2014 IEEE Congress on Evolutionary Computation (CEC)
July 6-11, 2014, Beijing, China

Distributed Wireless Sensor Scheduling for Multi-Target Tracking Based on Matrix-Coded Parallel Genetic Algorithm

Zixing Cai, Sha Wen, Lijue Liu

Abstract— **The aim of designing a sensor scheduling scheme for target tracking in wireless sensor network is to improve the tracking accuracy, balance the network energy and prolong the network lifespan. It is viewed as a multi-objective optimization problem. A modified matrix-coded parallel genetic algorithm (MPGA) is proposed in which multiple subpopulations evolve synchronously and satify the specific constraint arised from the senario of multi-target tracking that a sensor can only track just one target. Simulation results show that MPGA , compared with traditional genetic algorithm, converges to the better result with higher speed when applied in multi-target tracking in wireless sensor network. And our proposed distributed sensor scheduling scheme based on MPGA outperforms than existed schemes.**

I. Introduction

WIRELESS sensor network (WSN), as one of the most influential technology in the 21th century, is of great application value in both military and civilian fields [1]. Target tracking is one of the typical and important applications in WSN [2]. With the complication of application environment, the demand for target tracking is improved from early single-target to current multi-target [3].

In general terms, due to the limitation in storage, communication and data processing of sensor nodes in WSN, a sensor cannot track more than one target simultaneously. When a sensor is located in the detection area for multiple targets, which target should it be assigned to. That is the problem of sensor resource competition conflict [4]. To solve this problem, on the one hand, many researches focus on energy efficiency. Liu et al. [5] established a communication energy model from the perspective of the whole tracking system, which guides the allocation result into the aim of minimizing the total energy consumption for communication of the entire network. B. Krishnamachari in University of Southern California, with the goal of balancing the ratio of the remaining energy and born load of each cluster head that will prolong the network lifespan, researched the optimal task allocation strategy for cluster heads, and inducted this problem into a 0-1 nonlinear programming and objective function optimization problem [6]. On the other hand, a great deal of work is carried out for the purpose of raising tracking accuracy. Nash [7], E. Ertin [8], and X. S. Liu [9] respectively apply the trace of error covariance matrix, root-mean-square error, and maximal information gain to construct fitness functions, and then combine with different optimization methods to fulfill task allocation for sensors.

In general, tracking accuracy and energy consumption are two conflictive indexes. To achieve high tracking accuracy, more sensor nodes are needed to participate in the task of target tracking, so energy consumption of the network will be increased. Additionally in terms of network lifespan, load balance and so remaining energy balance should be considered besides the energy consumption quantity to avoid the network disconnection, energy hole, etc. resulted from the premature depletion of a few nodes.

In this paper, we study the comprehensive performance of above indexes for multi-target tracking in wireless sensor network, which is actually a multi-objective optimization problem. A modified matrix- coded parallel genetic algorithm is proposed to solve the optimization problem by synchronously evolving multiple subpopulations. The algorithm not only has fast convergence speed and high search efficiency, but also especially fits the problem of multi-target tracking. The sensor scheduling scheme based on proposed matrix-coded parallel genetic algorithm can achieve high tracking accuracy, superior energy efficiency as well as long network lifespan.

The rest of this paper is organized as follows. Tracking accuracy model and energy efficiency model are establised, and the sensor scheduling problem is formulated in Section II. In Section III, the modified matrix-coded parallel genetic algorithm is described detailedly. Simulation results are reported in Section IV. Finally, conclusions and future work are given in Section V.

II. Distributed Sensor Scheduling for Multi-target Tracking

A two-target tracking scenario in WSN is shown in Fig.1. When several targets move in monitoring region and periodically broadcast their identity through wireless signal, the sensor nodes around the targets will be waken and some of them will be selected according to a specific sensor scheduling scheme to form tasking clusters. The number of clusters is equal to the number of targets, where, for each cluster, one of the nodes is elected as cluster head and else as cluster members [10]. All sensor nodes in the cluster estimate target state and tracking error collaboratively and then inform previously a part of nodes around the position where the target will be at the next time step. As the targets move, new

Zixing Cai is with the School of Information Science and Engineering, Central South University, Changsha 410083, China (e-mail: zxcai@csu.edu.cn).

Sha Wen is with the School of Information Science and Engineering, Central South University, Changsha 410083, China (corresponding author to provide phone: +86-15973113084; e-mail: wensha0103@ 163.com).

Lijue Liu is with the School of Information Science and Engineering, Central South University, Changsha 410083, China (e-mail: ljliu@csu.edu.cn).

