

Article

Simplify Belief Propagation and Variation Expectation Maximization for Distributed Cooperative Localization

Xueying Wang , Yan Guo ^{*}, Juliang Cao, Meiping Wu, Zhenqian Sun  and Chubing Lv

College of Intelligence Science and Technology, National University of Defense Technology, Changsha 410073, China; w_xy93@163.com (X.W.); cjvl@163.com (J.C.); meipingwu@263.net (M.W.); sunzhenqian_nudter@163.com (Z.S.); rimoben@163.com (C.L.)

* Correspondence: guoyan010417@126.com; Tel.: +86-189-7586-6763

Abstract: Only a specific location can make sensor data useful. The paper presents an simplify belief propagation and variation expectation maximization (SBPVEM) algorithm to achieve node localization by cooperating with another target node while lowering communication costs in a challenging environment where the anchor is sparse. A simplified belief propagation algorithm is proposed as the overall reasoning framework by modeling the cooperative localization problem as a graph model. The high-aggregation sampling and variation expectation–maximization algorithm is applied to sample and fit the complicated distribution. Experiments show that SBPVEM can obtain accurate node localization equal to NBP and SPAWN in a challenging environment while reducing bandwidth requirements. In addition, the SBPVEM has a better expressive ability than PVSPA, for SBPVEM is efficient in challenging environments.

Keywords: cooperative localization; belief propagation; expectation maximization; variation inference



Citation: Wang, X.; Guo, Y.; Cao, J.; Wu, M.; Sun, Z.; Lv, C. Simplify Belief Propagation and Variation Expectation Maximization for Distributed Cooperative Localization. *Appl. Sci.* **2022**, *12*, 3851. <https://doi.org/10.3390/app12083851>

Academic Editor: Stefan Fischer

Received: 21 March 2022

Accepted: 8 April 2022

Published: 11 April 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Positioning is essential in multi-sensor networks, vehicle ad hoc networks [1], underwater unmanned clusters [2], military environment research [3], forest emergency rescue [4], and other tasks that need environmental awareness. Cooperative localization uses measurements between unlocalized target nodes to localize all network nodes when anchor nodes are not sufficient or accessible.

Collaborative localization based on filtering [5] and optimization [6] requires that the nodes have a good initial state. The optimization result is unreliable when the target node does not have a good prior. However, nodes cannot provide good and reliable prior under divergent conditions and some extreme conditions. Refs. [7,8] use belief propagation (BP) methods to deal with the problems of no initial position of target nodes, few anchor nodes, and sparse links between nodes. The BP method is an accurate reasoning method applied to an acyclic probability graph, and the loopy belief propagation (LBP) method approximates the BP algorithm on an acyclic graph. Ref. [9] proposed a SPAWN method for location on the basis of LBP. In order to solve the positioning probability in continuous space, the BP algorithm needs to discretize the value space by the grid. The high probability interval may be smaller than the feasible region, which wastes many computational resources of the algorithm. Ref. [10] proposed the Nonparametric Belief Propagation (NBP) method, which uses weighted particles to represent the irregular distribution of node positions. However, communication overhead is the critical problem in the implementation, which mainly comes from the representation of information exchanged between neighboring nodes. The particle-based method requires a higher communication overhead between nodes.

In order to reduce the communication overhead, Ref. [11] combines the idea of a sigma point to fit the fitted parameters, which uses a Gaussian to represent the information between nodes. Refs. [12,13] show that when BP based on parameter fitting is applied to the positioning problem, it can only deal with the situation that the positioning center of nodes is Gaussian distribution. Ref. [14] considers the multimodal situation of node

position but confuses Gaussian distribution and mixed Gaussian distribution. Select a Gaussian to fit the mixed Gaussian. Ref. [15] proposed for the first time to use the VMP algorithm as a generalization of BP on a factor graph and to use a variational method to fit the probability of nodes, which can fit the non-Gaussian situation. However, the mean-field hypothesis destroys the association between multi-dimensional states. Ref. [16] applies the VMP method to the positioning problem. However, only single-mode and dual-mode positioning cases are considered. Ignore the constraint range of other nodes when the node can only communicate with a few anchor nodes.

This paper proposes the SBPVEM method for the distributed cooperative localization of sparse anchor targets in the algorithm. A simplified belief propagation algorithm is proposed as the overall reasoning framework by modeling the cooperative localization problem as a graph model. The high-aggregation sampling and variation expectation-maximization algorithm is applied to sample and fit the complicated distribution. In practice, our work solves the self-localization problem of distributed nodes under the condition of a minimal number of anchor nodes, makes cooperative localization with sparse anchor nodes in wireless sensor networks a reality, and lowers the cost in large-scale applications.

The Section 1 of this paper details the subject’s background and associated studies. Section 2 models cooperative localization problems as a graph model and describes belief propagation as necessary background knowledge. The proposed SBPVEM approach is detailed in depth in the Section 3. The Section 4 includes some experiments to show our algorithm’s effectiveness. Discussion is presented in the Section 5.

2. Model

2.1. System Model

The undirected graph defined by the node $v \in \mathcal{V}$ and a set of edges $(r, t) \in \mathcal{E}$ is used to represent the node location problem. Each node represents an intelligent unmanned platform. For the following discussion, we divide the nodes into anchor nodes \mathcal{V}_a that know their positions and nodes that need to be located \mathcal{V}_i . Each edge represents the communication link between sensors. If there is $(r, t) \in \mathcal{E}$, it means the communication relationship established between node T and node R , which can transfer the information of nodes and their mutual constraints Table 1.

Table 1. Basic Symbols Description.

Symbol	Description
x_t	State of node T
$\psi_t(x_t)$	Potential function of node T
$\psi_{tu}(x_t, x_u)$	Paired potential function between node T and node U
$m_{ut}(x_t)$	Message from node U to node T
$m_{ut}^i(x_t)$	The message passed by node U to node T in the i -th iteration
$B_t(x_t)$	Belief function of node T
$B_t^i(x_t)$	Belief function of node T in the i th iteration
\mathcal{G}_t	Neighbor node of node T
\mathcal{G}_t^0	The neighbor node of node t that is not the anchor node
$\mathcal{N}(\mu, \Sigma)$	Multi-dimensional gaussian distribution
$U(a, b)$	Uniform distribution
z	Normalization factor
α_k	The weight of the K -th Gaussian distribution in the Gaussian mixture model
θ_k	Parameters of the k th Gaussian distribution in the mixed Gaussian model, including μ_k and Σ_k
γ	Characteristic function, which indicates the source of the sample. The dimension is K -dimension, when sample J comes from the K -th Gaussian distribution, $\gamma_{jk} = 1$, and the rest $\gamma_{jn n \neq k} = 0$
$G(\alpha, \mu, \Sigma)$	Gaussian mixture model

The positioning problem is modeled as a probability graph model problem, and the posterior distribution of node positions is the positioning result we want.

