Control of Congestion Effects in Wireless Sensor Network with Mobile Sink Node Based on Wavelet Neural Network

Asst. Prof. Dr. Nadia Adnan Shiltagh Al-Jamali

University of Baghdad /Computer Engineering

Baghdad, Iraq

Email:dr-nadiaat@hotmail.com

Nariman Riyadh Mohammed Ali Al-Tahan

University of Baghdad /Computer Engineering

Baghdad, Iraq

Email: nariman.riyadh@yahoo.com

Abstract:-

In this paper, the Wireless Sensor Network with random mobility sink node is introduced in order to improve the Quality of Service (QoS) based on intelligent controller. This controller is designed based on (Wavelet-Neural Network). The clustering technique is used in the WSN, to enhance the routing algorithm. To quantify the level of performance of the algorithm proposed in this paper, Network Simulator (NS-2) is used which is installed on Ubuntu 14.04. Simulation results show that the proposed algorithm can effectively reduce congestion and enhance the QoS by achieve better throughput, better packet delivery ratio, less End-To-End delay and Packet-loss. The Comparison of the performance of the network with No Congestion Controller (NCC), Neural Network Congestion Controller (NNCC) and Wavelet Neural Network Congestion Controller (WNNCC) is introduced. It is found that the performance with WNNCC is better than performance with NCC and NNCC. The study of the WNNCC and NNCC is done. The resultant average throughput, End-To-End Delay, count receive packet, count Packet-Loss, and Packet Delivery Ratio are compared between WNNCC and NNCC. The simulation of BP algorithm is performed using MATLAB R2014a software running the computer with the following specifications: windows 7 (32-bit), core i5, RAM 4 GB.

Keywords: Wireless Sensor Network, mobile sink node, congestion control

I. Introduction

Wireless sensor networks (WSNs) are a special category of networks which is a collection of a small size and low-cost sensor units called sensor nodes. They can be placed in areas of various geographical and environmental situations to establish a specific practical operational task by continuously measuring various parameters such as pollution, temperature, humidity, visibility and a wide range of environmental and other kinds of data. WSN's wide and varied application has deemed it an important and active research area. Its applications range from environmental, industrial, social, to security and military areas [1].

Mobile sink node is utilized to reduce the number of communication hops between sensor nodes and the sink. Mobile sink style is an effective approach to collect data from wireless sensor networks in an energy effective mode. In mobile sink style, the sink or data collector has mobility and it moves towards the sensor nodes to collect the sensed data from it. In this approach, the mobile sink will move between each cluster to collect information from the cluster heads nodes [18].

The number of mobile sinks essential to gather data from wireless sensor networks with a particular number of sensor nodes has an essential role in data gathering. If the number of mobile sinks is greater than the desired count, there will be reserve underutilization. Since sensor nodes are costly devices this is not a cost effective approach. At the same time, if we use less number of mobile sinks than the desired level, it will be energy as well as time-consuming approach [18].

Data traffic in WSNs has special characteristics that set it apart from other conventional networks. The main traffic has a many-to-one pattern. Most data are moving upwards from the sensor nodes to the base station. This kind of traffic can be continuous, event-driven, query-driven or a mixed of the three types. Each of these types has its own requirements for Quality of Service (QoS) and reliability subject to the specific application. For instance, in wind speed measurement application event driven model is adopted. In other applications such as target tracking, data should continuously be collected and delivered to the base station. Downstream data movement occurs only in situations where query-driven is to be utilized such as in the cases of requesting data or sending control commands from the sink to other nodes. Query-driven traffic is similar to the event-driven case except that the data is pulled by the sink in the first case while it is pushed by the sensor nodes in the latter [7].

In some cases, it is necessary to include a combination of the three models: continuous, event and query-driven types [7].

Congestion is a common problem in a packet switched networks. Congestion occurs whenever data sources are pushing data at rates exceeding the capacity of the network at one or more intermediate routes, which in the case of WSN passes through one or more nodes. Packet drop is inevitable when traffic congestion occurs and that has serious negative implication on the QoS of the network [10].

Several types of researches have been done in the area of congestion in wireless sensor network, clustering technique and the effect of sink node mobility [2,4,9,12-17,19-22].

II. The Proposed Algorithm of Wavelet Neural Network Congestion Controller (WNNCC):

The controller is deployed at the buffer of the sink node side in order to control the data rate of active cluster head nodes as a measure to reduce congestion in the network. When the traffic is received by the sink node the congestion detection starts checking the state of the sink buffer.

When the buffer occupancy reaches 90% full, the congestion control is activated. Next data traffic is then estimated at t+1 and the controller decides accordingly whether to increase or decrease the data rate of active cluster head nodes. Otherwise the rate is set to 225 packets of active cluster head nodes as shown in the following statements.

