Journal of Southeast University (English Edition) Vol. 23, No. 2, pp. 255 - 260 June 2007 ISSN 1003—7985

Energy-efficient mechanism based on ACO for the coverage problem in sensor networks

Huang Ru[1] Zhu Jie[1] Xu Guanghui[2]

([1]Department of Electronic Engineering, Shanghai Jiaotong University, Shanghai 200240, China)

([2]Institute of Communications Engineering, PLA University of Science and Technology, Nanjing 210016, China)

Abstract: An energy-efficient heuristic mechanism is presented to obtain the optimal solution for the coverage problem in sensor networks. The mechanism can ensure that all targets are fully covered corresponding to their levels of importance at minimum cost, and the ant colony optimization algorithm (ACO) is adopted to achieve the above metrics. Based on the novel design of heuristic factors, artificial ants can adaptively detect the energy status and coverage ability of sensor networks via local information. By introducing the evaluation function to global pheromone updating rule, the pheromone trail on the best solution is greatly enhanced, so that the convergence process of the algorithm is speed up. Finally, the optimal solution with a higher coverage-efficiency and a longer lifetime is obtained.

Key words: sensor networks; coverage problem; ant colony optimization (ACO); energy-efficiency

The coverage problem is a fundamental mechanism in sensor networks, which represents the quality of service (QoS) provided by sensor networks. A typical coverage problem can be modeled as a set covering problem (SCP), which is an NP-hard combinatorial optimization problem. Most previous work on coverage problems has focused on the energy efficient algorithm design to prolong the lifetime of sensor networks and meet various coverage requirements. In Ref. [1], the coverage problem was modeled as finding the maximal number of disjoint set covers with each set completely covering all targets. Ref. [2] presented a randomized and coordinated sleep algorithm which can maintain network coverage by using low duty-cycles and each sensor independently sleeping under a certain probability. In Ref. [3], the authors proposed an energy conserving protocol to extending the lifetime of sensor networks by maintaining only a set of sensors in working mode and ensuring coverage and connectivity with high probability. In Refs. [4 - 5], the authors addressed a k-coverage maintenance algorithm. Each target in a given region can be covered by at least k distinct sensors and each sensor decides whether it is in redundant status by checking the state of its sensing perimeter. The ant colony optimization (ACO) is a constructive meta-heuristic optimization method, which can be seen as probabilistic construction heuristics that generate solutions iteratively by considering the accumulated past search experience and the heuristic information on the instance under solution[6]. ACO performs excellently in solving most combinatorial optimization problems and it has been successfully applied to TSP, QAP, VRP, etc. [7-8]. It is also a positive feedback search algorithm[9], the optimal solution will accumulate a larger amount of pheromone and in turn will be selected more often in the future.

1 Model Description

1.1 Network model

Given randomly and densely deployed a set of φ sensors, $\boldsymbol{N}=\{n_1, \dots, n_i, \dots, n_\varphi\}$ on the sensor area. Each sensor $n_i \in \boldsymbol{N}$ with a known location obtained by GPS has sensing range ξ_s^i and communication radius ξ_c^i ($\xi_c^i \geqslant 2\xi_s^i$). The set of neighbors of n_i is denoted as $O^{(i)}$. Let $\boldsymbol{R}=\{r_1, \dots, r_i, \dots, r_\theta\}$ be a set of static targets in the monitor field. We define the threshold vector $\boldsymbol{\vartheta}_{\text{thresh}}=\{r_{\text{thresh}}^{(1)}, \dots, r_{\text{thresh}}^{(i)}, \dots, r_{\text{thresh}}^{(\theta)}\}_{1\times\theta}$ ($r_{\text{thresh}}^{(i)} \in [0, 1]$) to denote the different levels of importance of all targets in $\boldsymbol{R}$. Each element of $\boldsymbol{\vartheta}_{\text{thresh}}$ is the predefined critical coverage probability of a corresponding target, which reflects the requirement of coverage accuracy to the above target and that the value of each element in $\boldsymbol{\vartheta}_{\text{thresh}}$ is proportional to the level of importance of the corresponding target. Then we define the norm of vector $\boldsymbol{\vartheta}_{\text{thresh}}$ as Eq. (1), which can evaluate the average coverage accuracy of the monitor field.

Received 2007-01-17.

Foundation item: The Natural Science Foundation of Jiangsu Province (No. BK2005409).

