Energy Management using Optimal Fuzzy Logic Control in Wireless Sensor Network

Chenglong Cao [✉]
Anhui Finance and Trade Vocational College, Hefei, China
chenglongcao@sina.cn

Xiaoling Zhu
Hefei University of Technology, Hefei, China

Abstract—Energy is a key factor that affects the lifetime of wireless sensor network (WSN). This paper proposes an adaptive energy management model to improve the energy efficiency in WSN. Unlike existing clustering routing protocols, the overall performance indicators of the network are introduced into fuzzy logic control (FLC). And the output of FLC, i.e., the adjustment value of cluster head scale, is fed back and used to generate a new cluster. Considering the design of membership functions (MFs) of FLC has a significant impact on system performance, particle swarm optimization (PSO) is used. The optimization goal of MFs is to reduce the number of dead nodes and increase the remaining energy level. Simulation experiments were conducted for the low energy adaptive clustering hierarchy protocol (LEACH), the conventional FLC, the FLC using genetic algorithm (GA), and the FLC using PSO. The results show that the proposed FLC-PSO has the best performance among the four protocols. Therefore, it can be used efficiently in energy management of WSN.

Keywords—wireless sensor network (WSN), energy control; fuzzy logic control (FLC), particle swarm optimization (PSO)

1 Introduction

Wireless sensor network (WSN) has been widely used in military facilities, environmental monitoring, intelligent transportation, smart home and medical care due to the advantages of cheapness, easy implementation and reliability [1][2]. In WSN, sensor nodes perceive and collect data from surrounding environment. Meanwhile, they have limited battery power. If the nodes are deployed in a complex environment where battery power is difficult to supply in time, they will die after a period of use [3]. Therefore, energy management is a key part of WSN research [4][5].

In recent years, some methods have been proposed to reduce energy consumption, and they are mainly involved in the following aspects.

(1) Data aggregation method. Tang et al. [6] proposed a routing algorithm for WSN to avoid transmitting relevant data repeatedly. In [7], the sink node uses a

greedy algorithm to reconstruct linear compression projection signals according to finite difference and wavelets sparse basis, respectively. Then, the least square method is used to aggregate different reconstruction signals.

(2) Mobile strategy. Since energy consumption of the nodes near the base station is large, it is easy to cause a hot spot issue. If a sink or a base station is mobile, it may shorten routing path. In [8], the location of a sink is selected randomly. The moving trajectory of the sink is predestined to avoid location broadcast, and it lacks flexibility and extensibility. An optimal deadline-based strategy [9] was proposed that the direction of the sink is determined by network parameters, and the sink makes a purposeful movement to extend the network lifetime.

(3) Clustering hierarchy method. Clustering routing is beneficial to save electricity of WSN [10]. A node is selected as a cluster head (CH), and near nodes are cluster members. CH collects and forwards messages from cluster members. In general, it runs out of energy and dies earlier than other nodes. Efficient selection strategy for CH will prolong the lifetime of WSN and improve the network performance. In the low energy adaptive clustering hierarchy protocol (LEACH) [11], a node is selected as a CH randomly, and then energy costs are allocated to each node. One drawback of LEACH is that the probability of CH selection is fixed but under a continuously changing network environment. Some studies focus on selection mechanism. Based on the LEACH protocol, Zhou et al. [12] used the remaining energy and the geographical location to design the selection protocol for CH. In [13], the remaining energy is predicted according to the number of neighbor nodes and the distance from the node to the base station.

The above studies [11][12][13] considered the local performance indicators such as the remaining energy of one node. The overall performance indicators of WSN, such as the overall remaining energy and the number of dead nodes, are also the key factors in balancing energy consumption. Our energy management model incorporates them to adjust the scale of CHs.

(4) Fuzzy logic control (FLC). In recent years, several FLC approaches have been introduced to improve energy management in WSNs. Based on predicted remaining energy, a competition radius of a cluster is calculated by a fuzzy algorithm [13]. A proposed fuzzy logic based mechanism in [14] determines the sleeping time of home devices in an automation environment according to the battery level and the ratio of throughput to workload. In the distributed approach [15], a fuzzy logic engine is developed on each wireless network node to decrease the number of message transmissions. Considering most energy is consumed in data transmission to the sink, some researchers have suggested mobile sink methods to minimize power consumption. Masdari et al. [16] introduce an approach in which a fuzzy-based mechanism controls the movement of the sink; the inputs of fuzzy algorithm include the number of neighbor nodes, the remaining energy, and the distance between a sensor node and a sink.

Unlike [13-16], we use FLC to control the selection of CHs in WSN according to the overall remaining energy level and the number of dead nodes.

