International Journal of Future Generation Communication and Networking
Vol. 8, No. 2 (2015), pp. 21-32
http://dx.doi.org/10.14257/ijfgcn.2015.8.2.03

A Harmony Search Based Low-Delay and Low-Energy Wireless Sensor Network

Zhen-Rui Peng, Hong Yin, Hai-Tang Dong, Hui Li and An Pan

School of Mechatronics Engineering, Lanzhou Jiaotong University, China
pengzr@mail.lzjtu.cn

Abstract

Aiming at different service quality requirements under different circumstances of wireless sensor network (WSN), harmony search (HS) algorithm is used to solve the low-delay and low-energy double-objective optimization problem in wireless sensor networks. First, analyze the low-delay and low-energy model, and establish the objective function. Then, use the priority-based path encoding algorithm to update harmony memory iteratively while searching the optimal path. Finally, simulation of 100-node network is carried out by using Matlab. The results show that sensor network data transmission path can be controlled according to the users' needs of delay and power.

Keywords: *wireless sensor network, low-delay, low-energy, harmony search, priority*

1. Introduction

Wireless Sensor Networks (WSNs) consist of large amounts of wireless sensor nodes, which have the features of high adaptability, large coverage area, and high fault tolerance. Meanwhile, WSNs are compact, light-weighted, and battery-powered devices wireless sensor networks. Because of these unique characteristics, they are employed in a wide range of applications in military and national security, environmental monitoring, and many other fields. Most of the time, wireless sensor nodes are located in harsh physical environments. Sensor nodes are severely constrained by the amount of battery power available, limiting the lifetime and quality of the network. Sensor nodes must save their scarce energy by all means and stay active in order to maintain the required sensing coverage of the environment [1].

Energy efficiency is one of the key design issues that need to be enhanced in order to improve the life span of the network. And more attention has been paid to above energy issue [2]. Many of the energy saving algorithms are clustering-based schemes such as Low-Energy Adaptive Clustering Hierarchy (LEACH) [3], Power-Efficient Gathering in Sensor Information Systems (PEGASIS) [4], and Collection Tree Protocol (CTP) [5].

With the development and wide use of wireless sensor networks, other factors influencing the performance of WSNs are also needed to take into consideration. Among these factors, the factor that minimizes delay to ensure the WSN real-time performance is one higher performance requirement. At present, the study about network delay mainly focuses on congestion control. Most of the energy saving algorithms focuses on clustering algorithms, such as above mentioned LEACH, PEGASIS, and CTP. These methods are realized through hierarchical data packets. These methods perform direct data transmission between nodes and base station. Both transmission distance and energy consumption is reduced. Queue competition among cluster members exits when data is transmitted to cluster head node. At the same time, multiple-hops data are sent to base station to increase the network delay is increased. So, the real-time performance of WSN is affected.

ISSN: 2233-7857 IJFGCN

Harmony search (HS) is a recently developed meta-heuristic algorithm that mimics the improvisation process of music players [6-8]. The HS algorithm, which demonstrates several advantages over traditional optimization techniques, has been successfully used in a wide variety of engineering optimization problems [9-11].

In this paper, a HS based method for selecting the WSN optimal path is proposed to realize the double objective optimization, which not only minimizes the transmission energy consumption but also decreases network transmission delay.

2. WSN Model Establishment

2.1. Path Transmission Delay

Path transmission delay is the time that data packet is transmitted from the source node to destination node. Data packet will transmit in multiple hop mode. Intermediate node is as the transfer station to reduce the single transmission distance, which inevitably leads to network delay. The total network communication delay is composed of four parts [12].

(1) Processing delay: Processing time that node receives and then forwards data.

(2) Queue delay: Queue time for multiple data passing through a certain same node.

(3) Transmission delay: Transmission time for the total data group transmitting from the first byte to the last byte.

(4) Propagation delay: Time consumption for data propagating in the transmission media.

Processing delay is determined by the node processor. Transmission delay and propagation delay is decided by network bandwidth and propagation speed respectively. These three network delays are constant for a certain network. So the same weights are assigned for them. These three delays are regarded as only having relation with the number of hops in transmission path. It is obvious that there are more intermediate nodes and there is the more delay. The network delay for single node is χ . If we want to reduce queue delay, we have to restrict data flow as little as possible to pass through a certain same node to decrease the completion in data transmission.

