



Improving Data Aggregation Performance in Wireless Sensor Networks using Software-Defined Networks

Marwa K. Hasan*

Electronic and Control Engineering Department, Technical Engineering College / Kirkuk (TECK), Northern Technical University (NTU), Kirkuk, Iraq

Email: marwa.khaleel@ntu.edu.iq

Abstract

In this paper, we present a novel methodology to improve the performance of collection operations in wireless sensor networks by the application of software-defined networking technology on (SD WISE) platform. The conditions for selecting the grouped nodes in the controller were determined by adjusting the weights of the (Dijkstra) algorithm. The grouped nodes that include the node were determined based on the paths chosen by the algorithm. The SDN-WISE platform supports reading the payload of the packet and not just the header, in addition to the possibility of dealing with a packet depending on another packet, and the flexibility to modify the routing tables to achieve the appropriate rules for the proposed aggregation algorithm. The results show a significant reduction in the energy consumed after applying the novel suggested algorithm.

Received: August 15, 2023 Revised: November 22, 2023 Accepted: April 01, 2024

Keywords: Networks; Wireless sensor networks; Data aggregation, Software-defined networks.

1. Introduction

Wireless sensor networks are one of the most important wireless networks consisting of hundreds of small, low-cost nodes with limited memory, power, and processing. Each node consists of one or more sensors, depending on the need. There are also nodes called aggregated nodes that are responsible for collecting data from peripheral nodes and performing certain processing operations [1]. Also, they are considered a type of decentralized network (ad-hoc) that relies on establishing a connection between two or more nodes [2] without using a central connection point (access-point). Each node in wireless sensor networks has a sensing area, and it can sense events and things within this range, in addition, each node can communicate via a wireless communication interface with other nodes located in the communication range of this node. The information collected about this event is transmitted to the so-called base station through the multi-hop connection. In turn, the database sends the network data via the internet to the application server.

The significant increase in the number of applications required to be implemented in wireless sensor networks has led to a shortage of resources due to the need to deploy several overlapping wireless sensor networks to implement the various applications required. In contrast, they can be implemented in a single network architecture. Also, wireless sensor networks are considered rigid in terms of rules related to external factors (such as business operation and user access) [3].

The connection of full control to networked devices was a major obstacle in implementing the requirements of diverse and multiple applications with the required flexibility, and traditional data aggregation methods in wireless sensor networks suffered from redundancy in transmitted data, which caused an excessive and useless load in the network. It also faced delays that are not suitable for modern applications.

All the above reasons are for the trend to separate the control domain from the data domain or the so-called routing domain, therefore, software-defined networking technology was proposed, which mainly depends on the separation of the two mentioned domains, this means that routing nodes do not communicate with each other to publish routing tables as in traditional networks, but the controller configures these tables. Software-defined networks are considered an emerging, dynamic, manageable, cost-effective, and adaptive architecture, which makes them ideal for high bandwidth and the dynamic nature of today's applications. The selection of the clustered nodes and the clustering mechanism through the rules specified in the controller improves network performance by reducing the power consumption of nodes and prolonging the network's Lifetime.

2. Literature Studies

The researchers proposed [5] a scheme for scheduling collision-free data aggregation in wireless sensor networks in which duty cycles are applied, which significantly reduces the data aggregation delay. The proposed scheme consists of a DTC algorithm and a fast algorithm to compile data with high efficiency. The simulation results show that the data collection time at the sink node was significantly reduced, but this study was applied in networks based on work cycles only.

In [6], the researchers presented a distributed algorithm based on local minimization with the aim of dynamically constructing a routing path in order to reduce data congestion for compression sampling-based clustering in wireless sensor networks. The primary goal is to minimize the traffic in the hybrid data aggregation process while reducing overhead costs. The authors assume that data collection is carried out in cycles by programming the network in the correct form so that no transmission error occurs during the application of the source code scheme. Therefore, the routing path in the data collection process forms a collection tree rooted by the main station (Base station). The node is identified as a collector when the volume of the data associated with it is greater than a specified threshold. This method achieves a very little overhead cost but does not support saving the node power.

