Engineering Applications of Artificial Intelligence 33 (2014) 127–140

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

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

Energy efficient clustering and routing algorithms for wireless sensor networks: Particle swarm optimization approach

Pratyay Kuila*, Prasanta K. Jana

Department of Computer Science and Engineering, Indian School of Mines, Dhanbad 826004, India

ARTICLE INFO

Article history:
Received 21 November 2013
Received in revised form
20 February 2014
Accepted 24 April 2014

Keywords:
Wireless sensor networks
Routing
Clustering
Gateways
Network life
Particle swarm optimization

ABSTRACT

Energy efficient clustering and routing are two well known optimization problems which have been studied widely to extend lifetime of wireless sensor networks (WSNs). This paper presents Linear/Nonlinear Programming (LP/NLP) formulations of these problems followed by two proposed algorithms for the same based on particle swarm optimization (PSO). The routing algorithm is developed with an efficient particle encoding scheme and multi-objective fitness function. The clustering algorithm is presented by considering energy conservation of the nodes through load balancing. The proposed algorithms are experimented extensively and the results are compared with the existing algorithms to demonstrate their superiority in terms of network life, energy consumption, dead sensor nodes and delivery of total data packets to the base station.

1. Introduction

1.1. Background

With the proliferation of soft computing techniques, nature-inspired algorithms have drawn enormous attention among researchers. Such algorithms have been studied to solve many optimization problems. For examples, genetic algorithm (GA) has been applied to enhance the efficiency of construction automation system (Wi et al., 2012). Similarly, particle swarm optimization (PSO) has been applied to solve various optimization problem in manufacturing (Issam et al., 2013; Thitipong and Nitin, 2011). Clustering and routing are two well known optimization problems which are well researched for developing many nature-inspired algorithms (Saleem et al., 2011; Kulkarni et al., 2011) in the field of wireless sensor networks (WSNs). PSO (Kennedy and Eberhart, 1995) is one such metaheuristic technique that has gained immense popularity in the recent years. In this paper, the authors propose two PSO-based algorithms for clustering and routing in wireless sensor networks.

A WSN consists of a large number of tiny and low power sensor nodes, which are randomly or manually deployed across an unattended target area. WSNs have potential applications in environment monitoring, disaster warning systems, health care, defense reconnaissance, and surveillance systems (Akyildiz et al., 2002). However, the main constraint of the WSNs is the limited power sources of the sensor nodes. Therefore, energy conservation of the sensor nodes is the most challenging issue for the long run operation of WSNs. Various issues have been studied for this purpose that include low-power radio communication hardware (Calhoun et al., 2005), energy-aware medium access control (MAC) layer protocols (Ahmad et al., 2012; Aykut et al., 2011), etc. However, energy efficient clustering and routing algorithms (Abbasi and Mohamad, 2007; Kemal and Mohamed, 2005) are the most promising areas that have been studied extensively in this regard.

In a two-tier WSN, sensor nodes are divided into several groups called clusters. Each cluster has a leader known as cluster head (CH). All the sensor nodes sense local data and send it to their corresponding CH. Then the CHs aggregate the local data and finally send it to the base station (BS) directly or via other CHs. The functionality of a cluster-based WSN is shown in Fig. 1. Clustering sensor nodes has the following advantages: (1) It enables data aggregation at cluster head to discard the redundant and uncorrelated data; thereby, it saves energy of the sensor nodes. (2) Routing can be more easily managed because only CHs need to maintain the local route set up of other CHs and thus require small routing information; this in turn improves the scalability of the network significantly. (3) It also conserves communication bandwidth as the sensor nodes communicate with their CHs only and thus avoid exchange of redundant messages among themselves.

However, CHs bear some extra work load contributed by their member sensor nodes as they receive the sensed data from their member sensor nodes, aggregate them and communicate it to

* Corresponding author.
E-mail addresses: pratyay_kuila@yahoo.com (P. Kuila), prasantajana@yahoo.com (P.K. Jana).

http://dx.doi.org/10.1016/j.engappai.2014.04.009
0952-1976/

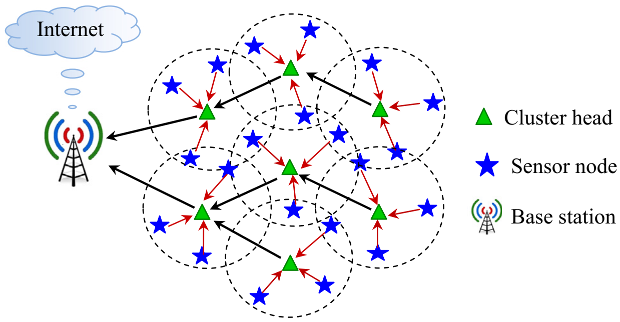

Fig. 1. A wireless sensor network model.

the BS. Moreover, in many WSNs, the CHs are usually selected amongst the normal sensor nodes which can die quickly for this extra work load. In this context, many researchers (Gupta and Younis, 2003; Low et al., 2008; Ataul et al., 2009; Kuila and Jana, 2012a; Kuila et al., 2013) have proposed the use of some special nodes called gateways, which are provisioned with extra energy. These gateways act like cluster heads and are responsible for the same functionality of the CHs. Therefore, gateways and CHs are used interchangeably in the remainder of the paper.

Unfortunately, the gateways are also battery-operated and hence power constrained. Lifetime of the gateways is very crucial for the long run operation of the network. It is noteworthy that the transmission energy (E) which mainly dominates the overall energy consumption is proportional to the distance (d) between transmitter and receiver, i.e., $E \propto d^{\lambda}$, where λ is the path loss exponent and $2 \le \lambda \le 4$ (Habib and Sajal, 2008). Therefore, minimization of transmission distance can reduce the energy consumption. However, some applications are very time-critical in nature. Hence, they should satisfy strict delay constraints so that the BS can receive the sensed data within a specified time bound. But the delay is proportional to the number of forwards on the dissemination path between a source and the BS. In order to minimize the delay, it is necessary to minimize the number of forwards, which can be achieved by maximizing the distance between consecutive forwards. Therefore, while designing routing algorithms we need to incorporate a trade-off between transmission distance and number of forwards as they pose two conflicting objectives. Furthermore, load balancing is another important issue for WSN clustering. Particularly, this is a pressing issue when the sensor nodes are not distributed uniformly. In this paper we address the following problems:

(1) Energy efficient routing with a trade-off between transmission distance and number of data forwards.
(2) Energy efficient load balanced clustering with energy conservation of the WSN.

Note that given n sensor nodes and m gateways, the number of possible clusters is m^n. It should also be noted that if the gateways have an average of d valid one-hop neighbor relay nodes, then the number of valid routes is d^m. Therefore the computational complexity of finding the optimal route and cluster for a large WSN seems to be very high by a brute force approach. Moreover, an optimization method requires reasonable amount of memory and computational resources and yet finding out good results is desirable. In order to obtain a faster and efficient solution of the clustering and routing problem with the above issues, a metaheuristic approach such as particle swarm optimization (PSO) is highly desirable. The main objective of this paper is to develop an efficient PSO-based clustering and routing algorithms for WSNs with the consideration of energy consumption of the sensor nodes for prolonging network life time.

1.2. Authors' contribution

In this paper, first Linear Programming (LP) and Non-linear Programming (NLP) formulations are presented for the routing and clustering problems respectively. Then two PSO-based algorithms for the same are proposed. The PSO-based routing builds a trade-off between energy consumption of the CHs and delay in forwarding the data packets. It finds out a route from all the gateways to the base station which has comparably lower overall distance with less number of data forwards. We present an efficient particle encoding scheme for complete routing solution and design the multi-objective fitness function using weighted sum approach.

The proposed PSO-based clustering takes care of energy consumption of the normal sensor nodes as well as the gateways. For clustering, particles are cleverly encoded to produce complete clustering solution. A different fitness function is also used by taking care of those gateways which inevitably consumes more energy by acting as relay node in packet forwarding. We perform extensive simulation on the proposed methods and evaluate them with several performance metrics including network life-time, number of active sensor nodes, energy consumption, total number of packets delivery and so on. The results are compared with GA-based clustering (Kuila et al., 2013), GLBCA (Low et al., 2008) and LDC (Ataul et al., 2008). Our main contributions can be summarized as follows:

- LP and NLP formulations for the routing and clustering problems respectively.
- PSO-based routing algorithm with a trade-off between transmission distance and number of data forwards with efficient particle encoding scheme for complete routing solution and derivation of efficient multi-objective fitness function.
- PSO-based clustering algorithm with efficient particle encoding scheme and fitness function.
- Simulation of the proposed algorithm to demonstrate superiority over some existing algorithms.

The rest of the paper is organized as follows. The related work is presented in Section 2. An overview of particle swarm optimization is given in Section 3. The system model and used terminologies are described in Section 4 which includes energy model and network model. The proposed algorithms and the experimental results are presented in Sections 5 and 6 respectively and we conclude in Section 7.

2. Related works

A number of clustering and routing algorithms have been developed for WSNs. We present the review of such works based on heuristic and metaheuristic approaches. However, we emphasize on the metaheuristic approach as our proposed algorithm is based on it.

2.1. Heuristic approaches

Low et al. (2008) have proposed a clustering algorithm by considering a breadth-first search (BFS) tree of the sensor nodes to find out the least loaded gateway for assigning a sensor node to a CH. The algorithm has the time complexity of $O(mn^2)$ for n sensor nodes and m CHs. For a large scale WSN, it seems that execution time of this algorithm is very high. Their algorithm also takes substantial amount of memory space for building a BFS tree for individual sensor node. We have proposed a load balanced clustering algorithm (Kuila and Jana, 2014) that runs in $O(n \log n)$

which is an improvement over Low et al. (2008). Gupta and Younis (2003) have proposed a clustering algorithm called LBC, which takes $O(mn \log n)$ time in worst case. Kuila and Jana (2012b) have proposed an energy efficient load-balanced clustering algorithm (EELBCA) with $O(n \log m)$ time. EELBCA addresses energy efficiency as well as load balancing. EELBCA is a min-heap based clustering algorithm. A min-heap is build using cluster heads (CHs) on the number of sensor nodes allotted to the CHs. However, the algorithms do not consider residual energy of the sensor nodes.

