Electrical, Electronics and communications, and Computer Engineering

Spike neural network as a controller in SDN network

Anwar Dhyaa Majeed
M.Sc. student
Computer Engineering Department
College of Engineering
University of Baghdad
a.dhyaa1305@coeng.uobaghdad.edu.iq

Nadia Adnan Shiltagh Al-Jamali
Asst. Prof. Dr.
Computer Engineering Department
College of Engineering
University of Baghdad
nadia.aljamali@coeng.uobaghdad.edu.iq

ABSTRACT

The paper proposes a methodology for predicting packet flow at the data plane in smart SDN based on the intelligent controller of spike neural networks(SNN). This methodology is applied to predict the subsequent step of the packet flow, consequently reducing the overcrowding that might happen. The centralized controller acts as a reactive controller for managing the clustering head process in the Software Defined Network data layer in the proposed model. The simulation results show the capability of Spike Neural Network controller in SDN control layer to improve the (QoS) in the whole network in terms of minimizing the packet loss ratio and increased the buffer utilization ratio.

Keywords: Software Defined Network, spike neural networks, The Internet of Things, packet loss ratio, buffer utilization ratio

الشبكة العصبية المتصاعدة كوحدة تحكم في شبكة

انور ضياء مجيد
طالب ماجستير
جامعة بغداد / كلية الهندسة / قسم هندسة الحاسوب

نادية عدنان شلتاغ الجمالي
أ.م.د.
جامعة بغداد / كلية الهندسة / قسم هندسة الحاسوب

الخلاصة

يقترح البحث منهجية للتنبؤ بتدفق حركة المرور على مستوى البيانات في شبكة SDN الذكية استنادًا إلى وحدة التحكم الذكية للشبكات العصبية الأشواك (SNN). ستساعد هذه المنهجية الشبكة على التنبؤ مسبقًا لتدفق الحزمة ، والتي ستسهم بالتالي في تقليل الازدحام الذي قد يحدث. للنموذج المقترح يعمل كوحدة التحكم المركزية ووحدة تحكم تفاعلية لإدارة عملية رأس التجميع في طبقة بيانات الشبكة المعرفة بالبرمجيات. تظهر نتائج المحاكاة قدرة وحدة تحكم الشبكة العصبية الاشواك في طبقة التحكم SDN لتحسين (QoS) في الشبكة بأكملها ، من حيث تقليل نسبة فقدان الحزمة وزيادة نسبة استخدام المخزن المؤقت.

الكلمات الرئيسية: الشبكة المعرفة بالبرمجيات ، الشبكات العصبية المرتفعة ، إنترنت الأشياء ، نسبة فقدان الحزمة ، نسبة استخدام المخزن المؤقت.

1. INTRODUCTION

In recent years, it has become been widely expected to use software-defined networking (SDN) in wireless sensor networks (WSNs), so the WSNs' have great dynamic performance. SDN is a novel type of network structural design, which can overcome the drawbacks of existing network structures **(Gelberger, et al., 2013).** It acts on separating the data plane (DP) and control plane (CP) of a network, meaning that Smart Controller conFig.s forwarding elements (FEs) using precise forwarding procedures for data packets of dissimilar streams **(F., Richard Yu, et al., 2020)**. The controller gets enough information to perform the mission. That's why the control protocols distributed are not required anymore between (FEs). Besides optimizing the network, the controller might interact with applications **(Buurman, Ben, et al., 2020)**. WSN is composed of sensor nodes having capabilities to communicate, compute and sense. Most sensing nodes contain batteries, eventually leading to shortening the lifetime of nodes. Common requirements for software-defined networking deployment in WSNs are surveyed in **(Kobo, et al., 2017, and Ndiaye, et al., 2017)**. Ndiaye et al. concentrated on the mechanism of managing WSN by Software-defined networking. Kobo et al. focused on the architecture view of software-defined networking in WSNs. The authors of **(Jagadeesan et al.,2014, Haque et al., 2016)** presented a section on SDN application in wireless networks. However, none of these sections sectioned what could be controlled by software-defined networking in WSNs and how applying software-defined networking in WSNs is dissimilar to wireline networks. Software-defined wireless sensor networks (SD-WSNs) are lately proposed to enable Wireless sensor networks to take advantage of Software-defined networking. It is essential to simplify sensing nodes' operation for saving power and managing wireless sensor networks with a robust controller having full vision over the whole network instead of distributed control protocols **(Hayes, et al., 2016)**. This paper contains other sections that can be listed as the following: Section two presents the SDN controller and artificial intelligence overview. Section three presents the method and algorithm of the proposed intelligent control. Section four presents the results of proposed intelligent SDN and compare with other technique. Section 5 contains the conclusions and suggestions for future work

A. MOTIVATION

Data flowing volume in the data plane can be a highly significant matter in traffic management in software-defined networking. Since the sensor number connecting to switches goes up in the data plane, that will be faced with an increase in the traffic load in queuing buffer. Furthermore, when the switch number in a software-defined networking Gateway goes up, the centralized controller's performance in its controlling plane would not succeed in processing the entire requests that come from the switches. More importantly, in recent years, using machine learning and artificial intelligence networks with software-defined networking has gained increased attention.

B. CONTRIBUTIONS

The paper has introduced an SNN as a congestion controller in the model proposed. A spiking neural network can be regarded as a Spiking neural network. Furthermore, with a different controller, which is based upon Artificial Neural Network, it is introduced for managing sensors in the spike Smart SDN network. However, the paper's essential contributions can be listed as the following: We proposed a spike intelligent Smart SDN model with an intelligent controller in a software-defined network control plane. Based on (SNN), that intelligent controller functions for estimating the volume of packets flowing in the network. A model of smart queuing for estimating buffer size capacity in the spike intelligent software defined network was proposed.