Network delay for waiting a data packet is ω and k data packets wait for reception, the network delay is $k * \omega$. Suppose P is the only transmission path from V_0 to V_n . $P = (V_0, V_1, \dots V_i, \dots, V_n)$, where $V_i (i < n)$ is the intermediate node in that path. d_i is the transmission delay between node i and node i -1, and then $d_i = \chi + k_i \omega$. Thus, network delay D_p corresponding path P can be expressed as

$$D_p = \sum_{i=1}^{n} d_i \tag{1}$$

Suppose there are m paths from node V_0 to V_n , the probability of each path to be selected can be expressed as

$$f_1 = \frac{1 / D_p}{\sum_{j=1}^{m} 1 / D_p} \tag{2}$$

2.2. Network Consumption Model

Since energy consumption is always a key issue in the investigation of wireless sensor networks, the aim of this paper is, therefore, to achieve the proposed network structure while keeping the energy consumption in the data collection process at low value. A wireless sensor node can be considered as a device, which is composed of three major units, the microcontroller unit (MCU), the transceiver unit (TCR), and the sensor board (SB). Each of these units will consume a certain amount of energy. The energy consumed by sensor node i can be expressed as

$$E_{i\_SN} = E_{i\_MCU} + E_{i\_TCR} + E_{i\_SB} \tag{3}$$

where E_{i_MCU} denotes the energy consumed by MCU; E_{i_TCR} the energy consumed by TCR, and E_{i_SB} the energy consumed by SB. E_{i_TCR} can be further expressed as

$$E_{i\_TCR} = E_{i\_TCR\_RX} + E_{i\_TCR\_TX}(d_i) \tag{4}$$

where $E_{i\_TCR\_RX}$ denotes the energy consumed by TCR in receive mode, and $E_{i\_TCR\_TX}(d_i)$ the energy consumed by TCR transmitting for a distance d_i.

Thus, the total energy consumed by *n*-nodes wireless sensor network can be expressed as

$$\begin{aligned} &E_{TOT}(N) \\ &= \sum_{i=1}^{N}(E_{i\_MCU} + E_{i\_TCR\_RX} + E_{i\_TCR\_TX}(d_i) + E_{i\_SB}) \\ &= C_1 + \sum_{i=1}^{N} E_{i\_TCR\_TX}(d_i) \end{aligned} \tag{5}$$

where $C_1 = \sum_{i=1}^{N}(E_{i\_MCU} + E_{i\_TCR\_RX} + E_{i\_SB})$ is constant.

In equation (5), $E_{i\_TCR\_TX}(d_i)$ can be expressed as

$$E_{i\_TCR\_TX}(d_i) = \begin{cases} E_{i\_TCR\_EC} + E_{i\_TCR\_FS} d_i^2, d_i < d_0 \\ E_{i\_TCR\_EC} + E_{i\_TCR\_AMP} d_i^4, d_i > d_0 \end{cases} \tag{6}$$

where $E_{i\_TCR\_EC}$ denotes the energy consumed by the electronic circuitry of TCR, $E_{i\_TCR\_FS}$ and $E_{i\_TCR\_AMP}$ the energy amplification coefficients corresponding to data transmission distance. Suppose threshold value $d_0 = \sqrt{\frac{E_{fs}}{E_{amp}}}$, when $d < d_0$, free space model is adapted to compute, and energy consumption is proportional to d_i^2; when $d > d_0$, multiple attenuation model is adopted to compute and energy consumption is proportional to d_i^4.

The total energy consumption can be expressed in a more compact form

$$E_{TOT}(N) = C_1 + C_2 + C_3 \sum_{i=1}^{N} \lambda d_i^2 \tag{7}$$

where $C_2 = E_{i\_TCR\_EC}$; γ is amplification coefficient; λ is 2 or4. In the modeling process, data transmission distance between two points is less than d_0, and the model can be simplified as

$$E_{TOT}(N) = C_1 + C_2 + C_3 \sum_{i=1}^{N} d_i^2 \tag{8}$$

From equation (8), reducing $\sum_{i=1}^{N} d_i^2$ can minimize the network energy consumption. Suppose there are m paths from node V_0 to V_n, the probability of each path to be selected can be expressed as

