International Journal of Control, Automation, and Systems, vol. 4, no. 2, pp. 165-171 April 2006

A Data Fusion Algorithm of the Nonlinear System Based on Filtering Step By Step

Cheng-lin Wen and Quan-bo Ge

Abstract: This paper proposes a data fusion algorithm of nonlinear multisensor dynamic systems of synchronous sampling based on filtering step by step. Firstly, the object state variable at the next time index can be predicted by the previous global information with the systems, then the predicted estimation can be updated in turn by use of the extended Kalman filter when all of the observations aiming at the target state variable arrive. Finally a fusion estimation of the object state variable is obtained based on the system global information. Synchronously, we formulate the new algorithm and compare its performances with those of the traditional nonlinear centralized and distributed data fusion algorithms by the indexes that include the computational complexity, data communicational burden, time delay and estimation accuracy, etc.. These compared results indicate that the performance from the new algorithm is superior to the performances from the two traditional nonlinear data fusion algorithms.

Keywords: Centralized fusion, distributed fusion, EKF, nonlinear system, step by step.

1. INTRODUCTION

The constantly increasing complexity in modern military affairs and civil areas is in urgent need of new technological methods for comprehensively processing, explaining, and estimating tremendous amounts of information. Accordingly, the multisensor data fusion theory can be further developed and the corresponding technology can be further perfected. The key problems of data fusion are model design and the fusion algorithm. By the efforts of many researchers, we have a series of multisensor system models and data fusion algorithms all aiming at different objects [1-12].

Most of the current fusion algorithms demand the systematic state and observation equation to be linear. However, many fusion systems can't always be described in a simple linear system. So the study and design of the nonlinear system fusion algorithm have much sense in theory and a bright future in application [1]. The traditional fusion algorithms mainly include two types; centralized fusion and distributed fusion (including feedback fusion and non-feedback fusion and so on). The advantage of the former is its high fusion accuracy while its defects are great difficulty of data association and high requirement of the central processor. Although the latter algorithm can lighten the computational burden of the central processor through the local processor, its total computational complexity (C.C.) and communication burden will increase [2]. That is to say, none of the current fusion algorithms can achieve the best in overall performance; consequently it will affect the application of these algorithms in a practical system.

In light of the above problems, this paper introduces the idea of filtering step by step with a nonlinear system of synchronous sampling as its object, and proposes a data fusion algorithm of the nonlinear system based on filtering step by step. We prove the effectiveness of the new algorithm by comparing the performance indexes including C.C. of the algorithm, communicational complexity, and time delay of predict and estimate accuracy, etc. with the traditional nonlinear centralized data fusion algorithm and distributed data fusion algorithms. It's basic idea is: when all of the observations aiming at the target are obtained, firstly we can predict the object state based on previous system information and then use extended Kalman filtering and all of the local observations to update the estimated value of an object state in turn. Accordingly we can obtain a global fusion estimate value of an object state based on the global information.

Manuscript received October 1, 2005; accepted January 12, 2006. Recommended by Editor Zengqi Sun. This work was supported by NSFC (60434020, 60572051), International Cooperative Project Fund (0446650006), Henan Outstanding Youth Science Fund (0312001900), and Ministry of Education Science Fund (205092).

Cheng-lin Wen is with the College of Automation, Hangzhou Dianzi University, Hangzhou, 310018, China (e-mail: wencl@hziee.edu.cn).

Quan-bo Ge is with the Deptpartment of Electrical Automation, Shanghai Maritime University, 1550 Pudong Road, Shanghai, 200135, China (e-mail: qqbt@ hziee.edu.cn).