Many heuristics have also been proposed for routing in WSNs. LEACH (Heinzelman et al., 2002) is a popular cluster-based routing algorithm that dynamically rotates the work load of the CHs amongst the sensor nodes which is useful for load balancing. However, the main disadvantage of this approach is that a node with very low energy may be selected as a CH which may die quickly. Moreover, the CHs communicate with base station via single-hop which is impractical for WSNs with large coverage area. Therefore, a large number of algorithms have been developed to improve LEACH which can be found in Tyagi and Kumar (2013), Al-Refai et al. (2011). Kuila and Jana (2012c) have proposed a cost-based distributed energy balanced clustering and routing algorithm for CH selection and cluster formation. But, the algorithm suffers from the connectivity problem of the selected CHs.

2.2. Metaheuristic approaches

A number of metaheuristic based clustering algorithms have been reported for WSNs. However, most of them have dealt with CH selection only. Recently, we have proposed a GA-based load balanced clustering algorithm for WSNs (Kuila et al., 2013). The algorithm forms clusters in such way that the maximum load of each gateway is minimized and it works for both the equal and unequal load of the sensor nodes. The algorithm has faster convergence and better load balancing than the traditional GA (Goldberg, 2007). However, it has the demerit that the CHs directly communicate with the BS which may not be realistic for large area networks. Moreover, the algorithm does not consider residual energy of the sensor nodes and gateways in cluster formation which may lead to imbalance energy consumption of the sensor nodes.

Ataul et al. (2009) have proposed a GA-based algorithm for data routing between gateways in a two-tire wireless sensor network. Selection of individuals is carried out using the Roulette-wheel selection method and the fitness function is defined by network lifetime in terms of rounds. We have also proposed GA-based routing algorithm called GAR where the overall communication distance from the gateways to the BS is minimized (Gupta et al., 2013). However, both of the algorithms (Ataul et al., 2009; Gupta et al., 2013) consider only routing of aggregated data from the gateways to the BS without considering data communication from the sensor nodes to the gateways within each cluster. Enan et al. (2011) have presented an evolutionary aware routing protocol (EAERP) for dynamic clustering of wireless sensor networks. Here the authors have made an attempt to minimize the energy consumption throughout the network by choosing a set of efficient cluster heads from the normal sensor nodes and all non-CH sensor nodes determine nearest CH to join. EAERP suffers same problem as LEACH, as some sensor node may become a CH which may not have sufficient energy. Moreover, EAERP requires re-clustering in each round to rotate the extra work load of CH. Unfortunately, being a centralized approach; EAERP requires whole network information in each round for re-clustering. Chakraborty et al. (2012) have presented a differential evolution based routing algorithm for more than a 1000 relay nodes such that the energy consumption of the maximum energy-consuming relay node is minimized. However, the authors do not take care about the cluster formation. Some improper clustering may lead to serious energy inefficiency of the relay nodes. Singh and Lobiyal (2012) and Abdul et al. (2007) have used the PSO for CH selection amongst the normal sensor nodes and do not take care of the cluster formation. PSO and ant colony optimization (ACO) are used in WSNs for other optimization problems also and they can be found in Saleem et al. (2011), Kulkarni et al. (2011), Zungeru et al. (2012).

However, none of the above algorithms consider the overhead of the data routing in cluster formation phase. Even, none of them except Kuila et al. (2013) focus on cluster formation using nature-inspired approach. Many works have been proposed for CH selection. However, selection of the CHs merely cannot form the clusters. To the best of our knowledge, there is no nature-inspired clustering algorithm such as PSO which considers cluster formation rather than CH selection for WSNs.

3. Overview of particle swarm optimization

Particle swarm optimization (PSO) is inspired by natural life, like bird flocking, fish schooling and random search methods of evolutionary algorithm (Kennedy and Eberhart, 1995; Wei and Nor, 2014). It can be observed from the nature that animals, especially birds, fishes, etc. always travel in a group without colliding. This is because each member follows the group by adjusting its position and velocity using the group information. Thus it reduces individual's effort for searching of food, shelter etc. The various steps of a PSO are depicted in the flowchart as shown in Fig. 2.

PSO consists of a swarm of a predefined size (say N_P) of particles. Each particle gives a complete solution to the multidimensional optimization problem. The dimension D of all the particles is equal. A particle P_i, $1 \le i \le N_P$ has position X_{id}, $1 \le d \le D$ and velocity V_{id} in the dth dimension of the hyperspace. We adopt the notation for representing the ith particle P_i of the population as follows:

$$P_i = [X_{i,1}, X_{i,2}, X_{i,3}, ..., X_{i,D}] \tag{3.1}$$

Each particle is evaluated by a fitness function to judge the quality of the solution to the problem. To reach up to the global best position, the particle P_i follows its own best, i.e., personal best called $Pbest_i$ and global best called $Gbest$ to update its own velocity and position. In each iteration, its velocity V_{id} and position X_{id} in the dth dimension is updated using the following equations

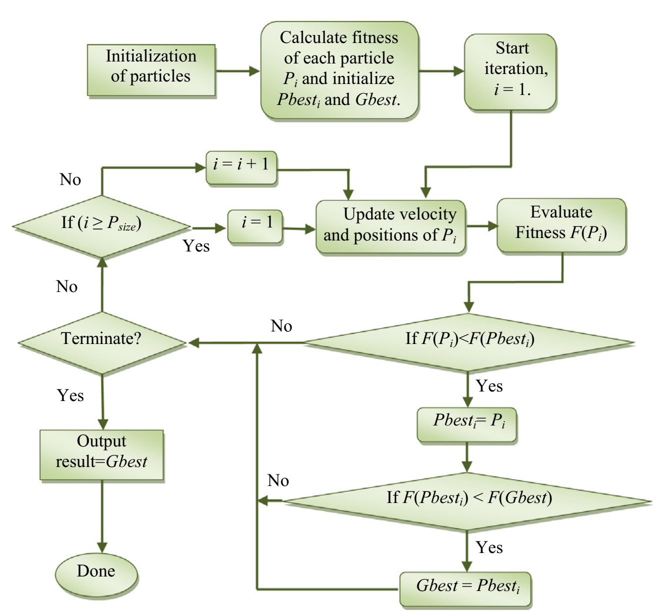

Fig. 2. Flowchart of the PSO.

respectively:

$$V_{i,d}(t) = w \times V_{i,d}(t-1) + c_1 \times r_1 \times (Xpbest_{i,d} - X_{i,d}(t-1)) + c_2 \times r_2 \times (Xgbest_d - X_{i,d}(t-1)) \tag{3.2}$$

$$X_{i,d}(t) = X_{i,d}(t-1) + V_{i,d}(t) \tag{3.3}$$

where w is the inertial weight, c_1 and c_2 are two non-negative constants called acceleration factor and r_1 and r_2 are two different uniformly distributed random numbers in the range [0,1]. The update process is iteratively repeated until either an acceptable *Gbest* is achieved or a fixed number of iterations t_{max} is reached.

4. System model and terminologies

4.1. Energy model

The radio model for energy used in this paper is same as discussed by Heinzelman et al. (2002). In this model, both the free space and multi-path fading channels are used depending on the distance between the transmitter and receiver. When the distance is less than a threshold value d_0, then the free space (*fs*) model is used, otherwise, the multipath (*mp*) model is used. Let E_{elec}, ε_{fs} and ε_{mp} be the energy required by the electronics circuit and by the amplifier in free space and multipath respectively. Then the energy required by the radio to transmit an *l*-bit message over a distance *d* is given as follows:

$$E_T(l,d) = \begin{cases} lE_{elec} + l\varepsilon_{fs}d^2 & \text{for} \quad d < d_0 \\ lE_{elec} + l\varepsilon_{mp}d^4 & \text{for} \quad d \ge d_0 \end{cases} \tag{4.1}$$

The energy required by the radio to receive an *l*-bit message is given by

$$E_R(l) = lE_{elec} \tag{4.2}$$

The E_{elec} depends on several factors such as digital coding, modulation, filtering, and spreading of the signal, whereas the amplifier energy, $\varepsilon_{fs}d^2/\varepsilon_{mp}d^4$, depends on the distance between the transmitter and the receiver and also on the acceptable bit-error rate. It should be noted that this is a simplified model. In general, radio wave propagation is highly variable and difficult to model.

4.2. Network model

We assume a WSN model where all the sensor nodes are deployed randomly along with a few gateways and once they are deployed, they become stationary. A sensor node can be assigned to any gateway if it is within the communication range of the sensor node. Therefore, there are some pre-specified gateways onto which a particular sensor node can be assigned. Thus each sensor node has a list of gateways and it can be assigned to only one gateway amongst them. Similar to LEACH, the data gathering operation is divided into rounds. In each round, all sensor nodes collect the local data and send it to their corresponding CH (i.e., gateway). On receiving the data, the gateways aggregate them to discard the redundant and uncorrelated data and send the aggregated data to the base station via other CH as a next hop relay node. Between two adjacent rounds, all nodes turn off their radios to save energy. All communication is over wireless link. A wireless link is established between two nodes only if they are within the communication range of each other. Current implementation supports TDMA (Baronti et al., 2007) protocol to provide MAC layer communication. Gateways use slotted CSMA/CA MAC protocol to communicate with base station (Baronti et al., 2007).

Various definition of the network life is given in the literature (Dietrich and Dressler, 2006; Madan et al., 2005), such as this is the time until first node dies, the time until last node dies or the time until a desired percentage of nodes die. Moreover in some scenario (Cardai and Du, 2005) network life is considered as the period until the entire region is covered. Pan et al. (2003) have defined several metrics for network life time, e.g., *N*-of-*N* lifetime, *K*-of-*N* lifetime and *m*-in-*K*-of-*N* lifetime. *N*-of-*N* lifetime means the time duration until first gateway dies. *K*-of-*N* lifetime means survival of the network until *K* gateways out of *N* are alive and *m*-in-*K* of *N* lifetime means the time duration until all *m* supporting gateways and overall a minimum of *K* gateways are alive. In this paper, we use *N*-of-*N* lifetime.