(5) Optimization for FLC. In a fuzzy system, there are different parameters and rules, which require to be optimized for good results. Various artificial intelligence methods and random search techniques, such as GA, PSO [17] and artificial neural

network (ANN), have been proposed to improve the performance of a conventional FLC system. For instance, an ANN is applied in FLC [18]. Since an ANN-based system requires a large volume of training data to obtain the consistent results, the shortcoming restricts its application in WSN. Among the intelligence methods, PSO is manageable and its encoding is simple. PSO discovers the optimal points through the interaction of individuals in a population of particles. In general, it can find a high-quality solution in a smaller computation time [19] and it achieves a well-built convergence characteristic than other stochastic methods.

In a FLC system, the design of membership functions (MFs) has a significant impact on the system performance. In a non-optimized FLC system, each MF has an interval with the same or uncertain length, which does not ensure that the number of the dead nodes is small, the remaining energy is large, and the WSN lifetime is long. Therefore, the intervals of the MFs need to be optimized.

This paper proposes an efficient energy management model under fuzzy logic control. The main contributions are: 1) unlike existing clustering routing protocols, we integrate the overall performance indicators (the overall remaining energy and the number of dead nodes) into WSN fuzzy control. 2) FLC is used to control the scale of CHs. The out of FLC, as an adjustment value, is fed back to the network to control the generation of new clusters. Based on FLC, the selection protocol for CHs is designed. 3) Since the PSO method can produce a high-quality solution in a smaller computation time, PSO is used to optimize the MFs of FLC, and the optimization objective is to reduce the number of death nodes and increase the remaining energy. 4) Simulation results show that in terms of the number of dead nodes, the remaining energy level, the number of cluster heads and the number of surviving nodes, the proposed FLC method using PSO has the best performance among the four protocols, i.e., the LEACH, the conventional FLC, the FLC using GA and the FLC using PSO. This work provides a new energy management method for WSN from the perspective of optimizing the MFs of FLC.

The rest of this paper is organized as follows. Section 2 presents the review optimization algorithms. Section 3 describes the framework and details of FLC for energy management. Section 4 discusses how to optimize the MFs of FLC using PSO, and describes the FLC-PSO algorithm. Simulation and performance evaluation are presented in Section 5. Finally, some conclusions are drawn in Section 6.

2 Review of optimization algorithms

In recent years, various optimization algorithms have been proposed, and some have shown promising performances on the optimization of multidimensional problems.

2.1 Genetic Algorithm (GA)

GA is a well-known meta-heuristic algorithm based on the genetic evolution of the population. GA has been extensively used in the past and current research works for

optimization of single and multi-objective problems [20]. It starts the optimization process with a set of randomly generated solutions. In each generation, the fitness of every individual is evaluated. The good fitness individuals are selected to be the parents, and the next generation is produced by crossover and mutation of the parents. Commonly, the algorithm terminates when either a maximum number of generations has been produced, or a satisfactory fitness level has been reached. GA has good global search ability but poor local search ability, which means that it usually takes a long time to reach the real optimal solution.

2.2 Ant colony optimization algorithm (ACO)

ACO algorithm is a probabilistic technique for solving computational problems which can be reduced to finding good paths through graphs [21]. The first ACO algorithm was called the ant system and it aimed to solve the travelling salesman problem, in which the goal is to find the shortest round trip to link a series of cities. In the natural world, ants of some species (initially) wander randomly, and upon finding food return to their colony while laying down pheromone trails. If other ants find such a path, they are likely not to keep travelling at random, but instead to follow the trail. ACO algorithm simulates the foraging behavior of ants in nature. Ants perceive the pheromones released by other ants, prefer to choose the path according to high pheromone concentration and a short distance, and release a certain amount of pheromone to form positive feedback. Finally, the shortest path is found. So ACO algorithm is strong robustness, and suitable for distributed computing. However, the parameters in the algorithm are usually determined by experiments, and closely related to human experience. If they are not chosen properly, the speed in problem solving is slow and the quality of the solution is poor.

2.3 Particle swarm optimization (PSO)

PSO is a meta-heuristic algorithm. And it makes few or no assumptions about the problem being optimized and searches very large spaces of candidate solutions. It is a global optimization algorithm inspired by the flocking behavior of birds [22]. The algorithm starts by randomly generated particles within the search space. During each iteration, particles fly around the search space in order to find the best solution. The velocity of each particle updates based on the current velocity of the particle, the particle's individual best solution, and the global best solution. The location is updated based on the new velocity until the stopping criteria are reached. Formal descriptions are given as follows.

$$v_i(t+1) = v_i(t) + c_1 r_1(pbest_i(t) - x_i(t)) + c_2 r_2(gbest(t) - x_i(t))$$
$$x_i(t+1) = x_i(t) + v_i(t+1)$$

$$(1)$$

Here, $v_i(t)$ and $x_i(t)$ are the velocity and the position of the ith particle at the tth step, respectively; r_1 and r_2 are random numbers; $pbest_i(t)$ represents the best solution of the ith particle at the tth step, and $gbest(t)$ is the best global solution at the tth step. If

the number of the optimized parameters is n, $v_i(t)$ and $x_i(t)$ are both n-dimension vectors.

