A Parallel Integration Method of Cooperative Target-Localization and Cooperative Self-localization

*WANG Leigang**, *KONG Depei*, *ZHOU Jihang*, *WANG Jianlu*

State Key Laboratory of Complex Electromagnetic Environment Effects on Electronics and Information System, Luoyang 471003, P. R. China

(Received 8 September 2021; revised 10 March 2022; accepted 20 March 2022)

Abstract: When a group of mobile agents track a target, they can locate themselves and the target in a cooperative manner. To maximize the group advantage, a parallel integration strategy of cooperative target-localization (CTL) and cooperative self-localization (CSL) is designed. Firstly, a global cost function containing the agents' positions and the target's position is established. Secondly, along with the agents' positions being re-estimated during CTL, the U-transform is employed to propagate the error covariance of the position estimations among the agents. The simulation results show that, the proposal exploits more information for locating the target and the agents than the cases where CTL and CSL run separately, and the global optimal position estimations of the agents and the target are obtained.

Key words: cooperative self-localization; cooperative target-localization; non-classical multi-dimensional scaling; majoring function

CLC number: TN925 **Document code:** A **Article ID:** 1005-1120(2022)02-0231-08

0 Introduction

Multi-agent (such as unmanned aerial vehicle, robot) system has great potential on target monitoring, disaster rescuing and so on. Multi-agent can cooperate for different goals, such as cooperative self-localization (CSL)[1-4], cooperative target-localization (CTL), cooperative searching, and cooperative communication. In this paper, CSL and CTL are highlighted.

CSL and CTL have different goals. Usually, CSL and CTL are researched as two topics. CSL aims at improving the localization accuracy of each agent by utilizing the relative information among the agents. Now, the CSL based on Kalman filtering (KF-CSL) or its extension has been heavily researched[5-9]. However, CTL aims at improving the target localization accuracy by fusing the relative information between the agents and the target.

Traditionally, CSL and CTL run separately. CSL can provide each agent's position estimation for CTL. When these estimations are used for CTL, they are usually assumed to be deterministic and independent with each other. Based on this assumption, some algorithms have been proposed for CTL, such as least squares, D-S evidence theory, and neural networks[10]. In Ref.[11], the Bayesian estimation method was proposed to deal with multi-source uncertainty data. In Ref.[12], a data aggregation algorithm based on the convolutional neural network model was proposed. In Ref.[13], an improved data fusion based on the D-S evidence theory was proposed.

However, in the KF-CSL, the position estimations of the agents are mutually referred. As a result, the position estimations are non-deterministic characterized in variance and relevant characterized in covariance[5-8]. When these estimations are further used for CTL, the non-deterministic and relevant characteristics must be considered.

*Corresponding author, E-mail address: wanglg12@tsinghua.org.cn.

How to cite this article: WANG Leigang, KONG Depei, ZHOU Jihang, et al. A parallel integration method of cooperative target-localization and cooperative self-localization[J]. Transactions of Nanjing University of Aeronautics and Astronautics, 2022, 39(2): 231-238.

http://dx.doi.org/10.16356/j.1005-1120.2022.02.009

In this paper, CTL and CSL are together regarded as a global non-classical multi-dimensional scaling (nMDS) problem, that is, at one moment, the agents, the target and the distances (contain noise) among them constitute a snap-shot. For the nMDS, a distributed nMDS algorithm was proposed in Ref.[14], where the majoring function (MF) is adopted for estimating the static nodes. In Ref.[15], the MF-based optimization algorithm was used for mobile multi-agents.

In the typical studies, the followings are assumed: (1) The agents' position estimations are independent each other. (2) CSL and CTL are respectively solved. This paper is characterized as follows:

(1) A global cost function including the position variables of the agents and the target is established, where the position uncertainty of each agent and the position correlation among them are introduced.

(2) During the optimization iteration of the established cost function, the U-transform is employed to propagate the error covariance of the position estimations among the agents.

1 Problem Statement

The symbols used throughout this paper are defined in Table 1.

Table 1 Symbols used in text and derivations

Notation	Description
N	The total number of the agents
n	The number of agents which find the target
C	The set of agents which find the target
$\boldsymbol{x}_i$	Actual state vector of agent i
$\hat{\boldsymbol{x}}_i$	Posteriori state vector of agent i
$\boldsymbol{x}$	Whole state vector, $\boldsymbol{x}=\left[x_1,\cdots,x_N\right]$
$\hat{\boldsymbol{P}}$	Whole error covariance of the state estimates
$\hat{\boldsymbol{P}}_{ij}$	Error covariance of the state estimates between agent i and agent j
$\boldsymbol{x}_o$	Actual state vector of the target o
$\boldsymbol{z}_{io}$	Relative measurement between the agent i and the target o
σ_{io}^2	Noise variance of the measurement $\boldsymbol{z}_{io}$