4.3. Terminologies

We use the following terminologies in the proposed algorithm:

(1) The set of sensor nodes is denoted by $S=\{s_1, s_2, \ldots, s_N\}$.
(2) The set of gateways is denoted by $\xi=\{g_1, g_2, \ldots, g_M\}$ and g_{M+1} indicates the base station (BS).
(3) $L(i)$ denotes the lifetime of the gateway g_i. If g_i has residual energy $E_{residual}(g_i)$ and energy consumption per round $E_{Gateway}(g_i)$ then $L(i)$ can be calculated as follows:

$$L(i) = \left\lfloor \frac{E_{residual}(g_i)}{E_{Gateway}(g_i)} \right\rfloor \tag{4.3}$$

(4) d_{max} denotes the maximum communication range of the gateways.
(5) $dis(s_i, s_j)$ denotes the distance between s_i and s_j.
(6) $ComCH(s_i)$ is the set of all those gateways, which are within the communication range (R_S) of sensor node s_i. In other words,

$$ComCH(s_i) = \{g_j | dis(s_i, g_j) \le R_S \wedge g_j \in \xi\} \tag{4.4}$$

Therefore, s_i can be assigned to any one of the gateway from $ComCH(s_i)$, where $ComCH(s_i) \subseteq \xi$.
(7) $Com(g_i)$: The set of gateways, which are within communication range of g_i. The BS may also be a member of $Com(g_i)$. In other words,

$$Com(g_i) = \{g_j | \forall g_j \in \{\xi + g_{M+1}\} \wedge dis(g_i, g_j) < d_{max}\} \tag{4.5}$$

(8) $PNextHops(g_i)$: The set of gateways those might be selected as a next hop relay of g_i. The next hop relay node must be towards the BS. Therefore,

$$PNextHops(g_i) = \{g_j | \forall g_j \in \{Com(g_i) - g_{M+1}\} \wedge dis(g_j, g_{M+1}) \le dis(g_i, g_{M+1})\} \tag{4.6}$$

(9) $NextHop(g_i)$ is the gateway g_j, $g_j \in PNextHops(g_i)$ which is selected as next-hop relay node from g_i towards BS in data routing phase. Here, the next hop node may be the BS when BS is within communication range of g_i.
(10) $HopCount(g_i)$ denotes the number of next hops required to reach to the BS from g_i. If g_i directly communicates with BS, then $HopCount(g_i)$ is one. Therefore, $HopCount(g_i)$ can be recursively defined as

$$HopCount(g_i) = \begin{cases} 1, & NextHop(g_i) = g_{M+1} (\text{i.e., BS.}) \\ 1 + HopCount(g_j), & NextHop(g_i) = g_j (\text{i.e., other than BS.}) \end{cases} \tag{4.7}$$

(11) A delay is the time elapsed between the departure of a collected data packet from a source (say g_i) and its arrival to

the BS (Habib and Sajal, 2008). A delay $D(g_i)$ includes the average value of queuing delay (say d_q) per intermediate data disseminator, transmission delay (say d_t), and propagation delay (say d_p). So, $D(g_i)$ is formulated as

$$D(g_i) = (d_q + d_t + d_p) \times HopCount(g_i)$$
$$\text{i.e.,} \quad D(g_i) = K \times HopCount(g_i) \tag{4.8}$$

where $K = d_q + d_t + d_p$, which is a constant for a particular network (Habib and Sajal, 2008). Therefore, minimizing delay is equivalent to minimizing the hop count.

(12) Maximum distance (*MaxDist*) between two nodes in the routing path can be defined by

$$MaxDist = Max\{dis(g_i, NextHop(g_i)) | \forall i, 1 \le i \le M, g_i \in \xi\} \tag{4.9}$$

(13) Maximum hop count (*MaxHop*) of the gateways can be defined by

$$MaxHop = Max\{HopCount(g_i) | \forall i, 1 \le i \le M, g_i \in \xi\}. \tag{4.10}$$

5. Proposed algorithms

Network setup is performed in three phases: bootstrapping; route setup; and clustering. During the bootstrapping process, all the sensor nodes and gateways are assigned unique IDs. Then the sensor nodes and the gateways broadcast their IDs using CSMA/CA MAC layer protocol. Therefore, the gateways can collect the IDs of the sensor nodes and the other gateways those are within their communication range and finally send the local network information to the base station. Now, using the received information of the network, base station executes the routing and clustering algorithm. Note that after execution of the routing algorithm, the base station uses the final route setup for proper formation of the cluster. When the routing and clustering is over, all the gateways are informed about their next hop relay node towards the base station and the sensor nodes are also informed about the ID of the gateway they belong to. Then the gateways provide a TDMA schedule to their member sensor nodes for intra cluster communication. Gateways use slotted CSMA/CA MAC protocol to communicate with its next hop relay node. Now, we present our proposed (1) routing and (2) clustering algorithms as follows.

5.1. PSO based routing

5.1.1. LP formulation for routing problem

Now, we address the routing problem where our main objective is to minimize the maximum transmission distance between two nodes in the routing path and maximum hop count. Let a_{ij} be a Boolean variable defined as

$$a_{ij} = \begin{cases} 1, & \text{If } NextHop(g_i) = g_j \\ 0, & \text{otherwise.} \end{cases} \tag{5.1}$$

Then the Linear Programming (LP) of the routing problem is formulized as follows:

Minimize $W = \alpha \times MaxDist + \beta \times MaxHop$

Subject to

$$\sum_{j=1}^{M+1} a_{ij} = 1, 1 \le i \le M, \forall g_i \in \xi, \forall g_j \in \{\xi + g_{M+1}\} \quad \text{and} \quad i \ne j \tag{5.2}$$

$$dis(g_i, g_j) \times a_{ij} \le d_{max}, 1 \le i \le M, \forall g_i \in \xi, \forall g_j \in \{\xi + g_{M+1}\} \quad \text{and} \quad i \ne j \tag{5.3}$$

$$\alpha = 1 - \beta \quad \text{and} \quad 0 < \beta < 1 \tag{5.4}$$

The constraint (5.2) ensures that the gateway g_i, $\forall i$, $1 \le i \le M$ forwards its data to only one next hop node g_j and the constraint (5.3) ensures that the selected next hop node is within the transmission range. α and β are two control parameters. α controls the total path distance and β controls the total hop count. The constraint (5.4) defines the range of α and β.

5.1.2. Proposed routing algorithm

We now present our PSO-based routing algorithm which consists of particle initialization and determination of fitness function followed by the velocity and position update phase as follows.

5.1.2.1. Initialization of particles. We represent the particles in such a way that each particle provides the route from each CH to the BS. The dimension of the particles is same and equal to the number of gateways (i.e., M) in the network. We initialize each component, i.e., $X_{i,d}$, $1 \le i \le N_P$, $1 \le d \le M$ with a randomly generated uniformly distributed number $Rand(0,1]$, $0 < Rand(0,1] \le 1$. The value of the dth component (i.e., $X_{i,d}$) maps a gateway (say g_k) as a next hop relay towards BS from g_d. Therefore, $X_{i,d} = Rand(0,1]$ maps a gateway (say g_k), indicating that g_d send data to g_k. The mapping is done as follows:

$$g_k = Index(PNextHops(g_d), n) \tag{5.5}$$

where $Index(PNextHops(g_d), n)$ is an indexing function that indexes the nth gateway from $PNextHops(g_d)$ and

$$n = Ceiling(X_{i,d} \times |PNextHops(g_d)|)$$

***Illustration* 5.1:** Consider a WSN with 10 gateways, i.e., $\xi = \{g_1, g_2, \ldots, g_{10}\}$ as shown in Fig. 3(a). Therefore, the dimension of the particles is same as the number of gateways, i.e., $M = 10$. Consider the directed acyclic graph $G(V,E)$ shown in Fig. 3(a), where E represents the set of edges. The edge $g_i \to g_j$ indicates that g_i can use g_j as a next hop relay towards BS. Here, g_j is closer to BS than g_i and also g_j is within communication range of g_i. It can be observed form Fig. 3(a) that the gateway g_3 can use any one of the three gateways amongst $\{g_5, g_6, g_7\}$ as a next hop relay node towards the BS. In other words, $PNextHops(g_3) = \{g_5, g_6, g_7\}$. Table 1 shows the gateways and their possible next hop node (gateway/base station) as per Fig. 3(a).

Now, for each $X_{i,d}$, $1 \le d \le M$ of the particle P_i, a random number is generated to initialize it. Let us assume that a particle $P_i = [0.38, 0.63, 0.46, 0.17, 0.86, 0.73, 0.94, 0.81, 0.34, 0.13]$ has been randomly generated as shown in the second column (i.e., $X_{i,d}$) of Table 2.

We show that this particle actually represents the complete solution of the routing problem as follows. Let consider the second element, i.e., $X_{i,2} = 0.63$. Therefore, $Ceiling(X_{i,2} \times |PNextHops(g_2)|) = 2$, which implies that the 2nd gateway from $PNextHops(g_2)$ is selected as next hop relay node of g_2. Thus g_3 (2nd gateway from $PNextHops(g_2)$) serves as the next hop relay for g_2. In the same way all the gateways are given a next hop relay using the randomly generated particle. Thus, the above randomly generated particle P_i maps the complete routing solution from each CH as shown in the fifth column (i.e., *Next-hop*) of Table 2. Therefore, the route from g_1 to the base station can be expressed as the path $g_1 \to g_3 \to g_7 \to g_{10} \to BS$. The final routing schedule for the sub-graph network is shown in Fig. 3(b).

5.1.2.2. Fitness function derivation. Now, we construct a fitness function to evaluate the individual particle of the population. This helps us to periodically update the personal best and global best of the particles. We have two objectives in our proposed algorithm. The first objective is to minimize the maximum distance between two nodes and the second objective is to minimize the maximum number of hops used by the gateways.

