Compensation Algorithm Based on Estimation State for Transmission Delay in Cooperative Localization

Wang Leigang(王雷钢)[1,2], *Zhang Tao*(张涛)[1*], *Yang Weifeng*(杨伟锋)[1]

1. Department of Automation, Tsinghua University, Beijing, 100084, P. R. China;
2. Luoyang Electronic Equipment Test Center of China, Luoyang, 471003, P. R. China

(Received 7 November 2013; revised 27 December 2013; accepted 30 December 2013)

Abstract: In order to maximize the utilization of the observation information in the cooperative localization, a compensation algorithm based on the estimation state is presented for transmission delay. Under the framework of the Kalman filter, two different processes of state estimating with and without transmission delay are investigated and contrasted. The expression of difference quantity caused by transmission delay is derived. It is used to compensate the present estimation state instead of the observed information compensation. According to the characteristics of state transition matrix, an equivalent expression of which successively impacts on the covariance factor in delay time is obtained. The simulation results show that the present estimated state is effectively corrected by transmission information and the relevance among agents is accurately updated. As a result, a higher positioning accuracy is achieved. Meanwhile, the consumption of recording and multiplication of the state transition matrix is saved.

Key words: cooperative localization; Kalman filter; transmission delay; covariance factor

CLC number: U674.7 **Document code**: A **Article ID**: 1005-1120(2014)01-0091-08

1 Introduction

In the cooperative localization (CL), by integrating and sharing the absolute measurements (e. g., GPS, landmark, etc.) and the relative measurement (distance/bearing), the accuracy and reliability of the estimation position of each agent in a group are improved drastically[1-2]. The advantage of the CL stems from the individual measurement information can be shared among the members of a group[3-4]. However, this advantage comes at the cost of increased computation and communication requirements[5]. For example, the agents equipped with high precision position device or luckily receiving GPS need to broadcast their own state information to other agents in a timely and accurate communication manner.

When there exists the communication link, the transmission delay is inevitable for the environment or information processing. Based on Kalman filter, there is some research on this issue[6-7], such as extrapolation of measurements[8], re-organized innovation[9], state augmented and so on. In Ref. [10], by establishing a delay, state-space model for state prediction and delay information is fed back to the model. In Ref. [11], the history state parameters are reserved for observation update, by which the predicted estimation state is acquired and then used to substitute the present state estimation. While the relevance of the information is not considered and the available information is thus utilized incompletely. A delayed extended Kalman filter (DEKF) is designed by state reverse in Ref. [12], where the relevance of the information is considered. But big storage space is consumed and the estimation is conservative. In Ref. [13], based on the path prediction of master unmanned underwater vehicles (UUV), an unequal interval filtering algorithm without state reverse is proposed. However, the state of the master UUV need to

Foundation item: Supported by the National Basic Research Program of China (2010CB731800).

* **Corresponding author**: Zhang Tao, Professor, E-mail: taozhang@tsinghua.edu.cn.

be propagated in each slave UUV. Though it is suitable for leader-follower cooperative structure, it does not fit for the distributed structure in which all the agents are identical in a group.

2 Related Models

Let us consider a group of n mobile agents which is moving in two-dimension (2D). And the proprioceptive sensors (e. g. , wheel encoders), exteroceptive sensors (e. g. , laser rangefinder, radio ranging) and GPS receiver are mounted on each agent and all of these sensors are homogeneous. The proprioceptive sensors measure the self motion of the agent and monitor the environment for localization features. And the exteroceptive sensors detect and identify other agents moving in its vicinity and measure their respective distance. The objective of CL is to combine these measurements with the equation of motion to obtain more accurate position estimation.

(1) Agent motion model

The ith agent equation of motion is as follows[14]

$$\boldsymbol{x}_i = \begin{bmatrix} \dot{x}_i \\ \dot{y}_i \\ \dot{\psi}_i \end{bmatrix} = \begin{bmatrix} V_i \cos(\psi_i) \\ V_i \sin(\psi_i) \\ \omega_i \end{bmatrix}$$

where $\boldsymbol{x}_i = [x_i, y_i, \psi_i] \in \mathbf{R}^{3\times 1}$ is the agent state, including agent location (x_i, y_i) and agent heading ψ_i. It is assumed that onboard introspective sensors measure the linear speed V_i and angular speed ω_i of the agent. Each agent has a GPS receiver and an exteroceptive sensor to measure agent-to-agent distance.