If (T > (0.9*Bf))

Activated congestion control

Compute traffic

Compute rate

Else

Rate= 225 (90% of buffer size)

End

Where

T: The Traffic of an active cluster head node.

Bf: The buffer size.

The neural network is used to estimate the traffic of an active cluster head nodes with different rate. An updating the weight of layers is accomplished by using the error E between the desired buffer and occupancy buffer. The Back Propagation (BP) training algorithm is used for updating the connection weights.

The activation function of the hidden layer neurons is the modified wavelet function. There is one hidden layer in the design of the neural network.

The neural network consists of three layers to accomplish traffic signal controller. The number of neurons in each of the three layers are as follows: There is one node in the input layer while in the hidden layer there are seven nodes (by trial and error) and one in the output layer.

The structure of neural network is shown in Fig. (1) where **Ψ** is the wavelet function. The input of the neural network is the traffic from active cluster head nodes and denoted by **T(t)**. The output of neural network is the estimated traffic in the next time denoted by *T* (t+1).

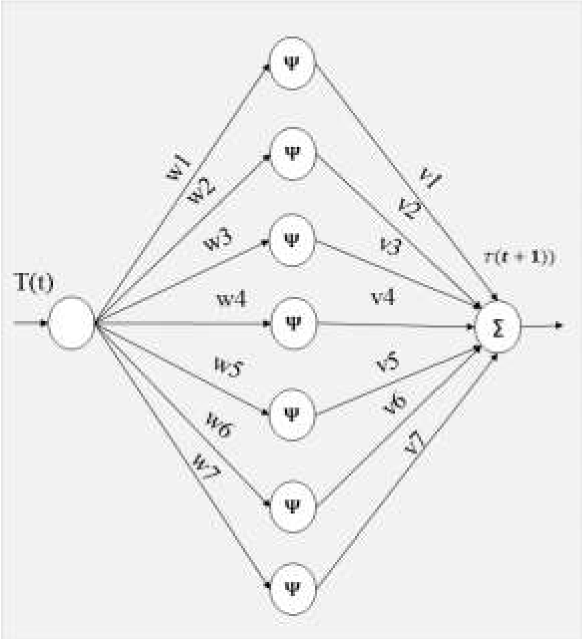

Fig. 1 Structure of the Wavelet Neural Network.

The modified Gaussian wavelet equation is used as an activation function of this network as given in equation (1) and the derivative of this activation function as given in equation (2) [5].

$$\Psi(t) = \frac{psum(b)}{\sqrt{2\pi}} e^{-\frac{psum(b)^2}{2}} \qquad \ldots(1)$$

$$\hat{\Psi}(t) = \frac{1}{\sqrt{2\pi}} e^{-\frac{psum(b)^2}{2}} * (1 - psum(b)^2) \qquad \ldots(2)$$

The following equations, perform the step of adjusting the weights during training phase using linear transform to estimate the next traffic [6,8].

$$psum(b) = w_b * \mathrm{T(t)} \qquad \ldots(3)$$

$$O_b = f\big(psum(b)\big) \qquad \ldots(4)$$

Here b represents the input and hidden layer neuron index while w_b is the weight of the connection between the input layer neuron and a hidden neuron. T(t) is the input at time t fed to the input layer. Gaussian function is chosen to be the activation function as defined by equation (1).

The following equation represents the output of the output layer neuron.

$$T(t+1) = \sum_b v_b * O_b \qquad \ldots(5)$$

Here, v_b represents the connection weights between the hidden and output layer.

The least mean squares are adopted as a measure of error and are defined in equation (6); as :

$$error = \frac{1}{2} \sum_{\substack{all\ trainning \\ pattrens}} (O_d - T(t+1))^2 \qquad \ldots(6)$$

$O_{d:}$ denotes the desired output.

The network is trained, that is the weights are modified using the backpropagation algorithm as per equations (7) and (8):

$$\Delta v_{bc} = \eta * E(t) * O_b \qquad \ldots(7)$$

$$E(t) = Od - T(t+1) \qquad \ldots(8)$$

Where E(t) is the error used to adapt the weight in NN , and η is the learning step size (set here to 0.05 by trial and error).

