within the total control vector $\mathbf{u}^{(1:n+1)}$ parametrizing the pdfs in (6), $\mathbf{u}^{(1:n)}$ has already been determined and is thus fixed.

According to (6), $-h\big(\mathsf{x}^{(n+1)} \big| \mathsf{y}^{(n+1)}; \mathbf{y}^{(1:n)}, \mathbf{u}^{(1:n+1)}\big)$ is a function of the next control vector $\mathbf{u}^{(n+1)}$, to be denoted as

$$D_h\big(\mathbf{u}^{(n+1)}\big) \triangleq -h\big(\mathsf{x}^{(n+1)} \big| \mathsf{y}^{(n+1)}; \mathbf{y}^{(1:n)}, \mathbf{u}^{(1:n+1)}\big). \tag{7}$$

This function will be used by each agent as the objective function for control. At each time n, toward a maximization of $D_h\big(\mathbf{u}^{(n+1)}\big)$, we perform one step of a gradient ascent [25] with reference vector $\mathbf{0}$. Thus, $\mathbf{u}^{(n+1)}$ is determined as

$$\hat{\mathbf{u}}^{(n+1)} = c^{(n+1)} \nabla D_h\big(\mathbf{u}^{(n+1)}\big)\big|_{\mathbf{u}^{(n+1)} = \mathbf{0}}, \tag{8}$$

where $c^{(n+1)} > 0$ is a step size. We have

$$\nabla D_h(\mathbf{u}^{(n+1)}) = \left[\frac{\partial D_h(\mathbf{u}^{(n+1)})}{\partial \mathbf{u}_l^{(n+1)}}\right]_{l\in\mathcal{A}}$$

since $\mathbf{u}^{(n+1)} = \big[\mathbf{u}_l^{(n+1)}\big]_{l\in\mathcal{A}}$. It then follows that the gradient ascent (8) is equivalent to separate local gradient ascents at the individual agents l, each performed only with respect to the respective local control vector $\mathbf{u}_l^{(n+1)}$, i.e.,

$$\hat{\mathbf{u}}_l^{(n+1)} = c_l^{(n+1)} \left.\frac{\partial D_h\big(\mathbf{u}^{(n+1)}\big)}{\partial \mathbf{u}_l^{(n+1)}}\right|_{\mathbf{u}^{(n+1)}=\mathbf{0}}, \quad l\in\mathcal{A}\,. \tag{9}$$

Note that, following [17], we allow for different local step sizes $c_l^{(n+1)}$ at the individual agents l. This deviation from (8) accounts for the possibly different bounds $u_l^{\max}$ and avoids the necessity of determining a common step size across all the agents. Each local step size $c_l^{(n+1)}$ is constrained by the condition $\|\hat{\mathbf{u}}_l^{(n+1)}\| \le u_l^{\max}$.

Next, we will derive a convenient expression of the gradient $\frac{\partial D_h(\mathbf{u}^{(n+1)})}{\partial \mathbf{u}_l^{(n+1)}}\Big|_{\mathbf{u}^{(n+1)}=\mathbf{0}}$ in (9). To simplify the notation, we no longer indicate the conditioning on $\mathbf{y}^{(1:n)}$ and $\mathbf{u}^{(1:n)}$ because at time $n+1$, $\mathbf{y}^{(1:n)}$ has already been observed and $\mathbf{u}^{(1:n)}$ has already been determined, hence both are fixed. Furthermore, we suppress the time index n and designate variables at time $n+1$ by the superscript "+"; for example, we write $h(\mathbf{x}^+|\mathbf{y}^+;\mathbf{u}^+)$ instead of $h\big(\mathbf{x}^{(n+1)}\big|\mathbf{y}^{(n+1)};\mathbf{y}^{(1:n)},\mathbf{u}^{(1:n+1)}\big)$. Finally, for calculating the gradient, following [16] and [17], we disregard the unknown driving noise $\mathbf{q}_l$ in (1) and thus rewrite (1) (with n replaced by $n+1$) as

$$\mathbf{x}_l^+ = \mathbf{x}_l + T_0\mathbf{u}_l^+, \tag{10}$$

which implies for the stacked vectors

$$\mathbf{x}^+ = \mathbf{x} + T_0\mathbf{u}^+. \tag{11}$$

Using (11) and the fact that $\mathbf{u}^+$ is deterministic, the objective function $D_h(\mathbf{u}^+) = -h(\mathbf{x}^+|\mathbf{y}^+;\mathbf{u}^+)$ in (7) can be expressed as [24, Chap. 8]

$$D_h(\mathbf{u}^+) = -h(\mathbf{x}|\mathbf{y}^+;\mathbf{u}^+) = -h(\mathbf{x}) + I(\mathbf{x};\mathbf{y}^+;\mathbf{u}^+)\,. \tag{12}$$

