SciencePG
Science Publishing Group

A Load Balancing Optimization Algorithm for Context-Aware Wireless Sensor Networks Based on Fuzzy Neural Networks

Wencheng Zuo[1], Hui Xie[2], Yuchi Lin[1], Hui Hu[3], Zhengying Cai[1, *]

[1]College of Computer and Information Technology, China Three Gorges University, Yichang, China

[2]School of Law and Public Administration, China Three Gorges University, Yichang, China

[3]School of Foreign Languages, China Three Gorges University, Yichang, China

Email address:
zuoaichimantou@foxmail.com (Wencheng Zuo), 1446228804@qq.com (Hui Xie), 903238732@qq.com (Yuchi Lin),
381606604@qq.com (Hui Hu), master_cai@163.com (Zhengying Cai)
[*]Corresponding author

Abstract: In wireless sensor networks, the load imbalance will seriously affect the performance of the whole networks, such as local traffic overload, congestion, idle resources and other problems. In this paper, a novel fuzzy neural network algorithm is proposed to solve the problem. First, the problem of load balancing in context-aware wireless sensor networks is analyzed, and the mathematical model is built up. Second, a load balancing optimization algorithm is brought combing neural network and fuzzy theory, and the whole process is also illustrated including learning, association, recognition and information processing. Third, through analyzing and studying a case, a load balancing problem is solved by simulation and comparison to show the potential of the proposed method. Last, some interesting conclusions and future work are indicated at the end of the paper.

Keywords: Context-Aware, Optimization Algorithm, Fuzzy Neural Networks, Load Balancing, Wireless Sensor Networks

1. Introduction

The research on load balancing in wireless sensor networks has never stopped. Han (2016) introduced a traffic load balancing framework for software-defined radio access networks powered by hybrid energy sources [1]. Baranidharan (2016) put forward DUCF, a distributed load balancing unequal clustering in wireless sensor networks using fuzzy approach [2]. Tall (2016) introduced a self-optimizing load balancing with backhaul-constrained radio access networks [3]. Fahimi (2016) implied a joint spectrum load balancing and handoff management in cognitive radio networks by a non-cooperative game approach [4]. Kim (2016) indicated a self-organizing spectrum breathing and user association for load balancing in wireless networks [5]. Glabowski (2016) concerned the modelling load balancing mechanisms in self-optimising 4G mobile networks with elastic and adaptive traffic [6]. Wang (2015) extended the dynamic load balancing with handover in hybrid Li-Fi and Wi-Fi networks [7]. Kim (2016) made an energy-efficient load balancing scheme to extend lifetime in wireless sensor networks [8]. Various frameworks and methods have been proposed to solve the problem of load balancing and its derivations in wireless networks to a certain extent.

By proposing the route optimization mechanism and the design of multicast routing protocol, some progress has been made in how to achieve load balancing. Xing (2016) built a load balancing-based routing optimization mechanism for power communication networks [9]. Yadav (2015) discussed DLBMRP, a design of load balanced multicast routing protocol for wireless mobile Ad-Hoc network [10]. Zhang (2015) built a load balancing in IP networks using generalized destination-based multipath routing [11]. Ren (2016) presented a fault-aware load-balancing routing for 2D-mesh and torus on-chip network topologies [12]. Later, people also

proposed a simple fault tolerance, memory, key value storage, energy sensing and other methods, making the load balance in the power communication network and on-chip network to be expanded. Trajano (2016) developed a two-phase load balancing of in-memory key-value storages using network functions virtualization (NFV) [13]. Xie (2016) evaluated a simple fault-tolerant method to balance load in network-on-chip [14]. Deng (2016) concerned a QoS-aware and load-balance routing for IEEE 802.11s basing neighborhood area network in smart grid [15]. Ricciardi (2016) modeled a hybrid load-balancing and energy-aware RWA algorithm for telecommunication networks [16]. Various methods have been put forward, which provide the study of load balancing with more ways and inspiration.

In cellular networks, the control of load balancing also enables people to make progress in location awareness and power control. Aguilar-Garcia (2016) implied an improving load balancing techniques by location awareness at indoor femtocell networks [17]. Shin (2016) gave a power control for data load balancing with coverage in dynamic Femtocell networks [18]. Farazmand (2016) extended a coalitional game-based relay load balancing and power allocation scheme in decode-and-forward cellular relay networks [19]. Ali (2016) described load balancing in heterogeneous networks based on distributed learning in near-potential games [20].

At the same time, in the context-aware wireless sensor network, the problem of load balancing should be solved based on the global congestion awareness level and the previous discussion on the relevant aspects. Ramakrishna (2016) discussed GCA, a global congestion awareness for

load balance in networks-on-chip [21]. Yan (2016) reviewed an enhanced global congestion awareness (EGCA) for load balance in networks-on-chip [22]. Aguilar-Garcia (2016) offered a context-aware self-optimization evolution based on the use case of load balancing in small-cell networks [23]. Sarma (2016) studied the deciding handover points based on context-aware load balancing in a WiFi-WiMax heterogeneous network environment [24].

In wireless sensor networks, the load imbalance will seriously affect the performance of the algorithm, resulting in local traffic overload, congestion, idle resources and other issues. In order to solve the problem of load balancing in context-aware wireless sensor networks, a fuzzy neural network algorithm is introduced in this paper. And the algorithm is improved by merging the rules weight and capturing the importance decision of a certain rule, and a weight checking method is proposed. Finally, the feasibility and potential of the algorithm are verified by experiments.

2. Context-Aware in Wireless Sensor Networks

In context-aware wireless sensor networks based on fuzzy neural networks, a load balancing optimization algorithm is linked to the travel alternatives features. And the perceptions of input are patterned as uncertain sets for the vagueness, and they can become fuzzy sets indefinitely. The scheme is shown in Fig. 1.
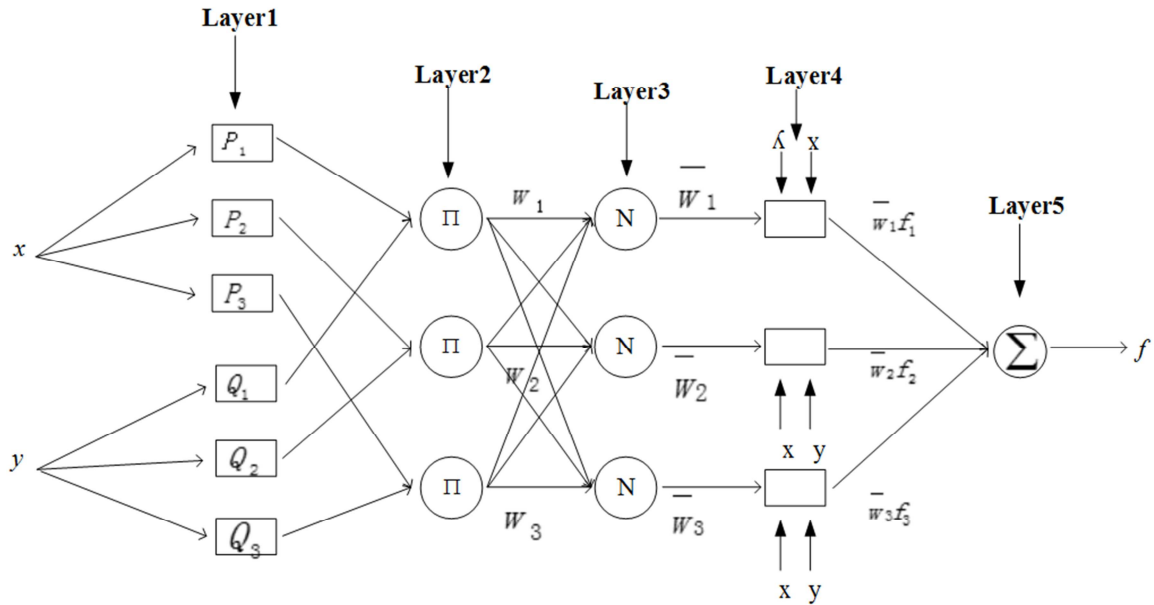