2.4 Grey wolf optimization (GWO)

GWO is a recent heuristic algorithm based on the natural hunting behavior of grey wolves [23]. The solution candidates are made up of four different types of wolves. The wolves (α, β and δ) can be considered as the first, second, and third best solutions. The rest of the candidate solutions are assumed to be ω. In addition, the main steps of hunting, searching for prey, encircling prey and attacking prey are implemented. GWA has the advantages of simple structure and easy operation, but in the process of optimization, it is easy to fall into a local optimal state due to the poor population diversity.

To sum up, the particles in a PSO algorithm may generally converge to the optimal solution faster than the individuals in a GA algorithm. In addition, ACO algorithm is to find the optimal path in a graph, and GWO is easy to fall into a local optimal state. PSO is a valuable approach for energy management in WSN, and thus we use it to optimize the parameters of FLC system.

3 Fuzzy energy control model

3.1 Fuzzy logic control framework

As shown in Fig.1, a FLC model includes CH selection, cluster operation, data extraction, FLC inference, and output feedback. In the CH selection module, a node randomly generates a random number r in the range of [0, 1] and compares it with the probability P. If $r \leq P$, then the node is as a CH. Next, other nodes choose the nearest CH, join the cluster and become cluster members. In a cluster, CH is responsible for the communication within the cluster and the communication between member nodes and the base station. Via base stations, the system obtains current network running indicators including the remaining energy of network and the number of dead nodes. The two indicators are as the two inputs of FLC. The FLC module includes three submodules: fuzzification, fuzzy inference, and defuzzification. Finally, the output value (i.e., the probability adjustment value) is fed back to the network for the next round of CH generation. Obviously, if the value is larger, the probability that a node is selected as CH is greater.
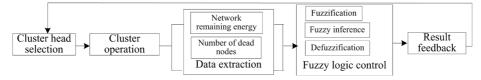

Fig. 1. FLC model

3.2 Fuzzy Control for Cluster Head Selection

Fuzzy variables. Define the variables D, E and Y as the number of dead nodes, the remaining energy of network, and the selection probability adjustment value for CH, respectively. Here, a dead node is the node whose energy is less than or equal to zero. And a node is selected as a CH only if the following two conditions are satisfied: (i) the node is not a dead node; (ii) the selection probability is less than or equal to the adjusted probability. So we assume that the input variables for a fuzzy control system are D and E, and the output variable is Y. In this paper, the ranges of D, E and Y are [0, 100], [0, 50], and [-0.04, 0.04], respectively. Then, the variable D is divided into three fuzzy subsets {High, Medium and Low} according to the number of dead nodes. Three fuzzy subsets correspond to three MFs, whose ranges are [-45, 45], [5, 95] and [55, 145], respectively. The variable E is quantified into five fuzzy subsets {Very High, High, Medium, Low, Very Low}, whose ranges are [-11.25, 11.25], [12.5, 23.75]), [13.75, 36.25]), [26.25, 48.75], and [38.75, 61.25], respectively. The variable Y is quantified into five fuzzy subsets {Reduced, Slightly Reduced, Unchanged, Slightly Increased, Increased}, whose ranges are [-0.058, -0.022]), [-0.038, -0.002]), [-0.018, 0.018], [0.002, 0.038], and [0.0222, 0.058]. In addition, the triangular MFs are chosen for all fuzzy variables, and then the membership degrees of conventional FLC algorithm are shown in Fig. 2
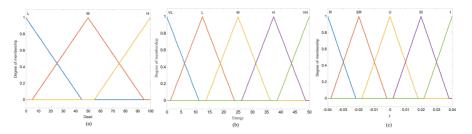

Fig. 2. Triangular MFs of input and output parameters: (a) number of dead nodes D, (b) remaining energy E, and (c) adjustment value Y

Fuzzy rules. Based on the fuzzy subsets of the two input variables, fifteen fuzzy inference rules are defined to adjust the probability of CH. As shown in Table 1, the output value is obtained by the inference rules using the IF-THEN statement. For instance, considering the last rule, if the number of dead nodes is High and the remaining energy is Very High, then the adjustment value will be increased.

Defuzzification. As it is known, FLC output is a fuzzy variable. Through defuzzification, the output can be handled by an actual system. In this paper, the centroid method is used. The final output is $y = \dfrac{\sum_i^m u_i y_i}{\sum_i^m u_i}$. Here, u_i is the truth value of the result for rule i. And it is the mass for the discrete case. y_i is the value for rule i where the result MF is maximum over the output variable fuzzy set and it is the location of the mass center for this result.