Here, $I(\mathbf{x};\mathbf{y}^+;\mathbf{u}^+)$ denotes the mutual information between $\mathbf{x}$ and $\mathbf{y}^+$ [24, Chap. 8] (with $\mathbf{u}^+$ being a deterministic parameter),

$$I(\mathbf{x};\mathbf{y}^+;\mathbf{u}^+) = \iint f(\mathbf{x},\mathbf{y}^+;\mathbf{u}^+)\,\log\frac{f(\mathbf{x},\mathbf{y}^+;\mathbf{u}^+)}{f(\mathbf{x})\,f(\mathbf{y}^+;\mathbf{u}^+)}\,d\mathbf{x}\,d\mathbf{y}^+.$$

Then, using (12) and the fact that $h(\mathbf{x})$ in (12) does not depend on $\mathbf{u}^+$, we obtain for the gradient

$$\begin{aligned}\frac{\partial D_h(\mathbf{u}^+)}{\partial \mathbf{u}_l^+} &= \frac{\partial I(\mathbf{x};\mathbf{y}^+;\mathbf{u}^+)}{\partial \mathbf{u}_l^+}\\ &= \iint \frac{\partial f(\mathbf{y}^+|\mathbf{x};\mathbf{u}^+)}{\partial \mathbf{u}_l^+}\,f(\mathbf{x})\,\log\frac{f(\mathbf{y}^+|\mathbf{x};\mathbf{u}^+)}{f(\mathbf{y}^+;\mathbf{u}^+)}\,d\mathbf{x}\,d\mathbf{y}^+,\end{aligned} \tag{13}$$

where the final expression follows by virtue of [17, Th. 1].

4.2. Sample-based Computation

We now present a cooperative computation of $\frac{D_h(\mathbf{u}^+)}{\partial \mathbf{u}_l^+}\Big|_{\mathbf{u}^+=\mathbf{0}}$ that uses importance sampling [26]. This computation requires communication with neighboring agents $k\in\mathcal{C}_l$ and uses the marginal posterior samples computed by the localization layer.

Due to the independence of the $v_{l,k}$ in (2), the likelihood function $f(\mathbf{y}^+|\mathbf{x};\mathbf{u}^+)$ occurring in (13) factorizes as

$$f(\mathbf{y}^+|\mathbf{x};\mathbf{u}^+) = \prod_{l\in\mathcal{A}}\prod_{k\in\mathcal{C}_l} f(\mathbf{y}_{l,k}^+|\mathbf{x}_l,\mathbf{x}_k;\mathbf{u}_l^+,\mathbf{u}_k^+)\,. \tag{14}$$

Here, because of (10), the local likelihood functions involved in (14) are given by

$$\begin{aligned}&f(\mathbf{y}_{l,k}^+|\mathbf{x}_l,\mathbf{x}_k;\mathbf{u}_l^+,\mathbf{u}_k^+)\\ &\quad= f(\mathbf{y}_{l,k}^+|\mathbf{x}_l^+,\mathbf{x}_k^+)\big|_{\mathbf{x}_l^+=\mathbf{x}_l+T_0\mathbf{u}_l^+,\,\mathbf{x}_k^+=\mathbf{x}_k+T_0\mathbf{u}_k^+}\,.\end{aligned} \tag{15}$$

Let $\alpha_l(\mathbf{y}^+,\mathbf{x},\mathbf{u}^+) \triangleq \prod_{k\in\mathcal{C}_l} f(\mathbf{y}_{l,k}^+|\mathbf{x}_l,\mathbf{x}_k;\mathbf{u}_l^+,\mathbf{u}_k^+)$ denote the part of the product (14) that depends on the local control vector $\mathbf{u}_l^+$. Then, using (14) and (15), the following sample-based approximation of (13) evaluated at $\mathbf{u}^+=\mathbf{0}$ can be derived [21]:

$$\begin{aligned}\left.\frac{\partial D_h(\mathbf{u}^+)}{\partial \mathbf{u}_l^+}\right|_{\mathbf{u}^+=\mathbf{0}} \approx\; &\frac{1}{JJ'}\sum_{j=1}^{J}\sum_{j'=1}^{J'}\frac{1}{\alpha_l(\mathbf{y}^{+(j,j')},\mathbf{x}^{(j)},\mathbf{0})}\\ &\times \left.\frac{\partial\alpha_l(\mathbf{y}^{+(j,j')},\mathbf{x}^{(j)},\mathbf{u}^+)}{\partial\mathbf{u}^+}\right|_{\mathbf{u}^+=\mathbf{0}}\\ &\times \log\frac{f(\mathbf{y}^{+(j,j')}|\mathbf{x}^{(j)};\mathbf{u}^+=\mathbf{0})}{f(\mathbf{y}^{+(j,j')};\mathbf{u}^+=\mathbf{0})}\,,\end{aligned}$$

where $f\big(\mathbf{y}^{+(j,j')};\mathbf{u}^+=\mathbf{0}\big)$ can in turn be approximated as

$$f\big(\mathbf{y}^{+(j,j')};\mathbf{u}^+=\mathbf{0}\big) \approx \frac{1}{J}\sum_{j''=1}^{J} f\big(\mathbf{y}^{+(j,j')}\big|\mathbf{x}^{(j'')};\mathbf{u}^+=\mathbf{0}\big)\,.$$

Here, $\mathbf{y}^{+(j,j')}$ and $\mathbf{x}^{(j)}$ are samples of $\mathbf{y}^+$ and $\mathbf{x}$, respectively that are drawn from the importance density [26] $q(\mathbf{y}^+,\mathbf{x}) \triangleq f(\mathbf{x})\,f(\mathbf{y}^+|\mathbf{x};\mathbf{u}^+=\mathbf{0})$ (note that $f(\mathbf{x})$ is short for $f(\mathbf{x}^{(n)}|\mathbf{y}^{(1:n)})$) via the following two-stage procedure:

1. Samples $\big\{\mathbf{x}^{(j)}\big\}_{j=1}^{J}$ are drawn from $f(\mathbf{x})$. This is done in a distributed way as follows. As a result of the localization layer, samples $\big\{\mathbf{x}_l^{(j)}\big\}_{j=1}^{J} \sim f(\mathbf{x}_l)$ are available at agent l. A flooding algorithm [27] is now used to make available to each agent l also the samples $\big\{\mathbf{x}_k^{(j)}\big\}_{j=1}^{J} \sim f(\mathbf{x}_k)$ of all the other agents $k\in\mathcal{A}\backslash\{l\}$. (The flooding algorithm requires each agent l to communicate with neighboring agents $k\in\mathcal{C}_l$.) Thus, at this point, all the sample sets $\big\{\mathbf{x}_k^{(j)}\big\}_{j=1}^{J}$, $k\in\mathcal{A}$ are available at each agent l. Then, samples $\big\{\mathbf{x}^{(j)}\big\}_{j=1}^{J} \sim f(\mathbf{x})$ can be obtained at each agent l via a simple stacking operation,[1] i.e., $\mathbf{x}^{(j)} = \big[\mathbf{x}_k^{(j)}\big]_{k\in\mathcal{A}}$ for $j=1,\ldots,J$.

2. For each sample $\mathbf{x}^{(j)}$, samples $\big\{\mathbf{y}^{+(j,j')}\big\}_{j'=1}^{J'}$ are drawn from the conditional pdf $f\big(\mathbf{y}^+\big|\mathbf{x}^{(j)};\mathbf{u}^+=\mathbf{0}\big)$. The method for doing this is based on the fact that, due to (2),

   $$\mathbf{y}^+ = \big[\|\mathbf{x}_{l'}^+ - \mathbf{x}_k^+\| + v_{l',k}^+\big]_{l'\in\mathcal{A},\,k\in\mathcal{C}_{l'}}\,. \tag{16}$$

   First, agent l obtains samples $\big\{\mathbf{x}_{l'}^{+(j)}\big\}_{j=1}^{J}$ for all $l'\in\mathcal{A}$ by evaluating (10) for $\mathbf{x}_{l'} = \mathbf{x}_{l'}^{(j)}$ and $\mathbf{u}_{l'}^+=\mathbf{0}$. Next, for each $j\in\{1,\ldots,J\}$, agent l draws samples $\big\{v_{l',k}^{+(j,j')}\big\}_{j'=1}^{J'} \sim f(v_{l',k}^+)$ for all $l'\in\mathcal{A}$ and $k\in\mathcal{C}_l$. Finally, agent l obtains the desired samples $\big\{\mathbf{y}^{+(j,j')}\big\}_{j'=1}^{J'} \sim f\big(\mathbf{y}^+\big|\mathbf{x}^{(j)};\mathbf{u}^+=\mathbf{0}\big)$ by evaluating (16) using the appropriate samples, i.e.,

   $$\mathbf{y}^{+(j,j')} = \big[\|\mathbf{x}_{l'}^{+(j)} - \mathbf{x}_k^{+(j)}\| + v_{l',k}^{+(j,j')}\big]_{l'\in\mathcal{A},\,k\in\mathcal{C}_{l'}}\,.$$