This work was supported in part by the National Natural Science Foundation of China under Grant 90820302 and 60805027, and the Research Fund for Doctoral Program of Higher Education under Grant 200805330005.

clusters will be formed dynamically to track them at each time step.

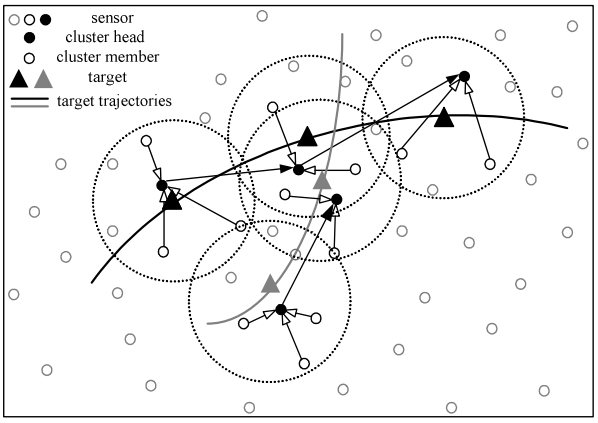

Fig.1 Two-target tracking scenario in WSN

A. Target State Estimation

Due to the limitation of the power and capability of sensor nodes, a sensor can only track one target usually and belongs to one cluster. Besides, a target is always tracked by one cluster which has no intersection with others. Unscented Kalman Filtering (UKF) algorithm is employed to estimate the state of the target.

Target state model is assumed as [11]:

$$X_{k+1} = F_k X_k + G_k W_k \tag{1}$$

where X_k is the state vector of the target at the k^{th} time step, $X_k = [x_{c,k}, x_{v,k}, y_{c,k}, y_{v,k}]^T$, $x_{c,k}$ and $y_{c,k}$ are *x*- and *y*-coordinates of the target, $x_{v,k}$ and $y_{v,k}$ are the velocities of the target along *x*- and *y*-directions at k^{th} time step, W_k is the white Gaussian process noise with covariance matrix Q_k, F_k and G_k are transition matrices of target state and process noise respectively.

The measure model is:

$$Z_k = [z_k^1, z_k^2, \cdots, z_k^{l_k}] = H(X_k) + V_k = \begin{bmatrix} h^1(X_k) \\ h^2(X_k) \\ \vdots \\ h^{l_k}(X_k) \end{bmatrix} + \begin{bmatrix} v_k^1 \\ v_k^2 \\ \vdots \\ v_k^{l_k} \end{bmatrix} \tag{2}$$

where Z_k is a l_k-dimension measurement vector, $H(\cdot)$ and V respectively denote the vector forms of $\{h^i(\cdot)\}_{i=1}^{l_k}$ and $\{v^i\}_{i=1}^{l_k}$, $h^i(\cdot)$ is the measurement function of s_i with the form of $h^i(X_k) = \sqrt{(x_k^i - x_{c,k})^2 + (y_k^i - y_{c,k})^2}$, where (x_k^i, y_k^i) is the coordinate of s_i, v^i is zero-mean Gaussian measurement noise with variance σ_i^2.

So the UKF algorithm used for target state estimation is depicted in Fig.2 [12-15]:

In this paper, the trace (sum of diagonal elements) of the updated state error covariance $P_{xx,k+1|k+1}$ is used to measure the tracking accuracy. And for N_{k+1} targets, tracking accuracy index is defined as:

$$\Phi_{k+1} = \sum_{\tau=1}^{N_{k+1}} \Phi_{k+1}^{\tau} = \sum_{\tau=1}^{N_{k+1}} \text{tr}(P_{xx,k+1|k+1}^{\tau}) \tag{3}$$

Step 1 State prediction

$$\hat{X}_{k+1|k} = F_k \hat{X}_{k|k}$$

$$P_{xx,k+1|k} = F_k P_{xx,k|k} F_k^T + G_k Q_k G_k^T$$

Step 2 Sigma points selection

$$X_{l,k+1|k} = \hat{X}_{k+1|k}, \quad l = 0$$

$$X_{l,k+1|k} = \hat{X}_{k+1|k} + \alpha(\sqrt{nP_{xx,k+1|k}})_l, \; l = 1,2,\cdots,n$$

$$X_{l,k+1|k} = \hat{X}_{k+1|k} - \alpha(\sqrt{nP_{xx,k+1|k}})_l, \; l = n+1,\cdots,2n$$

Here α is an adjusting parameter, and $(\sqrt{nP_{xx,k+1|k}})_l$ denotes the l^{th} column of the matrix square root.

