コンピュータビジョンと
イメージメディア 121－2
(2000. 3. 23)

An Efficient Computational Scheme for Multitarget Tracking By Decentralized Cooperative Processing

LIANG CHEN and NAOYUKI TOKUDA†

Exploiting a new cooperative decentralized processing scheme of 9), 11) where multiple processors cooperate in finding a global minimum, we have developed a new computationally efficient maximum likelihood (ML)-based relaxation method for mulititarget motion analysis under a fixed networked multisensor environment. The marked improvement in computational efficiency and also in stability is achieved by replacing the well known Hungarian type assignment algorithm of 10), 12) with a much simpler sorting algorithm of $O(N \log N)$ and fusing the result with locally minimized average square errors of the relaxation. We have proved a theorem which asserts that an optimal data assignment matrix can best be given in terms of sorted bearing measuring vectors of targets. Embedding locally an optimal data association algorithm of $O(N \log N)$ into each of Gauss-Newton's downhill iteration loops, our numerical experiments were able to track as many as 8 targets and 12 targets separately within one minute by 400MHZ Dell computer with improved accuracy and efficiency, where all targets are allowed to move in variable directions at varying speeds if 4 and 6 processors are used respectively. The solution we have developed constitutes a suboptimal solution in the sense of 3), 12) because an optimal solution is embedded within part of the entire optimization problem.

Keywords: data associate problem, multitarget motion tracking, cooperative processing

分散協調処理による効率のよい多標的追跡計算法

陳　亮　　徳田尚之†

この論文では，多 CPU 協調による大域的最小値を見つける吉田等の 9), 11) 協調的分散処理方式を使って，固定多センサー網による多標的追跡解法に最大尤推定法（ML）による計算効率のよい緩和解法を開発した．局所的な緩和解の最小平均二乗誤差計算法に，これまで広く使われていたハンガリ型の割り当てアルゴリズム 10), 12) の替わりに $O(N \log N)$ という簡単な分類アルゴリズムを持ち込むことにより，計算効率だけでなく，解の安定性も大幅に改良されることを示した．本論文では，標的の最適割り当て行列が，全標的の並び替えた方位ベクトルにより与えられることを証明した．方向・速度とも任意の運動する標的数が８個，１２個の場合，それぞれ４個，６個のＣＰＵを使って 400Mhz の PC で１分以内に計算することが出来た．この解は，全体の最適化問題のなかの一部に最適解が組み込まれているという意味で，Tingiltis of 3), 12) の準最適解にあたる．

キーワード　標的割り当て問題，多標的運動追跡法，協調処理

1. Introduction

Multitarget tracking from bearings-only measurements under a distributed sensor network is a hard inverse problem[12], largely due to uncertainty in data assignment problem of tracks with respect to the multitarget objects whose time dependent motions are entirely independent and are often subject to components of independent noise. In spite of extensive research work on the subject[1),2),8)], the problem remains unsolved because it is shown that the statistic data assignment problems with more than three sensors is an NP hard problem[6),7)]. The difficulty increases perhaps beyond an exponential complexity if the number of targets, the number of sensors and the number of sampling scans increase[3),12)] with notable exceptions of the case studied by Zhou and Bose[1)], where targets can

† 宇都宮大学工学部 情報工学科
Computer Science Department, Faculty of Engineering, Utsunomiya University, Utsunomiya, 321-8505 Japan
E-mail: lchen@alfin.mine.utsunomiya-u.ac.jp, tokuda@cc.utsunomiya-u.ac.jp

be properly grouped into clusters so that the associate problem can be decomposed into smaller subproblems.

Among all the methods applied to multitarget tracking analysis, a maximum likelihood principle (ML)-based relaxation method is most commonly used. Yet, as is true with most of NP hard problems, the ML-based relaxation methods for the multitarget tracking problem encounter severe difficulties firstly by unnecessarily complex algorithms which are most easily trapped by local minima. Much progress has been reported in resolving a local minimum entrapping problem including methods of simulated annealing, and neural computing to help obtain a global minimum[3),12)].