Assume that N agents move in a 2-D area to track a target. A fixed reference frame is set, where the actual position of the agent i at the time step k is denoted as $\boldsymbol{x}_i(k)=[x_i(k), y_i(k)]$. The outputs of the KF-CSL, i. e., the position estimations of the agents, are expressed as $\{\hat{\boldsymbol{x}}_i(k)\}_{i=1}^N$, and the corresponding error covariance is expressed as

$$\hat{\boldsymbol{P}}(k)=\begin{bmatrix}\hat{\boldsymbol{P}}_{11}(k) & \cdots & \hat{\boldsymbol{P}}_{1N}(k)\\ \vdots & \ddots & \vdots\\ \hat{\boldsymbol{P}}_{N1}(k) & \cdots & \hat{\boldsymbol{P}}_{NN}(k)\end{bmatrix}\in\mathrm{R}^{2N\times 2N}$$

In Fig.1, the agents (numbered in $1,2,\cdots,N$) track the target "o". The dashed circles denote the position estimations of the agents through CSL. The solid circles denote the true positions of the agents.

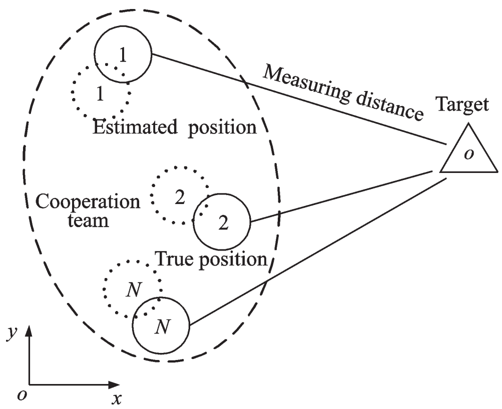

Fig.1 Application demonstration of CSL and CTL

Whether one agent can find the target is changing for its varying position or view. At the time k, assume that $n(n\leqslant N)$ agents (constitute a collection C) find the target "o" and obtain the relative measurements to the target. The measurements are denoted as $\boldsymbol{z}_{io}(k)(i\in C)$. Assume that the measurements are independent each other and $\boldsymbol{z}_{io}\sim N(0,\sigma_{io}^2)$. Then, utilizing $\{\hat{\boldsymbol{x}}_i(k)\}_{i=1}^N$, $\hat{\boldsymbol{P}}(k)$, and $\boldsymbol{z}_{io}(k)$, we attempt to re-estimate the agents and the target. In this process, the followings need to be solved.

(1) Under the case of the KF-CSL, the factors that affect CTL are the performance of the exteroceptive sensor which measures the relative measurement, and the agents' positions that are non-deterministic and relevant.

(2) Since the agents' positions estimated by the KF-CSL are non- deterministic, they are re-estimated during CTL. Then the re-estimated results

should include the uncertainty and the correlation to support the periodic CSL.

2 CTL Based on nMDS

2.1 KF-CSL and CTL

The KF-CSL is a premise of this paper. The KF-CSL is employed for its excellent recursion mechanism including the predictive and posteriori update[16-17]. Using the agent's linear velocity measurements and the relative measurements, etc., the position estimations and the corresponding error covariance are continually updated. Provide that the output form of the KF-CSL is unified as the posteriori estimations. The input/output of the KF-CSL and CTL is listed in Table 2. It can be found that the uncertainty and correlation are essential for CSL. In the traditional serial mode where the position estimations of the agents from CSL are constant input for CTL, the agents' positions are estimated in CSL and unchanged during CTL.

Table 2 Comparison of input/output between KF-CSL and CTL

	Input		Output
KF-CSL	Predictive update	The velocity of each agent (linear/ rotational velocity), the posteriori estimates of agents and their covariance.	Posteriori estimates of the agents and their covariance.
	Posteriori update	The relative measurements among the agents, the predictive position estimates of the agents and their covariance.	
CTL	The positions of agents, the relative measurement between the agent and the target (distance, azimuth).		Estimated position of target, position estimate of agents and its covariance(necessary in the case of the parallel integration mode).

In this paper, the operational logic of CSL and CTL is converted from the serial mode to the integration one. The following two facts are considered. Firstly, all the agents are cooperative while the target is not. Hence, the frequency of the data supplying for CSL is higher than that for CSL. Secondly, the re-estimated positions of the agents in CTL need to support CSL smoothly. The relationship between CSL and CTL are demonstrated in Fig.2.