Biographies: Huang Ru (1975—), male, graduate; Zhu Jie (corresponding author), male, doctor, professor, zhujie@ sjtu. edu. cn.

$$\|\boldsymbol{\vartheta}_{\text{thresh}}\|_1 = \frac{\sum_{j=1}^{\theta} r_{\text{thresh}}^{(j)}}{\theta} \tag{1}$$

1.2 Cache and information packet model of sensor

It is assumed that each sensor $n_i \in \boldsymbol{N}$ maintains a cache$_i$, which comprises $\text{tab}_i^{\text{cover}}$, $\text{tab}_i^{\text{energy}}$ and $\text{tab}_i^{\text{prob}}$. At the initial stage of the algorithm, n_i receives a packet$_i$ from $O^{(i)}$ and stores it into cache$_i$, whose structure is shown as follows: $\text{tab}_i^{\text{cover}} = \{\ldots, \boldsymbol{\sigma}_t, \ldots\}$, where $\boldsymbol{\sigma}_t$ denotes the coverage-vector of senor $n_t \in O^{(i)}$; $\text{tab}_i^{\text{prob}} = \{\ldots, \boldsymbol{P}_t, \ldots\}$, where $\boldsymbol{P}_t$ denotes the probability-coverage-vector of $n_t \in O^{(i)}$; $\text{tab}_i^{\text{energy}} = \{\ldots, e_t, \ldots\}$, where e_t denotes the normalized remaining energy level of $n_t \in O^{(i)}$. The n_i's packet$_i$ is shown as

ID$_i$	Location$_i$	e_i	$\boldsymbol{P}_i$	$\boldsymbol{\sigma}_i$

where ID$_i$ is the identity of n_i, Location$_i$ is n_i's location information obtained by GPS, e_i is the normalized remaining energy level of n_i, $\boldsymbol{P}_i$ is n_i's probability vector, and $\boldsymbol{\sigma}_i$ is n_i's coverage vector.

1.3 Memory model of ant

Each ant$_k$, $k = 1, 2, \ldots, m$, has a memory $\boldsymbol{M}^k$ that is used to store the information about the sensors visited, i.e. $\boldsymbol{L}^{(k)}$, and the targets covered, i.e. $\boldsymbol{\Theta}^{(k)}$. The structure of ant$_k$'s memory is shown as

ID$^{(k)}$	$\boldsymbol{L}^{(k)}$	$\boldsymbol{\Theta}^{(k)}$

where ID$^{(k)}$ represents distinguishing the ant$_k$ from the data packet. $\boldsymbol{L}^{(k)}$ is a tabu list, which records the solution built by ant$_k$, and is shown as

$$\boldsymbol{L}^{(k)} = \{n_1 X_1^{(k)}, \ldots, n_i X_i^{(k)}, \ldots, n_\varphi X_\varphi^{(k)}\}$$

where $X_i^{(k)} = \begin{cases} 1 & n_i \in \boldsymbol{L}^{(k)} \\ 0 & n_i \notin \boldsymbol{L}^{(k)} \end{cases}$. $\boldsymbol{s}^{(k)} = \{s_1^{(k)}, \ldots, s_i^{(k)}, \ldots, s_\varphi^{(k)}\}_{1\times\varphi}$ is defined as solution vector corresponding to $\boldsymbol{L}^{(k)}$, $s_i^{(k)} = X_i^{(k)}$. $\boldsymbol{\Theta}^{(k)}$ is shown as $\boldsymbol{\Theta}^{(k)} = \{\Theta_1^{(k)}, \ldots, \Theta_i^{(k)}, \ldots, \Theta_\theta^{(k)}\}_{1\times\theta}$ and the i-th element of $\boldsymbol{\Theta}^{(k)}$ denotes the times that target r_i has been covered by $\boldsymbol{L}^{(k)}$. Then, the Boolean vector $\boldsymbol{\Theta}_{\text{norm}}^{(k)}$ is defined as

$$\boldsymbol{\Theta}_{\text{norm}}^{(k)} = \{g(\Theta_1^{(k)}), \ldots, g(\Theta_i^{(k)}), \ldots, g(\Theta_\theta^{(k)})\}_{1\times\theta} \tag{2}$$

where $g(x)$ is a sign function,

$$g(x) = \begin{cases} 1 & x \geqslant 0 \\ 0 & x < 0 \end{cases}$$

$$g(\Theta_i^{(k)}) = \begin{cases} 1 & \text{if } r_i \text{ has been covered by } \boldsymbol{L}^{(k)}, r_i \in \boldsymbol{R} \\ 0 & \text{otherwise} \end{cases}$$

With the element sequence, $\boldsymbol{\Theta}_{\text{norm}}^{(k)}$ and vector $\mathbf{1}_{1\times\theta}^{\mathrm{T}} = \{1, 1, \ldots, 1\}_{1\times\theta}$ are processed bit by bit with the exclusive-OR or operation. Then the Boolean vector $\widehat{\boldsymbol{\Theta}_{\text{norm}}^{(k)}}$ is the result, which denotes those targets uncovered by the current partial solution $\boldsymbol{L}^{(k)}$. The current feasible neighborhood of ant$_k$ at n_i ($i = 1, 2, \ldots, \varphi$) is defined as $A_K = \widehat{\boldsymbol{L}^{(k)}} \cap O^{(i)}$, where $\widehat{\boldsymbol{L}^{(k)}} = \boldsymbol{N} - \boldsymbol{L}^{(k)}$.