A decentralized cooperative search[8)] offers a new approach to obtaining a global minimum in multidimensional space having multiple-local minimal functions. Unlike a centralized processing system where all of the sensor measured data are processed by a single central processor, in the decentralized cooperative approach[9),11)], all bearing measured data are processed by a multiple of local processors, where each processor estimates multitarget tracking by exchanging their intermediate estimated results obtained by other processors when needed. The centralized single processor processing scheme encounters not only an instability in computation due to the presence of widely scattered local minima over the solution space but also due to an additional cost expected in data transfer. How to find a global minimum without being trapped by local minima forms the core of the solution methods of the problem. Under multisensor and multitarget tracking, we do not need to have everything done by a single, centralized computer and the concentrated computing may be costly in the end. The decentralized cooperative search of this paper on the other hand exploits the multiple processor scheme as with the multiple sensor environment where each processor has its own smaller search space and when necessary, they cooperatively search for a global minimum over the multiply branched solution space in parallel.

In this paper we formulate the multitarget, multisensor tracking problem by the decentralized cooperative computational scheme[8)] where multiple processors cooperate. As in most of the literature on the subject, we obtain an optimal solution from bearing measurement data of multiple targets by maximizing the most popular ML(maximum likelihood) principle-based conditional probabilities. Our basic strategy in computation is the following: We first observe and then prove the theorem that an optimal minimum error assignment matrix can be most easily found if the bearings in position of the targets are known and sorted in magnitude. Encouraged by the theorem , we obtain a suboptimal solution in the sense of 3), 12) by fusing the locally optimal data assignment result with locally valid minimal least square errors until results from all processors converge. A solution is called a suboptimal one if an optimal solution is embedded within part of the entire optimization problem[3),12)]. We develop a new rapidly converging relaxation algorithm for each processor in the decentralized processing scheme by embedding the assignment matrices into the loops of Gauss-Newton downhill iterations without increasing much of the complexity. We have tested the algorithm proposed by simulating the tracks of as many as 8 and 12 targets separately using 4 and 6 processors respectively.

2. Problem Formulation and ML principle

2.1 Problem Formulation

A typical multitarget-multisensor encounter can be shown in Fig.1, where each sensor consists of a passive array of acoustic sensors and a front-end direction-of-arrival (DOA) estimator. The positions and velocities of the targets are estimated by finding the set of targets that generates bearing histories that best match the bearing measurements from sensors.

We assume that there are N targets in the surveillance region covered, and s fixed bearing-

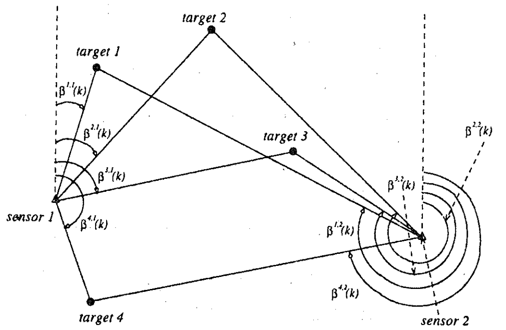

Fig. 1 Typical Target-Sensor Encounter

only sensors in the plane. The state of a target t at time index j is described by

$$X^t(j) = (r_x^t(j), r_y^t(j), v_x^t(j), v_y^t(j))'$$

where $(r_x^t(j), r_y^t(j))$ denote its Cartesian coordinates and $(v_x^t(j), v_y^t(j))$ its velocity components. The sensor i is located at the positions (r_{xs}^i, r_{ys}^i), $i = 1, 2, \cdots, s$. We assume that at time j, the measurement bearing data from target t for the sensor i can be written using the state vector $X^{t,i}$ as:

$$\beta^{t,i}(j) = \tan^{-1}\left[\frac{r_x^{t,i}(j)}{r_y^{t,i}(j)}\right] + w^{t,i}(j),$$

$$0 \le \beta^{t,i}(j) < 2\pi.$$

where $r_x^{t,i}(j) = r_x^t(j) - r_{xs}^i$, $r_y^{t,i}(j) = r_y^t(j) - r_{ys}^i$ and $w^{t,i}(j)$'s denote noise components of the ith sensor all of which are assumed to be white, Gaussian noises with mean zero and variance σ_i^2.