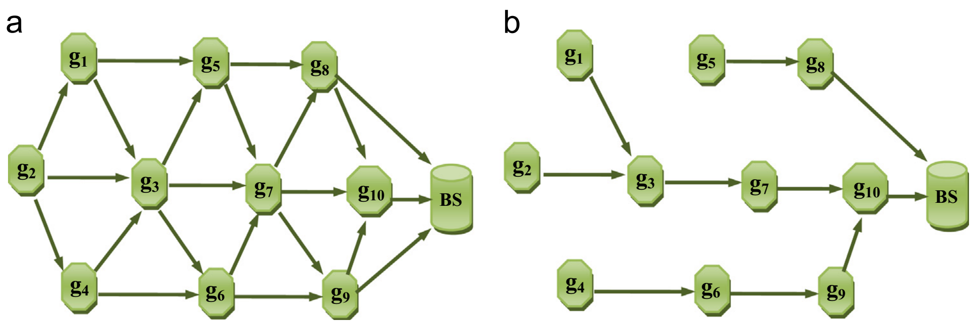

Fig. 3. (a) Sub graph network of a WSN and (b) routing path.

Table 1
Gateways and their possible next hops.

Gateways	*PNextHops* (g_d)	|*PNextHops* (g_d)|
g_1	$\{g_3, g_5\}$	2
g_2	$\{g_1, g_3, g_4\}$	3
g_3	$\{g_5, g_7, g_6\}$	3
g_4	$\{g_6, g_3\}$	2
g_5	$\{g_7, g_8\}$	2
g_6	$\{g_7, g_9\}$	2
g_7	$\{g_8, g_9, g_{10}\}$	3
g_8	$\{g_{10}, BS\}$	2
g_9	$\{g_{10}, BS\}$	2
g_{10}	$\{BS\}$	1

Table 2
Next-hop relay node selection from the randomly generated particle.

Gateways	$X_{i,d}$	|*PNextHops* (g_d)|	*Ceiling* ($X_{i,d}$ × |*PNextHops* (g_d)|)	*NextHop* (g_d)
g_1	0.38	2	1	g_3
g_2	0.63	3	2	g_3
g_3	0.46	3	2	g_7
g_4	0.17	2	1	g_6
g_5	0.86	2	2	g_8
g_6	0.73	2	2	g_9
g_7	0.94	3	3	g_{10}
g_8	0.81	2	2	BS
g_9	0.34	2	1	g_{10}
g_{10}	0.13	1	1	BS

Therefore, our two objectives are as follows:

$$\text{Objective 1}: \quad \text{Minimize } MaxDist = Max\{dis(g_i, NextHop(g_i)) \,|\, \forall i, 1 \le i \le M\} \tag{5.6}$$

$$\text{Objective 2}: \quad \text{Minimize } MaxHop = Max\{HopCount(g_i) \,|\, \forall i, 1 \le i \le M\} \tag{5.7}$$

It is noteworthy that the above two objectives conflict each other, i.e., lower distance of next hop relay node increases the hop count and vice versa. Therefore, optimization of one objective hampers in optimization of other. Our proposed work constructs the fitness function in such a way that a trade-off can be built with these conflicting objectives. We have used weight sum approach (WSA) (Konak et al., 2006) for the construction of the multi objective fitness function. WSA is a classical approach for solving the multi-objective optimization problem. In this approach, a weight value W_i is multiplied with each objective. Finally all the multiplied values are added to convert the multi objectives into a single scalar objective function as follows:

$$Fitness = W_1 \times MaxDist + W_2 \times MaxHops \tag{5.8}$$

Algorithm: PSO-Routing

Input: (1) Set of cluster heads $\xi = \{g_1, g_2..., g_M\}$.
(2) *PNextHops* (g_i) and *HopCount*(g_i), $\forall i, 1 \le i \le M$.
(3) Predefined swarm size N_P.
Output: Route $R : \xi \rightarrow \{\xi + g_{M+1}\}$.
==
Step 1: Initialize particles $P_i, \forall i, 1 \le i \le N_P$. /*As described in section 5.1.2*/
Step 2: ***for*** $i = 1$ to N_P ***do***
2.1: Calculate *Fitness*(P_i) /* Using equation 5.8.*/
2.2: $Pbest_i = P_i$
end
Step 3: $Gbest = \{Pbest_k \mid Fitness(Pbest_k) = \min(Fitness(Pbest_i), \forall i, 1 \le i \le N_P)\}$
Step 4: ***While*** **(!(*Terminate*))**
for $i = 1$ to N_P ***do***
4.1: Update velocity and position of P_i using equation 3.2 and 3.3
4.2: Calculate *Fitness*(P_i)
4.3: ***If*** $Fitness(P_i) < Fitness(Pbest_i)$ ***then***
$Pbest_i = P_i$
end
4.4: ***If*** $Fitness(Pbest_i) < Fitness(Gbest)$ ***then***
$Gbest = Pbest_i$
end
end
end
Step 5: Calculate *NextHop*(g_i), $\forall i, 1 \le i < M$, (i.e., route R) using *Gbest* as shown in illustration 5.1
Step 6: Stop.

Fig. 4. PSO based routing algorithm.

In our approach we have taken $W_2 = 1 - W_1$ and $0 < W_1 < 1$. Our objective is to minimize the fitness value. In other words,

$$\text{Objective}: \quad \text{Minimize } fitness \tag{5.9}$$

Therefore, lower the fitness value, the better is the particle position.

5.1.2.3. Velocity and position update. The velocity and the position are updated in each iteration using Eqs. (3.2) and (3.3) respectively. It is noteworthy that, the algebraic steps of addition and subtraction operation in Eqs. (3.2) and (3.3) may cause the new position of the particle to be negative or greater than one. In our scenario the position of the particle must satisfy the range [0,1]. Therefore, our algorithm should generate the positions of the particles in such a way that it can satisfy the range. This can be made if we choose the positions as follows:

- If new position is negative or zero, then replace the position value by a newly generated random number which tends to zero.
- If new position is greater than one, then replace the position value by one.

After getting the new position, the particle P_i is evaluated by the fitness function. Now, its personal best ($Pbest_i$) is replaced by itself, only if its current fitness value is better than its $Pbest_i$ fitness value. The updating process is as follows:

$$Pbest_i = \begin{cases} P_i & \text{if } (fitness(P_i) < fitness(Pbest_i)) \\ Pbest_i & \text{Otherwise.} \end{cases} \quad (5.10)$$

Now, the global best is also updated as follows:

$$Gbest = \begin{cases} P_i & \text{if } (fitness(P_i) < fitness(Gbest)) \\ Gbest & \text{Otherwise.} \end{cases} \quad (5.11)$$

The velocity and the positions are iteratively updated until the termination criteria are fulfilled. In our approach, the termination criterion is a predefined iteration number. After termination of the PSO-based routing algorithm, the particle *Gbest* represents the final routing solution. The algorithm is shown in Fig. 4.

5.2. PSO based clustering

5.2.1. NLP formulation for clustering problem

Now, we address the clustering problem where our basic objective is to maximize the lifetime of the network as well as minimize the energy consumption of the sensor nodes. By the network lifetime, we mean the time from the deployment of the WSN till the death of the first gateway. Therefore, network life can be maximized if we can maximize the minimum lifetime of the gateways. Energy consumption of the sensor nodes can be minimized by minimizing the distance between sensor nodes and their corresponding gateways. Let b_{ij} be a Boolean variable such that

$$b_{ij} = \begin{cases} 1 & \text{If sensor node } s_i \text{ is assigned to cluster head } g_j, \\ & \forall i,j : 1 \le i \le N, 1 \le j \le M \\ 0 & \text{Otherwise} \end{cases} \quad (5.12)$$

Let L be the minimum lifetime of the gateways, i.e., $L = \min\{L(i) \mid \forall i, 1 \le i \le M\}$ and *AvegDist* be the average distance between sensor nodes and their corresponding CH, i.e.,

$$AvegDist = \frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{M} dis(s_i, g_j) \times b_{ij} \quad (5.13)$$

Then the Non-linear Programming (NLP) of the clustering problem can be formulized as follows:

$$\text{Maximize} \quad Z = \frac{L}{AvegDist}$$

Subject to

$$\sum_{j=1}^{M} b_{ij} = 1, \quad 1 \le i \le N \quad (5.14)$$

$$\sum_{j=1}^{M} dis(s_i, g_j) \times b_{ij} \le d_{max}, 1 \le i \le N, s_i \in S, g_j \in \xi \quad (5.15)$$

The constraint (5.14) states that the sensor node s_i, $\forall i, 1 \le i \le N$ can be assigned to one and only one gateway. The constraint (5.15) ensures that the sensor nodes are assigned to the gateway within its communication range.

5.2.2. Proposed clustering algorithm

After executing the above routing algorithm, the base station executes the clustering algorithm in which the information of the routing solution is used for the cluster formation to balance the load of the CHs (i.e., gateways). Note that we use here particle initialization for clustering and fitness function different from that are used in the above routing algorithm as they do not fit for the proposed clustering.

5.2.2.1. Initialization of particles. Here, the dimension of the particle is same as the number of sensor nodes (i.e., N) in the network. Let, $P_i = [Y_{i,1}, Y_{i,2}, Y_{i,3}, \ldots, Y_{i,N}]$ be the ith particle of the population where each component, $Y_{i,d}$, $1 \le i \le N_P$, $1 \le d \le N$ maps the assignment of the sensor node s_d to a gateway. We initialize each component with a randomly generated uniformly distributed number $Rand(0,1]$, $0 < Rand(0,1] \le 1$. The random number is generated independently for each component. The component of the dth dimension of this particle, i.e., $Y_{i,d} = Rand(0,1]$, $1 \le d \le N$ maps a gateway (say g_k) to which the sensor node s_d is assigned. The mapping is done as follows:

$$g_k = Index(ComCH(s_d), n) \quad (5.16)$$

where $Index(ComCH(s_d),n)$ is an indexing function that indexes the nth gateway from $ComCH(s_d)$ and $n = Ceiling(Y_{i,d} \times |ComCH(s_d)|)$.

This is important to note that the above particle representation is a part of the clustering algorithm. As mentioned above that the dimension of each particle is equal to the number of the sensor nodes. Therefore, addition/deletion of any sensor node would change the particle dimension and require re-clustering.