1.4 IP model of optimization coverage mechanism

The goal of the optimization coverage mechanism (OCM) is to find a subset in sensor field $\boldsymbol{N}$ with a minimal evaluation function value; meanwhile, the whole monitor field $\boldsymbol{R}$ is completely covered by the subset and each target in $\boldsymbol{R}$ can be covered according to corresponding coverage accuracy. The integer programming (IP) formulation of the OCM is given as

$$\min f(\boldsymbol{s}^{(*)})$$

$$\text{s.t.} \sum_{j=1}^{\theta} \sum_{i=1}^{\varphi} y_j^{(i)} s_i^{(*)} \geqslant \theta \qquad y_j^{(i)} \in \boldsymbol{\sigma}_i,\ s_i^{(*)} \in \boldsymbol{s}^{(*)} \tag{3}$$

where $f(\cdot)$ is the evaluation function (defined in Eq. (9)); $\boldsymbol{s}^{(*)}$ is the optimal solution vector obtained by the algorithm and $s_i^{(*)}$ is the element of $\boldsymbol{s}^{(*)}$; $y_j^{(i)}$ is the element of coverage vector $\boldsymbol{\sigma}_i$ (defined in Eq. (7)); θ is the number of targets in $\boldsymbol{R}$.

2 Design of Algorithm

In this paper, artificial ants adopt stochastic solution construction procedures that probabilistically build a solution by iteratively adding sensors to partial solutions while considering heuristic information and pheromone trails.

2.1 Design of heuristic factor

First, we design the objective function, which is based on the idea that the unit cost of covering an additional target is minimal. So, the structure of the objective function is constructed as

$$f_{ij} = \frac{c_{ij}}{v_{ij}}$$

where v_{ij} is the coverage factor, which is the number of additional targets covered when adding sensor n_j to $\boldsymbol{L}^{(k)}$ when ant$_k$ at n_i; c_{ij} is the cost factor, which is the cost of selecting n_j as the next hop node when ant$_k$ at n_i. Then, we design the structure of heuristic factor η_{ij} based on the inverse of the objective function.

$$\eta_{ij} = \frac{1}{f_{ij}} \tag{4}$$

So, with the heuristic desirability, the artificial ant tends to choose a neighbor sensor, which covers the maximal number of uncovered targets and has a higher remaining-energy-level (i.e., less cost) as its next hop sensor.

2.1.1 Design of cost factor c_{ij}

For each sensor $n_i \in \boldsymbol{N}$, we define the remaining-energy-factor as

$$e_i = \frac{E_i^{(\mathrm{lef})}}{E_i^{(\mathrm{ini})}}$$

where $E_i^{(\mathrm{lef})}$ is the remaining energy of n_i, and $E_i^{(\mathrm{ini})}$ is the initial energy in n_i. Then, we construct the structure of the cost factor as

$$c_{ij} = \frac{1}{\tilde{\omega}_{ij}}$$

where $\tilde{\omega}_{ij} = e_j / \sum_{n_t \in O^{(i)}} e_t (n_t, n_j \in O^{(i)})$.

2.1.2 Design of coverage factor v_{ij}

We assume that each $n_i \in \boldsymbol{N}$ probabilistically covers $r_j \in \boldsymbol{R}$. The probability $p_j^{(i)}$ reflects how well the r_j is monitored by n_i, not considering the influence by n_i's neighbors, which also denotes the sensing ability decreasing with the increase of the distance.

$$p_j^{(i)} = \begin{cases} \dfrac{1}{[1 + \partial_1 d_j^{(i)}]^{\partial_2}} & d_j^{(i)} \leqslant \xi_s^i \\ 0 & d_j^{(i)} > \xi_s^i \end{cases} \tag{5}$$

where ξ_s^i is the sensing radius of sensor n_i; $d_j^{(i)}$ is the Euclidean distance between n_i and r_j; ∂_1 and ∂_2 are parameters reflecting the physical features of a sensor[10].