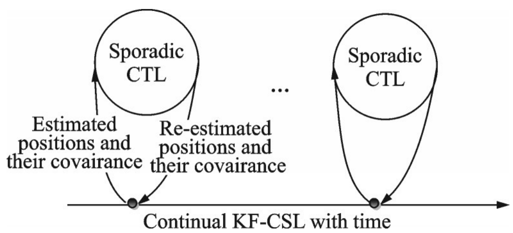

Fig.2 Operation relationship between CSL and CTL

2.2 Object function modeling

When multiple agents find the same static target at the same time, the agents, the target and the relative measurements constitute a snapshot, where the agents are the anchors with a priori knowledge (from the KF-CSL), the target is unknown, and the relative measurements among them are constraint[18]. According to the nMDS, a global loss function with respect to the position variables of the agents and the target, is established as①

$$S = \| \boldsymbol{x} - \hat{\boldsymbol{x}} \|^2_{\hat{P}} + \sum_{m \in C} \| z(\boldsymbol{x}_m, \boldsymbol{x}_o) - z_{mo} \|^2_{\sigma_{mo}} \qquad (1)$$

where $\boldsymbol{x} = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N]$ represents the position vector of all agents②, $z(\boldsymbol{x}_m, \boldsymbol{x}_o) = \sqrt{(\boldsymbol{x}_m - \boldsymbol{x}_o)^2 + (\boldsymbol{y}_m - \boldsymbol{y}_o)^2}$ the distance function between the agent m and the target o, and $\| \cdot \|$ the norm operation. Eq.(1) can be further rewritten as

$$S = (\boldsymbol{x} - \hat{\boldsymbol{x}})^{\mathrm{T}} \begin{bmatrix} \hat{\boldsymbol{P}}_{11} & \cdots & \hat{\boldsymbol{P}}_{1N} \\ \vdots & \ddots & \vdots \\ \hat{\boldsymbol{P}}_{N1} & \cdots & \hat{\boldsymbol{P}}_{NN} \end{bmatrix}^{-1} (\boldsymbol{x} - \hat{\boldsymbol{x}}) + \sum_{m \in C} \left(\frac{z(\boldsymbol{x}_m, \boldsymbol{x}_o) - z_{mo}}{|\sigma_{mo}|} \right)^2 \qquad (2)$$

The correlation and uncertainty of the elements

① Thereafter, the time index k is omitted to simplify the notation.

② For the correlation of all the agents, the states of agent which does not find the target is also changed in the deep integration mode. So the states of all agents are included.

$\hat{\boldsymbol{x}}_i(i=1,\cdots,N)$ are introduced to the loss function S by the block element $\hat{\boldsymbol{P}}_{ij}(i\neq j)$ and $\hat{\boldsymbol{P}}_{ii}$.

Let $(\hat{\boldsymbol{P}})^{-1}=\boldsymbol{A}$. Each block element $\boldsymbol{A}_{ij}\in\mathbf{R}^{2\times 2}$ represents the uncertainty of the position estimation($i=j$) and the correlation among the estimations($i\neq j$). Then Eq.(1) can be further rewritten as

$$S=(\boldsymbol{x}_i-\hat{\boldsymbol{x}}_i)^{\mathrm{T}}\boldsymbol{A}_{ii}(\boldsymbol{x}_i-\hat{\boldsymbol{x}}_i)+\sum_{i=1}^{N}\sum_{j=1,j\neq i}^{N}(\boldsymbol{x}_i-\hat{\boldsymbol{x}}_i)^{\mathrm{T}}\mathbf{A}_{ij}(\boldsymbol{x}_j-\hat{\boldsymbol{x}}_j)+\sum_{m\in C}\left(\frac{z(\boldsymbol{x}_m,\boldsymbol{x}_o)-\boldsymbol{z}_{mo}}{|\sigma_{mo}|}\right)^2 \tag{3}$$

Usually, the optimization method is employed for Eq.(3). During optimizing, the agents' position estimations are iterated, and the corresponding error covariance is changing synchronously. To propagate the covariance, the majorizing function(MF) is employed, which can provide an analytic representation between two-step iterations.

A majorizing function $T(X,Y)$ of $S(X)$ is a function that satisfies: (1) $\forall Y, T(X,Y)\geqslant S(X)$, (2) $T(X,X)=S(X)$.

In the MF, the iteration is described as follows: let $Y=X^0$ and substitute it to $T(X,Y)$ as the initial value. The iteration is repeated until the convergence condition is satisfied, i.e.

$$S^{s+1}-S^s\leqslant\varepsilon \tag{4}$$

The upper-notation$s$is the mark of iteration step. Following Refs.[14-15], Eq.(3) is rewritten as

$$S=\eta_z^2+\eta^2-2\rho(X) \tag{5}$$

where

$$\begin{cases}\eta_z^2=\sum_{m\in C}\left(\dfrac{\boldsymbol{z}_{mo}}{|\sigma_{mo}|}\right)^2\\ \eta^2=\sum_{i=1}^{N}\sum_{j=1}^{N}(\boldsymbol{x}_i-\hat{\boldsymbol{x}}_i)^{\mathrm{T}}\boldsymbol{A}_{ij}(\boldsymbol{x}_j-\hat{\boldsymbol{x}}_j)+\sum_{m\in C}\left(\dfrac{z(\boldsymbol{x}_m,\boldsymbol{x}_o)}{|\sigma_{mo}|}\right)^2\\ \rho(X)=\sum_{m\in C}\dfrac{z(\boldsymbol{x}_m,\boldsymbol{x}_o)\cdot\boldsymbol{z}_{mo}}{|\sigma_{mo}|^2}\end{cases} \tag{6}$$

