Distributed Spatio-Temporal Information Based Cooperative 3D Positioning in GNSS-Denied Environments

Yue Cao, Shaoshi Yang, *Senior Member, IEEE*, Zhiyong Feng, *Senior Member, IEEE*, Lihua Wang, and Lajos Hanzo, *Life Fellow, IEEE*,

***Abstract*—A distributed spatio-temporal information based cooperative positioning (STICP) algorithm is proposed for wireless networks that require three-dimensional (3D) coordinates and operate in the global navigation satellite system (GNSS) denied environments. Our algorithm supports any type of ranging measurements that can determine the distance between nodes. We first utilize a finite symmetric sampling based scaled unscented transform (SUT) method for approximating the nonlinear terms of the messages passing on the associated factor graph (FG) with high precision, despite relying on a small number of samples. Then, we propose an enhanced anchor upgrading mechanism to avoid any redundant iterations. Our simulation results and analysis show that the proposed STICP has a lower computational complexity than the state-of-the-art belief propagation based localizer, despite achieving an even more competitive positioning performance.**

***Index Terms*—3D, cooperative positioning, factor graph, scaled unscented transform (SUT), wireless localization.**

I. Introduction

LOCATION awareness plays a crucial role in many emerging applications [1] relying on wireless networks. Distributed cooperative positioning (CP) [2], [3] is a promising technique capable of providing location information for wireless networks operating in global navigation satellite system (GNSS) denied environments, such as forests, tunnels, and underground space. This is a challenging and important scenario for both civilian and military applications.

Distributed CP algorithms are usually based on Bayesian estimation, which treats the node positions as random variables and relies on each node to be localized (usually called *agent*) inferring its own position with the aid of both the internal information from its own hardware and external information gleaned from its cooperating nodes via message passing or belief propagation (BP). Non-parametric BP is a popular self-localization method in sensor networks [4], where the BP messages are approximated by a large number of particles, thus imposing excessive computational complexity. For designing a computationally efficient alternative, the authors of [5] proposed a parametric BP approach that utilizes a statistical linear regression (SLR) based affine function for approximating the nonlinear measurement functions. Due to the extra computation of SLR at each linearization iteration, the computational complexity remains high. For better modelling a distributed wireless network using its locality structure, a class of distributed CP algorithms based on the factor graph (FG) framework were proposed in [6]–[8], where the FG facilitates the evaluation of the marginal function of multivariate global functions more efficiently, hence they are more suitable for distributed implementations than traditional BP. However, as a non-parametric techniques, the so-called sum-product algorithm (SPA) over a wireless network (SPAWN) [6] also suffers from high computational complexity caused by massive sample points and iterative message computations. The authors of [7] employed the first-order Taylor expansion (TE) for replacing nonlinear terms in the BP messages of the parametric SPA, which however degrades the positioning accuracy. In [8], a three-dimensional (3D) universal cooperative localizer (3D-UCL) was proposed for vehicular networks, but no temporal information was exploited. Moreover, the existing CP algorithms usually depend on certain types of ranging observations. For example, both the cooperative least square (CLS) algorithm of [6] and the SPAWN schemes operate with the aid of time-of-arrival (TOA) and received-signal-strength (RSS).

Against the above backdrop, we propose a low-cost high-performance distributed spatio-temporal information based cooperative positioning (STICP) algorithm for 3D wireless networks operating in GNSS-denied environments. Our STICP has the following benefits: 1) it supports any type of ranging measurements that can determine the distance between nodes. Hence, it is more suitable for large-scale heterogeneous wireless networks than the existing CP algorithms. 2) It exhibits better performance despite its lower computational complexity than the 3D-UCL of [8], since both spatial and temporal information [6] are exploited at a much reduced number of samples. 3) It also outperforms the SPA-TE approach of [7], since we approximate the nonlinear terms of the messages passing on the FG by a method having higher accuracy at similar computational complexity, but at the cost of higher communication overhead. 4) Our STICP has a much lower

Y. Cao, S. Yang and Z. Feng are with the Key Laboratory of Universal Wireless Communications, Ministry of Education, and the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing, 100876, China (E-mails: caoyue, shaoshi.yang, fengzy@bupt.edu.cn).

L. Wang is with China Academy of Launch Vehicle Technology, Beijing 100076, China.

L. Hanzo is with School of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, U.K. (E-mail: lh@ecs.soton.ac.uk).

L. Hanzo would like to acknowledge the financial support of the Engineering and Physical Sciences Research Council project EP/P003990/1 (COALESCE) as well as of the European Research Council's Advanced Fellow Grant QuantCom (Grant No. 789028)

computational complexity and communication overhead than SPAWN, and supports a wider variety of ranging measurements than SPAWN, albeit at a marginal performance erosion. These benefits are achieved by using the scaled unscented transform (SUT) [9] for approximating the messages more efficiently and the enhanced anchor upgrading (EAU) for simplifying the iteration process. Both of analysis and simulation results have demonstrated the advantages of our STICP algorithm.

The rest of this paper is organized as follows. Section II introduces the system model and the formulation of the cooperative positioning problem. The details of the proposed STICP algorithm, and the analysis of its computational complexity and communication overhead, are presented in Section III. Our simulation results and discussions are given in Section IV, and finally the conclusions are drawn in Section V.