$$f_2 = \frac{1/E_p}{\sum_{j=1}^{m} 1/E_p} \tag{9}$$

3. Principles of Harmony Search

Harmony search (HS) is a music-inspired evolutionary algorithm, mimicking the improvisation process of music players. HS is simple in concept, few in parameters, and easy in implementation, with theoretical background of stochastic derivative [13-17]. HS mimics the improvisation process of musicians during which each musician plays a note for finding a best harmony all together. When applied to optimization problems, the musicians typically represent the decision variables of the cost function, and HS acts as a meta-heuristic algorithm which attempts to find a solution vector that optimizes this function. In such a search process, each decision variable (musician) generates a value (note) for finding a global optimum (best harmony).HS algorithm, therefore, has a novel stochastic derivative (for discrete variable) based on musician's experience, rather than gradient (for continuous variable) in differential calculus. HS uses five parameters, including three core parameters such as the size of harmony memory (HMS), the harmony memory considering rate (HMCR), and the maximum number of iterations $T\max$, and two optional ones such as the pitch adjustment rate (PAR) and the adjusting bandwidth (later developed into fret width) (FW). The number of musicians N is defined by the problem itself and is equal to the number of variables in the optimization function.

HS imposes fewer mathematical requirements and does not require initial value settings of the decision variables. Therefore, it is potential to be implemented in real-time systems like WSNs. When applied to solve the problems depicted in this context, HS consists of five main steps as explained below [17].

Step 1: Initialize the Optimization Problem and Algorithm Parameters

Optimization problem can depicted as

$$\begin{aligned} &\min\ f(\mathbf{x}) \\ &s.t.\quad x_i \in X_i, i = 1, 2 \ldots, N \end{aligned} \tag{10}$$

where $f(\mathbf{x})$ is the objective function; $\mathbf{x}$ is the solution vector which is composed of decision variable x_i, i=1, 2… N; the domain of each decision variable is X_i. For discrete variable $X_i = (x_i(1), x_i(2), \ldots, x_i(K))$. For continuous variable $X_i : x_i^L \le X_i \le x_i^U$; N is the number of decision variables; K is the possible number of discrete variable. *HMS*, *HMCR*, *PAR*, *FW*, *Tmax* should be initialized.

Step 2: Initialize the Harmony Memory (HM)

Randomly generate HM with a HMS number of solution vectors $x^1, x^2, \ldots x^{HMS}$. The HM with the size of HMS can be represented by

$$HM = \begin{bmatrix} x^1 & f(x^1) \\ x^2 & f(x^2) \\ \vdots & \vdots \\ x^{HMS} & f(x^{HMS}) \end{bmatrix} = \begin{bmatrix} x_1^1 & x_2^1 & \cdots & x_N^1 & f(x^1) \\ x_1^2 & x_2^2 & \cdots & x_N^2 & f(x^2) \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ x_1^{HMS} & x_2^{HMS} & \cdots & x_N^{HMS} & f(x^{HMS}) \end{bmatrix} \tag{11}$$

Step 3: Improvise a New Harmony Memory for the HM

Generate a new harmony vector $x_i^{'} = (x_1^{'}, x_2^{'}, \cdots, x_N^{'})$. $x_i^{'}$ is generated using

$$x_i^{'} = \begin{cases} x_i^{'} \in HM(i) \text{ with probability } HMCR \\ x_i^{'} \in x_{candiate} \text{ with probability } (1\text{-}HMCR) \end{cases} \quad (12)$$

where $HM(i)$ is the *i*th column of the HM. *HMCR* is defined as the probability of selecting a component from the $HM(i)$ members, and $(1\text{-}HMCR)$ is the probability of generating it randomly from the set of candidates. If $x_i^{'}$ is generated from the *HM*, then it is further modified or mutated according to *PAR*. The *PAR* determines the probability of a candidate from the HM to be mutated and $(1\text{-}HMCR)$ is the probability of doing nothing.

The pitch adjustment for the selected $x_i^{'}$ is given by

$$x_i^{'} = \begin{cases} x_i^{n} \in HM \text{ with probability } PAR \\ x_j^{'} \text{ with probability } (1 - PAR) \end{cases} \quad (13)$$

where x_i^n is the nearest node whose energy is greater than the average residual energy. After each new element x_i^n is selected, it is eliminated from the set of cluster head candidates to avoid duplicated node.

Step 4: Update the HM

The newly generated harmony vector is evaluated in terms of the objective function value. If the objective function value for the new harmony vector is better than the objective function value for the worst harmony in the HM, then new harmony is included in the HM and the existing worst harmony is excluded from the HM. The solution vector with the smallest fitness value can be considered the optimal solution of the problem in the current iteration.