2. FORMULATION OF THE SYSTEM

Consider the kind of nonlinear system

$$x(k)=f\left(k-1,u(k-1),x(k-1)\right)+w(k-1), \quad (1)$$

$$z_i(k)=h_i\left(k,x(k)\right)+v_i(k),\ i=1,2,\cdots,N, \quad (2)$$

where the integer k is the sequential time index; $x(k)\in R^n$ is the state vector of the object; $u(k-1)\in R^m$ is the input vector ; Nonlinear function $f\left(k-1,u(k-1),x(k-1)\right): R^m\times R^n\to R^n$;the process noise of the system $w(k-1)\in R^{n\times 1}$ is the sequence of zero-mean white Gaussian process noise, satisfying

$$E\left\{w(k-1)\right\}=0, \quad (3)$$

$$E\left\{w(k-1)w^T(l-1)\right\}=Q(k-1)\delta_{k,l}, \quad (4)$$

where $k-1,\ l-1\geq 0$, and $Q(k-1)$ is non negative definite matrix.

There exist N sensors that observe the characteristics of object state $x(k-1)$ with the same sampling velocity $1/T$ and sampling simultaneously. The synchronous sampling of the multisensor system is shown in Fig. 1. In equation (2), $z_i(k)\in R^{p_i}$ is the observation vector aiming at object state $x(k)$ of sensor i at time k. $h_i\left(k,x(k)\right)$: $R^n\to R^{p_i}$, possesses of first order continuous partial derivative in relation to state. Vector $v_i(k)(i=1,2,\cdots,N)$ is the sequence of Gauss white noise possessing statistic property as follows:

$$E\left\{v_i(k)\right\}=0, \quad (5)$$

$$E\left\{v_i(k)v_j^T(l)\right\}=R_i(k)\delta_{i,j}\delta_{k,l}, \quad (6)$$

where $i,j=1,2,\cdots,$ N ; $k,l\geq 0$. $R_i(k)$ is positive definite matrix.

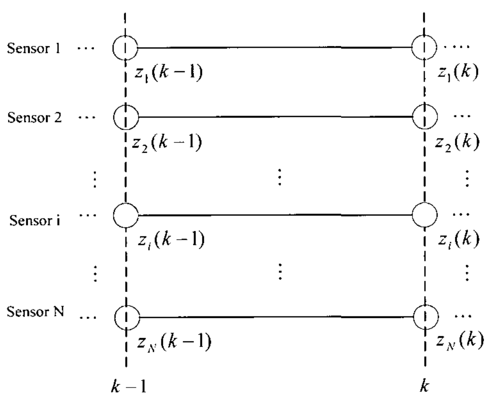


Fig. 1. Sampling of synchronous system.

The initial state $x(0)$ is a random vector, and satisfies

$$E\left\{x(0)\right\}=x_0, \quad (7)$$

$$E\left\{[x(0)-x_0][x(0)-x_0]^T\right\}=P_0, \quad (8)$$

where we suppose that $x(0)$, $w(k-1)$ and $v_i(k)$ $(i=1,\cdots,N)$ are mutually statistically independent.

3. NONLINEAR FUSION BASED ON FILTERING STEP BY STEP

3.1. Formulation of algorithm

Generally speaking, with the sensors and the observation dimension increasing, the C.C. of the central processor of nonlinear centralized data fusion algorithm (NLCDFA) will increase sharply and accordingly affect the executable speed of the central processor and reduce the practicality of the algorithm. Nonlinear distributed fusion algorithms consist of the nonlinear non-feedback fusion algorithm (NLNFFA) and nonlinear feedback fusion algorithm (NLFBFA). They realize parallel computation and improve the computational efficiency in the way of preprocessing their own measurement information by their local sensors. The defects of this kind of algorithm are that its total C.C and communicational burden are excessively heavy.

In order to reduce total C.C. and communicational complexity while maintaining high estimate accuracy, this section presents the nonlinear fusion algorithm based on filtering step by step (NLFAFSS).

Note

$$z_1^{k-1}(i)=[z_i^T(1),z_i^T(2),\cdots,z_i^T(k-1)]^T, \quad (9)$$

$$z_1^{k-1}=[z_1^{k-1}(1),z_1^{k-1}(2),\cdots,z_1^{k-1}(N)], \quad (10)$$

where $z_1^{k-1}(i)$ denotes the measurement sequence assembly of sensor i from 1 to $(k-1)$, and z_1^{k-1} denotes the measurement sequence assembly of all of sensors from 1 to $k-1$.