2. RELATED WORK AND BACKGROUND THEORETIC
2.1 RELATED WORK

In the last few years, many researches of WSN to increase the performance of the network are done. (**Arouna Ndam Njoya et al., 2017**) introduced a new optimization algorithm in SDN based on stochastic physics, which is characterized as effective, in terms of (ensuring complete coverage of target with fewer sensors and scalable), in terms of being implemented for very extensive or encompassing issues in sensible computational time. (**Qiaofeng Qin et al., 2018**) have combined results of robust experiments together with valid theoretical analysis and modeling. The results present a methodology that produces an array of optimum SDN controllers sites and assigns nodes to control units. The assessment results demonstrate the broad advantages our approach possesses over the latest methods. (**Murtaza Cicioğlu et al., 2018**) used software-defined networking. For this, an architecture of wireless body area network is proposed, which depends upon the approach of SDN having an algorithm with energy-efficient routing for healthcare architecture. For developing architecture with more flexibility, a controller is designed to manage the whole HUBs. The proposed architecture is designed by the use of Riverbed Modeler for analyzing performance. (**Mohammed Al-Hubaishi et al., 2018**) introduced a wireless sensor with SDN and operating network architecture. A new approach to the routing decision was developed in this study, which employs a fuzzy-based Dijkstra algorithm. The results suggest that the architecture proposed that supports SDN with fuzzy - based Dijkstra algorithm is better than those that employ the traditional Dijkstra algorithm. The proposed structure is capable of providing effective cluster routing and extending the lifetime of the network. (**Wei Ren et al., 2018**) proposed a hierarchical control framework with two levels (master and slave) to solve this matter. By an adjusted binary Particle swarm optimization algorithm for optimizing the control delay in SDN. Impossible to solve through modified Binary Particle Swarm Optimization (BPSO) algorithm for optimizing critical Internet of Things nodes' control cost and delay. However, comparing to different methods, the results demonstrate that it roughly minimizes the control delay of significant Internet of Things nodes by 30.56%. (**Ali Burhan Al-Shaikhli, et al., 2018**) proposed an interface protocol, which can also be referred to as Wireless Sensor and Actuator Network Flow, in charge of the whole communications among SDN controller and SDN oriented end devices. In this approach, the SDN controller possesses the intelligence of the network. The results have shown that the current Wireless Sensor and Actuator Network system's total performance is optimized. The processes of control and management are made simple through the model proposed. (**Mohammed J.F. Alenazi, et al., 2019**) have presented a new nodes metric, node-disjoint path (NDP) that is utilized for measuring the significance of the node regarding its varied connection to other nodes. According to a node-disjoint path, they propose two types of algorithms (NDP-global) and (NDP-cluster) to locate k-controllers to increase the network's robustness in the face of targeted attacks. In addition, the results indicated that the algorithm of the NDP-cluster has a similar delay performance to the k-median algorithm and can provide higher resilience to the SDN network. (**Murtaza Cicioglu et al., 2019**) proposed an architecture that can be managed, energy-sensitive, and having resilient architecture. The controller that is the SDN major component, handles all processes related to control and management around the network. The HUBsFlow interface protocol is used in a controller providing communications between the HUBs and controller and in wireless body area network interconnections. The findings have shown that the architecture proposed performs better than the classical wireless body area network architecture and meets the (QoS) requirements of IEEE / ISO 11073. (**Shirin Tahmasebi et al., 2020**) proposed the algorithm of SDN Controllers Placement. This algorithm functions based on the Cuckoo optimization algorithm (MhAs), taking inspiration from the lifestyle of birds called cuckoo. For evaluating this algorithm's performance, A comparison is made between the proposed approach and two modern

methods, (1) quantitative softening (QA) (2)simulated annealing (SA). The Experimentations show that this algorithm performs better than SA and QA about network performance by reducing the average distance among controllers sensors and up to 9% and 13%.

3. SOFTWARE-DEFINED- WIRELESS SENSOR NETWORK (SD-WSN)

Here, an overview SD-WSN will be given. First, a broad description of SD-WSN architecture and explain its differences regarding non- SD-WSN. After that, a comparison with SD-WSN and wireline software-defined wireless was made. Lastly, an overview of software tools of SD-WSNs is given **(Fernandez et al., 2013, Kreutz et al., 2014, Shiltagh et al., 2019, Ammari et al., 2020, and Salman et al., 2020)** as shown in **Fig. 1.**
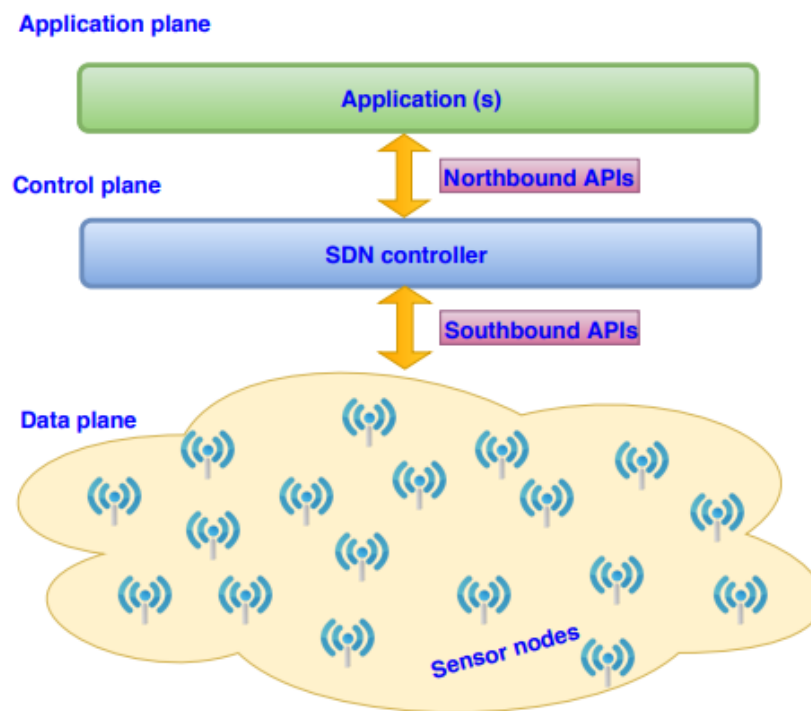