And the adjusted probability is

$$P = P_o + y \tag{2}$$

where P_o is the initial probability, y is the output of FLC, and P is the final selection probability that is fed back to the next round running of WSN.

Table 1. Inference Rules

Number of Dead Nodes	Remaining Energy	Adjustment Value
Low	Very Low	Reduced
Medium	Very Low	Unchanged
High	Very Low	Slightly Increased
Low	Low	Slightly Reduced
Medium	Low	Unchanged
High	Low	Slightly Increased
Low	Medium	Unchanged
Medium	Medium	Slightly Increased
High	Medium	Increased
Low	High	Unchanged
Medium	High	Increased
High	High	Increased
Low	Very High	Unchanged
Medium	Very High	Increased
High	Very High	Increased

4 Optimized FLC algorithm

In the FLC system, the controller inputs are the number of dead nodes D and the remaining energy E. There are three MFs (Low, Medium, High) for E and five MFs (Very Low, Low, Medium, High, Very High) for D. It means that eight MFs are required to optimize. First, an energy model is presented as follows.

4.1 Energy Model

According to the energy consumption proposed [11], if a node sends a message with the distance d and the length l, the energy consumption E_T is

$$E_T(l,d) = \begin{cases} l*(E_{TX} + E_{fs}*d^2), d \leq \sqrt{E_{fs}/E_{mp}} \\ l*(E_{TX} + E_{mp}*d^4), d \geq \sqrt{E_{fs}/E_{mp}} \end{cases} \tag{3}$$

Similarly, the energy consumption E_R of receiving a message is

$$E_R(l) = l*E_{RX} \tag{4}$$

Here, E_{TX} and E_{RX} are the energy consumptions of sending and receiving one bit, respectively. We assume $E_{TX}=E_{RX}$. If the distance d is less than the critical value E_{fs}/E_{mp}, a free space model is used where E_T is related to d^2. Conversely, a multi-channel attenuation model is used, and E_T is related to d^4. E_{fs} and E_{mp} are the coefficients for power amplification in the two models. Additionally, the receiving consumption E_R has nothing to do with d.

After selection, a new CH notifies the base station, causing the energy consumption $E_{CH1} = E_T(l,d_{toBS})$. During the transmission phase, a member sends some messages to CH, causing energy consumption of the member $E_M = E_T(l,d_{toCH})$. CH receives and converges data, so energy consumption $E_{CH2} = E_R(l) + EDA*l$, where *EDA* is the cost for converging one bit. Then, CH forwards data packets to the base station, causing $E_{CH3} = E_T(l,d_{toBS})$. In total, the energy consumption of CH is $E_{CH} = E_{CH1} + E_{CH2} + E_{CH3}$. So

$$E_{CH} = 2E_T(l,d_{toBS}) + E_R(l) + EDA*l \qquad (5)$$

Further, the energy consumption of all nodes in WSN

$$E_{total} = \sum_{all\ CHs} E_{CH} + \sum_{all\ Ms} E_M \qquad (6)$$

And thus, the remaining energy is

$$E = N*E_{node} - E_{total} \qquad (7)$$

Here, N is the number of nodes in WSN and E_{node} is the initial energy of a node.

4.2 Proposed FLC-PSO Algorithm

The main aim is to decrease D and increase E. Therefore, define the fitness function as

$$Fitness = D/E \qquad (8)$$

The remaining energy E can be obtained by (7), and D is the number of the nodes whose energy values are less than or equal to 0. In order to minimize *Fitness* value and achieve optimal FLC, the main process is shown in Fig. 3. First, input the upper and lower bounds of the parameters to be optimized. Then enter into the PSO optimization process. Run the simulation of WSN until the round number reaches the maximum value. During each simulation, the selection probability of CH is adjusted through fuzzy inference method. Based on D and E values obtained by the simulation, the fitness function is calculated, and then position and velocity continue to update until the stopping criteria are reached.
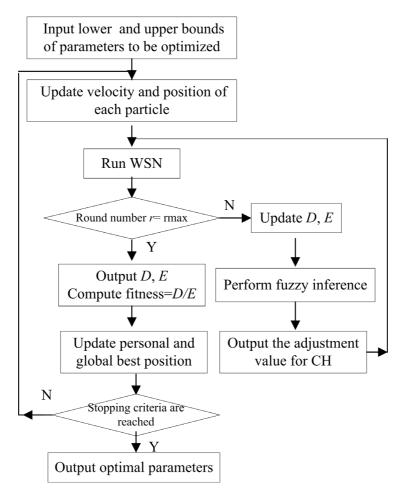
Fig. 3. Main process of optimal FLC

Furthermore, assume that the interval parameters of three MFs for E are aL_1, aL_2, aM_1, aM_2, aH_1, and aH_2. Obviously, $aL_1 < aL_2$, $aM_1 < aM_2$, and $aH_1 < aH_2$. Similarly, the interval parameters of five MFs for D are aVL_1, aVL_2, aL_1, aL_2, aM_1, aM_2, aH_1, aH_2, aVH_1, and aVH_2. The subscripts 1 and 2 represent the start point and the end point of one interval, respectively. So there are sixteen parameters to optimize. The following algorithm is to optimize n parameters for MFs.