Notation: $\mathrm{E}\{\cdot\}$ and $\boldsymbol{C}_{(\cdot)}$ represent the expectation and covariance of a random variable, respectively; $\|\cdot\|$ denotes the Euclidean norm and $\delta(\cdot)$ represents the Dirac delta function; $a{:}b$ denotes the integer-valued vector of $[a, a+1, \cdots, b]$; $\sqrt{\ }$ represents the Cholesky decomposition; $(\cdot)_a$ represents the ath column of a matrix; and $(\cdot)^{\mathrm{T}}$ represents the transpose. Finally, we denote by $p(\cdot)$ the probability density function (PDF).

II. System Model and Problem Formulation

Consider a wireless network composed of N agents whose positions are unknown and yet to be estimated, as well as A anchors whose positions are known *a priori* as reference. We assume that the transmission time is slotted, and the agents can move independently from their positions at time slot $(t-1)$ to new positions at time slot t. In the 3D space, the position of node i at time slot t is denoted by $\boldsymbol{x}_i^t = [x_i^t, y_i^t, z_i^t]^{\mathrm{T}}$ $(i = 0, \cdots, N+A-1)$. Denote the set of anchors from which agent i receives signals during time slot t by $\mathbb{A}^t_{\to i}$, and the set of agents from which agent i receives signals during time slot t by $\mathbb{U}^t_{\to i}$.

At time slot t, agent i is able to obtain both the external measurements from its neighbors, and the internal measurements based on its own hardware. Denote all the internal and external measurements collected by all the agents at time slot t as the matrix $\boldsymbol{Z}^t$. Note that $\boldsymbol{Z}^t$ can be broken up into the matrices $\boldsymbol{Z}^t_{\text{self}}$ and $\boldsymbol{Z}^t_{\text{rel}}$, where $\boldsymbol{Z}^t_{\text{self}}$ consists of all the internal measurements of all the agents, and $\boldsymbol{Z}^t_{\text{rel}}$ consists of all the external measurements of all the agents relative to their individual neighbors. Additionally, the dimension of matrix $\boldsymbol{Z}^t$, $\boldsymbol{Z}^t_{\text{self}}$ and $\boldsymbol{Z}^t_{\text{rel}}$ is $\gamma n \times (N + N_{\text{rel}}N)$, $\gamma n \times N$ and $\gamma n \times N_{\text{rel}}N$, respectively, where n denotes the dimension of the agent position vector (i.e., 2 and 3 for 2D and 3D positioning, respectively), N_{rel} denotes the number of neighbor nodes. and γ represents the types of measurements considered. More specifically, we have $\gamma = 1$ if only a single type of information is measured (e.g., position); we have $\gamma = 2$ if two types of information are jointly measured (e.g., position and velocity); and we have $\gamma = 3$ if three types of information are jointly measured (e.g., position, velocity and acceleration). In this paper, we only consider the measurement of the 3D position information, hence we have $\gamma = 1$ and $n = 3$.

The noise-contaminated ranging measurement[1] from node j to agent i at time slot t can be written as:

$$z^t_{j\to i} = d^t_{ij} + e_{j\to i}, \tag{1}$$

where d^t_{ij} is the Euclidean distance between node j and agent i at time slot t, and $z^t_{j\to i}$ can be measured in a variety of ways, such as TOA, angle-of-arrival (AOA) and RSS, to name but a few. Furthermore, $e_{j\to i} \sim \mathcal{N}\left(0, \sigma^2_{j\to i}\right)$ represents the measurement error that obeys the Gaussian distribution with zero-mean and variance $\sigma^2_{j\to i}$. The goal of agent i is to estimate its position $\boldsymbol{x}_i^t$ at time slot t, given only the external and internal measurements up to time slot t, i.e., $p(\boldsymbol{x}_i^t|\boldsymbol{z}^{0:t}_{i,\text{self}}, \boldsymbol{z}^{0:t}_{i,\text{rel}})$.

We assume that agent i knows the following information: i) the *a priori* distribution $p\left(\boldsymbol{x}_i^0\right) \sim \mathcal{N}(\mathrm{E}\{\boldsymbol{x}_i^0\}, \boldsymbol{C}_{\boldsymbol{x}_i^0})$ at time slot 0, where $\mathrm{E}\{\boldsymbol{x}_i^0\} = [\mathrm{E}\{x_i^0\}, \mathrm{E}\{y_i^0\}, \mathrm{E}\{z_i^0\}]$ and $\boldsymbol{C}_{\boldsymbol{x}_i^0} = \text{diag}(\sigma^2_{x_i^0}, \sigma^2_{y_i^0}, \sigma^2_{z_i^0})$ represent the expectation and the covariance matrix of $\boldsymbol{x}_i^0$, respectively; ii) the mobility model characterized by the conditional distribution $p\left(\boldsymbol{x}_i^t|\boldsymbol{x}_i^{t-1}\right)$ at any time slot t; iii) the internal measurements $\boldsymbol{z}^t_{i,\text{self}}$ at any time slot t and the corresponding likelihood function $p\left(\boldsymbol{z}^t_{i,\text{self}}|\boldsymbol{x}_i^{t-1}, \boldsymbol{x}_i^t\right)$ at any time slot t; iv) agent i gets other information by data delivery over the wireless network.