**Figure 1**. The general architecture of software-defined wireless sensor networks.

4. THE PROPOSED MODEL: ARCHITECTURE AND SPECIFICATIONS

In this system, the intelligent controllers are proposed in the SDN platform. These controllers are able to compute the data flow of the physical area. Moreover, one of the functions in a proactive way is an SNN for estimating the sensing area's packet flowing. The other controller functions are reactive and are assigned to choose the head of cluster and cluster members. The controller of SNN is efficiently able to choose the head of a cluster and members of the cluster in the sensor zone. The proposed model can compute the rate of data flowing, contributing to coordinating the available buffer capacity by having a set of active sensing nodes in the network to prevent buffer overrun**.** In this paper, an SDN smart controller was proposed based on SNN Controller. Spiking Neural Networks possess a structure similar to conventional neural networks. However, dissimilarity existing in structural design can be outlined numbers of the presynaptic ending (synaptic terminals) within each layer of nerve cells. The architecture of neural networks contains a diverse layer called J, I, and H. These initials stand for the three layers: (1) input layer, (2) hidden layer (3) output layer separately, as presented in **Fig. 2 (A)**. The structure also consists of a feed forward fully- connected SNN with several delayed the presynaptic ending. Synapses are

signal connectivity consisting of a constant number of D the presynaptic end; each one acts as sub-connectivity related with a dissimilar weight and delay as **Fig. 2 (B)**.
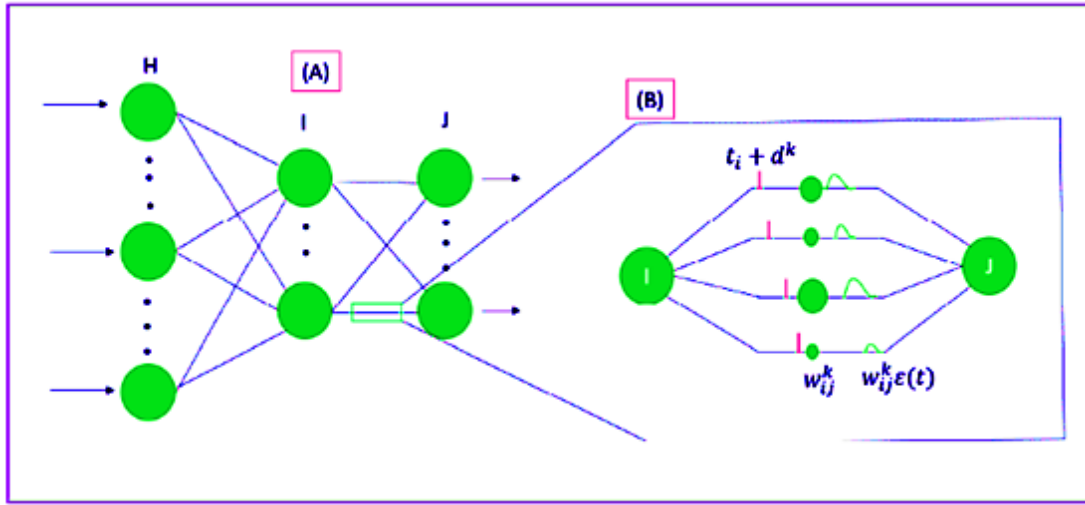
Figure 2.**(A)** Feedforward SNN,**(B)** connection composed of several delayed synaptic ending

The delay d^k of a synaptic terminal k can be determined through the dissimilarity among the presynaptic neuron's firing time i, and the moment the PSP begins to rise **Fig. 3.**
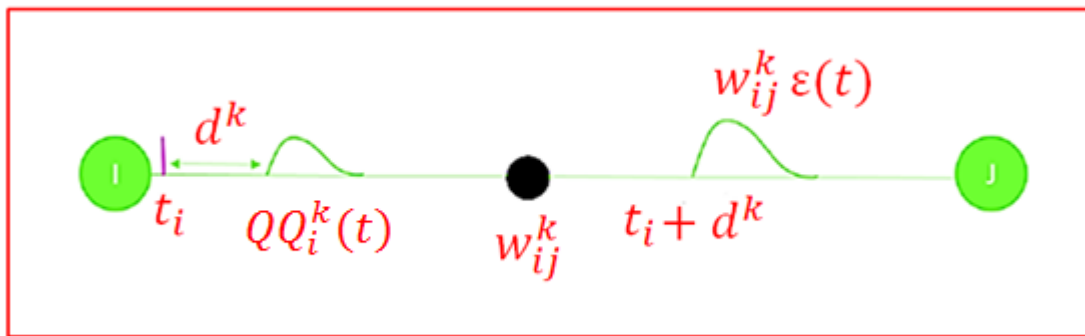
Figure 3. Single synaptic terminal.

A typical architecture of layer feed-forward neural network FF NN contains neuronal layers communicating only with neurons in successive layers. That is to say, organized in layers, and each unit in a layer is connected with the units in the whole preceding layer. Basically, there are three layers of this type of network, which can be illustrated as follows: the ***input layer*** behaves as network input; in fact, it contains no neurons, since no actual processing is involved, the so-called input-neurons must possess a particular output. In spiking neurons, the input-neurons work on firing spike-train defined in advance. ***Output-layer***, the neurons' spike-trains create the network output. In between the input and output-layer there could be any number of hidden layers. Each sub-connection possesses its weight variable w_{ij}^k that is remaining fixed, but is updated by SpikeProp, which is among the first basic machine learning rules and applies the backpropagation principle of feed-forward SNN. The variables $t_h^{(act)}$, $t_i^{(act)}$ and $t_j^{(act)}$ are representing certain neurons' actual spike times in the particular layers. The spike response function for a single sub-connection from a neuron i to neuron membrane in the following layer is (t) delayed through its spike time $t_i^{(act)}$ and delay d^k, and 0 for $t \le t_i^{(act)} + d^k$

$$QQ_i^k(t) = (t - t_i^{(act)} - d^k) \qquad \text{Eq. (1)}$$

a neuron MP is (t), equivalent to the weighted sum of incoming delayed Spike Response Functions(SRF) from the preceding layer:

$$m_j(t) = \sum_{i=1}^{NI} \sum_{k=1}^{D} w_{ij}^k(R)\, QQ_i^k(t) \qquad \text{Eq. (2)}$$

