Distributed Cubature Kalman Filter Cooperative Localization Based on Parameterized-belief Propagation

Lin Zhou, Lu Zhang*, Yong Jin, Zhentao Hu, Junwei Li

School of Artificial Intelligence, Henan University, China
zhoulin@henu.edu.cn, zlu_20@163.com, jy@henu.edu.cn, hzt@henu.edu.cn, lijunwei@henu.edu.cn

Abstract

Location awareness has been widely used for cooperative localization of target in wireless sensor networks (WSN). But as the number of agent node increases, cooperative localization based on nonparametric belief propagation (BP) causes high communication and computation burden. In addition, high localization accuracy is also the goal of this paper. To this end, this paper based on parameterized BP strategy proposed a distributed cubature Kalman filter (DCKF) algorithm named BP-DCKF. Firstly, basing on joint posterior probability density function of all nodes, this paper constructs a factor graph (FG) model, then the edge posterior distribution of the nodes is obtained by BP strategy. Secondly, considering Gaussian parameterized BP which reconstruct the parameterized message of agents transmitting, this paper proposed the improved DCKF method, moreover, obtaining the locating model of agent node related to posterior distribution of each node in the FG. Finally, based on the localization model, the location estimation of mobile agent node can be obtained utilizing DCKF method to iteratively solve the edge posterior distribution of agent node. Simulation results show that comparing with traditional algorithm, the proposed algorithm is higher on localization accuracy, lower on communication burden.

Keywords: Wireless sensor networks (WSN), Node localization, Cubature Kalman filter (CKF), Factor graph (FG), Belief propagation (BP)

1 Introduction

The localization is a fundamental issue in wireless sensor networks (WSN). It is a challenging to obtain the accurate location of sensors (or nodes). Different with non-cooperative localization (CL), the CL technology estimate location of sensor node itself by cooperating with each other, which can effectively solve the localization fail caused by the insufficient reference (or anchor) nodes in the traditional technology [1-2]. Since additional infrastructures are unnecessary, the CL technology are already widely used in some fields such as navigation, vehicle network, environmental and agricultural detection, internet of things and so on [3-8]. Therefore, to improve the overall localization performance of WSN, some researchers have focused on CL.

1.1 Related Work

In WSN, some researchers focused on non-Bayesian CL methods, but their error accumulation caused by deterministic state estimation. To solve the problem, Wymeersch et al. presented a framework of CL based on the FG and message passing schedule, and named a sum-product algorithm for wireless network (SPAWN) [9], which the belief propagation (BP) method was applied to estimate the posterior marginal probability density function (PDF) of the agent nodes location. To extend the application of SPAWN, Ihler et al. proposed a non-parametric BP (NBP) to locate agent nodes in the nonlinear system [10-11]. Savic et al. extended NBP to localization related to mobile agent nodes network. Slavic employed cooperative between agent nodes to transmit non-parametric messages firstly, then obtained weighted particles of localization node variables by distributed computation method, leading to the posterior probability distribution of node localization variables [12]. Wymeersch et al. proposed a weighted BP method to weight and quantify the message transmission posterior probability of variable nodes in the FG, enhancing stability of message transmission [13]. Although the above method can better improve the localization accuracy of agent nodes, leading to the increase of communication burden between agent nodes.

In order to reduce the communication burden between agent nodes, Pedersen proposed the variational message passing (VMP) algorithm, but it needed to minimize Kullback-Leibler (KL) divergence [14]. Xiong et al. proposed a localization method based on extended Kalman filter algorithm to improve localization accuracy [15], but the linearization of this method produced systematic error. Meyer et al. proposed a distributed CL named SPBP based on sigma point BP, and obtained the posterior PDF of each variable node in FG with lower communication burden and computational consumption [16]. However, in static network with many adjacent nodes, the SPBP algorithm caused higher dimensionality of transmitting message, leading to non-positive covariance matrix. Arasaratnam et al. proposed a CKF algorithm [17], which uses a deterministic particle sampling without linearization. At the same time, the agent node can pass only the mean and covariance matrices, which reduce the communication burden between the nodes. In addition, relative to a centralized localization method, the distributed localization method has a lower computation burden. In distributed localization, the agent node updates its own location by receiving the message of the neighboring node, and can effectively improve the localization accuracy while effective reducing the communication burden between the nodes.

*Corresponding Author: Lu Zhang; E-mail: zlu_20@163.com

DOI: 10.53106/160792642022052303008

1.2 Contributions and Paper Organization

In summary, CL is one of the most important issues in WSN. In order to solve the problems of heavy communication burden and low localization accuracy. This paper proposes BP-DCKF algorithm based on parametric messages passing scheme, the main contributions as follows:

(1) Compared with some traditional methods without CL, the proposed algorithm considers cooperative among agent nodes, leading to higher localization accuracy, the deployment of reference nodes can be reduced and the network cost can be saved.

(2) The proposed algorithm can autonomously process message of each agent node by CKF algorithm. Thus, it is essentially a distributed structure, thus it has better robustness and lower computation burden than traditional centralized methods.

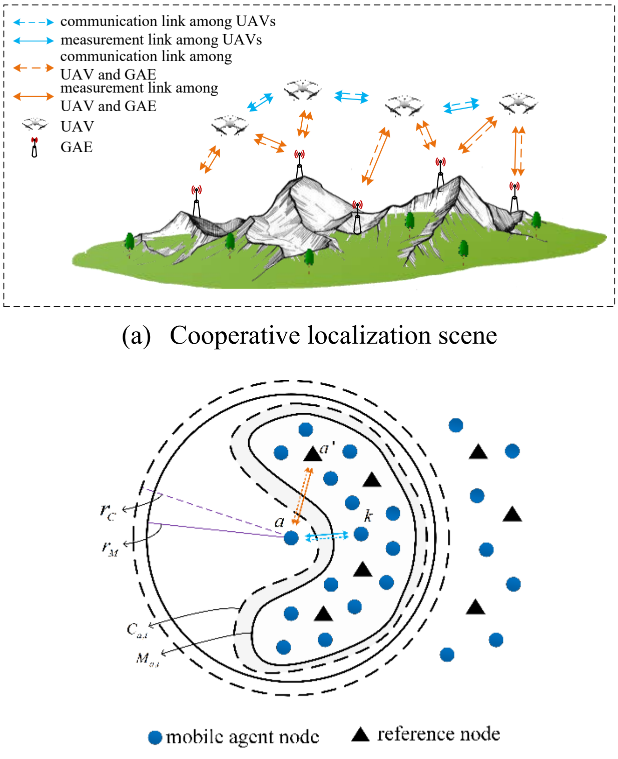
(3) Compared with non-parametric messages passing, the proposed algorithm only transmits parameterized messages (i.e., mean vector, covariance matrix), thus it has lower communication burden and better real time performance.

The rest of this paper is organized as follows: section 2 gives a system model to describe WSNs, section 3 briefly introduces involved knowledge about belief propagation between nodes, section 4 detailly introduces proposed algorithm, section 5 verifies feasibility of the proposed algorithm through simulation and analyzes performance, section 6 gives some conclusions of this paper.

2 System Model of WSN

Distributed CL is shown in Figure 1(a), in which an unmanned aerial vehicle (UAV) detects other unmanned aerial vehicles (UAVs) and ground auxiliary equipments (GAEs). Assuming localization error of GAE is not considered, and UAV can obtain distance between itself and other UAVs or GAEs, thus communication link for message transmission can be established. In this scene, every UAV's own localization estimation is received by cooperating with ground equipment and other UAVs.