(2) Measurement model

$$V_i = \bar{V}_i + \varepsilon_i, \quad \omega_i = \bar{\omega}_i + \zeta_i \tag{1}$$

$$z_i(k) = h_i(\bar{x}_i(k)) + w_i(k) \tag{2}$$

$$z_{ij}(k) = h_{ij}(\bar{x}_i(k), \bar{x}_j(k)) + v_{ij}(k) \tag{3}$$

where k is the agent index. $\bar{V}_i$, $\bar{\omega}_i$, $\bar{x}_i$ are the true values. z_i, z_{ij} are the absolute measurements and the relative measurements. Suppose that ε_i, ζ_i, w_i, v_{ij} are subject to the standard normal distribution. For normal distribution, there are moment parameter and information parameter two expressions. In this paper, the former, namely, the mean and covariance $\{\hat{\boldsymbol{X}}, \boldsymbol{P}\}$ are adopted.

For the nonlinear system, the extended Kalman filter (EKF) is used to estimate the moment parameters. By Taylor series expansion, the approximate transition matrix $\boldsymbol{\Phi}$, the control matrix $\boldsymbol{B}$, and the observation matrix $\boldsymbol{H}$ are acquired as follows

$$\boldsymbol{\Phi} = \begin{bmatrix} 1 & 0 & -VT_s\sin(\psi) \\ 0 & 1 & VT_s\cos(\psi) \\ 0 & 0 & 1 \end{bmatrix}$$

$$\boldsymbol{B} = \begin{bmatrix} T_s\cos(\psi) & 0 \\ T_s\sin(\psi) & 0 \\ 0 & T_s \end{bmatrix}, \ \boldsymbol{H} = \frac{\partial h(\boldsymbol{x})}{\partial \boldsymbol{x}} \Big|_{\boldsymbol{x}=\hat{\boldsymbol{x}}(k)}$$

And thus state propagation is given as

$$\tilde{\boldsymbol{x}}(k+1) = \hat{\boldsymbol{x}}(k) + \boldsymbol{B}(k) \cdot \boldsymbol{u}$$
$$\tilde{\boldsymbol{z}}(k+1) = \boldsymbol{H} \cdot \hat{\boldsymbol{x}}(k+1) \tag{4}$$

where T_s is the simulation cycle and $\boldsymbol{u} = [V, \omega]^{\mathrm{T}}$ and $\hat{\boldsymbol{x}}$ the estimated state vector. $\tilde{\boldsymbol{x}}$ and $\tilde{\boldsymbol{z}}$ denote the prediction state and observation, respectively.

Kalman filter estimation includes propagation and observation update cycles. In the perception cycle, based on the evolution of the system equations, the measured control inputs and the statistical description of the system noise, the knowledge about the system state is propagated to the next step. In the observation update cycle, where relative pose measurements are processed to update the propagated estimates calculated during the previous cycle. In the CL, most of the time the agents propagate their position and covariance estimation based on their own perception, while the observation updates are usually rare, and they take place only when the measurements occur [3].

3 Compensation Based on Estimation State

The transmission delay issue caused by poor communication in CL is focused in this paper. Assume that the agent i gets the GPS or the relative measurements to other agent at time step k and these measurements are processed and packed. Consequently, the information packet $C_i(k)$ is formed and then repeatedly broadcasts until it is received by the others in the group or a new measurement occurs. If $C_i(k)$ can be received by other agents immediately, the observation update will be done in the receiver as usual[3-4]. How ever, in sheltered areas such as buildings, forests, valleys or a variety of hostile jamming environment, the information packets $C_i(k)$ may be delayed for some time before being received by the agent j. If the available $C_i(k)$ at time step $k+m$ is discarded and not used to the present state updates of agent j, it will be a great waste. The primary contribution of the proposed algorithm is that the k-moment observation information of the agents i can be used to update the state of agent j at time step $k+m$. And the compensation manoeuvre is based on the estimated state rather than extrapolating or re-organizing the observed information as mentioned above.

The distribution algorithm is the foundation in the CL. And a large body of work has been done in the recent past. Since other agent′s position is used for new position estimation, the correlation between them arises naturally. The quantitative representation of the correlation is a covariance term, which can be regarded as an indicator about how much information has been shared by different agents. And the key of distributed implement is to assign the correlation to different agents and keep them update, such as singular value decomposition (SVD) [3], augmented information filter (AIF) [4], covariance intersection (CI) algorithm [15-16] and so on.