The measuring bearing data vector of sensor i at time j forms a N-tuple vector as

$$\beta^i(j) = (\beta^{1,i}(j), \beta^{2,i}(j), \cdots, \beta^{N,i}(j))'.$$

Because we do not have a priori knowledge on the origin of each measurement, we associate each measurement vector $\beta^i(j)$ with an $N \times N$ data assignment matrix $C^i(j)$, whose components consist of 0-1 elements only with just one 1 element appearing in each of the rows and columns. The entry $[C^i(j)]_{tm} = 1$ denotes that the mth element of the measurement vector $\beta^i(j)$ is associated with the tth target.

Our aim is to track the positions of the tracks $(r_x^t(j), r_y^t(j))$ for all targets t, $t = 1, 2, \cdots, N$, and all the time indexes j, $j = 0, 1, \cdots, K$ from the multisensor measuring data. In our formulation, we allow all the targets to move toward any directions moving at varying velocities. We try to estimate the tracks of targets just over a relatively short period of time, say, p time indexes. During such a short time period, say from time k_1 to time $k_1 - 1 + p$, we may assume that the targets can be considered as moving towards fixed directions at a fixed speed. In the following analysis, the time period during which we track the targets is in fact overlapped, thus for each target t, $(r_x^t(j), r_y^t(j))$ is always computed for several times and averaged values are used as the final result.

2.2 ML Principle

To estimate the tracks of targets during a linear period say from time k_1 to k_2 ($k_2 - k_1 + 1 = p$), we need in fact to estimate the initial state for each of the targets $X^t(j)$ at time index $j = k_1$ and the assignment matrixes $C^i(j)$ for $i = 1, 2, \cdots s$ and $j = k_1, k_1 + 1, \cdots, k_2$. Thus we seek a ML solution for this problem.

Given an estimated initial state vector for N targets by $\hat{X}(k_1) = (\hat{X}^1(k_1)', \hat{X}^2(k_1)', \cdots, \hat{X}^N(k_1)')$, we seek the cumulative bearing estimate measurement vector at time j ($k_1 \le j \le k_2$) denoted by

$$\hat{\beta}^i(j, \hat{X}_{k_1}) = \left(\hat{\beta}^{1,i}(j, \hat{X}_{k_1}), \hat{\beta}^{2,i}(j, \hat{X}_{k_1}), \cdots, \hat{\beta}^{N,i}(j, \hat{X}_{k_1})\right)'$$

where

$$\hat{\beta}^{t,i}(j, \hat{X}_{k_1}) = \tan^{-1}\left[\frac{\hat{r}_x^{t,i}(j)}{\hat{r}_y^{t,i}(j)}\right].$$

Denote $C^{p,k_1} = \{C^1(k_1), C^1(k_1+1), \cdots, C^1(k_2), C^2(k_1), \cdots, C^2(k_2), \cdots, C^s(k_1), \cdots, C^s(k_2)\}$ and $\beta^{p,k_1} = (\beta^1(k_1), \beta^1(k_1+1), \cdots, \beta^1(k_2), \beta^2(k_1), \cdots, \beta^2(k_2), \cdots, \beta^s(k_1), \cdots, \beta^s(k_2))$, the conditional likelihood of β^{p,k_1} given C^{p,k_1} and $\hat{X}_{k_1}$ can be given by

$$\Lambda\left(\beta^{p,k_1} | C^{p,k_1}, \hat{X}_{k_1}\right) = \frac{1}{c}\exp\{-\frac{1}{2}\sum_{i=1}^{s}\sum_{j=k_1}^{k_2}[C^i(j)\beta^i(j) - \hat{\beta}^i(j, \hat{X}_{k_1})]' R_i^{-1}[C^i(j)\beta^i(j) - \hat{\beta}^i(j, \hat{X}_{k_1})]\}$$

