Journal of Computers Vol. 29 No. 6, 2018, pp. 16-28
doi:10.3966/199115992018122906002

Neural Fuzzy Controller Based Transmission Power Control for Wireless Sensor Networks

Chu-Hang Wang[1], Man Zheng[2*], Wei-Na Shen[2]

[1] College of Computer and Science, Changchun Normal University, Changchun, China
wangchuhang@cncnc.edu.cn

[2] College of Computer Science and Engineering, Changchun University of Technology, Changchun, China
644595756@qq.com; swn0715@163.com

Received 5 July 2017; Revised 14 November 2017; Accepted 9 January 2018

Abstract. Properly adjusting the transmission power of the nodes in wireless sensor networks can reduce the energy consumption significantly. However, ignoring the variety of energy will make nodes with lower energy transmit data packets with higher power level to enter premature death state. Besides, lack of learning ability on the existing data set inevitably restricts the network scalability and applications in different environment. This paper introduces a self-adaptive Neural Fuzzy controller based Transmission power Control approach (NFTC) which aims to adjust the transmission power of the nodes dynamically. In NFTC, each node contains a fuzzy controller that consists of two inference engines whose parameters is provided from a neural network with a training data set and an if-then rules base respectively. Moreover, the outputs are feedbacked to the fuzzy controller in order to adapt to the change of packet reception ratio with respect to the residual energy. Consequently, NFTC reduces the actual energy consumption while makes the packet reception ratio be close to the desired value, and extends the network lifetime. The validation experiment results show NFTC outperforms its counterparts in terms of average packet reception ratio, total residual energy as well as network lifetime.

Keywords: balanced energy consumption, neural fuzzy controller, packet reception ratio, transmission power control, wireless sensor networks

1 Introduction

Wireless sensor networks (WSNs) are envisioned to be a major enabling technology for Cyber-Physical Systems (CPS) and Internet of Things (IOT) paradigm [1-2], which consist of a certain number of tiny sensor nodes with low power and finite storage, processing and communication abilities. Although being widely deployed in several application scenarios such as environmental monitoring, military surveillance, e-health and scientific exploration [3-4], WSNs still face a big challenge to maximize network lifetime under a constrained energy. Adjusting the transmission power of the individual nodes has shown to be an effective approach to reduce the energy consumption while at the same time to preserve the communication reliability [5].

It has been experimentally shown that transmission power has significant impact on link quality [4, 6], which means a high transmission power level provides a good link quality at the expense of increasing the energy consumption, and a low transmission power level degrades link quality while reducing the energy consumption. Therefore, most of the recent studies on Transmission Power Control (TPC) employ link level strategies to maximize WSN lifetime and improve network performance [3, 7]. Usually, the variation of link quality metrics such as Packet Reception Ratio (PRR), Reception Signal Strength Indicator (RSSI) and Link Quality Indicator (LQI) is used to adapt the transmission power [1, 3, 8-10]. However, they are susceptible to environmental interfere and network dynamics. So intelligent control techniques such as fuzzy control are used for developing adaptation strategies on dynamics of WSN and

* Corresponding Author

environment as well as constraints of the linear model, and the results show that these strategies can tolerant the uncertain interference and converge fast to keep the network stable, energy-efficient and communication-reliable [11-13].

This paper presents a novel self-adaptive Neural Fuzzy controller based Transmission power Control approach (NFTC) to adjust the transmission power of the nodes dynamically. In NFTC, the fuzzy logic controller consists of two inference engines. The one is responsible for adjusting the node transmission power, while the other is responsible for adjusting the desired packet reception radio according to the node's residual energy. Moreover, the parameters of the first inference engine are from a neural network with a training data set, and the parameters of the second inference engine are from an if-then rule base. Through the closed-loop feedback processes, the neural fuzzy controller can be adapt to the changes of packet reception ratio with respect to the residual energy, accordingly, control the transmission power of the nodes properly.

The rest of the paper is organized as follows. Section 2 gives a short survey of the related works. The system model is described in Section 3. Section 4 designs the neural fuzzy controller in detail. Section 5 provides the simulation results, and finally the conclusion is presented.

2 Related Works

Usually, transmission power control approaches concentrate on maintaining the lowest transmission power level compatible with the acceptable link quality, which are categorized into three major groups: network level, node level and link level [5, 7]. In network level solutions [14-15], a single transmission power for the whole network is adopted to achieve course tuning of power control, which inevitably leads to high energy consumption as well as not making full use of the configurable transmission power provided by radio hardware. In node level solutions [11, 16], an optimal transmission power is selected to maintain the communication between pair of nodes or among a node and its all neighbors, in order to reduce energy consumption while keeping communication reliability [1-2, 17]. However, the WSN is inevitably dynamic since the nodes will be quit or added to the network, and the residual energy ignorance will undoubtedly lead to unbalanced energy consumption with node premature death.

Recently, most of the studies employ link level strategies to control transmission power so as to maximize WSN lifetime [3, 18-21]. A link adaptation algorithm is proposed in [18] to adjust the transmission power level and the data rate by using the link quality information available at the transmitter. The channel quality is measured as reception or non-reception of the receiver's acknowledgment with respect to the power level and data rate in order to select the highest possible data rate under each link quality and adjust the transmission power accordingly. Moreover, a theoretical analysis of transmission power control is presented in [19], which employs the channel feedback obtained from the reception or non-reception packets of the receiver's acknowledgment. The channel is modeled as a finite state Markov channel and a dynamic programming solution for the finite horizon transmission power control problem is proposed. In [20], a transmission power control scheme is proposed to improve the WSN energy efficiency, in which the minimum transmission power level is used for data transmission on each link that ensures a predetermined target packet error probability whereas control packets are transmitted using the maximum power level. In [21], an approach to monitor link quality continuously for multiple transmission power levels is proposed, which enables the selection of lowest transmission power level that achieves the target reliability level. In [3], a lightweight algorithm for adaptive transmission power control in WSN is proposed. In this algorithm, each node builds a model for each of its neighbors to describe the correlation between transmission power and link quality, and with this model, a feedback-based transmission power control is used to dynamically maintain individual link quality over time. However, extensive empirical studies have shown that link quality is so largely influenced by the time and environment [7, 22] that it is nondeterministic to real world deployments.