$$\begin{aligned}
 P(\mathcal{X}|\mathcal{D}) &\propto P(\mathcal{D}|\mathcal{X})P(\mathcal{X}) \\
 &= \left(\prod_{(r,t) \in \mathcal{E}} p(d_{rt}|x_r, x_t) \right) \left(\prod_{v \in \mathcal{V}_i} p_v(x_v) \right) \tag{1}
 \end{aligned}$$

Among them, $p(d_{rt}|x_r, x_t)$: It means the probability density of distance measurement of d_{rt} based on the node position estimation of x_r, x_t . Let us assume that $d_{rt} = \|x_r - x_t\| + \omega_{rt}$, where $\omega_{rt} \sim \mathcal{N}(0, \sigma_{ij}^2)$ represents the measured noise, we can obtain:

$$p(d_{rt}|x_r, x_t) = \frac{1}{\sqrt{2\pi\sigma_{rt}^2}} \exp\left\{-\frac{(d_{rt} - \|x_r - x_t\|)^2}{2\sigma_{rt}^2}\right\} \tag{2}$$

$p_v(x_v)$ indicates the probability density of the location of the node. In the cooperative localization problem, the marginal probability of posterior distribution is the position estimation of nodes.

$$p(x_v|D) \propto \int p(X|D)dX \setminus x_v \tag{3}$$

where $X \setminus x_v$ represents the set of all variables in x except x_v .

2.2. Belief Propagation

The BP method is an accurate reasoning method, and the core process is belief calculation and message calculation. The belief of node A is the probability distribution density of variable A value.

The message from node A to node B is the probability distribution density of variable B 's possible values from the perspective of node A . The renewal of the belief of node B is the aggregation of information transmitted by node B to all neighboring nodes.

U sends a message to node t :

$$m_{ut}(x_t) = \int_{x_u} \psi_{tu}(x_t, x_u) \frac{B_u(x_u)}{m_{tu}(x_u)} dx_u \tag{4}$$

T receives the message from the surrounding nodes and starts belief calculation:

$$B_t(x_t) = z\psi_t(x_t) \prod_{u \in \mathcal{G}_t} m_{ut}(x_t) \tag{5}$$

where $\psi_u(x_u)$ is the node potential function, $\psi_{tu}(x_t, x_u)$ is the paired potential function of two nodes, and z is the normalization factor.

When the BP method is applied to the location problem, we consider the location problem based on distance. In this problem, the potential function of nodes $\psi_t(x_t)$ is the probability density function of nodes in different positions.

$$\psi_t(x_t) = p_t(x_t) \tag{6}$$

The paired potential function $\psi_{tu}(x_t, x_u)$ between two nodes is the probability density function of two nodes based on the distance. The combination Formula (2) is:

$$\begin{aligned}
 \psi_{tu}(x_t, x_u) &= p(d_{tu}|x_u, x_t) \\
 &= z \exp\left\{-\frac{(d_{tu} - \|x_u - x_t\|)^2}{2\sigma_{tu}^2}\right\} \tag{7}
 \end{aligned}$$

LBP, which can be applied to a graph with loops, adds iteration based on BP. The message from node u to node t in the i -th iteration is:

$$m_{ut}^i(x_t) = \int_{x_u} \psi_{tu}(x_t, x_u) \frac{B_u^{i-1}(x_u)}{m_{tu}^{i-1}(x_u)} dx_u \tag{8}$$

The belief update of node t in the i -th iteration is:

$$B_t^i(x_t) = z\psi_t(x_t) \prod_{u \in \mathcal{G}_t} m_{ut}^i(x_t) \tag{9}$$

3. Methodology

Since the viable solution to the positioning problem is in continuous space, directly dividing the feasible space and calculating the probability wastes much computational power. To lower the computation complexity and communication requirements, NBP uses particles to represent potential functions. Nevertheless, the amount of data to transmit is still huge Figure 1.

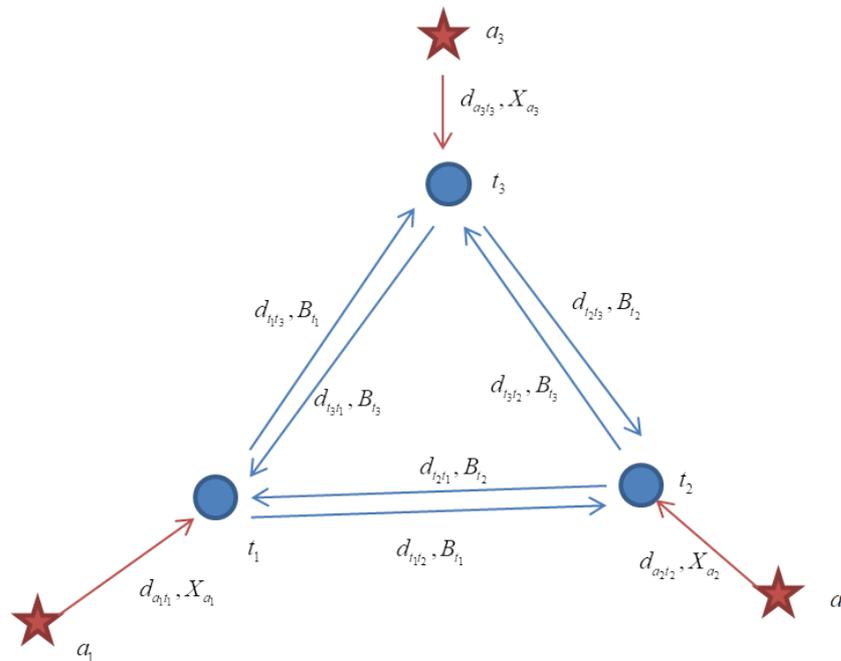


Figure 1. Localized anchor nodes are represented by the red five-pointed star, while unlocalized target nodes are represented by the blue circle. Each node sends its opinion of the distribution of the neighborhood node on the basis of its belief and measured distance.

This paper proposes a simple belief propagation and variation expectation–maximization (SBPVEM) method to realize the cooperative localization of nodes. The algorithm consists mainly of Simplify Belief Propagation (SBP), high-aggregation sampling, and variation expectation–maximization (VEM). The concrete operation of the algorithm on the node is shown in Figure 2.