Define the MF of $S(X)$ as $T(X,Y)$

$$T(X,Y)=\eta_z^2+\eta^2-2\rho(X,Y) \tag{7}$$

where $\rho(X,Y)=\sum_{m\in C}\dfrac{\boldsymbol{z}_{mo}}{|\sigma_{mo}|^2}\cdot\dfrac{(\boldsymbol{x}_m-\boldsymbol{x}_o)^{\mathrm{T}}(\boldsymbol{y}_m-\boldsymbol{y}_o)}{z(\boldsymbol{y}_m,\boldsymbol{y}_o)}$.

Through the MF, minimizing $S(X)$ is now a task of finding the minimum of $T(X,Y)$.

2.3 Optimizing derivation

The optimal solution can be obtained through taking the derivative of $T(X,Y)$, shown as

$$\begin{cases}\dfrac{\partial T(X,Y)}{\partial\boldsymbol{x}_i}=2(\boldsymbol{A}_{ii}\boldsymbol{x}_i-\boldsymbol{A}_{ii}\hat{\boldsymbol{x}}_i)+2\sum_{j=1,j\neq i}^{N}\boldsymbol{A}_{ij}(\boldsymbol{x}_j-\hat{\boldsymbol{x}}_j)+\dfrac{2(\boldsymbol{x}_i-\boldsymbol{x}_o)}{\sigma_{io}^2}-\dfrac{2\cdot\boldsymbol{z}_{io}}{\sigma_{io}^2\cdot z(\boldsymbol{y}_i,\boldsymbol{y}_o)}(\boldsymbol{y}_i-\boldsymbol{y}_o)=0 & i\in C\\ \dfrac{\partial T(X,Y)}{\partial\boldsymbol{x}_i}=2\boldsymbol{A}_{ii}\boldsymbol{x}_i+2\sum_{j=1,j\neq i}^{N}\boldsymbol{A}_{ij}(\boldsymbol{x}_j-\hat{\boldsymbol{x}}_j)=0 & i\notin C\\ \dfrac{\partial T(X,Y)}{\partial\boldsymbol{x}_o}=-\sum_{m\in C}\dfrac{2(\boldsymbol{x}_m-\boldsymbol{x}_o)}{\sigma_{io}^2}+2\sum_{m\in C}\dfrac{\boldsymbol{z}_{mo}}{\sigma_{io}^2}\cdot\dfrac{(\boldsymbol{y}_m-\boldsymbol{y}_o)}{z(\boldsymbol{y}_m,\boldsymbol{y}_o)}=0\end{cases} \tag{8}$$

According to the iteration principle, let $\boldsymbol{x}_j=\boldsymbol{x}_j^s$, $\boldsymbol{x}_o=\boldsymbol{x}_o^s$, $\boldsymbol{y}_i=\boldsymbol{x}_i^s$, $\boldsymbol{y}_o=\boldsymbol{x}_o^s$, $z(\boldsymbol{y}_i,\boldsymbol{y}_o)=\boldsymbol{z}_{io}^n$. Then the iteration models are given as

$$\boldsymbol{x}_i^{s+1}=\boldsymbol{a}_i\cdot\left[\sum_{j=1}^{N}\boldsymbol{A}_{ij}\hat{\boldsymbol{x}}_j+\boldsymbol{b}_i^s\cdot\boldsymbol{X}^s\right] \tag{9}$$

$$\boldsymbol{x}_o^{s+1}=\boldsymbol{a}_o\cdot\boldsymbol{b}_o^s\cdot\boldsymbol{X}^s \tag{10}$$

where $\boldsymbol{b}_i^s=[\boldsymbol{b}_{i1}^s,\boldsymbol{b}_{i2}^s,\cdots,\boldsymbol{b}_{iN}^s,\boldsymbol{b}_{io}^s]\in\mathbf{R}^{2\times 2(N+1)}$ and $\boldsymbol{X}^s=[\boldsymbol{x}_1^s,\boldsymbol{x}_2^s,\cdots,\boldsymbol{x}_N^s,\boldsymbol{x}_o^s]\in\mathbf{R}^{2(N+1)\times 1}$. The iteration coefficients are given according to two cases.