Consequently, fuzzy logic is used to deal with the uncertainty of ambiguity [11, 13, 23]. Fuzzy logic is characterized by model free, which means it can dispose of accidental interference and uncertain factors in transmission power control. In [11], a close-loop transmission power adjustment method based on fuzzy control theory is applied upon WSN, in which each node acts as a controller and its neighbor nodes as a plant, and the control action is depend on the number of its neighbor nodes. With the control system, uncertain interference in the network can be efficiently overcome and energy consumption can be reduced significantly. In [2], a self-adaptive system through two feedback control loops based on fuzzy control is proposed. In this system, the primary feedback control loop adjusts the node transmission power considering both its real and targeted number of neighbors, and the secondary feedback control loop manages the targeted node number of neighbors considering the battery level. The simulation showed that the self-adaptive system allows the nodes in the network to achieve a balance between a good enough power saving while keeping a high reliability of communications. Moreover, a novel localized fuzzy logical approach to adaptively control the transmission power of each node is proposed in [17] so as to achieve the desired node degree. Especially, in this approach the fuzzy logic controller is constructed from the training data set. Accordingly, it is proved to be accurate, stable and with short settling time. However, ignoring the variety of energy will make nodes with lower energy transmit data packets with higher power level to enter premature death state. Besides, lack of learning ability on the existing data set inevitably restricts the network scalability and applications in different environment.

In this paper, the fuzzy logic system serves as two inference engines for each sensor node to modify its transmission power according to its residual energy while keeping the desired PRR. Moreover, unlike other fuzzy logic control methods for WSNs, the parameters of the first inference engine are from a neural network with a training data set, and the parameters of the second inference engine are from an if-then rules base. Therefore, our proposal is more flexible to deal with network dynamics while keeping balanced energy consumption.

3 System Model

NFTC can dynamically adjust the transmission power of the nodes by a neural fuzzy controller. It can reduce the computation and improve the adaptability of the system at the same time. In this section, the system model is described in detail including the network model and energy model.

3.1 Network Model

In order to simplify the network, the assumptions on the network properties are made as follows:

- Nodes are distributed in a square field randomly, and each node has a unique identity.
- Nodes are stationary after deployment with limited energy.
- Nodes are homogenous in terms of initial energy, processing power, memory, transmission and reception capabilities.
- Nodes can obtain their own PRR and residual energy.

3.2 Energy Model

The energy dissipated by transmitting l-bit message to the distance d is given by:

$$E_{tx} = \begin{cases} l * E_{elec} + \varepsilon_{fs} * d^2, & if\ d < d_0 \\ l * E_{elec} + \varepsilon_{mp} * d^4, & if\ d \geq d_0 \end{cases} \tag{1}$$

where E_{elec} is the transmission energy to run the transmitter or receiver circuitry and ε_{fs}, ε_{mp} are energy dissipation values to run the amplifier for close and far distances with the threshold $d_0 = \sqrt{\varepsilon_{fs} / \varepsilon_{mp}}$ respectively. Energy consumed in receiving l-bit message is calculated as follows:

$$E_{rx} = l * E_{elec} \tag{2}$$

Moreover, energy consumption due to data aggregation with l-bit is represented in Eq. (3).

$$E_{DA} = l * E_{pDb} \tag{3}$$

where E_{pDb} is energy consumption for single bit data aggregation.

4 Design of NFTC

With respect to the power saving and reliability of the wireless sensor networks, it seems an impertinent approach to only apply a single fuzzy logic system when adjusting transmission power. Therefore, a neural fuzzy controller based system including two inference engines is provided to adaptively adjust the transmission power according to the residual energy of the node with desired PRR. By means of applying the self-learning neural network, the first inference engine (FIE) can learn from training data set to control transmission power, and by using the rules from domain experts, the second inference engine (SIE) can adjust the targeted PRR according the residual energy. The architecture of NFTC is depicted in Fig. 1.

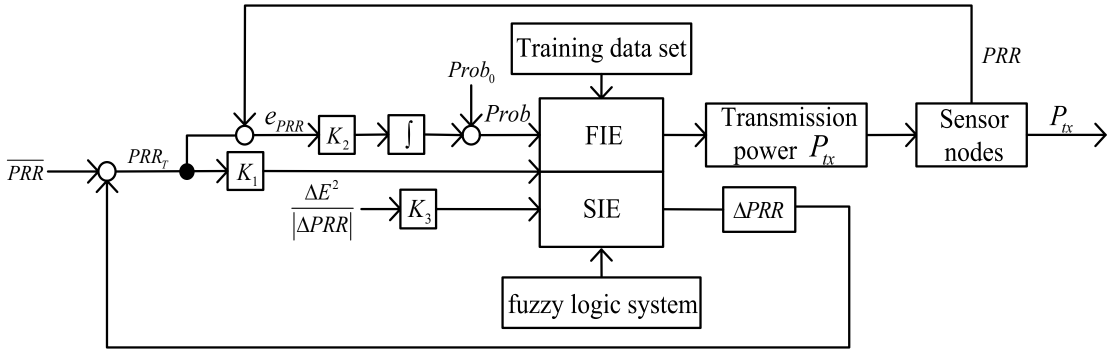

Fig. 1. Architecture of NFTC

For the convenience of reading, the parameters used in this paper is shown in Table 1.