Fig. 1. A context-aware wireless sensor networks based on fuzzy neural networks.

It is supposed that they prefer to selecting simple rules rather than putting the utility function to the most complicated position, as the travels come to a decision. So, rules are applied into modeling decision steps and depicting attitudes on the choice of those shown possible or ambiguous perceptions.

The variations xi, $i = 1, 2, 3, ..., n-1, n$, and yj, $j = 1, 2, 3, ..., m-1, m$, stand features of the set's alternatives of choice and the predilection of alternative j. And sets are uncertain, for example Pi, $i = 1, 2, 3, ..., n-1, n$, and Qj, offer linguistic values in system attribute of ith and of the tastes about alternative j.

Because an element x could not be included in one set only, and between P^* there exists an intersection, the perceptions of input, and many premise rules. And a rule may come to a degree which reflects the likeness at that time, between the personal perceptions and the premise of rule. Setting of a rule k, leads to the uncertain preference Q_j^{*k} with regard to alternative j. The composition mechanism includes the uncertain preferences Q_j^{*k} from all k rules that were invoked and computes all the uncertain preference Q_j^* of alternative j applying the aggregation operator:

$$\sum_k H_{Q_j^{*k}}(y)\rho_k = \sum_k H_{Q_j^{*k}}(y) = H_{Q_j^*}(y) \tag{1}$$

The Given uncertain set Q_j^* stands all the preference of alternative j, and because of a given attributes set, the mechanism of defuzzification is applied into deriving a crisp action (selection). They thought that the gravity center of the set of preference, and centroid$_j$ is deputy of the alternative j's attractiveness. They make a suggestion that two methods should be applied for translating the attractiveness into selection. Probabilistic rule presumes that the gravity center, centroid$_j$, stands the systematic section of the alternative j's utility. So the alternative j's utility for individual n is offered by: $U_{jn} = centoid_{jn} + \theta$, in which θ is a term of error. This model's interpretation is that the gravity center captures all the alternative's attractiveness, when the term randomly captures noise in the behavior of human, losing rules, etc.

An alternative attributes, in a specific situation, may not mate to the rules' premises on the base of rule particularly. The reasoning mechanism permits for correcting the consequence of rules based on the factual input.

In terms of a rule k with multidimensional conditions ρ_k, the losing ability of the rule k is given as the definition

$$q_k = \prod_{i=1,\dots,n} q_{ki} \tag{2}$$

The function of membership or resulting outputs' possibility distribution P^* is linked to the linguistic label Q 's possibility distributions in the consequence of rule and the likeness between input P^* and consistent label P in the rule 's premise. The similarity degree between input P^* and rule premise, which is often called as rule firing, is computed by applying the operator of max–min:

$$q = \max_{x\in X} \min(H_P(x), H_{P^*}(x)) \tag{3}$$

The function of membership of the output Q^* is gained by applying the encoding scheme of correlation-product, and it protects the membership function of set Q's shape in the result of a rule:

$$H_{Q^*}(y) = q.H_Q(y) \tag{4}$$

In which q_{ki} is computed by using Eq (2) to the ith rule premise.

The exertion of the reasoning mechanism decreases the rules' number which is asked on the base of rule because the premises are only typical attributes' labels and they don't have to stand all input values which are possible.

3. Fuzzy Neural Networks Algorithm

3.1. The Algorithm Structure

Fuzzy set theory was brought in by Zadeh (1965) as an ordinary way to show the diverse of inherent uncertainty in human systems. Zadeh (1973) said that men have capabilities to make important and precise statements about a system diminishes' behavior decreases, when this system's complexity increases. He came up with the use of fuzzy sets and approximate reasoning approaches to make model of such systems.

In fig 1, nodes in layer 1 are nodes of input. And in the 6th layer, there are nodes of output. In layer 2, node functions of linguistic values on the variables of input are donated by uncertain sets. In layer 1, each input node is linked to corresponding linguistic values which are in layer 2. In the same way, in layer 5 nodes operate as linguistic values on the nodes in layer 6 which are output of layer 5. In layer 3, nodes stand for rule nodes, and every rule node is linked to the nodes which are in layer 2 and stand the premises of rule. To the nodes which are in layer 5, they stand for its result, via a node which is in layer 4 and dispenses a weight to every rule.

Vague sets are generalization of crisp sets. Those members belong to uncertain sets which are with a possibility degree or membership degree. The membership's grades have values in the interval [0, 1], and stand for the degree by which an element is similar to or consistent to the concept come up by the uncertain set. A uncertain set P is defined on a universe discourse X and can be signified by a pair of sets which is ordered as

$$P = \{(x, H_X^P(x)) \mid x \in X\} \tag{5}$$

In which, $H_X^P(x)$ shows element x's membership to the uncertain set P.

Rule P, original estimates of the rule weights ψ_k and parameter ϕ and γ of the functions' membership are all required for weight correction. Rules are originally presumed to be of same importance. And all ψ_k are made equal to one. Original values about parameter ϕ and parameter γ can be estimated by applying the learning procedure which is self-organized.

An optional way for the decision of the rule base, is given as appropriate estimations of the values about the parameters ϕ and γ. The uncertain rule generator is a repetitive process that pursues to recognize the optimum result of a given rule k. Therefore some measure of effectiveness can be chosen. In the following description, all the number of choices is maximized and the choices are accurately predicted. Each rule k is tested in turn and its optimal output is recognized by testing all possible combinations.