III. The Proposed STICP Algorithm

A. Derivation of the SUT aided STICP algorithm

We first factorize $p(\boldsymbol{x}_i^{0:t}|\boldsymbol{Z}^{1:t})$ as:

$$\begin{aligned} p(\boldsymbol{x}_i^{0:t}|\boldsymbol{Z}^{1:t}) \ \propto p\left(\boldsymbol{x}_i^0\right) \prod_{\tau=1}^{t} \left\{ p\left(\boldsymbol{x}_i^\tau|\boldsymbol{x}_i^{\tau-1}\right) \right. \\ \left. \times p\left(\boldsymbol{z}^\tau_{i,\text{self}}|\boldsymbol{x}_i^\tau, \boldsymbol{x}_i^{\tau-1}\right) p\left(\boldsymbol{Z}^\tau_{\text{rel}}|\boldsymbol{x}_i^\tau\right) \right\}. \end{aligned} \tag{2}$$

The Forney-style FG [10] of $p(\boldsymbol{x}_i^{0:t}|\boldsymbol{Z}^{1:t})$ has a structure illustrated in Fig.1. For each factor, we create a vertex (drawn as a rectangle), and for each variable we create an edge (drawn as a line). When a variable appears in a factor, we connect the edge to the vertex. When a variable appears in more than two factors, an equality vertex is created. For example, the variable $\boldsymbol{x}_i^t$ appears in factors $\phi_{i\to j}(\boldsymbol{x}_i^t, \boldsymbol{x}_j^t)$, $\phi_{j\to i}(\boldsymbol{x}_i^t, \boldsymbol{x}_j^t)$, $\phi_{k\to i}(\boldsymbol{x}_i^t, \boldsymbol{x}_k^t)$, $f_i^{t|t-1}(\boldsymbol{x}_i^t, \boldsymbol{x}_i^{t-1})$, and $f_i^{t+1|t}(\boldsymbol{x}_i^t, \boldsymbol{x}_i^{t+1})$, thus we create an equality vertex and label it "=". Additionally, when agent i performs external measurement relative to its neighbour node j at time slot t, we created a factor $\phi_{j\to i}(\cdot)$ which is local to agent i and is a function of the ranging measurement $z^t_{j\to i}$. We then run an iterative SPA on the above FG, as detailed below. The *a posteriori* distribution (usually called *belief*) concerning the x-component of the position vector of agent i represents the x-component of the message broadcast by agent i at iteration l and time slot t, i.e., $b^l(x_i^t)$, and it satisfies:

$$\begin{aligned} b^l(x_i^t) \ \propto \mu_{f_i^{(t|t-1)}(\cdot)\to x_i^t}(\cdot) \prod_{k\in\mathbb{A}^t_{\to i}} \mu^l_{\phi_{k\to i}(\cdot)\to x_i^t}(\cdot) \\ \prod_{j\in\mathbb{U}^t_{\to i}} \mu^l_{\phi_{j\to i}(\cdot)\to x_i^t}(\cdot), \end{aligned} \tag{3}$$

[1] All the links considered are line-of-sight (LOS), while the non-line-of-sight (NLOS)/LOS mixed environment will be considered in our future work.

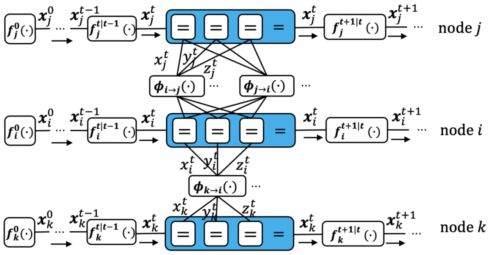


Fig. 1. FG of $p\left(\boldsymbol{x}^{0:t}|\boldsymbol{Z}^{1:t}\right)$, where nodes $i,j \in \mathbb{U}^t_{\to i}$, and node $k \in \mathbb{A}^t_{\to i}$. The arrows represent the temporal flow of the messages passed at different time slots inside a single node (from past to present). We use the following notations: $f_i^0(\boldsymbol{x}_i^0) = p\left(\boldsymbol{x}_i^0\right)$, $f_i^{t|t-1}(\boldsymbol{x}_i^t, \boldsymbol{x}_i^{t-1}) = p\left(\boldsymbol{x}_i^t|\boldsymbol{x}_i^{t-1}\right) p\left(\boldsymbol{z}_{i,\text{self}}^t|\boldsymbol{x}_i^{t-1}, \boldsymbol{x}_i^t\right)$ and $\phi_{i\to j}(\boldsymbol{x}_i^t, \boldsymbol{x}_j^t) = p\left(z^t_{i\to j}|\boldsymbol{x}_i^t, \boldsymbol{x}_j^t\right)$.

where $\mu_{f_i^{(t|t-1)}(\cdot)\to x_i^t}(\cdot)$ is the message passed from factor $f_i^{(t|t-1)}(\cdot)$ to variable x_i^t and it is the x-component temporal information of agent i satisfying:

$$\mu_{f_i^{(t|t-1)}(\cdot)\to x_i^t}(\cdot) \;\propto\; b^{l_{\max}}(x_i^{t-1}) f_i^{t|t-1}(\cdot), \tag{4}$$

with $l_{\max}$ representing the maximum number of iterations at time slot $t-1$, while $\prod_{j\in\mathbb{U}^t_{\to i}} \mu^l_{\phi_{j\to i}(\cdot)\to x_i^t}(\cdot)$ and $\prod_{k\in\mathbb{A}^t_{\to i}} \mu^l_{\phi_{k\to i}(\cdot)\to x_i^t}(\cdot)$ are the spatial ranging information satisfying:

$$\mu^l_{\phi_{k\to i}(\cdot)\to x_i^t}(\cdot) \;\propto\; \iiint \phi_{k\to i}(\cdot)\mu^l_{x_k^t\to\phi_{k\to i}(\cdot)}(\cdot) d\boldsymbol{x}_k^t dy_i^t dz_i^t, \tag{5}$$

and

$$\mu^l_{\phi_{j\to i}(\cdot)\to x_i^t}(\cdot) \;\propto\; \iiint \phi_{j\to i}(\cdot)\mu^l_{x_j^t\to\phi_{j\to i}(\cdot)}(\cdot) d\boldsymbol{x}_j^t dy_i^t dz_i^t, \tag{6}$$

respectively. In (5), the message passed from variable x_k^t to factor $\phi_{k\to i}(\cdot)$, i.e., the belief of node k broadcast at time slot t and iteration l can be written as:

$$\mu^l_{x_k^t\to\phi_{k\to i}(\cdot)}(\cdot) = b^l(x_k^t) = \delta(x_k^t - \mathrm{E}\{x_k^t\}), \tag{7}$$

where

$$\phi_{k\to i}(\cdot) = \frac{1}{\sqrt{2\pi\sigma^2_{k\to i}}} \exp\left\{-\frac{\left(z^t_{k\to i} - \|\boldsymbol{x}_i^t - \boldsymbol{x}_k^t\|\right)^2}{2\sigma^2_{k\to i}}\right\}. \tag{8}$$

For brevity, we define the exponential function in (8) as h_k, where $k \in \mathbb{A}^t_{\to i} \cup \mathbb{U}^t_{\to i}$. Then substituting (7) and (8) into (5) leads to:

$$\mu^l_{\phi_{k\to i}(\cdot)\to x_i^t}(\cdot) \propto \iiint h_k d\boldsymbol{x}_k^t dy_i^t dz_i^t. \tag{9}$$

Unfortunately, (9) involves integrals and it is difficult to obtain closed-form expressions for (9) due to the term h_k, which is a nonlinear function of $\boldsymbol{x}_i^t$. As a remedy, we approximate the PDF of h_k by employing the SUT technique, which can be used to estimate the result of applying a given nonlinear transformation to the likelihood function of the measurements by picking a minimal set of samples (called *sigma points*) around the mean. Then the sigma points are used for replacing the $\boldsymbol{x}_i^t$ in the nonlinear term h_k to form the new mean and variance estimates of h_k. In general, it is computationally efficient and convenient to generate a symmetric set of $2n$+1 sigma points to define a discrete distribution having a given mean and covariance in n dimensions [9]. First, we draw N_s=$2n$+1=7 samples $\{\boldsymbol{v}_a\}_{a=0}^{6}$ from the PDF of $\boldsymbol{x}_i^t$, thus we have:

$$\boldsymbol{v}_0 = \mathrm{E}\{\boldsymbol{x}_i^t\}, \quad a = 0; \tag{10}$$

$$\boldsymbol{v}_a = \mathrm{E}\{\boldsymbol{x}_i^t\} + \left(\sqrt{(n+\lambda)\boldsymbol{C}_{\boldsymbol{x}_i^t}}\right)_a, \tag{11}$$

when $a = 1,2,3$;

$$\boldsymbol{v}_a = \mathrm{E}\{\boldsymbol{x}_i^t\} - \left(\sqrt{(n+\lambda)\boldsymbol{C}_{\boldsymbol{x}_i^t}}\right)_{a-3}, \tag{12}$$

when $a = 4,5,6$; and

$$\lambda = 3\cdot\left(\alpha^2 - 1\right), \tag{13}$$

where α is a small positive number which controls the distribution of the samples. Then, assign a weight to each sample by:

$$w_0^m = \frac{\lambda}{n+\lambda}, \tag{14}$$

$$w_0^c = \frac{\lambda}{n+\lambda} + \left(1 - \alpha^2 + \beta\right), \tag{15}$$

$$w_a^m = w_a^c = \frac{1}{2(n+\lambda)}, \quad a = 1,2,\ldots,6, \tag{16}$$

where w_a^m and w_a^c denote the weight assigned to the mean and covariance of the ath sample, respectively; β is a non-negative number and chosen empirically as 2.

Similar to $\boldsymbol{x}_i^t$ in h_k, we perform the same operation on $\{\boldsymbol{v}_a\}_{a=0}^{2n}$ and obtain $\boldsymbol{g}$=$\{g_a\}_{a=0}^{6}$:

$$g_a = \exp\left\{-\frac{\left(z^t_{k\to i} - \|\boldsymbol{v}_a - \boldsymbol{x}_k^t\|\right)^2}{2\sigma^2_{k\to i}}\right\}. \tag{17}$$

Then the mean and variance of h can be obtained upon using the above SUT, and they satisfy:

$$\mathrm{E}\{h_k\} = \sum_{a=0}^{6} w_a^m g_a, \tag{18}$$

$$\sigma^2_{h_k} = \sum_{a=0}^{6} w_a^c (g_a - \mathrm{E}\{h_k\})^2. \tag{19}$$

Thus the message $\mu^l_{\phi_{k\to i}(\cdot)\to x_i^t}(\cdot)$ is given by:

$$\mu^l_{\phi_{k\to i}\to x_i^t}(\cdot) \propto \mathcal{N}(\mathrm{E}\{h_k\}, \sigma^2_{h_k}). \tag{20}$$