[1] This is based on the assumption that the $\mathbf{x}_l$ are conditionally independent given $\mathbf{y}^{(1:n)}$, i.e., $f(\mathbf{x}) = \prod_{l\in\mathcal{A}} f(\mathbf{x}_l)$. This assumption is also used in SPAWN [4], and thus also in the localization layer.

An alternative distributed implementation that uses consensus or gossip instead of flooding and an analysis of computation and communication costs can be found in [21].

5. SIMULATION RESULTS

We consider a scenario with one anchor agent $l = 1$ and three mobile agents $l = 2, 3, 4$. The anchor agent is static; it broadcasts its own (true) position to the mobile agents but does not perform any measurements. The driving noise $\mathbf{q}_l^{(n)}$ in (1) is zero-mean Gaussian with independent and identically distributed entries, i.e., $\mathbf{q}_l^{(n)} \sim \mathcal{N}(\mathbf{0}, \sigma_q^2 \mathbf{I})$. The agent network is fully connected. The sampling interval is $T_0 = 1$. Each mobile agent measures its distances to the other mobile agents and to the anchor agent according to (2). The measurement noise $v_{l,k}^{(n)}$ is zero-mean Gaussian with variance

$$\sigma_{l,k}^{(n)2} = \begin{cases} \sigma_0^2 \,, & \big\|\mathbf{x}_l^{(n)} - \mathbf{x}_k^{(n)}\big\| \leq d_0 \\ \sigma_0^2 \Big[\Big(\frac{\|\mathbf{x}_l^{(n)} - \mathbf{x}_k^{(n)}\|}{d_0} - 1 \Big)^{\kappa} + 1 \Big] , & \big\|\mathbf{x}_l^{(n)} - \mathbf{x}_k^{(n)}\big\| > d_0 \,. \end{cases}$$

That is, $\sigma_{l,k}^{(n)2}$ is a function of the distance $\big\|\mathbf{x}_l^{(n)} - \mathbf{x}_k^{(n)}\big\|$ that is constant up to d_0 and then increases polynomially with some exponent κ. This is a simple model for time-of-arrival distance measurements [28]. We set $\sigma_0^2 = 50$, $\kappa = 2$, and $d_0 = 50$.

In the localization layer, we use $J = 3600$ samples and the resampling scheme presented in [21]. We also use a censoring scheme [29] to reduce the number of samples and avoid numerical problems during the first time steps where the mobile agents still have uninformative beliefs. More specifically, only agents l with $\mathrm{tr}\big(\mathbf{C}_l^{(n)}\big) < 10$ are used as localization partners by the other agents. Here, $\mathbf{C}_l^{(n)}$ is a sample-based approximation of the covariance matrix of $f\big(\mathbf{x}_l^{(n)} \big| \mathbf{y}^{(1:n)}; \mathbf{u}^{(1:n)}\big)$ [21]. In the control layer, this censoring scheme corresponds to the following strategy: as long as agent l is not localized (i.e., $\mathrm{tr}\big(\mathbf{C}_l^{(n)}\big) \geq 10$), its objective function is $\tilde{D}_h\big(\mathbf{u}^{(n+1)}\big) \triangleq -h\big(\mathbf{x}_l^{(n+1)} \big| \mathrm{y}_{l,1}^{(n+1)}; y_{l,1}^{(1:n)}, \mathbf{u}_l^{(1:n+1)}\big)$, i.e., the negative differential entropy of only the own state conditioned on only the own measurement relative to the anchor agent, $\mathrm{y}_{l,1}^{(n+1)}$. The step sizes $c_l^{(n)}$ in (9) are adapted such that $\|\hat{\mathbf{u}}_l^{(n+1)}\| = u_l^{\max}$. Thus, each mobile agent l moves with maximum nominal speed (determined by $u_l^{\max}$) in the direction of maximum local increase of the objective function. The number of samples used in the control layer is $JJ' = 60000$, with $J = 1200$ and $J' = 50$. The three mobile agents have different start points ($[-50\,, 0]^{\mathrm{T}}$, $[0\,, -50]^{\mathrm{T}}$, and $[0\,, 70]^{\mathrm{T}}$ for $l =$ 2, 3, and 4, respectively) and different nominal speeds ($u_l^{\max} =$ 1, 0.3, and 0.1 for $l =$ 2, 3, and 4, respectively). Example trajectories of the mobile agents are shown in Fig. 2. The anchor agent is located at $[-60\,, 0]^{\mathrm{T}}$.