In order to implement the distributed computing of the covariance term, in this paper a bank of propagation calculators $\bar{\boldsymbol{P}}_i = [\bar{\boldsymbol{P}}_i^1, \bar{\boldsymbol{P}}_i^2, \cdots, \bar{\boldsymbol{P}}_i^n]$ $(i=1,\cdots,n)$ are established in each agent. When the first relative measurement occurs between the master and slave agents (the master can detect other agents and receive the state estimation information from others and the slave is just opposite. This relationship is relative and it will determine whether original base cells should be set to identity matrix or not in an observation update cycle), a new covariance is calculated in the master agent and then retained in it as the original base cells of propagation calculators. Accordingly, in the slave agent, the counterpart is set to identity matrix automatically. In a time update cycle, all these original base cells in propagation calculators are used for update of covariance factors, and only its own state transition matrix is required. In the observation update cycle, as new original base cells, the new covariance matrix and identity matrix are re-assigned to the master and slave agents respectively. In the time update cycle, covariance factors are dependently propagated in each agent. By this method, not only the independence is guaranteed and the covariance update is maintained, but also the error propagation of matrix decomposition and calculation consumption is avoided. The above-mentioned method is used as basis for delay effect analysis in the following.

3.1 No transmission delay

In this case, firstly, the following assumptions are stated:

(1) The observed information is available instantaneously and the agent j receives the information packet $C_i(k)$ (at least including v_i, $\boldsymbol{H}_i$, $\boldsymbol{S}_i$, $\bar{\boldsymbol{P}}_i$, where the innovation is $v_i = z_i(k) - h_i(\tilde{x}(k))$, and the covariance of innovation is $\boldsymbol{S}_i = \boldsymbol{H}_i(k)\tilde{\boldsymbol{P}}(k)(\boldsymbol{H}_i(k))^{\mathrm{T}} + \boldsymbol{R}_i$) from agent i.

(2) The agent j is the master and the agent i is the slave during the last progress of transmission, and before update the moment parameters of agent i and j are $\tilde{x}_i(k)$, $\bar{\boldsymbol{P}}_i(k)$, $\tilde{x}_j(k)$, $\bar{\boldsymbol{P}}_j(k)$, respectively.

And the covariance factors of them are thus updated as Eqs. (5—8).

Agent i

$$\boldsymbol{u}\_\bar{\boldsymbol{P}}_i^j(k) = \boldsymbol{I} \tag{5}$$

Agent j

$$\tilde{\boldsymbol{P}}_{ji}(k) = \bar{\boldsymbol{P}}_j^i(k) \cdot \bar{\boldsymbol{P}}_i^j(k) \tag{6}$$

$$\hat{x}_j(k) = \tilde{x}_j(k) + \tilde{\boldsymbol{P}}_{ji}(k)\boldsymbol{H}_i^{\mathrm{T}}(\boldsymbol{S}_i)^{-1} v_i \tag{7}$$

$$\boldsymbol{u}\_\tilde{\boldsymbol{P}}_j^i(k) = \tilde{\boldsymbol{P}}_{ji}(k) - \tilde{\boldsymbol{P}}_{ji}(k)\boldsymbol{H}_i^{\mathrm{T}}(\boldsymbol{S}_i)^{-1}\boldsymbol{H}_i\tilde{\boldsymbol{P}}_{ii}(k) \tag{8}$$

Based on Eqs. (5—8), in the following time update phrase, the covariance factors of agents are updated respectively. At time step $k+m$, the covariance factors are updated as

$$\bar{\boldsymbol{P}}_i^j(k+m)=\boldsymbol{u}\_\bar{\boldsymbol{P}}_i^j(k)\cdot\boldsymbol{\Gamma}_i^{\mathrm{T}}(k,m) \tag{9}$$

$$\bar{\boldsymbol{P}}_j^i(k+m)=\boldsymbol{\Gamma}_j(k,m)\cdot\boldsymbol{u}\_\bar{\boldsymbol{P}}_j^i(k) \tag{10}$$

where $\boldsymbol{\Gamma}_i(k,m)=\boldsymbol{\Phi}_i(k+m)\cdots\boldsymbol{\Phi}_i(k+1)$ is defined as the state transition matrix impacting factor.

Considering Eqs. (6, 8, 9, 10), the state of agent i and the covariance between agents i and j are obtained as

$$\tilde{x}_j(k+m)=\tilde{x}_j(k)+\tilde{\boldsymbol{P}}_{ji}(k)\boldsymbol{H}_i^{\mathrm{T}}(\boldsymbol{S}_i)^{-1}v_i+\sum_{l=1}^{m}\boldsymbol{B}(k+l)\boldsymbol{u}(k+l)T_{\mathrm{s}} \tag{11}$$