In Algorithm 1, x_1, x_2, …, and x_n represent n parameters to be optimized. For example, x_1, x_2, x_3, x_4, x_5 and x_6 represent six parameters of three MFs of D, i.e., aL_1, aL_2, aM_1, aM_2, aH_1, and aH_2. Algorithm 2 is the simulation of running WSN with FLC. And its inputs are the upper and lower bounds of x_1, x_2, …, and x_n; its outputs are D and E, which are used to compute the fitness value.

Algorithm 1: Parameter optimization for MFs

Input : The lower bounds *lb* and the upper bounds *ub* of *n* parameters.
Output: Optimal parameters $x = [x_1, x_2, …, x_n]$.
1: initialize each particle with position, velocity, personal and global best position;
2: **repeat**
3:     **for** (each particle) **do**
4:         update velocity and position according to (1);
5:         $[D, E] \leftarrow$ runfuzzycontrolwsn(*lb*, *ub*);
6:         compute fitness value according to (8);
7:         update personal and global best position according to fitness value;
8:     **end for**
9: **until** (the stopping criteria are reached)
10:return *x*;

Algorithm 2: Runfuzzycontrolwsn

Input: The lower bounds *lb* and the upper bounds *ub* of *n* parameters.
Output: Dead nodes number *D* and remaining energy *E*.
1: initialize network parameters including energy, positions of sink and sensor nodes;
2: **while** (round number $r < rmax$)
3:     **for** (each node *i*) **do**
4:         if *i* is not a dead node and random number <= *P*, node(*i*).CH='Y';
5:     **end for**
6:     **for** (each node *i*) **do**
7:         if node(*i*).CH='N', then select the nearest CH to join the cluster;
8:     **end for**
9:     compute *E* according to (7) and update *D*;
10:    run FLC model, perform fuzzy inference, and output the adjustment value *y*
11:    compute the adjusted selection probability *P* according to (2);
12: **end while**
13: return (D, E)

Fig. 4 shows the fitness values achieved by the PSO algorithm where the swarm size is ten. In Fig. 4(a), six parameters for *D* are optimized using 20 iterations. In Fig. 4 (b), ten parameters for *E* are optimized using 10 iterations. Their implementation results achieve the convergence after the first seven and five iterations, respectively.

The optimal MFs are depicted in Fig. 5. Compared with Fig. 2, the optimized intervals are different from the conventional results. Particularly, it is observed that the fuzzy subset *Medium* has a larger range. And it's probably because there are more distributions in the subset *Medium* for *D* and *E*.
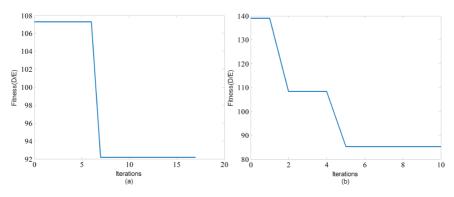.

Fig. 4. PSO results for (a) dead nodes, and (b) remaining energy
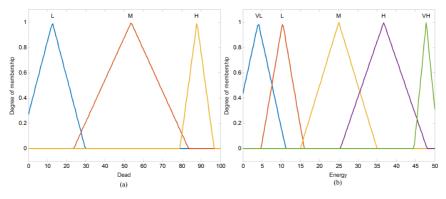

Fig. 5. Triangular MFs using PSO of (a) dead nodes , and (b) remaining energy

5 Simulation and Performance evaluation

5.1 Simulation parameters

In this paper, Matlab2016 is used to carry out simulation experiments on an Intel(R) Core(TM) i5-8250U CPU with the clock speed of 1.6 GHz. The experiments simulate the WSN with different control algorithms such as the LEACH, the conventional FLC and the optimal FLC. Based on the simulation results, the network performance indicators, such as remaining energy and surviving nodes, are discussed.

In the specific simulation, 100 sensor nodes are randomly distributed in a square area of 100m × 100m, and the base station is the central point (50, 50). More simulation parameters are shown in Table 2.