3.2. Solving Step

And the model inputs P^* stand for the perceptions of individuals. And they are matched with each rule premises k. An inference scheme is called approximate reasoning. And it is then applied to deduce the resulting consequence Q_k^*, assuming the perception is P^*. Rules are come up at the same time. And a composition mechanism counts the implications B_k^* to a fuzzy preference Q^*, with respect to its membership function. Final crisp choice arises from the defuzzification of preference Q^*.

The diverse elements will be discussed below and which is about the uncertain decision making mechanism. It is presumed that the fuzzy membership functions applied in making the rules are come up by bell shaped functions which are defined by breadth (standard deviation) γ, their center (mean) ϕ and have the form

$$H(x) = \exp\left(-\frac{(x-\phi)^2}{\gamma^2}\right) \tag{6}$$

The framework can be enlarged to illustrate rule weights which make model of these trade-offs. The trade-offs amongst the different rules could be gained by introducing rule weights ψ_k, in the collection operation that is applied to gain the fuzzy preference Q_j^* which is in the composition mechanism:

$$H_{Q_j^*}(y) = \sum_k \psi_k \rho_k H_{Q_j^{*k}}(y) = \sum_k \psi_k H_{Q_j^{*k}}(y) \tag{7}$$

Fig. 2 shows an example of the fuzzy membership function above to stand the next linguistic variables in neural networks: Very Low (VL), Low (L), Medium (M), High (H) and Very High (VH). For instance, a 32 minutes of network delay is belonging to the uncertain set "medium travel time" which is with membership 0.7 and to the uncertain set "high travel time" which is with membership 0.4. The correction method went in two steps: initialization and calculation.
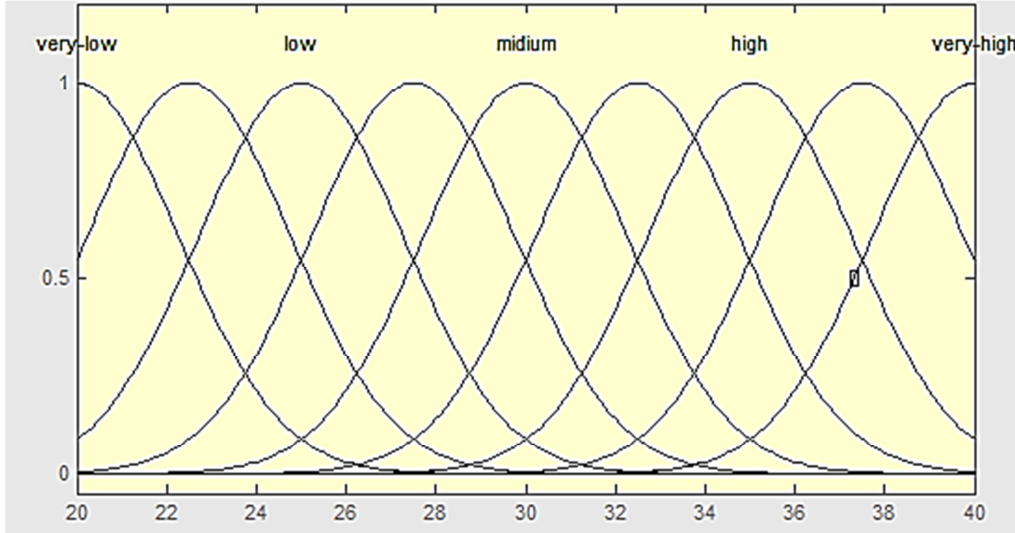

Fig. 2. Example of membership functions of possible linguistic values of travel time.

3.2.1. Initialization

The optimization procedure steps are shown below, in which $cnq(j)$ indicates the possible result of a rule.

It should be indicated that, generally, the aim of making the most use of the correct predictions' number might not be very suitable and may cause unstable models. Just small correction to the inputs may lead to diverse outcomes.

3.2.2. Calculation

The procedure calibration is applied to identify the best values of the membership function parameters and rules weights, so the deviation between model outputs and observations is minimized. Following discussion extends their method so it can correct both the membership functions, parameters and the rule weights at the same time.

A very minute description of each node function includes variables ipt_i^k and otp_i^k for description about the inputting

to a node and outputting from a node. Superscript k indicates the network layer. And subscript i indicates the setting of the node in the layer. The discussion presumes in multidimensional conditions and results.

3.2.3. The Neural Network Representation

The 1th layer in fig 1 is a layer of input variations. First layer acts as a data handler, and it is used to transfer its inputting to the similar nodes in the 2th layer.

$$otp_i^1 = ipt_i^1 \quad ipt_i^1 = x_i^* \tag{8}$$

The 2th Layer is input linguistic value node layer. The second layer calculates the membership of the inputting x_i^* to the fuzzy sets P_i.

$$otp_{i,j}^2 = \exp(ipt_{i,j}^2) \tag{9}$$

$$\text{ipt}^2_{i,j} = -\frac{(\text{otp}^1_i - \phi^{\text{ipt}}_{i,j})^2}{(\gamma^{\text{ipt}}_{i,j})^2} \qquad (10)$$

In which the subscript (i, j) means the jth linguistic value about the ith input. Assuming the node (i, j) is in the 2th layer, $\phi^{\text{ipt}}_{i,j}$ and $\gamma^{\text{ipt}}_{i,j}$ are the parameters of the jth value of the ith input variable.

In this layer, the input membership functions parameters are dateless

$$\frac{\partial E}{\partial \phi^{\text{ipt}}_{i,j}} = \frac{\partial \text{otp}^2_{i,j}}{\partial \text{ipt}^2_{i,j}} \cdot \frac{\partial \text{ipt}^2_{i,j}}{\partial \phi^{\text{ipt}}_{i,j}} \cdot \frac{\partial E}{\partial \text{otp}^2_{i,j}} \qquad (11)$$

In which

$$\frac{\partial \text{ipt}^2_{i,j}}{\partial \gamma^{\text{ipt}}_{i,j}} = \frac{2(\text{otp}^1_i - \phi^{\text{ipt}}_{i,j})^2}{(\gamma^{\text{ipt}}_{i,j})^3}$$

$$\frac{\partial E}{\partial \gamma^{\text{ipt}}_{i,j}} = \frac{\partial E}{\partial \text{otp}^2_{i,j}} \cdot \frac{\partial \text{ipt}^2_{i,j}}{\partial \gamma^{\text{ipt}}_{i,j}} \cdot \frac{\partial \text{otp}^2_{i,j}}{\partial \text{ipt}^2_{i,j}}$$

$$\frac{\partial \text{ipt}^2_{i,j}}{\partial \phi^{\text{ipt}}_{i,j}} = \frac{2(\text{otp}^1_i - \phi^{\text{ipt}}_{i,j})^2}{(\gamma^{\text{ipt}}_{i,j})^3}$$

$$\frac{\partial \text{otp}^2_{i,j}}{\partial \text{ipt}^2_{i,j}} = \exp(\text{ipt}^2_{i,j})$$

$$\frac{\partial \text{ipt}^3_k}{\partial \text{otp}^2_{i,j}} = \prod_{(\phi,v)\in\Gamma_k,(\phi,v)\neq(i,j)} \text{otp}^2_{\phi,v} = \frac{\text{ipt}^3_k}{\text{otp}^2_{i,j}}$$

$$\frac{\partial E}{\partial \text{otp}^2_{i,j}} = \sum_{k\in\Gamma_{(i,j)}} \delta^k_3 \cdot \frac{\partial \text{ipt}^3_k}{\partial \text{otp}^2_{i,j}}$$

In the 3th layer, every node can compute firing strength of every rule in the 3th layer.

$$\text{otp}^3_k = \text{ipt}^3_k \quad \text{ipt}^3_k = \prod_{(i,j)\in\Gamma_k} \text{otp}^2_{(i,j)} \qquad (12)$$

In which Γ_k is the gathering of nodes (i, j). And the second layer stands for linguistic values which is used in the previous part of rule k.