The UAV and the GAE in Figure 1(a) are denoted as mobile agent node, reference node, respectively. Without loss of generality, localization problem of a micro-UAV network can be modelled as a mobile sensor network which can be described in two-dimension space [18]. As shown in Figure 1(b), where $A \in N$ is a set of all reference nodes and mobile agent nodes, $(a', a, k) \in A$ represents a reference node, an agent node, and an arbitrary node, respectively. At time t, $C_{a,t}$ and $M_{a,t}$ denote set of the within communication radius r_C and measurement radius r_M topology nodes associated with agent node a, respectively. When node k is within the communication range and measurement range of node a, adjacent communication node $k \in C_{a,t}$, $C_{a,t}$ is a subset of A and denoted by $M_{a,t} \subseteq A \backslash (a)$.


communication link among UAVs
measurement link among UAVs
communication link among UAV and GAE
measurement link among UAV and GAE
UAV
GAE

(a) Cooperative localization scene

r_C
r_M
$C_{a,t}$
$M_{a,t}$
a'
a
k
mobile agent node
reference node

(b) Cooperative localization model

Figure 1. Distributed mobile agent nodes cooperative localization

Assuming a mobile agent node a has state $\boldsymbol{x}_{a,t}$ at time t, and measured $y_{a,k;t}$ between its own and adjacent node k, which can be described using Equations (1)-(3):

$$\boldsymbol{x}_{a,t} = g\left(\boldsymbol{x}_{a,t-1}, \boldsymbol{u}_{a,t}\right),\ a \in A,\ t = 1, 2, \cdots, T. \tag{1}$$

$$y_{a,k;t} = h\left(\boldsymbol{x}_{a,t},\ \boldsymbol{x}_{k,t},\ \nu_{a,k;t}\right),\ a \in A,\ k \in M_{a,t}. \tag{2}$$

$$y_{a,k;t} = \left\| \tilde{\boldsymbol{x}}_{a,t} - \tilde{\boldsymbol{x}}_{k,t} \right\| + \nu_{a,k;t},\ a \in A,\ k \in M_{a,t}. \tag{3}$$

where, $\boldsymbol{x}_{a,t} \in \mathrm{R}\hbar$ is n dimensions state vectors, $u_{a,t} - N(0 , \delta^2 I)$ is white Gaussian process noise, $g(\cdot)$ is a linear/nonlinear function of state transition. Here, the statistical relation between $\boldsymbol{x}_{a,t-1}$ and $\boldsymbol{x}_{a,t}$ can also be described by the state-transition PDF $f\left(\boldsymbol{x}_{a,t} \mid \boldsymbol{x}_{a,t-1}\right)$. $y_{a,k;t}$ is the measurement between agent node a and adjacent node k, $\nu_{a,k;t} - N(0, \sigma^2)$ is white Gaussian measure noise, $h(\cdot)$ is a nonlinear measurement function which may specified by Euclidean norm described as Equation (3). The statistical dependence of $y_{a,k;t}$ on $\boldsymbol{x}_{a,t}$ and $\boldsymbol{x}_{k,t}$ can be described by the local likelihood function $f\left(y_{a,k;t} \mid \boldsymbol{x}_{a,t}, \boldsymbol{x}_{k,t}\right)$.

Assuming the state of all agent nodes and measurement at time $t \in \{1, 2, \cdots, T\}$ is $\boldsymbol{x}_t = \left(\boldsymbol{x}_{a,t}\right)$, $y_t = \left(y_{a,k;t}\right)$, respectively. So, states and measurements at all total time can be described as $\boldsymbol{x}_{1:T} = \left(\boldsymbol{x}_1^{\mathrm{T}}, \cdots, \boldsymbol{x}_T^{\mathrm{T}}\right)^{\mathrm{T}}$ and $\boldsymbol{y}_{1:T} = \left(y_1, \cdots, y_T\right)^{\mathrm{T}}$, respectively, and the nodes satisfy assumption that are consistent in [16].

According to Bayesian estimation, the joint posterior PDF of all nodes state $\boldsymbol{x}_{1:T}$ can be factorized as Equation (4), therefore, local state estimation of agent nodes according to the edge posterior PDF, then global state estimation can be achieved by BP:

$$f\left(\boldsymbol{x}_{1:T} \mid \boldsymbol{y}_{1:T}\right) \propto \prod_{a \in A} f\left(\boldsymbol{x}_{a,0}\right) \cdot \prod_{t=1}^{T}\left[\prod_{a \in A} f\left(\boldsymbol{x}_{a,t} \mid \boldsymbol{x}_{a,t-1}\right) \cdot \prod_{k \in M_{a,t}} f\left(y_{a,k;t} \mid \boldsymbol{x}_{a,t}, \boldsymbol{x}_{k,t}\right)\right] \tag{4}$$

3 Belief Propagation Between Nodes

The SPAWN algorithm is very suitable for distributed CL in WSNs, based on its advantage, this paper uses the distributed message transmission between adjacent nodes to realize CL of nodes.

The FG of the joint posterior PDF $f\left(\boldsymbol{x}_{1:T} \mid \boldsymbol{y}_{1:T}\right)$ contains variable nodes, factor nodes, the edge structure connecting variable nodes and the factor nodes based on Equation (4). To describe the process, insert a specific example, such as Figure 2 shows the connection state between the agent node $\boldsymbol{x}_a$, $a \in \{1,2\}$ and the reference node $\boldsymbol{x}_{a'}$, $a' \in \{1,2,3\}$, from the time $t-1$ to t.

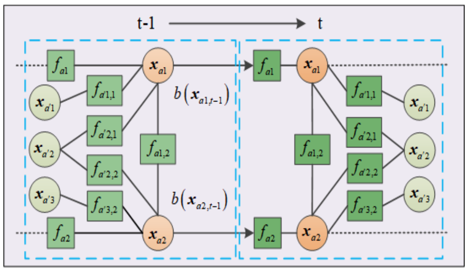


Figure 2. Message passing in the FG

Supposing an agent node a can observe the adjacent node k, the state relationship between adjacent node and agent node can be described by the factor node $f_{a,k;t} = f\left(y_{a,k;t} \mid \boldsymbol{x}_{a,t}, \boldsymbol{x}_{k,t}\right)$, $f_{a,t} = f\left(\boldsymbol{x}_{a,t} \mid \boldsymbol{x}_{a,t-1}\right)$ is the factor node to describe state transition probability between $\boldsymbol{x}_{a,t}$ and $\boldsymbol{x}_{a,t-1}$. It can be found that the message passing in the FG is divided into two types, i.e., from variable nodes to factor nodes, and from factor nodes to variable nodes. The process of message passing and nodes state updating is shown in Figure 2.

According to message passing strategy in the FG, edge posterior PDF $f\left(\boldsymbol{x}_{a,t} \mid \boldsymbol{y}_{1:t}\right)$ of the agent node $\boldsymbol{x}_{a,t}$ can be approximated by belief $b^{(n)}\left(\boldsymbol{x}_{a,t}\right)$ described by Equation (5) as follows:

$$b^{(n)}\left(\boldsymbol{x}_{a,t}\right) \propto \varphi_{f_a \to a}\left(\boldsymbol{x}_{a,t}\right) \cdot \prod_{k \in M_{a,t}} m_{k \to a}^{(n)}\left(\boldsymbol{x}_{a,t}\right) \tag{5}$$

where, $n \in \{1, \cdots, N_{iter}\}$ is the number of message iterations in loopy FG at time t, the predicted message $\varphi_{f_a \to a}\left(\boldsymbol{x}_{a,t}\right)$ from the factor node f_a to the variable node $\boldsymbol{x}_{a,t}$, it can be obtained by Equation (6):

$$\varphi_{f_a \to a}\left(\boldsymbol{x}_{a,t}\right) = \int f\left(\boldsymbol{x}_{a,t} \mid \boldsymbol{x}_{a,t-1}\right) \cdot b^{(n)}\left(\boldsymbol{x}_{a,t-1}\right) d\boldsymbol{x}_{a,t-1} \tag{6}$$

where, the state transition probability function and the belief at time $t-1$ are defined as $f\left(\boldsymbol{x}_{a,t} \mid \boldsymbol{x}_{a,t-1}\right)$ and $b^{(n)}\left(\boldsymbol{x}_{a,t-1}\right)$, respectively. In Equation (5), the measurement message $m_{k \to a}^{(n)}\left(\boldsymbol{x}_{a,t}\right)$ passed form the factor node $f_{a,k;t}$ to the variable node $\boldsymbol{x}_{a,t}$ is obtained by (7), and the belief of adjacent nodes passed form the variable node to the factor node is obtained by (8) as follow:

$$m_{k \to a}^{(n)}\left(\boldsymbol{x}_{a,t}\right) = \int f\left(y_{a,k;t} \mid \boldsymbol{x}_{a,t}, \boldsymbol{x}_{k,t}\right) \cdot b^{(n-1)}\left(\boldsymbol{x}_{k,t}\right) d\boldsymbol{x}_{k,t} \tag{7}$$

$$b^{(n-1)}\left(\boldsymbol{x}_{k,t}\right) \propto \varphi_{f_k \to k}\left(\boldsymbol{x}_{k,t}\right) \cdot \prod_{u \in M_{k,t} \setminus \{a\}} m_{u \to k}^{(n-1)}\left(\boldsymbol{x}_{k,t}\right) \tag{8}$$

where, $b^{(n-1)}\left(\boldsymbol{x}_{k,t}\right)$ represents the belief adjacent node k of mobile agent node a, u is the adjacent node of node k.

Directly solving Equation (7) is difficult due to nonlinearity of Equation (3), iterative approximate method based on random particle sampling may can handle above problem with high computational burden [19]. In this paper, we employ cubature Kalman filter which is based on deterministic sampling method to solve Equation (7) with low computational burden.