For brevity, we further denote the message $\mu^l_{\phi_{m\to i}(\cdot)\to x_i^t}(\cdot)$ by μ^l_m ($m \in \mathbb{A}^t_{\to i} \cup \mathbb{U}^t_{\to i}$). Moreover, the above derivation also applies to message μ^l_j:

$$\mu^l_{\phi_{j\to i}(\cdot)\to x_i^t}(\cdot) = \mu^l_j \propto \mathcal{N}\left(\mathrm{E}\{\mu^l_j\}, (\sigma_{\mu^l_j})^2\right), \tag{21}$$

where

$$\mathrm{E}\{\mu^l_j\} = (\sigma_{\mu^l_j})^2 \left(\frac{\mathrm{E}\{h_j\}}{(\sigma_{h_j})^2} + \frac{\mathrm{E}^l\{x_j^t\}}{(\sigma^l_{x_j^t})^{-2}}\right), \tag{22}$$

and

$$(\sigma_{\mu^l_j})^2 = \left((\sigma_{h_j})^{-2} + (\sigma^l_{x_j^t})^{-2}\right)^{-1}. \tag{23}$$

Upon substituting (4), (20) and (21) into (3) we obtain:

$$p(x_i^t|\boldsymbol{Z}^t) = b^{l_{max}}(x_i^t) \propto \mathcal{N}(\mathrm{E}\{x_i^t|\boldsymbol{Z}^t\}, \sigma^2_{x_i^t|\boldsymbol{Z}^t}), \tag{24}$$

where

$$\begin{aligned}\mathrm{E}\{x_i^t|\boldsymbol{Z}^t\} = \left(\sigma_{x_i^t|\boldsymbol{Z}^t}\right)^2 \Bigg(& \frac{\hat{x}_i^{t|t-1}}{\left(\hat{\sigma}_{i,x}^{t|t-1}\right)^2} + \sum_{j\in\mathbb{U}^t_{\to i}} \frac{\mathrm{E}\left\{\mu_j^{l_{\max}}\right\}}{\left(\sigma_{\mu_j^{l_{\max}}}\right)^2} \\ & + \sum_{k\in\mathbb{A}^t_{\to i}} \frac{\mathrm{E}\left\{h_k\right\}}{\left(\sigma_{h_k}\right)^2}\Bigg),\end{aligned} \tag{25}$$

and

$$\begin{aligned}\left(\sigma_{x_i^t|\boldsymbol{Z}^t}\right)^2 = \Bigg(& \frac{1}{\left(\hat{\sigma}_{i,x}^{t|t-1}\right)^2} + \sum_{j\in\mathbb{U}^t_{\to i}} \frac{1}{\left(\sigma_{\mu_j^{l_{\max}}}\right)^2} \\ & + \sum_{k\in\mathbb{A}^t_{\to i}} \frac{1}{\left(\sigma_{h_k}\right)^2}\Bigg)^{-1}.\end{aligned} \tag{26}$$

At any time slot t, each agent can determine the minimum mean squared error (MMSE) based estimate of the x-component of its own position by taking the mean of x_i^t, as follows:

$$\hat{x}_i^t = \int x_i^t b^{l_{\max}}\left(x_i^t\right) \mathrm{d}x_i^t, \tag{27}$$

while $\hat{y}_i^t$ and $\hat{z}_i^t$ can be obtained in a similar manner. Then the resultant STICP is summarized in Algorithm 1.

Algorithm 1 STICP

```
Require: The a priori distribution $p(\boldsymbol{x}_i^0)$, the local mobility model
    $p(\boldsymbol{x}_i^t|\boldsymbol{x}_i^{t-1})$, the local likelihood function $p(\boldsymbol{z}_{i,\text{self}}^t|\boldsymbol{x}_i^{t-1}, \boldsymbol{x}_i^t)$
Ensure: The distribution $\mathrm{E}\{\boldsymbol{x}_i^t|\boldsymbol{Z}^t\}$ and $\boldsymbol{C}_{\boldsymbol{x}_i^t|\boldsymbol{Z}^t}$
  for node $i \in \mathbb{U}$ do
    initialize $b^0(\boldsymbol{x}_i^t) = \mu_{f_i^{t|t-1}(\cdot)\to\boldsymbol{x}_i^t}(\cdot)$.
    for iteration $l$ = 1 to $l_{\max}$ do
      broadcast $b^{l-1}(\boldsymbol{x}_i^t)$.
      receive $b^{l-1}(\boldsymbol{x}_j^t)$.
      using (3) to calculate the belief.
      obtain $\mathrm{E}\{\boldsymbol{x}_i^t|\boldsymbol{Z}^t\}$ and $\boldsymbol{C}_{\boldsymbol{x}_i^t|\boldsymbol{Z}^t}$ by (25) and (26).
      using (27) to estimate the position of node.
    end for
  end for
```

B. Acceleration of STICP

Based on the change of the estimated position in two consecutive iterations of the proposed STICP algorithm, we propose an EAU technique for further reducing the computational complexity of STICP, by upgrading the agents that meet certain conditions to anchors and by reducing the redundant iterations. The EAU consists of anchor upgrading and iteration reduction. We set two positive thresholds η_1 and η_2, satisfying $\eta_1 < \eta_2$, for determining whether the value of the expectation of all components of the position vector, i.e., $\mathrm{E}\{\boldsymbol{x}_i^t|\boldsymbol{Z}^t\}$, has met the particular threshold to activate anchor upgrading, or whether the value of the expectation of any components of the position vector, has met the particular threshold to activate iteration reduction.