In this layer, there exists no appropriate parameter. The error which spreads to the preceding layer is counted as

$$\xi^3_{k3} = \xi^4_k \cdot \psi_k, \quad \frac{\partial \text{otp}^3_k}{\partial \text{ipt}^3_k} = 1 \quad \text{and} \frac{\partial \text{ipt}^4_k}{\partial \text{otp}^3_k} = \psi_k \qquad (13)$$

Where

$$\xi^3_k = \frac{\partial \text{ipt}^4_k}{\partial \text{otp}^3_k} \cdot \frac{\partial \text{otp}^3_k}{\partial \text{ipt}^3_k} \cdot \xi^4_k$$

3.2.4. Training the Neuro-Fuzzy Decision Model

Assuming the transformation above, the model can be corrected by applying regular skills from the nervous network. The nodes in the conventional NN structures accept inputting from nodes at the procedure layer and often apply a logic activation role to count their outputting. The inputting and outputting functions applied in the uncertain determination making representation, which are different from those applied in traditional NN models. The resultant connectionist model offers a handy framework for both showing the concurrent results and working of rules. It describes the devised selection model, and training (applying an introduction of the general delta rule). In order to train the network, a fault function needs to be spelled out, such as

$$E = \frac{1}{2}\sum_{n=1}^{N}\sum_{i=1}^{m}(y_i(n) - y^*_i(n))^2 \qquad (14)$$

In which $y^*_i(n)$ is the model outputting for inspection n, where n is the observations number, and m is the model outputs number.

Therefore, the aim in training stage is to recognize values of ψ_k, ϕ and γ which curtail the fault E. During training, there is difference between model outputting and observations and the difference is computed and the resulting fault is spread from the outputting layer to the hidden, until it gets to the inputting layer. The learning algorithmic rule renews the parameter values, ψ_k, ϕ and γ by the formulation.

$$\psi_k(i+1) = \psi_k(i) + \eta(-\frac{\partial E}{\partial \psi_k})$$

$$\phi_{i,j}(i+1) = \phi_{i,j}(i) + \eta(-\frac{\partial E}{\partial \phi_{i,j}}) \qquad (15)$$

$$\gamma_{i,j}(i+1) = \gamma_{i,j}(i) + \eta(-\frac{\partial E}{\partial \gamma_{i,j}})$$

In which i indicates the training repetition and η is the rate of learning.

3.2.5. Evolution

The standardization framework shown above can be summarized to consist of correction which is applied to stand the inputting perceptions too. But, a severe problem in the use of nervous networks to complicated problems is linked to the network's size, i.e., to the amount of the network parameters which will be identified. A lot of parameters may lead to network memorizing but not studying. Network record shows that the network fit to the data well, may not really seize the mechanism that depicts the system's behavior that produces the data set. In consequence, even though the model presents a proper performance when is utilized to train data set, and does not perform well certainly with other data sets.

4. Numerical Experiment

4.1. Problem Description

Here a numerical example was exploited to validate the proposed model. In this case, wireless sensor networks are used in the information collection and control of greenhouse environment. Hence a single greenhouse can become a wireless sensor network measurement and control area, and the wireless networks composed of different sensor nodes take advantages over a simple sensor in the measurement of soil moisture, soil composition, pH value, precipitation, temperature, air humidity, air pressure, light intensity, and CO_2 concentration. And actuator fan, motor, valves and other low voltage low current, and the biological information acquisition method are all applied to the wireless sensor nodes, providing us with a scientific basis for the precise control of greenhouse. Additionally, the sensor, the greenhouse standardization, the implementation data, and the gateway can work together to achieve an integrated control for different network devices.

$$V_{car} = \varsigma_{car} + \varsigma_{\Delta c}\Delta c + \varsigma_{Acc\_C}Acc\_C + \varsigma_{cen\_C}centr\_C + \varsigma_{\Delta t}\Delta t$$

Grounded on the designs above, the rule base includes a sum of twenty rules. Some rules represent the differentia in time of travel between network notes, some rules represent the

differentia in expense between the two ways, other rules represents the time of network access.

So, the all number of parameters to be corrected consists of fifteen rule weights, fifteen pairs of parameters and of the membership functions which represents the uncertain sets that is applied in the premises of rule and the membership functions that represents the labels which is used in the rule result and ten more pairs of parameters. Assumed that the specimen includes only 355 observations, so as to prevent overfitting/memorizing, solely the twenty rule weights were corrected by applying the arranged framework.

4.2. Results and Analysis

The parameters' values about the linguistic labels were decided all alone. The parameters ϕ and γ of the rule premises' labels were determined by dividing the inputting and outputting spaces into a prearranged number of regions and meditating ϕ to be each region's center and γ to attach importance that permits a wise overlapping between continual fuzzy sets. Fig. 2 tally up the fuzzy sets above which are used in the test, while