where $R_i = \sigma_i^2 I$ is the $N \times N$ diagonal noise covariance matrix at the ith sensor, and c is a constant independent of C^{p,k_1} and $\hat{X}_{k_1}$. To obtain a solution, we want to maximize Λ by minimizing the corresponding average square error ASE given by

$$ASE = \frac{1}{spN}\sum_{i=1}^{s}\sum_{j=k_1}^{k_2}[C^i(j)\beta^i(j) - \hat{\beta}^i(j,\hat{X}_{k_1})]' R_i^{-1} \cdot [C^i(j)\beta^i(j) - \hat{\beta}^i(j,\hat{X}_{k_1})$$

$$= \frac{1}{spN}\sum_{t=1}^{N}\sum_{i=1}^{s}\sum_{j=k_1}^{k_2}\left(\frac{\beta^{c(i,j,t),i} - \hat{\beta}^{t,i}}{\sigma_i}\right)^2$$

where $c(i,j,t)$ is the integer satisfying $[C^i(j)]_{tc(i,j,t)} = 1$. Write

$$E_t = \sum_{i=1}^{s}\sum_{j=k_1}^{k_2}\left(\frac{\beta^{c(i,j,t),i}(j) - \hat{\beta}^{t,i}(j)}{\sigma_i}\right)^2$$

we have

$$ASE = \frac{1}{skN}\sum_{t=1}^{N} E_t.$$

3. Cooperative Computing Algorithm

3.1 Outline of Existing Methods

Whether centralized or decentralized, the ML-based relaxation methods proceed along the following algorithm; For fixed C^{p,k_1}, ASE is minimized with respect to $\hat{X}_{k_1}$ by a Gauss-Newton relaxation method. For given $\hat{X}_{k_1}$, minimizing ASE with respect to C^{p,k_1} is equivalent to minimizing each individual term independently with respect to $C^i(j)$, and this can be done by an improved Hungarian algorithm of 10). The ML-based centralized cooperative computing relaxation methods always consists of the two steps of iteration loops.

3.1.1 Centralized Cooperative Computing

For centralized cooperative computing case, all the bearing measurements from all sensors are assumed transmitted to a global tracking center and processed by one computer. The flow of computing is shown as Table 1.

Table 1 Centralized Relaxation Method

Procedure (Centralized)
Set a suitable target initial state vector $\hat{X}_{k_1}$;
repeat
(1) Given $\hat{X}_{k_1}$, find the minimum error assignment matrices set C^{p,k_1};
(2) Given C^{p,k_1}, find the nonlinear least-square estimate $\hat{X}_{k_1}$;
until converged.

3.1.2 Decentralized Cooperative Computing

For the decentralized cooperative computing case, the system exploits several independent processors but in a cooperative manner; sensors are grouped into several clusters allowing always the overlapping of some sensor with the other clusters and each sensor cluster has its own processor. Each processor estimates multitarget tracking using bearing measurement data collected from local sensors. The processors cooperate and exchange intermediate estimates of target initial vectors with all other processors once in an iteration, then choose a best estimation to provide a least ASE. Suppose that for a sensor group $\mathcal{G}$, the indexes of the sensors in the group forms the set G, a local ASE function is defined as

$$ASE_{\mathcal{G}} = \frac{1}{|G|pN}\sum_{t=1}^{N}\sum_{i\in G}\sum_{j=k_1}^{k_2}\left(\frac{\beta^{c(i,j,t),i}(j) - \hat{\beta}^{t,i}(j)}{\sigma_i}\right)^2$$

The procedural flow of decentralized computing for the processors of each group $\mathcal{G}$ is shown as Table 2.