$$\begin{gathered}\tilde{\boldsymbol{P}}_{ji}(k+m)=\bar{\boldsymbol{P}}_j^i(k+m)\cdot\bar{\boldsymbol{P}}_i^j(k+m)=\\ \boldsymbol{\Gamma}_j(k,m)\cdot\bar{\boldsymbol{P}}_j^i(k)\cdot\bar{\boldsymbol{P}}_i^j(k)\cdot\boldsymbol{\Gamma}_i^{\mathrm{T}}(k,m)+\boldsymbol{\Gamma}_j(k,m)\cdot\\ (-\tilde{\boldsymbol{P}}_{ji}(k)(\boldsymbol{H}_i)^{\mathrm{T}}(\boldsymbol{S}_i)^{-1}\boldsymbol{H}_i\tilde{\boldsymbol{P}}_{ii}(k))\cdot\boldsymbol{\Gamma}_i^{\mathrm{T}}(k,m)\end{gathered} \tag{12}$$

3.2 *m* transmission delay cycles

In this case, the agent j gets the information packet $C_i(k)$ from agent i at time step $k+m$ not k. So in the agent j, the time update is still going on at time step k. (In principle, the observation update should be done at this time.) In the following time update phrase, the role of the agent i is still the master relative to agent j. However, under normal communication, the master-slave relationship between agents i and j should be changed at time step k. Therefore, this relationship is erroneously remained for m transmission delay cycles. At time step $k+m$, the covariance $\bar{\boldsymbol{P}}_i^j(k+m)$ $\bar{\boldsymbol{P}}_j^i(k+m)$ are updated as

Agent i

$$\bar{\boldsymbol{P}}_i^j(k+m)=\boldsymbol{\Gamma}_i(k,m)\cdot\bar{\boldsymbol{P}}_i^j(k) \tag{13}$$

Agent j

$$\bar{\boldsymbol{P}}_j^i(k+m)=\bar{\boldsymbol{P}}_j^i(k)\cdot\boldsymbol{\Gamma}_j^{\mathrm{T}}(k,m) \tag{14}$$

$$\bar{\tilde{x}}_j(k+m)=\tilde{x}_j(k)+\sum_{l=1}^{m}\boldsymbol{B}(k+l)\boldsymbol{u}(k+l) \tag{15}$$

Contrasting Eqs. (11, 15), it obtains the difference quantity of estimated state parameters between no delay and m delay cycles as follows

$$\Delta\tilde{x}_i=\tilde{\boldsymbol{P}}_{ji}(k)(\boldsymbol{H}_i)^{\mathrm{T}}(\boldsymbol{S}_i)^{-1}v_i \tag{16}$$

Notice that in the estimation of $\Delta\tilde{x}_i$ the knowledge of $\tilde{\boldsymbol{P}}_{ji}(k)$ is required. Assume that the state transition matrix impacting factor $\boldsymbol{\Gamma}$ is known. And then the covariance factor $\bar{\boldsymbol{P}}_j^i(k)$ in agent j at time step k can be deduced by Eq. (14). Furthermore, the prediction covariance $\tilde{\boldsymbol{P}}_{ji}(k)$ is acquired. So how to get $\boldsymbol{\Gamma}$ becomes critical for the issue at hand. From the original form of $\boldsymbol{\Gamma}$, it needs historical state transition matrix to be multiplied continuously. As these can be prohibitively expensive in calculation and storage, it is motivated to pursue a reduced-complexity equivalent expression that is introduced next.

Since

$$\boldsymbol{\Phi}(k)=\boldsymbol{I}+\begin{bmatrix}0 & 0 & -V(k)T_{\mathrm{s}}\sin\psi(k)\\ 0 & 0 & -V(k)T_{\mathrm{s}}\cos\psi(k)\\ 0 & 0 & 0\end{bmatrix}$$

$$\text{Let }\boldsymbol{M}(k)=\begin{bmatrix}0 & 0 & -V(k)T_{\mathrm{s}}\sin\psi(k)\\ 0 & 0 & -V(k)T_{\mathrm{s}}\cos\psi(k)\\ 0 & 0 & 0\end{bmatrix}$$

Then

$$\boldsymbol{\Phi}(k)\cdot\boldsymbol{\Phi}(k+1)=(\boldsymbol{I}+\boldsymbol{M}(k))(\boldsymbol{I}+\boldsymbol{M}(k+1))=\boldsymbol{I}+\boldsymbol{M}(k)+\boldsymbol{M}(k+1)$$

By extension

$$\boldsymbol{\Gamma}=\boldsymbol{I}+\begin{bmatrix}0 & 0 & -T_{\mathrm{s}}\sum\limits_{l=k}^{l=k+m}V(l)\sin(\psi(l))\\ 0 & 0 & -T_{\mathrm{s}}\sum\limits_{l=k}^{l=k+m}V(l)\cos(\psi(l))\\ 0 & 0 & 0\end{bmatrix} \tag{17}$$

The equivalent expression of the state transition matrix impacting factor $\boldsymbol{\Gamma}$ is shown in Eq. (17). The expression form is transformed from multiplication to addition, which has two advantages. On the one hand, the consumption of matrix multiplication is saved. On the other, only measured velocity, angular velocity are needed for $\boldsymbol{\Gamma}$. It is not necessary to record all the state transition matrix from the previous time of $C_i(k)$ arrival to now.