In [7], a three-layer big data aggregation architecture was proposed in addition to the priority-based Dynamic. Data aggregation protocol in wireless sensor networks is considered a hybrid method based on the type of application, where tree-based methods are used in non-real-time applications that do not need speed in data delivery, while the cluster method is used in real-time applications. Also, it depends on the selection of active nodes that provide full network coverage. It achieves efficiency in reducing the data processing time and load at the level of big data servers relying on the selection of cluster heads as the nodes closest to the main terminal will cause the power of these nodes to decrease significantly.

In [8], the method of practical offline flow table aggregation (FTTA) and incremental flow table aggregation in Internet-connected nodes (IFFTA) in SDN networks was introduced to solve the problems caused by the introduction of aggregation in the flow table, because data aggregation policies can slow down Table updates and prolong the update duration. This is during which the data level is inconsistent with the control level. In addition to the fact that many errors are expected, this may include access failures, forwarding loops, traffic isolation, and leakage during forwarding.

In [9, 20], the researchers proposed the method of fast protected forwarding (LFR) in SDN networks, to reduce the update of the flow when an error occurs in the local routing path. It aggregates all the error-affected flows into one large flow and a new path is imposed by the SDN microcontroller.

In [10, 18], the focus was on the data streams that were optimally grouped for measurements in SDN networks, and they were able to decompile and measure the most useful data streams to measure each stream directly through the triple addressable content memory (TCAM) entries of the switches. Due to the specific TCAM sizes, its entries were classified into two groups that include optimal grouping of incoming flows for grouping measurements and decompilation of the most useful flows for in each flow. The intelligent stamp SDN-based measurement and (assembly/unassembled) model were used to compile flow measurements to effectively estimate network flows, and an optimal grouping Matrix was introduced to reduce the error of estimating the flow size, which is an effective scheme of the flow grouping matrix under resources constrained by specific TCAM sizes.

In [11,15-17], authors proposed an estimation-based adaptive flow aggregation method to reduce control congestion directly in wireless sensor networks where software-defined networking technology is applied. It is the best method because it facilitates achieving good network capacity with less packet loss and control messages. Through their research, they found that when control messages increase, the available capacity decreases, but since control traffic

and data traffic travel together through the same channel, this problem is difficult to overcome. The flow rate can only be measured when the scalable node is reached.

For programming at the data level, it is always difficult to determine the correct channel capacity for flows, so the researcher proposed an estimation method where the flow is placed in a larger channel and the volume is measured. Then, it is migrated to a small channel but has sufficient channel capacity to transmit the flow.

In [12-14], the authors discussed the features of scalability of open-flow routing tables and dynamic grouping within the network. It is considered one of the best studies of the compilation of control packets in SDWSN networks, where three types of nodes were implemented, including the controller node, the intermediate node, and the terminal node (source node). Aggregation occurs based on a threshold defined at certain nodes, so flow tables need a mechanism capable of performing small calculations on the values coming from the sensors for intermediate nodes to perform aggregation.

It is clear from previous studies that the selection of the aggregation mechanism based on the aggregated nodes clearly affects the energy consumption, in addition, the aggregated nodes are selected so as to achieve full network coverage in addition to relying on the remaining power of the node, as evidenced in the wireless sensor networks, the control burdens borne by the aggregated node, which consumes the node's energy significantly. When implementing the requirements of modern networks of various applications, this is what will be addressed in the application of Software-Defined Networking Technology (SDN).

2.1 The importance of research and its goals

The importance of the research lies in the application of clustering at the node level in wireless sensor networks through the use of the SD-WISE platform, which enables us to apply software-defined networking technology in wireless sensor networks, and this platform is characterized as stateful, i.e. sensor nodes can be programmed as finite state machines, which can be useful to reduce signals between nodes and the controller and allow the package, not just the header, in addition to the possibility of dealing with a package depending on another package. Therefore, the assembly mechanism will be programmed in a separate central controller. We will develop a flow table to apply the appropriate rules of the clustering process to the sensor nodes. This method will enable the selection of clustered nodes in a very efficient and accurate way, with minimal power consumption.