Spike-Prop is involved only with the first spike. Therefore it is possible to ignore successive membrane dynamics. Error-function represents half the smallest mean squares, where T_j represents the training target spike time of output neuron j:

$$E = \frac{1}{2}\sum_{j\,\epsilon J}(t_j - T_j)^2 \qquad \text{Eq. (3)}$$

Spike-Prop is a method of gradient descent, acts on adjusting each weight in the direction reducing E. (54) To train the network, weights must be updated each between the output layer and hidden layers for minimizing the error:

$$\frac{\partial}{\partial t}\left(QQ_i^k\right) = QQ_i^k \left(\frac{1}{t-t_i^{(act)}-dk} - \frac{1}{\tau}\right) \qquad \text{Eq. (4)}$$

$$\text{Note}: \frac{\partial}{\partial t(QQ_i^k)} = 0 \; For \; t \le t_i^{(act)} + d_k$$

$$\delta_j = \frac{T_j - t_j^{(act)}}{\sum_{i=1}^{NI}\sum_{k=1}^{D} w_{ij}^k(R)\frac{\partial}{\partial t}(QQ_i^k)} \qquad \text{Eq. (5)}$$

$$\Delta w_{ij}^k(R) = \eta . \delta_j . QQ_i^k \qquad \text{Eq. (6)}$$

$$\delta_i = \frac{\sum_{j=i}^{NI}\delta_j \sum_{k=1}^{D} w_{ij}^k(R)\frac{\partial}{\partial t}(QQ_i^k)}{\sum_{h=1}^{NH}\sum_{k=1}^{D} w_{hi}^k(R)\frac{\partial}{\partial t}(QQ_h^k)} \qquad \text{Eq. (7)}$$

The firing process of a neuron, which can refer to action potential (AP), is conducted either by entire or complete operation or effect.

5. RESULTS AND   DISCUSSION

In this paper, MATLAB® R2018b was used in the proposed simulation. This paper proposed a smart SDN controller by using multi spike neural network, in this work, 100 sensors randomly placed in a sensing area of square shape. The implementation for the area is shown in **Fig. 5**. At the start of every scheduling period, the sensor generates traffic; in other words, sensor(s) conducts low-flow to high-flow. After that, the traffic is directed to Forwarding Cluster Head. The controller of SNN reduces the level of congestion. However, Forwarding Cluster Heads can be categorized as congestion when this ratio goes beyond the threshold level. **Table 1** shows the Simulation parameters of the proposed system.

Table 1. Simulation parameters of the proposed system.

Parameter	Value
Sensing field dimensions	100*100
Number of nodes	100
Router location	Any point in the map
Buffer size of Forwarding Cluster Heads	250 packet

Buffer size of each node	50 packet
Data packet size	1000 byte
Data Packet generating for each node	6 (packet /msec.)
All Nodes sill distance	5
Cluster Heads threshold value	15
Initial energy E_0	0.5 J
Adjusting Parameter	0.1
Energy consumption on circuit	50 nJ/bit
Energy Data	5 nJ/bit
Free-Space channel parameter	10 PJ/bit/m^2
path channel parameter	0.0013 pJ/bit/m^4

In this work, the random distribution achieved by generation random select 100 random point map (100*100) and distrusted around the base station at (50,50) as it is illustrated in **Fig. 4** . After distributing node, the K-mean cluster was applied to select the best location of FCH, where that achieve to get all the nodes (100) and enter to k mean cluster and cluster it to 3 groups, where each group belong the subset of nodes. The Center point will represent the location of a head cluster, as shown in **Fig. 4. Table 2** shows the position of FCH based on the k-mean cluster, where the position of FCH represent the three center point and through **Table 4** below note each node connected to any point of FCH.

Table 2. The location of FCH based of K-mean cluster.

Id FCH	X position FCH	Y position FCH
1	20	72
2	38	18
3	81	53

After distributing the node and determining the FCH, then connecting the FCH with the node, the model will be the LEACH protocol as shown in **Fig. 4**
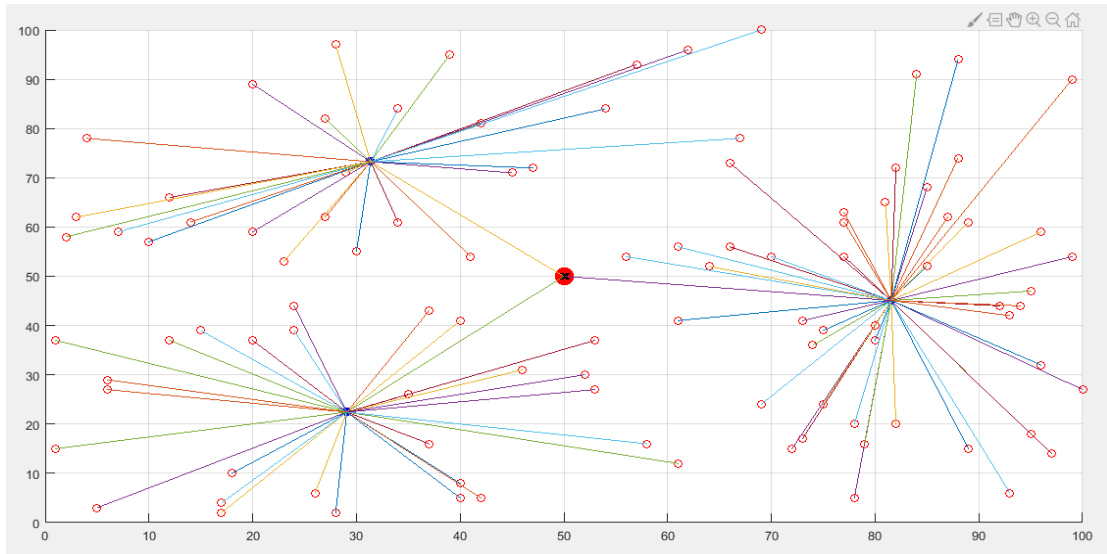
Figure 4. The connection of node and FCH based on the Leach protocol