The weights between the output and hidden layers are modified as in equation (9)

$$v_{bc}(t+1) = v_{bc}(t) + \Delta v_{bc} \qquad \ldots(9)$$

While the connecting weights between hidden layer and input layer are modified using equations (10) to (12):

$$w_b(t+1) = w_b + \Delta w_b \qquad \ldots(10)$$

$$\Delta w_b = \eta * \delta_b * \mathrm{T(t)} \qquad \ldots(11)$$

$$\delta_b = E(t) * v_{bc}\left(\frac{1}{\sqrt{2\pi}} e^{-\frac{psum(b)^2}{2}} * (1 - psum(b)^2)\right) \qquad \ldots(12)$$

Where δ_b represents the hidden layer error.

Another controller is deployed using Neural Network with unipolar sigmoid activation function to compare it with WNNC.

The structure of this network is one node in the input layer and seven nodes in the hidden layer and one node in the output layer, the learning rate is 0.01 (by trial and error).

Fig. (2) shows the block diagram of the congestion controller. E(t) is the error used to adapt the weight in the NN as given in equation (8) and to estimate the next traffic the following equation derived from block diagram is used:

$$R(t+1) = Bf + G \cdot E(t) - T(t+1) \qquad \ldots(13)$$

Where

G: The proportional gain.

$R(t+1)$: The estimated traffic load of the network in the next time.

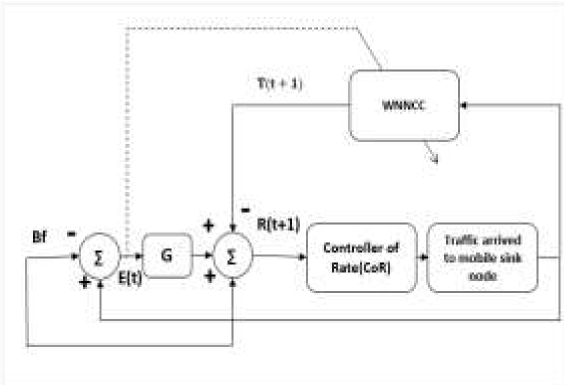

Fig. 2 Block diagram of controller.

The Controller of Rate (CoR) divides the estimated traffic calculated from Buffer Occupancy Control (BOC) among the upstream active cluster head nodes. The distribution of the rate is arranged proportionally to the node rate in previous time to achieve fairness among the nodes. This can be explained in equation (14).

$$B_R(S_i(t+1)) = R(t+1) \cdot \frac{B_R(S_i(t))}{\sum_{i=0}^{m} B_{Ri}} \quad \ldots(14)$$

Where

m: the of number of active cluster head nodes.

$B_R(S_i(t+1))$: the new rate is communicated to upstream node S_i to avoid congestion in next time.

The CoR calculates the new rate for each traffic source. The new rate is then sent to the CoR. In turn the CoR will be responsible for notifying all the active nodes of the new rate by including the value of the new rate in the packet header of message acknowledgement.

The flowchart (Fig. (3)) shows the overall units of the proposed modified Wavelet Neural Network Congestion Controller at the mobile sink node using WNNCC to estimate the network traffic and distribute it among active cluster head nodes fairly.

III. The Network Design

The network consists of five clusters. Each cluster contains nine source nodes and one cluster head. When an event occurs in one of these clusters, source nodes sense the data then sends it to the cluster head. Furthermore, the cluster head sends the packet to the sink node when the sink node moves in this cluster vicinity.

The mobility of sink node in this network is random. It starts moving at time 0.0 second among the clusters, then stops at random location close to one cluster or between two or three clusters for twenty seconds period. It then moves for another location then stops and so on until the simulation time finished. The simulation time in this network is 200 second and the speed of mobile sink node is random.

IV. Simulation Model and Parameter

The model contains one mobile sink sensor node that is moving randomly among clusters. The transmission data that is collected by wireless sensor nodes are generated randomly, when an event occur in any cluster the source node sends the data to the cluster head node then the cluster head node send it to the sink node.