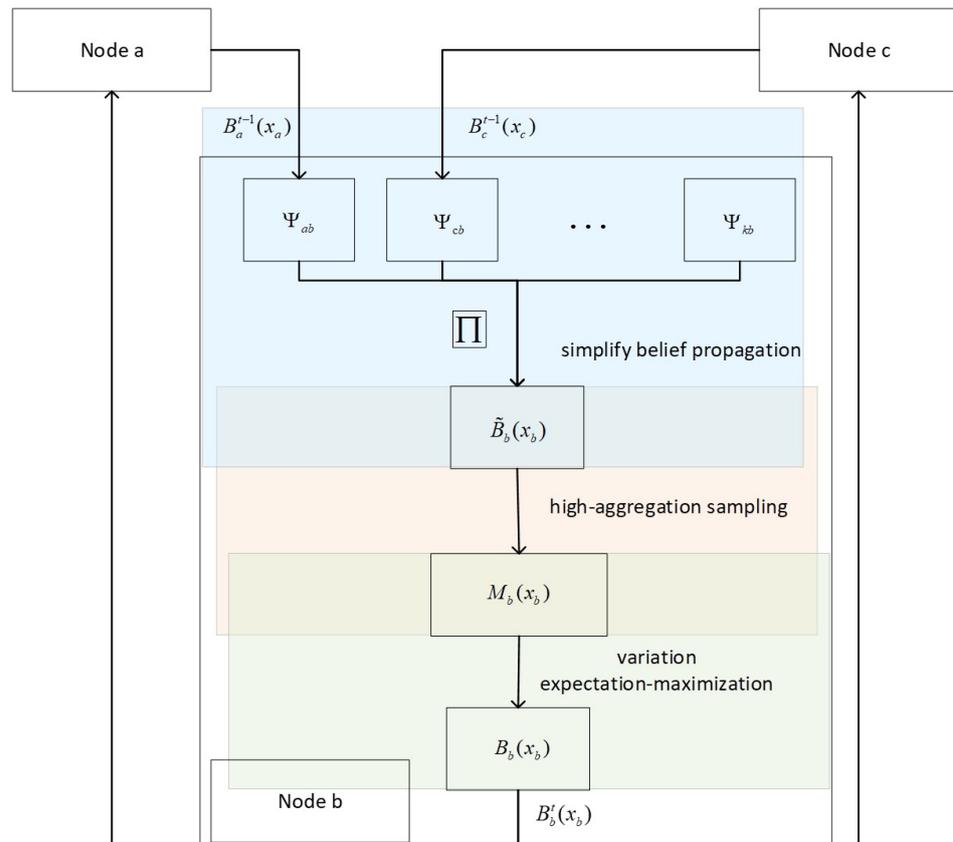


Figure 2. SBPVEM intra-node data flow. The node receives neighbors belief $B_{i \in \mathcal{G}}$ and computes potential function Ψ between nodes under the constraints of distance measurement. The belief \tilde{B} can be obtained by multiplying all the neighboring Ψ . \tilde{B} is complicated, by high-aggregation sampling and variation expectation-maximization, a GMM B is obtained to represent \tilde{B} .

3.1. Simplify Belief Propagation

According to the message update formula, the message m_{ut} transmitted by node U to node T is the accumulation of the product of the potential function between two nodes. There is an aggregation of messages transmitted by node U to neighboring nodes except T on the possible values of each node U . Here, the information transmitted by the T node to the U node is removed to prevent the confidence of the T node from being used many times, resulting in overconfidence of the node. However, the approximation of LBP to BP shows that the repeated use of node confidence will not affect the expectation of node positioning, so here, we simplify the message from node U to node T as follows:

$$m_{ut}^i(x_t) = \int_{x_u} \psi_{tu}(x_t, x_u) B_u^{i-1}(x_u) dx_u \tag{10}$$

Because of the equivalence of the solution, we can make:

$$m_{ut}^i(x_t) = B_u^{i-1}(x_u) \tag{11}$$

$$\Psi_{ut}(x_t, x_u) = \int_{x_u} \psi_{tu}(x_t, x_u) m_{ut}^i(x_t) dx_u \tag{12}$$

$$\tilde{B}_t^i(x_t) = k \psi_t(x_t) \prod_{u \in \mathcal{G}_t} \Psi_{ut} \tag{13}$$

In this way, we disassemble the SBP problem into three steps: The first step, the same as that shown in Formula (11), is to broadcast the belief of the node as the message of the node to all neighboring nodes. Secondly, according to the Formula (12), the potential function Ψ between the target node and the neighbor node is calculated based on the

belief of the source node and the constraint between the two nodes. In Step 3, based on Formula (13), the target node calculates the belief estimate of its own positioning \hat{B} based on the potential function.

Assuming that the Gaussian mixture model can express the potential function of nodes, the Gaussian mixture model results from the superposition of multiple Gaussian distributions.

$$G(\alpha, \mu, \Sigma) = \sum_{k=1}^n \alpha_k \mathcal{N}(\mu_k, \Sigma_k) \tag{14}$$

Among them, \mathcal{N} represents Gaussian distribution, and μ_k and Σ_k are the parameters of the k -th Gaussian distribution. The formula of multi-dimensional Gaussian distribution is:

$$\mathcal{N}(\mu_k, \Sigma_k) = \frac{1}{2\pi \sqrt{\det(\Sigma_k)}} \exp^{-\frac{1}{2}(x-\mu_k)^T \Sigma_k^{-1} (x-\mu_k)} \tag{15}$$

The potential function of nodes is:

$$B_u^i(x_u) = G_u(\alpha, \mu, \Sigma) \tag{16}$$

Formulas (14) and (16) are brought into (12), and there are:

$$\begin{aligned} & \Psi_{ut}(x_t, x_u) \\ &= \sum_{k=1}^K \alpha_k \int_{x_u} \psi_{tu}(x_t, x_u) \mathcal{N}_u(\mu_k, \Sigma_k) dx_u \end{aligned} \tag{17}$$

According to Formulas (7) and (15), $\psi_{tu}(x_t, x_u)$ can be regarded as one-dimensional Gaussian, and $\mathcal{N}_u(\mu_k, \Sigma_k)$ can be regarded as part of the integral that can be understood as the paired potential function ψ_{tu} , the expectation under the probability $\mathcal{N}_u(\mu_k, \Sigma_k)$. For the convenience of calculation, we can directly bring the paired potential function between nodes into the belief distribution of nodes. Let x_s be the equivalent variable:

$$x_s = \frac{x_u - \mu}{|x_u - \mu|} (|x_u - \mu| - d) \tag{18}$$

in which $|x_u - \mu| - d$ considers the distance constraint between nodes and transforms the two-dimensional problem into one-dimensional. $\frac{x_u - \mu}{|x_u - \mu|}$ keeps the directionality of the node position and restores the one-dimensional distance to a two-dimensional direction. Bring the equivalent variable x_s into the two-dimensional belief distribution of nodes and obtain:

$$\begin{aligned} & \int_{x_u} \psi_{tu}(x_t, x_u) \mathcal{N}_u(\mu_k, \Sigma_k) dx_u \\ &= z \exp\left(-\frac{x_s \Sigma_k^{-1} x_s}{2}\right) \\ &= z \exp\left(-\frac{\left[\frac{x_u - \mu}{|x_u - \mu|} (|x_u - \mu| - d)\right] \Sigma_k^{-1} \left[\frac{x_u - \mu}{|x_u - \mu|} (|x_u - \mu| - d)\right]}{2}\right) \end{aligned} \tag{19}$$

The results are shown schematically in Figure 3. Bring Formula (19) into (17) and obtain the potential function obtained by fusion node potential function and paired potential function.