Belief $b^{(n)}\left(\boldsymbol{x}_{a,t}\right)$ of the mobile agent node a is iteratively calculated n times at time t, and the belief is approximated to the edge posterior PDF of the localization variable, so the minimum mean square error (MMSE) estimation of each node can be calculated. So the agent node belief $b^{(n)}\left(\boldsymbol{x}_{a,t}\right)$ is obtained by n times iteration to approximate the edge posterior PDF $f\left(\boldsymbol{x}_{a,t} \mid \boldsymbol{y}_{1:t}\right)$ of the localization variable, leading to $\hat{\boldsymbol{x}}_{a,t}^{MMSE}$ with respect to location estimation of agent node a [15]:

$$\hat{\boldsymbol{x}}_{a,t}^{MMSE} = \int \boldsymbol{x}_{a,t} \cdot f\left(\boldsymbol{x}_{a,t} \mid \boldsymbol{y}_{1:t}\right) d\boldsymbol{x}_{a,t} \tag{9}$$

4 BP-DCKF Cooperative Localization Algorithm

Based on section 3, this section mainly presents the implementation of Gaussian parameterized BP and message reconstruction strategies, as well as derives the multidimensional Gaussian model representation of the belief and the posterior distribution of variable nodes in the FG. The agent node uses DCKF update message to realize localization. Finally, details of the BP-DCKF algorithm is given.

4.1 Gaussian Parameterized BP and Reconstruction Strategy

Gaussian parameterized BP and message reconstruction strategy, which use adjacent agent nodes to transmit Gaussian parameters representing their own belief firstly, and then reconstruct a combined message including adjacent nodes state and measurements, calculate the posterior distribution of agent nodes state variables in a high-dimensional space finally. Since effective using the belief of adjacent nodes, this strategy can enhance the localization accuracy of the agent, leading to the localization performance of network improving.

The number of agent node a adjacent nodes is denoted by $N_{a,t} \in \{1,2,\cdots,N\}$ at time t. High dimensional combined state vector consisted of agent node state $\boldsymbol{x}_{a,t}$ and the adjacent agent node state $\boldsymbol{x}_{k,t}$ can be defined as $\boldsymbol{X}_{at} = \left(\boldsymbol{x}^{\mathrm{T}}_{a,t}, \boldsymbol{x}^{\mathrm{T}}_{k_1,t}, \cdots, \boldsymbol{x}^{\mathrm{T}}_{k_{N_{a,t}},t}\right)^{\mathrm{T}}$. High-dimensional combined measurement vector consisted of distance measurements related to the agent node a can be defined as $\boldsymbol{Y}_{a,t} = \left(y_{a,k1;t}, y_{a,k2;t}, \cdots, y_{a,k_{N_{a,t}};t}\right)^{\mathrm{T}}$. Thus, the dimension of $\boldsymbol{X}_{a,t}$ can be describe as $D_a = d_a + \sum_{k=1}^{N_{a,t}} d_k$, where d_a represents the dimension of $\boldsymbol{x}_{a,t}$, and d_k represents the dimension of the adjacent node state $\boldsymbol{x}_{k,t}$. According to the BP strategy, we can obtain Equation (10) by substituting Equation (7) into Equation (5):

$$b^{(n)}\left(\boldsymbol{x}_{a,t}\right) \propto f\left(\boldsymbol{x}_{a,t}\right)\cdot\int\prod_{k\in M_{a,t}} f\left(y_{a,k;t} \mid \boldsymbol{x}_{a,t}, \boldsymbol{x}_{k,t}\right)\cdot b^{(n-1)}_{k\to a}\left(\boldsymbol{x}_{k,t}\right) \quad \textbf{(10)}$$

Note that $f\left(\boldsymbol{x}_{a,t}\right)$ to be the prior distribution of the agent node and to be describes as $\varphi_{f_a\to a}\left(\boldsymbol{x}_{a,t}\right)$, and then the agent node belief can be calculated by the combining state $\boldsymbol{X}_{a,t}$ and the measurement message $\boldsymbol{Y}_{a,t}$. The belief calculation Equation (10) can be rewritten as Equation (11):

$$b^{(n)}\left(\boldsymbol{x}_{a,t}\right) \propto \int f\left(\boldsymbol{Y}_{a,t} \mid \boldsymbol{X}_{a,t}\right)\cdot f^{(n-1)}\left(\boldsymbol{X}_{a,t}\right) d\boldsymbol{X}^{\to a}_{a,t} \quad \textbf{(11)}$$

where, the combination state $\boldsymbol{X}^{\to a}_{a,t}$ can be obtained by the removal of the sub-vector $\boldsymbol{x}_{a,t}$ from $\boldsymbol{X}_{a,t}$, $f\left(\boldsymbol{Y}_{a,t} \mid \boldsymbol{X}_{a,t}\right)$ denotes the likelihood function described as Equation (12), $f^{(n-1)}\left(\boldsymbol{X}_{a,t}\right)$ is given by Equation (13) is the independent prior probability distribution related to iterative process:

$$f\left(\boldsymbol{Y}_{a,t} \mid \boldsymbol{X}_{a,t}\right) = \prod_{k\in M_{a,t}} f\left(y_{a,k;t} \mid \boldsymbol{x}_{a,t}, \boldsymbol{x}_{k,t}\right) \quad \textbf{(12)}$$

$$f^{(n-1)}\left(\boldsymbol{X}_{a,t}\right) \propto f\left(\boldsymbol{x}_{a,t}\right)\cdot\prod_{k\in M_{a,t}} b^{(n-1)}_{k\to a}\left(\boldsymbol{x}_{k,t}\right) \quad \textbf{(13)}$$

According to the Bayesian estimation theory, to obtain the measurement message, it is necessary to calculate the combination measurement message represented by Equation (14):

$$\boldsymbol{Y}_{a,t} = H\left(\boldsymbol{X}_{a,t}\right) + \boldsymbol{V}_{a,t} \quad \textbf{(14)}$$

where, $H\left(\boldsymbol{X}_{a,t}\right) = \left(h\left(\boldsymbol{x}_{a,t}, \boldsymbol{x}_{k_1,t}\right), \cdots, h\left(\boldsymbol{x}_{a,t}, \boldsymbol{x}_{k_{N_{a,t}},t}\right)\right)^{\mathrm{T}}$ $\boldsymbol{V}_{a,t} = \left(v_{a,k_1;t}, \cdots, v_{a,k_{N_{a,t}};t}\right)^{\mathrm{T}}$, and Equation (11) can be rewritten as Equation (15), which $b^{(n)}\left(\boldsymbol{X}_{a,t}\right)$ represent the combined belief that is calculated by Equation (16):

$$b^{(n)}\left(\boldsymbol{x}_{a,t}\right) \propto \int b^{(n)}\left(\boldsymbol{X}_{a,t}\right) d\boldsymbol{X}^{\to a}_{a,t} \quad \textbf{(15)}$$

$$b^{(n)}\left(\boldsymbol{X}_{a,t}\right) \propto f\left(\boldsymbol{Y}_{a,t} \mid \boldsymbol{X}_{a,t}\right)\cdot f^{(n-1)}\left(\boldsymbol{X}_{a,t}\right) \quad \textbf{(16)}$$

The localization variables of the agent node need satisfy the Gaussian distribution, the belief and probability distribution of the variable nodes in the FG accord with the multi-dimensional Gaussian distribution model. Employing the CKF algorithm into the FG, this paper defines the combined mean vector $\boldsymbol{\mu}^{(n-1)}_{X_{a,t}}$ and the combined covariance matrix $\boldsymbol{C}^{(n-1)}_{X_{a,t}}$ to represent the combined prior distribution message $f^{(n\text{-}1)}\left(\boldsymbol{X}_{a,t}\right)$ according to the Gaussian parameterized BP and reconstruction strategy, and the defined $\boldsymbol{\mu}^{(n-1)}_{X_{a,t}}$ and $\boldsymbol{C}^{(n-1)}_{X_{a,t}}$ as follows, respectively:

$$\boldsymbol{\mu}^{(n-1)}_{X_{a,t}} = \left(\left(\boldsymbol{\mu}^{(n-1)}_{X_{a,t}}\right)^{\mathrm{T}}, \left(\boldsymbol{\mu}^{(n-1)}_{k1\to a,t}\right)^{\mathrm{T}}, \cdots, \left(\boldsymbol{\mu}^{(n-1)}_{k_{N_{a,t}}\to a,t}\right)^{\mathrm{T}}\right)^{\mathrm{T}} \quad \textbf{(17)}$$