Illustration 5.2: Consider a WSN with 15 sensor nodes and 6 gateways, i.e., $S = \{s_1, s_2, \ldots, s_{15}\}$ and $\xi = \{g_1, g_2, g_3, g_4, g_5, g_6\}$ as shown in Fig. 5(a). Therefore, the dimension of the particle is same

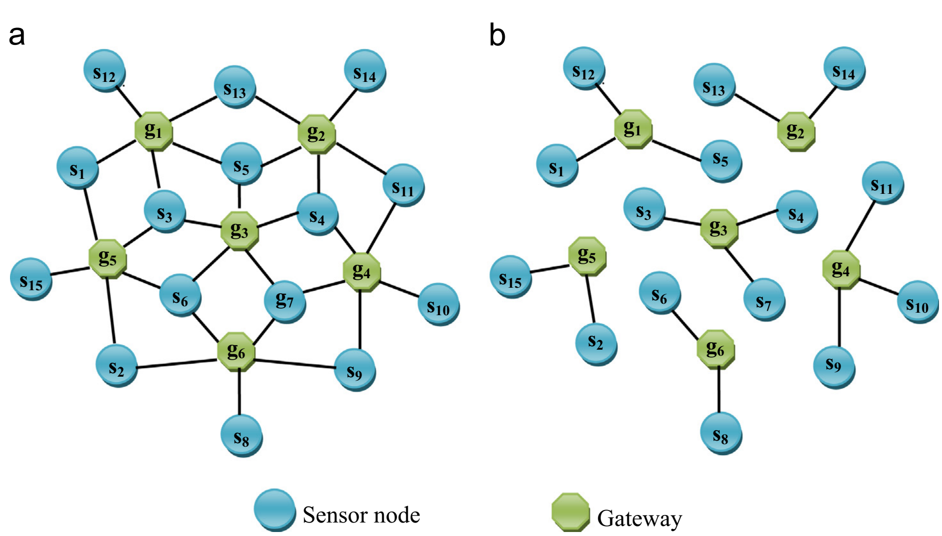

Fig. 5. A WSN with gateways (a) before clustering and (b) after clustering.

as the number of sensor nodes, i.e., $N=15$. The edges between the sensor nodes and the gateways indicate that the gateways are within communication range of the sensor nodes. It can be observed from Fig. 5(a) that the sensor node s_4 is connected with three gateways. In other words, $ComCH(s_4)=\{g_3, g_4, g_2\}$. Table 3 shows the sensor nodes and the gateways within its communication range.

Now, for each element of the ith particle at the Gth generation, a random number is generated to initialize the element. Let, the generated random number for the fourth element is 0.18, i.e., $Y_{i,4}=0.18$ as shown in Table 4. Hence,$Ceiling(Y_{i,4} \times |ComCH(s_4)|)=1$, therefore the 1st gateway from $ComCH(s_4)$, i.e., g_3 is selected for assigning s_4 as shown in Table 4. In the same way all the sensor nodes are assigned to a gateway using the randomly generated particle. The final assignment of the sensor nodes to their corresponding gateways are shown in Table 4.

Therefore, the particle $P_i=[0.26, 0.86, 0.91, 0.18, 0.77, 0.12, 0.47, 0.62, 0.24, 0.53, 0.71, 0.46, 0.81, 0.92, 0.39]$ maps the assignment of the sensor nodes to their gateways shown in Fig. 5(b). This is important to note that the different order of gateways under $ComCH$ may produce another assignment which is also valid.

5.2.2.2. Fitness function derivation. The fitness function is derived in such a way that it takes care of energy consumption of the CHs as well as the sensor nodes. The derivation depends on some parameters described as follows:

(A) ***Lifetime of the CHs:*** Our first objective is to maximize the network life. This can be possible if we can maximize the lifetime of the gateway that has least lifetime. The general principle behind the maximization of the gateway life is that the gateway with lower residual energy should have lower rate of energy consumption per round than the gateways with higher residual energy. Thus the lifetime of the gateways with lesser remaining energy can be prolonged effectively. The gateways consume their energy for receiving sensed data from their member sensor nodes, aggregation of data and finally to send the aggregated data to the base station. Therefore, energy consumption of a gateway g_i with n_i number of member sensor nodes due to inter-cluster activity in a single round can be formulated as follows:

$$E_{CLUSTER}(g_i)=n_i \times E_R+n_i \times E_{DA}+E_T(g_i, NextHop(g_i)) \quad (5.17)$$

where E_R, E_{DA} and E_T are the energy consumption due to data receiving, aggregation and transmission to BS respectively. Apart from these inter-cluster activities, in a multi-hop scenario, g_i also consumes its energy due to forwarding the data from the gateways whose routing path to the base station goes through g_i. Before calculating the energy consumption due to data forwarding, we need to calculate the total number of incoming packets those are coming from other gateways to g_i toward the base station. It can be recursively calculated as follows:

$$N_{IN}(g_i)=\begin{cases} 0, & \text{If } NextHop(g_j)\neq g_i, \forall g_j \in \xi \\ \sum\{N_{IN}(g_j) | NextHop(g_j)=g_i, g_j \in \xi\}, & \text{Otherwise} \end{cases} \quad (5.18)$$

The gateway g_i having $N_{IN}(g_i)$ number of incoming packets will consume its energy for receiving and transmitting these packets. The overall data forwarding energy consumption can be calculated as follows:

$$E_{FORWARD}(g_i)=N_{IN}(g_i) \times E_R+N_{IN}(g_i) \times E_T(g_i, NextHop(g_i)) \quad (5.19)$$

Therefore, total energy consumption of g_i can be calculated by adding Eqs. (5.17) and (5.19) as follows:

$$\begin{aligned} E_{Gateway}(g_i) &= E_{CLUSTER}(g_i)+E_{FORWARD}(g_i) \\ &= n_i \times E_R+n_i \times E_{DA}+E_T(g_i, NextHop(g_i)) \\ &\quad +N_{IN}(g_i) \times E_R+N_{IN}(g_i) \times E_T(g_i, NextHop(g_i)) \\ &= (n_i+N_{IN}(g_i)) \times E_R+n_i \times E_{DA}+(N_{IN}(g_i)+1) \\ &\quad \times E_T(g_i, NextHop(g_i)) \end{aligned} \quad (5.20)$$

Let, g_i has the residual energy of $E_{residual}(g_i)$. Then, lifetime of g_i can be calculated as

$$L(i)=\left\lfloor \frac{E_{residual}(g_i)}{E_{Gateway}(g_i)} \right\rfloor \quad (5.21)$$

Our first objective is to maximize the minimum lifetime of the CHs. In other words,

$$\text{Objective 1}: \quad \text{Maximize } L=\text{ Min }\{L(i) | \forall i, 1 \le i \le M\} \quad (5.22)$$

Therefore, larger the value of L, higher is the fitness value, i.e.,

$$Fitness \propto L \quad (5.23)$$

Remark 6.1. In Eqs. (5.18) and (5.19), we have calculated the number of incoming packets those are coming from the other gateways towards base station over the gateway g_i

Table 3
Sensor nodes with the list of possible gateways.

Sensor nodes	*ComCH* (s_d)	|*ComCH* (s_d)|
s_1	$\{g_1, g_5\}$	2
s_2	$\{g_6, g_5\}$	2
s_3	$\{g_1, g_5, g_3\}$	3
s_4	$\{g_3, g_4, g_2\}$	3
s_5	$\{g_3, g_2, g_1\}$	3
s_6	$\{g_6, g_5, g_3\}$	3
s_7	$\{g_4, g_3, g_6\}$	3
s_8	$\{g_6\}$	1
s_9	$\{g_4, g_6\}$	2
s_{10}	$\{g_4\}$	1
s_{11}	$\{g_2, g_4\}$	2
s_{12}	$\{g_1\}$	1
s_{13}	$\{g_1, g_2\}$	2
s_{14}	$\{g_2\}$	1
s_{15}	$\{g_5\}$	1

Table 4
Sensor nodes assignment from particle representation.

Sensor Nodes (s_d)	$Y_{i,d}$	|*ComCH* (s_d)|	*Ceiling* ($Y_{i,d} \times$ |*ComCH* (s_d)|)	Assigned Gateway
s_1	0.26	2	1	g_1
s_2	0.86	2	2	g_5
s_3	0.91	3	3	g_3
s_4	0.18	3	1	g_3
s_5	0.77	3	3	g_1
s_6	0.12	3	1	g_6
s_7	0.47	3	2	g_3
s_8	0.62	1	1	g_6
s_9	0.24	2	1	g_4
s_{10}	0.53	1	1	g_4
s_{11}	0.71	2	2	g_4
s_{12}	0.46	1	1	g_1
s_{13}	0.81	2	2	g_2
s_{14}	0.92	1	1	g_2
s_{15}	0.39	1	1	g_5

and the corresponding energy consumption respectively. As our first objective (Eq. (5.22)) is to maximize the lifetime of the network, the prior information of the data forwarding overhead of the gateways helps in the clustering phase to properly balance the load of the gateways and prolong the network life. Thus we have takes care of the gateways those inevitably deplete their energy due to serving as relay node in data routing phase.

(B) ***Average cluster distance*:** In order to maximize the lifetime of the gateways, some sensor nodes are forced to be assigned to the gateway which is farther from it. Thus these sensor nodes consume their energy faster and die quickly due to long haul communication with their CH. Therefore, we should also take care about the assignment of the sensor nodes to minimize their energy consumption. In order to minimize the energy consumption of the sensor nodes, they should be assigned to their nearest CH. Therefore, we have measured the average distance between sensor nodes and their corresponding CH and our second objective is to minimize it. In other words,

$$\text{Objective 2}: \quad \text{Minimize } AvegDist = \frac{1}{N}\sum_{i=1}^{N} dist(s_i, CH_i) \tag{5.24}$$

where CH_i is the cluster head of sensor node s_i. The shorter the *AvegDist*, the higher is the fitness value. Therefore, the fitness function is reversely proportional to the *AvegDist*, i.e.,

$$Fitness \propto \frac{1}{AvegDist} \tag{5.25}$$

Eqs. (5.23) and (5.25) combinedly implies that

$$Fitness \propto \frac{L}{AvegDist}$$

$$\text{i.e.,} \quad Fitness = K \times \frac{L}{AvegDist} \tag{5.26}$$

where K is the proportionality constant. It is noteworthy that the fitness value is used only for comparison purpose. Therefore, the value of K does not hamper our objective. Without loss of generality, we assume that $K=1$. Therefore,

$$Fitness = \frac{L}{AvegDist} \tag{5.27}$$

Our objective is to maximize the *Fitness* value. In other words,

$$\text{Objective}: \text{ Maximize } Fitness \tag{5.28}$$

Therefore, higher the fitness value, the better is the particle position.