Step 5: Go to Step 3 until Termination Criterion is reached

The current best solution is selected from the HM after the termination criterion is satisfied. This is the solution for the optimization problem formulated.

4. WSN Model Solution Based on HS

4.1. Encoding for Path Problem

When harmony search is applied to WSN data transmission, the difficult problem is how to encode a path in a network graph into harmony memory. Meanwhile, the encoding technique in turn affects the effectiveness of a solution/search process. There are several encoding schemes for solving this problem [18-21]. Here, a priority-based path encoding algorithm is adopted [18].

Suppose $N_{\max}$ is the node number in WSN. Node ID is from 1 to $N_{\max}$. V_p^k denotes the node set to be selected to enter transmission path. x^k represents the variable in harmony memory. This variable is assigned to random number between $-2/N_{\max}$ and $-2/N_{\max}$ .The path starts from 1 and generate one after another. When a new node joins V_p^k , a greater negative priority value $-2/N_{\max}$ is assigned to the variable corresponding to the node, and that node is difficult to be selected again. The more detailed description is as follows.

Step 1: **Initiate the Parameters.** k=1, $V_p^k \leftarrow \{1\}$, t^k =1, $x^k(t^k) = -2/N_{max}$;

Step 2: **Determine Whether Termination Criterion is reached**. If $t^k = N_{max}$ or $t^k \in V_p^{k-1}$, jump to Step 4; Otherwise, k=k+1, jump to Step 3.

Step 3: **Path Expansion**. Select node t^{k-1} having data link. Determine the node with the maximum priority as next node t^k, $V_p^k \leftarrow \{V_p^{k-1}, t^k\}$, $x^k(t^k) = -2/N_{max}$, jump to Step 2.

Step 4: **Path Completion**. If termination node obtained last is the objective node, the generated set V_p^k is the valid path; if termination node is not the objective node, the generated set V_p^k is invalid path.

Take N =20 for example, generate 20 nodes randomly within 100*100 region. The transmission radius of each node is R. Here R=50. If the distance between two nodes is less than 50, they are adjacent nodes. Data package is transmitted from node 1 to node 20. The network topological structure is Figure 1. Meanwhile, randomly generate encoding scheme as shown in Table 1.

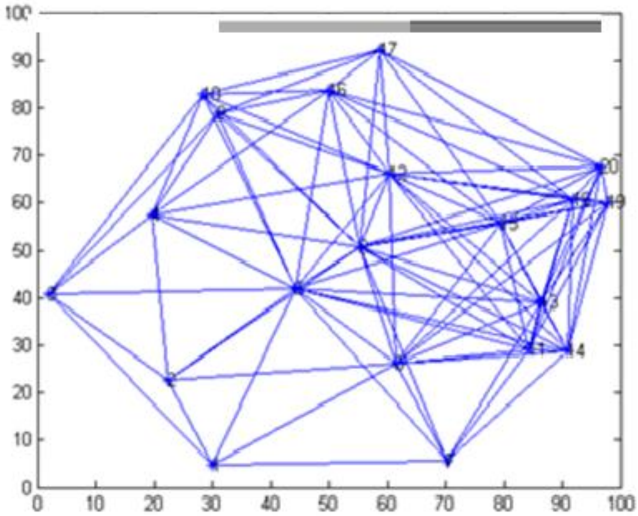


Figure 1. Topology of the Network

From the topological structure, node 1 can exchange data with node 2, 3, 5, 6, 7. Compare the variable priority value corresponding to 2, 3, 5, 6, 7. Select node 3 as the node of next hop. Assign -10 to node 1 and make sure that node 1 will not be selected as intermediate node in the following search. Meanwhile, node 3 has data linkage with node 2, 4, 5,9,10. Determine node as the next intermediate node. In the same way, transmission path V = {1, 3, 4, 8, 17, 20} is obtained.

4.2. Flow Chart Based on HS

According to the network model, two factors, network delay and communication energy consumption constraint the network performance. Trough different requirements about the above two aspects, delay and energy consumption have different weights. Delay weight is α and consumption weight is β . $\alpha + \beta = 1$. The quality of harmony vector is estimated by the objective function. By minimizing (maximizing) the objective function, the global optimal solution is obtained. The objective in this text can be expressed as

$$f = \alpha f_1 + \beta f_2 \quad (14)$$

Considering the two aspects of network delay and energy consumption, the flow chart of the algorithm is obtained as shown in Figure 2.