The present covariance can be acquired by Eqs. (13, 14) as follows

$$\bar{\tilde{\boldsymbol{P}}}_{ji}(k+m)=\boldsymbol{\Gamma}_j(k,m)\cdot(\bar{\boldsymbol{P}}_j^i(k))^{\mathrm{T}}\cdot(\bar{\boldsymbol{P}}_i^j(k))^{\mathrm{T}}\cdot\boldsymbol{\Gamma}_i^{\mathrm{T}}(k,m) \tag{18}$$

Contrasting Eqs. (12, 18), the difference of the covariance between no transmission delay and m delay cycles is obtained as

$$\Delta\boldsymbol{P}=\boldsymbol{\Gamma}_j(k,m)\cdot(-\tilde{\boldsymbol{P}}_{ji}(k)(\boldsymbol{H}_i)^{\mathrm{T}}\cdot(\boldsymbol{S}_i)^{-1}\boldsymbol{H}_i\tilde{\boldsymbol{P}}_{ii}(k))\cdot\boldsymbol{\Gamma}_i^{\mathrm{T}}(k,m)=-\bar{\boldsymbol{P}}_j^i(k+m)\cdot$$

$$(\bar{\boldsymbol{P}}_i^j(k))^{\mathrm{T}} \cdot (\boldsymbol{H}_i)^{\mathrm{T}}(\boldsymbol{S}_i)^{-1} \cdot \boldsymbol{H}_i \cdot \widetilde{\boldsymbol{P}}_{ii}(k) \cdot (\bar{\boldsymbol{P}}_i^j(k))^{-\mathrm{T}} \cdot \bar{\boldsymbol{P}}_i^j(k+m)^{\mathrm{T}} \tag{19}$$

Notice that $\bar{\boldsymbol{P}}_i^j(k)=\bar{P}_i^j(k_p) \cdot \prod_{l=k_p}^{l=k}\boldsymbol{\Phi}(l)$ (where k_p denotes the time when the last observation occurs before step k and $\bar{\boldsymbol{P}}_i^j(k_p)$ is derived from the covariance matrix decomposition) and $|\bar{\boldsymbol{P}}_i^j(k)|=|\bar{\boldsymbol{P}}_i^j(k_p)| \cdot |\prod_{l=k_p}^{l=k}\boldsymbol{\Phi}(l)|\neq 0$. And $\bar{\boldsymbol{P}}_i^j(k)$ is thus invertible matrix and Eq. (19) is valid.

By far, both the state and covariance compensation amount can be obtained by Eqs. (16, 19). The present estimated state can be updated by $\Delta\tilde{x}$ and $\Delta\boldsymbol{P}$ accordingly. The state and covariance factor of agents i and j are updated respectively as follows

Agent i

$$\boldsymbol{u}\_\bar{\boldsymbol{P}}_i^j(k)=\boldsymbol{I} \tag{20}$$

Agent j

$$\tilde{x}_j(k+m)=\bar{\tilde{x}}_j(k+m)+\Delta\tilde{x}$$

$$\boldsymbol{u}\_\bar{\boldsymbol{P}}_j^i(k)=\bar{\widetilde{\boldsymbol{P}}}_{ij}(k+m)+\Delta\boldsymbol{P} \tag{21}$$

In summary, the algorithm flow on CL is described as Fig. 1 considering the transmission delay. When the information packet arrives, it should be determined whether there exits time delay or not (each agent requires rigorously time synchronization). If not, the state parameters and covariance factors are updated naturally. Otherwise, the compensation should be considered and then the compensation update is completed.