We define $z_j^{(i)}$ as

$$z_j^{(i)} = 1 - [1 - p_j^{(i)}] \prod_{n_t \in O^{(i)}} [1 - p_j^{(t)}] \tag{6}$$

It is the probability of sensor n_i to cover target r_j, taking the influence by n_i's neighbors into consideration. Each n_i can calculate the vector $\boldsymbol{Z}^{(i)} = \{z_1^{(i)}, \ldots, z_j^{(i)}, \ldots, z_\theta^{(i)}\}_{1\times\theta}$ according to Eq. (6), based on the information from $\mathrm{tab}_i^{\mathrm{prob}}$. If $z_j^{(i)} \geqslant r_{\mathrm{thresh}}^{(j)}$, then r_j can be covered by n_i with corresponding coverage accuracy, otherwise not. So the coverage vector of n_i is defined as

$$\boldsymbol{\sigma}_i = \{g(z_1^{(i)} - r_{\mathrm{thresh}}^{(1)}), \ldots, g(z_j^{(i)} - r_{\mathrm{thresh}}^{(j)}), \ldots, g(z_\theta^{(i)} - r_{\mathrm{thresh}}^{(\theta)})\}_{1\times\theta} \tag{7}$$

$$g(z_j^{(i)} - r_{\mathrm{thresh}}^{(j)}) = \begin{cases} 1 & r_j \text{ covered by } n_i \\ 0 & \text{otherwise} \end{cases}$$

Then, the coverage matrix $\boldsymbol{B}_{\varphi\times\theta}$ is shown as

$$\boldsymbol{B}_{\varphi\times\theta} = \begin{bmatrix} \cdots \\ \boldsymbol{\sigma}_j \\ \cdots \end{bmatrix}_{\varphi\times\theta}$$

where the j-th row of matrix $\boldsymbol{B}_{\varphi\times\theta}$ is n_j's coverage vector, i. e. $\boldsymbol{\sigma}_j (n_j \in O^{(i)})$. When ant_k reaches n_i, it takes the status information of each $n_j \in O^{(i)}$ in n_i's cache and the stored information in memory $\boldsymbol{M}^k$. Then it does the following operation of matrix cross products and obtains the vector $\boldsymbol{\Lambda}$, which denotes the coverage information of sensors in $O^{(i)}$.

$$[\widetilde{\boldsymbol{\Theta}_{\mathrm{norm}}^{(k)}}]_{1\times\theta} \times \boldsymbol{B}_{\theta\times\varphi}^{\mathrm{T}} = \boldsymbol{\Lambda}_{1\times\varphi} \tag{8}$$

where $\boldsymbol{\Lambda}_{1\times\varphi} = \{\ldots, a_j^{(i)}, \ldots\}_{1\times\varphi}$, $a_j^{(i)}$ represents the number of the targets that are not covered by $\boldsymbol{L}^{(k)}$, but covered by n_j. If $a_j^{(i)} = \max\{\ldots, a_t^{(i)}, \ldots\}$ $(n_t, n_j \in O^{(i)})$, then n_j can be chosen by ant_k as the next hop. So we define the coverage factor as $v_{ij} = a_j^{(i)} / \theta$.

2.2 Design of evaluation function of energy efficiency to solution

Function f is designed to evaluate the quality of the solution based on the energy-efficiency. We define $f_1(\boldsymbol{s}^{(k)})$ as the function to evaluate the average-remaining-energy-level of solution $\boldsymbol{L}^{(k)}$ and $f_2(\boldsymbol{s}^{(k)})$ as the function to evaluate the energy-balance-level to the same solution. The value of $f_1(\boldsymbol{s}^{(k)})$ is in direct proportion to the average-remaining-energy-level and the value of $f_2(\boldsymbol{s}^{(k)})$ is in inverse proportion to the energy-balance-level.

$$f_1(\boldsymbol{s}^{(k)}) = \sum_{i=1}^{\varphi} e_i s_i^{(k)} \Big/ \sum_{i=1}^{\varphi} s_i^{(k)}$$

$$f_2(\boldsymbol{s}^{(k)}) = \frac{\sqrt{\sum_{n_i \in L^{(k)}(t)} \left(e_i s_i^{(k)} - \sum_{i=1}^{\varphi} e_i s_i^{(k)} \Big/ \sum_{i=1}^{\varphi} s_i^{(k)} \right)^2}}{\sum_{i=1}^{\varphi} e_i s_i^{(k)} \Big/ \sum_{i=1}^{\varphi} s_i^{(k)}}$$

$$e_i = \frac{E_i^{(\mathrm{lef})}}{E_i^{(\mathrm{ini})}}$$

$$e_i \in [0, 1],\ s_i^{(k)} \in \boldsymbol{s}^{(k)},\ k \in [1, m],\ i \in [1, \varphi]$$

The relation between the range of evaluation function value and its corresponding level of energy-efficiency is shown in Tab. 1.