Table 1. The parameters used in this paper

Parameter	Meaning of the parameter
PRR	packet reception ratio
$\overline{PRR}$	desired packet reception ratio
PRR_T	targeted packet reception ratio
e_{PRR}	difference between PRR and PRR_T
$Prob_0$	initial probability that a node has $\overline{PRR}$
$Prob$	probability that a node has $\overline{PRR}$
P_{tx}	transmission power
ΔPRR	regulation amount of packet reception ratio
$\frac{\Delta E^2}{\lvert\Delta PRR\rvert}$	tendency of energy consumption in terms of PRR
K_1, K_2, K_3	scale coefficient

4.1 Input and Output

As seen from Fig. 1, NFTC consists of two inference engines which have a common input denoted by desired packet reception ratio $\overline{PRR}$. Also $\overline{PRR}$ is used to calculate the targeted packet reception ratio PRR_T by adding the change to be applied ΔPRR estimated by SIE based on the residual energy. In addition, the other input of FIE is the probability $Prob$ that a node has PRR_T. Moreover, adjusting the transmission power is a very common capability in many sensor nodes, hence, the output of FIE is the transmission power P_{tx}. On the other hand, the second input of SIE is the ratio of the residual energy deviation and PRR deviation denoted by $\frac{\Delta E^2}{|\Delta PRR|}$ which indicates the tendency of energy consumption in terms of PRR.

The probability that the successful packet reception of a φ-byte packet transmitted at power level P_{tx} between pair of nodes i and j is given by Eq. (4).

$$p_{ij}(P_{tx},\varphi)=[1-\frac{1}{2}exp(\frac{P_n+P_0+10nlog_{10}(d_{ij}/d_0)+X\sigma-P_{tx}}{2})]^{8\varphi} \quad \textbf{(4)}$$

where d_{ij} is the distance between transmitter and receiver, d_0 is the reference distance, P_0 is the path loss at the reference distance, n is the path loss exponent, and $X\sigma$ is a zero-mean Gaussian random variable with standard deviation σ. And P_n is the noise floor which is usually -145dB at the temperature of 300 Kelvin for Mica motes [3, 7]. Like in [3, 7], the parameter values provided for Mica motes as $n=4$, $\sigma=4$, $d_0=1m$, and $P_0=55dB$ are adopted. As illustrated in Fig. 1 and Eq.(4), the inputs of FIE are PRR_T and $Prob$, and the output is P_{tx}. Given the above parameter values, $PRR_T \in \{k_1,k_2,...,k_m\}$ and $P_{tx} \in \{p_1,p_2,...,p_n\}$, then $Prob=f(PRR_T,P_{tx})$ can be calculated from Eq. (4). The training data set T is a $s\times 3$ matrix in the form of $[PRR_T,Prob,P_{tx}]$, where $s=m\times n$.

4.2 The First Inference Engine

As shown in Fig. 1, the neural fuzzy controller consists of two inference engines, the first inference engine can learn from the training data set by a neural network. The architecture of the first inference engine is depicted in Fig. 2.

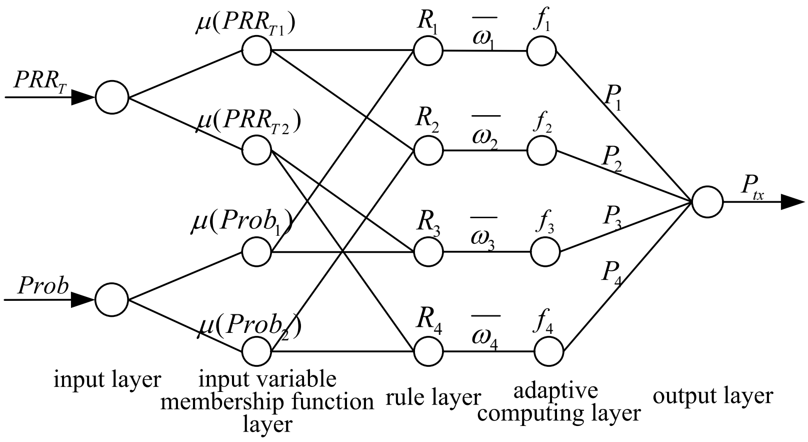

Fig. 2. Architecture of the first inference engine

The first inference engine consists of four layers which are described respectively as follows.

Input layer. The network has two inputs, namely PRR_T and $Prob$.

Membership functions layer. According to the collected data of PRR_T, $Prob$ and P_{tx}, the training data set $[PRR_T,Prob,P_{tx}]$ is obtained which is used for training the model. For the j_{th} data set, Gauss transformation is used to fuzzy the input variables. Membership function of each variable is given by Eq. (5).

$$\begin{cases}\mu(PRR_{Ti})=\exp(-(PRR_T{}^j-c_j^i)^2/b_j^{i\,2}) \\ \mu(Prob_i)=\exp(-(Prob^j-c_j^i)^2/b_j^{i\,2})\end{cases} \quad (i=1,2) \quad \textbf{(5)}$$

where i is the number of fuzzy subsets, c_j^i,b_j^i are the centre value and width of membership functions.

Rule layer. This layer is used to carry out fuzzy operation. The outputs are normalized values of each neuron input after multiplication, that is, normalization for incentive strength of each rule. Each node output is given by Eq. (6).

$$\begin{cases} \omega_1 = \mu(PRR_{T1}) \cdot \mu(Prob_1) \\ \omega_2 = \mu(PRR_{T1}) \cdot \mu(Prob_2) \\ \omega_3 = \mu(PRR_{T2}) \cdot \mu(Prob_1) \\ \omega_4 = \mu(PRR_{T2}) \cdot \mu(Prob_2) \end{cases} \Rightarrow \begin{cases} \overline{\omega_1} = \dfrac{\omega_1}{\omega_1 + \omega_2 + \omega_3 + \omega_4} \\ \overline{\omega_2} = \dfrac{\omega_2}{\omega_1 + \omega_2 + \omega_3 + \omega_4} \\ \overline{\omega_3} = \dfrac{\omega_3}{\omega_1 + \omega_2 + \omega_3 + \omega_4} \\ \overline{\omega_4} = \dfrac{\omega_4}{\omega_1 + \omega_2 + \omega_3 + \omega_4} \end{cases} \tag{6}$$