The linguistic labels were linked, and they seemed on a range from 0 to 1. 0 shows no preference and 1 intense preference. Fig. 3 shows the load parameters of membership functions for rule premises (attributes).
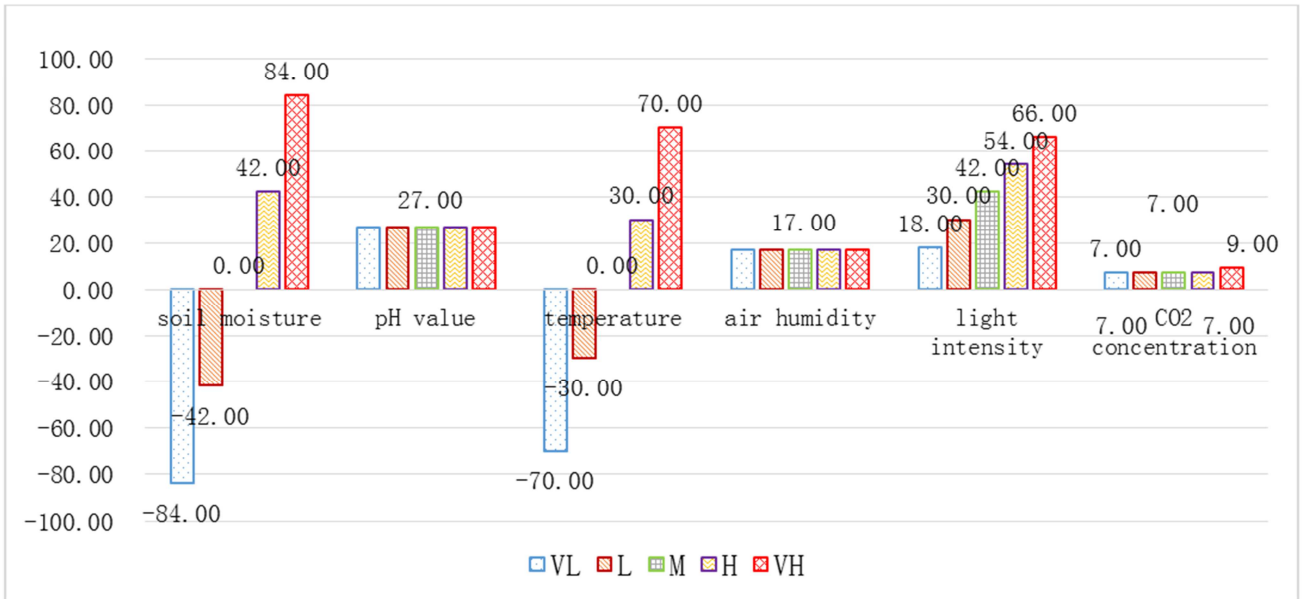

Fig. 3. Load parameters of membership functions for rule premises (attributes).

The performance of uncertain decision making model was illustrated as the one that was with the conceivable choice rule and optimal weights. A dualistic was reckoned with the under after requirement. And it was contrasted to a utility maximization model which was based on the formulation above.

$$V_{car} = q_{car} + \varsigma_{\Delta t}\Delta t + \varsigma_{Acc\_C}Acc\_C + \varsigma_{\Delta c}\Delta c$$

All the variations are defined earlier. TRN stands the transfers' number for the option of networking. Fig. 3 sums up the estimation's results.

The load performance, with respect to accurate predictions' number, is somewhat worse than the fuzzy-based model's performance. The load foresaw 222 observations contrasting to the 264 that was rightly with optimum weights, as shown in fig. 4, with soil moisture, pH value, temperature, air humidity, light intensity, and CO_2 concentration.
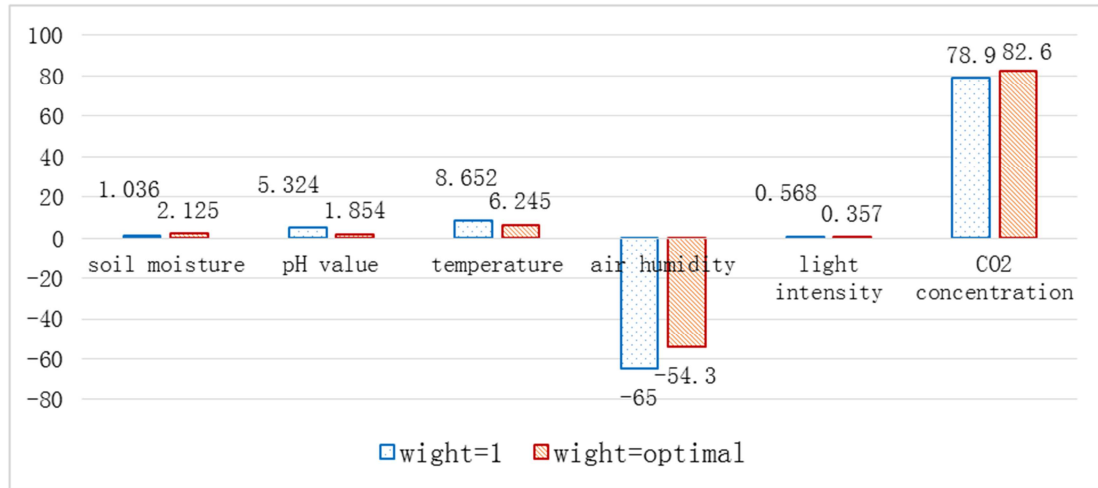
Fig. 4. Load statistics that is in parenthesis L(0)=62.9.

Fig. 5 sums up the Load statistics that is in parenthesis L(0)=62.9 from different approaches to the proposed model and contrasts it to the consistent consequence which is from the uncertain model with best weights. The consequences are in favor of the conclusion above. Moreover, with respect to the personal estimation consequences (evaluation at each repetition of the approach of cross-validation and leaving one),

the uncertain model shows a more stable behavior.

$$V_{rail} = \varsigma_{Acc\_R} Acc\_R + \varsigma_{TRN} TRN$$

In the Fig. 5, fan represents the fan motor, water pump can provides water for green house, and light for illumination.
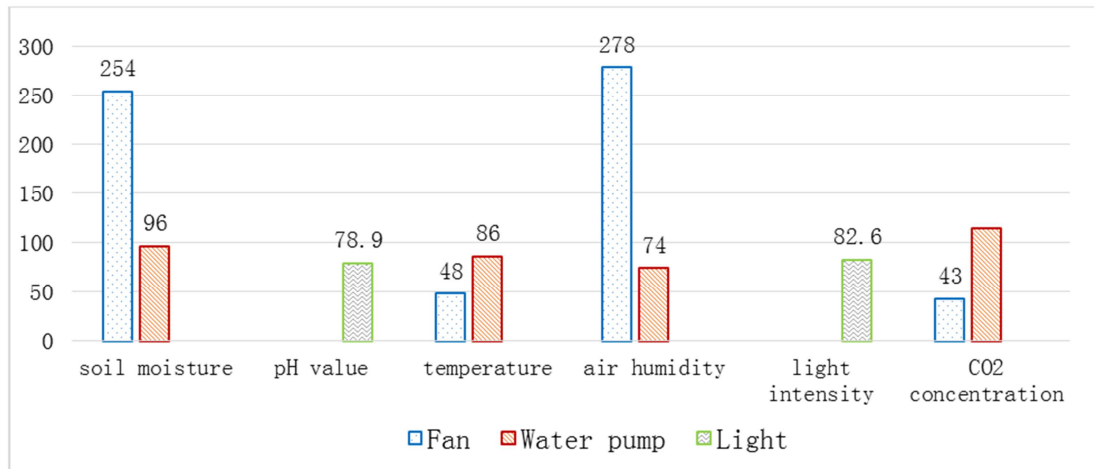

Fig. 5. The approach results of cross-validation and leaving one.