The performance of the SNN-SDN model is explained in terms of quality of service concerning Buffer Utilization Ratio, Network Lifetime, Network Throughput Ratio, Network Energy Consumption, and Packet Loss Ratio.
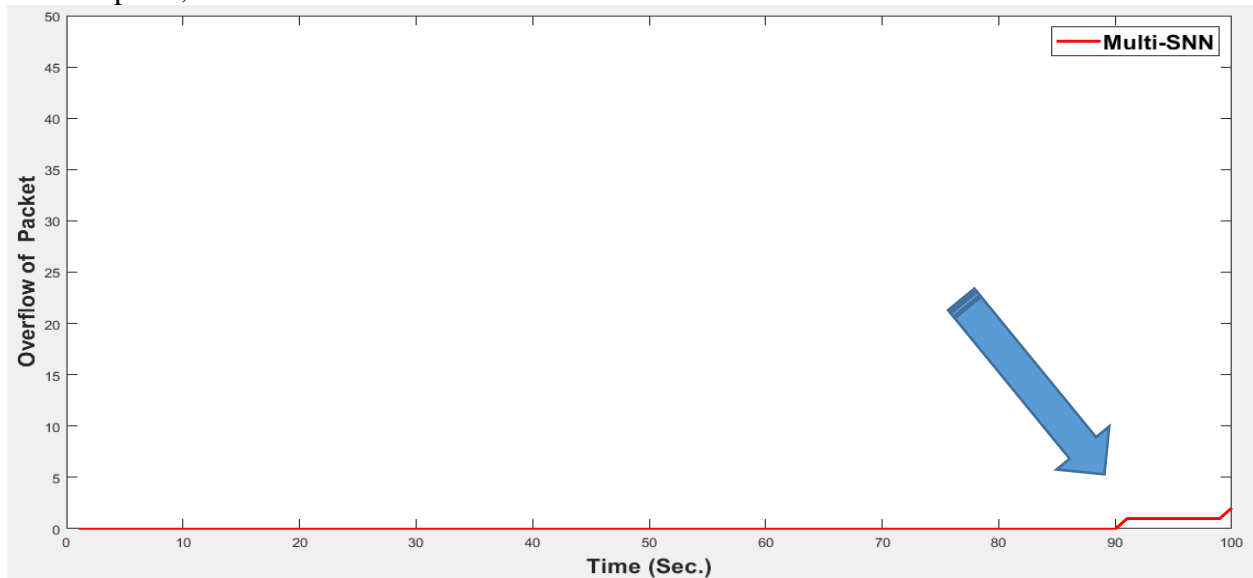

Figure 5. Packet overflow in SNN-SDN

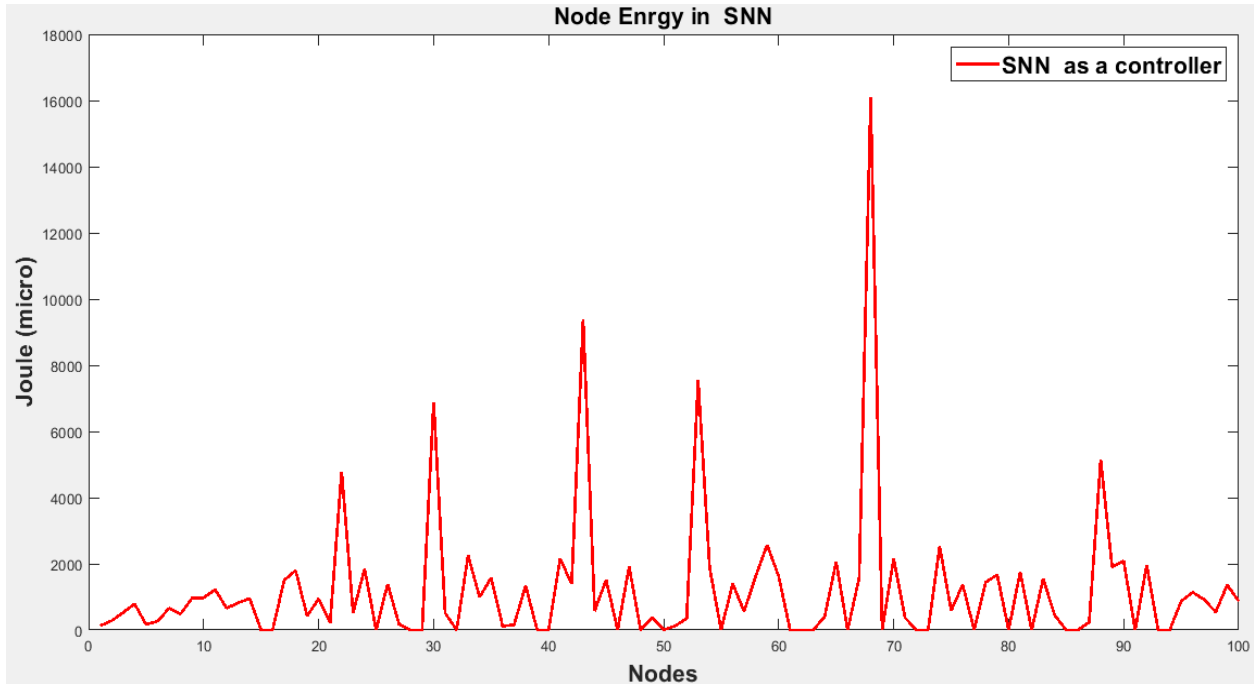In time 50 seconds, overflow was zero in SN, but in second 90, there was overflow in SNN of packet.