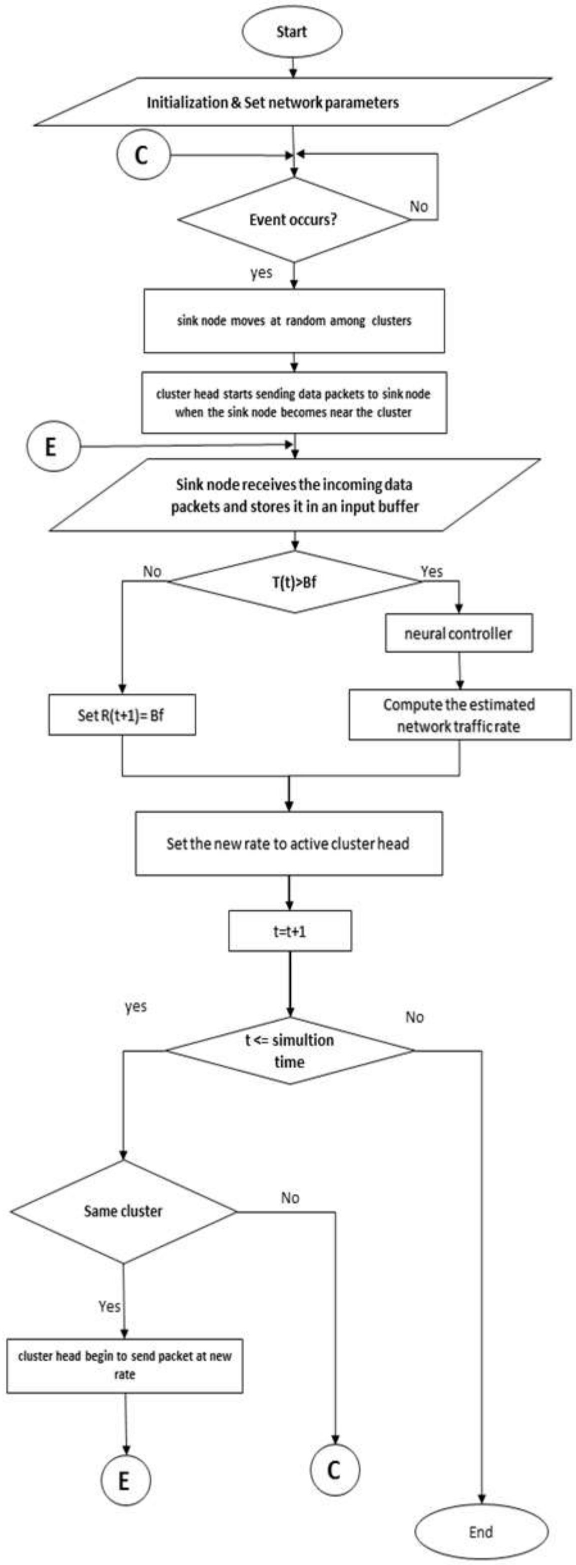

Fig. 3 Flow Chart diagram of the proposed congestion controller.

In the simulation model, the type of simulated traffic is Constant Bit Rate (CBR). The size of each packet is 512 bytes, the buffer size of each source node is set as 50 packets, the buffer size of the cluster head node is 150 packets, and the buffer size of the sink node is 250 packets.
Fig. (4) shows the simulation model before running.
The performance is measured after 10 simulation rounds. The time of simulation round is 200 second. The simulation parameters are summarized in Table 1.

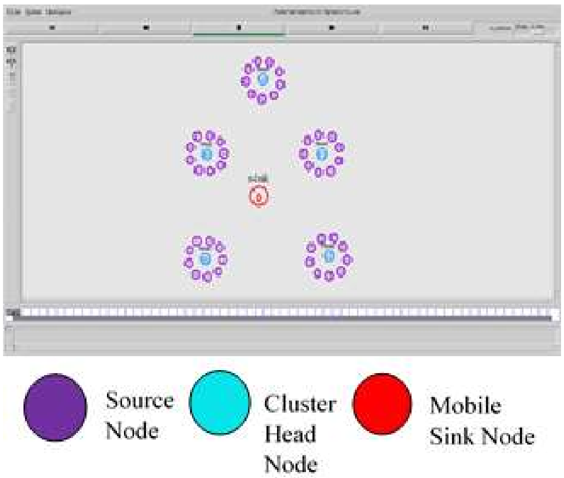

Fig. 4 Simulation Model before Running.

Table 1. Simulation Parameters.

Parameters	Values
Simulations Area	550*550 m^2
Simulation Time	200 sec
Number of Nodes	45
Number of Cluster Heads	5
Number of Sinks	1
Routing Protocol	Ad-hoc On-demand Multipath Distance Vector (AOMDV)
Buffer Size of Node	50 packets
Buffer Size of Cluster Head	150 packets
Buffer Size of sink	250 packets
Packet Length	512Byte
Channel Bandwidth	2(Mega bit per sec)
Bit Rate	1(Mega bit per sec)
Radio Propagation Model	Two-Ray-Ground model
Channel Type	Wireless Channel
Traffic Type	CBR
Antenna Model	Omni directional antenna
MAC Layer	IEEE 802.11
Interface Queue Type	Queue/DropTail
Sink Speed	Random

V. Simulation Results

The NNCC and the WNNCC are designed in the sink node, depending on the wireless sensor network shown in Fig. (4), the performance metrics are measured, such as the Throughput, the Packet Delivery ratio, the End_To_End delay and the packet-loss, and compared with the NCC.