Tab. 1 Value of evaluation function and corresponding level

$f_1(\boldsymbol{s}^{(k)})$	Remaining-energy-level of $\boldsymbol{s}^{(k)}$	$f_2(\boldsymbol{s}^{(k)})$	Energy-balance-level of $\boldsymbol{s}^{(k)}$
0	Low	0	High
(0, 1)	Middle	(0, 1)	Middle
1	High	1	Low

The evaluation function of energy efficiency to the solution is designed based on f_1 and f_2.

$$f(\boldsymbol{s}^{(k)}) = [(1-\delta) f_1(\boldsymbol{s}^{(k)}) + \delta(1 - f_2(\boldsymbol{s}^{(k)}))]^{-1} \tag{9}$$

where δ is the scaling factor $(0 \leqslant \delta \leqslant 1)$ and the value of $f(\boldsymbol{s}^{(k)})$ is inversely proportional to the quality of solution $\boldsymbol{s}^{(k)}$. When $\delta = 1$, the evaluation criterion is based on the energy-balance-level of the solution. When $\delta = 0$, the evaluation criterion is based on the average-remaining-energy-level of the solution. When $\delta \in (0, 1)$, the evaluation criterion is based on the trade off between the above two criteria.

2.3 Global pheromone updating rule

After each $\mathrm{ant}_k (k = 1, 2, \ldots, m)$ has completed the construction of a solution, we consider the 2-tuple set $(\boldsymbol{S}, f)$, where $\boldsymbol{S}$ is the set of candidate solutions built by ants and f is the evaluation function, stated in Eq. (9), which assigns to each candidate solution vector $\boldsymbol{s}^{(k)} \in \boldsymbol{S}$

and measures the quality of it. $\boldsymbol{S} = \{\boldsymbol{s}^{(1)}, \dots, \boldsymbol{s}^{(k)}, \dots, \boldsymbol{s}^{(m)}\}$, where $\boldsymbol{s}^{(k)} = \{s_1^{(k)}, \dots, s_i^{(k)}, \dots, s_\varphi^{(k)}\}_{1\times\varphi}$ is the solution vector corresponding to $\boldsymbol{L}^{(k)}$. And $f_k = f(\boldsymbol{s}^{(k)})$ is the evaluation function value of $\boldsymbol{s}^{(k)}$. Sort all these values in an ascending order into set $\boldsymbol{f} = \{f_e^{gb}, \dots, f_k^{w}, \dots, f_s^{m-1}\}$ ($w = 1, 2, \dots, m_o - 1$, $m_o < m$). The superscript of f_k^w is the rank of ant_k and we adopt the extension elitist strategy[11] to sort the ranks of ants according to the quality of solutions. f_e^{gb} is the minimal value in set $\boldsymbol{f}$. So $\boldsymbol{s}^{(e)}$ is the iteration-best solution vector constructed by the iteration-best ant_e. The other ant with rank w is the w-th best ant. Only the iteration-best ant and $(m_o - 1)$ best ants are allowed to deposit pheromones and the intensity of pheromone is based on their ranks. The w-th best ant contributes to the pheromone update with a weight given by $\max\{0, m_o - w\}$, while ant_e reinforces the pheromone trails with weight m_o. So, at each iteration, the pheromones are enhanced at the sensors belonging to the iteration-best solution. The global pheromone updating is applied by the following rule:

$$\tau_i(t+1) = (1-\rho)\tau_i(t) + \sum_{w=1}^{m_o-1}(m_o - w)\Delta\tau_i^w(t) + m_o\Delta\tau_i^{gb}(t)$$

where

$$\Delta\tau_i^{gb}(t) = \begin{cases} Q/f_e^{gb} & \text{if } n_i \in \boldsymbol{L}^{(e)} \\ 0 & \text{if } n_i \notin \boldsymbol{L}^{(e)} \end{cases}$$

$$\Delta\tau_i^{w}(t) = \begin{cases} Q/f_k^{w} & \text{if } n_i \in \boldsymbol{L}^{(k)}, k \neq e \\ 0 & \text{if } n_i \notin \boldsymbol{L}^{(k)} \end{cases}$$

where Q is the positive constant; ρ is the pheromone decay rate.

Based on the updating rule, the global pheromone deposited is a function of the solution quality and will guide the search of the remaining ants in the future.