3.2 New Decentralized Computing

The following theorem shows that the Hungarian assignment algorithm can be replaced by an optimal $O(N \log N)$ complexity of the sorting algorithm as along as local analyses of some given sensor are concerned. **Theorem** Suppose that the cumulative bearing estimate measurement vectors for sensor i at time j ($k_1 \le j \le k_2$) is given by $\hat{\beta}^i(j,\hat{X}_{k_1}) = (\hat{\beta}^{1,i}(j,\hat{X}_{k_1}), \hat{\beta}^{2,i}(j,\hat{X}_{k_1}), \cdots, \hat{\beta}^{N,i}(j,\hat{X}_{k_1}))'$, and that the real bearing data of sensor i is given by $\beta^i(j) = (\beta^{1,i}(j), \beta^{2,i}(j), \cdots, \beta^{N,i}(j))'$. Given two $0-$

Table 2 Decentralized Relaxation Method

Procedure (Decentralized)

Set a suitable target initial state vector $\hat{X}_{k_1}$;
repeat
(1) Given $\hat{X}_{k_1}$, find a set of minimum error assignment matrices C^{p,k_1};
(2) Given C^{p,k_1}, find a nonlinear least-square estimate $\hat{X}_{k_1}$;
(3) Send the target initial state vectors to all other processors;
(4) Receive target initial state vectors sent from all other processors;
(5) With all the target initial state vector of each sensor s, calculate and choose a best estimate having a minimal E_s together with all the best initial state vectors for each sensor combining the initial state vectors of all the sensors;
until converged.

1 distinct matrices C_1 and C_2, each having just only one 1 element in every row and column of the matrices, such that $C_1 \cdot \beta^i(j) = (a_1, a_2, \cdots, a_N)'$, $C_2 \cdot \hat{\beta}^i(k, \hat{X}_{k_1}) = (b_1, b_2, \cdots, b_N)'$, where $a_1 \leq a_2 \leq \cdots \leq a_N$ and $b_1 \leq b_2 \leq \cdots \leq b_N$, then the $C^i(j) = C_2' C_1$ is a best data assignment matrix for minimizing E for fixed $\hat{X}_{k_1}$.

Proof Writing $E(C^i(j)) = [C^i(j)\beta^i(j) - \hat{\beta}^i(j, \hat{X}_{k_1})]' R_i^{-1} [C^i(j)\beta^i(j) - \hat{\beta}^i(j, \hat{X}_{k_1})]$, we have

$$ASE = \frac{1}{spN} \sum_{t=1}^{N} \sum_{j=k_1}^{k_2} E(C^i(j)).$$

Noting that $C_2' = C_2^{-1}$, we have the following relation:

$$\begin{aligned} E(C^i(j)) = & \ [C_2' C_1 \beta^i(j) - \hat{\beta}^i(j, \hat{X}_{k_1})]' R_i^{-1} \\ & \cdot [C_2' C_1 \beta^i(j) - \hat{\beta}^i(j, \hat{X}_{k_1})] \\ = & \ [C_1 \beta^i(j) - C_2 \hat{\beta}^i(j, \hat{X}_{k_1})]' R_i^{-1} \\ & \cdot [C_1 \beta^i(j) - C_2 \hat{\beta}^i(j, \hat{X}_{k_1})] \\ = & \ \frac{1}{\sigma_i} \sum_{t=1}^{N} (a_t - b_t)^2. \end{aligned}$$

To show that $C^i(j) = C_2' C_1$ is the best matrix possible, consider any other assignment matrix, say, $C_0^i(j)$, where m is the smallest integer such that $[C_2' C_0^i(j)]_{mm} \neq 1$, with $[C_2' C_0^i(j)]_{mt_1} = 1$ and $[C_2' C_0^i(j)]_{t_2 m} = 1$. It is now easy to construct a new assignment matrix $C_1^i(j)$ such that

$$[C_2' C_1^i(j)]_{jt} = \begin{cases} 1 & \textbf{if } j=t=m \\ & \quad \text{or } (j = t_2 \text{ and } t = t_1), \\ 0 & \textbf{if } (j = t_2 \text{ and } t = m) \\ & \quad \text{or } (t = t_1 \text{ and } j = m), \\ [C_2' C_0^i(j)]_{jt} & \textbf{otherwise}. \end{cases}$$

Noting that $(a_m - b_m)^2 + (a_{t_1} - b_{t_2})^2 \leq (a_m - b_{t_2})^2 + (a_{t_1} - b_m)^2$ for $t_1 > m$ and $t_2 > m$, we immediately have $E(C_1^i(j)) \leq E(C_0^i(j)$. The theorem follows immediately by mathematical induction.