Adaptive computing layer. This layer is combined with four control rules to complete the adaptive operation and calculate the output decided by each rule. The output in this layer is given by Eq. (7).

$$\begin{cases} P_1 = \overline{\omega_1} f_1 = \overline{\omega_1}(p_1 \cdot PRR_T{}^j + q_1 \cdot Prob^j + r_1) \\ P_2 = \overline{\omega_2} f_2 = \overline{\omega_2}(p_2 \cdot PRR_T{}^j + q_2 \cdot Prob^j + r_2) \\ P_3 = \overline{\omega_3} f_3 = \overline{\omega_3}(p_3 \cdot PRR_T{}^j + q_3 \cdot Prob^j + r_3) \\ P_4 = \overline{\omega_4} f_4 = \overline{\omega_4}(p_4 \cdot PRR_T{}^j + q_4 \cdot Prob^j + r_4) \end{cases} \tag{7}$$

where $\{p_i, q_i, r_i\}$ is the conclusion parameter of the node.

Output layer. Predicted by the targeted packet reception ratio PRR_T and the probability $Prob$ that a node has PRR_T, the total output of network training indicates the node transmission power P_{tx}, whose value is the sum of the outputs of four nodes in the adaptive computing layer, which is given by Eq. (8).

$$P_{tx} = P_1 + P_2 + P_3 + P_4 \tag{8}$$

Calculating Eq. (9) with the integration of Eq. (6), (7), (8), the output value P_{tx} of this network is obtained as Eq. (9).

$$\begin{aligned} P_{tx} = [&\mu(PRR_{T1}) \cdot \mu(Prob_1) \cdot (p_1 \cdot PRR_T{}^j + q_1 \cdot Prob^j + r_1) + \\ &\mu(PRR_{T1}) \cdot \mu(Prob_2) \cdot (p_2 \cdot PRR_T{}^j + q_2 \cdot Prob^j + r_2) + \\ &\mu(PRR_{T2}) \cdot \mu(Prob_1) \cdot (p_3 \cdot PRR_T{}^j + q_3 \cdot Prob^j + r_3) + \\ &\mu(PRR_{T2}) \cdot \mu(Prob_2) \cdot (p_4 \cdot PRR_T{}^j + q_4 \cdot Prob^j + r_4)] / \\ &[\mu(PRR_{T1}) \cdot \mu(Prob_1) + \mu(PRR_{T1}) \cdot \mu(Prob_2) + \\ &\mu(PRR_{T2}) \cdot \mu(Prob_1) + \mu(PRR_{T2}) \cdot \mu(Prob_2)] \end{aligned} \tag{9}$$

The purpose of fuzzy neural network controller learning is to determine the controlled parameters and control rules according to the actual input and output training set. The error function of learning in FIS is given by Eq. (10).

$$e = \frac{1}{2}(P_{txd} - P_{txc})^2 \tag{10}$$

where P_{txd} and P_{txc} are the expected output and actual output transmission power value. In the course of learning, the parameters to adjust are weight ω_i, the central value c_j^i and width b_j^i of the Gauss type membership function. Formulas for adjustment are given by Eq. (11-13).

$$\omega_j^i(k) = \omega_j^i(k-1) - \alpha \frac{\partial e}{\partial \omega_j^i} \tag{11}$$

$$c_j^i(k) = c_j^i(k-1) - \alpha \frac{\partial e}{\partial c_j^i} \quad \textbf{(12)}$$

$$b_j^i(k) = b_j^i(k-1) - \alpha \frac{\partial e}{\partial b_j^i} \quad \textbf{(13)}$$

where k is the learning frequency, and α is the network learning rate. The fuzzy neural network achieves the desired control effect by constantly learning.

4.3 The Second Inference Engine

For the second inference engine, the two input variables are PRR_T and $\frac{\Delta E^2}{|\Delta PRR|}$, the output variable is the deviation of the packet reception ration ΔPRR. According to the variation of energy consumption, the fuzzy inference controller SIE can adjust PRR_T through a closed-loop feedback so as to get the appropriate output P_{tx}, thus the energy consumption is reduced. The second inference engine comprises of fuzzification, fuzzy rule and defuzzification which are described in detail next.

Fuzzification. The crisp values of inputs need to be changed into fuzzy linguistic variables. For the inputs, "LOW", "LOW_MIDDLE", "MIDDLE", "MIDDLE_HIGH", "HIGH" is the fuzzy linguistic variable for PRR_T whose crisp values are-2,-1,0,1,2. "SMALL", "MIDDLE", "LARGE" for $\frac{\Delta E^2}{|\Delta PRR|}$ with the crisp values 0, 1, 2. And "LOW", "HIGH", "SMALL" and "LARGE" follows trapezoidal membership function. "LOW_MIDDLE", "MIDDLE", "MIDDLE_HIGH" follows triangle membership function. The membership function for input variables PRR_T and $\frac{\Delta E^2}{|\Delta PRR|}$ is depicted in Fig. 3 and Fig. 4, respectively.