Step 3 Measurement prediction

$$\hat{Z}_{k+1|k} = \sum_{l=0}^{2n} \eta_l Z_{l,k+1|k} = \sum_{l=0}^{2n} \eta_l H(X_{l,k+1|k})$$

$$P_{zz,k+1|k} = \sum_{l=0}^{2n} \eta_l (Z_{l,k+1|k} - \hat{Z}_{k+1|k})(Z_{l,k+1|k} - \hat{Z}_{k+1|k})^T + R_{k+1}$$

$$P_{xz,k+1|k} = \sum_{l=0}^{2n} \eta_l (X_{l,k+1|k} - \hat{X}_{k+1|k})(Z_{l,k+1|k} - \hat{Z}_{k+1|k})^T$$

Here $\eta_l = 1 - 1/\alpha^2$ if $l = 0$; or $\eta_l = 1/2n\alpha^2$ if $l = 1,2,\cdots,2n$.

Step 4 State update

$$\hat{X}_{k+1|k+1} = \hat{X}_{k+1|k} + K_{k+1}(Z_{k+1} - \hat{Z}_{k+1|k})$$

$$P_{xx,k+1|k+1} = P_{xx,k+1|k} - K_{k+1} P_{xz,k+1|k}$$

$$K_{k+1} = P_{xz,k+1|k}(P_{zz,k+1|k})^{-1}$$

Fig. 2. UKF algorithm for target tracking in WSN

B. Energy Model

We assume that the energy consumption by s_i for sensing data of *b* bits is $E_s(s_i) = e_s b$ and that for transmitting *b* bits to s_j is $E_t(s_i, s_j) = [e_t + e_d \| (s_i, s_j) \|^{\nu}]b$, where e_t and e_d are determined by the specifications of transmitter s_i, $\| (s_i, s_j) \|$ is the Euclidean distance function and *v* depends on the channel characteristic. The energy for receiving data of *b* bits by s_j is $E_r(s_j) = e_r b$ [16].

Based on the predicted state $\hat{X}_{k+1|k}^{\tau}$ of the τ^{th} (τ=1,2,$\cdots$, N_{k+1}) target, the set of sensor nodes around the predicted target position is $G_{k+1}^{\tau} = \{g_{k+1}^{\tau,i}\}_{i=1}^{L_{k+1}^{\tau}}$, ($\tau$ = 1,2,$\cdots$,N_{k+1}). Any combination of sensor nodes from this set can construct a candidate cluster $C_{k+1}^{'\tau} = \{c_{k+1}^{'\tau,i}\}_{i=1}^{l_{k+1}^{\tau}}$ to track target τ, where $c_{k+1}^{'\tau,i} \in G_{k+1}^{\tau}$ is a cluster node in $C_{k+1}^{'\tau}$. Assuming the cluster head is $CH_{k+1}^{'\tau}$, so the set of cluster members is $CM_{k+1}^{'\tau} = C_{k+1}^{'\tau} - CH_{k+1}^{'\tau} = \{cm_{k+1}^{'\tau,i}\}_{i=1}^{l_{k+1}^{\tau}-1}$.

The total energy consumption of tracking N_{k+1} targets at the k+1$^{\text{th}}$ time step is:

$$E_{k+1} = \sum_{\tau=1}^{N_{k+1}} E_{k+1}^{\tau} = \sum_{\tau=1}^{N_{k+1}} [E_{k+1}(CH_k^{\tau}) + E_{k+1}(CH_{k+1}^{'\tau}) + E(CM_{k+1}^{'\tau})] \quad (4)$$

where the details of $E_{k+1}(CH_k^{\tau})$, $E_{k+1}(CH_{k+1}^{'\tau})$ and $E(CM_{k+1}^{'\tau})$ can be found in [11].

The remaining energy of CH_k^{τ}, $CH_{k+1}^{'\tau}$ and $CM_{k+1}^{'\tau}$ will be calculated as follows:

$$R_{k+1}(CH_k^{\tau}) = R_k(CH_k^{\tau}) - E_{k+1}(CH_k^{\tau}) \quad (5)$$

$$R_{k+1}(cm_{k+1}^{'\tau,i}) = R_k(cm_{k+1}^{'\tau,i}) - E_{k+1}(cm_{k+1}^{'\tau,i}),\ i = 1,2,\cdots,l_{k+1}^{\tau} - 1 \quad (6)$$

$$R_{k+1}(CH_{k+1}^{'\tau}) = R_k(CH_{k+1}^{'\tau}) - E_{k+1}(CH_{k+1}^{'\tau}) \quad (7)$$