$$\boldsymbol{C}^{(n-1)}_{X_{a,t}} = \mathrm{diag}\left(\boldsymbol{C}^{(n-1)}_{a,t}, \boldsymbol{C}^{(n-1)}_{k_1\to a,t}, \cdots, \boldsymbol{C}^{(n-1)}_{k_{N_{a,t}}\to a,t}\right) \quad \textbf{(18)}$$

where, the dimension of $\boldsymbol{\mu}^{(n-1)}_{X_{a,t}}$ is $D_a \times 1$, and the dimension of $\boldsymbol{C}^{(n-1)}_{X_{a,t}}$ is $D_a \times D_a$. The Gaussian parameterization of the belief $b^{(n-1)}_{k\to a}\left(\boldsymbol{x}_{k,t}\right)$ related to the adjacent nodes can be expressed as the mean vector $\boldsymbol{\mu}^{(n-1)}_{k\to a,t}$ and the covariance matrix $\boldsymbol{C}^{(n-1)}_{k\to a,t}$, respectively. $\mathrm{diag}\left(\cdot\right)$ is a high-dimensional block diagonal matrix composed of multiple agent node covariance matrices. Each agent node uses the CKF algorithm to calculate the mean vector $\boldsymbol{\mu}^{(n)}_{a,t}$ and the covariance matrix $\boldsymbol{C}^{(n)}_{a,t}$, and approximates the belief $b^{(n)}\left(\boldsymbol{x}_{a,t}\right)$ of the node state $\boldsymbol{x}_{a,t}$ to its edge posterior PDF, denoted as $f(\boldsymbol{x}_{a,t}|\boldsymbol{y}_{1:t}) - N(\boldsymbol{x}_{a,t}|\boldsymbol{\mu}^{(n)}_{a,t}, \boldsymbol{C}^{(n)}_{a,t})$.

4.2 DCKF Message Processing

Based on above sections, the detailed steps of the DCKF cooperative localization algorithm based on BP method, that is, the proposed algorithm is described in Figure 3.

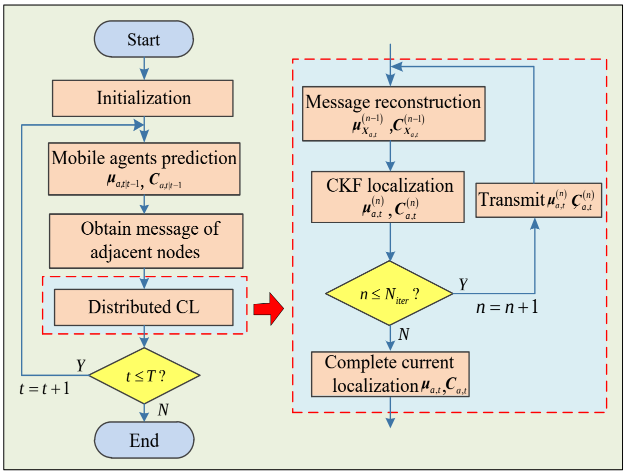

Start
Initialization
Mobile agents prediction $\boldsymbol{\mu}_{a,t|t-1}$, $\boldsymbol{C}_{a,t|t-1}$
Obtain message of adjacent nodes
Distributed CL
$t \le T$?
Y
$t = t+1$
N
End
Message reconstruction $\boldsymbol{\mu}_{X_{a,t}}^{(n-1)}$,$\boldsymbol{C}_{X_{a,t}}^{(n-1)}$
CKF localization $\boldsymbol{\mu}_{a,t}^{(n)}$,$\boldsymbol{C}_{a,t}^{(n)}$
Transmit $\boldsymbol{\mu}_{a,t}^{(n)}$ $\boldsymbol{C}_{a,t}^{(n)}$
$n \le N_{iter}$?
Y
$n = n+1$
N
Complete current localization $\boldsymbol{\mu}_{a,t}$,$\boldsymbol{C}_{a,t}$

Figure 3. Flow chart of the BP-DCKF

At the initial time, the mobile agent node performs the initial message representation according to the neighboring reference nodes. According to [20], we assume that the message of the mobile agent node may satisfy Gaussian distribution when mobile agent has three reference nodes.

Parametric message CL mainly includes three parts: mobile agent nodes predict message, message reconstruc- tion and message update, they are introduced as follows:

(1) Mobile agent nodes predict message

All mobile agent nodes $a \in A$ predicted state vector and covariance matrix are gotten by the function of state transition, the prior distribution message $f\left(\boldsymbol{x}_{a,t}\right)$ can be denoted by Gaussian distribution whose mean vector and covariance matrix are described as $\boldsymbol{\mu}_{a,t|t-1}$ and $\boldsymbol{C}_{a,t|t-1}$, respectively, and they can be obtained by Equation (19) and (20) as follows:

$$\begin{cases} \boldsymbol{x}_{a,t|t-1}^{(i)} = g\left(\xi_{a,t-1}^{(i)},\ \boldsymbol{u}_{a,t-1}\right),\ i = 1,\cdots,2d_a \\ \boldsymbol{\mu}_{a,t|t-1} = \dfrac{1}{2d_a}\displaystyle\sum_{i=1}^{2d_a} \boldsymbol{x}_{a,t|t-1}^{(i)} \end{cases} \tag{19}$$

$$\boldsymbol{C}_{a,t|t-1} = \frac{1}{2d_a}\sum_{i=1}^{2d_a}\left(\boldsymbol{x}_{a,t|t-1}^{(i)} - \boldsymbol{\mu}_{a,t|t-1}\right)\cdot\left(\boldsymbol{x}_{a,t|t-1}^{(i)} - \boldsymbol{\mu}_{a,t|t-1}\right)^{\mathrm{T}} + \boldsymbol{Q}_{t-1} \tag{20}$$

where, $\xi_{a,t-1}^{(i)}$ represents the $i-th$ cubature point of the state of agent node a, $\boldsymbol{Q}_{t-1}$ is the covariance matrix of process noise at time $t-1$.

(2) Message reconstruction

The number of message iterations is denoted by $n \in \{1,\cdots,N_{iter}\}$. The combined mean vector $\boldsymbol{\mu}_{X_{a,t}}^{(n-1)}$ and the combined covariance matrix $\boldsymbol{C}_{X_{a,t}}^{(n-1)}$ are described in Equation (17) and (18) by the message reconstruction strategy to represent the prediction of combined prior distribution message $f^{(n-1)}\left(\boldsymbol{X}_{a,t}\right)$.

(3) Updated nodes message

The mobile agent obtains the mean and covariance as its belief. Table 1 is a segment of the proposed BP-DCKF algorithm, which mainly describes the update process of the agent nodes.

Table 1. Update node message

```
Algorithm 1. DCKF message update
1. Agent $a \in A$ in parallel
2. Computing combine cubature points:
     $\hat{\boldsymbol{X}}_{a,t}^{(i)} = \boldsymbol{\mu}_{X_{a,t}}^{(n-1)} + \sqrt{\boldsymbol{C}_{X_{a,t}}^{(n-1)}}\hat{\boldsymbol{\xi}}^{(i)},\ i = 1,\cdots,2D_a$
3. Computing cubature points propagation and measur-
   ement prediction:
     $\hat{\boldsymbol{Y}}_{a,t}^{(i)} = H\left(\hat{\boldsymbol{X}}_{a,t}^{(i)}\right) + \boldsymbol{V}_{a,t},\ i = 1,\cdots,2D_a$
     $\boldsymbol{m}_{a,t}^{(n)} = \frac{1}{2D_a}\sum_{i=1}^{2D_a}\hat{\boldsymbol{Y}}_{a,t}^{(i)}$
4. Computing covariance and cross-covariance matrix:
     $\boldsymbol{P}_{YY}^{(n)} = \frac{1}{2D_a}\sum_{i=1}^{2D_a}\left(\hat{\boldsymbol{Y}}_{a,t}^{(i)} - \boldsymbol{m}_{a,t}^{(n)}\right)\left(\hat{\boldsymbol{Y}}_{a,t}^{(i)} - \boldsymbol{m}_{a,t}^{(n)}\right)^{\mathrm{T}} + \boldsymbol{R}_{a,t}$
     $\boldsymbol{S}_{XY}^{(n)} = \frac{1}{2D_a}\sum_{i=1}^{2D_a}\left(\hat{\boldsymbol{X}}_{a,t}^{(i)} - \boldsymbol{\mu}_{X_{a,t}}^{(n-1)}\right)\left(\hat{\boldsymbol{X}}_{a,t}^{(i)} - \boldsymbol{\mu}_{X_{a,t}}^{(n-1)}\right)^{\mathrm{T}}$
5. Computing filter gain [6]:
     $\boldsymbol{K}_{a,t}^{(n)} = \boldsymbol{S}_{XY}^{(n)}\left(\boldsymbol{P}_{YY}^{(n)}\right)^{-1}$
6. Computing mobile agent's mean and covariance:
     $\boldsymbol{\mu}_{a,t}^{(n)} = \boldsymbol{\mu}_{a,t}^{(n-1)} + \boldsymbol{K}_{a,t}^{(n)}\left(\boldsymbol{Y}_{a,t} - \boldsymbol{m}_{a,t}^{(n)}\right)$
     $\boldsymbol{C}_{a,t}^{(n)} = \boldsymbol{C}_{a,t}^{(n-1)} - \boldsymbol{K}_{a,t}^{(n)}\boldsymbol{P}_{YY}^{(n)}\left(\boldsymbol{K}_{a,t}^{(n)}\right)^{\mathrm{T}}$
7. Computing nodes belief message:
     $b(\boldsymbol{x}_{a,t}) \sim N(\boldsymbol{x}_{a,t} \mid \boldsymbol{u}_{a,t}, \boldsymbol{C}_{a,t})$
8. end parallel
```

4.3 BP-DCKF Algorithm Analysis and Comparison

Base on above Section 4.1 and 4.2, this paper proposed BP-DCKF algorithm is described in Table 2.