The basic idea of the nonlinear data fusion algorithm based on filtering step by step: if we have obtained the total estimate value $\hat{x}(k-1\,|\,k-1)$ and corresponding estimate error covariance matrix $P(k-1\,|\,k-1)$ based on the state $x(k-1)$ at time $k-1$, when all of the observations aiming at the target are obtained, we use extended Kalman filtering and all of the local observations to estimate the $x(k)$ of the object state in turn, finally we can obtain a global fusion estimate value $\hat{x}(k\,|\,k)$ based on the overall information and corresponding estimate error covariance matrix $P(k\,|\,k)$. The detailed steps are as follows:

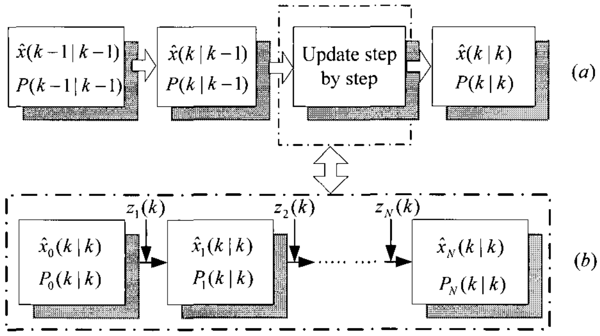


Fig. 2. The process of filtering step by step.

1) Use $\hat{x}(k-1|k-1)$ and $P(k-1|k-1)$ to compute one step predict value $\hat{x}(k|k-1)$ and predict error covariance matrix $P(k|k-1)$.

2) Use $z_i(k)$ $(i=1,2,\cdots,N)$ to update the estimate value of $x(k)$ sequentially. Accordingly, we can obtain corresponding estimate value and estimate error covariance matrix orderly

$$\begin{aligned} \hat{x}_i(k|k) &= E\{x(k)\,|\,\hat{x}_{i-1}(k|k), z_i(k)\} \\ &= E\left\{x(k)\,|\,z_1^{k-1}, z_1(k),\cdots,z_i(k)\right\}, \quad (11) \\ P_i(k|k) &= E\left\{\tilde{x}_i(k|k)\tilde{x}_i^T(k|k)\right\}, \end{aligned}$$

where $i=1,2,\cdots,N$, and

$$\hat{x}_0(k|k) = \hat{x}(k|k-1),$$

$$P_0(k|k) = P(k|k-1), \quad (12)$$

$$\tilde{x}_i(k|k) = x(k) - \hat{x}_i(k|k) \ (1 \le i \le N). \quad (13)$$

3) Finally, we can obtain the estimate value of state $x(k)$ based on z_1^k and corresponding estimate error covariance matrix

$$\hat{x}(k|k) = \hat{x}_N(k|k) = E\{x(k)\,|\,z_1^k\}, \quad (14)$$

$$P(k|k) = P_N(k|k). \quad (15)$$

The above process of filtering step by step of the nonlinear system is demonstrated by Fig. 2.

3.2. Theoretical deduction of algorithm

This section adopts the orthogonal principle to deduce the new algorithm. The concise steps are presented here.

1) One step predict estimate value $\hat{x}(k|k-1)$ and corresponding predict error covariance matrix $P(k|k-1)$ based on $\hat{x}(k-1|k-1)$ are as follows respectively

$$\hat{x}(k|k-1) = f\left(k-1, u(k-1), \hat{x}(k-1|k-1)\right), \quad (16)$$

$$\begin{aligned} P(k|k-1) &= F\left(k-1,u(k-1),\hat{x}(k-1|k-1)\right)P(k-1|k-1) \\ &\quad \times F^T\left(k-1,u(k-1),\hat{x}(k-1|k-1)\right)+Q(k-1), \end{aligned} \quad (17)$$

$$\begin{aligned} &F\left(k-1,u(k-1),\hat{x}(k-1|k-1)\right) \\ &= \left.\frac{\partial f\left(k-1,u(k-1),x(k-1)\right)}{\partial x}\right|_{x(k-1)=\hat{x}(k-1|k-1)}. \end{aligned} \quad (18)$$