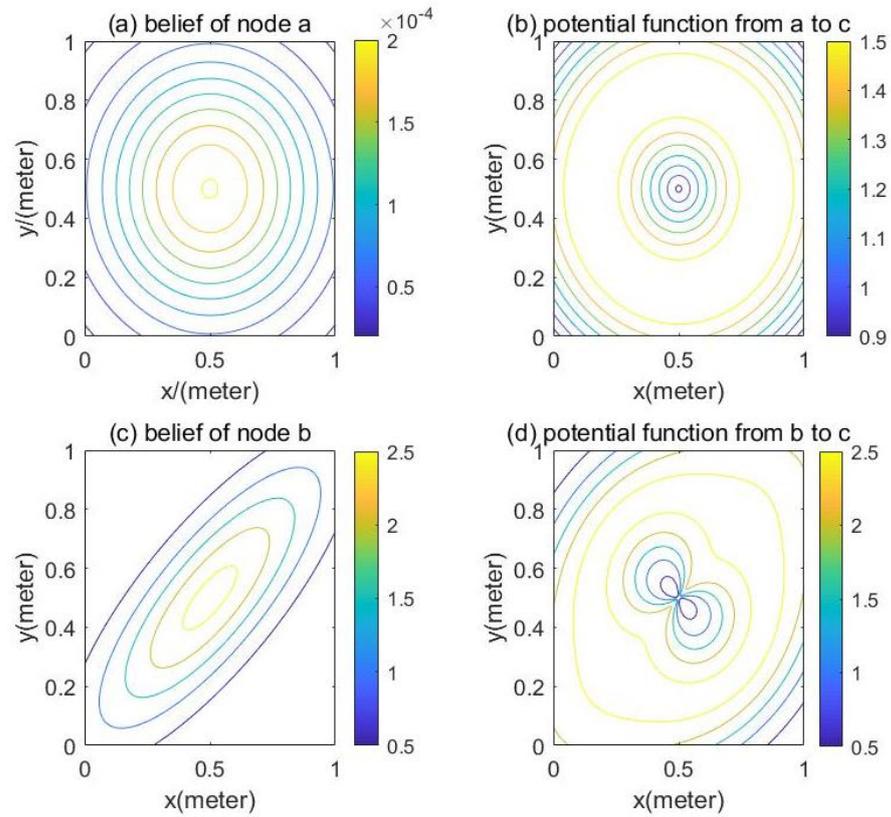


Figure 3. Node potential function under different distributions. (a) indicates belief of node a (\mathcal{N}_a), where $\mu = [0.5, 0.5]$, $\sigma = [0.1, 0.1]$, $\rho = 0$; (b) indicates the potential function Ψ_{ac} with $dis_{ac} = 0.35$; (c) indicates belief of node b (\mathcal{N}_b), where $\mu = [0.5, 0.5]$, $\sigma = [0.1, 0.1]$, $\rho = 0.8$; (d) indicates the potential function Ψ_{bc} with $dis_{bc} = 0.35$.

3.2. High-Aggregation Sampling

In SBP, all nodes transmit the position belief of the node itself. However, the belief of nodes is the aggregation of paired potential functions based on mixed Gaussian, and its form is very complicated. For the convenience of expression, we can consider approximating the accurate distribution of nodes by sampling and re-fitting. We propose a highly condensed sampling method, which can simultaneously reduce the amount of calculation, improve the sampling efficiency, and improve the effectiveness of samples.

Sampling is divided into five steps: uniform sampling in sampling space, probability calculation, node filtering, resampling, and importance sampling.

(1) Determine the sampling space and obtain uniform sampling $\tilde{x} \sim U(x_{min}, x_{max})$ in the space.

$$x_{min} = \max\{x_{min}; \max(x_g - d_g)\} \tag{20}$$

$$x_{max} = \min\{x_{max}; \min(x_g + d_g)\} \tag{21}$$

(2) Apply the belief of the node \tilde{B} to the sampled sample particles, and obtain the particle probability $P(\tilde{x})$

(3) Filter P_{gate} for nodes with particularly low probability to obtain effective nodes x_{gate} .

$$x_{gate} = \tilde{x} \quad \text{if} \quad P(\tilde{x}) > P_{gate} \tag{22}$$

(4) When the number of valid samples is insufficient, resample by sample expansion and random walk.

Expand the sample by copying the sample:

$$\tilde{x}_{gate} = \left[\underbrace{x_{gate}; \dots; x_{gate}}_{\text{floor}(\eta)} \right]; \tag{23}$$

where $\eta = n / \text{size}(x_{gate})$, n is the total number of samples of the node.

Through random walk of the expanded samples, resampling of the samples is realized.

$$x_{tem} = \tilde{x}_{gate} + \beta \tilde{\eta} \sigma v \tag{24}$$

among $\tilde{\eta} = \max\{3, \eta\}$, the combination of $\tilde{\eta}$ and σ is used to limit the distance traveled. v is used to indicate the direction of random walk. In a two-dimensional environment, $v = [\cos(\theta), \sin(\theta)]\beta \sim U[0, 1]$ means uniform sampling in distance. $\theta \sim U[0, 2\pi)$ means uniform sampling in angle.

(5) Importance sampling of the obtained new particles.

$$x_{sample} \sim P(x_{tem}) \tag{25}$$

In this algorithm, uniform sampling is used to obtain sufficient effective particles, which ensures the expressiveness of the samples as a whole. Stepping uniform sampling by importance sampling can realize the approximation of distribution by particles.

3.3. Variation Expectation Maximization

The EM method is a parameter solution method of the Gaussian mixture model, which is usually used to solve a low-dimensional model with a known number of components. However, it is impossible to determine in advance how many Gaussian models are needed to achieve proper fitting for strange shapes. In order to better fit the possible positions of nodes, this paper proposes a variable expectation–maximization method, which can adapt to the fitting situation with an uncertain number of Gaussian components. In this paper, we consider initializing the results using a Gaussian mixture model with enough components and then filter the Gaussian distribution used for fitting by component filtering, leaving the part with higher responsiveness as the selected model Algorithm 1.