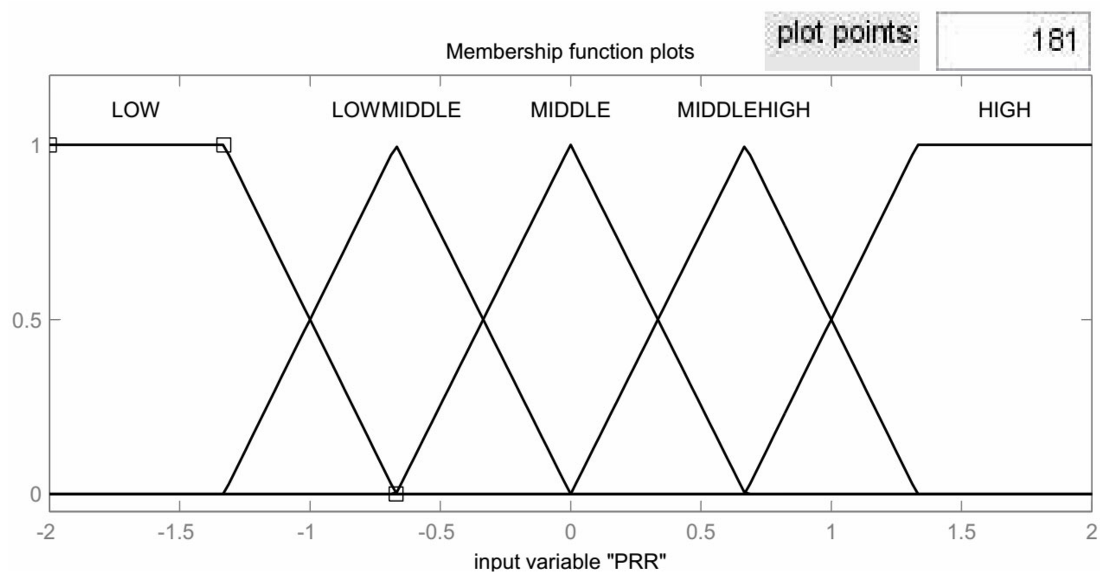

Fig. 3. Membership function for PRR_T

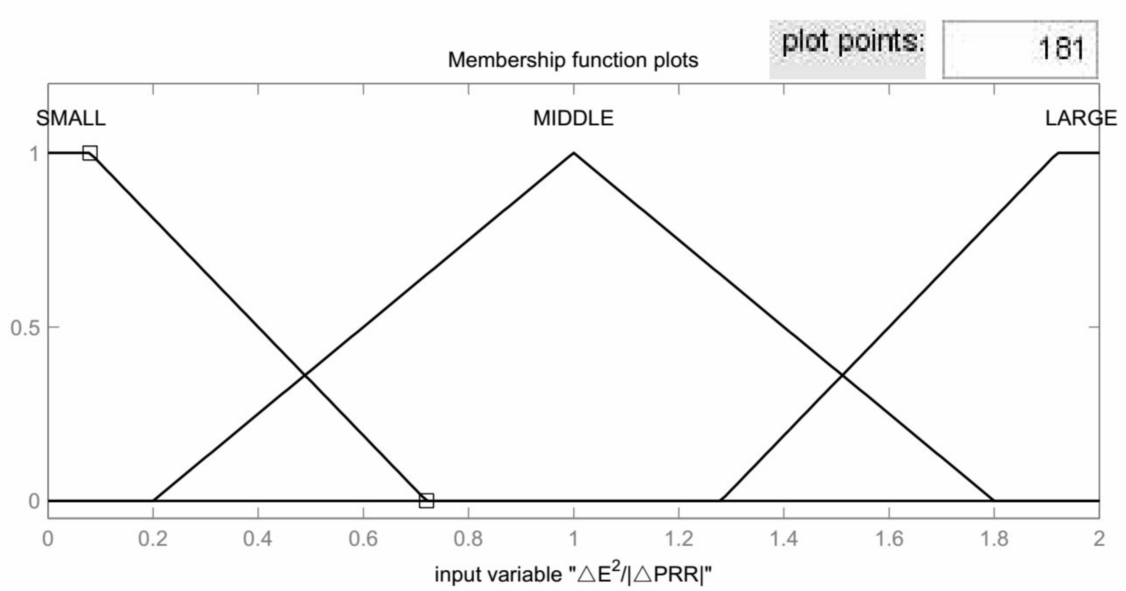

Fig. 4. Membership function for $\frac{\Delta E^2}{|\Delta PRR|}$

The fuzzy output variable ΔPRR has "D2S", "D1S", "HOLD", "U1S", "U2S" as its five output linguistic variables, whose crisp values are-2, -1, 0, 1, 2. In these variables, "D2S", "U2S" have trapezoidal membership function, 'D1S', 'HOLD', 'U1S' have triangle membership function. Fig. 5 shows the ΔPRR membership functions.