Energy balance degree can be measured by the standard deviation of remaining energy of all current and candidate sensor nodes involved in sensor scheduling:

$$\sigma_{k+1} = \mathrm{std}(\{\bigcup_{\tau=1}^{N_{k+1}}\bigcup_{i=1}^{L_{k+1}} R_{k+1}(g_{k+1}^{\tau,i}), \bigcup_{\tau=1}^{N_{k+1}} R_{k+1}(CH_k^{\tau})\}) \quad (8)$$

C. Mathematical Formulation

Based on the given energy model and tracking accuracy index , the sensor scheduling problem can be formulated as a multi-objective optimization problem [17]. At the k^{th} time step, the sensor nodes to be scheduled are determined such that:

$$\begin{cases} \min\ E_{k+1} \\ \min\ \sigma_{k+1} \\ \min\ \Phi_{k+1} \\ \text{s.t. } R_{k+1}(CH_k^{\tau}), R_{k+1}(c_{k+1}^{'\tau,i}) \ge \theta \end{cases} \quad (9)$$

The constraint that the remaining energy must be larger or equal to a threshold θ ensures the sensor nodes has enough remaining energy for further tasks.

We don't need all the Pareto solutions of above multi-objective optimization problem but the one that meets the application requirements, so the multi-objective problem is transformed to a single-objective one by weighing.

Combining the energy consumption and remaining energy balance, a comprehensive index of energy efficiency is constructed as:

$$\varphi_{k+1} = (1-\omega_1)E_{k+1} + \omega_1\sigma_{k+1} \quad (10)$$

where $\omega_1 \in [0,1]$ is a weight parameter used for the tradeoff between energy consumption quantity and balance degree.

Then the multi-objective optimization problem can be transformed into a single-objective one as following:

$$J_{k+1} = (1-\omega_2)\gamma\varphi_{k+1} + \omega_2\Phi_{k+1} \quad (11)$$

where $\omega_2 \in [0,1]$ is another weight parameter, and γ is a matching factor that makes the value of energy efficiency and estimation accuracy in the same magnitude.

In the application of target tracking, the tracking accuracy should be put in the first place, followed by the energy efficiency, and for the two sub-objectives of energy efficiency, their priority is almost equal. So we have $\omega_1 = 0.5$ and $\omega_2 > 0.5$. From the analysis and through the adjustment in extensive experiments, we have $\omega_2 = 0.7$ so that energy efficiency achieve optimal under the premise of satisfactory tracking accuracy.

III. Modified Matrix-Coded Parallel Genetic Algorithm

Sensor scheduling is indeed to find the best tasking clusters for tracking the targets. The solution of this problem can be represented as a matrix. The common coding methods of Genetic Algorithm are binary coding, real coding, ordinal coding, etc, which have their own application fields, but fail to solve the problem whose best solution is a matrix. So we present a matrix-coded Genetic Algorithm and carry out genetic operations with matrixes as individuals in a population.

A. Matrix Coding

Definition 1 (Matrix Coding) Suppose the k^{th} generation of population is denoted as $P_k = \{A_1, A_2, \cdots, A_s\}$, where s is the number of individuals in the population, and A_i (i=1,2, ⋯,s) is the i^{th} individual and can be represented as an m×n matrix

$$A_i = \begin{bmatrix} a_{11}^i & a_{12}^i & a_{1n}^i \\ a_{21}^i & a_{22}^i & a_{2n}^i \\ & & \\ a_{m1}^i & a_{m2}^i & a_{mn}^i \end{bmatrix}$$

Then A_i is called matrix chromosome, and a_{uv}^i (u=1,2, ⋯,m; v=1,2, ⋯,n) is a gene in this chromosome.

In this paper, binary coding method is employed. Particularly, we introduce a special code '×' to denote a forbidden gene which means the corresponding sensor cannot detect the target. In initialization, a forbidden gene is assigned the code '×' which will not change in any genetic operations, and deservedly it won't be selected to tracking the corresponding target.