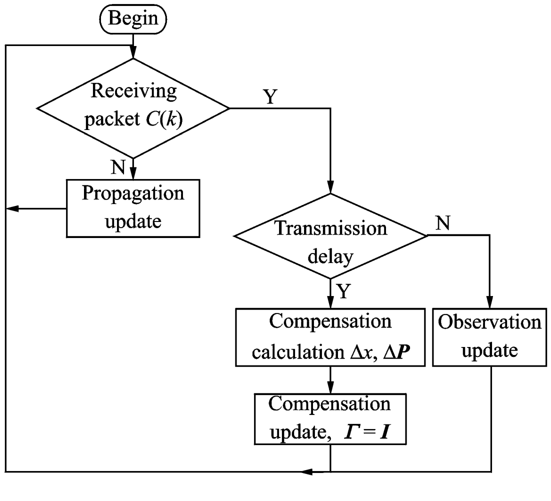

Fig. 1 Flowchart of CL considering transmission delay

By the algorithm, there are several characteristics as follows:

(1) During each observation cycle as well as "delay" observation, the total communication cost per agent is reduced from $O((n-1)(t_2-t_1)/T_s)$ to $O(n-1)$ (where t_1, t_2 denote the moments when last and present observation take place, respectively). Even so, the relation among agents can be well-kept.

(2) The state transition matrix impacting factor $\boldsymbol{\Gamma}$ in "delay" observation cycle can be obtained by addition rather than matrix multiplication. The computation cost per agent is $O(2m)$.

(3) Since the moment when the observation occurs is not available for the other receiver agents in advance, theoretically, all state information $\boldsymbol{\Phi}(k)$ should be recorded for later use. By Eq. (12), it shows that history velocity, angular velocity instead of $\boldsymbol{\Phi}(k)$ are enough for $\boldsymbol{\Gamma}$.

4 Simulation Results

It is supposed that three robots are adopted in simulation, since the generalization to an arbitrary number of robots is straightforward and all of them move along different circles in 2D. The relative positions are shown in Fig. 2. The starting positions are (0, −0. 4), (1. 5, −0. 4), (0, 1. 9). Robot 1 is moving counter-clockwise. And Robot 2 and Robot 3 are moving clockwise. The simulation period is $T_s=0.4$ s. The odometry and radio measurements of linear and rotational velocity and agent-to-agent distance are corrupted by zero-mean Gaussian noise with $\sigma_V=0.05$, $\sigma_\omega=0.1$, $\sigma_r=0.001$. It should be noted that since the performance of navigation unit (e. g. , accelerometer, gyroscope) is not considered in this paper, they are assumed to be ideal in drift herein and accumulated error caused by drift is omitted advisedly for the purpose of this paper.

In simulation, Agent 2 receives GPS at certain intervals. Agents 1 and 3 cannot obtain GPS but they can detect Agent 2 at some time. The scheme is designed as follows:

(1) The time series when Agent 2 gets GPS are 0. 4, 3. 6, …, 19. 6, with an interval of 3. 2 s

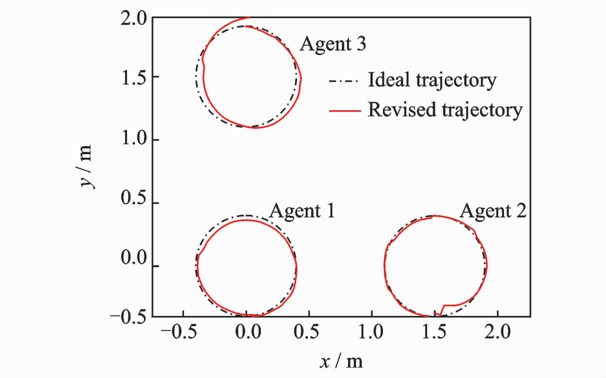

Fig. 2 Trajectories of three agents with delay compensation

(the simulation lasts 20 s).

(2) The time series when Agent 1 detects Agent 2 are 2, 3.2, 4.8, 9.6, 14.4.

(3) The time series when Agent 3 detects Agent 2 read 2.4, 4, 6, 11.2, 16.

It is supposed that at $t=6.8$ s ($k=18$), the observation information of Agent 2 cannot be received by Agent 1 and Agent 3 immediately. Until five cycles pass ($m=5$), the observation is accepted by Agents 1 and 3.

Fig. 2 shows the relative layout and the trajectories of three agents, where the red solid curve represents the revised trajectory by GPS and relative distance with transmission delay. Although Agents 1 and 3 receive the observation information from Agent 2 with transmission delay, by the compensation the delay effect on position estimation can be eliminated. In Fig. 3, the comparisons of the position error of Agent 1 under no-delay, and delay with/without compensation are demonstrated. The black dashed curve represents the transmission delay, which is compensated by the compensation algorithm in this paper. It can be seen from Figs. 3—5 that the position estimation precision will be reduced (As in Fig. 5, there is at least no increase error) if delay information can be utilized. By the proposed method, the estimation curve is drawn back to the circumstances with no delay once compensation occurs. Otherwise, it will continue to be the green curve. After compensation, the estimated position curve is almost in accord with that of no transmission delay.