In the Gaussian mixture model shown as Formula (14), α_k is the weight of the k -th Gaussian distribution, $\alpha_k \in [0, 1]$, and $\sum_{k=1}^K \alpha_k = 1$. We introduce the hidden variable γ_{jk} to indicate whether the j -th sample comes from the K -th Gaussian distribution. When the j -th sample comes from the k -th Gaussian distribution, $\gamma_{jk} = 1$; otherwise, $\gamma_{jk} = 0$. When the weight of the k -th Gaussian distribution is more significant, the more samples from the k -th Gaussian distribution there will be. On the other hand, the more samples from the k -th class, the heavier the weight of the k -th class. The relationship between the hidden variable γ_{jk} and the weight α_k is:

$$p(\gamma_{jk} = 1) = \alpha_k \tag{26}$$

Because $\gamma_{j \times} = [0, \dots, 0, 1, 0, \dots, 0]$ for every sample j is a 1-of- K hot unique code, it can be written as follows:

$$p(\gamma_{j \times} | \alpha) = \prod_{k=1}^K \alpha_k^{\gamma_{jk}} \tag{27}$$

After specifying the classification, there are:

$$p(y_j | \gamma_{jk} = 1, \mu, \Sigma) = \mathcal{N}(y_j | \mu_k, \Sigma_k) \tag{28}$$

Algorithm 1: Parameter fitting based on VEM

Input: Sampling sample x_{sample} , initial category number K , initial Gaussian mixture model parameters $\alpha_0, \mu_0, \Sigma_0$, threshold divergence KL_{gate} , iteration number n_{iter}

Output: α, μ, Σ

initialization;

while $n < n_{iter}$ **do**

$n = n + 1$

E step:

 Using the current parameters, calculate the response matrix $\hat{\gamma}_{jk}$ based on Formula (36)

M step:

 Calculate the weight of the component α_n based on Formula (37)

 Filter the components based on Formula (38) and obtain $\alpha_{pick}, \mu_{pick}$ and Σ_{pick}

$\hat{\gamma}_{jk} = \hat{\gamma}_{jk_{pick}}$

 Update the expected μ_n based on Formula (39)

 Update the covariance σ_n based on Formula (40)

Check step:

while $n = n_{iter}$ **do**

 Calculate KL divergence between two distributions: $KL(q||p)$

if $KL(q||p) > KL_{gate}$ **then**

$n = 0$

else

break

end

end

end

The joint probability of all samples considering sampling:

$$p(\gamma|\alpha) = \prod_{j=1}^N \prod_{k=1}^K \alpha_k^{\gamma_{jk}} \quad (29)$$

In case of multi-sample and multi-classification:

$$p(y|\gamma, \mu, \Sigma) = \prod_{j=1}^N \prod_{k=1}^K \mathcal{N}(y_j|\mu_k, \Sigma_k)^{\gamma_{jk}} \quad (30)$$

The likelihood probability under the given distribution parameters is:

$$P(y, \gamma|\alpha, \mu, \Sigma) = \prod_{k=1}^K \alpha_k^{n_k} \prod_{j=1}^N [\mathcal{N}(y_j|\mu_k, \Sigma_k)]^{\gamma_{jk}} \quad (31)$$

Among them, $n_k = \sum_{j=1}^N \gamma_{jk}$ is used to represent the number of nodes from the k -th Gaussian distribution and can represent the intensity weight of the k -th component in the calculation.

$$\alpha_k = \frac{n_k}{N} \quad (32)$$

For the convenience of calculation, log-likelihood can be taken:

$$\log P(y, \gamma|\theta) = \sum_{k=1}^K \left(n_k \log \alpha_k + \sum_{j=1}^N \gamma_{jk} \log [\mathcal{N}(y_j|\theta_k)] \right) \quad (33)$$

where:

$$\log[\mathcal{N}(y_j|\theta_k)] = -\log(2\pi) - \frac{1}{2} \log(|\Sigma_k|) - \frac{1}{2} (y - \mu_k)^T \Sigma_k^{-1} (y - \mu_k) \tag{34}$$

The goal of the EM algorithm is to find the distribution parameters that can make the sample appear the most expected. Through two steps of expectation and maximization, it is realized iteratively. The expectation is to calculate the expectation of hidden variables under the current observation data and parameters. Maximizing is to find the parameters that can maximize the expectation of the currently hidden variable.

Given the observation y and the current parameter θ^i , the expectation of conditional distribution of hidden variable γ is:

$$Q(\theta, \theta^{(i)}) = \mathbb{E}_\gamma [\log P(y, \gamma|\theta) | Y, \theta^i] = \sum_{k=1}^K \left(n_k \log \alpha_k + \sum_{j=1}^N \hat{\gamma}_{jk} \log [\mathcal{N}(y_j|\theta_k)] \right) \tag{35}$$

in which θ is the real parameter of distribution, $\theta^{(i)}$ is the distribution parameter calculated in the i -th iteration, and $\hat{\gamma}_{jk} = \mathbb{E}_\gamma [\gamma_{jk}]$

$$\hat{\gamma}_{jk} = \frac{\alpha_k \mathcal{N}(y_j|\theta_k)}{\sum_{k=1}^K \alpha_k \mathcal{N}(y_j|\theta_k)} \tag{36}$$

$\hat{\gamma}_{jk}$ indicates the possibility that sample J is in the k -th component of the Gaussian mixture model. $[\hat{\gamma}_{jk}]$ is called the responsivity matrix. When there are more samples from the k -th possibility, the weight of the k -th component will be more significant.

$$\alpha_k = \frac{n_k}{N} \approx \sum_j \hat{\gamma}_{jk} \tag{37}$$

When K is extensive, many components make up the Gaussian mixture model. However, there will be some components that do not play a vital role. In order to reduce the amount of further calculation, it is necessary to screen the components of the Gaussian model.

$$\hat{k}_{pick} \leftarrow \alpha_k > \tau \frac{1}{|\alpha|} \tag{38}$$

where τ is the scaling factor. When the weight α_k of the k -th component in the mixed Gaussian model is greater than $\tau \frac{1}{|\alpha|}$, the k th component will be selected to enter the next iteration; otherwise, it will be deleted. In the iterative process, the number of components can be controlled through continuous selection.

The parameters that can maximize the current expectation can be determined by derivation. Extreme values can be solved by making the partial derivative of the mean value zero:

$$\mu_k = \frac{\sum_{i=1}^N \hat{\gamma}_{ik} y_i}{\sum_{i=1}^N \hat{\gamma}_{ik}} \tag{39}$$

You can obtain the updated Σ_k :

$$\Sigma_k = \frac{\sum_{j=1}^N \hat{\gamma}_{jk} (y_j - \mu_k)^T (y_j - \mu_k)}{\sum_{j=1}^N \hat{\gamma}_{jk}} \tag{40}$$

3.4. Summary

The proposed method contains three steps: simplified belief propagation, high agglutination sampling, and variation expectation-maximization. A simplified belief propagation algorithm is proposed as the overall reasoning framework by modeling the cooperative localization problem as a graph model. The high-aggregation sampling and variation

expectation–maximization algorithm is applied to sample and fit the complicated distribution. The overall flow is depicted in Algorithm 2.