The theorem shows that an optimal local matrix $C^i(j)$ can be obtained by sorting $\beta^i(j)$ and $\hat{\beta}^i(j, \hat{X}_{k_1})$ for fixed $\hat{X}_{k_1}$ which requires a remarkably efficient $O(N \log N)$ computing time, offering a new procedural flow of table 3. The complexity of computing the present assignment matrix $C^i(j)$ is far more efficient than the improved Hungarian Algorithm of 10), whose computational complexity is $O(N^3)$.

Table 3 The Algorithm for Finding A Best Assignment Matrix

Procedure (Find A Best Assignment Matrix)

Set a suitable target initial state vector $\hat{X}_{k_1}$;
(1) Sort the terms in $\beta^i(j)$ into increasing order as $\beta^{e_1,i}(j), \beta^{e_2,i}(j), \cdots, \beta^{e_N,i}(j))$
(2) Sort the terms in $\hat{\beta}^i(j, \hat{X}_{k_1})$ into increasing order as $\hat{\beta}^{f_1,i}(j, \hat{X}_{k_1}), \hat{\beta}^{f_2,i}(j, \hat{X}_{k_1}), \cdots, \hat{\beta}^{f_N,i}(j, \hat{X}_{k_1})$
(3) Construct C_1 for (1) such that $[C_1]_{e_t i} = 1$ for each $1 \leq t \leq N$;
(4) Construct C_2 for (2) such that $[C_1]_{f_t i} = 1$ for each $1 \leq t \leq N$;
(5) $C^i(j) = C_2' C_1$

This leads to the new cooperative computing algorithm given in table 4.

Note that, in step 2.1, the Guass-Newton's downhill algorithm can be found in the Appendix A of 12). The step 2.2, could be done by the procedure described in table 3.

The procedure in table 4 computes the tracks of targets in one linear period, i.e. from time k_1 to k_2. A complete algorithm computing the multitarget motion analysis over the entire time period, from time index 0 to K, will be described in table 5.

Table 4 New Decentralized Relaxation Method

Procedure (New Decentralized)

Set up a suitable initial state vector $\hat{X}_{k_1}$ for targets;
repeat

1. Given $\hat{X}_{k_1}$, find a set of minimum error assignment matrices C^{p,k_1};
2. **Repeat**
   - 2.1. Fix C^{p,k_1}, according to Gaus-Newton downhill principle, go one step with respect to $\hat{X}_{k_1}$;
   - 2.2. Fix $\hat{X}_{k_1}$, find a set of minimum error assignment matrices C^{p,k_1};

   Until converged
3. Send the initial state vectors of all targets to all other processors;
4. Receive initial state vectors of all targets sent by all other processors;
5. With initial state vectors of all the targets from each sensor s, calculate and choose a best vector giving minimal E_s together with all the best initial state vectors of each sensor from the initial state vectors of all the sensors;

until converged.

Table 5 Algorithm for Tracking Targets

1. Given a suitable initial target state vector $\hat{X}_1$;
2. For $k_1 = 1$ to $k_1 = K - p + 1$ do
   - 2.1. Use Decentralized Computing Procedure to find a best solution with respect toASE_G for a time period from k_1 to k_2;
   - 2.2. For each target t, let $\hat{v}_x^t(k_1+1) = \hat{v}_x^t(k_1)$ and $\hat{v}_y^t(k_1+1) = \hat{v}_y^t(k_1)$.
3. For each k and t, calculate the position of each object as follows:

$$r_x^t(j) = \frac{1}{\min\{K-p+1,k\} + \min\{k+1,p\} - k} \cdot \sum_{j=k-\min\{k+1,p\}+1}^{\min\{K-p+1,k\}} (\hat{r}_x(j) + (k-j)\hat{v}_x^t\Delta,$$

$$r_y^t(j) = \frac{1}{\min\{K-p+1,k\} + \min\{k+1,p\} - k} \cdot \sum_{j=k-\min\{k+1,p\}+1}^{\min\{K-p+1,k\}} (\hat{r}_y(j) + (k-j)\hat{v}_y^t\Delta,$$

where Δ is the sampling period.