Table 2. Parameter Description

Parameters	Value
Region /m^2	100*100
Number of nodes	100
Initial selection probability	0.2
Node initial energy /J	0.5
E_{TX} /pJ	50
E_{RX} /pJ	50
Efs/nJ	10
Emp/nJ	0.0013
EDA /pJ	5
Packet size /bits	4000
Number of rounds	2000

5.2 Performance Analysis

Fig. 6 shows that the network state changes with the increase of rounds. As seen in Fig. 6 (a), there are few dead nodes before 1000 rounds. It is because the overall energy is abundant during this period. But, dead node increases rapidly after 1300*th* round. Fig. 6 (b) shows the change of network energy with the number of rounds. It can be observed that network energy approaches zero after 1500 rounds. In Fig. 6 (c), the number of CHs decreases obviously after 1300 rounds. In Fig. 6 (d), before 1000 rounds, there exists a linear change trend about data packets transmitted from CHs to the base station. Next, the change tends to be gentle.

5.3 Performance comparisons

As stated previously, the related experiments show that the ranges of the optimized MFs obtained by PSO in Fig. 4 are quite different from those of the MFs obtained by a conventional FLC in Fig. 2.

For further analyzing the optimization effects, we have carried out some WSN simulation experiments including the FLC, the proposed FLC-PSO and the classic LEACH. In addition, we introduce another intelligent algorithm, GA, to optimize the parameters of the FLC system. Some parameters in the GA algorithm are set as follows: crossover probability = 0.7, mutation probability =0.01 and population size=10. The upper and lower bounds of the parameters and the fitness function in the GA algorithm are consistent with the PSO algorithm. In the simulation, we observed that there is a big difference among LEACH, FLC, FLC-FSO and FLC-GA after 1300 rounds; meanwhile, their remaining energy is zero at 1600*th* round. Therefore, the simulation results of three algorithms from 1300*th* to 1600*th* rounds are presented in Fig. 7, 8, 9, and 10.
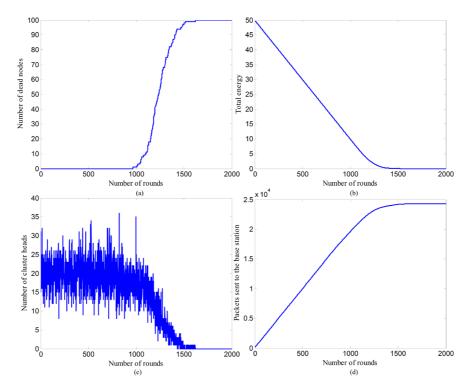
Fig. 6. Network performance using FLC-PSO: (a)number of dead nodes, (b) remaining energy, (c)number of CH, and (d) data packets from CHs to base stations

In Fig. 7, dead nodes of LEACH are more than those of other protocols. Among the four protocols, the number of dead nodes is in the descending order of LEACH, FLC, FLC-GA, and FLC-PSO. For example, at the 1450*th* round, there are 100 dead nodes for LEACH, 100 for FLC, 98 for FLC-GA, and 94 for FLC-PSO. In Fig. 8, the remaining energy of FLC, FLC-PSO and FLC-GA is more than that of FEACH in any round after the 1300 rounds. And between the 1360*th* and 1460*th* round, the performance of FLC-PSO is obviously better than FLC-GA. Specifically, the remaining energy of LEACH, FLC, FLC-GA and FLC-PSO at the 1450*th* round is 0, 0, 0.0294, and 0.0505, respectively. From the overall point of view, the blue curve in Fig. 9 is at the top, followed by the green one, the red one and the black one in the descending order. It means that the CHs of FLC-PSO are usually more than those of LEACH, FLC and FLC-GA. This may be because the network with FLC-PSO runs relatively well and thus provides a relative large number of CHs. In Fig. 10, the surviving nodes are decreasing with the increase of the number of rounds. In the case of LEACH, there is no surviving node when $r = 1414$; for PSO, there is no surviving nodes when $r = 1450$; for PSO-GA, there are 2 surviving nodes at the same $r = 1450$ and no surviving node when $r = 1524$; for PSO-FLC, there are 6 surviving nodes when $r = 1450$ and no surviving node when $r = 1662$.
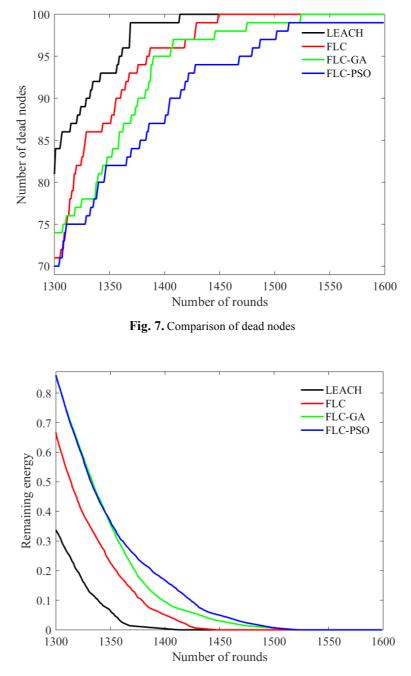
Fig. 7. Comparison of dead nodes