3. Research Methods and Materials

In our research, we used the Cooja 2.7 program, which runs through the Contiki OS operating system and on the Ubuntu Linux kernel. It can be downloaded directly as a live operating system, and the simulation program is installed on it automatically.

The architecture applied to wireless sensor networks using software-defined networking technology (SD-WISE) is based on the separation of the control and data domains. Also, the presence of a controller that has an overview of the entire network topology is characterized as stateful [14].

The network architecture of wireless sensors when applying the SDN-WISE platform is divided into:

An infrastructure layer represents the bottom and consists of routing nodes.

A control layer includes a set of controllers that adjust data paths via the so-called (Southbound Interface) and the most commonly used interface is open flow.

Also, the application layer is represented by the set of applications that entered the controllers to install the appropriate rules on the data paths, and these rules are entered via the northbound Interface [4].

The C and JAVA programming languages were used.

3.1 The proposed algorithm:

The algorithm is based on the following types of nodes:

sensor nodes: sensor nodes are multifunctional, small-sized, low-power, and low-cost smart sensors that are randomly deployed in the network area to obtain information reflecting changes in the surrounding environment. These nodes collect data and transfer it to a static pool when an event is sensed.

Sink node: The sink node is a sensor node equipped with resources and processing capacity that exceeds the normal sensor node, and communicates with the controller to provide it with information about the network status and receive control rules and commands to adjust the work of network nodes.

Controller node: A controller is located in the control domain that provides the network with control commands and rules to perform all operations within the sensing domain based on information provided by the application layer so that the controller has an overview of the entire network.

3.1.1 Algorithm steps

- 1] Randomly spread the nodes
- 2] Determining the condition for selecting an assembly node utilizing the central module in SDN networks

How to set the shortest path to the Collector?

The problem is to choose the clustered node so that the node with the largest number of connections is in this path.

- 3] Selecting the node grouped within the central module in SDN networks and setting the appropriate rules.
- 4] Spreading the rules within the wireless sensor network.
- 5] Apply the assembly process to incoming packages.

Determine the number of neighboring nodes for each node so that these nodes use it as the next jump towards the Collector node [13]:

Our adoption of the sd-WISE platform for the application of software-defined networking technology in wireless sensor networks is an ideal solution to know the status of the network and its updates permanently without interruption, as the SD-wise platform is characterized by being (stateful). So, it depends entirely on knowing the status of network nodes and their changes through Report Messages that are periodically sent from the nodes to the controller and contain information about the node. Therefore, the first step to determine the network topology by the controller will receive Report Messages (report). Table (1) represents the format of the report message.

Table 1: Report

Byte	Name	Description
0-9	Header	SDN-WISE Packet header
10	Distance	Distance from the sink in no. of hops
11	Battery	Battery level. Full = 0xFF, Empty = 0x00
12	Neighbors Size	The no. of nodes in the neighbors' list
13-14	Neighbor Address 1	Address of the 1st neighbor in the list
15	Link Quality 1	Rssi between the 1st neighbor and the source
..-..	Neighbor Address n	Address of the n-th neighbor in the list
..	Link Quality n	Rssi between the n-th neighbor and the source

This message contains a list of nodes that are one jump away from the source node, so that each report packet has a maximum of 35 neighbor nodes. This packet is routed by default to the Collector node of the receiving node and from there to the controller. The Received Signal Strength Indicator RSSI is a quality indicator of the link between the source node and the nodes in the neighbor list. Bytes 0-9 represent the sd-wise header, Figure (1) represents the SD-wise header field [4].