2.4 Implementation of the algorithm

At the initial stage of the algorithm, Sink broadcasts information frame_i ($i = 1, 2, \dots, \varphi$) (shown as follows) to each sensor $n_i \in \boldsymbol{N}$.

ID_i	Instruction_i	$\boldsymbol{D}_i$	$\boldsymbol{\vartheta}_{\text{thresh}}$

where ID_i is the identity of sensor n_i and Instruction_i is the instruction of constructing an artificial ant. If $\text{Instruction}_i = 1$, an artificial ant in n_i will be created, otherwise not. $\boldsymbol{D}_i$ is the distance-vector, $\boldsymbol{D}_i = \{d_1^{(i)}, \dots, d_j^{(i)}, \dots, d_\theta^{(i)}\}_{1\times\theta}$, $d_j^{(i)}$ is the Euclidean distance between n_i and r_j. $\boldsymbol{\vartheta}_{\text{thresh}}$ is the threshold vector.

When receiving the frame_i, n_i compares the ID in the information frame with its own ID_i. If they are matched, n_i accepts it and calculates $\boldsymbol{P}_i$ according to Eq. (5). Then, n_i broadcasts packet_i to $O^{(i)}$. After that, each sensor calculates its coverage vector according to Eq. (7) and broadcasts the updated information packet to sensors within its neighborhood. The initial stage of the algorithm is over when each sensor has kept the information from its neighbors. Then, each ant starts with an initial state to construct a solution. It is assumed that ant_k ($k = 1, 2, \dots, m$) which launched from n_t begins to build its own solution $\boldsymbol{L}^{(k)}$. First, ant_k respectively adds n_t into $\boldsymbol{L}^{(k)}$ and $\boldsymbol{\sigma}_t$ into $\boldsymbol{\Theta}^{(k)}$ and obtains the initial values of them. When ant_k reaches n_i, it adds n_i to its current partial solution $\boldsymbol{L}^{(k)}$; i. e., $\boldsymbol{L}_{\text{new}}^{(k)} = \boldsymbol{L}_{\text{old}}^{(k)} \cup \{n_i\}$, sets $X_i^{(k)}$ to 1 and takes $\boldsymbol{\Theta}^{(k)}$ from its memory $\boldsymbol{M}^k$, and makes it do a vector addition operation with $\boldsymbol{\sigma}_i$ to get the targets set covered by the current partial solution $\boldsymbol{L}^{(k)}$ ($\boldsymbol{\Theta}_{\text{new}}^{(k)} = \boldsymbol{\Theta}_{\text{old}}^{(k)} + \boldsymbol{\sigma}_i$). After that, ant_k calculates the value of heuristic factor η_{ij} according to Eq. (4), and selects the next unvisited sensor in A_K with $p_{ij}^k = \tau_j^\alpha \eta_{ik}^\beta / \sum_{r\in A_k} \tau_r^\alpha \eta_{ik}^\beta$ [12]. This process is repeated until the termination condition

$$\sum_{i=1}^{\theta} \boldsymbol{\Theta}_{\text{norm}}^{(k)} = \theta \tag{10}$$

is satisfied, which means that all targets are fully covered. Then, the construction procedures of ant_k stops. The algorithm is terminated when no improved solution is found for a given number of iterations. And the algorithm returns the best solution found; i. e., $\boldsymbol{L}^{(*)}$, which can meet the coverage requirements, meanwhile, has good energy efficiency. Then let the other redundant sensors in $\boldsymbol{N} - \boldsymbol{L}^{(*)}$ be turned into sleeping mode.

3 Simulation Results

The following simulations evaluate the performance of the OCM. And the parameters in the ACO are set as follows: $\alpha = 0.7$, $\beta = 2.3$, $\rho = 0.2$, $m = 25$, $Q = 1\,000$. Firstly, we define the coverage ratio as Eq. (11), which denotes the coverage ability of sensor networks.

$$r_{\text{opt}} = \sum_{i=1}^{\theta} \frac{\Theta_i^{(*)}}{\theta} \tag{11}$$

where $\Theta_i^{(*)} \in \boldsymbol{\Theta}^{(*)}$ denotes the times of target r_i covered by $\boldsymbol{L}^{(*)}$ and θ is the number of total targets. When the coverage ratio reaches the stable value, it means the whole monitor field is completely covered by the solution. So, the coverage lifetime is defined as the period of keeping a stable coverage ratio and the solution with a long coverage lifetime is of high quality. We adjust the level of each target by setting the corresponding element value in $\boldsymbol{\vartheta}_{\text{thresh}}$ and evaluate the average coverage accuracy of the monitor field according to the value $\|\boldsymbol{\vartheta}_{\text{thresh}}\|_1$.