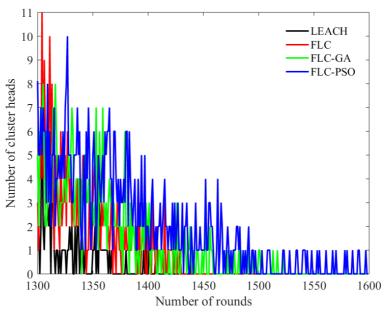

Fig. 8. Comparison of remaining energy
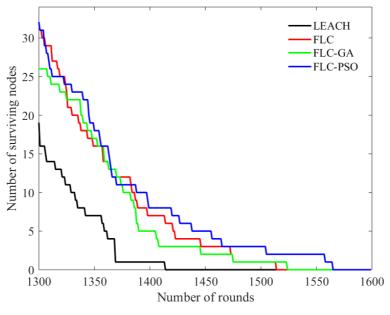
Fig. 9. Comparison of the number of CHs

Fig. 10. Comparison of surviving node

Therefore, energy management performance of FLC-PSO is the best among the four protocols. Unlike LEACH and FLC, FLC-PSO uses optimization method to minimize the ratio of the number of dead nodes to the remaining energy. Obviously, it is beneficial to save energy and further prolong the lifetime of network. In addition, compared with GA, PSO has a memory function, and each iteration and movement of PSO is moving in the local optimal and global optimal direction. In general, PSO is better than GA when solving continuous problems. This is just as we see in Fig. 7 to 10 that the performance of FLC-PSO is a little better than FLC-GA.

6 Conclusion

Energy management is a critical issue and a primary requirement in WSN. In this paper, we presented an energy management model based on the clustering hierarchy method. In the model, FLC is used to control the scale of cluster heads. Meanwhile, the output of FLC is obtained through the fifteen fuzzy logic rules. As a probability adjustment value, the output is fed back to the network to control the generation of new clusters. Unlike existing clustering routing protocols, our model takes the overall performance indicators of WSN, i.e., the overall remaining energy (D) and the number of dead nodes (E) as the input variables of FLC. The variable D has three MFs, and the variable E has five MFS. And sixteen parameters of eight MFs need to be optimized. Furthermore, PSO is used to optimize the MFs, and its optimization objective is to reduce the number of dead nodes and increase the remaining energy. In addition, a detailed description of the FLC-PSO algorithm is presented. And simulation results show that in terms of the number of dead nodes, the remaining energy level, the number of cluster heads and surviving nodes, the proposed FLC-PSO has the best performance among the four protocols (i.e., LEACH, conventional FLC, FLC-GA and FLC-PSO). Therefore, it provides an efficient energy management method for WSN from the perspective of optimizing MFs of FLC.

In this work, we have mainly considered two key factors: the remaining energy and the number of dead nodes. For future work, we plan to integrate more factors into the fitness function while taking into account efficiency.

7 Acknowledgment

8   References

[1] Ghosh, K., Neogy, S., Das, P. K., & Mehta, M. (2017). Intrusion detection at international borders and large military barracks with multi-sink wireless sensor networks: an energy efficient solution. *Wireless Personal Communications* No.2, 1-19.

[2] Kim, H. G., & Jin, Y. K. (2017). Environmental sound event detection in wireless acoustic sensor networks for home telemonitoring. *China Communications (English Edition)*, *14*(9), 1-10. https://doi.org/10.1109/CC.2017.8068759

[3] Chen, M., Gonzalez, S., Vasilakos, A., Cao, H. and Leung, C.M.(2011). Body area networks: a survey. *Mobile Networks and Applications*, *16*(2), 171–193. https://doi.org/10.1007/s11036-010-0260-8

[4] Anisi, M. H., Abdul, S.G. and Abdullah, A. H. (2015). A survey of wireless sensor network approaches and their energy consumption for monitoring farm fields in precision agriculture. *Precision Agriculture*, *16*(2), 216-238. https://doi.org/10.1007/s11119-014-9371-8

[5] Jindal, V., Verma, A. K. and Bawa, S. (2015). Quantitative Reduction in Communication Load for Energy Efficiency in WSN. *Wireless Personal Communications*, *85*(4), 2795-2810. https://doi.org/10.1007/s11277-015-2933-x

[6] Tang, J., Ma, S. F., Cao C. and Zhang, X. (2016). Research on clustering routing algorithm for Wireless Sensor Network based on data compression. *Journal of Yunnan University(Natural Sciences Edition)* , *38*(1), 37-43.