Figure 1:

* Scope defines a group of controllers that have shown interest in the content of the package. The range value is initially set to 0 (by default) but can be adjusted by appropriate entries in the WISE flow table of the sensor node that generates the packet

The U flag is used to mark packages that need to be delivered to the nearest dock*

*The packet field type (Type) is used to distinguish between different message types along with data packets, where the report packet is given Type=2

*The next step ID (Next Hope ID) is the field that must be provided in the array of accepted identifiers in order for the packet to be further processed by the sensor node [14].

Testing the contract selection conditions grouped in the controller:

Node (A), for example, was chosen as a clustered node based on two factors

The cost of the link (Link cost) is defined at the node as the distance between node (A) and the node to which it will be sent divided by the remaining operating power at this node according to the relationship.

Connection cost=distance / remaining power

*A controller was selected that relies on the most famous algorithm to determine the shortest path towards the node of the complex, which is the Dijkstra algorithm.

*The weights are determined as the consumed energy that was set in the previous relationship, so that the paths with lower weights are selected accordingly with less consumed energy i.e. nodes with shorter paths and higher residual energy.

*The distance is known by the controller who has full knowledge of the network topology

*The report message contains the operating power level of the node, which determines the duration of the node's life, and the operating power of the sensor node is calculated by the equation (1) [15]:

$$E_{MCU} = P_{ON}(f_{MCU}) \cdot T_{MCU}(f_{MCU}) \tag{1}$$

Where $P_{ON}(f_{MCU})$ is the power consumed by the node microcontroller that depends on the frequency of the f_{MCU} microcontroller and the wake duration, respectively.

Figure (2) shows the change in the consumed power with a change in frequency [15].

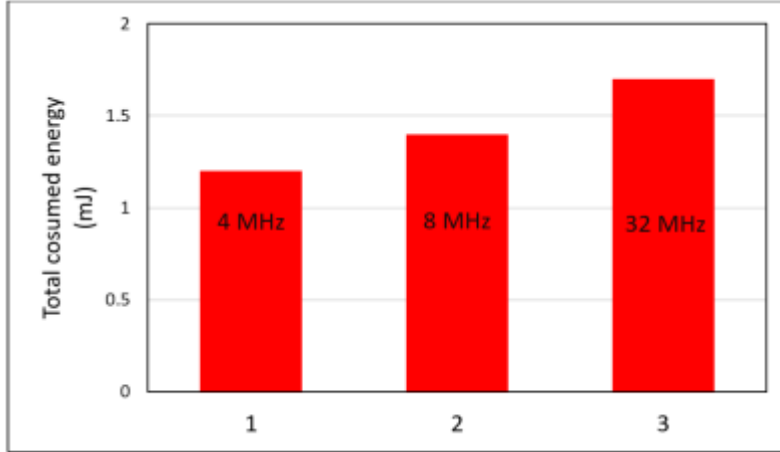


Figure 2: The amount of consumed power with a change in frequency

The number of paths that the node contains:

Based on the link cost relationship that adjusts the weights in the Dijkstra algorithm, which is implemented within the controller, the shortest paths towards the Sink node are determined, so the controller selects the grouped nodes based on the number of paths that this node contains. In other words, the node whose paths are greater than a certain threshold will be considered a grouped node.

In Figure (3), we will give an example of the topology of a wireless sensor network in which the shortest path was chosen and the proposed method of identifying clustered nodes will be illustrated.

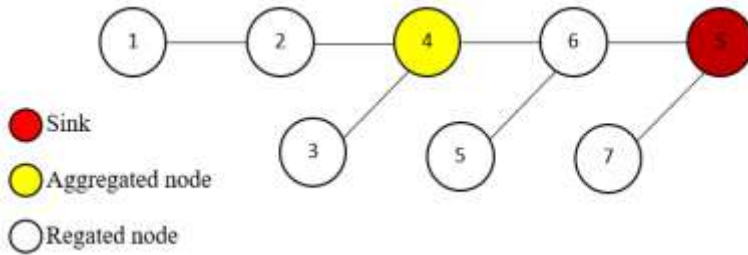


Figure 3: Short path topology of a wireless sensor network

The task is to determine the shortest paths to the node of the complex (S) for each node within the network and, based on Figure (3), the paths are:

From node	To node	The shortest link
1	Sink (S)	(2,4,6,S)
2	Sink (S)	(4,6,S)
3	Sink (S)	(4,6,S)
4	Sink (S)	(6,S)
5	Sink (S)	(6,S)
6	Sink (S)	(-,S)
7	Sink (S)	(-,S)

We have chosen the comparison threshold in the condition of choosing the clustered node as follows:

NodeX is an aggregated node if $LN_x \geq 3$.