Throughput:

It is defined as the total number of packets received per unit time [3,7,11], Fig. (5) shows the comparison between the average throughput with NCC, NNCC and WNNCC over ten rounds.

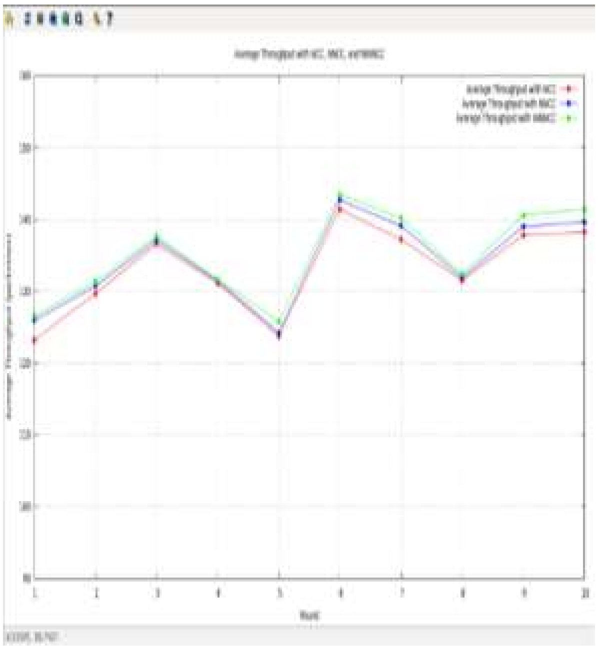

Fig. 5: The Average Throughput with NCC, NNCC, and WNNCC over ten rounds.

End-to-End Delay:

Delay is defined as the duration when a packet enters a queue of a beginning node until it arrives at the ending node [3,7,11]. The definition of beginning and ending nodes depends on the type of delay. The End-to-End delay is the type of delay that is used in paper. The End-to-End delay is the duration time of the cluster head node that transmits data to the sink node.

The comparison between the average End-to-End Delay with NCC, NNCC and WNNCC over ten rounds is shown in Fig. (6).

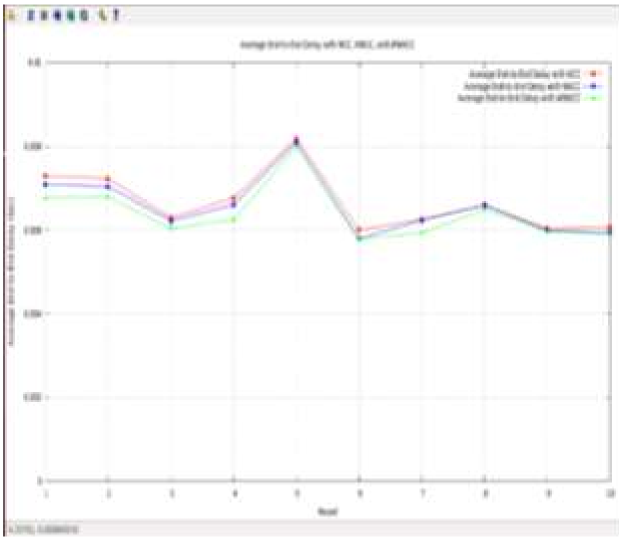

Fig. 6: The average End-to-End Delay with NCC, NNCC and WNNCC over ten rounds.

Packet-Loss:

Is defined as the difference between the number of packets sent by the source and the number of packets received at the destination [3,7,11], the count of packet-loss is decreased with WNNCC and NNCC in comparison with NCC over ten rounds as shown in Fig. (7).

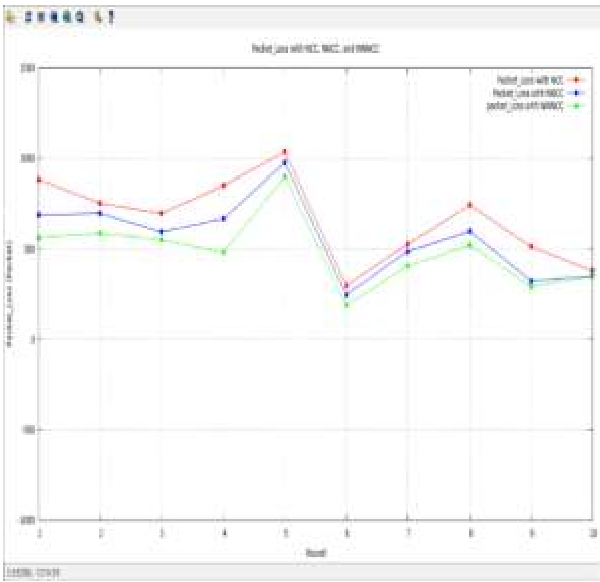

Fig. 7 The count of packet-loss without and with WNNCC over ten rounds.