When N_{k+1} targets move in the monitoring region at the k+1th time step with L_{k+1} sensor nodes in their sensing area. A chromosome which represents N_{k+1} candidate clusters is encoded as a matrix with L_{k+1} rows and N_{k+1} columns. For example as Fig.3, the chromosome is encoded to a 6×3 matrix which corresponds to 3 targets (T_1,T_2,T_3) and 6 sensor nodes (s_1, s_2, ⋯, s_6) in their sensing area. A gene $ge_{k+1}^{\mu,\tau}$ in the μ^{th} row (μ=1,2,⋯,6) and τ^{th} column (τ=1,2,3) is interpreted as:

$$ge_{k+1}^{\mu,\tau} = \begin{cases} 1, \text{if } s_\mu \text{ is selected for } T_\tau \\ 0, \text{if } s_\mu \text{ is not selected for } T_\tau \\ \times, \text{if } s_\mu \text{ is forbidden} \end{cases}$$

In order to deal with the constraint that a sensor node is just assigned to one target, all genes in each row should follow:

$$\sum_{\tau=1}^{N_{k+1}} ge_{k+1}^{\mu,\tau} \le 1 \quad (12)$$

Note that the forbidden gene '×' is considered as "0" when calculate the sum of a line of the chromosome. In Fig.3, s_1, s_2, s_3, s_4 can detect T_1 but only s_1 and s_3 is selected to track the target. There are two sensor nodes, s_1 and s_2 detect more than one target, but due to the constraint in (12), each of them is assigned to only one target (s_1 for T_1, and s_2 for T_2).

	T_1	T_2	T_3
s_1	1	0	0
s_2	0	1	×
s_3	1	×	×
s_4	0	×	×
s_5	×	1	×
s_6	×	×	1

Fig. 3. Representation of a chromosome

B. Selection Operation

With the reciprocal of the function in (11) as the fitness function, calculate the fitness values of all the matrix chromosomes in population, and use wheel selection method to select best individuals to generate next generation. The probability of each individual to be selected is equal to the ratio between its fitness value and the sum of all fitness values.

Elitist preservation strategy is applied in our selection process to ensure that the obtained best individual will not be damaged by further genetic operation and so converge to the global optimal solution. After crossover and mutation, if the obtained best fitness value in new generation is less than the one in previous generation, the best individual in previous generation is duplicated and replaces the worst individual in new generation. For example, the parent generation is:

$$P = \{A_1, A_2, \cdots A_i, \cdots A_j, \cdots A_s\}$$

The new generation after crossover and mutation is:

$$Q = \{B_1, B_2, \cdots B_m, \cdots B_n, \cdots B_s\}$$

where A_i and A_j , B_m and B_n are the best and worst individual in parent and new generations, respectively. If fitness(B_m) < fitness (A_m), we have the offspring generation as following:

$$O = \{B_1, B_2, \cdots B_m, \cdots A_i, \cdots B_s\}$$

C. Crossover Operation

In order to ensure the offspring chromosomes always satisfy the constraint in (12), i.e. these offspring are sure to be feasible solutions of the problem, a row (column) of genes between two parent chromosomes are exchanged with a specified probability P_c. An example is shown as Fig.4.

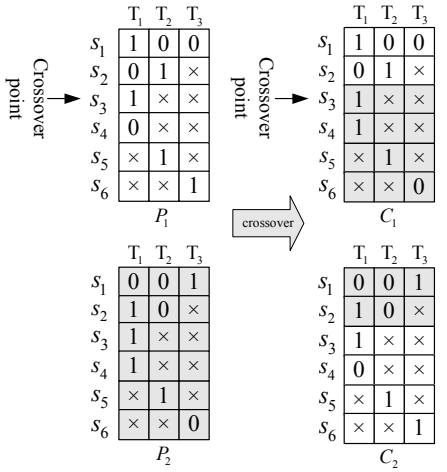

Fig. 4. Crossover of chromosomes

Local tournament selection is used to choose the better individuals between parent and offspring generations as following:

$$O_1 = \max\{f_{itness}(P_1), f_{itness}(C_1)\}$$
$$O_2 = \max\{f_{itness}(P_2), f_{itness}(C_2)\}$$

D. Mutation

Two random numbers μ and τ is generated in range of [1, N_{k+1}] and [1, L_{k+1}], respectively. The gene $ge_{k+1}^{\mu,\tau}$ switches from 0 to 1 or from 1 to 0 if two conditions are both satisfied: (a) $ge_{k+1}^{\mu,\tau}$ is not a forbidden gene; (b) $\sum_{l=1,l\neq\tau}^{N_{k+1}} ge_{k+1}^{\mu,l} = 0$, which guarantees the mutated chromosome still feasible for problem.

E. Migration

Migration operation makes this Genetic algorithm executed parallelly.

Every several generations, the best individual of each subpopulation migrates to other subpopulations in following topology as shown in Fig.5.