Table 1. Encoding Scheme

Node	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Priority value variable	-9	2	6	5	-8	3	-8	10	2	-7	-2	6	-2	-2	2	-6	9	-5	-6	4

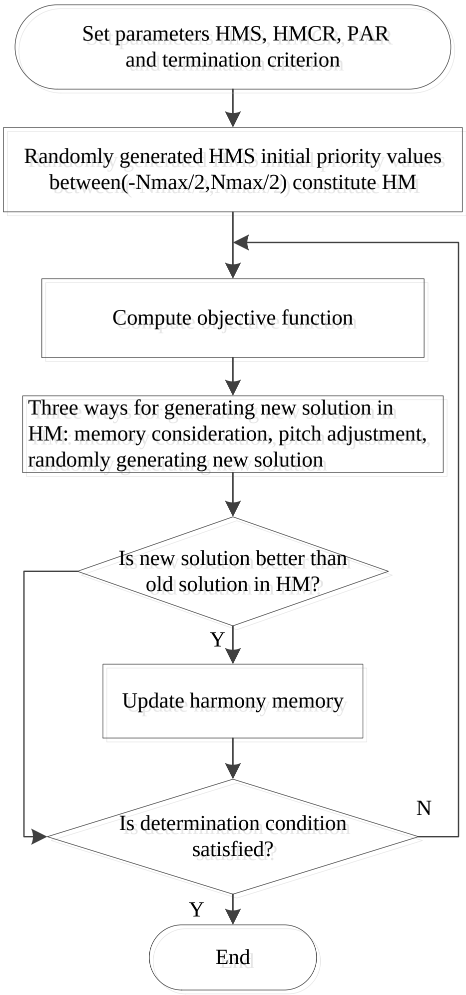


Figure 2. Flow Chart of the Algorithm

4.3. Results

Simulation of 100-nodes WSN is carried out using harmony search algorithm under MATLAB. The distance between adjacent nodes, *R*, is less than 30.The effect of different network delay and energy consumption on the network data transmission path is analyzed.

For *N*=10, nodes distribution is shown in Figure 3.

For α =0, β =1; α =0.5, β =0.5; α =1, β =0; number of iterations *Tmax* =1200. In Figure 4(a), horizontal coordinate is the number of iterations and vertical coordinate is energy consumption. In Figure 4(b), horizontal coordinate is the number of iterations and vertical coordinate is the number of delay.

In Figure 4, after about 600 iterations, stable data transmission path can be obtained. Simulation results show that harmony search algorithm can implement the path establishment for wireless sensor network.

Consumption and delay number is shown in Table 2.

For α =0, β =1, consider energy consumption alone, the energy consumption is 2695.86 and delay is 10.4.

For α =1, β =0, consider delay alone, energy consumption is 3355.73 and delay is 6.2.

For α =0.5, β =0.5, energy consumption is 2851.1 and delay is 7.4.