Algorithm 2: Cooperative localization based on SBPVEM method

Input: Position of anchor point, distance estimation between nodes
Output: Location estimation of target nodes

initialization;
while $i < i_iter$ **do**
 $i = i + 1$
 for $t \in \mathcal{V}_t$ **do**
 1. Receive the belief of neighbor nodes.: $B_{u|u \in \mathcal{G}_t} = m_{u|u \in \mathcal{G}_t}^{i-1}$
 2. Based on the Formula (12), calculate the potential function $\psi_{ut}(x_u, x_t)I$
 3. Calculate the aggregate \tilde{B}_t^i of the potential functions of neighboring nodes based on Formula (13) 4. Using (20)–(25), the feasible spatial sampling of the node is x_{sample}^i
 5. Based on VEM, fit the sampling results to $B_t^i = g(\alpha, \mu, \sigma)$
 6. Broadcast your own location distribution $m_t^i = b_t^i$
 end
end

The point-to-point message transmission between nodes is turned into the node’s position estimation broadcast in simplified belief propagation. Before importance sampling, high-aggregation sampling adds a filter layer to improve efficiency by clustering particles in high-probability intervals. Finally, the particle is fitted into a Gaussian mixture model with an unknown number of components using variable expectation maximum.

4. Experiment and Results

4.1. Setting Up

To verify the algorithm, we use static numerical experiments in 2d with a range of 20 m × 20 m. Consider the following three connection scenarios to gradually increase the difficulty of locating the target node: two anchor nodes, one anchor node, and no anchor node. NBP [10], SPAWN [9], and PVSPA [16] are compared to SBPVEM.

The RMSE of the position error is calculated to evaluate the accuracy of different algorithms.

$$err = \frac{1}{N} \| \mathbf{x}_{gt} - \mathbf{x}_{pre} \| \tag{41}$$

where N indicates the number of nodes to be located. $\mathbf{x}_{gt} \in \mathbb{R}^{N \times 2}$ is a vector composed of the ground truth positions of all nodes to be located. $\mathbf{x}_{pre} \in \mathbb{R}^{N \times 2}$ is a vector composed of the predicted positions of all nodes to be located.

$$err = \frac{1}{N} \sum_{i=1}^N \| x_{gt}^i - x_{pre}^i \| \tag{42}$$

where $x_{gt}^i = [p_{gt}^{i1}, p_{gt}^{i2}]$ is the ground truth position of node i . $x_{pre}^i = [p_{pre}^{i1}, p_{pre}^{i2}]$ is the predicted position of node i . $\| x_{gt}^i - x_{pre}^i \| = \sqrt{(p_{gt}^{i1} - p_{pre}^{i1})^2 + (p_{gt}^{i2} - p_{pre}^{i2})^2}$ is the Euclidean distance between the ground truth position and the predicted position of the node i .

4.2. Connection Scenario 1: Two Anchors

In connection scenario 1, four anchor nodes and two target nodes are set up with each target node connecting to two anchor nodes, and the target nodes connecting to each other. Figure 4 depicts the location distribution (a), communication relationship between nodes (a), positioning error of various algorithms (b–e), and overall positioning error to contrast

(f). The four algorithms can achieve nice positioning and convergence after two iterations, as shown in Figure 4f.

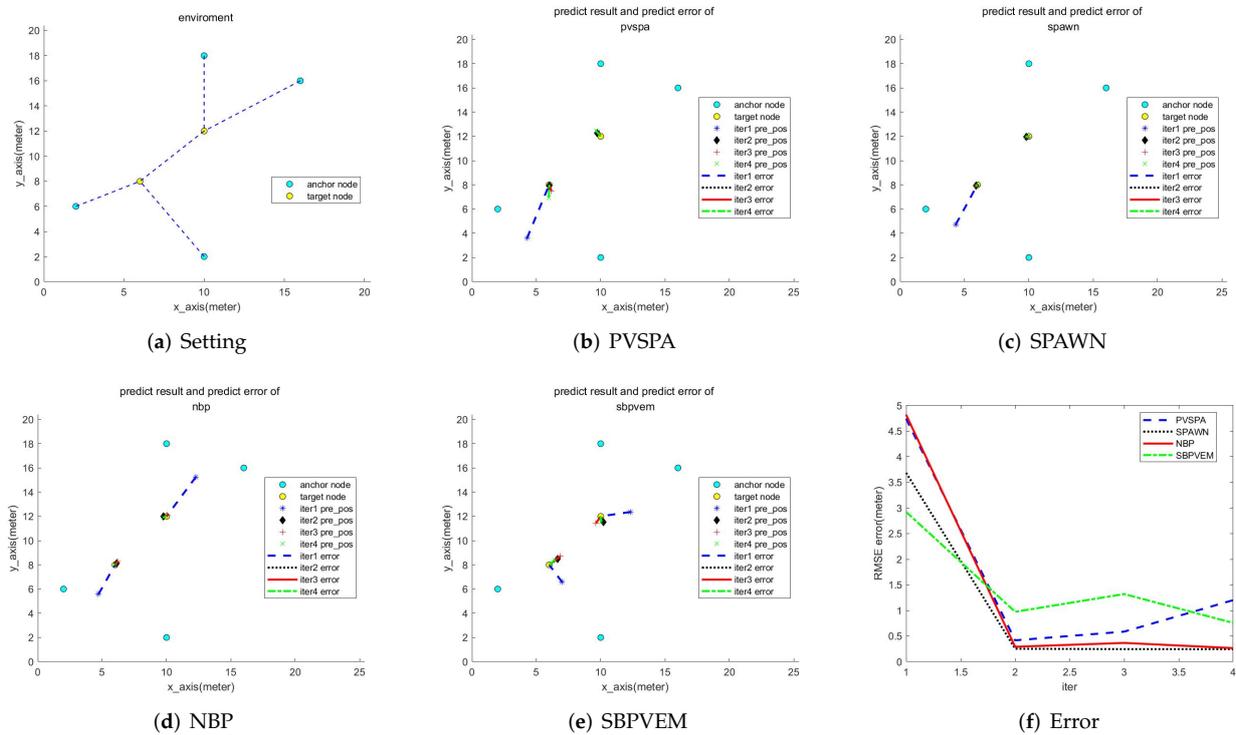


Figure 4. Location results of connection scenario 1. (a) exhibits the setting of scenario 1. Anchors are represented by cyan nodes, whereas yellow nodes represent targets. Blue dotted lines link two nodes with communication. PVSPA, SPAWN, NBP, and SBPVEM iterative positioning results and corresponding positioning errors are shown in (b–e). (f) comparing the prediction errors of various algorithms in the iterative process.

4.3. Connection Scenario 2: One Anchor

In connection scenario 2, three anchor nodes and three target nodes are set up, with each target node connecting to one anchor node, and the target nodes connecting to each other. Figure 5 depicts the location distribution (a), communication relationship between nodes (a), positioning error of various algorithms (b–e), and overall positioning error to contrast (f). As the diagram depicts, SBPVEM can obtain accurate locate results equivalent to SPAWN and NBP. Nevertheless, PVSPA can hardly reduce the position error.