In the $15-th$ step of Table 2, the number of cubature points required by BP-DCKF algorithm is $2D_a$, which the mean and covariance matrix can be obtained by Gaussian parameterization. Only passed mean and covariance matrix using this way, so the communication burden of the proposed algorithm is $6N_a$, and state updating computational cost of agent node need to handle $2\times 8N_a$ elements. While the communication burden of non-parametric BP strategy based SPAWN1 and SPAWN2 is $2J\cdot N_a$ and $2J\cdot N_a$, respectively, and calculated overhead are $2J\cdot N_a$ and $2J^2\cdot N_a$, where J and N_a indicated particle number and number o adaacent nodes. The communication burden o VMP algorithm based on parameterized-BP is $3N_a$, but computational cost is $3N_a + 2J$ due to minimizing particles KL divergence. As can be seen rom Table 3, the overhead o the proposed algorithm is the lowest, because usually the number N_a o adaacent nodes is much less than the number J o particles.

Table 2. Pseudo-code of the proposed algorithm

Algorithm 2. BP-DCKF

```
Initialization
Agent node $a \in A$ in parallel
end parallel
for $t = 1 : T$ do
    Nodes $a \in A$ in parallel
    Computing prediction message $\varphi_{f_a \to a}\left(\boldsymbol{x}_{a,t}\right)$
      according to Equation (19) and (20)
    Computing measurement message $m_{a' \to a}\left(\boldsymbol{x}_{a,t}\right)$
      using Equation (7)
    Computing initialize belief message $b\left(\boldsymbol{x}_{a,t}\right)$
      using Equation (10)
    end parallel
    for $n = 1 : N_{iter}$ do
        Nodes $a \in A$ in parallel
        Receiving adjacent nodes message
          $\mathrm{N}\left(\boldsymbol{x}_{k,t} \mid \boldsymbol{\mu}_{k,t}^{(n-1)}, \boldsymbol{C}_{k,t}^{(n-1)}\right)$
        Computing combine message $f^{(n-1)}\left(\boldsymbol{X}_{a,t}\right)$
          according to Equation (17) and (18)
        Running Algorithm 1
        Estimating state $\hat{\boldsymbol{x}}_{a,t}^{(n)}$ of agent node $a$
        Transmitting message $b(\boldsymbol{x}_{a,t}) -$
          $\mathrm{N}\left(\boldsymbol{x}_{a,t} \mid \boldsymbol{\mu}_{a,t}^{(n)}, \boldsymbol{C}_{a,t}^{(n)}\right)$
        end parallel
        if $n < N_{iter}$
            $n = n + 1$
        end if
    end for
    if $t < T$
        $t = t + 1$
    end if
end for
```

Table 3. Cost of different algorithms

Algorithm	Calculated Cost	Communicated Cost
SPAWN1	$2J \cdot N_a$	$2J \cdot N_a$
SPAWN2	$2J^2 \cdot N_a$	$2J \cdot N_a$
VMP	$3N_a + 2J$	$3N_a$
BP-DCKF	$2 \times 8N_a$	$6N_a$

5 Simulation Results and Analysis

Two scenes are made in this simulation section: static link networks scene and dynamic link networks scene, which are designed to illustrate the feasibility and effectiveness of the proposed BP-DCKF algorithm. Besides, the performance of the BP-DCKF algorithm is verified by comparing with some algorithms including VMP [14], the one NBP based sum-product algorithm for wireless network named SPAWN1 [11] and another NBP based sum-product algorithm for wireless network named SPAWN2 [12] algorithm.

In simulation, two models are used to describe the motion of mobile agents as follows:

$$\boldsymbol{x}_{a,t} = \boldsymbol{F}\boldsymbol{x}_{a,t-1} + \boldsymbol{R}\boldsymbol{u}_{a,t},\ a \in A,\ t = 1, \cdots, T$$

where, the one is uniform linear model and state transition matrix $\boldsymbol{F}$ can be defined as $\boldsymbol{G}_1$. Another is curve model and state transition matrix $\boldsymbol{F}$ can be defined as $\boldsymbol{F} = \begin{cases} \boldsymbol{G}_1,\ \omega_{a,t} = 0 \\ \boldsymbol{G}_2,\ \omega_{a,t} \neq 0 \end{cases}$, $\boldsymbol{R}$ is driving matrix, $\boldsymbol{G}_1$ and $\boldsymbol{G}_2$ are set as follows:

$$\boldsymbol{G}_1 = \begin{pmatrix} 1 & 0 & \tau & 0 \\ 0 & 1 & 0 & \tau \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$\boldsymbol{G}_2 = \begin{pmatrix} 1 & 0 & \sin\left(\omega_{a,t}\tau\right) / \omega_{a,t} & \left(\cos\left(\omega_{a,t}\tau\right) - 1\right) / \omega_{a,t} \\ 0 & 1 & \left(1 - \cos\left(\omega_{a,t}\tau\right)\right) / \omega_{a,t} & \sin\left(\omega_{a,t}\tau\right) / \omega_{a,t} \\ 0 & 0 & \cos\left(\omega_{a,t}\tau\right) & -\sin\left(\omega_{a,t}\tau\right) \\ 0 & 0 & \sin\left(\omega_{a,t}\tau\right) & \cos\left(\omega_{a,t}\tau\right) \end{pmatrix}$$

where, τ is the sampling interval, $\omega_{a,t}$ is the turning angular velocity. The initial state of all agent nodes is represented by $\boldsymbol{\mu}_{a,0}$ and the initial covariance matrix $\boldsymbol{C}_{a,0} = \mathrm{diag}\left(10,\ 10,\ 0.01,\ 0.01\right)$.

The main parameters related to experiment simulation are shown in Table 4.

In order to evaluate the performance of the BP-DCKF algorithm, the average root mean square error (ARMSE) and average running time are used to evaluate them, and the ARMSE is also used to measure the localization performance of all algorithms:

$$ARMSE = \sqrt{\sum_{j=1}^{M_c} \sum_{a=1}^{A_a} \frac{\left\| \hat{\boldsymbol{x}}_{a,j} - \tilde{\boldsymbol{x}}_a \right\|^2}{A_a M_c}}$$

where, A_a is the number of agent nodes to be located, $\hat{\boldsymbol{x}}_{a,j}$ is the state estimation localization vector of mobile agent node a in the $j-th$ Monte Carlo run and $\tilde{\boldsymbol{x}}_a$ is the actual location vector.

5.1 Scene 1: Agent Node Linear Moving in the Static Link Network

In this scene, agents move along a straight line in the area which was deployed static link network. In addition, the connection relationship between all nodes in the static link network will not be changed in the time of agents moving. The mobile agent nodes can measure all nodes and transmit Gaussian parameterized message to adjacent mobile agent nodes. The localization trajectories of the four algorithms are shown in Figure 4, which the VMP, SPAWN1, SPAWN2 algorithms take the particle number of 250 as an example, respectively and the proposed BP-DCKF algorithm only uses 72 cubature points.

In Figure 4, it is can be seen that trajectories with respect to the VMP, SPAWN1, and SPAWN2 algorithms are significantly deviated from the real trajectories of agent nodes, while trajectories of the BP-DCKF algorithm is close to real trajectories of agent nodes. Meanwhile, it can also find that the deviation of four trajectories between each algorithm and the real trajectory, and the BP-DCKF algorithm is very close to the real trajectory in the four trajectories. For example, location of four agent nodes are localization by the four algorithms at the 89s, which can be shown in the Table 5. In this table, the BP-DCKF algorithm has higher localization accuracy than the VMP, SPAWN1 and SPAWN2 algorithms.