5.2.2.3. Velocity and position update. In the same way as routing algorithm, the velocity and the position are updated in each iteration using Eqs. (3.2) and (3.3) respectively. Then same way $Pbest_i$ and *Gbest* are also updated. The clustering algorithm can be developed in the same way as routing algorithm (refer Fig. 4).

6. Experimental results

We performed extensive experiments on the proposed algorithm using MATLAB R2012b and C programming language. The experiments were performed with diverse number of sensor nodes ranging from 200 to 700 and 60 to 90 gateways. Each sensor node was assumed to have initial energy of 2 J and each gateway has 10 J. In the simulation run, we used following parameter values same as in Heinzelman et al. (2002) as shown in Table 5.

Table 5
Simulation parameters.

Parameter	Value
Area	500×500 m^2
Sensor nodes	200–700
Gateways	60–90
Initial energy of sensor nodes	2.0 J
Number of simulation iterations	200
Communication range	150 m
E_{elec}	50 nJ/bit
ε_{fs}	10 pJ/bit/m^2
ε_{mp}	0.0013 pJ/bit/m^4
d_0	87.0 m
E_{DA}	5 nJ/bit
Packet size	4000 bits
Message size	200 bits

Table 6
PSO parameters.

Parameter	Value
N_P	60
C_1	1.4962
C_2	1.4962
w	0.7968
V_{max}	0.5
V_{min}	−0.5

We have tested our proposed algorithms extensively and depict the experimental results for both the routing and clustering in a combined way. For the sake of simulation we considered two different network scenarios (WSN#1 and WSN#2). Both of them have the sensing field of 500×500 m^2 area. For the WSN#1, the position of the base station was taken at (500,250), i.e., in a side of the region and for the WSN#2, the position of the base station was taken at (250,250), i.e., in the center of the region. To execute our proposed algorithms, we considered an initial population of 60 particles and the values of PSO parameters are taken same as in Daniel and Kennedy (2007), Sahoo et al. (2011). This is shown in Table 6. The size of the swarm can be defined differently. However, we use a predefined swarm size of 60 particles. The same experiment can be tested for 50 or 70 particles. In weight sum approach we had tested for different values of the weight factor, W_1 and W_2 and observed that for $W_1=0.2$ and $W_2=1-W_1=0.8$, it was showing comparably better result. Therefore, in the simulation we have taken the same value. It should be noted that it is very difficult to precisely and accurately select these weight values, even for someone familiar with the problem domain (Konak et al., 2006).

For the sake of comparison, we also executed the GA-based clustering algorithm presented by Kuila et al. (2013) and another two clustering algorithms GLBCA (Low et al., 2008) and LDC (Ataul et al., 2008). Note that all the above three algorithms assume that the base station is within the direct communication range of the gateways and thereby they do not consider any multi-hop routing algorithm between gateways and the base station. As the proposed work consists of clustering algorithm along with multi-hop routing, for the fair comparison a popular GA-based multi-hop routing algorithm (Ataul et al., 2009) is executed for the three clustering algorithms (Ataul et al., 2008; Low et al., 2008; Kuila et al., 2013).

First, we ran the algorithms for comparing lifetime of the network by varying the sensor nodes from 200 to 700 and the number of gateways for 60 and 90 on both of the network

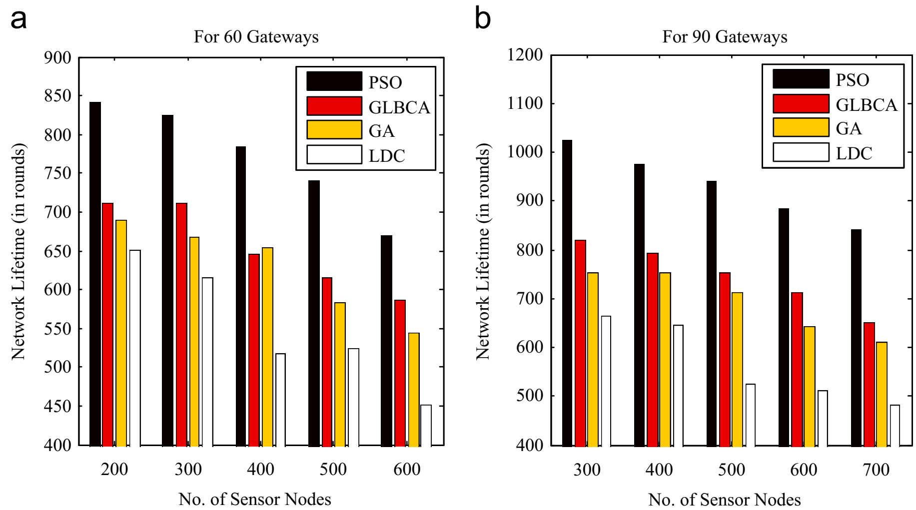

Fig. 6. Comparison in terms of network life in rounds for (a) 60 gateways and (b) 90 gateways in WSN#1.

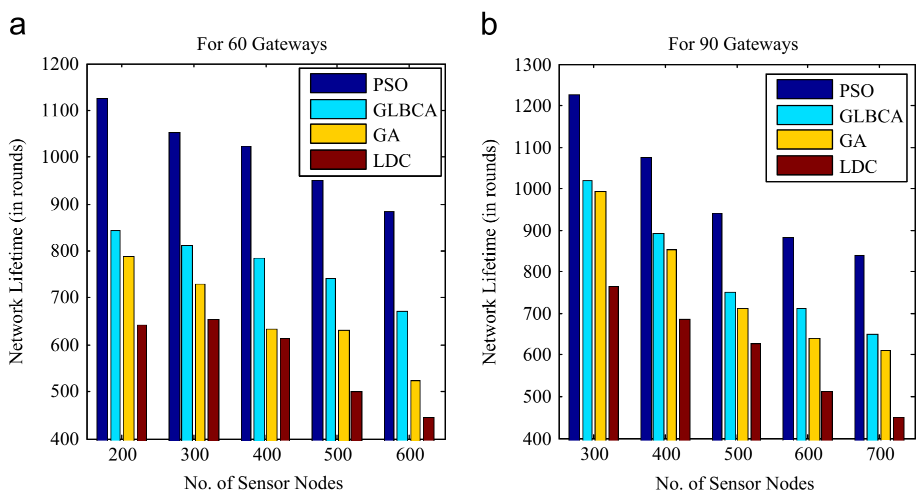

Fig. 7. Comparison in terms of network life in rounds for (a) 60 gateways and (b) 90 gateways in WSN#2.

scenarios, WSN#1 and WSN#2. Figs. 6 and 7 show the comparison of the proposed PSO-based algorithm with GLBCA, GA and LDC in terms of network life in WSN#1 and WSN#2 respectively. It can be observed from Figs. 6 and 7 that the proposed algorithm has better network lifetime than the GLBCA, LDC and GA-based clustering algorithms. This is due to the clustering phase the proposed PSO based algorithm where it takes care of the CHs those are inevitably used as a relay node to forward the data packets to the base station. Thus it helps to delay in initial death of the CH and increase network lifetime. Whereas the existing clustering algorithms do not deal with the uneven data forwarding effects on the CHs, thereby the CHs those are near to the base station die quickly due to extra work load of frequent data forwarding. The average (mean) network life time for 25 runs of the algorithms along with their standard deviations (SD) for both the scenarios WSN#1 and WSN#2 are also calculated by varying number of sensor nodes and gateways. The results are shown in Tables 7 and 8 for 60 and 90 gateways respectively. It is clear that the average network life time for the proposed PSO is maximum. However, GA has the minimum fluctuations in the average network life.

Next we ran the algorithms to compare the balancing of lifetime of the gateways by varying the sensor nodes from 200 to 500 for 60 gateways on both the network scenarios, WSN#1 and WSN#2. Here, we calculate the duration between first gateway die (FGD) and last gateway die (LGD) in rounds. This is to be noted that lower the duration, better is the balancing of the lifetime. Fig. 8 shows the comparison of the proposed algorithm, GLBCA, LDC and GA-based clustering algorithms in terms of balancing of

Table 7
Mean network life time and standard deviation in WSN#1 (60 gateways).

Algorithms	200 Sensor nodes		400 Sensor nodes		600 Sensor nodes	
	Mean	SD	Mean	SD	Mean	SD
PSO	831.23	22.21	814.12	21.21	691.25	19.36
GLBCA	715.21	23.21	661.14	23.53	601.41	20.54
GA	674.11	21.14	651.71	23.64	541.87	18.63
LDC	643.52	26.84	523.39	24.32	441.69	19.23

Table 8
Mean network life time and standard deviation in WSN#2 (90 gateways).

Algorithms	300 Sensor nodes		500 Sensor nodes		700 Sensor nodes	
	Mean	SD	Mean	SD	Mean	SD
PSO	1185.63	27.56	954.85	23.51	874.53	26.35
GLBCA	1036.32	29.35	774.52	24.12	675.12	24.13
GA	986.74	24.56	714.23	21.54	621.98	29.41
LDC	754.31	29.54	621.5	28.14	422.12	28.53

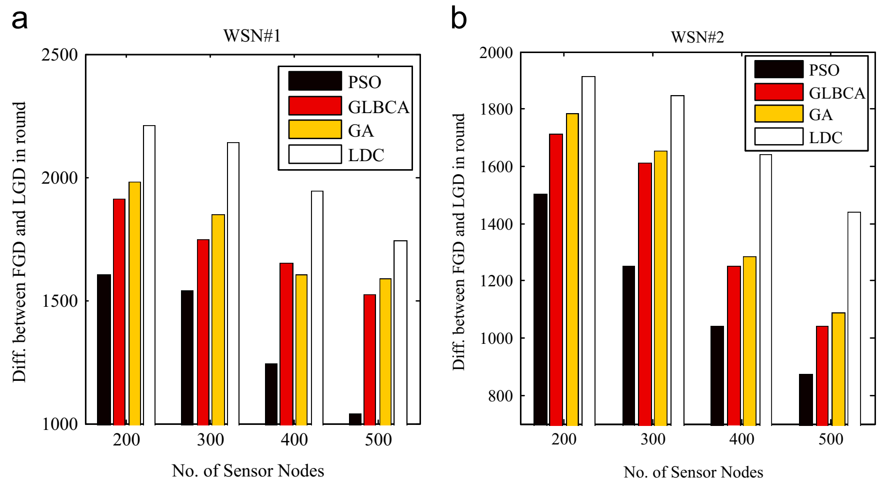

Fig. 8. Difference between first gateway die (FGD) and last gateway die (LGD) in rounds for (a) WSN#1 and (b) WSN#2.