(1) If $i\in C$, the iteration coefficients are given as

$$\begin{cases}\boldsymbol{a}_i=(\boldsymbol{A}_{ii}+I_2\sigma_{io}^{-2})^{-1}\\ \boldsymbol{b}_{ij}^s=-\boldsymbol{A}_{ij} & j\neq i,j\neq o\\ \boldsymbol{b}_{ij}^s=\dfrac{\boldsymbol{z}_{io}}{\sigma_{io}^2\cdot\boldsymbol{z}_{io}^s}I_2 & j=i\\ \boldsymbol{b}_{io}^s=\dfrac{1}{\sigma_{io}^2}\left(1-\dfrac{\boldsymbol{z}_{io}}{\boldsymbol{z}_{io}^s}\right)I_2\end{cases} \tag{11}$$

(2) If $i\notin C$

$$\begin{cases}\boldsymbol{a}_i=\boldsymbol{A}_{ii}^{-1}\\ \boldsymbol{b}_{ij}^s=-\boldsymbol{A}_{ij} & j\neq i,j\neq o\\ \boldsymbol{b}_{ij}^s=0 & j=i\text{ or }j=o\end{cases} \tag{12}$$

For the target o

$$\begin{cases} a_o = \frac{1}{n} \\ b_{oi}^s = 0 & i \notin C \\ b_{oi}^s = \left(1 - \frac{z_{io}}{z_{io}^s}\right) I_2 & i \in C \\ b_{oo}^s = \left(\sum_{i \in C} \frac{z_{io}}{z_{io}^s}\right) I_2 \end{cases} \tag{13}$$

In the case where $C = \phi$, it means that no target is found and CTL doesn't run, then Eqs.(11, 13) are not required. At this moment, the proposed algorithm is degraded into a common CSL.

2.4 Covariance propagation

When the position estimate x_i^s is iterated to x_i^{s+1}, its covariance also changes from $\hat{P}^s$ to $\hat{P}^{s+1}$. In this paper, the U transformation is utilized to propagate the position estimations and their covariance. The process is given as:

(1) Calculate $(4N+1)$ σ points

$$\begin{cases} \hat{\xi}_k^{(0)} = x^s \\ \hat{\xi}_k^{(p)} = x^s + (\sqrt{(2N+\lambda)\hat{P}^s})_p \quad p = 1, 2, \cdots, 2N \\ \hat{\xi}_k^{(p)} = x^s - (\sqrt{(2N+\lambda)\hat{P}^s})_{p-2N} \\ \qquad p = 2N+1, 2N+2, \cdots, 4N \end{cases} \tag{14}$$

where λ is constant, and $(\sqrt{(2N+\lambda)\hat{P}^s})_p$ the p'th column of the matrix $\sqrt{(2N+\lambda)\hat{P}^s}$. The detail can be seen in Ref.[19].

(2) Propagate the σ points as

$$\hat{\xi}_{k+1}^{(p)}(i) = f_i(\hat{\xi}_k^{(p)}) \qquad p = 0, 1, \cdots, 2n \tag{15}$$

$$x^{s+1} = \sum_{p=0}^{2n} \omega_p^m \bar{\xi}_{k+1}^{(p)} \tag{16}$$

$$\hat{P}^{s+1} = \sum_{p=0}^{2n} \omega_p^c (\hat{\xi}_{k+1}^{(p)} - x^{k+1})(\hat{\xi}_{k+1}^{(p)} - x^{k+1})^{\mathrm{T}} \tag{17}$$

where $\omega_p^c = \omega_p^m = 0.5/(2N+\lambda)$.

2.5 Performance analysis

The fisher information matrix (FIM) and its determinant are adopted to evaluate the information value. FIM is defined as

$$F = \left[\frac{\partial z}{\partial X}\right]^{\mathrm{T}} R_z^{-1} \left[\frac{\partial z}{\partial X}\right] \tag{18}$$

where the Jacobian matrix of the relative measurement with respect to the whole state $X = [x_1, \cdots, x_N, x_o]$ is

$$j = \begin{bmatrix} \frac{\partial z_{1o}}{\partial x_1} & 0 & \cdots & \frac{\partial z_{1o}}{\partial x_o} \\ \vdots & & & \vdots \\ & & \ddots & \\ \frac{\partial z_{no}}{\partial x_1} & & \cdots & \frac{\partial z_{no}}{\partial x_o} \end{bmatrix} = [B_1 \quad B_2] \tag{19}$$

When $x_1, \cdots, x_N$ are constant

$$J_1 = [0, B_2] \ \det(F_a) = \det(B_2 B_2^{\mathrm{T}}) \tag{20}$$

When $x_1, \cdots, x_N$ are variable

$$\det(F_b) = \det(B_1 B_1^{\mathrm{T}} + B_2 B_2^{\mathrm{T}}) \tag{21}$$

For $B_1 B_1^{\mathrm{T}}$ and $B_2 B_2^{\mathrm{T}}$ are the symmetrical matrix, then

$$\det(F_b) > \det(F_a) \tag{22}$$

It means that by the proposal, more information is exploited from the relative measurements for estimating the state X.

