Information Sciences 300 (2015) 100–113

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

An endocrine cooperative particle swarm optimization algorithm for routing recovery problem of wireless sensor networks with multiple mobile sinks

Yi-Fan Hu [a,b,c,d], Yong-Sheng Ding [a,c,*], Li-Hong Ren [a,c], Kuang-Rong Hao [a,c], Hua Han [a,e]

[a] Engineering Research Center of Digitized Textile & Fashion Technology, Ministry of Education, Donghua University, 201620 Shanghai, PR China
[b] Shandong Computer Science Center (National Supercomputer Center in Jinan), 250014 Jinan, PR China
[c] College of Information Sciences and Technology, Donghua University, 201620 Shanghai, PR China
[d] Shandong Provincial Key Laboratory of Computer Network, 250014 Jinan, PR China
[e] College of Electronic and Electrical Engineering, Shanghai University of Engineering Science, 201620 Shanghai, PR China

ARTICLE INFO

Article history:
Received 9 December 2011
Received in revised form 4 November 2014
Accepted 19 November 2014
Available online 27 December 2014

Keywords:
Wireless sensor networks
Multiple mobile sinks
Routing recovery
PSO algorithm
Endocrine cooperative mechanism

ABSTRACT

In the wireless sensor networks with multiple mobile sinks, the movement of sinks or failure of sensor nodes may lead to the breakage of the existing routes. In most routing protocols, the query packets are broadcasted to repair a broken path from source node to sink, which cause significant communication overhead in terms of both energy and delay. In order to repair broken path with lower communication overhead, we propose an efficient routing recovery protocol with endocrine cooperative particle swarm optimization algorithm (ECPSOA) to establish and optimize the alternative path. In the ECPSOA, mutation direction of the particle is determined by multi-swarm evolution equation, and its diversity is enriched by the endocrine mechanism, which can enhance the capacity of global search and improve the speed of convergence and accuracy of the algorithm. By using this method, the alternative path from source nodes to the sink with the optimal QoS parameters can be selected. Simulation results show that our routing protocol significantly improves the robustness and adapts to rapid topological changes with multiple mobile sinks, while efficiently reducing the communication overhead and the energy consumption.

1. Introduction

The Internet-of-Things (IoTs) are regarded as the extension of current Internet to the real world of physical objects [5]. The basic idea of IoT is pervasively providing us with a variety of things or objects, such as radio frequency identification (RFID) tags, sensors, wireless sensor networks (WSNs), and mobile phones, which are able to interact and cooperate with each other to realize the tasks of communication, computation, and service. An important direction of the IoTs is to facilitate suitable WSNs technologies based on an efficient standard protocol to support the network of things [10,6].

Several data dissemination protocols have been proposed for the WSNs with a static sink [21,26,29]. For example, the directed diffusion approach [11] assumes that each sink needs to periodically flood its location information through the

* Corresponding author at: College of Information Sciences and Technology, Donghua University, 201620 Shanghai, PR China. Tel.: +86 21 67792301; fax: +86 21 67792353.

E-mail address: ysding@dhu.edu.cn (Y.-S. Ding).

http://dx.doi.org/10.1016/j.ins.2014.11.052
0020-0255/

sensor field. The procedure sets up a gradient from the sensor nodes to the sink, so that each sensor is aware of the sink location for sending future events and measurements. However, the static sink may limit the network lifetime as the 1-hop neighbors of the sink, which is the bottleneck of the network. The WSNs with mobile sink is a powerful solution to take advantage of short-range transmission. When the energy of the sensors around one sink is exhausted, the sink can move to a new location in an area with richer sensors' energy [20]. This approach can balance the energy consumption and prolong the network lifetime.

The method of adding mobile sinks to the WSNs infrastructure has attracted much attention recently, such as the methods in [12,27]. Scalable Energy-efficient Asynchronous Dissemination (SEAD) [28], is a mechanism for routing sensing data to mobile sinks. The idea is to construct a minimum Steiner tree for the mobile sink and designate some nodes on the tree as access points. The mobile sink registers itself with the closest access node. When the sink moves out of range of the access node, the route is extended through the inclusion of new access node. Such partial path extension is allowed only for a limited number of hops and then the branch of the Steiner tree for that sink is modified by finding the least cost path to the sink.

In the ALURP [31], a sink only floods its location to the nodes within an adaptive area if the sink only moves in the adaptive area. The ALURP can significantly reduce unnecessary transmissions overhead by local flooding, but the mobile sink still needs to flood its location information throughout the entire network to announce a new adaptive area.

The Two-Tier Data Dissemination (TTDD) [18] represents the early work on data dissemination of the WSNs with mobile sinks. The protocol initially builds a grid structure which divides the network into cells with several dissemination nodes. When the sink requests data, the query packets are flooded locally within the cell until it reaches a dissemination node. A data path from the source to the dissemination node is then established. But the TTDD is not suitable for applications where the flooding area expands as the cell size grows, while a small sized grid structure incurs high overhead for the grid construction.

The Intelligent Agent-Based Routing Protocol (IAR) [13] provides efficient data delivery to mobile sinks. The IAR algorithm reduces signal overhead and improves degraded route called triangular routing problem. The sink periodically examines its distance from the current immediate relay, initiates a new relay path establishment, and reduces the packet loss as the experiment results and signaling overhead.

However, the routing recovery mechanism for the movement of multiple mobile sinks and sensor node failure has rarely been considered. As the fault-tolerant optimization problem to find the optimal routing is a NP-hard one, the heuristic deterministic methods always fall into local optimum, and obtain the approximate optimal routing result. Moreover, in these routing protocols, packets are broadcasted to repair a broken route from source node to sink, which causes significant communication overhead in terms of both energy and delay.

We design an efficient swarm intelligent algorithm to optimize the alternative path from the sources to the sinks in this paper. Our proposed method is developed based on IAR and differs from the above works. We address the routing problem of the WSNs consisting of a large number of sensor nodes and multiple mobile sinks for data gathering. The multiple mobile sinks form a virtual backbone and are concerned with maintaining the backbone connectivity as a result of sinks' movements. When the energy of the nodes around these sinks is exhausted, these sinks can move to a new location with richer sensor energy to gather information. This approach can balance energy consumption and prolong network lifetime.

The main contributions of this paper are as follows: (1) We build the WSNs model with multiple mobile sinks, address to its routing recovery problem with both the movement of multiple sinks and sensor node failure. (2) We propose an endocrine cooperative particle swarm optimization algorithm (ECPSOA), which offers faster global convergence and higher solution quality, and provides fast bio-heuristic routing recovery for the path from source node to the mobile sink. (3) The ECPSOA based routing recovery protocol is designed and proposed for efficiently solving the path breakage problem of the WSNs with multiple mobile sinks, which can decrease the control overhead and minimize energy consumption.

The rest of this paper is organized as follows. Section 2 formulates the routing recovery problem and states our network model. Section 3 describes the proposed algorithm in details, and the routing recovery protocol is presented and analyzed. Section 4 reports the experiment results of the propose approach. Finally, Section 5 concludes the paper.

2. Definition and analysis of network model

The WSNs with multiple mobile sinks is modeled as a connected graph $G(V, E)$ [7,2], where V is a finite set of sensor nodes and E is the set of edges representing connection between these nodes. Suppose there exist n source nodes and m mobile sinks, several source nodes can route the packets through multi-hops to one mobile sink. As such, there would be$n$paths from n given source nodes to m given mobile sinks, which are denoted by p_j $(j \in 1, 2, \ldots, n)$. Each source v_s^j connects a sink $v_{\sin k}^i$ $(i \in 1, 2, \ldots, m)$ with one path p_j. The k-th relay node on the path p_j is denoted by $v_j^k,\ k \in 1, 2, \ldots, h_j$, where h_j is the hop count on path p_j. e_j^m represents the m-th direct edge between two neighbor nodes on p_j. Let $N(p_j) = \left\{ v_j^1, v_j^2, \ldots, v_j^k, \ldots, v_j^n \right\} \subset N(v)$ be the set of the sensor nodes existing along the path p_j, where k represents the distance from the sink to the node on a hop scale. We introduce an agent to directly connect the mobile sink with 1-hop. This similar concept is adopted with different terminologies by several protocols in [20,19,23].