71

Figure 6. Network Energy Consumption in SNN-SDN

Fig. 6 in sensor number 10 has spent energy in SNN about 1000 Micro Joule. In sensor number 59 has spent energy about 2300 Micro Joule in SNN technique. The highest energy was 16000 Micro Joule in SNN technique. The main motion to use SNN as controller of congestion lies in increasing network power in packet flow estimation. SNN strength can be obtained by accurately modeling the synaptic interactions among the biological neurons, considering the spike firing time.
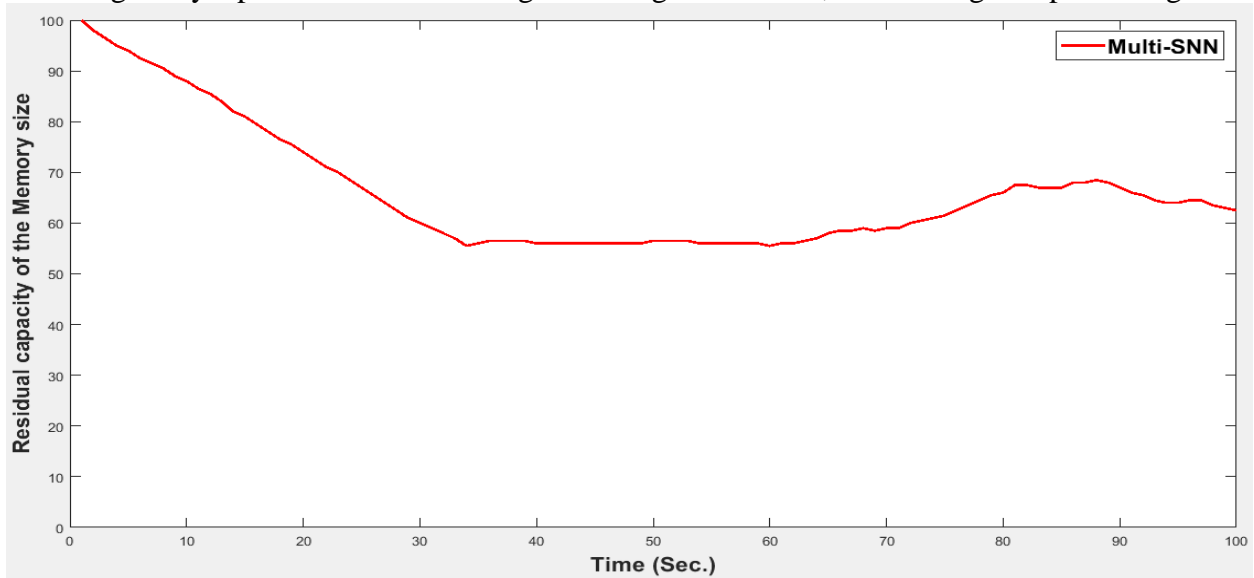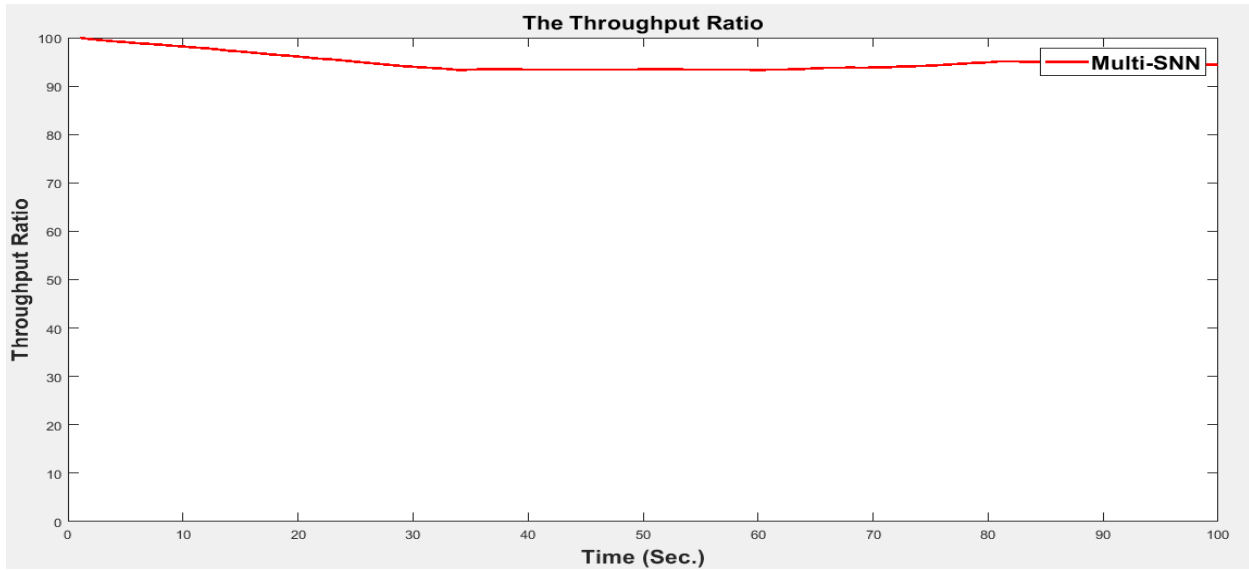

Figure 7. Residual buffer capacity in SNN-SDN

In 40 second in SNN the residual capacity of the memory size is 12, but in 70 second it is 28. In 100 second it is 21.

Figure 8. Network Throughput in SNN-SDN

In 20 second throughput ratio is 75 in SNN. In period number 50 second in SNN technique it is 59, and in 100 second is 61.
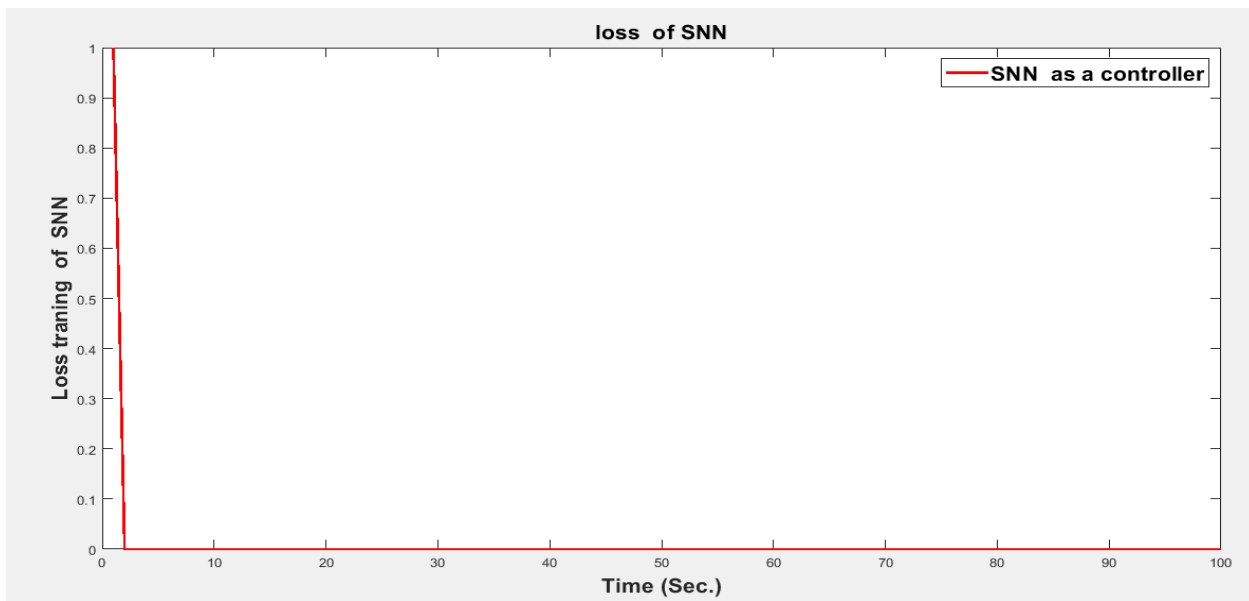
Figure 9. Error of Training Process in SNN-SDN