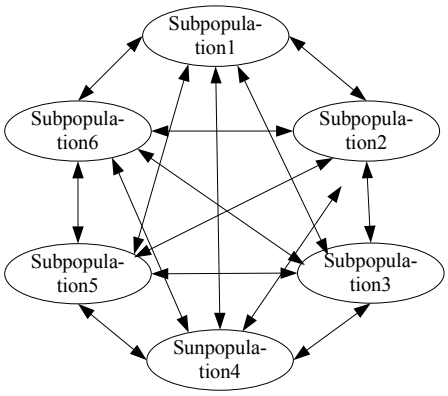

Fig.5 Topology for Migration

In this way the excellent individuals are delivered among subpopulations, which also ensure the evolution mechanism to reach the global optimal solution rapidly and efficiently.

IV. Simulation Results

In order to verify the performance of our proposed energy-efficient sensor scheduling scheme based on Matrix-coded Parallel Genetic Algorithm (EESMPGA), 100 Monte Carlo experiments are conducted with following three other sensor scheduling schemes as comparison.

- Energy-saving scheduling based on matrix-coded parallel genetic algorithm (ESSMPGA): to minimize the quantity of energy consumption, i.e. $\omega_1 = 0$ in (10).
- Energy-efficient scheduling based on traditional genetic algorithm (EESBTGA): to apply traditional Genetic Algorithm for searching cluster nodes.
- Energy-efficient scheduling based on Extended Kalman Filtering (EKF) algorithm (EESBEKF): to employ EKF for target state estimation [16].

A. Simulation Setup

Suppose that 50 sensor nodes are disseminated in a 100×

100 rectangle monitoring area where two targets T1 and T2 move inside with variable speed. They move linearly in first 20 seconds, T1 towards positive direction of x-coordinate while T2 towards negative, then bend in next 25 seconds, and finally stay stationary over rest simulation time, as shown as Fig.6. The initial values of target state and error covariance are set to:

$$X_0^1 = [2 \;\; 1 \;\; 4.5 \;\; 1];$$

$$X_0^2 = [92 \;\; 1 \;\; 99 \;\; 1];$$

$$P_{xx,0|0}^1 = P_{xx,0|0}^2 = 0.01 \times diag(1 \;\; 1 \;\; 1 \;\; 1)$$

The initial energy of all sensor nodes is set to be 0.5J, except the 4th, 11th, 12th, 44th nodes to be 0.1J. Other parameters and their values are list in Table 1.

TABLE I
SIMULATION PARAMETERS

Parameter	Denotation	Value
α	Adjusting parameter in UKF	0.5
ω_1	Weight in energy efficiency model	0.5
ω_2	Weight in single- objective optimization	0.3
γ	Energy matching factor	10
λ	Adjusting parameter of population size	5
P_c	Crossover probability	0.8
P_m	Mutation probability	0.01
κ	Number of generations in migration	5

B. *Results and analysis*

Fig.6 shows real and estimated trajectories with abovementioned four schemes, and Fig.7 shows the tracking error. We find that the four schemes all performed well during linear motion stage of the targets, while during the curve motion stage just the estimated trajectory with EESBEKF diverged obviously from the real trajectory, which proves that the presented scheduling scheme based on UKF has better estimation performance in contrast to EKF-based scheduling scheme [16] for nonlinear random motion.

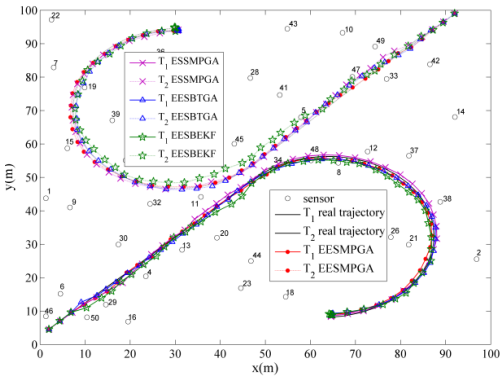

Fig. 6. Real and estimated trajectories of two targets

The remaining energy of all sensor nodes with four schemes is illustrated in Fig.8. The balance level of remaining energy with EESMPGA and ESSMPGA is best, followed by EESBTGA, and worst is ESSMPGA. The 4th, 11th, 12th, 44th nodes that have less initial energy were rarely scheduled with EESMPGA and EESBEKF scheme. As a comparison, even though considering remaining energy balance, EESBTGA still selected 4th, 11th, 12th, 44th nodes sometimes due to the fact that traditional genetic algorithm based on binary string-coded cannot efficiently converge to the global optimal solution.