An example of the network model is shown in Fig. 1. There exist 4 source nodes $\{v_s^1, v_s^2, v_s^3, v_s^4\}$ and 2 mobile sinks $\{v_{\sin k}^1, v_{\sin k}^2\}$ in the network. Four paths $\{p_1, p_2, p_3, p_4\}$ connect the corresponding sources with the sinks, in which p_1 contains

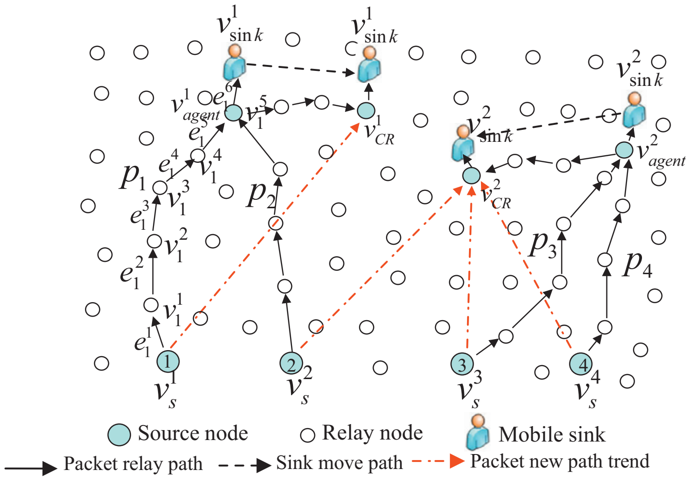


Fig. 1. The network model and routing recovery process for sinks' movement problem.

5 relay nodes $\{v_1^1, v_1^2, v_1^3, v_1^4, v_1^5\}$, and six relay edges $\{e_1^1, e_1^2, e_1^3, e_1^4, e_1^5, e_1^6\}$. The relay node which is the closest (1-hop) to its mobile sink $\{v_{\sin k}^1, v_{\sin k}^2\}$ is regarded as an agent node $\left\{v_{\text{agent}}^1, v_{\text{agent}}^2\right\}$. The nomenclature of used symbols is provided in Table 1, and the routing recovery procedure of the model is described as follows.

2.1. Selection of agent node

To select an agent, the i-th mobile sink $v_{\sin k}^i$ in the network proposes the following processes. It firstly broadcasts an Agent Request (AR) packet. The relay node which receives the AR will reply with an Agent Answer (AA) packet. The AA includes the sender's distance and remaining energy, therefore $v_{\sin k}^i$ can easily determine which node is the closest node around itself. This node is selected as the agent node v_{agent}^i. Then we would discuss the following procedure with two cases:

(1) When the sink moves away.

Step 1: When $v_{\sin k}^i$ moves within the communication range of v_{agent}^i, it can receive the gathered packet directly from v_{agent}^i. However, when $v_{\sin k}^i$ moves out of the range of v_{agent}^i, it will select the closest relay node v_{CR}^i among its neighbor nodes to relay packets from v_{agent}^i to $v_{\sin k}^i$. To be specific, $v_{\sin k}^i$ finds that it is out of the communication range of v_{agent}^i when there is no packet received from v_{agent}^i for a fixed time interval T. The sink will broadcast a Closest Node Request (CNR) packet to its neighbors. The neighbor nodes reply to the CNR packet with a Closest Node Answer (CNA) packet, which includes the coordinates and remaining energy of the sending node. Therefore, the sink can find the closest node v_{CR}^i based on this information, as shown in Fig. 1.

Then $v_{\sin k}^i$ sends an Alternative Path Setup (APS) packet to v_{agent}^i via the selected v_{CR}^i, and v_{agent}^i transmits the packet along the previous path to the closest source node v_s^j. Since $v_{\sin k}^i$ has moved out of the range, v_{agent}^i will temporally store the packets transmitted to the sink in order to prevent the packet loss in this time interval. These stored packets are routed to $v_{\sin k}^i$ when v_{agent}^i receives the APS packet. v_{CR}^i becomes the new agent for $v_{\sin k}^i$. As shown in Fig. 1,

Table 1
The nomenclature of used symbols.

p_j	The path from the j-th source node in the network to its mobile sink
p_b	The best (optimal) path we have calculated from the j-th source node in the network to its mobile sink
v_s^j	The source node on the path p_j
$v_{\sin k}^i$	The i-th mobile sink in the network
v_j^k	The k-th relay node on the path p_j
v_{agent}^i	The agent node belonging to the i-th sink in the network
v_{CR}^i	The closest relay node belonging to the i-th mobile sink in the network
e_j^m	The m-th direct edge between two neighbor nodes on the path p_j
$G(V,E)$ or G	The connected graph containing all the nodes in the network
N_p	The set of nodes which can be used to establish an alternative path p_j
$N(p_j)$	The set of the sensor nodes existing along the path p_j
$N(v)$	The set of all the sensor nodes in the network

since two sinks ($v^1_{\sin k}, v^2_{\sin k}$) have moved to the new places, v^1_{CR} and v^2_{CR} are selected to establish the new alternative paths for source nodes ($v^1_s, v^2_s, v^3_s, v^4_s$).

Step 2: To establish the optimal path, the source v^j_s broadcasts a Relay Node Selection (RNS) packet to the other nodes in the network. Each node relays the RNS packet containing its own state information (coordinate, remaining energy, delay and so on), until a new agent receives the RNS packet. The new agent will gather the information, and select a new optimal alternative path from source node to mobile sink using the proposed ECPSOA. The alternative path is the multi-hop path from the source node to the agent.

(2) When the relay node fails.

When one relay node fails due to physical damage or energy depletion, $v^i_{\sin k}$ will find there is no packet received from v^i_{agent} although it is still within the communication range of v^i_{agent}, as shown in Fig. 2. Then $v^i_{\sin k}$ directly sends an Alternative Path Setup (APS) packet to v^i_{agent}, and v^i_{agent} transmits the packet through the previous path to the source node to establish a new path. This procedure is the same as the Step 2 in (1).

2.2. Establishment of alternative path

The procedure of how to establish an optimal path is described as follows. v^i_{agent} gathers all the packets and information of these nodes, and extracts the set of nodes N_p to establish a new alternative path p_j in $G(V,E)$. Each node represents a particle, and the particle population size is n. Some nodes of N_p can form a particle sequence $\left\{v^1_{p_j}, v^2_{p_j}, v^3_{p_j}, \ldots, v^m_{p_j}\right\}$ $(m \leqslant n)$ according to their order, which can form a path p_j from the source v^j_s to the sink $v^i_{\sin k}$. The ECPSOA optimizes the particle sequence to obtain the optimal path p_b with the optimal fitness $fitness(p_b)$, where p_b includes the following nodes $\left\{v^b_s, v^1_{p_b}, v^2_{p_b}, v^3_{p_b}, \ldots, v^m_{p_b}, v^b_{\sin k}\right\}$, as shown in Fig. 2.

The factors affecting the choice of path p_j include: (1) remaining energy function of each node, $Rene\left(v^k_j\right)$; (2) distance function of the edge between neighboring nodes, $dist\left(e^k_j\right)$; (3) energy consumption function of node, $ene\left(v^k_j\right)$; (4) communication delay function of node, $delay\left(v^k_j\right)$. Then these parameters can determine the fitness function of p_j, $fitness(p_j)$:

$$fitness(p_j) = \frac{\sum_{v^k_j \in p_j} Rene\left(v^k_j\right)}{\omega_1 f_1 + \omega_2 f_2 + \omega_3 f_3} \tag{1}$$

$$f_1 = \frac{\sum_{v^k_j \in p_j} ene\left(v^k_j\right)}{\sum_{v \in V} ene(v)} \tag{2}$$

$$f_2 = \frac{\sum_{v^k_j \in p_j} delay\left(v^k_j\right)}{\sum_{n \in V} delay(v)} \tag{3}$$

$$f_3 = \frac{\sum_{e^k_j \in p_j} dist\left(e^k_j\right)}{\sum_{e \in E} dist(e)} \tag{4}$$

$$\text{s.t.} \quad \sum_{v^k_j \in p_j} ene\left(v^k_j\right) \leqslant Ene, \quad \sum_{v^k_j \in p_j} delay(v^k_j) \leqslant D, \quad \sum_{e^k_j \in p_j}^{dist(e^k_j)} > L \tag{5}$$