TABLE I
COMPARISON OF THE COMPUTATIONAL COMPLEXITY, THE COMMUNICATION OVERHEAD, AND THE NUMBER OF SAMPLES (N_{S})

Algorithm	computational complexity	Communication overhead	N_{s}
STICP	$\mathcal{O}(N_{\mathrm{rel}})$	14	7
SPA-TE	$\mathcal{O}(N_{\mathrm{rel}})$	2	N/A
3D-UCL	$\mathcal{O}(N_{\mathrm{s}} \cdot N_{\mathrm{rel}})$	$\mathcal{O}(N_{\mathrm{rel}})$	Large
SPAWN	$\mathcal{O}(N_{\mathrm{s}}^2 \cdot N_{\mathrm{rel}}) + \mathcal{O}(N_{\mathrm{s}})$	$\mathcal{O}(N_{\mathrm{s}})$	Large

1) Anchor upgrading: After several iterations in time slot t, the individual position estimates of some agents become converged and remain almost unchanged in the subsequent iterations. In particular, if the change of any component of the position vector is less than a given threshold, e.g., η_1, it is no longer necessary to continue updating the *belief* concerning the particular component of the position vector of agent i in this time slot. More specifically, if

$$|\mathrm{E}^{l+1}\{x_i^t|\boldsymbol{Z}^t\} - \mathrm{E}^{l}\{x_i^t|\boldsymbol{Z}^t\}| < \eta_1 \tag{28}$$

is satisfied at time slot t, then we set $b^{l+}(x_i^t) = b^l(x_i^t), l_+ = l+2, ..., l_{\max}$. Similar operations also apply to y_i^t and z_i^t. When the change of all the three components of the position vector, namely when the change of $\mathrm{E}\{\boldsymbol{x}_i^t|\boldsymbol{Z}^t\}$, is less than the threshold η_1 in two consecutive iterations, the agent i is upgraded as a pseudo-anchor. As a result, the number of agents in the wireless network can be decreased, then it becomes unnecessary to conduct ranging measurements among the pseudo-anchors as well as between a pseudo-anchor and an anchor. Hence the complexity of cooperative positioning is reduced.

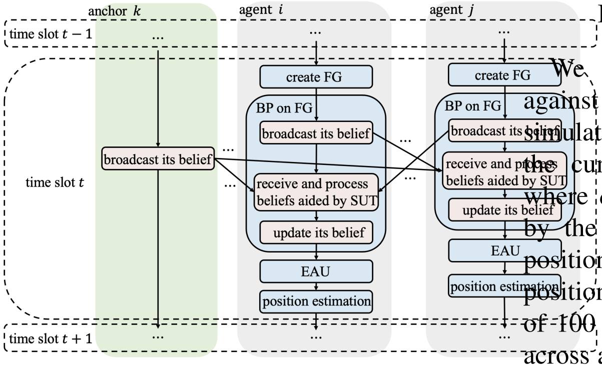
2) Iteration reduction: In order to further reduce the computational complexity, an iteration reduction technique is invoked. At time slot t, if $\mathrm{E}\{x_i^t|\boldsymbol{Z}^t\}$ changes less than the threshold η_2 but more than the threshold η_1 in two consecutive iterations of STICP, then the update of the *belief* will terminate at the next iteration. Otherwise, the belief is updated normally. More specifically, if

$$\eta_1 < |\mathrm{E}^{l+1}\{x_i^t|\boldsymbol{Z}^t\} - \mathrm{E}^{l}\{x_i^t|\boldsymbol{Z}^t\}| < \eta_2, \tag{29}$$

then we set $b^{l+2}(x_i^t) = b^{l+1}(x_i^t)$. Therefore, the number of samples is decreased and the computational complexity can be substantially reduced, despite possibly at the cost of marginally increased positioning error, as demonstrated by the subsequent analysis and simulation results. Additionally, η_1 and η_2 are chosen in the range of 0.05~0.25 and 0.5~0.8, respectively, based on our experimental experiences. Note that a too large value of either η_1 or η_2 leads to difficulty of convergence, and a too small value leads to unsatisfactory computational speedup.

C. Computational Complexity and Communication Overhead

We compare our STICP, the SPA-TE, 3D-UCL and SPAWN in terms of the computational complexity and the communication overhead in Table.I. The communication overhead is evaluated in terms of the number of parameters the agent is expected to broadcast during each iteration. Since the above algorithms are fully distributed, we only consider the



framework.png

Fig. 2. The algorithmic operations conducted on the agent i at time slot t.

agent i. The computational complexity and the communication overhead per SPA-TE [7] iteration are on the order of $\mathcal{O}(N_{\text{rel}})$ and $\mathcal{O}(N_{\text{s}})$ [7], respectively, where N_{rel} is the averaged number of cooperative agents. For SPAWN having N_{s} samples, the computational complexity and the communication overhead per iteration are on the order of $\mathcal{O}(N_{\text{s}}^2 \cdot N_{\text{rel}}) + \mathcal{O}(N_{\text{s}})$ and $\mathcal{O}(N_{\text{s}})$ [11], respectively. With regard to the 3D-UCL, the computational complexity and the communication overhead per iteration are $\mathcal{O}(N_{\text{s}} \cdot N_{\text{rel}})$ and $\mathcal{O}(N_{\text{rel}})$, respectively [8]. For the proposed STICP, all the messages are broadcast and updated as Gaussian distribution, so the number of operations is only related to the number of neighbor nodes N_{rel}, which is $\mathcal{O}(N_{\text{rel}})$. Furthermore, the communication overhead is quantified as 14 (parameters to be broadcast) since only the approximate mean and the covariance of the seven sigma points have to be broadcast per iteration. Note that the metric "the number of samples" is not applicable to SPA-TE, because it utilizes the first-order Taylor expansion, instead of a sample-based approach, to approximate the nonlinear term in the messages. Additionally, the number of samples coming from the SUT step used in STICP is seven, while both SPAWN and 3D-UCL rely on a large number of samples.