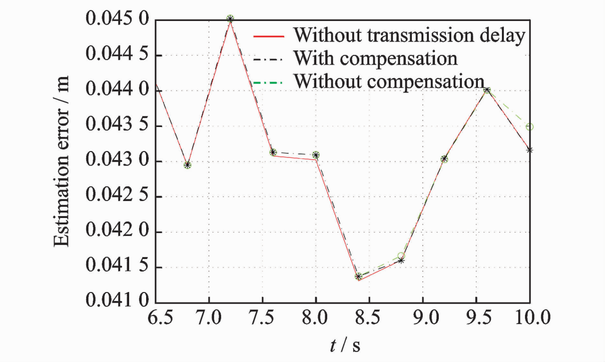

Fig. 3 Comparison of estimated position of Agent 1 with/without transmission delay

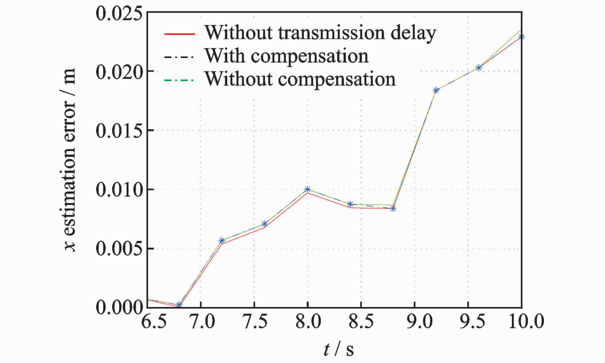

Fig. 4 Comparison of estimated x coordinate of Agent 1 with/without transmission delay

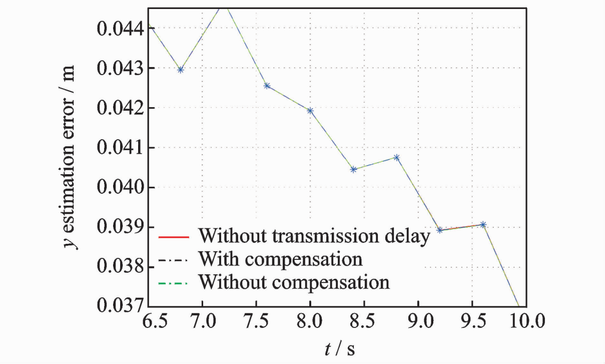

Fig. 5 Comparison of estimated y coordinate of Agent 1 with/without transmission delay

In order to verify the validity of the algorithm, three simulation modes are adopted in the simulation. In the first mode, the observation information from Agent 2 at $k=18$ is received by others, which is a benchmark for the other two modes. The second mode is that the observation information cannot be received before the next new measurement. While the third mode is that the observation information is received after 3 delay cycles pass ($k+m=21$) and then the receiver agents are compensated. The comparison results of the estimated state $\hat{x}$，$\hat{y}$，$\hat{\psi}$ and the error of Agent 3 under no delay and delay with/without compensation are demonstrated in the Tables 1—3. By compensation，the error amount is reduced to about 10%. So the precision of the estimated state is improved dramatically due to the compensation. Moreover，the simulations also show that even though no relative measurement occurs among agents during time step $k=18$—23，for relation which has been setup before，Agents 1 and 3 can benefit from the absolute measurement from Agent 2.

Table 1 Comparison of estimated $\hat{x}$ of Robot 3 under three modes m

Time	18	19	20	21	22	23
No delay	0.395 00	0.370 63	0.341 62	0.305 22	0.267 23	0.224 71
Delay without compensation	0.395 13	0.370 76	0.341 74	0.305 34	0.267 34	0.224 83
error	0.000 13	0.000 12	0.000 12	0.000 12	0.000 12	0.000 12
Delay with compensation	0.395 13	0.370 76	0.341 74	0.305 21	0.267 22	0.224 70
error	0.000 13	0.000 12	0.000 12	0.000 01	0.000 01	0.000 01

Table 2 Comparison of estimated $\hat{y}$ of Robot 3 under three modes m

Time	18	19	20	21	22	23
No delay	1.306 74	1.266 30	1.225 98	1.187 48	1.155 13	1.127 71
Delay without compensation	1.306 37	1.265 93	1.225 62	1.187 11	1.154 77	1.127 35
error	0.000 37	0.000 37	0.000 37	0.000 36	0.000 36	0.000 36
Delay with compensation	1.306 37	1.265 93	1.225 62	1.187 48	1.155 14	1.127 71
error	0.000 37	0.000 37	0.000 37	0.000 01	0.000 01	0.000 01