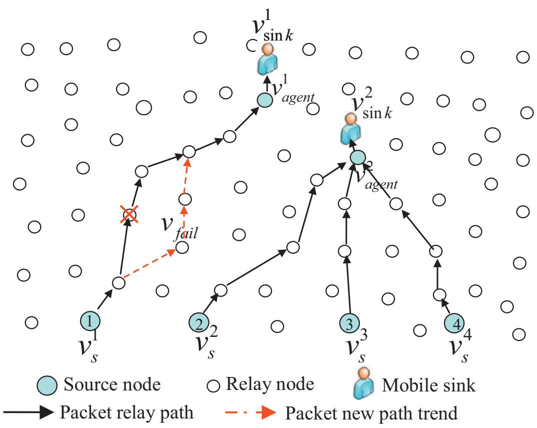


Fig. 2. Routing recovery process for the relay sensor node failure problem.

where f_1 is the distance of the edges of p_j versus the distance of all the edges in the network, f_2 is the delay of the nodes on path p_j versus the delay of all the nodes, f_3 is the energy consumed by the nodes on path p_j versus the energy consumed by all the nodes in the network. ω_1, ω_2, and ω_3 are the weights of remaining energy, delay, and distance constraints in the fitness function, respectively, and $\omega_1 + \omega_2 + \omega_3 = 1$. We set $\omega_1 = 0.4, \omega_2 = 0.2, \omega_3 = 0.4$. Eq. (5) is the path constraint equation, in which *ene* states for the energy constraint, *D* is the delay constraint, and *L* means the distance constraint. The higher fitness value indicates the more optimal path, at last the best path p_b with the optimal fitness $fitness(p_b)$ will be selected. Then v^i_{agent} transmits a Previous Path Clear (PPC) packet through the previous path to delete this previous path, and the new alternative path is established.

2.3. Assumption of network model

We use the energy model proposed in [22,1] for the WSNs model with multiple mobile sinks, in order to achieve an acceptable signal-to-noise ratio, the energy consumption equation of sensing and transferring *m* bit data is as follows:

$$ene_{sen}(m) = a_1 m \tag{6}$$

$$ene_{tx}(m, d) = (\beta_1 + \beta_2 d^n)m \tag{7}$$

$$ene_{rx}(m) = \gamma_1 m \tag{8}$$

where *d* is the distance from a node to its neighbor node, $ene_{sen}(m)$ is the energy consumption of sensing *m* bits of data, $ene_{tx}(m, d)$ and $ene_{rx}(m)$ are the energy consumption of sending and receiving *m* bits of data, *n* is the channel attenuation index, a_1, β_1, β_2 and γ_1 are energy consumption parameters of sensing circuit, sending circuit, sending amplifier and receiving circuit, respectively. For the ECPSOA, the energy consumption of the particle updating and the endocrine selection is defined as ene_{PU}, ene_{ES} per iteration, respectively. By using this method, the total energy consumption of the data transmission and executing the proposed ECPSOA per round can be calculated in the simulation.

The simple fault model proposed in [3] is also adopted. The fault-model is easy to analyze, but also sophisticated enough to capture the fault behavior effectively. The probability of the sensor nodes failure in the network is given by p_{node}, and the probability of the mobile sink failure is assumed to be $p_{\sin k} \approx 0$. If any sensor node is failed, our routing recovery method dealing with node failure problem is implemented.

The design of the method for the network is based on the following assumptions: (1) An area is covered by a large number of homogeneous sensor nodes which communicate with each other through short-range radios. Sensor nodes are stationary, but the multiple sinks move and change their positions constantly with a fixed speed. (2) Data is sensed and transmitted from each source node to the closest mobile sink in each time period *T*. (3) The sinks move in a limited region. Each sensor node and mobile sink can sense its own location.

3. Design of the protocol for the WSNs with multiple mobile sinks

3.1. PSOAs

As we known, particle swarm optimization algorithm (PSOA) searches for an optimum through each particle flying in the search space and adjusting its flying trajectory according to its personal and global best experience. Owing to its simple structure and high efficiency, the PSOA has become a widely adopted optimization technique [30,8].

In the traditional PSOA, each particle is a potential solution to the problem. Assume *N* particles fly in the *D*-dimensional search space, the position of the *i*-th particle is $x_i^t = (x_{i1}^t, x_{i2}^t, \ldots, x_{iD}^t)^T$, and its velocity is $v_i^t = (v_{i1}^t, v_{i2}^t, \ldots, v_{iD}^t)^T$. $p_i = (p_{i1}^t, p_{i2}^t, \ldots, p_{iD}^t)$ is the best previous position of the particle, and p_g is the best global position of the whole particle swarm. Therefore in each time step *t*, the velocity *V* and the position *X* of each particle is updated with following equations [14,24]:

$$v_{id}^{t+1} = w v_{id}^t + c_1 rand_1 (p_{id}^t - x_{id}^t) + c_2 rand_2 \left(p_{gd}^t - x_{id}^t\right) \tag{9}$$

$$x_{id}^{t+1} = x_{id}^t + v_{id}^{t+1} \tag{10}$$

where c_1, c_2 are learning factors, we select $c_1 = c_2 = 2$. $Rand_1$ and $Rand_2$ are random numbers uniformly distributed in $[0, 1]$.

In the PSOA, the diversity of population will decrease in the late stage of evolution, and optimizing stops when reaching a local optimal solution, which exhibits premature convergence problem. In order to improve the performance of the PSOA, Ratnaweera and Halgamure [25] introduced the accelerating factor to modify the parameters c_1 and c_2. The static non-linear method of modifying the inertia weight was introduced in [16] to improve the performance of the PSOAs. Lin and Chen proposed a cooperative PSOA (CPSOA) [17], which used the cooperative behavior of multiple swarms to improve the PSOAs by jumping out local minimum. It can compensate the limitation of an individual by a number of individuals from other symbiotic groups in the interaction, thus avoid mistake of judgment caused by single exchange of information. However, it still evolves with the formula of the SPSOA. The trajectory of each particle is unable to yield high diversity of particles to enlarge search space, so it may get a suboptimal solution.

Therefore, we apply the adjustment mechanism of the endocrine system [15] to propose the ECPSOA. We consider the best position of a particle and the global best position of a swarm in current generation. Then, we combine the supervision and controlling principle between stimulation hormones (SH) and releasing hormones (RH) of the endocrine system, and use the individual of the current solution set to control the nearest class of swarm. The particles are grouped by the SH, and the best positions of classes are proposed to update the positions of particles, so the convergence and distribution performance of the ECPSOA can be improved. The global optimization position of the class can reflect the influence of the nearest optimal particle to other particles, so that the ECPSOA can jump out of local optimization, improve the searching capability, and maintain the diversity of solution set.

3.2. Description of the proposed ECPSOA

The ECPOSA is used to address the problem of data transmission from the source nodes to the mobile sink. It can provide a fast recovery mechanism from path failure due to the sinks movement, or physical damage and energy depletion of sensor node problem. The flowchart of the ECPSOA is shown in Fig. 3, and the detailed procedures are described as follows.

3.2.1. Initialization step

To initialize the ECPSOA, the population size of particle is n, the division factor is k, so each particle swarm includes n/k particles. Then the D-dimensional vector (vector of particle's position and velocity) is divided into k swarms. The position and velocity of the i-th particle in t-th time is respectively $x_i^t = (x_{i1}^t, x_{i2}^t, \ldots, x_{iD}^t)^T$ and $v_i^t = (v_{i1}^t, v_{i2}^t, \ldots, v_{iD}^t)^T$. As the path should have n points (nodes), the number of which is the same as the number of particles, so the initial particle swarm can be represented as a matrix by $[D \times 3n]$. The first n columns of the matrix are the positions of particle, the middle n columns are the velocities of particle, and the last n columns are the amount of hormone (one column represents a particle). The optimal position vector of the whole sub-swarms is presented by a vector function $b(\cdot)$:

$$b(L, i) = (p_g S_1, \ldots, p_g S_{i-1}, L, p_g S_{i+1}, \ldots, p_g S_k) \tag{11}$$

where $x_m S_i$ represents the position vector of the m-th particle in the i-th swarm, $p_m S_i$ is the optimal history position vector of the m-th particle in the i-th swarm, and $p_g S_i$ represents the optimal experience position vector of the i-th swarm.