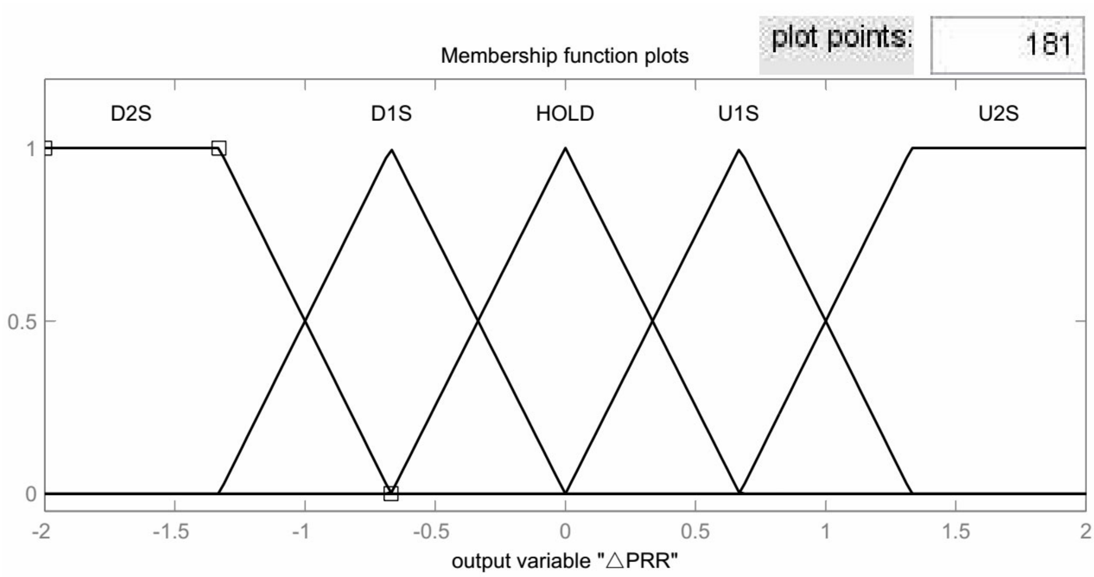

Fig. 5. Membership function for ΔPRR

Fuzzy rules and defuzzification. The crisp input values are fuzzified to appropriate linguistic variables by fuzzy inference system using the given membership functions. And then the fuzzified input variables are processed through the fuzzy if-then rule base. The rules are developed based on Mamdani method, which was simpler and yields better results [11, 17]. In total, 15 rules are there based on the combination of different linguistic variables which are specified in Table 2. Afterwards, center of area method is used to defuzzify the output to a crisp value ΔPRR . The specific defuzzification process is given by Eq. (14).

$$\Delta PRR = \frac{\sum_{i=1}^{n} x_i \mu_{\Delta PRR}(x_i)}{\sum_{i=1}^{n} \mu_{\Delta PRR}(x_i)} \quad (14)$$

Table 2. NFTC fuzzy rules

S.no	Input variables		Output variable
	PRR_T	$\frac{\Delta E^2}{\lvert\Delta PRR\rvert}$	ΔPRR
1	LOW	SMALL	U2S
2	LOW	MIDDLE	U2S
3	LOW	LARGE	U1S
4	LOW_MIDDLE	SMALL	U2S
5	LOW_MIDDLE	MIDDLE	U1S
6	LOW_MIDDLE	LARGE	U1S
7	MIDDLE	SMALL	HOLD
8	MIDDLE	MIDDLE	HOLD
9	MIDDLE	LARGE	HOLD
10	MIDDLE_HIGH	SMALL	D1S
11	MIDDLE_HIGH	MIDDLE	D1S
12	MIDDLE_HIGH	LARGE	D2S
13	HIGH	SMALL	D1S
14	HIGH	MIDDLE	D2S
15	HIGH	LARGE	D2S

5 Simulation Results

In order to verify the performance of NFTC, simulation tests are presented in this section using MATLAB. In the simulations, 100 nodes are deployed randomly in a square field of area $100m \times 100m$ with BS location $(50,50)$, and the initial energy of each node is 1J. Firstly, the effect of PRR and transmission distance on the transmission power is investigated. Then the comparison is implemented among the algorithms NFTC, FTC (Fuzzy-logic Topology Control) [17] and FCTP (Fuzzy Controller for Transmission Power) [11] in terms of average PRR and total residual energy. Every simulation result is the average of 50 independent experiments, and the parameters of the simulations are listed in Table 3.

Table 3. Simulation parameters

Parameters	Values
l	4000bits
node initial energy	1 J
E_{elec}	$50\ nJ \cdot bit^{-1}$
ε_{fs}	$10\ pJ \cdot bit^{-1}$
ε_{mp}	$0.0013\ pJ \cdot bit^{-1}$
d_0	$87\ m$
E_{pDb}	5nj/bit
data packet size	500 *bytes*
control packet size	25 *bytes*

Fig. 6 shows the average PRR with different transmission power levels from -20dBm to 20dBm in steps of 5dBm. We run the simulation for fifty times under each power levels. So we achieve the average PRR when the transmission power level varies from smaller to bigger. From Fig. 6 we can know average PRR increases with the increasing of the transmission power level, and tends to 100%.

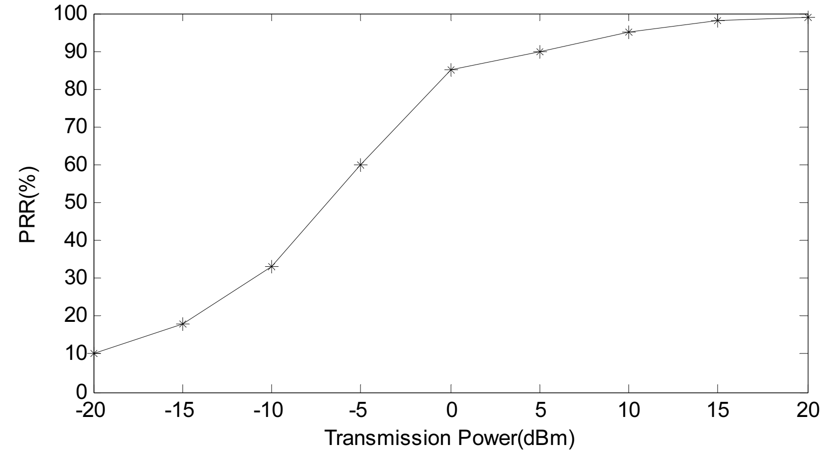

Fig. 6. Average PRR versus transmission power level

The distance between nodes is also another factor that may significantly affect PRR. We run the simulation conducted in a parking lot with MICAz motes for fifty times with each distance value from 1m to 30m, and calculate the average PRR accordingly. The results are listed in Table 4. The results show that PRR decreases gradually with the increasing of the distance. In this paper, we focus on the investigation of the relationship between transmission power and PRR, so the distance between nodes is supposed to be less than or equal to 8m in order to reduce the influence of distance.

Table 4. PRR versus distance

Transmission distance/m	PRR/%
≤ 8	100.0
9	82.1
10	53.6
11	20.1
12	17.8
13	7.5
14	2.7
≥ 15	0.0

Firstly, the simulation experiments are conducted to show the comparison of average PRR as the network running rounds changes from 0 to 1000 for the algorithms NFTC, FTC and FCTP. The results are shown in Fig. 7. We can see NFTC has more stable average PRR than FTC and FCTP because of its adaptive ability of adjusting the transmission power with the desired PRR. FCTP has the lowest average PRR because it pays more attention to the relationship between node degree and transmission power difference.