2) Use $z_i(k)$ to update $\hat{x}_{i-1}(k|k)$

$$\begin{aligned} \hat{x}_i(k|k) &= \hat{x}_{i-1}(k|k) \\ &\quad + K_i(k)\left[z_i(k) - h_i\left(k, \hat{x}_{i-1}(k|k)\right)\right], \quad (19) \\ P_i(k|k) &= \left[I - K_i(k)H_i\left(k,\hat{x}_{i-1}(k|k)\right)\right] \\ &\quad \times P_{i-1}(k|k), \quad (20) \end{aligned}$$

where

$$\begin{aligned} K_i(k) &= P_{i-1}(k|k)H_i^T\left(k,\hat{x}_{i-1}(k|k)\right) \\ &\quad \times\left[H_i\left(k,\hat{x}_{i-1}(k|k)\right)P_{i-1}(k|k)\right. \quad (21) \\ &\quad \left.\times H_i^T\left(k,\hat{x}_{i-1}(k|k)\right)+R_i(k)\right]^{-1}, \end{aligned}$$

$$H_i\left(k,\hat{x}_{i-1}(k|k)\right) = \left.\frac{\partial h\left(k,x(k)\right)}{\partial x}\right|_{x(k)=\hat{x}_{i-1}(k|k)}. \quad (22)$$

3) At last, obtain the estimate value $\hat{x}(k|k)$ of state $x(k)$ based on the overall information and the corresponding estimate error covariance $P(k|k)$

$$\begin{aligned} \hat{x}(k|k) &= \hat{x}_N(k|k) \\ &= \hat{x}(k|k-1) + \sum_{i=1}^{N} K_i(k)\left[z_i(k) - h_i\left(k,\hat{x}_{i-1}(k|k)\right)\right], \end{aligned} \quad (23)$$

$$\begin{aligned} P(k|k) &= P_N(k|k) \\ &= \prod_{i=1}^{N}\left[I - K_{N+1-i}(k)H_{N+1-i}\left(k,\hat{x}_{N-i}(k|k)\right)\right]P(k|k-1), \end{aligned} \quad (24)$$

where $\hat{x}(k|k-1)$ and $P(k|k-1)$ are presented in (16) and (17) respectively.

4. ANALYSIS ON ALGORITHM

4.1. Simple discussion

The nonlinear centralized data fusion algorithm always transfers all the gathered measurement information to the central processor, meanwhile processing them. This algorithm can maintain high fusion estimate accuracy. But because the dimension of the matrix in the algorithm increases sharply with the sensors and measurement dimensions accretion, it will cause the inversed C.C. involving matrix to become quite bulky, thereby easily creating a matrix singular problem.

The basic idea of the nonlinear distributed fusion algorithm is that each local sensor pre-processes their measurement information, and then transfers the local estimate value to the central processor for fusion. This algorithm can not only realize parallel compute but can ease the computational burden of the central processor as well. However, some problems still remain as follows:

1) The total C.C. of the distributed fusion algorithm is made up of the local processor's and central processor's computation complexity. Although parallel compute can lighten the computational burden of the central processor, it just shares part of the tasks performed by the fusion central processor to the local processor, so actually the total computational burden of the fusion algorithm is not reduced.

2) The large amount of inversed computation contained in the fusion formula manifolds the C.C. of the central processor.

3) Non-feedback fusion needs to transfer local predictive and estimative information to the central processor while feedback fusion should transfer the local estimative information to the central processor and the overall estimative result to local processors. So the communicational expanses of the two distributed fusion algorithms are large.

4) The need of certain communication time in the information feedback in the feedback fusion algorithm causes predict time delay of the local processor.

As above, the current nonlinear centralized and distributed fusion algorithms can't achieve the integral best in fusion estimate accuracy, C.C. and communicational expanse etc. However, the new nonlinear algorithm based on filtering step by step avoids the disadvantages of traditional fusion algorithms by updating the estimate with the measurements in turn, and moreover it maintains the same estimate accuracy with traditional algorithms.