Packet Delivery Ratio:

It is the ratio of received packet over the sent packet in the network [3,7,11], the packet delivery ratio is increased with WNNCC in comparison with the packet delivery ratio with NNCC and NCC, as shown in Fig. 8.

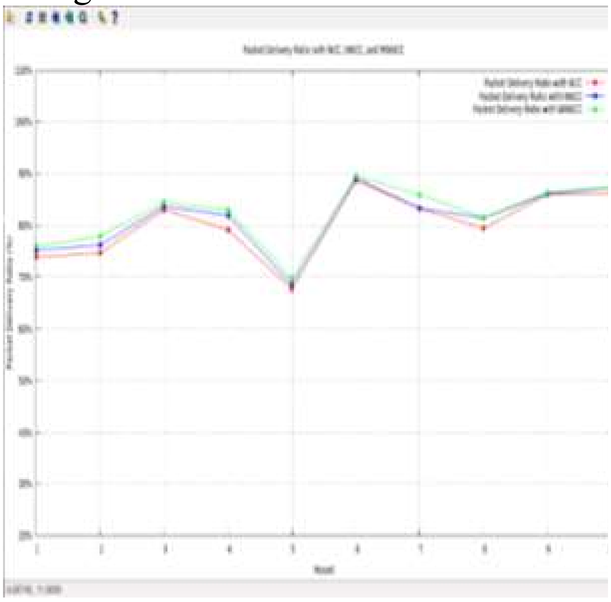

Fig. 8 The Packet Delivery Ratio with NCC, NNCC and WNNCC.

VI. The performance of wireless sensor network design:

Fig. 9 shows the ratio of increase and decrease for the performance metrics with NNCC compared with NCC in percentage, Fig. 10 shows the ratio of increase and decrease for the performance metrics with WNNCC compared with NCC in percentage, and Fig. 11 shows the ratio of increase and decrease for the performance metrics with WNNCC compared with NCC in percentage. These ratios are calculated using equ. 15.

$$Ratio = \frac{(N_P - O_P)}{O_P} \times 100\% \qquad \ldots 15$$

Where

N_P: referred to the New performance metrics with WNNCC**.**

O_P: referred to the Old performance metrics without WNNCC.

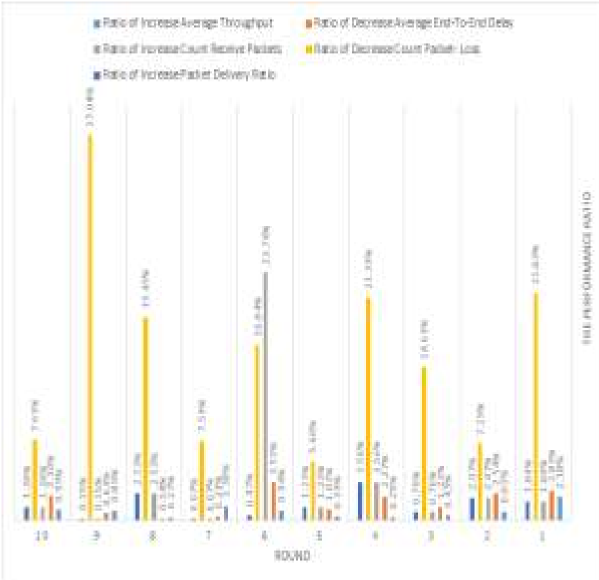

Fig. 9 The ratio of increase or decrease the performance metrics for ten rounds with NNCC in compared with NCC.

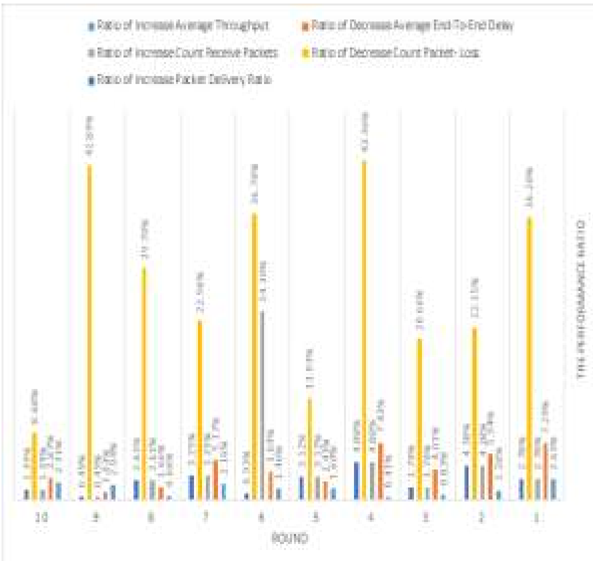

Fig. 10 The ratio of increase or decrease the performance metrics for ten rounds with WNNCC in compared with NCC.