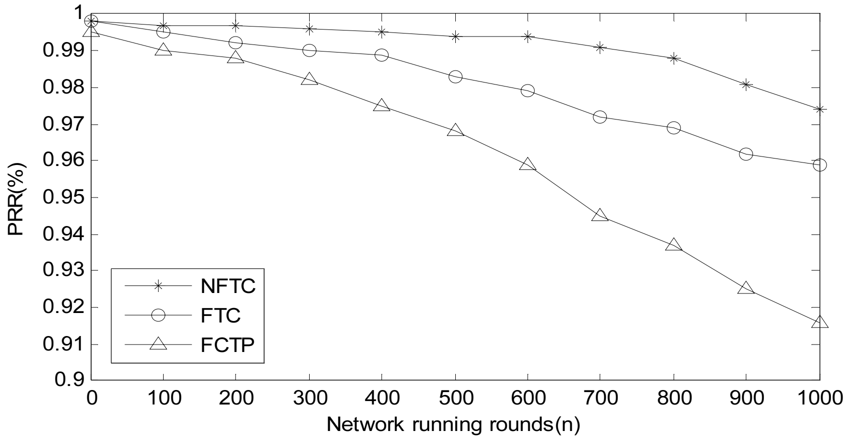

Fig. 7. Average PRR versus the running rounds

Next, the comparison of network survival nodes is presented in Fig. 8. FTC shows the worst performance than others because it adjusts the transmission power of the nodes only considering its node degree. The same problem continues in FTC which ignores the residual energy of the node, but it reduces the computation by constructing the fuzzy logic controller from the training data set. NFTC gives better results than FTC and FCTP since it uses self-learning neural fuzzy controller to adjust the transmission power of the node with its residual energy.

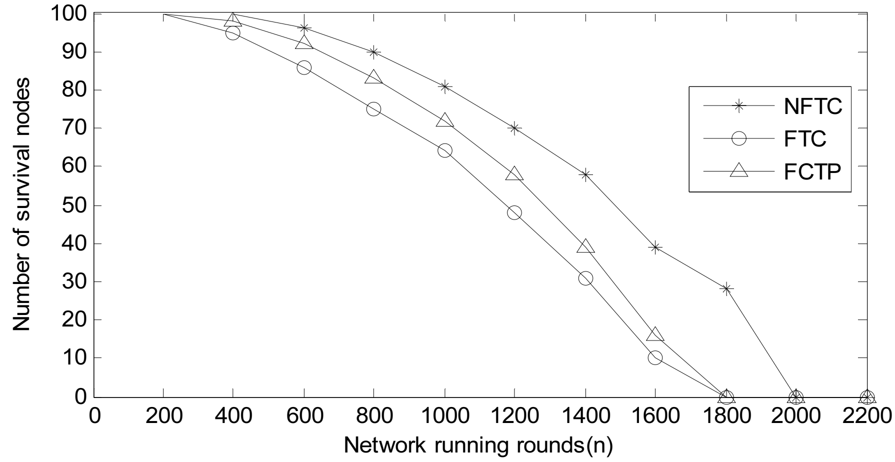

Fig. 8. Network survival nodes versus the running rounds

Afterwards, the total residual energy is measured by using the three algorithms, and the results are depicted in Fig. 9. We can see that NFTC has less fluctuation and longer survival time than FTC and FCTP. This is mainly because NFTC takes into account the residual energy of the node and reduces the computation while adjusting the transmission power. FTC and FCTP focus on maintaining node’s degree, and ignoring the residual energy of nodes. Thus, NFTC achieves the best energy efficiency.

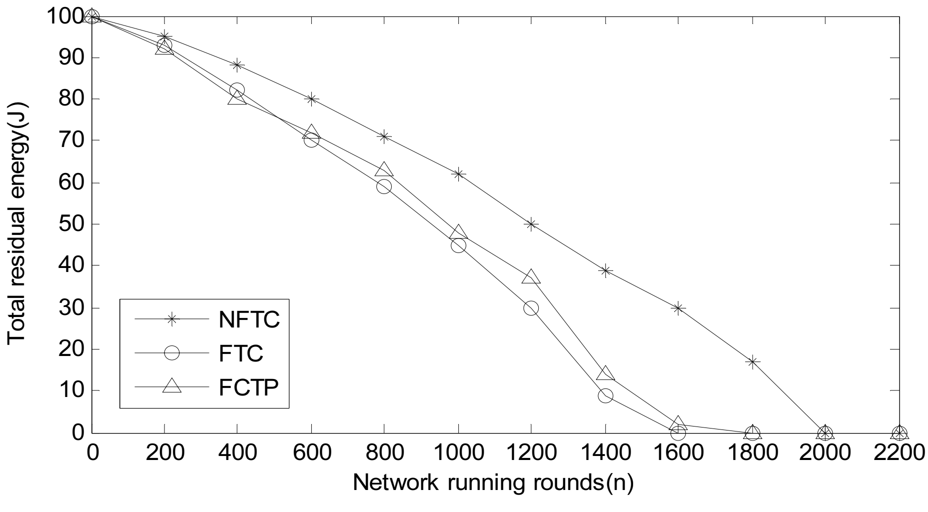

Fig. 9. Total residual energy versus the running rounds

6 Conclusion

A well-designed power control algorithm based on fuzzy logic for WSNs can reduce energy consumption as well as dispose accidental interference and uncertainty of ambiguity among nodes, which in the end prolongs the network lifetime. In this paper, NFTC is proposed to dynamically adjust the transmission power of the nodes using a self-learning neural fuzzy controller which consists of two inference engines. Through the closed-loop feedback processes, the neural fuzzy controller can adapt to the changes of packet reception ratio with respect to the residual energy, accordingly, control the transmission power of the nodes. Simulation results show that NFTC can obtain a better performance of average PRR, total residual energy and network lifetime than FTC and FCTP.