The following is a compare between the proposed algorithm and traditional centralized and distributed fusion algorithms from performance indexes including C.C. (total C.C. and C.C. of central processor), communicational complexity, time delay of predict and estimate accuracy, etc.

4.2. Analysis on computational complexity

By analysis, the C.C. involved in the above four fusion algorithms that are NLCDFA, NLNFFA, NLFBFA and NLFAFSS in the process of linearization of nonlinear functions is equal. For the sake of convenience and conciseness this paper will not consider this part of the calculation amount in the promise of the conclusion. As it doesn't involve data fusion when $N=1$, this paper only discusses the situation when $N \geq 2$. The computational complexity analysis in this chapter is made up of total C.C. and C.C. of the central processor.

4.2.1 Analysis on total C.C. of algorithm

We must obey the following rules when we are analyzing the computational complexity:

i) We reduce the C.C. types to addition calculation, evaluation calculation, multiplication calculation and division calculation.

ii) One evaluation calculation equals one addition calculation.

iii) Operate matrix according to the element.

All C.C. statistics in this chapter are in accordance with the above three rules. For convenience, the C.C. is analyzed in the situation $p_i = p$. The case of $p_i \neq p$ can be developed similarly.

Then the analysis results of total C.C. (except the C.C. of predict value) in the above four algorithms are shown in Table 1, where

$$M_j = N \cdot \sum_{i=1}^{j} (2j+2-i)(j-i), \quad j = n, p, Np. \quad (25)$$

Table 2 gives two practical examples of total C.C. in terms of different parameters. From Table 1 and Table 2 we can know that the total C.C. among the above four fusion algorithm obeys the following relation:

NLFAFSS <NLFBFA<NLNFFA<NLCDFA. (26)

It was noticed that the quantitative analysis doesn't include the C.C. of each algorithm's predict estimate, for nonlinear predict estimate is obtained by calculating the corresponding nonlinear function $f\left(k,u(k),x(k)\right)$. However the calculation in this part is difficult to achieve quantitatively when we don't know the analytic expression of the nonlinear function. So we analyze the C.C. in this part in a qualitative way with the result shown in Table 3.

O_p means the needed total C.C. The nonlinear function $f\left(k,u(k),x(k)\right)$ is used to compute a predict estimate value. As demonstrated in Table 2, the C.C. to compute predict estimate above fusion algorithms exists in the following relation:

NLFAFSS=NLCDFA=NLFBFA<NLNFFA. (27)

But above C.C is very low comparatively and won't influence the relation shown by equation (26). So from that we can get the relation of total C.C. among the four fusion algorithms

NLFAFSS <NLFBFA<NLNFFA<NLCDFA. (28)

4.2.2 Analysis of C.C. of central processor

In NLFAFSS and NLCDFA, all the work is finished by the central processor, while in NLNFFA and NLFBFA most of the work is finished by the local processors.

So the central processor only understates the fusion

Table 1. Computational complexity of four fusion algorithms.

Algorithms \ Type	Addition	Multiplication	Division
NLFAFSS	$(N+2)n^3+(3p+1)Nn^2$ $+(2Np^2+1)n+NM_p$	$(N+2)n^3+(3Np+1)n^2$ $+2Np(p+1)n+NM_p$	$Np(p+1)/2$
NLCDFA	$3n^3+(3Np+1)n^2$ $+(2N^2p^2+1)n+M_{Np}$	$3n^3+(3Np+1)n^2$ $+2Np(Np+1)n+M_{Np}$	$Np(Np+1)/2$
NLFBFA	$(N+2)n^3+(3Np+3N+4)n^2$ $+(2Np^2+3N-1)n$ $+NM_p+3(N+1)M_n$	$(N+2)n^3+(3Np+N+3)n^2$ $+(2Np^2+2Np+1)n$ $+NM_p+(3N+1)M_n$	$Np(p+1)/2$ $+3(N+1)n(n+1)/2$
NLNFFA	$(3N+2)n^3$ $+(3Np+4N+3)n^2$ $+(2Np^2+5N)n$ $+NM_p+2(2N+1)M_n$	$(3N+2)n^3$ $+(3Np+2N+2)n^2$ $+2Np(p+1)n+NM_p$ $+4(N+2)M_n$	$Np(p+1)/2$ $+(2N+1)n(n+1)$

Table 2. Example analysis of C.C.