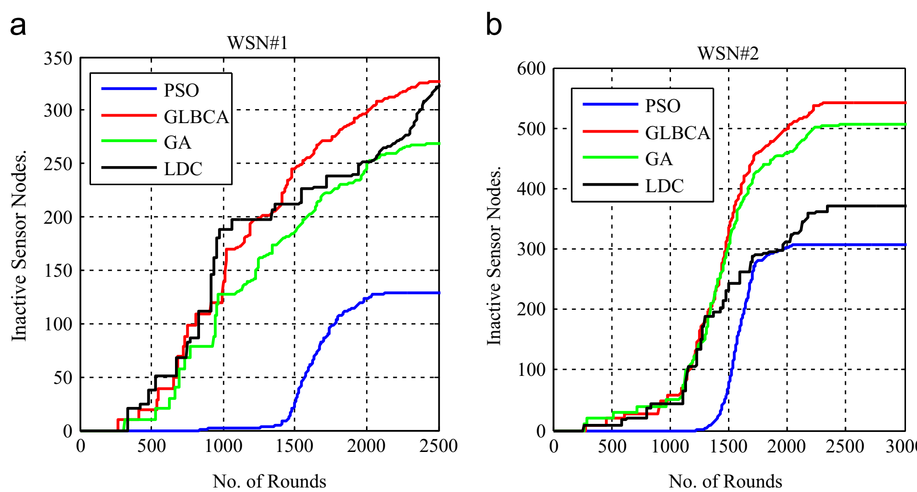

Fig. 9. Comparison in terms of inactive sensor nodes in (a) WSN#1 and (b) WSN#2.

lifetime in WSN#1 and WSN#2 respectively. It can be observed that the proposed algorithm has better balancing than the existing clustering algorithms.

Now, we show the comparison of the algorithms in terms of number of inactive sensor nodes per round in WSN#1 and WSN#2 respectively. A sensor node is considered as active if its existing energy is not zero and also there must be at least one CH within its communication range. Sometimes few CHs die quickly for improper load balancing in clustering and extra overload of data forwarding. As a result, few sensor nodes are unable to find any CH within their range, though they still may have some existing energy. In our scenario this type of sensor nodes are also considered as inactive. Simulations are performed by means of different algorithms for 600 sensor nodes and 60 gateways. It can be observed from Fig. 9 that the rate of inactive of the sensor nodes in both of the scenarios for the proposed algorithm is lesser than the existing algorithms. This is due to the fact that our derived fitness function takes care about the energy consumption of the normal sensor nodes by reducing the distances between sensor nodes and the gateways. Moreover, long life of the CHs helps the sensor nodes to be active for the long time. It can be observed that GLBCA and the GA-based clustering algorithms only balance the load of the CHs. To achieve this goal, some sensor nodes are assigned to the CH which may be farther from it. As a result their energies are drained out due to long haul transmission and die quickly. Whereas, though LDC assigns the sensor nodes to their nearest CH to reduce energy consumption of the normal sensor nodes, it does not take care of the load balancing of the CHs and the data forwarding overhead. In a result, frequent death of the CHs lead some sensor nodes become inactive though they may have some remaining energy.

Fig. 10 shows the comparison of energy (J) consumption of the network per round for 600 sensor nodes and 60 gateways in WSN#1 and WSN#2 respectively. Though the proposed algorithm, GLBCA and the GA-based clustering algorithms consume more or less same amount of energy it can be claimed that the proposed algorithm performs better in this respect. The justification behind it is that the higher number of active sensor nodes in the network consume more energy than the others. LDC consumes comparably lesser energy due to its sensor assignment strategy. As the sensor nodes are assigned to their nearest CH, they consume less energy and as a result the overall energy consumption of the network becomes lesser than the others. However, overall performance of the network is not only the measurement of energy consumption. It should be noted that in terms to assign the sensor nodes to their nearest CH, load of the CHs is not properly balanced which leads to initial death of the overloaded CHs. The initial death of the CHs may cause the network be disconnected and the maximum sensed data packets are unable to reach to the base station. It can be observed from Fig. 11 that number of data packets received by the

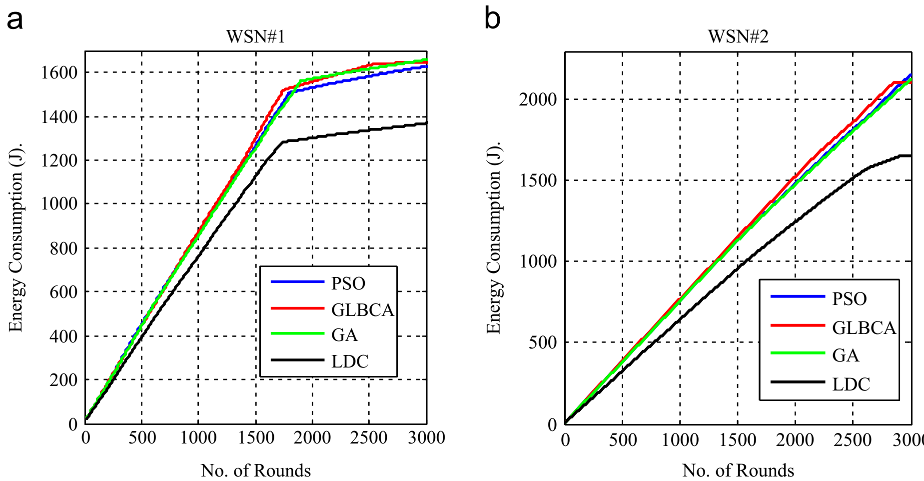

Fig. 10. Comparison in terms of energy consumption in (a) WSN#1 and (b) WSN#2.

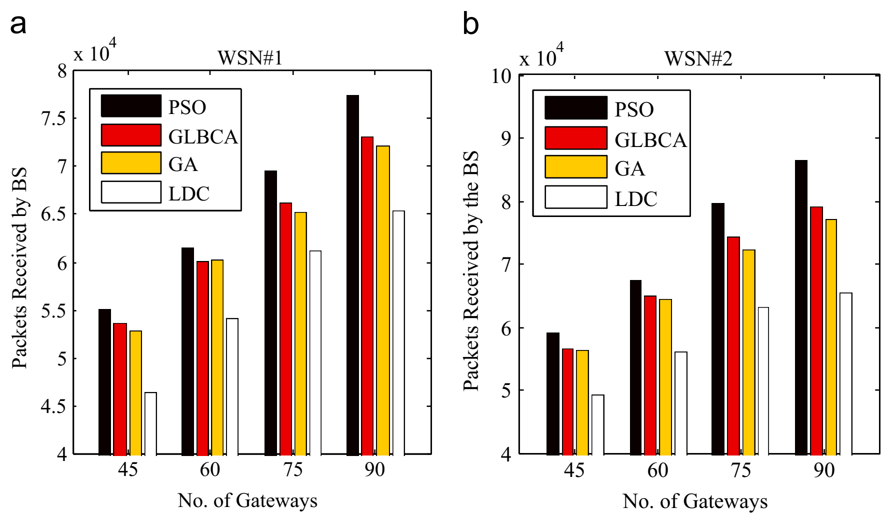

Fig. 11. Comparison in terms of total data packets received by the base station in (a) WSN#1 and (b) WSN#2.

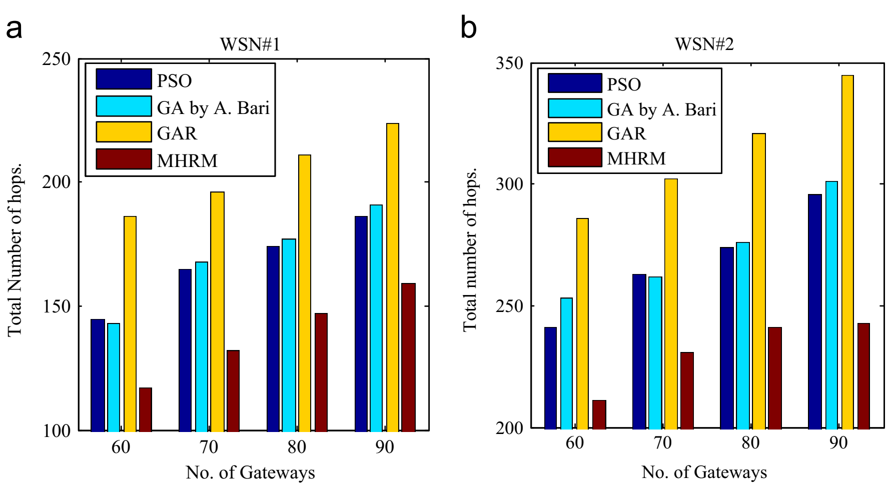

Fig. 12. Comparison in terms of total number of hops for (a) WSN#1 and (b) WSN#2.

base station in LDC is comparably very lesser than the proposed algorithm as well as others. In this case the proposed algorithm is far better than the other existing algorithms.