We compare the proposed method for cooperative self-localization with information-seeking control (abbreviated as C–C) with a method for noncooperative self-localization with information-seeking control (N–C) and a method for cooperative self-localization without intelligent control (C–N). In the N–C method, the mobile agents do not measure their distances and thus use only the measured distance to the anchor for self-localization. In the C–N method, the mobile agents cooperate in the localization layer but their control degenerates in that each mobile agent randomly chooses a direction initially and then moves in that direction with constant nominal speed determined by $u_l^{\max}$. Fig. 3 shows the average root-mean-square errors (ARMSEs) of the three methods, which were determined by averaging over the three mobile agents and over 300 simulation runs. It can be seen that the ARMSEs of the N–C and C–N methods decrease only very slowly whereas, after about 70 time steps, the ARMSE of the proposed C–C method decreases rather quickly to a low value. This can be explained as follows. Without cooperation (N–C) or without intelligent control (C–N), agents 3 and 4 need a long time to localize themselves because they are slow and initially far away from the anchor. On the other hand, agent 2 localizes itself very quickly because it is fast and initially close to the anchor. With cooperation and control (C–C), agent 2 moves in such a way that it supports the self-localization of the two other agents. In fact, as can be seen in Fig. 2, agent 2 first localizes itself by starting to turn around the anchor and then makes a sharp turn to approach agents 3 and 4, which helps them localize themselves. These results demonstrate the function and benefits of cooperative estimation with information-seeking control.

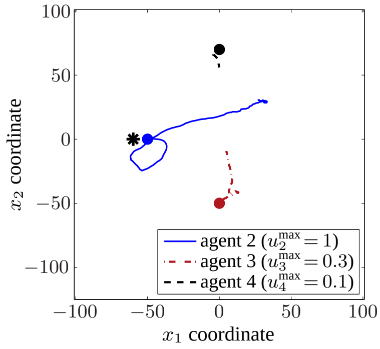

Fig. 2. Example trajectories of the mobile agents. The initial agent positions and the anchor position are indicated by bullets and a star, respectively.

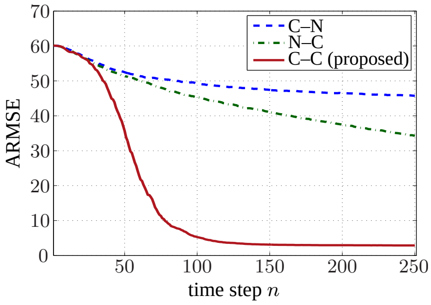

Fig. 3. ARMSE of the proposed localization/control method and two reference methods.

Simulation source code and animated plots are available at http://www.nt.tuwien.ac.at/about-us/staff/florian-meyer/.

6. CONCLUSION

We proposed a Bayesian framework and method for distributed, cooperative, sequential localization with information-seeking control in mobile agent networks. Localization is achieved by a sample-based SPAWN message passing scheme. The resulting sample representations of the marginal posterior pdfs of the agent positions are used by the controller to steer the movement of the agents. This is based on a criterion of maximal information jointly carried by the measurements of all the agents. Our main contribution is a cooperative sample-based scheme for calculating the control vector at each agent. Numerical simulations demonstrate intelligent agent behavior and substantial improvements of localization accuracy resulting from cooperation and information-seeking control.

7. REFERENCES

[1] A. Nayak and I. Stojmenović, *Wireless Sensor and Actuator Networks: Algorithms and Protocols for Scalable Coordination and Data Communication*. Hoboken, NJ: Wiley, 2010.

[2] F. Zhao and L. J. Guibas, *Wireless Sensor Networks: An Information Processing Approach*. Amsterdam, The Netherlands: Morgan Kaufmann, 2004.

[3] F. Bullo, J. Cortés, and S. Martínez, *Distributed Control of Robotic Networks: A Mathematical Approach to Motion Coordination Algorithms*. Princeton, NJ: Princeton University Press, 2009.