(a) $N=n=p=3$

	Add	Mul	Div
NLFAFSS	630	663	18
NLCDFA	1422	1473	45
NLFBFA	981	911	90
NLNFFA	1222	1122	102

(b) $N=10, n=p=5$

	Add	Mul	Div
NLFAFSS	9005	9275	150
NLCDFA	1333280	13775	1275
NLFBFA	13045	12680	645
NLNFFA	16775	14700	780

Table 3. C.C. computing predict estimation of four algorithms.

Algorithms	NLFAFSS	NLCDFA	NLFBFA	NLNFFA
C.C.	O_p	O_p	O_p	$(N+1)O_P$

Table 4. Communicational complexity of four algorithms.

Algorithms	Communication Complexity
NLFAFSS	$\sum_{i=1}^{N}\left(p_i^2+(n+1)p_i\right)$
NLCDFA	$\sum_{i=1}^{N}\left(p_i^2+(n+1)p_i\right)$
NLFBFA	$\sum_{i=1}^{N}\left(p_i^2+(n+1)p_i\right)$
NLNFFA	$\sum_{i=1}^{N}\left(p_i^2+(n+1)p_i\right)$

work of the local estimate. Combining the total C.C. analysis, we can get the relation between the central processor's C.C. among the four fusion algorithms:

$$\text{NLFBFA}<\text{NLNFFA}<\text{NLFAFSS}<\text{NLCDFA}. \quad (29)$$

4.3. Analysis on communicational complexity

The principle of analysis on communicational complexity is that the translational unit is a numerical value, and the matrix operates in terms of the element. The analysis result is shown in Table 4.

Because $p_i \le n (i=1,2,\cdots,N)$, we can acquire the relation of communication complexity for the four fusion algorithms

$$\text{NLFAFSS}=\text{NLCDFA}<\text{NLNFFA}=\text{NLFBFA}. \quad (30)$$

4.4. Analysis on estimate accuracy

From the point of information quantity the estimate accuracy of NLFAFSS and NLCDFA should be equal for both of them in sending the primary measurement information to the central processor for processing, which will be proved by the computer simulation. The literature [4] proves that the estimate effect of centralized fusion and two kinds of distributed fusion algorithms are totally equal, providing the communication from the sensor to local nodes is at fall rate. Moreover the estimate effects of the feedback fusion algorithm and non-feedback fusion algorithm are equal, in which only the local sensor's performance in the feedback fusion algorithm is improved obviously [2]. Consequently, the estimate accuracy among every algorithm is equal, that is

$$\text{NLFAFSS}=\text{NLCDFA}=\text{NLFBFA}=\text{NLNFFA}. \quad (31)$$

4.5. Brief summary

Combining the analysis of Section 4.1 to 4.4, Table

Table 5. Comparative result of every performance index of four algorithms.

Algorithms \ Type	C.C. of central processor	Total C.C.	Time delay of predict	Communication complexity	Accuracy of estimation
NLFAFSS	Middle	Lowest	No	Low	High
NLCDFA	Highest	Highest	No	Low	High
NLFBFA	Lowest	Lower	t_c	High	High
NLNFFA	Lower	Middle	No	High	High

5 presents the comparative result about each performance index of four nonlinear fusion algorithms.

In Table 5, t_c is the needed time transmitting information from the central processor to the local processor. Because NLFBFA begins the next local filtering until total estimate information feed backs to the local processors from the central processor, so there is the time delay t_c of predict in local processors. If $t_c < T$, the performance of the feedback fusion algorithm is not affected. But if $t_c > T$ then the fusion algorithm will appear under the weight of the measurement information and time delay of the local estimate, and as such, the application ability of the feedback fusion algorithm will be affected greatly. Conversely, the other three fusion algorithms don't have such difficulty.