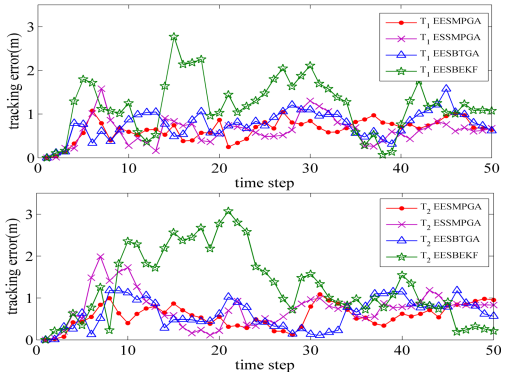

Fig. 7. Tracking errors of two targets

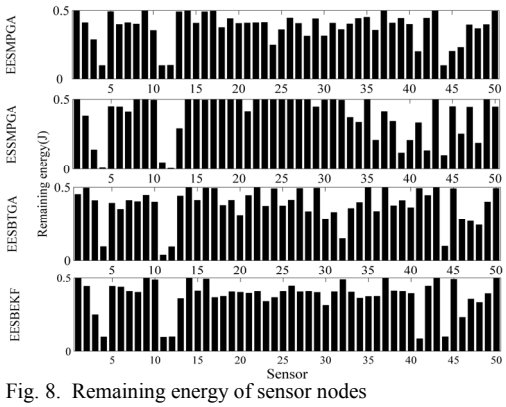

Fig. 8. Remaining energy of sensor nodes

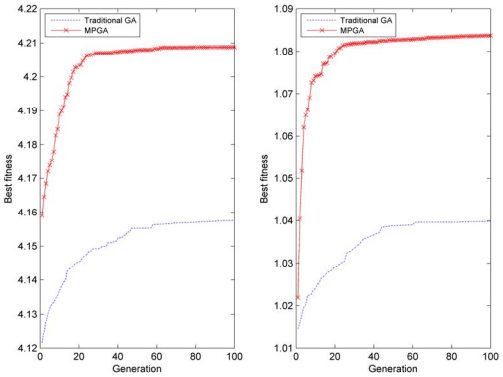

Fig. 9. Best fitness of traditional GA and MPGA

Tab.2 lists some indexes about energy efficiency and network lifespan (time period from the beginning of tracking to the occurrence of first node that doesn't have sufficient energy to perform tracking task). ESSMPGA consumed least energy but lived shortest lifespan because of not taking energy balance into account. And with EESBTGA, the energy balance degree is worse than that of rest two schemes because traditional genetic algorithm cannot always find the optimal tasking clusters to track all targets. However, the proposed EESMPGA not only considers remaining energy balance but also applies MPGA, so it takes on superior

performance in every aspect above.

A comparison of best fitness value (reciprocal of objective function value) with EESMPGA and EESBTGA at two arbitrary time steps (16^{th} and 36^{th} time step) is shown in Fig.9. We can see that, with the same fitness function, EESMPGA always converged to a higher fitness value with faster speed than EESBTGA, which fully proves the superior performance of MPGA.

TABLE Ⅱ
ENERGY EFFICIENCY AND NETWORK LIFESPAN

Scheduling schemes	Energy efficiency index	Total energy consumption/J	Least remaining energy/J	Network lifespan / time steps
EESMPGA	0.1114	4.4119	0.1000	≥50
ESSMPGA	0.1534	4.1107	0.0058	14
EESBTGA	0.1193	4.4149	0.0462	≥50
EESBEKF	0.1124	4.4284	0.1000	≥50

V. CONCLUSIONS

In this paper, we have presented a distributed sensor scheduling scheme based on matrix-coded parallel genetic algorithm for multi-target tracking. After establishing the tracking accuracy model, energy consumption model and energy balance model, the problem of sensor scheduling is formulated as a multi-objective optimization problem, whose best solution is found by weighing all the sub-objectives. A modified matrix-coded parallel genetic algorithm (MPGA) is developed. It allows multiple subpopulations to evolve independently and synchronously, and satisfies the constraint that a sensor can only track just one target. Simulation results have shown that the proposed distributed sensor scheduling scheme based on MPGA reduces the energy consumption and prolongs the network lifespan while improving the tracking accuracy.

For the multi-objective optimization problem proposed in this paper, the method of weighing is used, which is under the premise that the bias introduced by weights is appropriate for user demand. As future work, other advanced multi-objective optimization algorithms will be considered to find all the Pareto solutions of our problem.

REFERENCES

[1] L.M. Sun, J.Z. Li and Y. Chen, Wireless Sensor Network, Beijing: Tsinghua University Press: 2005, pp.3-4.