3.2.2. Particle updating step

The velocity and position of the particle are updated as Eqs. (9) and (10). The updating equation of optimal position vector of particles in each sub-swarm is as follows:

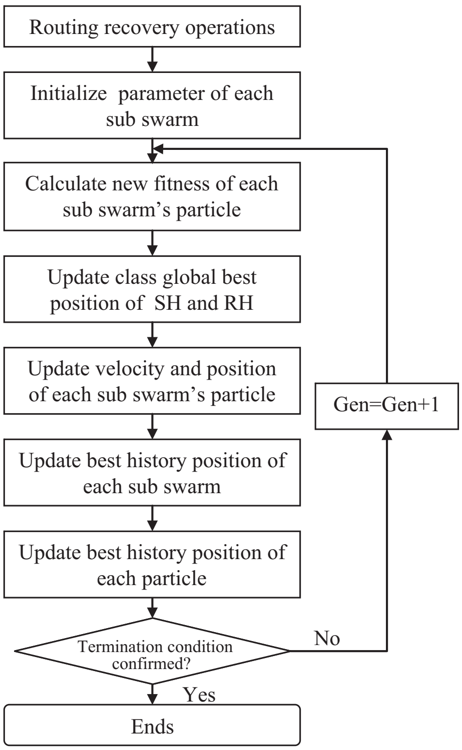


Fig. 3. The flowchart of the ECPSOA.

$$b(p_m S_i, i) = \begin{cases} b(x_m S_i, i), & fitness(b(x_m S_i, i)) \geqslant fitness(b(p_m S_i, i)) \\ b(p_m S_i, i), & fitness(b(x_m S_i, i)) < fitness(b(p_m S_i, i)) \end{cases} \tag{12}$$

where $1 \leqslant i \leqslant k$. The updating equation of optimal position of each sub-swarm is:

$$b(p_g S_i, i) = \underset{P(p_m S_i, i)}{\arg} \max fitness(b(p_m S_i, i)), \quad 1 \leqslant m \leqslant n/k, \quad 1 \leqslant i \leqslant k \tag{13}$$

Eq. (13) indicates that the optimal position of the m-th sub-swarm selects the personal optimal position with the optimal fitness of particle in the swarm. In this step, the limitation of an individual can be compensated by a number of individuals from other symbiotic groups in the interaction. It can avoid mistake of judgment caused by single exchange of information.

3.2.3. Endocrine step

In this process, according to the endocrine principle, we initialize two swarms for each sub-swarm: stimulation hormone (SH) S_t and releasing hormone (RH) R_t, which both have the same structure and size [32].

Firstly, we combine the swarm S_t and R_t, and generate the new swarm U_t ($U_t = S_t \cup R_t$). The optimal solution of U_t is selected as the candidate swarm CS_t of SH. Then all the solution crowding distances of CS_t are sorted in descending order according to the crowding distance algorithm in [4]. The first q solutions are selected as S_{t+1}, where q is a constant. The method is as follows:

(a) If the solution number of CS_t is smaller than or equal to q, then $S_{t+1} = \mathrm{CS}_t$.
(b) If the solution number of CS_t is bigger than q, then S_{t+1} is confirmed by:

$$S_{t+1} = \cup\{x_i = u_i, u_i \in \mathrm{CS}_{t+1}, i = 1, 2 \ldots q\} \tag{14}$$

We assume that the size of S_t in SH is $N(S) = sizeof(S_t)$, and select the $0.1 \times N(S)$ individuals with bigger crowd distance to generate a solution set according to $N(S)$. Then an individual of the set is selected randomly as the current global best position $P_g(t)$ of particle.

As the SH plays the supervision role for the RH, we classify the RH in $N(S)$ classes, each individual S_i of the SH controls a class $C(S_i)$ as shown in Eq. (15):

$$C(S_i) = \{c \in P, dist(c, S_i) = \min(dist(c, S_j))\} \quad j = 1, 2 \ldots N(S) \tag{15}$$

Therefore, a supervisor S_i coming from the same class $C(S_i)$ is embedded in each individual of the RH during the updating step. S_i is used as the class global best position C_{gd} of $C(S_i)$. By using this method, each particle flies to the best history direction due to the selection of local best position. It flies to the solution with large crowd distance of searching domain due to the selection of searching domain global best position, thus avoids convergence to local optimum area. The global best position in a class plays the role of maintaining swarm dispersion, thus the global information and local information are combined completely.

Thereafter, in each time step t, the velocity and position of each particle are updated according to the following equations:

$$v_{id}(t+1) = w v_{id}(t) + c_1 Rand_1 (p_{id} - x_{id}(t)) + c_2 Rand_2 (p_{gd} - x_{id}(t)) + c_3 Rand_3 (C_{gd} - x_{id}(t)) \tag{16}$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t) \tag{17}$$

where c_1, c_2, c_3 are learning factors. We select $c_1 = c_2 = 2.05$, and $c_3 = 2$. $Rand_1, Rand_2$ and $Rand_3$ are uniformly distributed in $[0, 1]$. w plays an important role to the convergence of the result among the adjustable parameters [9]. To improve the system performance, we propose the decreased weight w as follows:

$$w(t) = w_{\max} - (w_{\max} - w_{\min}) \times \exp\left(-\frac{p_{id}}{p_{gd}} \times \frac{1}{t}\right) \tag{18}$$

As the iteration time increases, the weight becomes lower. Moreover, the best global and personal positions of the particle are involved, so the weight can take specific condition into consideration, and alter adaptively for better convergence. Here we set $w_{\max} = 0.85, w_{\min} = 0.35$.

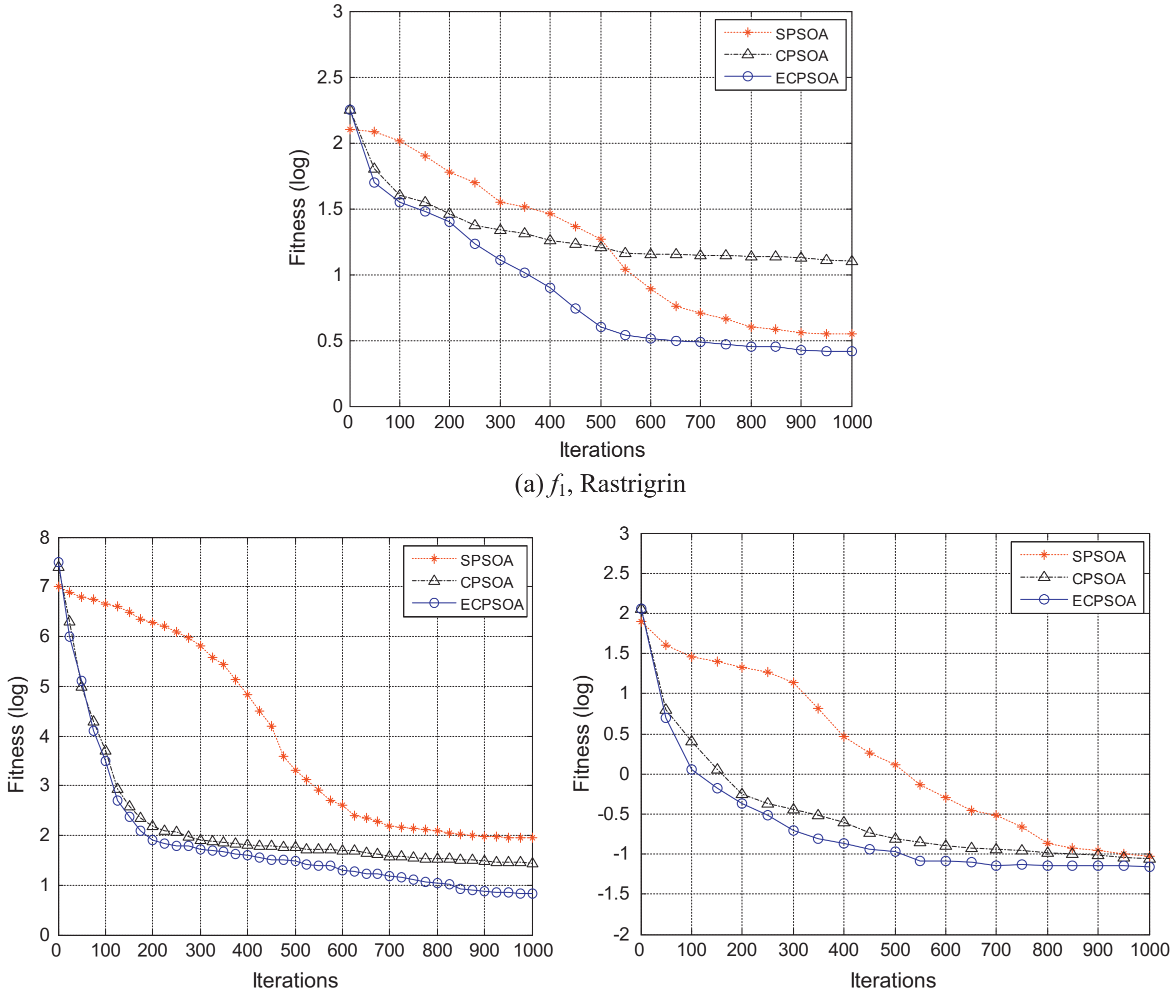
3.2.4. Termination criterion

If the solution is satisfied with the termination conditions: The fitness of function $fit(\cdot)$ is the optimal fitness or the number of iterations decreases from 80 to zero, then the procedure ends; Otherwise returns to **Step 2**. Thereafter the optimal alternative path would be established.