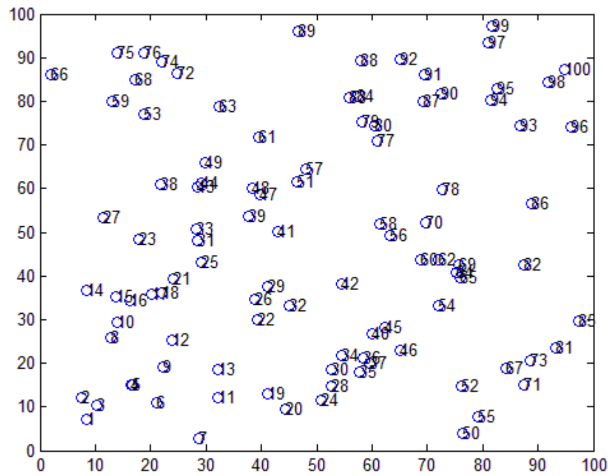


Figure 3. Nodes Distribution

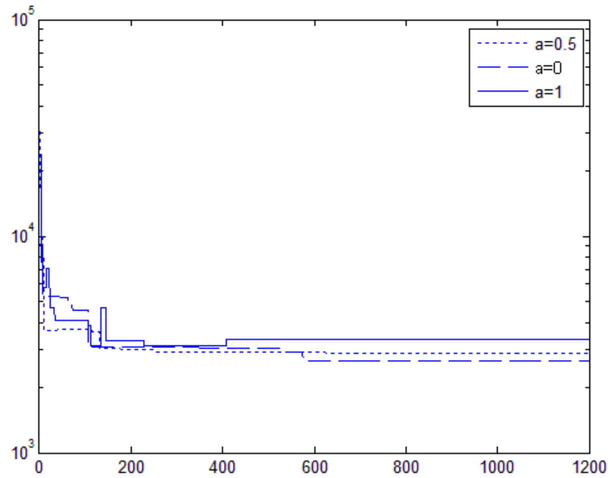


Figure 4 (a). Energy of Network

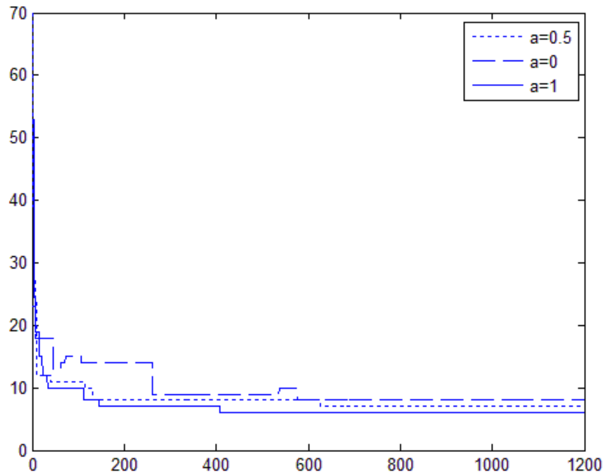


Figure 4 (b). Delay of Network

5. Conclusions

Harmony search is applied to solve the low-delay and low-energy double-objective optimization problem in wireless sensor networks. Delay and energy consumption model is analyzed. While searching the optimal path, a priority-based path encoding algorithm is adopted to update the harmony memory. After simulation under Matlab, interaction between network delay and energy consumption is vilified. According to the different requirements, the corresponding optimal path is obtained.

Table 2. Data of Energy and Delay

Aim	Weight	Run time					Mean
		1	2	3	4	5	
Energy consumption($C1*m^2$)	$\alpha=0$, $\beta=1$	2858.99	2436.55	2731.81	2747.57	2704.38	2695.86
	$\alpha=0.5$ $\beta=0.5$	2924.11	2964.95	2733.57	2779.11	2853.77	2851.10
	$\alpha=1$, $\beta=0$	3281.13	3354.57	3385.56	3337.92	3419.49	3355.73
Delay ($C2*s$)	$\alpha=0$, $\beta=1$	10	12	11	10	9	10.4
	$\alpha=0.5$ $\beta=0.5$	7	7	7	7	9	7.4
	$\alpha=1$, $\beta=0$	6	7	6	6	6	6.2

Acknowledgements

The authors wish to thank the reviews for their helpful suggestions. This work was supported by National natural science fund of China (No. 61463028)

Fundamental Research Funds for Colleges and Universities in Gansu Province (No. 213054), and Research Project of Gansu Education Department (No. 213027).