Although NFTC can improve the network performance in some aspects, however, there are still several limitations including lack of actual tests in real networks, and only considering the packet reception ratio and residual energy for power adjustment. So, next, we will perform further tests for NFTC in a real wireless sensor networks, and optimize the neural fuzzy controller by using more inputs such as the amount of delivery data, node degree and packet length.

Acknowledgments

This work was supported by Jilin Provincial Department of science and technology pilot project (20160312002ZG).

References

[1] Z.-Q. Guo, Q. Wang, M.-H. Li, J. He, Fuzzy logic based multidimensional link quality estimation for multi-hop wireless sensor networks, IEEE Sensors Journal 13(10)(2013) 3605-3615.

[2] H.D. Vicente, F.M. Jose, L.M. Nestor, D.T. Raul, Self-adaptive strategy based on fuzzy control systems for improving performance in wireless sensors networks, ACM Transaction on Sensor Networks 15(9)(2015) 24125-24142.

[3] S. Lin, J. Zhang, G. Zhou, L. Gu, T. He, J.A. Stankovic, ATPC: adaptive transmission power control for wireless sensor networks, in: Proc. International Conference on Embedded Networked Sensor Systems, 2006.

[4] L.M. Borges, F.J. Velez, A.S. Lebres, Survey on the characterization and classification of wireless sensor network applications, IEEE Communications Surveys & Tutorials 16(4)(2014) 1860-1890.

[5] N.A. Pantazis, D.D. Vergados, A survey on power control issues in wireless sensor networks, IEEE Communications Surveys & Tutorials 9(4)(2007) 86-107.

[6] C. Huseyin, B. Kemal, T. Bulent, U. Erkam, The impact of transmission power control strategies on lifetime of wireless sensor networks, IEEE Transactions on Computers 63(11)(2014) 2866-2879.

[7] U.Y. Huseyin, T. Bulent, Transmission power control for link-level handshaking in wireless sensor networks, IEEE Sensors Journal 16(2)(2016) 561-576.

[8] M.M.Y. Masood, G. Ahmed, N.M. Khan, Modified on demand transmission power control for wireless sensor networks, in: Proc. IEEE International Conference on Information and Communication Technologies, 2011.

[9] N. Maciej, S. Aariusz, S. Tomasz, W. Bartosz, Transmission power control based on packet reception rate, in: Proc. IEEE 2014 6th International Conference on New Technologies, Mobility and Security, 2014.

[10] K. Seungku, E. Doo-seop, Link-state-estimation-based transmission power control in wireless body area networks, IEEE Journal of Biomedical and Health Information 18(4)(2014) 1294-1302.

[11] J. Zhang, J. Chen, Y. Sun, Transmission power adjustment of wireless sensor networks using fuzzy control, Wireless Communication & Mobile Computing 9(6)(2009) 805-818.

[12] A.A. Ahmed, N. Fisal, Experiment measurements for packet reception rate in wireless underground sensor networks, International Journal of Recent Trends in Engineering 2009(2) 71-75.

[13] M. Mirjana, V. Valdimir, M. Vladimir, Fuzzy logic and wireless sensor networks-A survey, Journal of Intelligent & Fuzzy Systems 27(2)(2014) 877-890.

[14] S. Narayanaswamy, V. Kawadia, R.S. Sreenivas, P.R Kumar, Probability versus possibility and their application to actuator

sensor networks: Power control in Ad Hoc networks: theory, architecture, algorithm and implementation of the COMPOW protocol, Soft Computing 12(5)(2008) 425-434.

[15] P. Santi, D.M. Blough, H. Bostelmann, The critical transmitting range for connectivity in sparse wireless ad hoc networks, IEEE Transactions on Mobile Computing 2(1)(2003) 25-39.

[16] X. Wang, X. Wang, G. Xing, Y. Yao, Minimum transmission power configuration in real-time sensor networks with overlapping channels, ACM Transactions on Sensor Networks 9(2)(2013) 1-28.

[17] Y. Huang, J.-F. Martinez, V.H. Diaz, J. Sendra, A novel topology control approach to maintain the node degree in dynamic wireless sensor networks, Sensors 14(3)(2014) 4672-4688.

[18] P. Chevillat, J. Jelitto, H.L. Truong, Dynamic data rate and transmit power adjustment in IEEE 802.11 wireless LANs, International Journal of Wireless Information Networks 12(3)(2005) 123-145.

[19] R. Srivastava, C.E. Koksal, Energy optimal transmission scheduling in wireless sensor networks, IEEE Transaction on Wireless Communication 9(5)(2010) 1650-1660.

[20] J.V. Alonso, E.E. Lopez, A.M. Sala, Performance evaluation of MAC transmission power control in wireless sensor networks, Computer Networks 51(6)(2007) 1483-1498.

[21] W.B. Pottner, L. Wolf, Probe-based transmission power control for dependable wireless sensor networks, in: Proc. IEEE International Conference on Distributed Computing in Sensor Systems, Computer Society, 2013.

[22] B.H. Son, K.J. Kim, Y.M. Choi, Active-margin transmission power control for wireless sensor networks, International Journal of Distributed Sensor Networks 2014(1)(2014) 1-7.

[23] Y.K. Tamandani, M.U. Bokhari, SEPFL routing protocol based on fuzzy logical control to extend the lifetime and throughout of wireless sensor network, Wireless Networks 22(1)(2016) 1-7.

[24] M.L. Sichitiu, R. Dutta, Benefits of multiple battery levels for the lifetime of large wireless sensor networks, in: Proc. NETWORKING 2005: Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communications Systems, 2005.