The application of proposed model as a way of good of fitting could be deceptive. The Fig .6 is grounded on the application of the intended users number, $E(N_{ij})$, who truly employed alternative i but was dispensed by the model to j alternative, as shown in Fig. 5.

If the center value really caught the under selection mechanism and individuals preferences, then the individual attributes' contribution should be tiny. So, the hypothesis is verified by $\varsigma_{car} = \varsigma_{\Delta t} = \varsigma_{Acc\_R} = \varsigma_{\Delta c} = \varsigma_{Acc\_C} = \varsigma_{TRN} = 0$. In which $P_n(j)$ is the possibility that individual n picks the alternative jth, and yin =1 if i is chosen else yin =0.

Fig.6 below sums up the consequence applying this performance measure.

$$V_{rail} = \varsigma_{Acc\_R} Acc\_R + \varsigma_{TRN} TRN$$

Now to pay our attentiveness to the explanation of the center of the preference set to be as a method of the all attractiveness of the consistent alternative. In order to verify this hypothesis, the following test will be performed. A logic model is corrected. Systematic utilities of the model consist of all the attributes which is applied above with the adding of the alternative's corresponding center.

The Fig. 6 shows the learning errors of fuzzy neural networks, where fan represents the fan motor, water pump can provides water for green house, and light for illumination. The uncertain model which is with the optimal weights displays better than other models still.
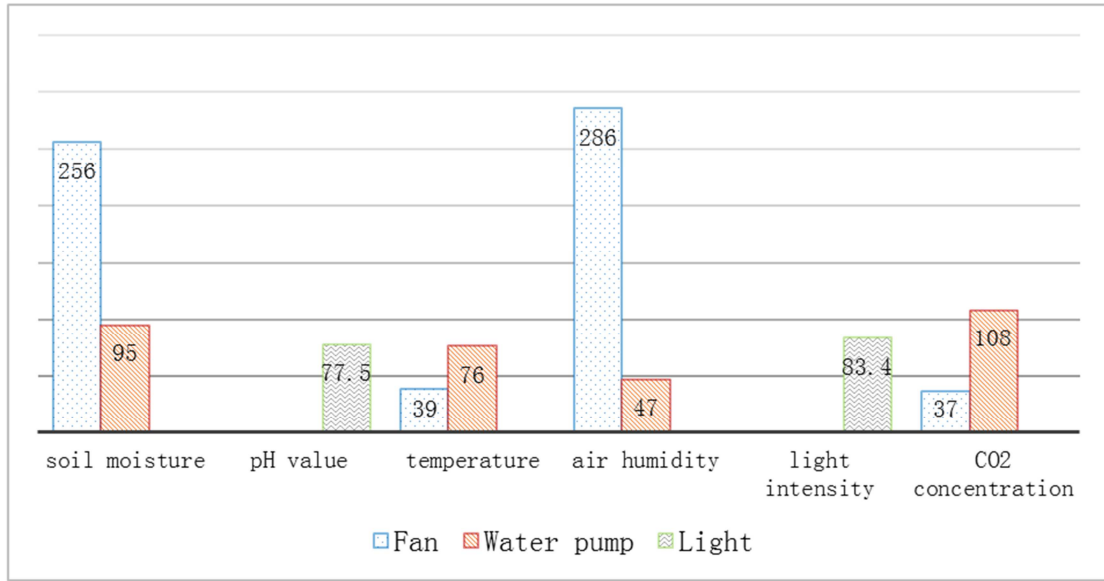
Fig. 6. *Learning errors of fuzzy neural networks.*

In order to verify the hypothesis, the statistic $-2(L(\varsigma_R) - L(\varsigma_u))$ is applied. b_R are the reckoned modulus of the finite model. The finite model links to the possible choice rule with consistent center. And ς_u is the reckoned module of the unbounded neural networks with the complete set of parameters or attributes.

Fig. 7 sums up the comparison of different neural networks. The data above is v^2 distributed with $K = K_u - K_R$, in which K_R and K_u are the freedom of parameters in the model. The crucial value is 263, and the 92% level for six degrees is freedom. Therefore, the invalid hypothesis is not refused at the 92%.
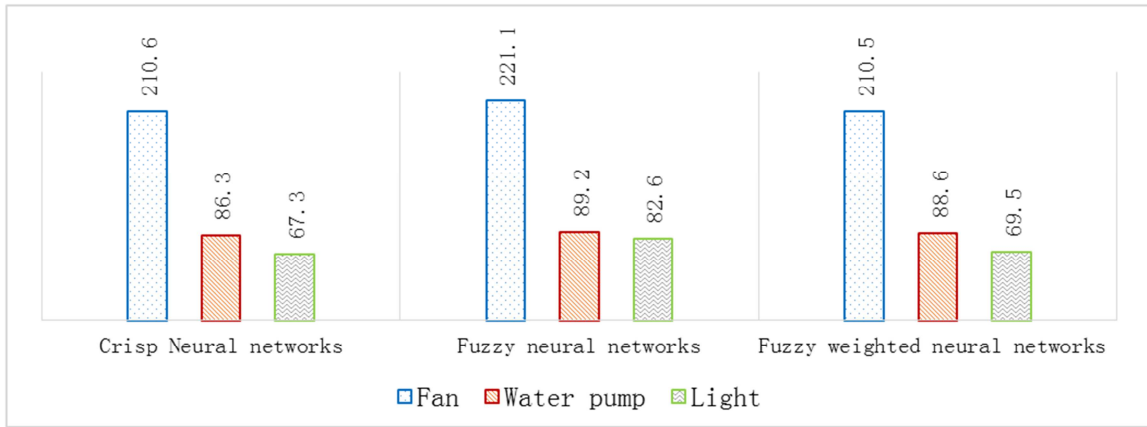

Fig. 7. *Comparison of different neural networks.*

It is clear that the statistical tests' results are in favor of the explanation that the preference set center is a measure all attractiveness of the consistent alternative (in the condition of optimum weights). But the conclusions above are not right if the uncertain model with weights alike to one is applied. In this condition, the likelihood in the extensive model, is 221.1. The result of fuzzy weighted neural networks is 210.5 better than critical ϑ^2 of 210.6 in crisp neural networks. And the invalid hypothesis is dumped for optimal scheme of fuzzy weighted neural networks.

4.3. Further Discussion

In this part, several model's performance are compared

which is developed in section 3, gaining conclusions according to the weights importance, assessing the conclusive choice rules and probabilistic impact, and making a comparison between the proposed model and traditional choice model in reference 8. The comparison of learning errors by applying the procedure is described in Fig.8.