[2] Y.G. Xiao, "Target tracking algorithm in smart sensor reconnaissance Network," China University of Technology, HeFei, China, 2007,pp. 46-47.

[3] J. Liu, M. Chu, and J.E. Reich, "Multitarget Tracking in Distributed Sensor Networks," *IEEE Signal Processing Magn*, vol. 5, no. 24, pp. 36-47, 2007.

[4] H.H. Li, M. Liu, Y. Shen, and D.L. Qiao, "Research on Task Allocation Technique for Multi-Target Tracking in Wireless Sensor Network," *Mechatronics and Automation, 2007. ICMA 2007. International Conference,* pp.360,365, 5-8 Aug. 2007

[5] M. Liu, H.H. Li, Y. Shen, J.F. Fan and S.N Huang, "Elastic neural network method for multi-target tracking task allocation in wireless sensor network," *Computers and Mathematics with Applications*, vol. 57, pp. 1822_1828, Jun. 2009

[6] B. Krishnamachari, S.B. Wicker and R. Béjar, "On the complexity of distributed self-configuration in wireless networks", *Telecommunication Systems*, vol. 22, pp. 33-59, Jan. 2003

[7] Nash J.M., "Optimal allocation of tracking resources", *In Proceedings of the IEEE Conference on Decision and Control*, Piscataway, NJ, USA, 1977, pp. 1177~1180

[8] Ertin E., Fisher J. and Potte L., "Maximum mutual information principle for dynamic sensor query problems", In *Proceedings of the 2nd International Workshop on Information Processing in Sensor Networks*. Berlin, 2003, pp. 56~58

[9] Liu X.S., Zhou L. and Du X.Y., "A Method of Sensor Management Based on Target Priority and Information Gain", *ACTA Electronica SINICA*, vol.33, no. 9, pp. 1683-1687, Sep. 2005

[10] J.X. Chen, C.Z. Zang, W. Liang and H.B. Yu, "A Dynamic Task Allocation Scheme for Wireless Sensor Networks," *Information and Control*, vol. 35, no. 2, pp. 189-199, Apr. 2006

[11] X.Q. Hu, B.G. Xu, S. Wen and Y.G. Liu, "Distributed energy balanced optimal cluster*ing scheme," Journal of South China University of Technology (Natural Science Edition),* vol. 40, no. 8, pp. 1-8, Aug. 2012

[12] Y. Zhou, J.X. Li and D.L. Wang, "Unscented Kalman Filtering based quantized innovation fusion for target tracking in WSN with feedback," In *Proceedings of the Eighth International Conference on Machine Learning and Cybernetics (ICMLC),* Baoding, Hebei, China, 2009, pp. 1457-1463.

[13] L.Q. Zhang, Q. Cheng, Y.G. Wang and S. Zeadally, "A Novel Distributed Sensor Positioning System Using the Dual of Target Tracking," *IEEE Trans. Compu.*, vol. 57, no. 2, pp. 246-260, Feb. 2008

[14] W.F. Leven, A.D. Lanteman, "Unscented Kalman Filters for Multiple Target Tracking With Symmetric Measurement Equations". *IEEE Trans. Automat. Contr.*, vol.54, no.2, pp. 370- 375, Feb. 2009

[15] R. Zhan and J. Wan, "Iterated Unscented Kalman Filter for Passive Target Tracking". *IEEE Trans. Aero. Elec. Sys.*, vol. 43, no.3, pp. 1155 -1163, Jul. 2007

[16] J.Y. Lin, W.D. Xiao, F.L. Lewis, and L.H. Xie, "Energy-Efficient Distributed Adaptive Multisensor Scheduling for Target Tracking in Wireless Sensor Networks," *IEEE Trans. Instrum. Meas.*, vol. 58, no. 6, pp. 1886-1896, Jun. 2009

[17] A.T. Han, J.F. Li, H. Peng, Z.Q. Chen, and Z. Liao, "Multiobjective Optimization Control in Monitoring System for Facility Agriculture Based on Wireless Sensor Network," In *Proceedings of 2011 International Conference on Computer Science and Automation Engineering (CSAE)*, Shanghai, China, 2011, pp.545-554

[18] W.P. Chang and X.J. Luo, "Application of Double Matrix Hybrid Coded Genetic Algorithm in Unit Commitment". *Electricity Distribution, 2008. CICED 2008. China International Conference*, Guangzhou, China, Dec. 2008, pp.1,6, 10-13

[19] Chipperfield A.J., Genetic Algorithm Toolbox. version 1.2 [DB/OL]. Available: http://www.geatbx.com/ea_matlab.html