In Fig. 1, according to different average coverage accuracies, the variety in coverage ratio associated with the process of solution construction is analyzed. Given $\|\boldsymbol{\vartheta}_{\text{thresh}}^{(2)}\|_1 > \|\boldsymbol{\vartheta}_{\text{thresh}}^{(1)}\|_1$, when $\|\boldsymbol{\vartheta}_{\text{thresh}}^{(2)}\|_1$ increases with about 0. 1% more than $\|\boldsymbol{\vartheta}_{\text{thresh}}^{(1)}\|_1$, the former coverage ratio at stable status will decrease with about 10% less than the latter one, and the number of iterations with $\|\boldsymbol{\vartheta}_{\text{thresh}}^{(2)}\|_1$ is about 47% more than the one with $\|\boldsymbol{\vartheta}_{\text{thresh}}^{(1)}\|_1$, which denotes that when the requirements of the average coverage accuracy to the monitor field is improved, the coverage ratio at stable status decreases accordingly, and it takes a longer time to reach stable coverage status. From Fig. 2, based on the two evaluation criteria of solution quality, i. e. f_1 and f_2, we make the comparison between the performance of the algorithm according to the number of rounds versus the normalized coverage ratio. In the simulation, we can know that with the increase of rounds, when $\delta = 0$, because of the balanced energy distribution, the composed sensors in solution run out of energy and die at almost the same time. When $\delta = 1$, the composed sensors in the solution run out of energy at different times. Because of the unbalanced energy consumption and because of the high remaining energy level of the solution, some composed nodes have a longer lifetime. And we can see that the coverage lifetime with $\delta = 0$ is longer than the one with $\delta = 1$, so the solution of $\delta = 0$ is of relatively high quality. In Fig. 3, we analyze the changing of the coverage ratio with different node densities according to different threshold vectors ($\|\boldsymbol{\vartheta}_{\text{thresh}}^{(2)}\|_1 > \|\boldsymbol{\vartheta}_{\text{thresh}}^{(1)}\|_1$). We can see that the coverage ratio in both cases ascend and get close to the same coverage ratio with the increase of node density, which denotes that the coverage ratio with different average coverage accuracies tends to the same value in a high node density area.