In Table 5 considering these performance indexes integrally, including the C.C. of the central processor, total C.C. of each algorithm, time delay of predict, communication complexity and estimate accuracy etc., we can make the conclusion that NLFAFSS is spurious to the traditional nonlinear centralized fusion algorithm and nonlinear distributed fusion algorithm.

5. COMPUTER SIMULATIONS

Next we will use the computer simulation to show the relation of estimate accuracy between the nonlinear fusion algorithm based on filtering step by step and the traditional nonlinear centralized fusion algorithm. And each simulation result is the average value of 100 Monte Carlo simulations.

5.1. Example 1

Consider the following nonlinear system:

$$x(k) = 0.1x^2(k-1) + 0.9x(k-1) + 0.02u(k-1) + w(k-1), \quad (32)$$

$$z_i(k) = H_i(k)x(k) + v_i(k), \ (i = 1,2,3). \quad (33)$$

In the nonlinear system described in (32) and (33), $x(k) \in R^1$, $Q(k)$=0.01, $H_1(k)$=0.92, $H_2(k)$=0.95, $H_3(k)$= 0.98, $R_i(k) = 0.015$ $(i = 1,2,3)$. The initial state $x_0 = 1$ and $P_0 = 10$. Then the simulation result is shown by Figs. 3 and 4. The absolute error averages of two algorithms are shown by Table 6.

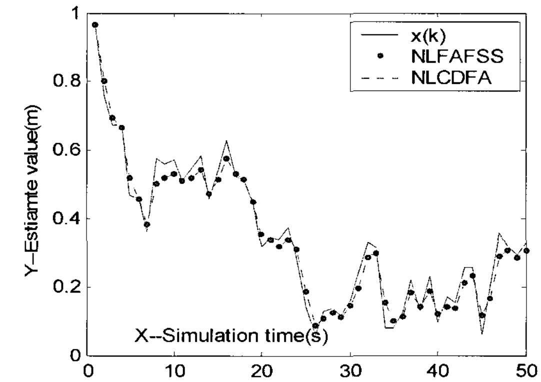


Fig. 3. Estimate curves of two algorithms in Example 1.

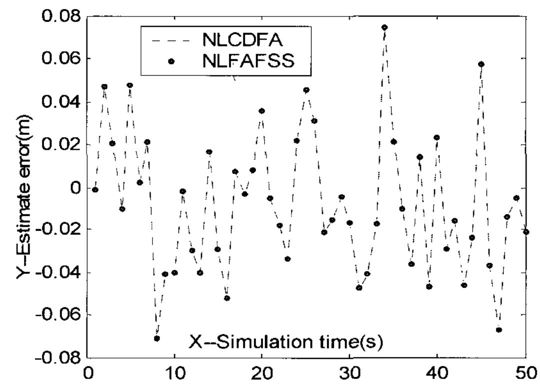


Fig. 4. Estimate error curves of two algorithms in Example 1.

Table 6. The absolute error average of two algorithms in Example 1.

Algorithm	NLFAFSS	NLCDFA
Absolute error average	0.0279	0.0279

From Fig. 3 to Fig. 4 and Table 5 we can know that the estimate accuracy aiming at the object state of NLFAFSS is equal to the estimate accuracy of NLCDFA.

Combining [2,4] we can easily see that the estimate accuracy aiming at the object state of NLFAFSS, NLCDFA, NLFBFA and NLNFFA is equal. So equation (31) comes into existence.

6. CONCLUSIONS

This paper presents a nonlinear fusion algorithm based filtering step by step on the basis of the study on data fusion methods of nonlinear multisensor dynamic systems with sampling synchronously, and puts forward the theoretical deductive process of this new algorithm and the result of computer simulation. From Table 1 to Table 5 we can know that the NLFAFSS algorithm is better than the traditional nonlinear centralized fusion algorithm and distributed fusion algorithm integrally under the comprehensive consideration of the total C.C. of the algorithm, the C.C. of the central processor, time delay of predict, communicational complexity and estimate accuracy etc. The discussions in Section 4.4 and Section 5 demonstrate that NLFAFSS, NLCDFA, NLNFFA and NLFBFA have the same estimate accuracy aiming at object state. From the amount of algorithm information obtained by these algorithms and fusion frames, the above conclusion of estimate accuracy is reasonable. But theoretic proof needs further research concerning whether the estimate accuracy of the nonlinear fusion algorithm based on filtering step by step is equal to the estimate accuracy of the nonlinear centralized fusion algorithm.