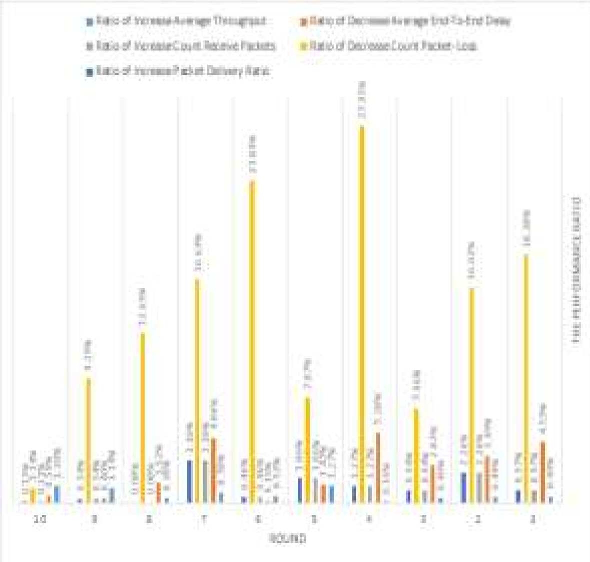

Fig. 11 The ratio of increase or decrease the performance metrics for ten rounds with WNNCC in compared with NNCC.

VII. Conclusions

This paper proposed a congestion controller algorithm that is implemented in WSN using NS2 simulation software. We calculated the average of performance measures to achieve more accurate indicator. The outcome of implementation indicates the following conclusions:

1. The proposed algorithm WNNCC enhances the transmission rate and reduces the level of congestion, resulting in a better overall performance of WSN.
2. The WNNCC improves the QoS in the form of End-To-End delay and Packet-Loss minimization as well as enhancing the throughput and Packet Delivery ratio of the network.
3. The Average throughput achieved by WNNCC algorithm is better than that NNCC as shown in Fig. (11).
4. The Average End-To-End delay achieved by the WNNCC algorithm is lower than NNCC as shown in Fig. (11).
5. The WNNCC algorithm give less average Packet-loss than that NNCC as shown in Fig. (11).
6. The WNNCC algorithm give high Packet Delivery ratio than that NNCC as shown in Fig. (11).

REFERENCES:

[1] A. Abu Taleb, T. Alhmiedat, O. A. Hassan, and N. M. Turab, "A Survey of Sink Mobility Models for Wireless Sensor Networks", Journal of Emerging Trends in Computing and Information Sciences, CIS Journal, Vol. 4, No. 9, pp. 679_687, September 2013.

[2] A. Ghaffari, "Congestion control mechanisms in wireless sensor networks: A survey ", Journal of Network and Computer Applications, Elsevier Ltd, pp.1_15, 2015.

[3] A. Varshney, and P. Maheshwari, "Comparative Study of AODV & AOMDV Routing Protocol", International Science Press (I J C T A), pp. 1-7, 2016.

[4] Chandni, K. Sharma, "Improvement in Qos using sink mobility in WSN", International Journal of Computer Trends and Technology (IJCTT), vol. 4, pp.1_4, April 2013.

[5] D. Veitch, "Wavelet Neural Networks and their Application in

The Study of Dynamical Systems", M.Sc. Thesis, University of York, UK, August 2005.

[6] F. O. Karray, and C. D. Silva, "Soft Computing and Intelligent Systems Design", 1st edition, Pearson Education Limited, 2004.

[7] I. M. M. El Emary, and S. Ramakrishnan,"Wireless Sensor Networks from Theory to Applications", 1st edition, Taylor & Francis Group, LLC, pp. 164_166, 2014.

[8] J. M. Zurada, "Introduction to Artificial Neural Network", 1st edition, West Publishing Company, United States of America, 1992.

[9] J. Wang, X. Yang, Z. Zhang, B. Li, and J. Kim, "A Survey about Routing Protocols with Mobile Sink for Wireless Sensor Network", International Journal of Future Generation Communication and Networking, Vol.7, No.5, pp.221-228, 2014.

[10] J. Zhao, L. Wang, S. Li, X. Liu, Z. Yuan, and Z. Gao, "A Survey of Congestion Control Mechanisms in Wireless Sensor Networks", Sixth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, IEEE, pp.719_722, 2010.

[11] K. Sohraby, D. Minoli, and T. Znati, " Wireless Sensor Networks Technology, Protocols, and Applications", John Wiley & Sons, Inc., 1st edition,2007.