Table 4. Simulation parameters

Parameter name	Symbol	Scene 1	Scene 2
Area of simulation (m²)	S	100×100	100×100
Number of agent node	A_a	4	8
Number of reference node	$A_{a'}$	4	41
Number of total nodes	A	8	49
Location vector of the reference node	$\boldsymbol{x}_{a'}$	$(x_{1,a'}, x_{2,a'})^{\mathrm{T}}$	$(x_{1,a'}, x_{2,a'})^{\mathrm{T}}$
State vector of agent node	$\boldsymbol{x}_a$	$(x_{1,a}, x_{2,a}, \dot{x}_{1,a}, \dot{x}_{2,a})^{\mathrm{T}}$	$(x_{1,a}, x_{2,a}, \dot{x}_{1,a}, \dot{x}_{2,a})^{\mathrm{T}}$
Real location vector of agent node	$\tilde{\boldsymbol{x}}_a$	$(x_{1,a}, x_{2,a})^{\mathrm{T}}$	$(x_{1,a}, x_{2,a})^{\mathrm{T}}$
Simulation time (s)	T	100	100
Simulation times	M_c	100	100
Number of the iterations	N_{iter}	2	2
Measurement radius of the reference node	$r_{M_{a'}}$	100	20
Communication radius of the reference node	$r_{C_{a'}}$	100	20
Measurement radius of the agent node	r_{M_a}	200	50
Communication radius of the agent node	r_{C_a}	200	50
Process noise variance	δ^2	10^{-4}	10^{-4}
Measurement noise variance	σ^2	1	1

Table 5. Localization performance comparison of different algorithms

Agent node		1	2	3	4
Actual coordinates		(74.39, 70.81)	(30.67, 76.18)	(64.46, 32.51)	(29.77, 28.56)
Estimations by already existing algorithms	SPAWN1	(73.81, 71.41)	(30.98, 75.96)	(64.87, 32.35)	(30.78, 27.94)
	SPAWN2	(74.46, 71.09)	(30.82, 75.76)	(64.94, 32.62)	(30.29, 28.05)
	VMP	(74.06, 70.75)	(30.95, 75.98)	(64.26, 32.03)	(29.17, 28.14)
Estimations by proposed algorithm	BP-DCKF	(74.21, 70.78)	(30.69, 76.04)	(64.21, 32.19)	(29.70, 28.06)
Localization error with different algorithms (meter)	SPAWN1	0.8345	0.3801	0.4401	1.1851
	SPAWN2	0.2886	0.4460	0.4924	0.7284
	VMP	0.3354	0.3441	0.5200	0.7324
	BP-DCKF	0.1825	0.1414	0.4061	0.5049

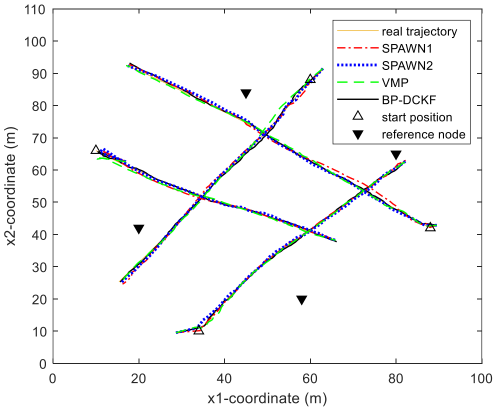


Figure 4. Traaectories of mobile agent nodes

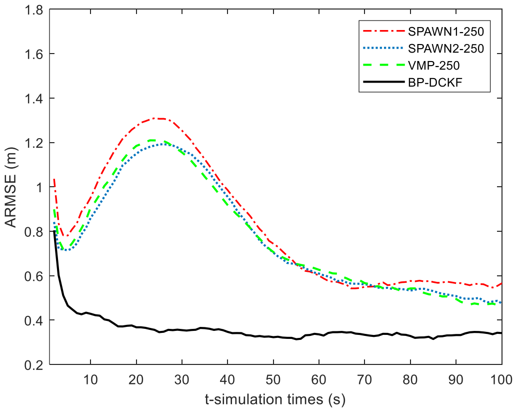


Figure 5. The ARMSE of mobile agent nodes localization

The ARMSE results of the four algorithms are shown in Figure 5. It can be seen that the localization accuracy of the four algorithms listed from high to low are BP-DCKF, SPAWN2, VMP, SPAWN1 algorithm, in addition, ARMSE of SPAWN2, VMP, SPAWN1 algorithm show apparent fluctuations comparing with that of BP-DCKF. Compared with the other three algorithms, the ARMSE of four algorithms and the percentage of the ARMSE decreasing of the BP-DCKF algorithm are shown in Table 6. To demonstrate the localization performance of proposed algorithm, this paper compares its CDF (cumulative distribution function) and ARMSE with those of BP-DCKF, SPAWN1, SPAWN2 and VMP at the 72 cubature points. Figure 6 shows the CDF of the localization errors of the BP-DCKF, SPAWN1, SPAWN2 and VMP algorithms. It is clearly that the BP-DCKF algorithm has the best localization performance.

In Figure 7, the number of particles related to the VMP, SPAWN1 and SPAWN2 algorithms are set to be 250, 500 and 1000, respectively. In the case of different particles, since the localization information of the adjacent cooperating nodes will change with the time slot during the movement of the agent node, resulting in localization accuracy changing. The ARMSE of four algorithms change with time are given. It is that accuracies of four algorithms are rowed from high to low in order: the BP-DCKF, SPAWN1, SPAWN2 and VMP algorithm, moreover, the ARMSE curves of the VMP, SPAWN2 and SPAWN1 algorithm greatly fluctuate when the number of particles is small. Except BP-DCKF algorithm, other three algorithms with 500 particles are severely fluctuation at the 25s, and the ARMSE of the four algorithms the percentage of the ARMSE decreasing of the BP-DCKF algorithm are shown in Table 7. It is clearly that the BP-DCKF algorithm has higher localization accuracy.

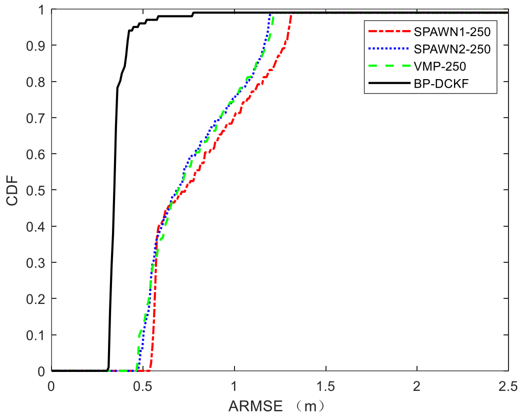


Figure 6. The CDF of mobile agent nodes ARMSE

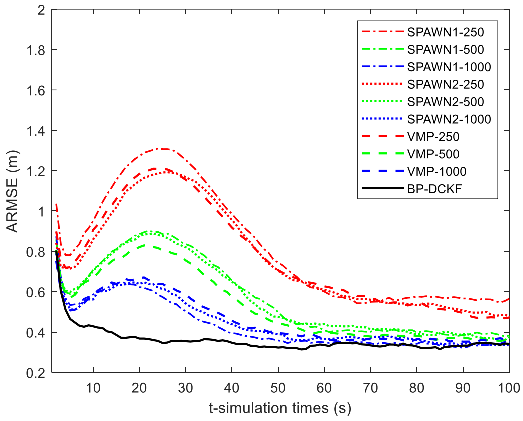


Figure 7. The ARMSE of mobile agent nodes localization

Table 6. ARMSE and the improved performance percentage

Algorithms	SPAWN1	SPAWN2	VMP	BP-DCKF
ARMSE (m)	1.3	1.18	1.2	0.3
Percentage (%)	76.92	74.58	75.00	-

Table 7. ARMSE and the improved performance percentage

Algorithms	SPAWN1	SPAWN	VMP	BP-DCKF
ARMSE (m)	0.8	0.89	0.9	0.3
Percentage (%)	62.50	66.29	66.67	-

Besides, in Figure 7, with the number of particles increasing from 250 to 1000, the ARMSEs of the VMP, SPAWN1 and SPAWN2 algorithm are significantly enhanced, when the number of particles is 1000, the ARMSE of the SPAWN1 and SPAWN2 algorithm are close to the BP-DCKF algorithm, but the BP-DCKF algorithm only uses 72 cubature points. Compared with the three algorithms, the BP-DCKF algorithm effectively improves the low time execution efficiency caused by increasing the number of particles, improves the performance of the three algorithms in the case of fewer particles, so it has higher time execution efficiency.