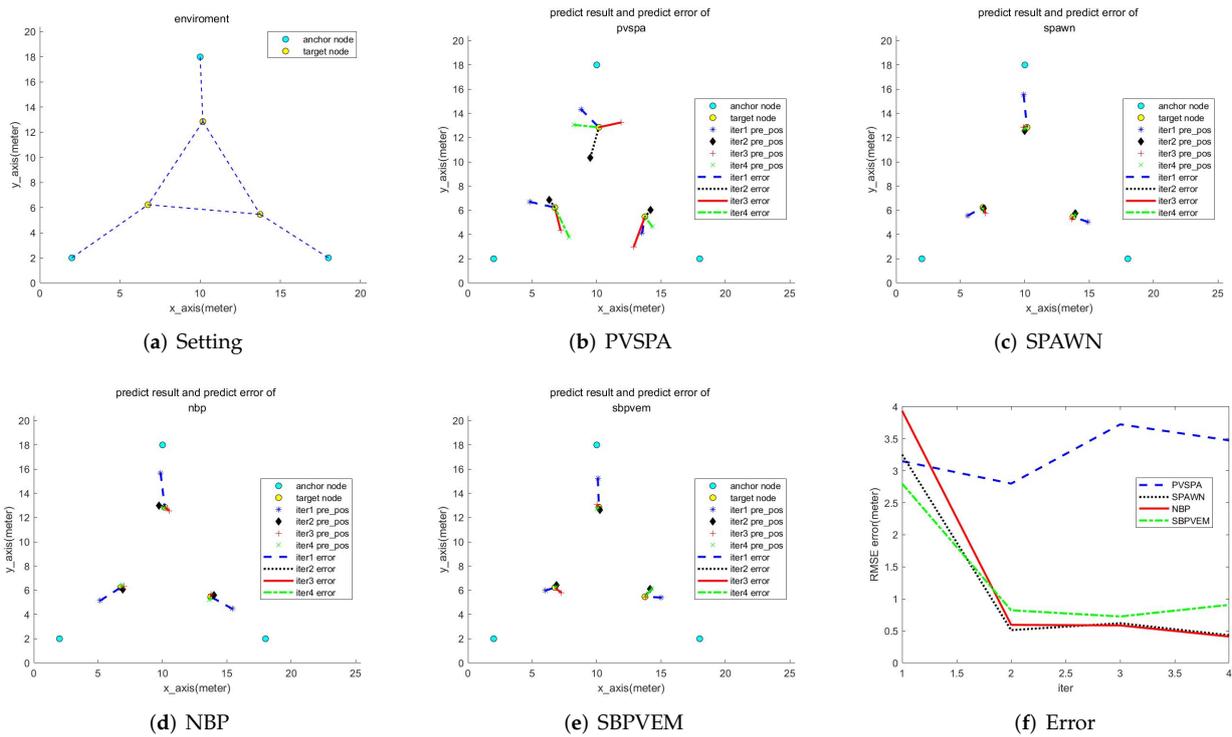


Figure 5. Location results of connection scenario 2. (a) exhibits the setting of scenario 2. Anchors are represented by cyan nodes, whereas yellow nodes represent targets. Blue dotted lines link two nodes with communication. PVSPA, SPAWN, NBP, and SBPVEM iterative positioning results and corresponding positioning errors are shown in (b–e). (f) comparing the prediction errors of various algorithms in the iterative process.

4.4. Connection Scenario 3: No Anchor

In connection scenario 3, three anchor nodes and six target nodes are set up. Three of six target nodes connect with one anchor, while others only connect with neighbor target nodes. Figure 6 depicts the location distribution (a), communication relationship between nodes (a), positioning error of various algorithms (b–e), and overall positioning error to contrast (f). There was no efficacious message in iteration 1 transmitted to target nodes with no anchor neighbor, resulting in their first prediction near the center. After two iterations, all three algorithms except PVSPA improve prediction accuracy. PVSPA makes locating even worse.

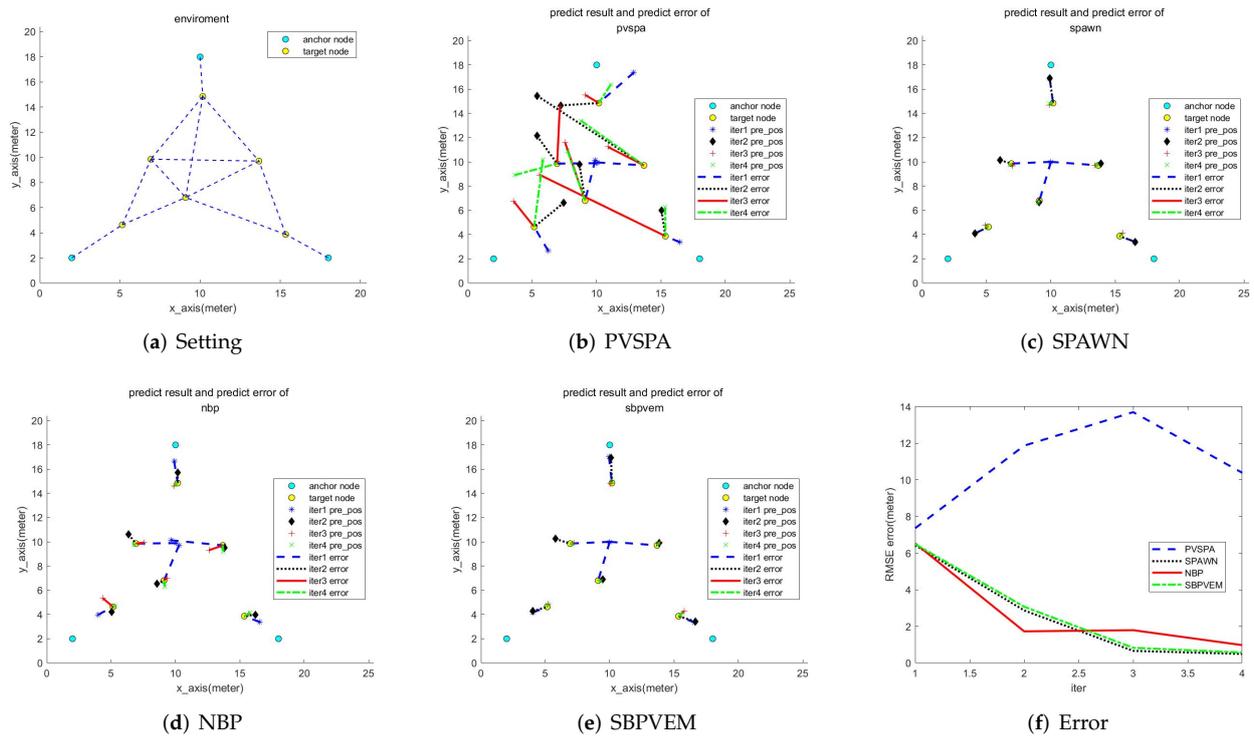


Figure 6. Location results of connection scenario 3. (a) exhibits the setting of scenario 2. Anchors are represented by cyan nodes, whereas yellow nodes represent targets. Blue dotted lines link two nodes with communication. PVSPA, SPAWN, NBP, and SBPVEM iterative positioning results and corresponding positioning errors are shown in (b–e). (f) comparing the prediction errors of various algorithms in the iterative process.