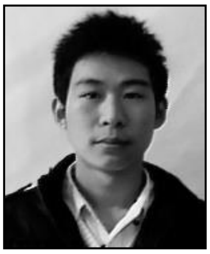
References

[1] C. T. Cheng, C. K. Tse, and C. M. Francis, IEEE Sensors Journal, vol. 3, no. 11, **(2011)**.
[2] S. D. Muruganathan, D. C. F. Ma, R. I. Bhasin, and A. Fapojuwo, IEEE Communications Magazine, vol. 3, no. 43, **(2005)**.
[3] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, IEEE Translations on wireless communications, vol. 1, no. 4, **(2002)**.
[4] S. Lindsey, and C. S. Raghavendra, "PEGASIS: Power-efficient gathering in sensor information systems", In Proc. IEEE Conf. Aerosp, Big Sky, MT, USA, **(2002)** March, pp.1125–1130.
[5] R. Fonseca, O. Gnawali, K. Jamieson, S. Kim, P. Levis, and A. Woo, "The collection tree protocol", SenSys'09, **(2009)** November 4–6, Berkeley, CA, USA.
[6] M. Mahdavi, M. Fesanghary, and E. Damangir, "An improved harmony search algorithm for solving optimization problems", Applied Mathematics and Computation, vol. 88, no. 2, **(2007)**, pp.1567- 5791.
[7] Z. W. Geem, J. Kim, and G. Loganathan, "Simulation", vol. 76, no. 2, **(2001)**.
[8] S. L Kang, and Z. W. Geem, "A new structural optimization method based on harmony search algorithm", Computers and Structures, vol. 82, nos. 9-10, **(2004)**, pp. 781−798.
[9] Z. W. Geem, K. S. Lee, and Y. Park, American Journal of Applied Sciences, vol. 12, no. 2, **(2005)**.
[10] Z. W. Geem, Engineering Optimization, vol. 3, no. 38, **(2006)**.
[11] Z. W.Geem, Lecture Notes in Computer Science, vol. 4507, **(2007)**.
[12] C. Li, and H. Zhao, X. Ge. ACTA ELECTRONICA SINICA, vol. 12, no. 37, **(2009)**.
[13] M. Fesangharya, M. Mahdavib, M. M. Jolandanc, and Y. Alizadeha, Computer Methods in Applied Mechanics and Engineering, vol. 197, **(2008)**.
[14] J. H. Kim, Z. W. Geem, and E. S. Kim, Journal of the American Water Resources Association, vol. 5, no. 37, **(2001)**.
[15] D. C. Hoang, P. Yadav, R. Kumar, S. K. Panda, P. Yadav, R. Kumar, and S. K. Panda, IEEE Transactions on Industrial Informatics, vol. 1, no. 10, **(2014)**.
[16] D. C. Hoang, P. Yadav, R. Kumar, and S. K. Panda, 2010 IEEE International Conference on Communications Workshops, **(2010)** May 23-27, Cape town, South Africa.
[17] D. Ren, and S. Qiang, IEEE Transactions on System, Man, and Cybernetics—Part B: Cybernetics, vol. 6, no. 42, **(2012)**.
[18] A. W. Mohemmed, N. C. Sahoo, and T. K. Geok, Applied Soft Computing, vol. 4, no. 8, **(2008)**.
[19] C. W. Ahn, and R. S. Ramakrishna, IEEE Transactions on Evolutionary Computation, vol. 6, no. 6, **(2002)**.
[20] F. B. Zahn, and C. E. Noon, Transportation Science, vol. 1, no. 32, **(1998)**.
[21] W. Huang, and L. X. Ding, IEEE Transactions on Communications, vol. 11, no. 60, **(2012)**.

Authors

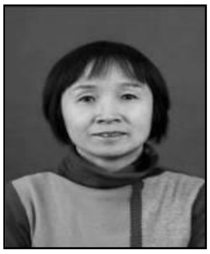
Zhen-Rui PENG, was born in Gansu Province, China, in 1972. He received the B.S. degree and M.S. degree from Lanzhou Jiaotong University, Lanzhou, in 1995 and in 2001 respectively, both in mechanical engineering. He received the Ph.D. degree from Zhejiang University, Hangzhou, in 2007, in control science and engineering. He is now with the school of mechatronic engineering, Lanzhou Jiaotong University. His research interests include measurement and control, intelligence optimization.

Hong YIN, was born in Xinjiang Uygur Autonomous Region, China, in 1978. She received the B.S. degree from Lanzhou Jiaotong University, Lanzhou, in 2000, in mechanical engineering. She received the M.S. degree from Lanzhou Jiaotong University, Lanzhou, in 2003, in vehicle engineering. She is currently pursuing the Ph.D. degree with the school of mechatronic engineering, Lanzhou Jiaotong University. Her research interests include measurement and control, intelligence optimization.



Hai-Tang DONG, Hong Yin was born in Gansu Province, China, in 1973. She received the B.S. degree from Shanghai Railway University, Shanghai, in 1995, in electrical engineering. She received the M.S. degree from Lanzhou Jiaotong University, Lanzhou, in 2001, in vehicle engineering. Her research interests include measurement and control.

Hui LI, was born in Anhui Province, China, in 1990. He received the B.S. degree from Nanjing Institute of Technology, Nanjing, in 2011, in electrical engineering. He received the M.S. degree from Lanzhou Jiaotong University, Lanzhou, in 2014, in mechanical engineering. He is now with Shanghai Railway Bureau. His research interests include wireless sensor network.



An PAN, was born in 1986 and grew up in Gansu Province, China. He received the B.S. degree from North University of China, Taiyuan, in 2008, in optical information science and technology. He is currently pursuing the M.S. degree with the school of mechatronic engineering, Lanzhou Jiaotong University. He research interests include optimal sensor placement.