In fact the linearization of the nonlinear system will be a corresponding error, and in the practical multisensor dynamic we often encounter the question as to whether the modeling is inaccurate. These questions are not discussed in this paper. Subsequent work of the paper is to solve the above questions using iterative least square method and STF, and the solving of these questions will further improve the application ability of these fusion algorithms in a practical system.

REFERENCES

[1] C. L. Wen and D. H. Zhou, *Multiscale Estimate Theory with Application,* Tsinghua University Press Inc., Beijing, China, pp. 127-140, 2002.

[2] Y. He, G. H. Wang, D. X. Lu etc., *Multisensor Information Fusion with Application,* Electronic Industry Press, Beijing, China, 2000.

[3] C. L. Wen, B. Lü, and Q. B. Ge, "A data fusion algorithm based on filtering step by step," *Acta Electronica Sinica*, vol. 32, no. 8, pp. 1264-1267, 2004.

[4] K. C. Chang, R. K. Saha, and Y. Bar-Shalom, "On optimal track-to-track fusion," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 33, no. 4, pp. 1271-1276, 1997.

[5] X. R. Li, "Comparison of two measurement fusion methods for Kalman-Filter-Based multisensor data fusion," *IEEE Trans. on Aerospace and Electronic Systems,* vol. 37, no. 1, pp. 273-280, 2001.

[6] S. Blackman, *Multiple-target Tracking with Radar Application,* Artech House, London, pp. 350-380, 1986.

[7] B. S. Rao and H. F. Durrant-Whyte, "Fully decentralized algorithm for multisensor Kalman filtering," *IEEE Proceedings-D*, vol. 138, no. 5, pp. 413-420, 1991.

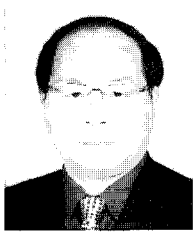
[8] Y. Bar-Shalom, X. R. Li, and T. Kirubarajam, *Estimation with Application to Tracking and Navigation,* John Wiley & Sons, Inc., New York, 2001.

[9] S. Blackman and R. Popoli, *Design and Analysis of Modem Tracking Systems,* Artech House, London, 1999.

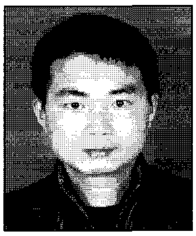
[10] L. Hong, "Centralized and distributed multi-sensor integration with uncertainties in communication networks," *IEEE Trans. on Aerospace and Electronic Systems,* vol. 27, no. 2 pp. 370-379, 1991.

[11] R. Mahler, "Random sets: Unification and computation for information fusion-a retrospective assessment," *Proc. of the 7th International Conference on Information Fusion* Sweden, pp. 1-20, 2004.

[12] S. Hedvig, "Multi-target particle filtering for the probability hypothesis density," *Proc. of the 6th International Conference on Information Fusion* Cairns, Australia, pp. 800-806, 2003.



Cheng-lin Wen is a Professor in the College of Automation at Hangzhou Dianzi University of China. He received the Ph.D. degree from the Department of Automation Control, Northwestern Polytechnical University of China in 1999. His research interests include multiscale estimation theory, multisensor information fusion technology, fault diagnostics and fault tolerant control, etc.



Quan-bo Ge is a Doctoral student in the Department of Electrical Automation at Shanghai Maritime University of China. He received the M.S. degree from the College of Computer and Information Engineering, Henan University of China in 2002. His research interests include multisensor information technology based on network, closed-loop control and multiscale estimation theory, etc.