[12] L. D. Shanthi, R. Archana, M. Harshitha, N. M. Niveditha, " Improving the Network Lifetime in Wireless Sensor Networks Using Fuzzy Based Centrality Algorithm", International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE), Vol. 4, pp.327_331, May 2015.

[13] M. Koç and I. Korpeoglu, "Controlled Sink Mobility Algorithms for Wireless Sensor Networks", International Journal of Distributed Sensor Networks, Hindawi Publishing Corporation, Vol. 2014, pp.1_12, 2014.

[14] M. R. Abd Al-Redha, "An Optimal Energy Aware Clustering Protocol for WSN with Sink Mobility", M.Sc. Thesis, Al-Nahrain University, Iraq, February 2016.

[15] N. Prabakaran, B. S. Raja, R. Prabakaran, and D. V. R. Sarma, "Rate Optimization Scheme for Node Level Congestion in Wireless Sensor Networks", IEEE, 2011.

[16] P. Nayak, and A. Devulapalli, "A Fuzzy Logic based Clustering Algorithm for WSN to extend the Network Lifetime", IEEE Sensors Journal, pp.1_8, 2015.

[17] R. Wan, X. Zhang, J. Zhou, and J. Liu, "An Adaptive Congestion Control Strategy for Wireless Sensor Networks", International Conference of Information Technology, Computer Engineering and Management Sciences, IEEE, pp.265_268, 2011.

[18] S. A. Raj, and S. Jose, "Analysis of Mobile Sink Based Big Data Gathering in Dynamic Wireless Sensor Networks", International Conference on Automatic Control and Dynamic Optimization

Techniques (ICACDOT) International Institute of Information Technology (I²IT), Pune, pp.875_880, 2016.

[19] U. Urathal alias Sri Swathiga, and C. Chandrasekar, "An Efficient Fuzzy based Congestion Control Technique for Wireless Sensor Networks", International Journal of Computer Applications, Vol. 40, No.14, pp.47_55, February 2012.

[20] Z. A. Ismaeel, "Improvement of Network Lifetime of WSN Using Controlled Mobility Algorithms", M.Sc. Thesis, Al-Nahrain University, Iraq, June 2016.

[21] Z. G. Faisal, "Traffic Management in Wireless Sensor Network Based on Intelligent Monitoring Algorithms", M.Sc. Thesis, University of Baghdad, Iraq, May 2014.

[22] Z. Meng, J. Qiao, and L. Zhang, "Design and Implementation: Adaptive Active Queue Management Algorithm based on Neural Network", Tenth International Conference on Computational Intelligence and Security, IEEE, pp.104_108, 2014.

السيطرة على اثار الازدحام في شبكات الاستشعار اللاسلكيه مع عقده تجميع متحركة اعتمادا على الشبكة العصبية المويجية

أ.م.د. نادية عدنان شلتاغ

قسم هندسة الحاسبات/ جامعة بغداد

نريمان رياض محمد علي

قسم هندسة الحاسبات/ جامعة بغداد

الخلاصة :-

اصبح التطور لتطبيقات شبكات الاستشعار اللاسلكية عنصرا اساسيا في الانظمة المترابطة المعقدة وبات التصميم المثالي لشبكة الاستشعار اللاسلكية من المهام التي تشكل تحديا لايستهان به نتيجة اجتماع اكثر من عامل ومتطلبات تقنية واجب النظر بها وتحقيقيها في ذات الوقت. في هذا البحث, تم تقديم دراسة لاعتماد عقدة التجميع التي تنتقل بشكل عشوائي بين مجاميع عقد الشبكة لغرض تحسين مستوى الخدمة اعتمادا على مسيطر ذكي. صمم هذا المسيطر باعتماد الشبكة العصبية المويجية. تم تدريب هذا المسيطر بشكل غير متصل باستخدام خوارزمية الانتشار الخلفي. ولكي يتم اختبار الاداء لهذا النظام فقد تم اعتماد محاكي الشبكات (NS-2), والذي تم تنصيبه على Ubuntu 14.04. اظهرت نتائج المحاكاة ان الخوارزمية المقترحة تحسن الاداء من حيث تقليل الاختناق وتعطي عامل اداء افضل ومعدل اعلى لسرعة ايصال البيانات, ونسبة ايصال افضل للبيانات اضافة الى نسبة تاخير اقل ونسبة ضياع اوطأ للبيانات. تم تدريب الشبكة العصبية باستخدام برنامج ماتلاب 2014أ, على حاسبة بالمواصفات التالية:
windows 7 (32-bit), core i5, RAM 4 GB.