3 Simulation and Discussion

The simulation parameters are listed in Table 3. Additionally, when the distance between the agent and the target is less than 400 m, the agent finds the target in a probability of 0.8. When the distance between the agent and the target is less than 200 m, the simulation is stopped.

Table 3 Simulation setting

Item	Value
Simulation time/s	300
Sample step/s	0.5
Linear velocity/(m•s^{-1})	1
Noise of linear velocity measuring	$N(0, 0.25)$
Rotational velocity/(rad•s^{-1})	0
Noise of rotational velocity measuring	$N(0, 0.0025)$
Measurement noise of relative distance among agent	$N(0, 25)$
Measurement noise of relative distance between agent and target	$N(0, 100)$

Three aspects are verified by the simulation. (1) Whether the proposal is effective. (2) Whether the proposal has contributed to the self-localization. (3) Whether the proposal has contributed to the target-localization.

3.1 Validating for effectiveness

In order to verify the effectiveness of the proposal, the trajectories generated by different algorithms are contrasted in Fig.3. Assume that four agents move to the same target. The initial positions of four agents are known and marked as $A1$, $A2$, $A3$ and $A4$. The black line is the true trajectory. The red line is the trajectory from the independent localization (IL) where no relative measurements are used. The green line is the trajectory from the proposed CSL-CTL. It can be seen that the estimated trajectories based on the CSL-CTL are closer to the real trajectories than that of IL. The effectiveness of the proposed algorithm is intuitively demonstrated in Fig.3.

In Fig.4, the boundaries determine the 3σ confidence region of the distance error. They are calculated according to the variance of x, y directions. The red boundaries are from IL. The green boundaries are from CSL-CTL. The blue line represents the Euclidean distance error in the case of the CSL-CTL. In the proposal, for the target acting as the node, the more constraint information is added. As a result, the 3σ region determined by the proposed CSL-CTL is smaller than that of IL, which means that the localization uncertainty is reduced. Additionally, it can be found that the distance errors are always within the 3σ region. Thus, the proposed algorithm is effective.

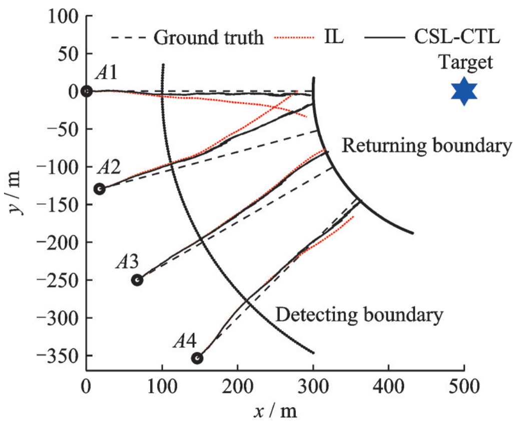

Fig.3 Estimated trajectories of four agents under IL and CSL-CTL

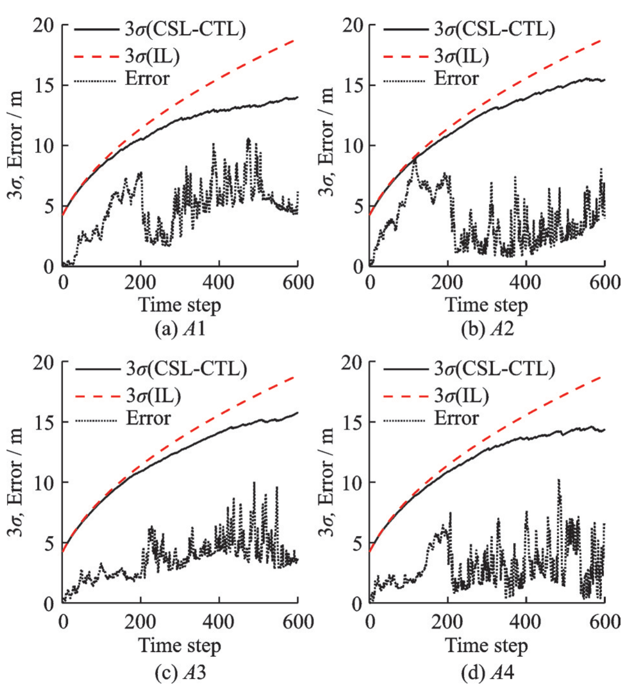

Fig.4 3σ boundaries and error curves of four agents' position estimates

3. 2 Validating for self-localization

With the parallel integration or not, the average accuracy of the self-localization is contrasted in Fig.5, where the curves represent the average distance error between the estimated position