The average running time of the four algorithms are shown in Table 8, it shows that average running time of the BP-DCKF algorithm is the shortest in the four algorithms. The average running time of the SPAWN1, SPAWN2 and VMP algorithm are increasing rapidly as the number of particles increases, but the BP-DCKF algorithm has not changed because the number of particles is only related to the number of adjacent nodes, the number of particles (72 cubature points) does not change when the adjacent nodes do not change.

From above Figure 7, it is that the number of particles in the SPAWN1, SPAWN2 and VMP algorithm is 1000, respectively, the ARMSE is relatively close to the BP-DCKF algorithm, however, the average running time is 22.4277s, 36.2856s and 1.2754s, respectively. The BP-DCKF algorithm only uses 0.3580s, time consumption decreasing percentage of proposed algorithms over SPAWN1, SPAWN2 and VMP are shown in Table 9. Moreover, the average running time of the SPAWN1 and SPAWN2 algorithm increased an order of magnitude increase, the VMP algorithm is superior to the two SPAWN algorithms in terms of average running time, but it has insufficient accuracy. Therefore, the BP-DCKF algorithm has high localization accuracy and high time execution efficiency in the static link network.

Table 8. Average running time (s)

Particles	SPAWN1	SPAWN2	VMP	BP-DCKF
250	1.8180	2.8303	0.3791	-
500	6.3811	10.6526	0.7030	-
1000	22.4722	36.2856	1.2754	-
72	-	-	-	0.3580

Table 9. Improved performance percentage

Algorithms	SPAWN1	SPAWN2	VMP	BP-DCKF
Percentage (%)	98.40	99.00	71.30	-

5.2 Scene 2: Agent Interactive Moving in the Dynamic Link Network

In this scene, agents move according to straight line and curves interactive multiple model (IMM) in the area, which is deployed dynamic link network. In addition, the connection relationship between all nodes in the dynamic link network may be changed during the time of agent moving, while the mobile agents can only obtain the measurement and belief from the reference nodes and the mobile agent nodes within the measuring range of mobile agents. The localization trajectories of the four algorithms are shown in Figure 8, which the particle numbers of the VMP, SPAWN1 and SPAWN2 algorithms are all set to be 250 and the BP-DCKF algorithm are 8 N_a cubature points that the number of particles is only related to the number N_a of adjacent nodes.

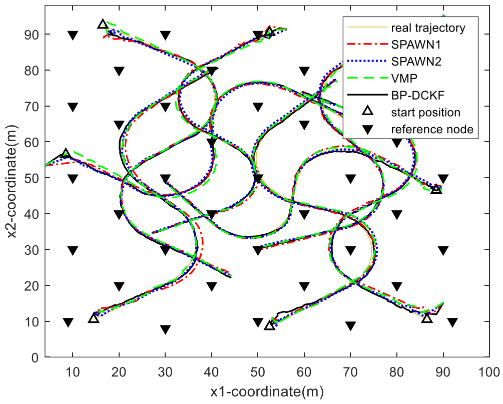


Figure 8. Traaectories of mobile agent nodes

In Figure 8, it can be seen that the real trajectories of agent nodes are changing with time, it is because of the localization information of the adjacent cooperating nodes changing with time slot during the movement of the agent node, the trajectory of the VMP, SPAWN1 and SPAWN2 algorithms significantly deviate from the real trajectory, while trajectories of the BP-DCKF algorithms is close to real trajectories of agents. It can also find that all trajectories do not always deviate greatly. Usually, the VMP, SPAWN1 and SPAWN2 algorithms will deviate significantly from the real trajectory at the starting point. After a period of time, the localization error gradually decreases and the trajectories are close to the real trajectory. The BP-DCKF algorithm is very close to the real trajectory during the entire running. Since the three algorithms all use random sampling methods to realize node localization, although the operation is simple, they are greatly affected by the number of particles, and have a larger localization error when the number of particles is small, but, the BP-DCKF algorithm uses a deterministic sampling method with fewer particles and does not appear to be significantly deviated from the real trajectories. In conclusion, it is obvious that the BP-DCKF algorithm has higher localization accuracy than the VMP, SPAWN1 and SPAWN2 algorithms.

Figure 9 shows that the ARMSE of four algorithms changed with time t. The number of particles related to the VMP, SPAWN1 and SPAWN2 algorithms are set to be 250, 500 and 1000, respectively, the BP-DCKF algorithm use $8N_a$ cubature points. It can be found that the ARMSE of the four algorithms show a decline trend as a whole, which localization accuracies are rowed from high to low in order: the BP-DCKF, SPAWN1, SPAWN2 and VMP algorithm. Moreover, the ARMSE of the VMP, SPAWN2 and SPAWN1 algorithm greatly fluctuate when the number of particles is 250, the three algorithms that the number of 250 particles are largely fluctuation when t is 40s, and the ARMSE and the ARMSE decreasing percent- age of proposed algorithms over SPAWN1, SPAWN2 and VMP are shown in Table 10.

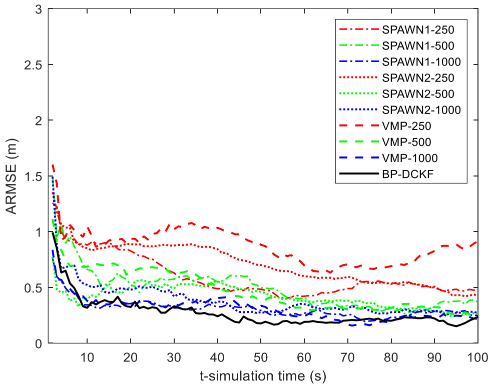


Figure 9. The ARMSE of mobile agent nodes localization

Table 10. ARMSE and improved performance percentage

Algorithms	SPAWN	SPAWN	VMP	BP-DCKF
ARMSE (m)	0.5	0.80	1.00	0.2
Percentage (%)	60.00	75.009	80.00	-

With the number of particles increasing, the localization accuracy of the VMP, SPAWN2 and SPAWN1 algorithms are significantly enhanced, and they close to the BP-DCKF algorithm in the number of particles to be 1000, but the BP-DCKF algorithm uses low cubature points. Compared with the three algorithms, the BP-DCKF algorithm effectively improves the low time execution efficiency caused by the increase in the number of particles, improves the performance of the three algorithms in the case of fewer particles. Thus, the BP-DCKF algorithm has higher time execution efficiency.

The average running time of the four algorithms in the IMM models are shown in Table 11, it shows that the average running time of the SPAWN1, SPAWN2 and VMP algorithms are increasing rapidly as the number of particles increases, however, the BP-DCKF algorithm has not changed because the cubature points is only related to the number of adjacent nodes. From above Figure 9, it shows that the number of particles in the SPAWN2, SPAWN1 and VMP algorithms is 1000, respectively, which the ARMSE is relatively close to the

BP-DCKF algorithm, but the average running time is 265.5315s, 172.2324s and 6.0408s, respectively.

Table 11. Average running time (s)

Particles	SPAWN1	SPAWN2	VMP	BP-DCKF
250	15.1455	20.6177	1.8129	-
500	48.4743	76.8544	3.1337	-
1000	172.2324	265.5315	6.0408	-
8 N_a	-	-	-	2.1459

The BP-DCKF algorithm only uses 2.1459s, time consumption decreasing percentage of proposed algorithms over SPAWN1, SPAWN2 and VMP are shown in Table 12. Moreover, the average running time of the SPAWN1 and SPAWN2 algorithm increased an order of magnitude increase. The VMP algorithm is superior in terms of average running time, while it has insufficient accuracy. Thus, the BP-DCKF algorithm has high localization accuracy and high time execution efficiency in the dynamic link network.

Table 12. Improved performance percentage

Algorithms	SPAWN1	SPAWN2	VMP	BP-DCKF
Percentage (%)	98.70	99.19	60.40	-

Figure 10 shows ARMSE of the BP-DCKF algorithm is changing with time at different measurement noise conditions. It can be seen that the ARMSE of the BP-DCKF algorithm is decreasing with time increasing. Moreover, higher measurement accuracy is lead smaller localization error. Figure 11 shows the CDF of the localization error of the BP-DCKF algorithm at different measurement noise conditions. It can be seen that the localization performance of the BP-DCKF algorithm is positively correlated with the measurement noise.