For clarity, the logical framework of our STICP is illustrated in Fig. 2. Note that the spatio information, namely the position and ranging information of the spatially distributed nodes, is exchanged between the anchors and the agents, which will affect the traffic in a wireless network [12]. Meanwhile, the temporal information is the position and ranging information related to any node at different time slots, thus it is local to each node and not exchanged among the nodes. On the other hand, when a distributed wireless network, such as an ad hoc network, is considered, the centralized scheduling is usually infeasible and an random access mechanism is typically used. However, it is also possible to assume that the network has a central scheduler, which may be independent of the anchors and agents, or the anchors may act as the hosts of the scheduler (e.g., multiple base stations can serve as the anchors and host the scheduler). In such a case, we can rely on the contributions of [12] to analyze the impact of wireless traffic imposed by the cooperative positioning algorithm.

IV. Simulation Results and Discussions

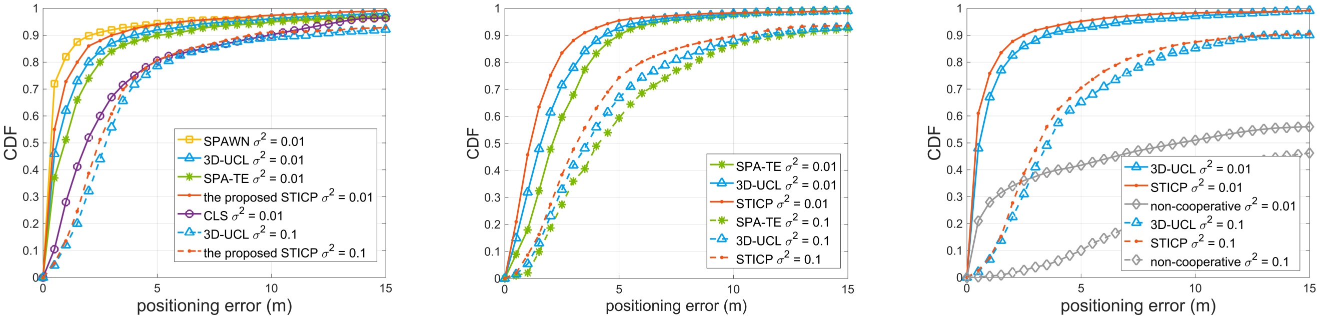
We evaluate the performance of our STICP algorithm against several representative CP algorithms by numerical simulations. The positioning performance is quantified by the cumulative distribution function (CDF) $P(e \leq \epsilon[\text{m}])$, where e denotes the positioning error and it is characterized by the average mean squared error between the estimated positions and the true position, while ϵ represents the tolerable positioning errors. Specifically, we consider a wireless network of 100 agents and 20 anchors, which are randomly scattered across a $(200 \times 200 \times 20)\text{m}^3$ space. The distance within which to perform ranging and communication is set to 50m. The default number of iterations l_{max} is set to 20. Each agent moves a distance d_i^t in a random direction, with $d_i^t \sim \mathcal{N}(0,1)$. η_1 and η_2 are set to 0.18 and 0.6, respectively. The parameter α is set to 3×10^{-2}.

We first compare the performance of our STICP against the SPA-TE [7], SPAWN [6], 3D-UCL [8], and CLS [6] schemes under TOA measurements in Fig. 3 (a). We observe that STICP outperforms the 3D-UCL, SPA-TE, and CLS schemes when the noise variance is set to $\sigma^2 = 0.01$. For example, $P\left(e \leq 4[\text{m}]\right)$ is 0.93 for STICP, while 0.90, 0.87 and 0.75 for the 3D-UCL, SPA-TE, and CLS schemes, respectively. This is attributed to the gains from internal temporal information. However, STICP performs worse than SPAWN when $P\left(e \leq 2.5[\text{m}]\right)$, due to the use of approximated messages generated by the SUT. But the performance loss is tolerable considering that the computational complexity of STICP is order-of-magnitudes lower than that of SPAWN. Our STICP also outperforms the 3D-UCL under $\sigma^2 = 0.1$, especially when $P\left(e \leq 4[\text{m}]\right)$.

We then compare the performance of our STICP against the SPA-TE and 3D-UCL schemes in Fig. 3 (b) under different noise levels, while relying on AOA measurements. We see that STICP is significantly better than 3D-UCL for $P\left(e \leq 4[\text{m}]\right)$, and the gain is mainly contributed by the temporal information within agents. Moreover, when σ^2 is increased from 0.01 to 0.1, the performance degradation of STICP is similar to that of the other benchmarking schemes and the STICP still outperforms the others.

Fig. 3 (c) shows the comparison results of STICP, 3D-UCL and the traditional non-cooperative schemes under different noise levels with RSS measurements. Again, we observe that STICP has the best performance. For instance, when $\sigma^2 = 0.01$, $P\left(e \leq 5[\text{m}]\right)$ is 0.94 for STICP, while 0.93, 0.81, 0.22 and 0.45 for 3D-UCL, CLS and non-cooperative schemes, respectively. When σ^2 is increased from 0.01 to 0.1, STICP still outperforms the others, which validates that STICP is more robust in noisy environments.