Fig. 8 sums up the consequences from the use of the model applying weights similar to one and best weights separately. In both conditions, the rule base is unchanged. The matrix of rule is performed better in the condition of optimum weights. The model is with rule weights similar to one showing poorly in terms of the network alternative.
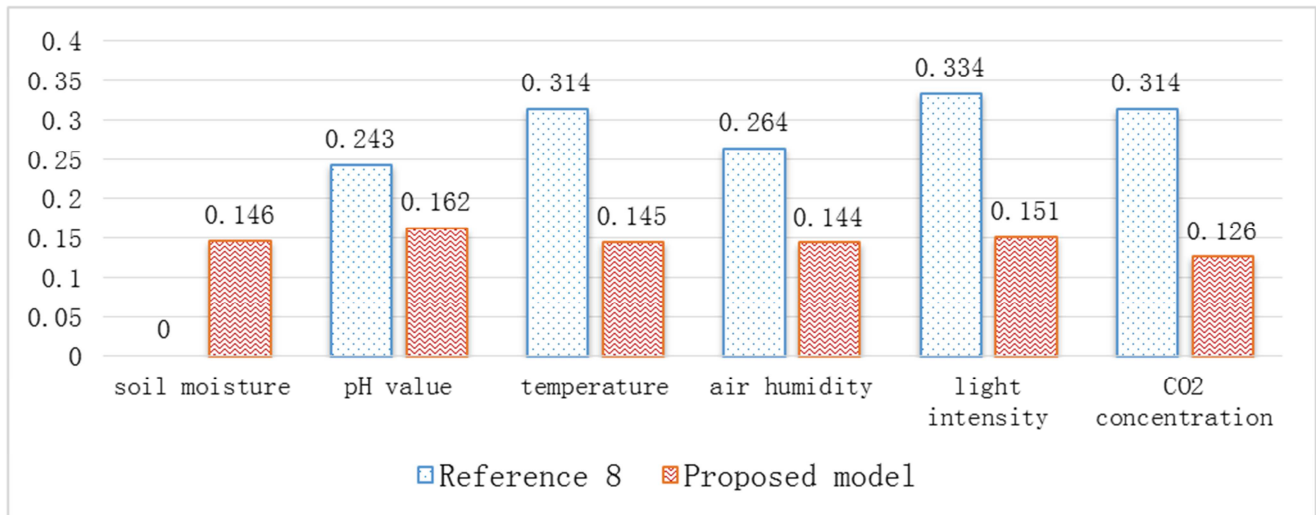
Fig. 8. *Comparison of learning errors.*

The connected total mistake is high, too. The model is with the corrected rule weights increasing its performance by paying more attention to the network mode. The rightly predicted network choices raised from 14.2% in the condition of weights identical to one to 61.1% with optimum weights. These consequences clearly show the significance of allotting weights to the rules. Furthermore, the error value before and after correction, shows the lack of applying the number of proper prophecy as a standard in the generation of rule.

The probabilistic choice rule is also enlarged by making definition of the utility, U_{ipt}, of individual n and an alternative i as

$$U_{ipt} = \varsigma_{centr\_i} centroid_i + \varsigma_0 + \theta_{ipt}$$

The motive for this method is the understanding that both type of the fuzzy fallibility of the behavior of human in modeling choice behavior must be taken into consideration. The first type is linked to ambiguity in the awareness and treating of information that is human reasoning feature. In the model shown here this type of ambiguity is seized by the uncertain framework. The second type of ambiguity is linked to arbitrariness in action and the tastes changeability and individuals' experiences and is gained by the fault term.

In the Fig. 9, the max loads of several references 2, 8, 16 and proposed model are compared in Fig 9 form soil moisture, pH value, temperature, air humidity, light intensity, and CO_2 concentration.
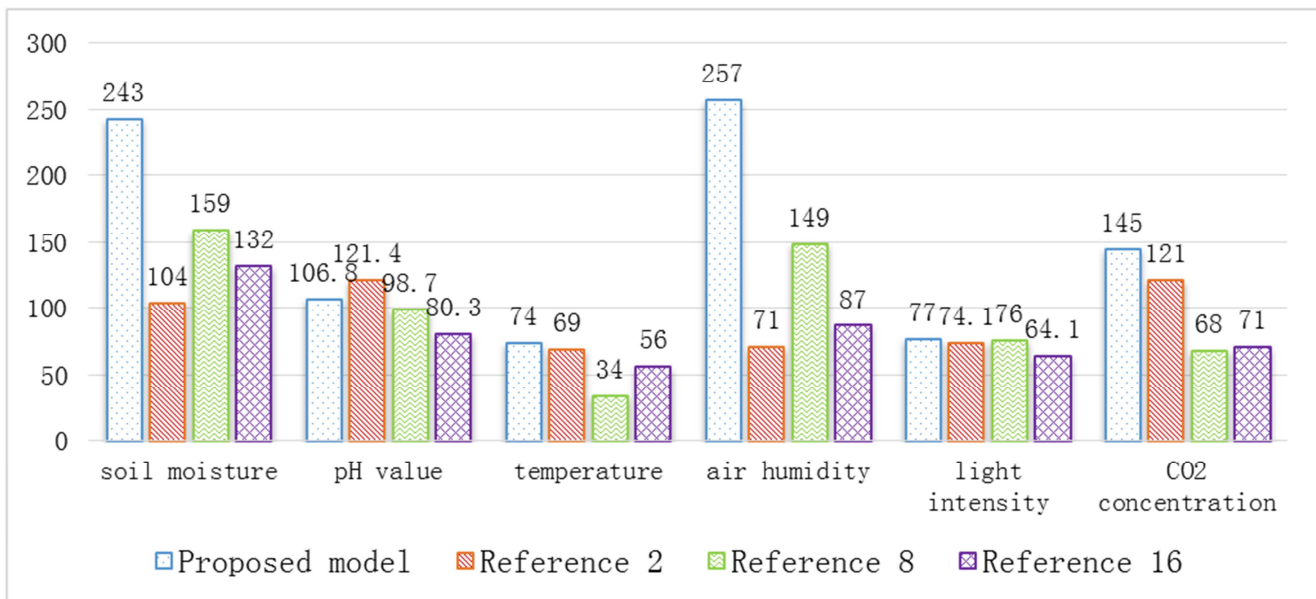

Fig. 9. *Max load comparison of different models.*

Apparently the proposed model gets the most max load capability in all 4 models. The reckoned models for the center of mass homologous to weights identical to one and optimum weights help to sum up load decision. To assess and make a comparison of the load performance, it is preferable to apply one of the sample subset into the design set, and take the remainder as the test set.

For practice, the proposed approach, for a specimen size of observation N, is used by testing the model N times. And each time a specimen of N -1 observation applies. The ith observation did not consist of the specimen, and the tested model is then applied to train the neural networks to make the load balance consistent to the ith observation or the ith estimation. The goodness of fit is counted from the consequences of the training process of the neural networks, and on the observation, all sensor data was not applied for its test.