3.3. Framework of proposed routing recovery protocol

The ECPSOA is the kernel algorithm of our routing recovery protocol of the WSNs with multiple mobile sinks. To deal with the path failure problem due to the movement of sinks and the physical damage or energy depletion of nodes, the ECPSOA optimizes the path fitness function to provide the fast routing recovery mechanism with an alternative optimal-fitness path. Apparently, the more suitable alternative path selected contains more available energy, less path distance, less energy consumption and less delay from the source to the mobile sink. We now demonstrate with an example how the routing recovery protocol for the sinks' movement or nodes' failure is implemented in our protocol.

Step 1: As illustrated in Fig. 1, if the i-th sink $v^i_{\sin k}$ moves out of the communication range of its agent v^i_{agent}, it will broadcast the CNR to select the closest node v^i_{CR}, else go to **Step 2**. Then it sends the APS to the agent via v^i_{CR}, and v^i_{agent} transmits the stored packets to $v^i_{\sin k}$. Go to **Step 3**.
Step 2: As illustrated in Fig. 2, if $v^i_{\sin k}$ finds that a relay node is failed in the previous path, it sends the APS via v^i_{agent} to the source to establish an alternative path. Go to **Step 3**.
Step 3: The source v^j_s broadcasts the RNS to the surrounding relay nodes. Each relay node relays the RNS with its own state information, until the RNS reaches v^i_{agent}. Then v^i_{agent} gathers the information, and selects the new optimal path P_b from v^j_s to $v^i_{\sin k}$ using the proposed ECPSOA algorithm.
Step 4: The optimal alternative path (OAP) packet that we proposed includes the IDs of selected nodes $\left\{v^1_{p_b}, v^2_{p_b}, v^3_{p_b}, \ldots, v^n_{p_b}\right\}$ on P_b. v^i_{agent} sends the packet to the sink, and transmits through the new alternative path to the source v^j_s. Once v^j_s and $v^i_{\sin k}$ receive this packet, the alternative path from v^j_s to $v^i_{\sin k}$ is established, and v^i_{agent} sends the PPC along the previous path to delete this previous path information. Then the protocol ends.



(a) f_1, Rastrigrin

(b) f_2, Rosenbrock

(c) f_3, Griewank

Fig. 4. Comparison of the ECPSOA, the SPSOA and the CPSOA.

Table 2
Comparison results of various PSOAs on three functions.

Function		SPSOA	CPSOA	ECPSOA
f_1	Mean	39.4	25.15	0.04
	SD	10.1	5.06	0.09
	$t_{1/s}$	7.93	9.12	7.67
f_2	Mean	12.5	35.6	1.31
	SD	13.65	28.35	1.24
	$t_{2/s}$	6.90	8.25	8.50
f_3	Mean	8.58×10^{-3}	5.12×10^{-3}	0.02
	SD	1.05×10^{-2}	9.24×10^{-3}	0.01
	$t_{3/s}$	10.65	14.50	8.56

3.4. Overhead analysis of proposed protocol

The communication overhead control of the protocol is very important to reduce its energy consumption. Assume that N sensor nodes are randomly distributed in area A, and $\sqrt{N}$ nodes are on each side, the path connecting the source and the sink is assumed to be the square of the area A, which contains $\sqrt{2N}$ nodes. The communication radius of nodes is r, and the number of neighbor nodes of a node is calculated as follows:

$$N_{nn} = N\frac{\pi r^2}{A} \tag{19}$$

Both the request and the response packets have the same size S. The communication overhead of a node generated by the packets CNR, APS, RNS, PPC and OAP is:

$$O = \left(3\sqrt{2N} + N_{nn}\right)S + 4S = \left(3\sqrt{2N} + 4 + N\frac{\pi r^2}{A}\right)S \tag{20}$$

We can see that our protocol generates almost the same control overhead with the IAR, and less than most of other conventional protocols for the WSNs.

4. Experimental result evaluation

4.1. Performance comparison between the ECPSOA and the PSOAs

We have conducted several simulation experiments in Matlab platform to evaluate the performance of the ECPSOA and compare it with the PSOAs. The simulation is constructed on Windows 7 with Intel core i3-2100 Dual-Core CPU (3.10 GHz)

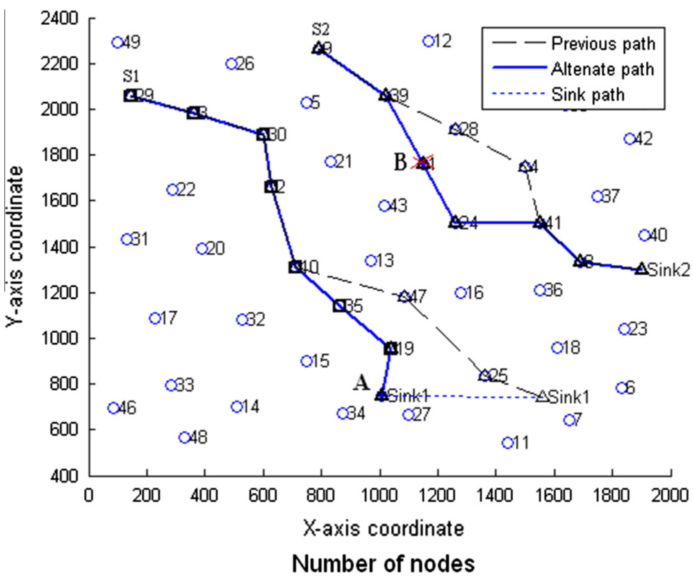


Fig. 5. The snapshot of establishing the alternative path using the ECPSOA.

and 4 GB RAM. Firstly, three well-known test functions adopted widely in benchmarking optimization algorithms, namely Rastrigrin, Rosenbrock, and Griewank, are used to evaluate the ECPSOA. The standard PSOA (SPSOA) [8] and the CPSOA [17] are used for comparisons, as they are typical PSOAs that are reported to perform well on complex optimization problems. The parameters used for the test are: function dimension $D = 40$, iterated generations $P_{Gen} = 1000$. The simulation results presented in Fig. 4(a)–(c) illustrate the evolution of optimal fitness for these PSOAs.