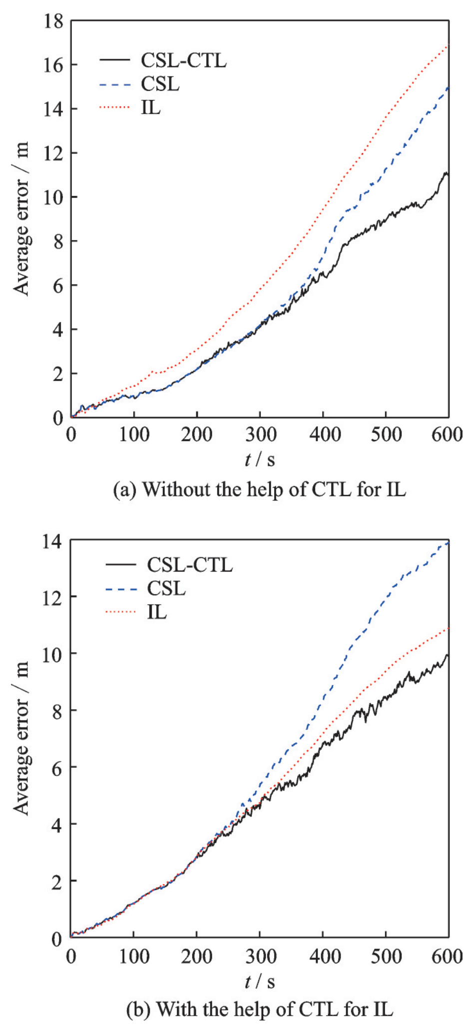

Fig.5 Comparison of average estimate accuracy of four agents under CSL and IL

and the true position, i. e. Average error= $\sqrt{(\hat{x}_i^{est}(k)-x_i(k))^2+(\hat{x}_i^{est}(k)-y_i(k))^2}/4$. The following conclusions are obtained.

(1) In the case of CSL-CTL, the self-localization accuracy is improved, as the green line in Fig. 5. It can be explained as when multiple agents obtain the relative measurements to the same target, these measurements as the new constraints are added. Consequently, the error of the position estimation is reduced.

(2) Without the parallel integration, the localization accuracy of IL (red line) is inferior to CSL (blue line) as shown in Fig.5(a). However, as Fig.5(b), by integrating with CTL, the localization accuracy from IL is higher than that from CSL.

3. 3 Validating for target-localization

With the parallel integration or not, five sets of the target estimations are given in Table 4. Each item is obtained under different noise setting. From Table 4, it can be found that under the parallel integration mode, the target-localization accuracy is improved.

Table 4 Comparison of target estimate under IL and CSL

Actual target position	Target estimate		Error	
	IL-CTL	CSL-CTL	IL-CTL	CSL-CTL
(500,0)	(490.765 2, −4.742 7)	(500.430 8, 1.123 2)	10.381 4	1.203 0
	(476.983 1, 8.596 3)	(497.566 9, −1.008 2)	24.569 8	2.633 7
	(497.972 4, −0.170 7)	(499.166 0, −0.200 7)	2.034 8	0.857 8
	(477.996 3, −12.824)	(497.459 7, −0.855 6)	25.468 3	2.680 5
	(491.375 3, 1.763 8)	(500.221 3, 1.211 5)	8.803 2	1.231 6

4 Conclusions

CSL and CTL are two tasks for multi-agent system. In this paper, two tasks are integrated in parallel to further improve their accuracy. The following problems are solved: (1) Use the position estimations generated from the KF-CSL for CTL. (2) During CTL, the agents regarded as the variables are re-estimated. Then ensure the re-estimated results support the consecutive KF-CSL.

The CSL-CTL algorithm is proposed. Essentially, the proposed algorithm utilizes the data which are respectively prepared for CSL and CTL at the same time. The proposed algorithm can work properly even if one type of data is absent. The algorithm is robustness.

The effectiveness is verified by the simulation. The results show that the accuracy of two tasks in the parallel integration mode is higher than the case where CSL and CTL are executed one by one. It should be noted that the study is based on the case of one static target. In the future, the research can be extended to the case of the dynamic target or multiple targets.

References

[1] MAZA I, CABALLERO F, CAPITAN J, et al. Firemen monitoring with multiple UAVs for search and rescue missions[C]//Proceedings of IEEE International Workshop on Safety Security and Rescue Robotics. Bremen: IEEE, 2010.

[2] SANDERSON A C. A distributed algorithm for cooperative navigation among multiple mobile robots[J]. Advanced Robotics, 1997,12(4): 335-349.

[3] NICOSIA J. Decentralized cooperative navigation for spacecraft[C]//Proceedings of IEEE Aerospace Conference. Big Sky, MT:IEEE, 2007.

[4] SHARMA R. Bearing-only cooperative-localization and path-planning of ground and aerial robots[D]. Provo, USA: Brigham Young University, 2011.