Error reduction through the process of training is clearly shown in **Fig. 9.** It can be seen that SNN can reach the error goal. **Table 3** shows a comparsion of this work compare work with other works.

Table 3. Comparison of the proposed work with   other works.

S	Authors	Study	My Study
1	Arouna Ndam Njoya et al., 2017	Development in the infrastructure layer	Development in the control layer
2	Qiaofeng Qin et al., 2018	Developed on the location of the controller	Development on the controller by use SNN technology
3	Murtaza Cicioğlu et al., 2018	The algorithm improves energy for healthcare architecture.	The algorithm improves overflow, energy, memory, density, loss of training, and data loss.
4	Mohammed Al-Hubaishi et al., 2018	Developed which employs a fuzzy-based Dijkstra algorithm.	Development on the controller by use SNN technology
5	Hamid Reza Faragardi et al., 2018	Improved the sink's position sort and network controllers.	Improved on a controller, By replacing the controller with one that uses artificial intelligence, specifically SNN technology
6	Wei Ren et al., 2018	The results demonstrate that it roughly minimizes the control delay	The results demonstrate that it roughly minimizes the overflow, energy, loss of training, and data loss
7	Ali Burhan Al-Shaikhli, et al., 2018	Proposed an interface protocol, the results have shown that of the current Wireless Sensor and Actuator Network system's total performance is optimized	The proposed intelligent controller use SNN technology; the results have shown that the Network system's total performance is optimized
8	Mohammed J.F. Alenazi, et al., 2019	Have presented a new nodes metric, node-disjoint path (NDP). The results indicated that the algorithm of the NDP-cluster has a similar delay performance to the k-median algorithm and can provide higher resilience to the network.	Have presented a new controller. The results indicated that the algorithm of improving on the controller by use SNN technology is better than the traditional controller
9	Murtaza Cicioglu et al., 2019	The findings have shown that the architecture proposed to perform better compared to the classical wireless body area network architecture and meets the (QoS) requirements of IEEE / ISO 11073.	The findings have shown that the architecture proposed to perform better compared to the classical SDN controller
10	Abdelhamied A. Ateya, et al., 2019	Developed a dynamic optimization algorithm, the results of Simulation have shown that this algorithm performs better than meta-heuristic algorithms (MhAs) and game-theory -based algorithms regarding reliability and time of execution	Developed an optimization algorithm, the results of Simulation have shown that this algorithm performs better than a traditional controller. The algorithm improves overflow, energy, memory, density, loss of training, and data loss.
11	Rong Chai et al., 2019	Considered the problem of placing the capacitive controller, capacitated controller placement problem (CCPP).	Development on the controller by use SNN technology
12	Kostas Choumas et al., 2019	Investigated an optimum controller placement for minimal control traffic. The heuristic placement involves 2.62% more bandwidth than the optimum range.	The algorithm improves overflow, energy, memory, density, loss of training, and data loss.
13	Amit Dvir et al., 2019	Introduced a new strategy for addressing the problem of controller placement, which provides protection to potential link failure, latency, and transparency	Improved controller, by replacing the controller with one that uses artificial intelligence, specifically SNN technology

		regarding wireless southbound interfacing.	
14	Zifu Fan et al., 2019	The results have shown the proposed algorithm's effectiveness and the strategy proposed are capable of ensuring the control layer's delay and reliability in most cases of link failures	The results have shown the proposed algorithm's effectiveness, and the proposed strategy is capable of improving overflow, memory, density, energy, loss of training, and data loss.
15	Biao Han et al 2019	The genetic algorithm was developed for solving such a problem.	The neural algorithm was developed for solving such a problem.
16	Deepak Singh Rana et al., 2019	Provided SDN network model, problems, and solutions that have a fundamental role in creating software and notify the user, and improved the efficiency of the SDN must be smart in the network management to avoid errors are presented.	Improved controller, by replacing the controller with one that uses artificial intelligence, specifically SNN technology
17	Shirin Tahmasebi et al., 2020	The Experimentations show that this algorithm performs better than simulated annealing (SA) and quantitative softening (QA) about the performance of the network by reducing the average distance among controllers sensors and up to 9% and 13%	Development on the controller by use SNN technology
18	Sahand Torkamani-Azar et al., 2020	The simulation results demonstrated that this solution could reduce memory consumption compared with various algorithms concerning placing controllers in dissimilar topologies of the SDN network.	Developed an optimization algorithm, the results of Simulation have shown that this algorithm performs better than traditional controller. The algorithm improves overflow, energy, memory, density, loss of training, and data loss.

6. CONCLUSIONS

 The paper proposes smart SDN architecture for utilization in wireless sensor network applications. The proposed model presents smart controllers in the Software-defined networking control plane, and these controllers are able to estimate the sensing area's packet flow. Moreover, one of the functions is in a proactive way in an SNN for estimating the sensing area's packet flow. Simulation results show that the quality of service is optimized in the smart SDN network. The SNN controller can efficiently choose the head of a cluster and its members in the sensor area, as displayed in the quality of service results. More importantly, the proposed model estimates the rate of the packet flow. It contributes to coordinating the available buffer Capacity with a set of active sensing nodes in the network for preventing a buffer overrun. The network control by this model is more accurate than SNN (single spike) due to the proposed training algorithm's spiking power.