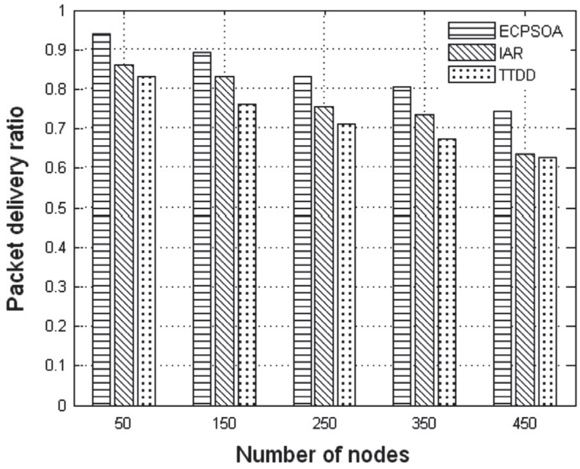
$$\text{(1) Rastrigrin}\quad f_1(x) = \sum_{i=1}^{n}(x_i^2 - 10\cos(2\pi x_i)) + 10 \quad x_i \in [-5, 5] \tag{21}$$

$$\text{(2) Rosenbrock}\quad f_2(x) = \sum_{i=1}^{n} 100\left(x_{i+1} - x_i^2\right)^2 + (1 - x_i)^2 \quad x_i \in [-100, 100] \tag{22}$$

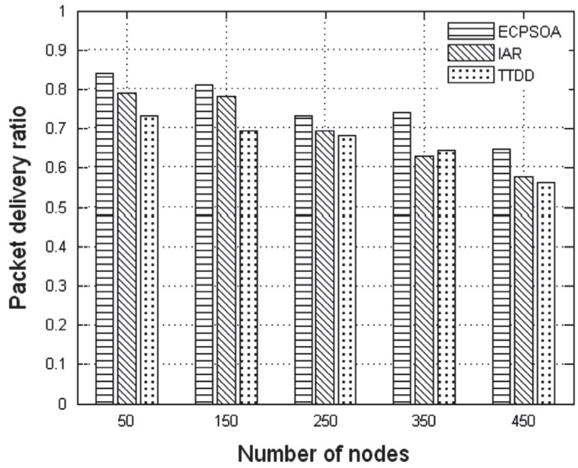
$$\text{(3) Griewank}\quad f_3(x) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \quad x_i \in [-600, 600] \tag{23}$$

f_1 (Rastrigrin) is a multimodal function with high dimensionality. The search improvements of the SPSOA are a bit slower but eventually better than the CPSOA. The ECPSOA performs best and gives consistently a near-optimum result, whereas other methods stagnate with no further improvement. The ECPSOA demonstrates the best performance among the three PSOAs for the multimodal function.

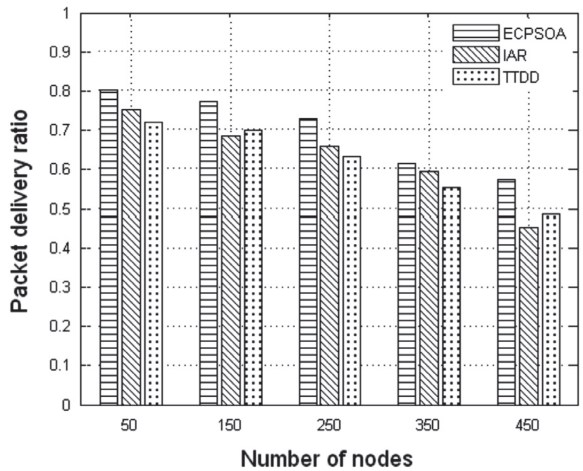
f_2 (Rosenbrock) is unimodal in a search space but can be treated as a multimodal function in high-dimensional cases. Its convergence to the global optimum is difficult. The SPSOA converges slowly and ultimately produces poor optimal fitness for



(a) When the node failure probability is 0.01

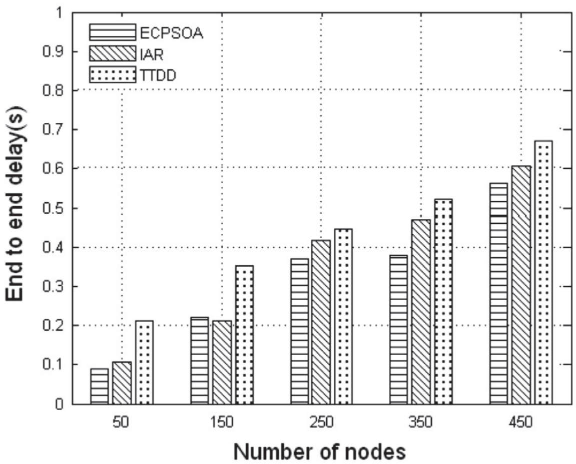


(b) When the node failure probability is 0.02

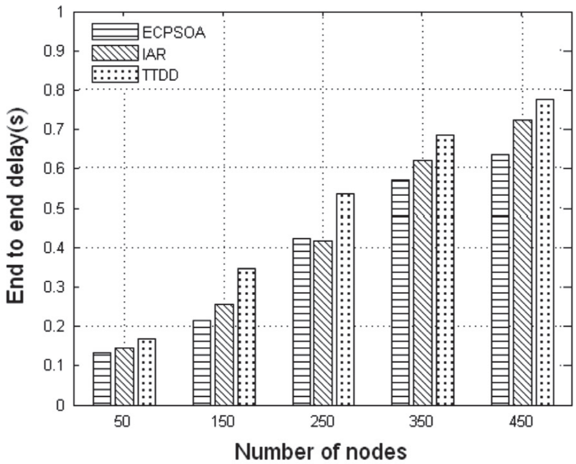


(c) When the node failure probability is 0.04

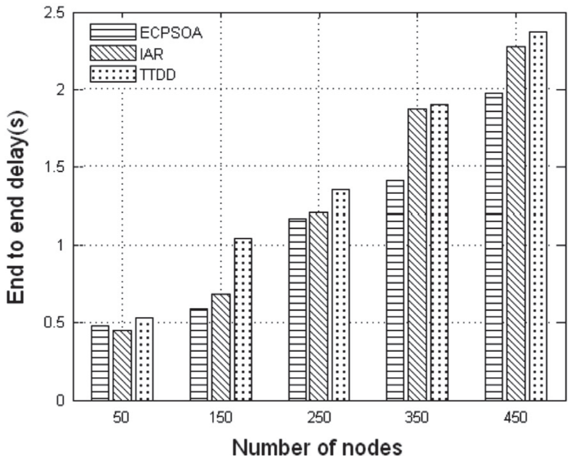
Fig. 6. Average packet delivery ratio with respect to different node failure probabilities.



(a) When the node failure probability is 0.01



(b) When the node failure probability is 0.02



(c) When the node failure probability is 0.04

Fig. 7. Average end-to-end delay with respect to different node failure probabilities.

this case. In contrast, the ECPSOA improves the average optimum solution significantly, and performs better than the SPSOA and the CPSOA in terms of convergence speed and final solution.

f_3 (Griewank) is a rotated multimodal function. It can be used to test the capability of exploring global optimal solution of proposed algorithms in multi-dimension space. As shown in Fig. 4(c), the SPSOA leads to local minimum solution in initial iterations, and fails to make further progress in later iterations for solving multi-dimensional function. We can see that the ECPSOA can overcome the shortcoming of converging to local optimum for the other PSOAs, and improve the global search ability.

Table 2 shows the global mean values, the standard deviation (SD) of the final solutions and the average running time during 10 rounds simulation. We can observe that the ECPSOA achieves the best solution on most of the functions. These comparisons verify that the multi-swarm evolution strategy and the endocrine mechanism indeed make the ECPSOA perform better than the other PSOAs in most of the test functions. It offers the ability of avoiding local optima and premature convergence, providing optimal solution accuracy in multimodal functions. The time required by these algorithms converging to the optimal solution is also shown in Table 2. t_1, t_2 and t_3 represent average running time for the PSOAs, which also reflect the computational time complexity. From Table 2, we can see that the ECPSOA consumes less time than the SPSOA and the CPSOA. The reason is that it takes less iteration times for the ECPSOA to converge to the optimal value, due to its contribution to the capability improvement of diversity and jumping out of likely local optima of multi-dimension function.