4. Experimental Results

We have employed the algorithm described in the previous section to estimate the tracks of 8 targets and 12 targets separately which we allow to move in any directions at varying speed. We have used 4 and 6 sensors for 8 and 12 targets respectively. For both cases, sensors form a cluster(group) of 2 sensors and 1 processor

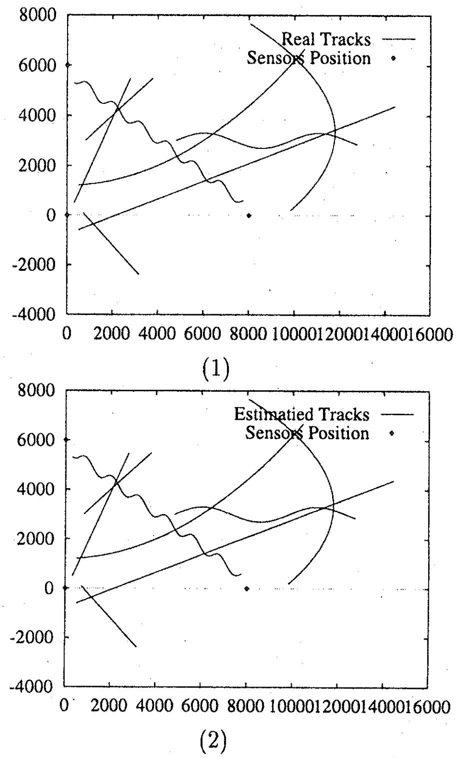

Fig. 2 The Real Tracks and The Estimated Tracks for 8 Targets

with at least one sensor being overlapped with other groups. We have allowed each sensor to belong to 2 groups. So, for the 8 targets problem, we have 4 groups, while for the 12 targets problem, we have 6 groups. We use 200 data samples where each of the continuous 4 data samples are regarded as a linear portion so that p of the above algorithm is set to 4. The standard deviations of white Gaussian noises are 0.2° for each sensor. In the simulation, we in fact only use one Dell computer driven by PentiumII 400MHz machine. The entire algorithm have converged within 25 and 42 seconds for 8 and 12 target problem respectively. The result of simulations is shown as figure 2 and figure 3. The agreement by simulation is excellent.

As far as we know, multitarget, multisensor simulations exceeding 4 targets have never been reported including the widely used annealing methods, except that of a special case when the

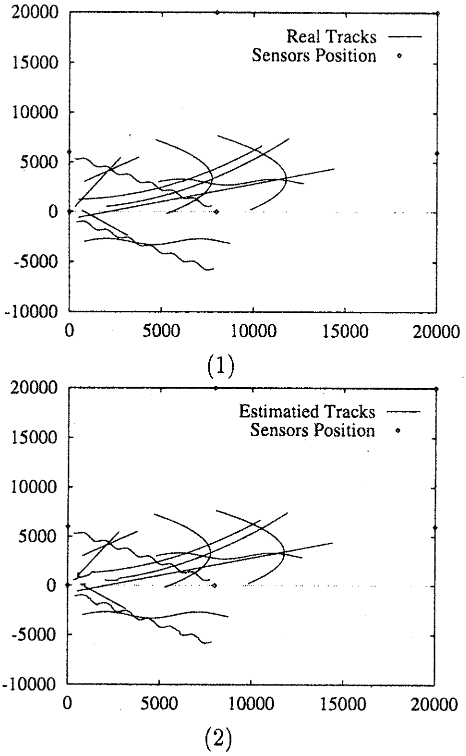

Fig. 3 The Real Tracks and The Estimated Tracks for 12 Targets

sensors are uniformly spaced in line and some hundreads of snapshots of sensor data is available within the time interval that the fixed data associate assigment can be maintained[5].

This is apparently due to the fact that it is extremely difficult to find a global minimum due to the presence of too many local minima in the solution space.

5. Conclusion

In the previous sections, we have presented a distributed cooperative computational method for the multi-target tracking problem.