5. Discussion

Considering the bandwidth requirement, the required bandwidth of SPAWN is proportional to grid divisions, for its message is the probability of the grid. The data amount of NBP are proportional to the number of particles used to express potential function. SBPVEM and PVSPA fit the given distribution before sending parameters as messages, so their requirement in bandwidth is small. In summary, $C_{SPA} \gg C_{NBP} \gg C_{SBPVEM} \approx C_{PVSPA}$.

SBPVEM and PVSPA reduce the communication bandwidth. Experiments show that the SBPVEM algorithm can sufficiently cope with the challenging environment that the PVSPA algorithm cannot handle. SBPVEM can obtain accurate node localization while reducing bandwidth requirements and has a better expressive ability than PVSPA.

6. Conclusions

This paper proposes the SBPVEM method for the distributed cooperative localization of sparse anchor targets. A simplified belief propagation method is proposed as the overall reasoning framework by modeling the cooperative localization problem as a graph model. The high-aggregation sampling and variation expectation–maximization algorithm are applied to sample and fit the complicated distribution.

SBPVEM can obtain accurate node localization in a challenging environment while reducing bandwidth requirements. Our work solves the self-localization problem of distributed nodes under the condition of a minimal number of anchor nodes, makes cooperative localization with sparse anchor nodes in wireless sensor networks a reality, and lowers the cost in large-scale applications.

Author Contributions: Conceptualization, M.W. and J.C.; software, X.W.; validation, X.W.; formal analysis, X.W. and Y.G.; investigation, X.W.; resources, Y.G.; data curation, X.W.; writing—original draft preparation, X.W., C.L.; writing—review and editing, Y.G. and Z.S.; visualization, X.W. and Z.S.; supervision, Y.G. and M.W.; project administration, M.W. and J.C.; funding acquisition, Y.G. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by the National Natural Science Foundation of China (No. 62103424) and Junior faculty Innovation Research Project, College of Intelligence Science and Engineering, National University of Defense Technology (No. ZN2019-16).

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

BP	Belief Propagation
LBP	Loopy Belief Propagation
NBP	Nonparametric Belief Propagation
SPAWN	Sum-Product Algorithm over a Wireless Network
PVSPA	Parametric Variational Sum-Product Algorithm
SBPVEM	Simplify Belief Propagation and Variation Expectation Maximization

References

1. Elazab, M.; Noureldin, A.; Hassanein, H.S. Integrated cooperative localization for Vehicular networks with partial GPS access in Urban Canyons. *Veh. Commun.* **2017**, *9*, 242–253. [\[CrossRef\]](#)
2. Li, Y.C.; Wang, Y.Y.; Yu, W.B.; Guan, X.P. Multiple Autonomous Underwater Vehicle Cooperative Localization in Anchor-Free Environments. *IEEE J. Ocean. Eng.* **2019**, *44*, 895–911. [\[CrossRef\]](#)
3. Benini, A.; Mancini, A.; Longhi, S. An IMU/UWB/Vision-based Extended Kalman Filter for Mini-UAV Localization in Indoor Environment using 802.15.4a Wireless Sensor Network. *J. Intell. Robot. Syst.* **2013**, *70*, 461–476. [\[CrossRef\]](#)
4. Shi, Q.; Cui, X.; Li, W.; Xia, Y.; Lu, M. Visual-UWB navigation system for unknown environments. In Proceedings of the 31st International Technical Meeting of the Satellite Division of the Institute of Navigation, ION GNSS+ 2018, Miami, FL, USA, 24–28 September 2018; pp. 3111–3121. [\[CrossRef\]](#)
5. Manoharan, A.; Sharma, R.; Sujit, P.B. Nonlinear model predictive control to aid cooperative localization. In Proceedings of the 2019 International Conference on Unmanned Aircraft Systems, ICUAS 2019, Atlanta, GA, USA, 11–14 June 2019; pp. 26–32. [\[CrossRef\]](#)
6. Nguyen, T.V.; Jeong, Y.; Shin, H.; Win, M.Z. Least-Square Cooperative Localization. *IEEE Trans. Veh. Technol.* **2015**, *64*, 1318–1330. [\[CrossRef\]](#)
7. Song, Y.; Wang, C.X.; Tay, W.P.; Law, C.L. Grid-based belief propagation. In Proceedings of the 2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Sapporo, Japan, 18–21 September 2017; pp. 1–6.
8. Naseri, H.; Koivunen, V. A Bayesian algorithm for distributed network localization using distance and direction data. *IEEE Trans. Wirel. Commun.* **2017**, *5*, 290–304. [\[CrossRef\]](#)
9. Wymeersch, H.; Lien, J.; Win, M.Z. Cooperative localization in wireless networks. *Proc. IEEE* **2009**, *97*, 427–450. [\[CrossRef\]](#)
10. Ihler, A.T.; Fisher, J.W.; Moses, R.L.; Willsky, A.S. Nonparametric belief propagation for self-localization of sensor networks. *IEEE J. Sel. Areas Commun.* **2005**, *23*, 809–819. [\[CrossRef\]](#)
11. Meyer, F.; Hlinka, O.; Hlawatsch, F. Sigma point belief propagation. *IEEE Signal Process. Lett.* **2013**, *21*, 145–149. [\[CrossRef\]](#)
12. Caceres, M.A.; Penna, F.; Wymeersch, H.; Garello, R. Hybrid cooperative positioning based on distributed belief propagation. *IEEE J. Sel. Areas Commun.* **2011**, *29*, 1948–1958. [\[CrossRef\]](#)
13. Liu, Y.; Lian, B.; Zhou, T. Gaussian message passing-based cooperative localization with node selection scheme in wireless networks. *Signal Process.* **2019**, *156*, 166–176. [\[CrossRef\]](#)
14. Li, S.; Hedley, M.; Collings, I.B. New efficient indoor cooperative localization algorithm with empirical ranging error model. *IEEE J. Sel. Areas Commun.* **2015**, *33*, 1407–1417. [\[CrossRef\]](#)
15. Dauwels, J. On variational message passing on factor graphs. In Proceedings of the 2007 IEEE International Symposium on Information Theory, Nice, France, 24–29 June 2007; pp. 2546–2550.
16. Li, W.; Liu, D. Parametric Variational Sum-Product Algorithm for Cooperative Localization in Wireless Sensor Networks. *IEEE Access* **2021**, *9*, 89834–89845. [\[CrossRef\]](#)