To evaluate the performance of the proposed routing algorithm, we ran two GA-based routing algorithms, i.e., GAR (Gupta et al., 2013) and the algorithm proposed by Ataul et al. (2009). We also ran MHRM (Chiang et al., 2007). Fig. 12 shows the comparison of the algorithms in terms of the total number of hops used. In this case, MHRM always shows the better result. This is because MHRM uses maximum possible distance for the next hop selection, thus it can easily reduce the hop count but unfortunately it extends the transmission distance and use long haul transmission. Whereas,

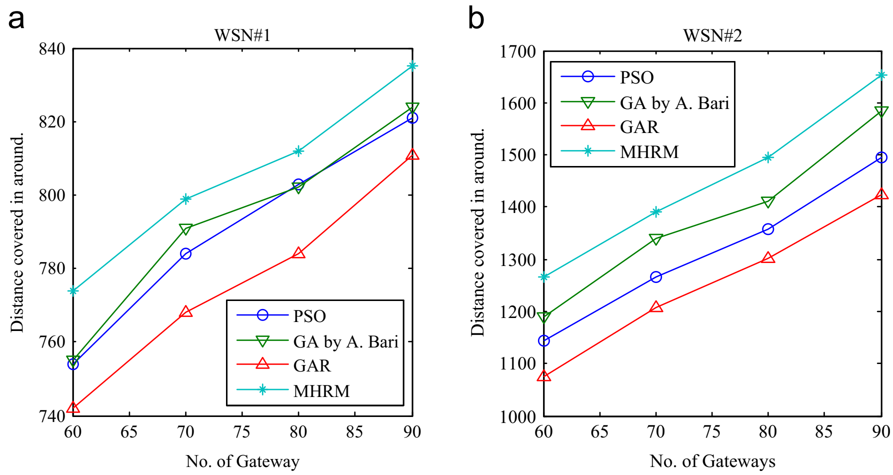

Fig. 13. Comparison in terms of total distance covered in a round for (a) WSN#1 and (b) WSN#2.

GAR minimizes the transmission distance to reduce the energy consumption but it uses comparably high number of hops. The proposed PSO-based routing algorithm builds a trade-off between transmission distance and hop count, i.e., GAR and MHRM. We also compare all the algorithms with respect to the total distance covered in each round. This can be observed from Fig. 13 that the proposed algorithms cover comparably less distance than MHRM and GA-based routing algorithm proposed by Ataul et al. (2009) in a single round.

7. Conclusions

In this paper, first a Linear and a Non-linear Programming have been formulated for two important optimization problems for wireless sensor networks, i.e., energy efficient routing and clustering respectively. Then, two algorithms have been presented for the same based on particle swarm optimization. The routing algorithm has been developed by considering a trade-off between transmission distance and the number of hop-count. In the clustering phase, routing overhead of the CHs is considered for balancing the energy consumption of the CHs. All the CHs which are heavily used as next hop relay nodes in data forwarding are assigned lesser number of sensor nodes. Thus the energy consumption of the CHs is significantly balanced and the lifetime of the network is improved. The algorithms are based on the derivation of efficient particle encoding scheme and fitness function for routing and clustering separately. The algorithms have been extensively tested with several scenarios of WSNs by varying number of sensor nodes and gateways. The experimental results have shown that the proposed algorithms perform better than the existing algorithms in terms of network life, number of inactive sensor nodes and the total data packets transmission.

References

Abbasi, A.A., Mohamad, Y., 2007. A survey on clustering algorithms for wireless sensor networks. Comput. Commun. 30, 2826–2841.

Abdul, L.N.M., et al., 2007. Energy aware clustering for wireless sensor networks using particle swarm optimization. In: IEEE PIMRC, pp. 1–5.

Ahmad, A., et al., 2012. MAC layer overview for wireless sensor networks. In: CNCS 2012, pp. 16–19.

Akyildiz, I.F., et al., 2002. Wireless sensor networks: a survey. Comput. Netw. 38, 393–422.

Al-Refai, H., et al., 2011. Efficient routing LEACH (ER-LEACH) enhanced on LEACH protocol in wireless sensor networks. Int. J. Acad. Res. (Part I) 3, 42–48.

Ataul, B., et al., 2008. Clustering strategies for improving the lifetime of two-tiered sensor networks. Comput. Commun. 31, 3451–3459.

Ataul, B., et al., 2009. A genetic algorithm based approach for energy efficient routing in two-tiered sensor networks. Ad Hoc Netw. 7, 665–676.

Aykut, Y.M., et al., 2011. QoS-aware MAC protocols for wireless sensor networks: a survey. Comput. Netw. 55, 1982–2004.

Baronti, P., et al., 2007. Wireless sensor networks: a survey on the state of the art and the 802.15.4 and ZigBee standards. Comput. Commun. 30, 1655–1695.

Calhoun, B.H., et al., 2005. Design considerations for ultra-low energy wireless microsensor nodes. IEEE Trans. Comput. 54, 727–740.

Cardai, M., Du, D.Z., 2005. Improving wireless sensor network lifetime through power aware organization. Wirel. Netw. 11, 333–340.

Chakraborty, U.K., et al., 2012. Energy-efficient routing in hierarchical wireless sensor networks using differential-evolution-based memetic algorithm. In: IEEE WCCI 2012, pp. 1–8.

Chiang, S.S., et al., 2007. A minimum hop routing protocol for home security systems using wireless sensor networks. IEEE Trans. Consum. Electron. 53, 1483–1489.

Daniel, B., Kennedy, J., 2007. Defining a standard for particle swarm optimization. In: Proceedings of the 2007 IEEE Swarm Intelligence Symposium, pp. 120–127.

Dietrich, I., Dressler, F., 2006. On the Lifetime of Wireless Sensor Networks, University of Erlangen, Department of Computer Science, 7, Technical Report, 04/06, December 2006.

Enan, A., et al., 2011. Energy-aware evolutionary routing protocol for dynamic clustering of wireless sensor networks. Swarm Evol. Comput. 1, 195–203.

Goldberg, D.E., 2007. Genetic Algorithms: Search Optimization and Machine Learning. Addison Wesley, Massachusetts.

Gupta, G., Younis, M., 2003. Load-balanced clustering of wireless sensor networks. In: IEEE ICC 2003, pp. 1848–1852.

Gupta, S.K., Kuila, P., Jana, P.K., 2013. GAR: an energy efficient GA-based routing for wireless sensor networks. In: ICDCIT 2013, LNCS 7753, pp. 267–277.

Habib, M.A., Sajal, K.D., 2008. A trade-off between energy and delay in data dissemination for wireless sensor networks using transmission range slicing. Comput. Commun. 31, 1687–1704.

Heinzelman, W.B., et al., 2002. Application specific protocol architecture for wireless microsensor networks. IEEE Trans. Wirel. Commun. 1, 660–670.

Issam, M., et al., 2013. Particle swarm optimization for solving engineering problems: a new constraint-handling mechanism. Eng. Appl. Artif. Intell. 26, 1263–1273.

Kemal, A., Mohamed, Y., 2005. A survey on routing protocols for wireless sensor networks. Ad Hoc Netw. 3, 325–349.

Kennedy, J., Eberhart, R., 1995. Particle swarm optimization. In: Proceedings of the IEEE International Conference on Neural Networks, vol. 4, pp. 1942–1948.

Konak, A., et al., 2006. Multi-objective optimization using genetic algorithms: a tutorial. Reliab. Eng. Syst. Saf. 91, 992–1007.

Kuila, P., Jana, P.K., 2014. Approximation schemes for load balanced clustering in wireless sensor networks. J. Supercomput. 68, 87–105, http://dx.doi.org/10.1007/s11227-013-1024-6.

Kuila, P., Gupta, S.K., Jana, P.K., 2013. A novel evolutionary approach for load balanced clustering problem for wireless sensor networks. Swarm Evol. Comput. 12, 48–56.

Kuila, P., Jana, P.K., 2012a. Improved load balanced clustering algorithm for wireless sensor networks. In: ADCONS 2011, LNCS 7135, pp. 399–404.

Kuila, P., Jana, P.K., 2012b. Energy efficient load-balanced clustering algorithm for wireless sensor network. In: ICCCS 2012, Procedia Technology, vol. 6, pp. 771–777.

Kuila, P., Jana, P.K., 2012c. An energy balanced distributed clustering and routing algorithm for wireless sensor networks. In: PDGC 2012, IEEE Xplore, pp. 220–225.

Kulkarni, R.V., et al., 2011. Particle swarm optimization in wireless-sensor networks: a brief survey. IEEE Trans. Syst., Man, Cybern.–Part C: Appl. Rev. 41, 262–267.
Low, C.P., et al., 2008. Efficient load-balanced clustering algorithms for wireless sensor networks. Comput. Commun. 31, 750–759.
Madan, R., et al., 2005. Cross-layer design for lifetime maximization in interference-limited wireless sensor networks. In: Proceedings of the IEEE INFOCOM, vol. 3, pp. 1964–1975.
Pan, J., et al., 2003. Topology control for wireless sensor networks. In: Proceedings of the 9th Annual International Conference on Mobile Computing and Networking, pp. 286–299.
Sahoo, N.C., Ganguly, S., Das, D., 2011. Simple heuristics-based selection of guides for multi-objective PSO with an application to electrical distribution system planning. Eng. Appl. Artif. Intell. 24, 567–585.
Saleem, M., et al., 2011. Swarm intelligence based routing protocol for wireless sensor networks: survey and future directions. Inf. Sci. 181, 4597–4624.
Singh, B., Lobiyal, D.K., 2012. Energy-aware cluster head selection using particle swarm optimization and analysis of packet retransmission in WSN. Procedia Technol. 4, 171–176.
Thitipong, N., Nitin, V.A., 2011. Optimization of tile manufacturing process using particle swarm optimization. Swarm Evol. Comput. 1, 97–109.
Tyagi, S., Kumar, N., 2013. A systematic review on clustering and routing techniques based upon LEACH protocol for wireless sensor networks. J. Netw. Comput. Appl. 36, 623–645.
Wei, H.L., Nor, A.M.I., 2014. Particle swarm optimization with increasing topology connectivity. Eng. Appl. Artif. Intell. 27, 80–102.
Wi, S.Y., et al., 2012. Genetic algorithm-based steel erection planning model for a construction automation system. Autom. Constr. 24, 30–39.
Zungeru, A.M., et al., 2012. Classical and swarm intelligence based routing protocols for wireless sensor networks: a survey and comparison. J. Netw. Comput. Appl. 35, 1508–1536.