Table 3 Comparison of estimated $\hat{\psi}$ of Robot 3 under three modes m

Time	18	19	20	21	22	23
No delay	−1.985 66	−2.113 12	−2.194 63	−2.328 08	−2.436 27	−2.568 68
Delay without compensation	−1.985 73	−2.113 18	−2.194 69	−2.328 14	−2.436 34	−2.568 74
error	0.000 06	0.000 06	0.000 06	0.000 06	0.000 06	0.000 06
Delay with compensation	−1.985 73	−2.113 18	−2.194 69	−2.328 08	−2.436 27	−2.568 68
error	0.000 06	0.000 06	0.000 06	0.000 00	0.000 00	0.000 00

5 Conclusions

The reliability of the algorithm in CL must be considered in poor communication scenario. The proposed method of state and covariance compensation algorithm for communication delay is useful，especially for the case where the observation information is subject to several times broadcasting and then is received by other agents. By using the method，the observation information is maximally utilized. The distinguishable feature of the algorithm is that it is based on the estimated state compensation rather than the observed information correction. However，it is assumed that the estimated moment parameters in the two different cases are approximate in a short period. Whereby，the state transition matrices at same moments in different cases are identical. But there are some differences in practice，which explains why the results in compensated states are not completely consistent with those of no transmission delay in the simulations.

References：

[1] Burgard W，Moors M，Fox D，et al. Collaborative multi-robot exploration［C］// IEEE International Conference on Robotics and Automation. ［S. l.］：IEEE，2000，1：476-481.

[2] Nicosia J. Decentralized cooperative navigation for spacecraft［C］// IEEE Aerospace Conference. ［S.

l.]: IEEE, 2007: 1-6.

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

[4] Mu Hua. Decentralized algorithms of cooperative navigation for mobile platform[D]. Changsha: National University of Defense Technology, 2010. (in Chinese)

[5] Trawny N, Roumeliotis S I, Giannakis G B. Cooperative multi-robot localization under communication constraints[C]// IEEE International Conference on Robotics and Automation. [S. l.]: IEEE, 2009: 4394-4400.

[6] Wang Z D, Yang F W, Daniel W C Ho, et al. Robust H_∞ filtering for stochastic time-delay systems with missing measurements[J]. IEEE Transactions on Signal Processing, 2006, 54 (7):2579-2587.

[7] Lu X, Xie L H, Zhang H S, et al. Robust Kalman filtering for discrete-time systems with measurement delay[J]. IEEE Transactions on Circuits and Systems, 2007,54(6):522-526.

[8] Larsen T D, Andersen N A, Ravn O, et al. Incorporation of time delayed measurements in a discrete-time Kalman filter[C]// 37th IEEE Conference on Decision and Control. Tampa, USA: [s. n.], 1998, 4:3972-3977.

[9] Lu X, Zhang H S, Wang W, et al. Kalman filtering for multiple time-delay systems[J]. Automatica, 2005,41(8):1455-1461.

[10] Chen H Y, Wang J D, Xu T. Flight delay state space model based on genetic EM algorithm[J]. Transactions of Nanjing University of Aeronautics and Astronautics, 2011,28(3): 276-281.

[11] Maczka D K, Gadre A S, Stilwell D J. Implementation of a cooperative navigation algorithm on a platoon of autonomous underwater vehicles[C]// IEEE International Conference on OCEANS. [S. l.]: IEEE,2007:1-6.

[12] Yao Y, Xu D, Yan W. Cooperative localization with communication delays for MAUVs[C]// IEEE International Conference on Intelligent Computing and Intelligent Systems. [S. l.]: IEEE, 2009, 1: 244-249.

[13] Yao Yao, Xu Demin. Cooperative localization of multiple UUVs with communication delays a real-time update method based on path prediction[J]. Robot, 2011,33(2):161-168. (in Chinese)

[14] Rajnikant Sharma. Bearing-only cooperative localization and path-planning of ground and aerial robots [D]. Provo, USA: Department of Electrical and Computer Engineering, Brigham Young University, 2011.

[15] Chen L, Arambel P O, Mehra R K. Estimation under unknown correlation: Covariance intersection revisited[J]. IEEE Transactions on Automatic Control, 2002, 47(11): 1879-1882.

[16] Arambel P O, Rago C, Mehra R K. Covariance intersection algorithm for distributed spacecraft state estimation[C]// IEEE Proceedings of the 2001 American Control Conference. [S. l.]: IEEE, 2001, 6: 4398-4403.

(Executive editor: Zhang Tong)