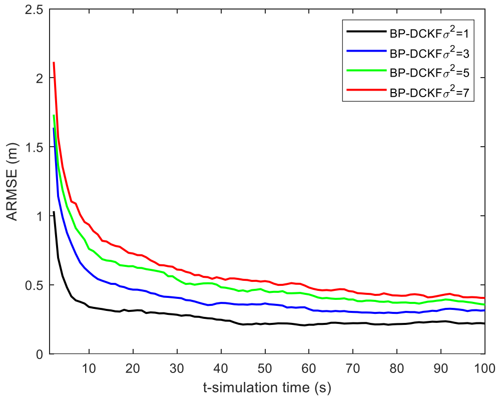


Figure 10. The ARMSE of mobile agent nodes localization

From above simulation results, it is obvious that the BP-DCKF algorithm has higher localization accuracy than that of traditional algorithms in both the static link network model and the dynamic link network model, moreover, it is superior to the VMP and SPAWN algorithm with non-parametric BP method in the aspects including time execution efficiency, communication burden, and so on. So, the BP-DCKF algorithm is tractable in practical application.

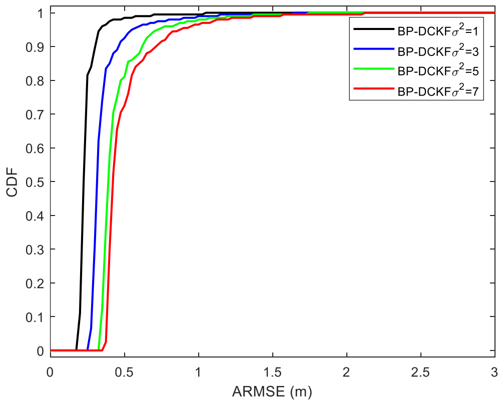


Figure 11. The CDF of mobile agent nodes ARMSE

6 Conclusion

The BP-DCKF algorithm is proposed for node localization of WSN. Using the deterministic sampling of CKF algorithm and parameterized message transmitting of BP strategy, state variable of mobile agent node is modeled as an edge posterior probability estimation problem of multivariable nodes. The posterior distribution of the message for each variable node in the FG by parameterized message transmission, reconstruction strategy and DCKF algorithm. The agent node estimates its own localization by parameter message from other adjacent nodes, and sends its own localization message to adjacent agent nodes. Simulation experiments show that localization accuracy of proposed algorithm is higher about 86% than that of traditional algorithm. Moreover, since only to transmit the mean vector and covariance matrix between the adjacent agent nodes, proposed algorithm has lower about 75% communication burden than that of traditional algorithm. In conclusion, the proposed CL framework has good localization performance.

Acknowledgements

This work was supported by the National Science Foundation Council of China (61976080, 61771006), the Key Research Projects of University in Henan Province, China (21A413002, 20B510001), the Programs for Science and Technology Development of Henan Province, China (212102310298, 222102210002, 222102210004), the Soft Science Research Program of Henan Province, China (202400410097).

References

[1] N. Patwari, J. N. Ash, S. Kyperountas, A. O. Hero, R. L. Moses, N. S. Correal, Locating the Nodes: Cooperative Localization in Wireless Sensor Networks, *IEEE Signal Processing Magazine*, Vol. 22, No. 4, pp. 54-69, July, 2005.

[2] U. A. Khan, S. Kar, J. M. F. Moura, Distributed Sensor Localization in Random Environments Using Minimal Number of Anchor Nodes, *IEEE Transactions on Signal Processing*, Vol. 57, No. 5, pp. 2000- 2016, May, 2009.

[3] M. Z. Win, A. Conti, S. Mazuelas, Y. Shen, W. M. Gifford, D. Dardari, M. Chiani, Network Localization and Navigation Via Cooperation, *IEEE Communications*

Magazine, Vol. 49, No. 5, pp. 56-62, May, 2011.

[4] P. Corke, T. Wark, R. Jurdak, W. Hu, P. Valencia, D. Moore, Environmental Wireless Sensor Networks, *Proceedings of the IEEE*, Vol. 98, No. 11, pp. 1903-1917, November, 2010.

[5] H. Chen, F. Gao, M. Martins, P. Huang, J. Liang, Accurate and Efficient Node Localization for Mobile Sensor Networks, *Mobile Networks and Applications*, Vol. 18, No. 1, pp. 141-147, February, 2013.

[6] S. R. Jondhale, R. S. Deshpande, Efficient Localization of Target in Large Scale Farmland Using Generalized Regression Neural Network, *International Journal of Communication Systems*, Vol. 32, No. 16, pp. 1-19, November, 2019.

[7] L. L. Hung, Intelligent Sensing for Internet of Things Systems, *Journal of Internet Technology*, Vol. 23, No. 1, pp. 185-191, January, 2022.

[8] B. Hammi, R. Khatoun, S. Zeadally, A. Fayad, L. Khoukhi, IoT technologies for smart cities, *IET Networks*, Vol. 7, No. 1, pp. 1-13, January, 2018.

[9] H. Wymeersch, J. Lien, M. Z. Win, Cooperative Localization in Wireless Networks, *Proceedings of the IEEE*, Vol. 97, No. 2, pp. 427-450, February, 2009.

[10] A. T. Ihler, J. W. Fisher, R. L. Moses, 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, April, 2005.

[11] J. Lien, U. J. Ferner, W. Srichavengsup, H. Wymeersch, M. Z. Win, A Comparison of Parametric and Sample-based message Representation in Cooperative Localization, *International Journal of Navigation and Observation*, Vol. 2012, pp. 1-10, September, 2012.

[12] V. Savic, S. Zazo, Cooperative Localization in Mobile Networks Using Nonparametric Variants of Belief Propagation, *Ad Hoc Networks*, Vol. 11, No. 1, pp. 138-150, January, 2013.

[13] H. Wymeersch, F. Penna, V. Savic, Uniformly Reweighted Belief Propagation for Estimation and Detection in Wireless Networks, *IEEE Transactions on Wireless Communications*, Vol. 11, No. 4, pp. 1587-1595, April, 2012.

[14] C. Pedersen, T. Pedersen, B. H. Fleury, A Variational Message Passing Algorithm for Sensor Self-localization in Wireless Networks, *2011 IEEE International Symposium on Information Theory Proceedings*, St. Petersburg, Russia, 2011, pp. 2158-2162.

[15] J. Xiong, Z. Xiong, J. W. Cheong, J. Xu, Y. Yu, A. G. Dempster, Cooperative Positioning for Low-Cost Close Formation Flight Based on Relative Estimation and Belief Propagation, *Aerospace Science and Technology*, Vol. 106, pp. 1-14, November, 2020.

[16] F. Meyer, O. Hlinka, F. Hlawatsch, Sigma Point Belief Propagation, *IEEE Signal Processing Letters*, Vol. 21, No. 2, pp. 145-149, February, 2014.

[17] I. Arasaratnam, S. Haykin, Cubature Kalman Filters, *IEEE Transactions on Automatic Control*, Vol. 54, No. 6, pp. 1254-1269, June, 2009.

[18] Q. Guo, Y. Zhang, J. Lloret, B. Kantarci, W. K. G. Seah, A Localization Method Avoiding Flip Ambiguities for Micro-UAVs with Bounded Distance Measurement Errors, *IEEE Transactions on Mobile Computing*, Vol. 18, No. 8, pp. 1718-1730, August, 2019.

[19] F. Meyer, F. Hlawatsch, H. Wymeersch, Cooperative Simultaneous Localization and Tracking (coslat) with Reduced Complexity and Communication, *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, Vancouver, BC, Canada, 2013, pp. 4484-4488.

[20] Y. Liu, B. Lian, T. Zhou, Gaussian Message Passing-based Cooperative Localization with Node Selection Scheme in Wireless Networks, *Signal Processing*, Vol. 156, pp. 166-176, March, 2019.

Biographies

Lin Zhou received the M.S. in application mathematics from Henan University, China, in 2005, and the Ph.D degree in control theory and control engineering from Northwestern Polytechnical University China, in 2013. Now, she is an Associate Professor of school of artificial intelligence, Henan University. Her research interests include information fusion.

Lu Zhang is currently working toward the M.S. degree in Henan University. Now, he is a student member of the lab of intelligent technology and systems of Henan University and his research interests include wireless sensor networks, localization and target tracking.

Yong Jin received Ph.D. degrees in Information and Communication Engineering from Northwestern Polytechnical University China, in 2010. Now, he is a Professor of school of artificial intelligence, Henan University. Also, he has served as a peer-reviewer for various IEEE research aournals since 2010. His research interests include signal processing.

Zhentao Hu received the M.S. in application mathematics from Henan University China, in 2006, and a Ph.D degree in control science and engineering from Northwestern Polytechnical University China, in 2010. Now, he is a Professor of school of artificial intelligence, Henan University. His research interests include target tracking, nonlinear estimation.

Junwei Li received the M.S. and Ph.D degrees in control science and engineering from Northwestern Polytechnical University China, in 2009 and 2013 respectively. Now, he is an Associate Professor of school of artificial intelligence, Henan University. His current research interests include information fusion, pattern recognition.