[7] Zhu, L., Liu, Y. , Ci, B. S. and Pan, Y. Z. (2016). The Method of Data Aggregation for Wireless Sensor Network Based on Cluster Compressed Sensing of Multi-Sparsity Basis. *Chinese Journal of Sensors and Actuators*, *29*(3), 417-422.

[8] Guerroumi, M., Badache, N., & Moussaoui, S. (2015). Mobile sink and power management for efficient data dissemination in wireless sensor networks. *Telecommunication Systems*, *58*(4), 279-292. https://doi.org/10.1007/s11235-014-9877-4

[9] Tashtarian, F., Moghaddam, M. H. Y., Sohraby, K., & Effati, S. (2015). Odt: optimal deadline-based trajectory for mobile sinks in wsn: a decision tree and dynamic programming approach. *Computer Networks*, Vol. 77, 128-143. https://doi.org/10.1016/j.comnet.2014.12.003

[10] Awwad, S. A., Ng, C.K., Noordin, N. K. and Rasid, M.F.A. (2011). Cluster Based Routing Protocol for Mobile Nodes in Wireless Sensor Network. *Wireless Personal Communications*, *61*(2), 251-281. https://doi.org/10.1007/s11277-010-0022-8

[11] Heinzelman, W. R., Chandrakasan, A. and Balakrishnan, H. (2000). Energy efficient communication protocol for wireless microsensor networks. in *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*, pp, 1-10. https://doi.org/10.1109/HICSS.2000.926982

[12] Zhou, J., Shi, Z. D., Zhang, Z. , Shan, L. and Fang, W. D. (2013). An Improved Algorithm Based on LEACH in WSN. *Journal of Shanghai University (Natural Science Edition)*, *19*(2), 116-119.

[13] Yang, X., Wang, R. and Peng, L. (2015). Fuzzy clustering hierarchy arithmetic based on residual energy prediction for WSN. *Computer Engineeriing and Applications*, *51*(14), 72-77.

[14] Abbas, Z., and Yoon, W. (2015). A survey on energy conserving mechanisms for the internet of things: Wireless networking aspects. *Sensors*, *15*(10), 24818-24847. https://doi.org/10.3390/s151024818

[15] Alaybeyoglu, A. (2015). A distributed fuzzy logic-based root selection algorithm for wireless sensor networks. *Computers & Electrical Engineering*, *41*, 216-225. https://doi.org/10.1016/j.compeleceng.2014.09.001

[16] Masdari, M., and Naghiloo, F. (2017). Fuzzy logic-based sink selection and load balancing in multi-sink wireless sensor networks. *Wireless Personal Communications, 97*(17), 1-27. https://doi.org/10.1007/s11277-017-4631-3

[17] Collotta, M., Pau, G., and Maniscalco, V. (2017). A fuzzy logic approach by using particle swarm optimization for effective energy management in iwsns. *IEEE Transactions on Industrial Electronics, 64*(12), 9496-9506. https://doi.org/10.1109/TIE.2017.2711548

[18] Yau, H. T., and Wu, C. H. (2011). Comparison of extremum-seeking control techniques for maximum power point tracking in photovoltaic systems. *Energies, 4*(12), 2180-2195. https://doi.org/10.3390/en4122180

[19] Kouba, N. E. Y., Menaa, M., Hasni, M., and Boudour, M. (2016). Design of intelligent load frequency control strategy using optimal fuzzy-pid controller, *4*(1), 41-64.

[20] Noshadi, A., Shi, J., Lee, W. S., Shi, P., and Kalam, A. (2014). Genetic algorithm-based system identification of active magnetic bearing system: A frequency-domain approach. In Control & Automation (ICCA), *11th IEEE International Conference on,* pp. 1281-1286. https://doi.org/10.1109/ICCA.2014.6871108

[21] Dorigo, M., & Gambardella, L. M. (1997). Ant colony system: a cooperative learning approach to the traveling salesman problem. *IEEE Transactions on evolutionary computation*, *1*(1), 53-66. https://doi.org/10.1109/4235.585892

[22] Kennedy, J. (1995). Particle swarm optimization. *Proc. of 1995 IEEE Int. Conf. Neural Networks*, (Perth, Australia), Nov. 27-Dec., *4*(8), 1942-1948.

[23] Saremi, S., Mirjalili, S. Z., and Mirjalili, S. M. (2015). Evolutionary population dynamics and grey wolf optimizer. *Neural Computing & Applications*, *26*(5), 1257-1263. https://doi.org/10.1007/s00521-014-1806-7

Authors

Chenglong Cao is an associate professor in Anhui Finance and Trade Vocational College, Hefei 230601, China. His research interests include business intelligence, information management and wireless sensor network. (chenglongcao@sina.cn)

Xiaoling Zhu is a lecturer in Hefei University of Technology, Hefei 230009, China. Her research interests include network security and intelligent computing. (zhuxl@hfut.edu.cn)