Compared with the traditional centralized computing scheme which makes use of the standard or improved Hungarian assignment algorithm in data assignment problem, our simulations show that our computational method could reduce the computational complexity quite drastically in agreement with the prediction of 12).The key is the theorem we have proved in this paper which says that the sorted bearing measurement data provide an optimal data assignment matrix requiring only the sorting complexity of $O(N \log N)$.

As in all other existing works on the subject[3),8),9),12)], we have formulated the ML-based relaxation method exploiting a minimal value in the square errors between estimated and observed bearing measurements. Taking full advantage of distributed cooperative computing scheme where multiple processors cooperate to obtain a global minimum, we are able to compute 12 target motion analysis with variable directions and speeds within 1 minute of single 400MHZ Dell computer time. The solution we found seems to be *suboptimal* in the sense of Ting & Iltis 12).

Part of the paper has been presented in ISI-MADE'99[15)].

References

1) B.Zhou and N.K.Bose: Mulitarget Tracknig in Clutter: Fast Algorithm for Data Association. *IEEE Transactions on Aerospace and Electronic Systems*, 29(4):352-363, 1993.

2) K.R.Pattipati, S.Deb, Y. Bar-Shalom and R.B.Washburn Jr.: A New Relaxation Algorithm and Passive Sensor Data Association, *IEEE Transactions on Automatic Control*, 37(2):198-213, 1992.

3) A.G.Lindgren and K.F.Gong: Properties of a nonlinear estimator for detemining position and velocity from angle-of-arrival measurements, in Proc. 14th Asilomar. Conf. Circuits, Syst., and Comput., 99.394-401, Nov. 1980.

4) D.Sengapta and R.A.Iltis: Neural Solution to the multiple target tracking data association problem. IEEE Transactions on Aerospace and Electronic Systems, 25(1):96-108, 1989.

5) C.R.Rao, C.R.Sastry and B.Zhou: Tracking the Direction of Arrival of Multiple Moving Targets, IEEE Transactions on Signal Processing, 42(5):1133-1144, 1994.

6) M.R.Gray and D.S.Johnson: *Computers and Intractability: A Guide to the Theory of NP-Completeness.*

newblock San Francisco, CA:Freeman,1979.
7) C.H.Papadimitriou and K. Steiglitz: *Combinatorial Optimization: Algorithms and Complexity*. Englewood Cliffs, NJ: Prentice-Hall, 1982.
8) S.Narazaki, H.Yamamura and N.Yoshida: Strategies for Selecting communication structures in Cooperative Search. *International J. of Cooperative Information Systems*, 4(4):405-422, 1995.
9) N.Yoshida and A.Mitani: Decentralized Processing for Multitarget Motion Analysis. *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, 297-303, 1996.
10) G.Carpaneto and P.Toch: Algorithm 548:Solution of the assignment problem[H]. *ACM Trans. Math. software*, Vol.6, pp.104-111, March 1980.
11) N.Yoshida and A.Mitani: Multitarget Motion Analysis by Decentralized Cooperation (*In Japanese*). *J. Information Processing Soc. Japan*, 38(2):206-214, 1997.
12) P. Ting and A. Iltis: Multitarget Motion Analysis in a DSN, *IEEE Transactions on Systems, Man, and Cybernetics*, 20(5):1125-1139, Dept./Oct. 1991.
13) T.H.Cormen, C.E.Leiserson and R.L.Rivest: *Introduction to Algorithms*, MIT Press, Cambgidge, Mass., 1990.
14) J.E.Hopcroft and J.D.Ullman: *Introduction to Automata Theory, Language, and Computation*, Addison-Wesely Publishing Co., 1979.
15) L.Chen and N. Tokuda: Multitarget Tracking By Distributed Cooperative Processing, in *Advances in Intelligent Computing and Multimedia Systems (Proc. of Int. Symposium on Intelligent Multimedia and Distance Education, Baden Baden, Germany, August 2-7, 1999)*, M. Syed, O.Baiocchi and G.Lasker eds, pp.217-222, 1999.