4.2. Applied to routing problem of multiple mobile sinks

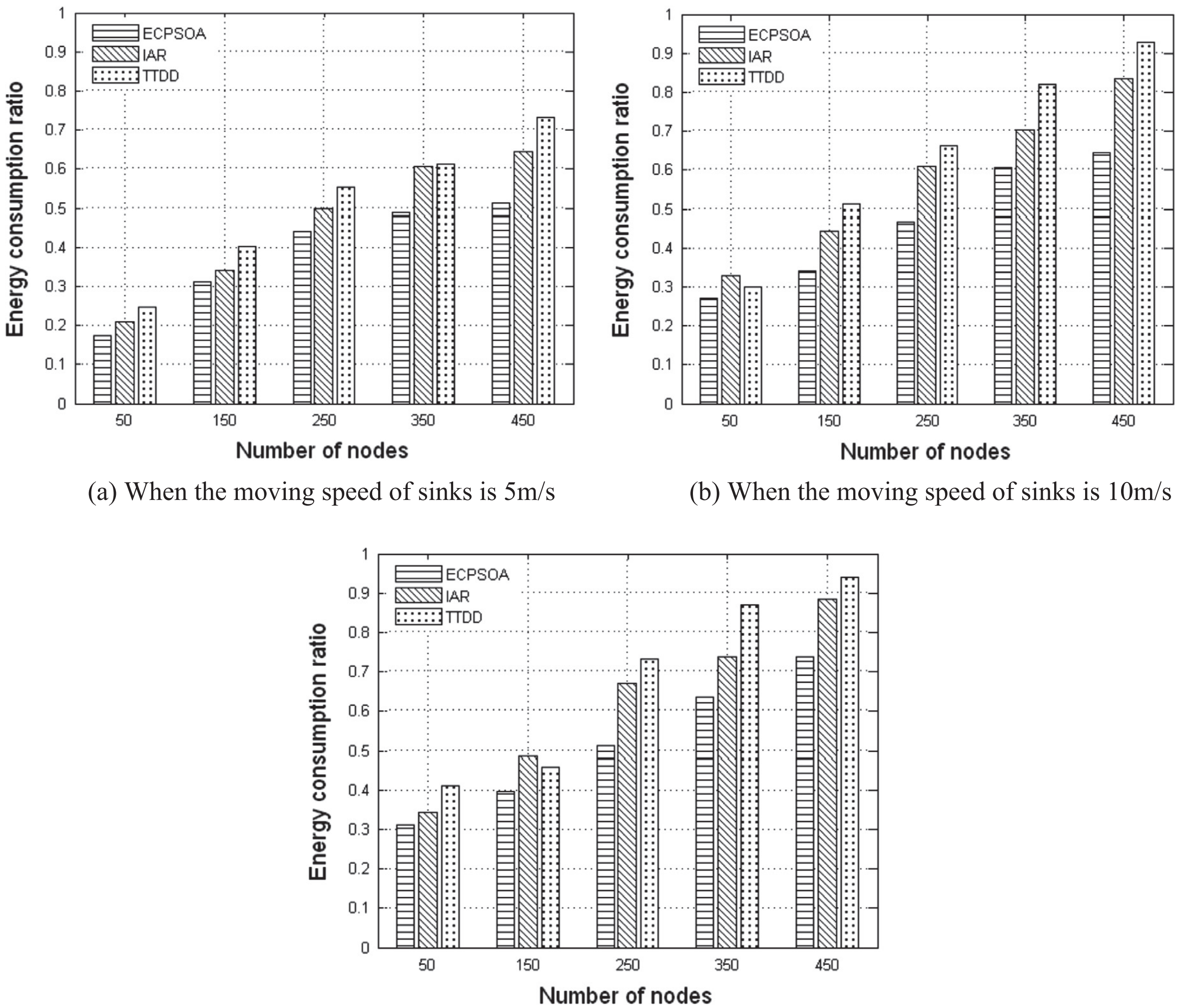
4.2.1. Simulation model

In this section, we have conducted several experiments to analyze the performance of the proposed protocol, and the experiment environment is the same as the performance simulation of the PSOAs in Section 4.1. From 50 to 450 sensor nodes are randomly deployed in a two-dimensional area ($5000 \times 5000\ \mathrm{m}^2$). The sensor nodes are homogeneous and have the same initial energy of 120 J. Their sensing radius is 300 m, and communication radius is 600 m. We assume there exist five mobile sinks in the network, and their speeds change within the range of 5 m/s to 20 m/s. The probability of node failure p_{node} changes from 0.01 to 0.04. The source node delivers packets at the rate of 20 data packets per round, with 1 KB of each packet size, and the simulation lasts for 600 rounds. According to the description of energy model in Section 2.3, we select $a_1 = 60$ nJ/bit, $\beta_1 = 45$ nJ/bit, $\beta_2 = 10$ nJ/bit, $\gamma_1 = 135$ pJ/bit/m^2, channel attenuation index $n = 2$. For the ECPSOA, $ene_{PU} = 80$ pJ, $ene_{ES} = 50$ pJ at per iteration. We consider that the sinks have sufficient energy and thus ignore the energy consumed by the sinks. The parameters used for the ECPSOA are: function dimension $D = 30$, division factor $k = 6$, and iterated generation $P_{Gen} = 800$. The purpose of the experiment is to demonstrate that the ECPSOA can provide a more robust transmission environment in a mobile sink network.

We take a snapshot during the simulation to illustrate the effect of the proposed protocol. Fig. 5 illustrates an area ($2000 \times 2000\ \mathrm{m}^2$) of the network with the two available paths, one is from the source node v_s^1 (node 29) to the mobile sink $v_{\sin k}^1$, and the other is from v_s^2 (node 9) to the mobile sink $v_{\sin k}^2$. We can see that whenever $v_{\sin k}^1$ moves to a new place A, the source node v_s^1 immediately establishes an optimal alternative path (path 29-3-30-2-10-47-25-Sink1 in Fig. 5) to reach $v_{\sin k}^1$. Once a relay node (node 1) of the second path fails in the place B, another optimal alternative path (path 9-39-28-4-41-8-Sink2 in Fig. 5) will be established to reconnect v_s^2 and $v_{\sin k}^2$, so as to replace the previous broken path.

4.2.2. Evaluation of the experimental results

We now present the simulation results of our proposed protocol along with the protocol IAR and TTDD, which are both suitable for dealing with routing problem of multiple mobile sinks. We observe the following metrics: average packet delivery ratio (measured as the average number of successfully delivered packet versus required packet per round), average



(a) When the moving speed of sinks is 5m/s
(b) When the moving speed of sinks is 10m/s
(c) When the moving speed of sinks is 20m/s

Fig. 8. Average energy consumption ratio with respect to different speeds of mobile sinks.

end-to-end delay [20], and average energy consumption ratio (measured as the average energy consumption from source to sink versus the initial energy per round). We compare the average packet delivery ratio and the average end-to-end delay for different network size (size is from 50 to 450 nodes) with respect to several node failure probabilities (value of 0.01, 0.02 and 0.04) in Figs. 6 and 7, while maintaining the mobile sink speed as 5 m/s. Then the average energy consumption ratio is compared for different network sizes with respect to several sink speeds (speed of 5 m/s, 10 m/s and 20 m/s) in Fig. 8, while maintaining the node failure probability as 0.01. Using this measurement, the influence of node failure probability and mobile sink speed to the network performance can be measured respectively.

Fig. 6(a)–(c) shows that our robust routing protocol with the ECPSOA outperforms TTDD and IAR up to 12% and 6% in average packet delivery ratio, respectively. The improvement is attributed to its fast responsiveness to the changing sink position and node failure. It provides an efficient routing recovery mechanism from the path failure with an optimal alternative path, and improves the successful delivery rate whenever the path link is broken. Note that the packet delivery ratio will reduce with the increase of the node failure probability p_{node} by comparing Fig. 6(a)–(c), but the ECPSOA can still deliver more packet than the other protocols with the same network size.

From Fig. 7(a), the end-to-end delay of our routing protocol is only slightly outgo to TTDD and IAR. This is because our protocol adopts the ECPSOA algorithm, and the node selection procedure with the algorithm during establishing the alternative path will prolong the time delay. Nevertheless, the more efficient routing recovery and less communication control overhead still make its end-to-end delay better than the other protocols. Routing information is stored in each relay node in the ECPSOA protocol to improve the speed of generating the alternative path, and packets are immediately forwarded from agent node or source nodes once the path is repaired. Note that the end-to-end delay increases with p_{node} as shown in Fig. 7(a)–(c), as the less stable topology consumes more time for the protocols to maintain the network and prolong the delay.

As shown in Fig. 8, the energy consumption ratio of all these protocols increases as the mobile sinks move faster, because the change of the frequent topology will incur heavier communication overhead. We can see that the energy consumption of our routing protocol is much less than that of IAR and TTDD, for the proposed protocol will select the relay node with optimized QoS parameters (energy, delay, energy consumption and so on) to establish an alternative path. Also, the communication overhead in the network is minimized and indirectly reduces energy consumption. At the same time, as the network size grows, the difference between our proposed protocol and the other protocols will become larger.

It can be explained by the reason that the larger network size indicates the longer delivery length of the path, thus the advantage of the proposed ECPSOA with the better alternative path selection is demonstrated more obviously, which means that our proposed protocol is more suitable for deploying in the large scale networks with multiple mobile sinks.