REFERENCES

- Alenazi, Mohammed JF, and Egemen K. Cetinkaya, 2020. Resilient placement of SDN controllers exploiting disjoint paths, Transactions on Emerging Telecommunications Technologies 31.2: e3725.
- Alenazi, Mohammed JF, and Egemen K. Cetinkaya, 2020. Resilient placement of SDN controllers exploiting disjoint paths, *Transactions on Emerging Telecommunications Technologies* 31.2: e3725.
- Al-Hubaishi, Mohammed, Celal Çeken, and Ali Al-Shaikhli, 2019. A novel energy-aware routing mechanism for SDN-enabled WSAN." International Journal of Communication Systems 32.17: e3724.
- Al-Shaikhli, Ali Burhan, Celal Ceken, and Mohammed Al-Hubaishi, 2018. WSANFlow: an interface protocol between SDN controller and end devices for SDN-oriented WSAN, Wireless Personal Communications 101.2: 755-773.
- Ammari, Habib M., Adnan Shaout, and Fatme Mustapha2020. Sensing Coverage in Three-Dimensional Space: A Survey, Sensor Technology: Concepts, Methodologies, Tools, and Applications. IGI Global, 989-1015.
- Ateya, Abdelhamied A., et al., 2019. Chaotic salp swarm algorithm for SDN multi-controller networks, *Engineering Science and Technology, an International Journal* 22.4 (2019): 1001-1012.
- Chai, Rong, et al., 2019. Control plane delay minimization-based capacitated controller placement algorithm for SDN, *EURASIP Journal on Wireless Communications and Networking* 2019.1: 1-17.
- Choumas, Kostas, et al., 2019. The SDN control plane challenge for minimum control traffic: distributed or centralized?." *2019 16th IEEE Annual Consumer Communications & Networking Conference (CCNC)*. IEEE.
- Cicioğlu, Murtaza, and Ali Çalhan, 2019. HUBsFLOW: A novel interface protocol for SDN-enabled WBANs, Computer Networks 160: 105-117.
- Cicioğlu, Murtaza, and Ali Çalhan, 2019. HUBsFLOW: A novel interface protocol for SDN-enabled WBANs, *Computer Networks* 160: 105-117.
- Cicioğlu, Murtaza, and Ali Çalhan, 2019. SDN-based wireless body area network routing algorithm for healthcare architecture, Etri Journal 41.4: 452-464.
- Cui, Laizhong, F. Richard Yu, and Qiao Yan, 2016. When big data meets software-defined networking: SDN for big data and big data for SDN, IEEE network 30.1: 58-65.
- Buurman, Ben, et al., 2020. Low-power wide-area networks: Design goals, architecture, suitability to use cases and research challenges, IEEE Access 8: 17179-17220.
- Dvir, Amit, Yoram Haddad, and Aviram Zilberman, 2019. The controller placement problem for wireless SDN, *Wireless Networks* 25.8: 4963-4978.
- Fan, Zifu, et al., 2019. A multi-controller placement strategy based on delay and reliability optimization in SDN, *2019 28th Wireless and Optical Communications Conference (WOCC)*. IEEE.
- Fernandez, Marcial P., 2013. Comparing openflow controller paradigms scalability: Reactive and proactive, 2013 IEEE 27th International Conference on Advanced Information Networking and Applications (AINA). IEEE.
- Gelberger, Alexander, Niv Yemini, and Ran Giladi., 2013. Performance analysis of software-defined networking (SDN), 2013 IEEE 21st International Symposium on Modelling, Analysis and Simulation of Computer and Telecommunication Systems. IEEE.

- Han, Biao, Xiangrui Yang, and Xiaoyan Wang, 2019. Dynamic controller-switch mapping assignment with genetic algorithm for multi-controller SDN." *2019 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)*. IEEE.

- Haque, Israat Tanzeena, and Nael Abu-Ghazaleh, 2016. Wireless software defined networking: A survey and taxonomy." IEEE Communications Surveys & Tutorials 18.4: 2713-2737.

- Hayes, Matthew, 2016. Scalability and Performance Considerations for Traffic Classification in Software-Defined Networks.

- Jagadeesan, Nachikethas A., and Bhaskar Krishnamachari, 2014. Software-defined networking paradigms in wireless networks: A survey, ACM Computing Surveys (CSUR) 47.2: 1-11.

- Kobo, Hlabishi I., Adnan M. Abu-Mahfouz, and Gerhard P. Hancke, 2017. A survey on software-defined wireless sensor networks: Challenges and design requirements, IEEE access 5: 1872-1899.

- Kreutz, Diego, et al., 2014. Software-defined networking: A comprehensive survey, Proceedings of the IEEE 103.1: 14-76.

- Ndiaye, Musa, Gerhard P. Hancke, and Adnan M. Abu-Mahfouz, 2017. Software defined networking for improved wireless sensor network management: A survey." Sensors 17.5: 1031.

- Njoya, Arouna Ndam, et al., 2017. Efficient scalable sensor node placement algorithm for fixed target coverage applications of wireless sensor networks, IET Wireless Sensor Systems 7.2: 44-54.

- Rana, Deepak Singh, Shiv Ashish Dhondiyal, and Sushil Kumar Chamoli, 2019. "Software defined networking (SDN) challenges, issues and solution." *Int J Comput Sci Eng* 7.1: 884-889.

- Ren, Wei, et al., 2018. A novel control plane optimization strategy for important nodes in SDN-IoT networks, IEEE Internet of Things Journal 6.2: 3558-3571.

- Salman, Mustafa Ismael, 2020. Link Failure Recovery for a Large-Scale Video Surveillance System using a Software-Defined Network, Journal of Engineering 26.1: 104-120.

- Shiltagh, Nadia A., and Mahmood Z. Abdullah., 2019. WSN-WCCS: A Wireless Sensor Network Wavelet Curve Ciphering System, Journal of Engineering 25.6: 67-82.

- Tahmasebi, Shirin, et al., 2020. Cuckoo-pc: An evolutionary synchronization-aware placement of SDN controllers for optimizing the network performance in wsns, Sensors 20.11: 3231.

- Njoya, Arouna Ndam, et al., 2017. Efficient scalable sensor node placement algorithm for fixed target coverage applications of wireless sensor networks, *IET Wireless Sensor Systems* 7.2: 44-54.