The root mean square error (RMSE) performance of the STICP with and without EAU, the SPA-TE, the SPAWN and the 3D-UCL estimators versus the number of iterations under TOA measurements are illustrated in Fig. 4. Firstly, we can see that the accuracy of all estimators improves upon increasing the number of iterations. Secondly, the RMSE curve of SPA-TE converges quickly around the 5th iteration and SPAWN converges in the 6th iteration. Thirdly, both



(a) TOA measurement (b) AOA measurement (c) RSS measurement

Fig. 3. CDFs of the positioning error of STICP against representative benchmarking schemes under TOA, AOA and RSS measurements ($l_{\max} = 20$).

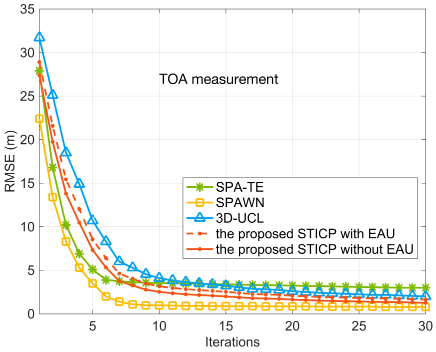


Fig. 4. RMSE of the STICP with and without EAU, the SPA-TE, the SPAWN and the 3D-UCL under TOA measurements (noise variance $\sigma^2 = 0.01$).

types of STICP converge faster than 3D-UCL and reach 1.8m and 2.1m positioning accuracy in terms of RMSE after 30 iterations, respectively. The performance degradation of STICP with acceleration may be deemed acceptable in the light of the computational speedup. Last but not least, after 20 iterations, the gains of increasing the number of iterations become insignificant, which allows us to compromise between positioning accuracy and the computational complexity.

V. Conclusion

We have proposed a low-complexity high-performance STICP algorithm for wireless cooperative localization that supports various types of ranging measurements in the GNSS-denied environment. We first create an FG network by factorizing the *a posteriori* distribution of the position-vector estimates and mapping the spatial-domain and temporal-domain operations of nodes onto the FG, which facilitates distributed implementations of wireless cooperative localization. To approximate the nonlinear terms of the messages passing across the FG, we exploited the symmetric sampling based SUT technique, which achieves high approximation accuracy with a dramatically reduced number of sample points and is able to obtain the closed-form expression of the belief. To further reduce the computational complexity and avoid any redundant iterations, we propose the EAU mechanism to filter out the agents whose position estimates have already converged so that we can terminate the rest of the iterations with minor performance loss. Our analysis and simulation results validate that the proposed STICP is capable of achieving competitive positioning performances at a low computational complexity.

References

[1] T. Lv, F. Tan, H. Gao, and S. Yang, "A beamspace approach for 2-D localization of incoherently distributed sources in massive MIMO systems," *Signal Processing*, vol. 121, pp. 30–45, Apr. 2016.

[2] T. Lv, H. Gao, X. Li, S. Yang, and L. Hanzo, "Space-time hierarchical-graph based cooperative localization in wireless sensor networks," *IEEE Transactions on Signal Processing*, vol. 64, no. 2, pp. 322–334, Jan. 2016.

[3] Y. Xiong, N. Wu, Y. Shen, and M. Z. Win, "Cooperative localization in massive networks," *IEEE Transactions on Information Theory*, vol. 68, no. 2, pp. 1237–1258, Feb. 2021.

[4] A. T. Ihler, J. W. Fisher, R. L. Moses, and A. S. Willsky, "Nonparametric belief propagation for self-localization of sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 4, pp. 809–819, Apr. 2005.

[5] A. F. Garcia-Fernandez, L. Svensson, and S. Särkkä, "Cooperative localization using posterior linearization belief propagation," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 1, pp. 832–836, Jan. 2018.

[6] H. Wymeersch, J. Lien, and M. Z. Win, "Cooperative localization in wireless networks," *Proceedings of the IEEE*, vol. 97, no. 2, pp. 427–450, Feb. 2009.

[7] N. Wu, B. Li, H. Wang, C. Xing, and J. Kuang, "Distributed cooperative localization based on Gaussian message passing on factor graph in wireless networks," *Science China Information Sciences*, vol. 58, no. 4, pp. 1–15, Sept. 2014.

[8] S. Wang and X. Jiang, "Three-dimensional cooperative positioning in vehicular ad-hoc networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 2, pp. 937–950, Feb. 2021.

[9] S. J. Julier, "The scaled unscented transformation," in *Proceedings of the 2002 American Control Conference*, vol. 6, May. 2002, pp. 4555–4559. Anchorage, AK, USA.

[10] G. D. Forney, "Codes on graphs: Normal realizations," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 520–548, Feb. 2001.

[11] J. Lien, U. J. Ferner, W. Srichavengsup, H. Wymeersch, and M. Z. Win, "A comparison of parametric and sample-based message representation in cooperative localization," *International Journal of Navigation and Observation*, vol. 2012, Jul. 2012.

[12] Y. Zhong, T. Q. Quek, and X. Ge, "Heterogeneous cellular networks with spatio-temporal traffic: Delay analysis and scheduling," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 6, pp. 1373–1386, June. 2017.