5. Conclusions

This study presents a novel routing recovery protocol based on the ECPSOA for the WSNs with multiple mobile sinks. The ECPSOA controls route keeping and routing recovery by estimating the sink mobility and node's failure. Once the previous path is broken due to the sinks movement or sensor nodes failure, an alternative reliable path is established for the packet delivery. Based on the path quality, the alternative path is able to adapt to the unpredictable varying network topology with multiple sinks, minimize the energy consumption, and prolong the network lifetime. In our proposed ECPSOA, the multi-swarm strategy guides particles to fly in better directions, and the endocrine mechanism yields high diversity of particles to increase search space, which can jump out of local optimization and improve the searching capability. Both analysis and simulation results show that our routing recovery protocol outperforms the other aforementioned protocols in communication overhead in terms of both energy and delay, and the network robustness against path breakage due to multiple sinks movement or nodes failure is also improved.

For the future work, we will focus on improving the convergence performance, reducing the computational complexity of the ECPSOA, and validating the proposed protocol on different scenarios with various movement trajectories of multiple mobile sinks. Maximizing the network lifetime is the most important optimization objective.

Acknowledgments

This work was supported in part by the Key Project of the National Nature Science Foundation of China (No. 61134009), the National Nature Science Foundation of China (Nos. 61473077, 61473078, 61401257, 61305014), Program for Changjiang Scholars from the Ministry of Education, Specialized Research Fund for Shanghai Leading Talents, Project of the Shanghai Committee of Science and Technology (Nos. 13JC1407500), Innovation Program of Shanghai Municipal Education Commission (No. 14ZZ067), The Fundamental Research Funds for the Central Universities (No. 2232012A3-04), Shandong Province Young and Middle-Aged Scientists Research Awards Fund (No. BS2013DX021), and Shandong Academy Young Scientists Fund Project (No. 2013QN037).

References

[1] A. Boukerche, R.W.N. Pazzi, R.B. Araujo, Fault-tolerant wireless sensor network routing protocols for the supervision of context-aware physical environments, J. Parallel Distrib. Comput. 66 (4) (2006) 586–599.

[2] M. Cardei, S.H. Yang, Algorithms for fault-tolerant topology in heterogeneous wireless sensor networks, IEEE Trans. Parallel Distrib. Syst. 19 (4) (2008) 545–558.
[3] L. Chitnis, A. Dobra, Aggregation methods for large-scale sensor networks, ACM Trans. Sensor Networks 4 (2) (2008) 1–36.
[4] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Trans. Evol. Comput. 6 (2) (2002) 182–197.
[5] Y.S. Ding, Y.L. Jin, L.H. Ren, K.R. Hao, An intelligent self-organization scheme for the internet of things, IEEE Comput. Intell. Mag. 8 (3) (2013) 41–53.
[6] Y.S. Ding, X.J. Lu, K.R. Hao, L.F. Li, Y.F. Hu, Target coverage optimization of wireless sensor networks using multi-objective immune co-evolutionary algorithms, Int. J. Syst. Sci. 42 (9) (2011) 1531–1541.
[7] B.H. Elyes, C. Guillaume, Strategies for data dissemination to mobile sinks in wireless sensor networks, IEEE Wireless Commun. 15 (6) (2008) 31–37.
[8] M. Hu, T. Wu, J.D. Weir, An adaptive particle swarm optimization with multiple adaptive methods, IEEE Trans. Evol. Comput. 17 (5) (2013) 705–720.
[9] Y.F. Hu, Y.S. Ding, K.R. Hao, An immune cooperative particle swarm optimization algorithm for fault-tolerant routing optimization in heterogeneous wireless sensor networks, Math. Probl. Eng. 2012 (2012) 1–19.
[10] S. Hong, D. Kim, M. Ha, S. Bae, S.J. Park, W. Jung, J.E. Kim, SNAIL: an IP-based wireless sensor network approach to the internet of things, IEEE Wireless Commun. 17 (6) (2010) 34–42.
[11] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, F. Silva, Directed diffusion for wireless sensor networking, IEEE/ACM Trans. Network. 11 (1) (2003) 2–16.
[12] K. Karenos, V. Kalogeraki, Traffic management in sensor networks with a mobile sink, IEEE Trans. Parallel Distrib. Syst. 21 (10) (2010) 1515–1530.
[13] J.W. Kim, J.S. In, K. Hur, J.W. Kim, D.S. Eom, An intelligent agent-based routing structure for mobile sinks in WSNs, IEEE Trans. Consumer Electron. 56 (4) (2010) 2310–2316.
[14] K.B. Lee, J.H. Kim, Multiobjective particle swarm optimization with preference-based sort and its application to path following footstep optimization for humanoid robots, IEEE Trans. Evol. Comput. 17 (6) (2013) 755–766.
[15] X. Liang, Y.S. Ding, L.H. Ren, K.R. Hao, Y.L. Jin, Data-driven cooperative intelligent controller based on the endocrine regulation mechanism, IEEE Trans. Control Syst. Technol. 22 (1) (2014) 94–101.
[16] W.D. Liao, J.Y. Wang, J.F. Wang, Nonlinear inertia weight variation for dynamic adaptation in particle swarm optimization, Adv. Swarm Intell. 6728 (2011) 80–85.
[17] C.J. Lin, C.H. Chen, C.T. Lin, A hybrid of cooperative particle swarm optimization and cultural algorithm for neural fuzzy networks and its prediction applications, IEEE Trans. Syst., Man Cybernet., Part C: Appl. Rev. 39 (1) (2009) 55–68.
[18] H. Luo, F. Ye, TTDD: two-layer data dissemination in large-scale wireless sensor networks, ACM Wireless Networks 11 (1–2) (2005) 161–175.
[19] M. Ma, Y. Yang, M. Zhao, Tour planning for mobile data-gathering mechanisms in wireless sensor networks, IEEE Trans. Veh. Technol. 62 (4) (2013) 1472–1483.
[20] G. Maia, D.L. Guidoni, A.C. Viana, A distributed data storage protocol for heterogeneous wireless sensor networks with mobile sinks, Ad Hoc Networks 11 (1) (2013) 1588–1602.
[21] F. Marcelloni, M. Vecchio, Enabling energy-efficient and lossy-aware data compression in wireless sensor networks by multi-objective evolutionary optimization, Inform. Sci. 180 (10) (2010) 1924–1941.
[22] M. Marta, M. Cardei, Improved sensor network lifetime with multiple mobile sinks, Pervasive Mobile Comput. 5 (5) (2009) 542–555.
[23] H.S. Mo, E. Lee, S. Park, S.H. Kim, Virtual line-based data dissemination for mobile sink groups in wireless sensor networks, IEEE Commun. Lett. 17 (9) (2013) 1864–1867.
[24] Y.V. Pehlivanoglu, A new particle swarm optimization method enhanced with a periodic mutation strategy and neural networks, IEEE Trans. Evol. Comput. 17 (3) (2013) 436–452.
[25] A. Ratnaweera, S.K. Halgamure, H.C. Watson, Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients, IEEE Trans. Evol. Comput. 8 (3) (2004) 240–255.
[26] M. Saleem, G.A.D. Caro, M. Farooq, Swarm intelligence based routing protocol for wireless sensor networks: survey and future directions, Inform. Sci. 181 (20) (2011) 4597–4624.
[27] A. Sharifkhani, N.C. Beaulieu, A mobile-sink-based packet transmission scheduling algorithm for dense wireless sensor networks, IEEE Trans. Veh. Technol. 58 (5) (2009) 2509–2518.
[28] S. Tian, S.M. Shatz, Y. Yu, J. Li, Querying sensor networks using ad hoc mobile devices: a two-layer networking approach, Ad Hoc Networks 7 (5) (2009) 1014–1034.
[29] C.K. Ting, C.C. Liao, A memetic algorithm for extending wireless sensor network lifetime, Inform. Sci. 180 (24) (2010) 4818–4833.
[30] J. Yang, H.S. Zhang, Y. Ling, C. Pan, W. Sun, Task allocation for wireless sensor network using modified binary particle swarm optimization, IEEE Sensors J. 14 (3) (2014) 882–892.
[31] F. Yu, E. Lee, S. Park, A simple location propagation scheme for mobile sink in wireless sensor networks, IEEE Commun. Lett. 14 (4) (2010) 321–323.
[32] F. Zou, D.B. Chen, J.T. Wang, Multi-objective PSO based on infeasibility degree and principle of endocrine, J. Comput. Appl. 30 (7) (2010) 1885–1888.