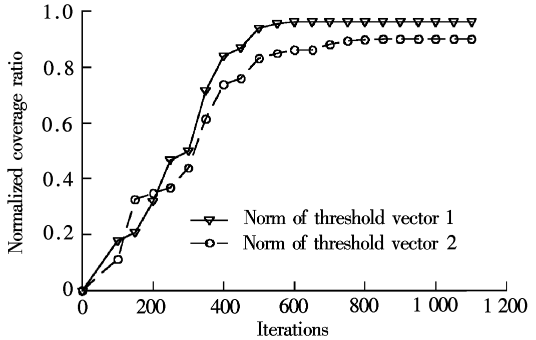

Fig. 1 Normalized coverage ratio vs. iterations

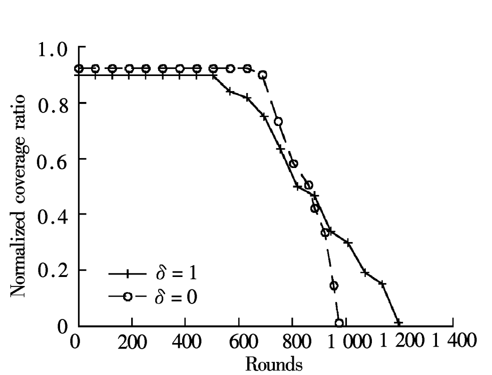

Fig. 2 Normalized coverage ratio vs. rounds with two evaluation criteria

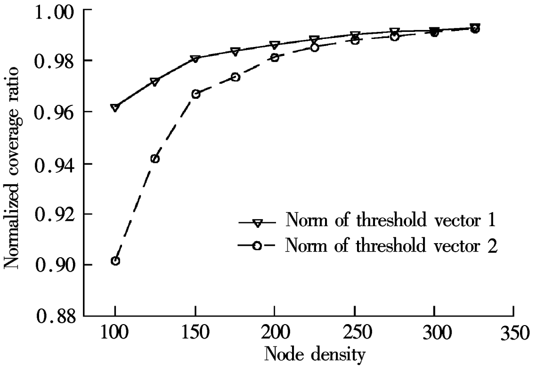

Fig. 3 Normalized coverage ratio vs. node density

4 Conclusion

An optimal mechanism with the characteristics of energy-efficiency and coverage-efficiency is presented for coverage problems in sensor networks. The mechanism based on ACO can guarantee that all targets are completely covered; meanwhile, the accuracy degree of coverage to each target is proportional to its level of importance. So the important targets are more reliably covered, on the premise that each target is covered by at least one active sensor. By introducing the coverage factor and the cost factor to the construction of the heuristic factor, as well as an evaluation function to the global pheromone updating rule, the artificial ants are equipped with the ability to be aware of the coverage-status of the monitor field and the energy-status of the sensor area. By iteration methods, the solutions' construction procedures are based on the energy status and the coverage ability of sensors by local information. The pheromone trails on the optimal solution are also greatly reinforced. Finally, the optimal solution with energy-efficiency and coverage-efficiency is obtained in polynomial time and the robustness of the solution is greatly improved.

References

[1] Cardei L M, Du D Z. Improving wireless sensor network lifetime through power aware organization [J]. *Journal of Wireless Networks*, 2005, **11**(3): 333 − 340.

[2] Hsin C, Liu M. Network coverage using low duty-cycled sensors: random and coordinated sleep algorithms [C]// *Proceedings of the Third International Symposium on Information Processing in Sensor Networks*. California, USA, 2004: 433 − 442.

[3] Ye F, Zhong G, Lu S, et al. PEAS: a robust energy conserving protocol for long-lived sensor networks [C]// *Proceedings of the 10th IEEE International Conference on Network Protocols*. Paris, France, 2002: 200 − 201.

[4] Huang C, Tseng Y. The coverage problem in a wireless sensor network [J]. *Journal of Mobile Networks and Applications*, 2005, **10**(4): 519 − 528.

[5] Abrams Z, Goel A, Plotkin S. Set *k*-cover algorithms for energy efficient monitoring in wireless sensor networks [C]// *Proceedings of the Third International Symposium on Information Processing in Sensor Networks*. California, USA, 2004: 424 − 432.

[6] Parpinelli R S, Lopes H S, Freitas A. Data mining with an ant colony optimization algorithm [J]. *IEEE Trans on Evolutionary Computation*, 2002, **6**(4): 321 − 332.

[7] Dorigo M, Maniezzo V, Colorni A. The ant system: optimization by a colony of cooperating agents [J]. *IEEE Transactions on Systems, Man and Cybernetics: Part B*, 1996, **26**(1): 29 − 41.

[8] Dorigo M, Gambardella L M. Ant colony system: a cooperative learning approach to the traveling salesman problem [J]. *IEEE Trans on Evolutionary Computation*, 1997, **1**(1): 53 − 66.

[9] Zecchin Aaron C, Simpson Angus R, Maier Holger R. Parametric study for an ant algorithm applied to water distribution system optimization [J]. *IEEE Trans on Evolutionary Computation*, 2005, **9**(2): 175 − 191.

[10] Meguerdichian S, Koushanfar F. Exposure in wireless adhoc sensor networks [C]// *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking*. Rome, Italy, 2001: 139 − 150.

[11] Bullnheimer B, Hartl R F, Strauss C. A new rank-based version of the ant system: a computational study [J]. *Central European Journal for Operations Research and Economics*, 1999, **7**(1): 25 − 38.

[12] Dorigo M, Caro G D. Ant colony optimization: a new metaheuristic [C]// *Proceedings of IEEE Congress on Evolutionary Computation*. Washington, DC, USA, 1999: 1470 − 1477.

基于蚁群优化算法的传感器网络能量有效性覆盖机制

黄　如[1]　　朱　杰[1]　　徐光辉[2]

([1] 上海交通大学电子工程系, 上海 200240)
([2] 解放军理工大学通信工程学院, 南京 210016)

摘要: 提出了一种解决无线传感器网络覆盖问题的能量有效性启发式机制. 该机制在节能的前提下,实现了对目标监控区域的完全覆盖,且覆盖精度与目标的重要性级别成正比关系. 机制的实现运用了蚁群优化算法,算法的设计过程采用了新颖的启发式因子构造方法和基于评价函数的全局信息素更新规则,由此,人工蚂蚁被赋予了对目标监控区域的覆盖状况和对传感器网络区域能量状况的自适应感知能力,并通过增加优化解集中节点上的信息素量,加速求取最优解的收敛过程. 最后,蚁群在迭代优化的基础上构建出解决无线传感器网络覆盖问题的健壮优化解,该优化解能够在能量有效性的基础上具备良好的覆盖有效性和较长的生命周期.

关键词: 传感器网络;覆盖问题;蚁群优化;能量有效性

中图分类号: TP393.01