[4] H. Wymeersch, J. Lien, and M. Z. Win, "Cooperative localization in wireless networks," *Proc. IEEE*, vol. 97, pp. 427–450, Feb. 2009.

[5] P. Corke, T. Wark, R. Jurdak, W. Hu, P. Valencia, and D. Moore, "Environmental wireless sensor networks," *Proc. IEEE*, vol. 98, pp. 1903–1917, Nov. 2010.

[6] J. Ko, C. Lu, M. B. Srivastava, J. A. Stankovic, A. Terzis, and M. Welsh, "Wireless sensor networks for healthcare," *Proc. IEEE*, vol. 98, pp. 1947–1960, Nov. 2010.

[7] O. Hlinka, F. Hlawatsch, and P. M. Djuric, "Distributed particle filtering in agent networks: A survey, classification, and comparison," *IEEE Signal Process. Mag.*, vol. 30, pp. 61–81, Jan. 2013.

[8] T. Shima and S. Rasmussen, *UAV Cooperative Decision and Control: Challenges and Practical Approaches*. Philadelphia, PA: SIAM, 2009.

[9] T. Zhao and A. Nehorai, "Distributed sequential Bayesian estimation of a diffusive source in wireless sensor networks," *IEEE Trans. Signal Process.*, vol. 55, pp. 1511–1524, Apr. 2007.

[10] 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 Commun.*, vol. 23, pp. 809–819, Apr. 2005.

[11] T. Sathyan and M. Hedley, "Fast and accurate cooperative tracking in wireless networks," *IEEE Trans. Mobile Comput.*, vol. 12, pp. 1801–1813, Sep. 2013.

[12] 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.

[13] S. Haykin, *Cognitive Dynamic Systems: Perception-Action Cycle, Radar and Radio*. New York, NY: Cambridge University Press, 2012.

[14] B. Grocholsky, *Information-Theoretic Control of Multiple Sensor Platforms*. PhD thesis, University of Sydney, Sydney, Australia, 2002.

[15] A. D. Ryan, H. Durrant-Whyte, and J. K. Hedrick, "Information-theoretic sensor motion control for distributed estimation," in *Proc. IMECE '07*, Seattle, WA, Nov. 2007.

[16] G. M. Hoffmann and C. J. Tomlin, "Mobile sensor network control using mutual information methods and particle filters," *IEEE Trans. Autom. Control*, vol. 55, pp. 32–47, Jan. 2010.

[17] B. J. Julian, M. Angermann, M. Schwager, and D. Rus, "Distributed robotic sensor networks: An information-theoretic approach," *Int. J. Robot. Res.*, vol. 31, pp. 1134–1154, Sep. 2012.

[18] F. Morbidi and G. L. Mariottini, "Active target tracking and cooperative localization for teams of aerial vehicles," *IEEE Trans. Control Syst. Technol.*, vol. 21, pp. 1694–1707, Sep. 2013.

[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] H. Wymeersch, "The impact of cooperative localization on achieving higher-level goals," in *Proc. ICC '13*, Budapest, Hungary, Jun. 2013.

[21] F. Meyer, H. Wymeersch, M. Fröhle, and F. Hlawatsch, "Distributed estimation with information-seeking control in agent networks." submitted, 2014. Available online: http://arxiv.org/abs/1408.3732.

[22] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Upper Saddle River, NJ: Prentice-Hall, 1993.

[23] F. Meyer, O. Hlinka, H. Wymeersch, E. Riegler, and F. Hlawatsch, "Cooperative simultaneous localization and tracking in mobile agent networks." submitted, 2014. Available online: http://arxiv.org/abs/1403.1824.

[24] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York, NY: Wiley, 2006.

[25] R. Fletcher, *Practical Methods of Optimization*. New York, NY: Wiley, 1987.

[26] A. Doucet, N. De Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*. New York, NY: Springer, 2001.

[27] H. Lim and C. Kim, "Multicast tree construction and flooding in wireless ad hoc networks," in *Proc. MSWIM '00*, New York, NY, pp. 61–68, Aug. 2000.

[28] G. E. Garcia, L. S. Muppirisetty, E. M. Schiller, and H. Wymeersch, "On the trade-off between accuracy and delay in cooperative UWB localization: Performance bounds and scaling laws," *IEEE Trans. Wireless Commun.*, vol. 13, pp. 4574–4585, Aug. 2014.

[29] 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.