[5] HUANG Yulong, ZHANG Yonggang, XU Bo, et al. A new outlier-robust student's based Gaussian approximate filter for cooperative localization[J]. IEEE/ASME Transactions on Mechatronics, 2017, 22(5): 2308-2386.

[6] HUANG Yulong, ZHANG Yonggang, XU Bo, et al. A new adaptive extended Kalman filter for cooperative localization[J]. IEEE Transactions on Aerospace and Electronic Systems, 2017, 26(9): 353-368.

[7] WANG Leigang, ZHANG Tao. Distributed cooperative localization algorithm for sparse communication network with multi-locating message[J]. Journal of

Systems Engineering and Electronics, 2016, 27(4): 746-753.

[8] ROUMELIOTIS S I, BEKEY G A. Distributed multirobot localization[J]. IEEE Transactions on Robotics and Automation, 2002, 18(5): 781-795.

[9] XU Bo, LI Shengxin, WANG Lianzhao, et al. A multi-AUV cooperative localization method based on adaptive neuro-fuzzy inference system[J]. Journal of Chinese Inertial Technology, 2019,27(4): 440-447.

[10] SARVESH R, URABHI R. Multi-sensor data fusion by a hybrid methodology—A comparative study[J]. Computers in Industry, 2016,75: 27-34.

[11] SUN Zhendong. Multi-source data fusion oriented bayesian estimation method research[J].Journal of Qilu University of Technology, 2018, 32(1): 73-76.

[12] MA Yongjun, XUE Yonghao, LIU Yang, et al. Data aggregation algorithm based on the model of deep learning[J]. Journal of Tianjin University of Science & Technology, 2017, 32(4): 71-75.

[13] ZHANG W. Design and Implementation of data fusion method based on D-S evidence theory[D]. Beijing: Beijing University of Posts and Telecommunications, 2018.(in Chinese)

[14] LOSSA J A, PATWARI N, HERO III A O. Distributed weighted-multidimensional scaling for node localization in sensor networks[J]. ACM Transactions on Sensor Networks, 2006, 2(1): 39-64.

[15] EFATMANESHNIK M, ALAM N, KEALY A, et al. A fast multidimensional scaling filter for vehicular cooperative positioning[J]. Journal of Navigation, 2012, 65(2): 223-243.

[16] WANG Leigang. Research on distributed cooperative localization for multi-mobile agents under communication constraints[D]. Beijing: Tsinghua University, 2017. (in Chinese)

[17] WANG Leigang, ZHANG Tao, GAO Feifei. Distributed cooperative localization with lower communication path requirements[J]. Robotics and Autonomous Systems, 2016, 79: 26-39.

[18] COSTA J A, PATWARI N, HERO A O. Distributed weighted-multidimensional scaling for node localization in sensor networks[J]. Transactions on Sensor Networks, 2006, 2(1): 39-64.

[19] HAN Chongzhao, ZHU Hongyan, DUAN Zhansheng. Multi-source information fusion[M]. Beijing: Tsinghua University Press, 2006. (in Chinese)

Author Dr. **WANG Leigang** received his Ph.D. degree in control science and engineering from Tsinghua University, Beijing, China, in 2017. He is now a research assistant with State Key Laboratory of Complex Electromagnetic Environment Effects, Luoyang, China. His current research activities are focused on inertial navigation system, cooperative localization for autonomous vehicles and multi-source information-fusion.

Author contributions Dr. **WANG Leigang** designed the study, complied the models, conducted the analysis, interpreted the results and wrote the manuscript. Mr. **KONG Depei** and Mr. **WANG Jianlu** contributed to the discussion and background of the study. Mr. **ZHOU Jihang** contributed to data and model components for the optimization model. All authors commented on the manuscript draft and approved the submission.

Competing interests The authors declare no competing inrests.

(Production Editor: SUN Jing)

一种协作目标定位与协作自定位的并行集成方法

王雷钢，孔德培，周继航，王建路

（电子信息系统复杂电磁环境效应国家重点实验室，洛阳 471003，中国）

摘要：当群智能体在进行目标跟踪时，它们可以以协作的方式对自身和目标进行定位。为了最大限度地发挥群协作的优势，本文设计了一种协作目标定位（Cooperative target-localization, CTL）和协作自定位（Cooperative target-localization, CSL）的并行集成的运行策略。首先，建立一个包含群智能体位置和目标位置的全局代价函数。其次，在CTL过程中，随着智能体位置被重新估计，使用U变换推算各智能体位置估计之间的误差协方差。仿真结果表明，与CTL和CSL独立运行的情况相比，该策略利用了更多的信息进行目标和智能体的位置估计，可以获得全局最优的智能体和目标位置估计结果。

关键词：协作自定位；协作目标定位；非经典多维变换；优势函数