Additionally, the consequence from the use of the proposed approach verifies the conclusions come up earlier. The model which is with the optimum weights is steadier and shows better load balancing performance, particularly, in terms of the alternative with the smaller share of load capability. Moreover, probabilistic choice rule is better than the deterministic one, from both an ideological and a performance viewpoint.

5. Conclusion

The article extended a load balancing optimization algorithm for context-aware wireless sensor networks based on fuzzy neural networks. The model's performance was verified by a study of a mode choice case. The consequence of the case study is in favor of the presented hypotheses. But the consequence of the utilization shown in this article illustrates that the approach has the potential to capture stable state and long steady behavior. The approach application to actual matters is ungainly at present and many problems have to be resolved, though the consequences are bright and approximate reasoning framework has demonstrated excellent flexibility in carrying on the decision process. Moreover, perform evaluation and hypothesis testing lack systematic methods, which is in need of further investigation.

In future work, the amount of parameters to be regulated is larger and larger, and it will increase with the amount of alternatives. This may result in over fit, except that proper data sets are available. Additionally, more fuzzy operators' behavioral interpretation will be used in the composition stage and approximate reasoning, more complete model and approach will be chosen, and the comparison of membership functions and different rules will be made for deeper study.

Acknowledgements

References

[1] Han, Tao; Ansari, Nirwan. A Traffic load balancing framework for software-defined radio access networks powered by hybrid energy sources, IEEE-ACM Transactions on Networking, 24 (2016) 1038-1051.

[2] Baranidharan, B.; Santhi, B.; DUCF: Distributed load balancing Unequal Clustering in wireless sensor networks using Fuzzy approach, Applied Soft Computing, 40 (2016) 495-506.

[3] Tall, Abdoulaye; Altman, Zwi; Altman, Eitan; Self-optimizing load balancing with backhaul-constrained radio access networks, IEEE Wireless Communications Letters, 4 (2015) 645-648.

[4] Fahimi, Mina; Ghasemi, Abdorasoul; Joint spectrum load balancing and handoff management in cognitive radio networks: a non-cooperative game approach, Wireless Networks, 22 (2016) 1161-1180.

[5] Kim, Hyea Youn; Kim, Hongseok; Cho, Yun Hee; Lee, Seung-Hwan; Self-organizing spectrum breathing and user association for load balancing in wireless networks, IEEE Transactions on Wireless Communications. 15 (2016) 3409-3421.

[6] Glabowski, Mariusz; Hanczewski, Slawomir; Stasiak, Maciej; Modelling load balancing mechanisms in self-optimising 4G mobile networks with elastic and adaptive traffic, IEICE Transactions on Communications, E99B (2016) 1718-1726.

[7] Wang, Yunlu; Haas, Harald; Dynamic Load balancing with handover in hybrid li-fi and Wi-Fi networks, Journal of Lightwave Technology, 33 (2015) 4671-4682.

[8] Kim, Hye-Young; An energy-efficient load balancing scheme to extend lifetime in wireless sensor networks, Cluster Computing-the Journal of Networks Software Tools And Applications, 19 (2016) 279-283.

[9] Xing, Ningzhe; Xu, Siya; Zhang, Sidong; Guo, Shaoyong, Load balancing-based routing optimization mechanism for power communication networks, China Communications, 13 (2016) 169-176.

[10] Yadav, Ajay Kumar; Tripathi, Sachin. DLBMRP: Design of load balanced multicast routing protocol for wireless mobile Ad-Hoc network, Wireless Personal Communications, 85 (2015) 1815-1829.

[11] Zhang, Junjie; Xi, Kang; Chao, H. Jonathan; Load balancing in IP networks using generalized destination-based multipath routing, IEEE-ACM Transactions on Networking, 23 (2105) 1959-1969.

[12] Ren, Pengju; Kinsy, Michel A.; Zheng, Nanning; Fault-aware load-balancing routing for 2D-mesh and torus on-chip network topologies, IEEE Transactions on Computers, 65 (2016) 873-887.

[13] Trajano, Alex F. R.; Fernandez, Marcial P.; Two-phase load balancing of in-memory key-value storages using network functions virtualization (NFV), Journal of Network And Computer Applications, 69 (2016) 1-13.

[14] Xie, Ruilian; Cai, Jueping; Xin, Xin; Simple fault-tolerant method to balance load in network-on-chip, Electronics Letters, 52 (2016) 1145-1159.

[15] Deng, Xiaoheng, He, Lifang, Zhu, Congxu, Dong, Mianxiong, Ota, Kaoru, Cai, Lin, QoS-aware and load-balance routing for IEEE 802.11s based neighborhood area network in smart grid, Wireless Personal Communications, 89 (2016) 1065-1088.

[16] Ricciardi, Sergio; Sembroiz-Ausejo, David; Palmieri, Francesco; Santos-Boada, German; Perello, Jordi; Careglio, Davide; A hybrid load-balancing and energy-aware RWA algorithm for telecommunication networks, Computer Communications, 77 (2016) 85-99.

[17] Aguilar-Garcia, A.; Fortes, S.; Garrido, A.; Fernandez-Duran, A.; Barco, R.; Improving load balancing techniques by location awareness at indoor femtocell networks, Eurasip Journal on Wireless Communications And Networking, Improving load balancing techniques by location awareness at indoor femtocell networks, (2016).

[18] Shin, Donghoon; Choi, Sunghee; Power control for data load balancing with coverage in dynamic femtocell networks, Wireless Networks, 22 (2016) 1145-1159.

[19] Farazmand, Yalda; Alfa, Attahiru S.; A coalitional game-based relay load balancing and power allocation scheme in decode-and-forward cellular relay networks, Wireless Communications & Mobile Computing, 16 (2016) 1124-1134.

[20] Ali, Mohd. Shabbir, Coucheney, Pierre, Coupechoux, Marceau. Load balancing in heterogeneous networks based on distributed learning in near-potential games, IEEE Transactions on Wireless Communications, 15 (2016).

[21] Ramakrishna, Mukund, Kodati, Vamsi Krishna, Gratz, Paul V., Sprintson, Alexander, GCA: Global congestion awareness for load balance in networks-on-chip, IEEE Transactions on Parallel And Distributed Systems 27 (2016) 2022-2035.

[22] Yan, Jili; Enhanced global congestion awareness (EGCA) for load balance in networks-on-chip, Journal of Supercomputing, 72 (2016) 567-587.

[23] Aguilar-Garcia, Alejandro; Fortes, Sergio; Fernandez Duran, Alfonso; Barco, Raquel; Context-aware self- optimization evolution based on the use case of load balancing in small-cell networks, IEEE Vehicular Technology Magazine, 11 (2016) 86-95.

[24] Sarma, Abhijit; Chakraborty, Sandip; Nandi, Sukumar; Deciding handover points based on context-aware load balancing in a wifi-wimax heterogeneous network environment; IEEE Transactions on Vehicular Technology, 65 (2016) 348-357.