Based on the previous condition, node 4 will be selected as a bundled node, and Node 6 will not be selected either, because after choosing node 4 as a bundled node, the number of paths that node 6 contains will be only two. Thus, based on the previous algorithm, the controller selects the grouped node, and then it will publish the rules within the data layer to adjust the mechanism of operation of the grouped nodes.

Publishing rules within the network:

As we explained earlier, we have chosen the SD-WISE platform to apply SDN technology in the wireless sensor network, and therefore the rules will be published by WISE FLOW TABLE, where the flow table consists of matching rules, actions, and statistics. In addition, it was impossible to achieve the desired used traditional solutions that resemble OpenFlow, for the following reasons:

The matching is performed only to verify that the match between a field in the packet header and the specified value. That is, it is not possible to look at data (Data) and relationships of type "higher than", for example, which are not supported.

Therefore, it is impossible to make the handling of a packet depend on the content of another packet in stateless solutions [14]

How to implement FSM

FSM keeps track of the current state, the list of valid state transitions. Each transition can be determined by specifying:

- FromState-the starting State of this transition
- * ToState - the final state of this transition
- * Condition - callable which when returned true means that this transition is valid
- * Callback-an optional callable function that is called when this transition is performed

Figure (4) shows sending two nodes with the addresses (10) and (20) to the aggregated node (Aggr) and from there to the main complex(Sink), to explain the mechanism of execution of cases to determine the work of the aggregated nodes by the flow table of this case, while Figure (5) represents the execution of the scheme of the finished state machine (Finite State Machine), which expresses the assembly process of two sensitive nodes in Figure (4).

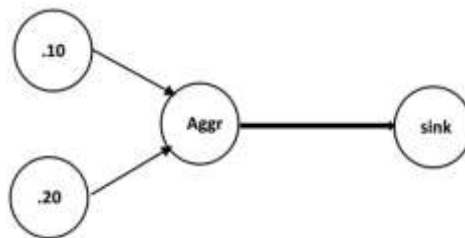


Figure 4: two nodes with the addresses (10) and (20)

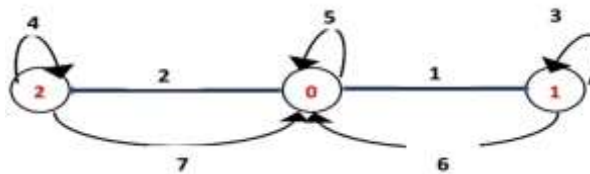


Figure 5: execution of the scheme of the finished state machine

Figure (5) shows the transition process between three states (0,1,2) according to certain grouping conditions set t in the controller related to the reception of packets from the two nodes (.10) and (.20) where the mechanism of action is as follows.

[1] The bundled node receives a message from the node (.10) namely, in the state0, the act is to store the next message and move to state1.

[2] The bundled node receives a message from the node (.20) namely, in the state0, the act is to store the next message and move to state2.

[3] The bundled node receives a message from the node (.10) namely, in the state1 the Act is to send what has been stored and store the new packet.

[4] The bundled node receives a message from the node (.20) namely, in the State2, the act is to send what has been stored and store the new packet.

[5] Sending the message for the aggregated node, which is done periodically when the time imposed by 1sec expires.

[6] The bundled node receives a message from the node (.20) it is in state 1, or sends the message to the clustered node and the act is to send what has been stored and move to State 0.

[7] The bundled node receives a message from the node (.10) it is in state 2, or sends the message to the clustered node and the act is to send what has been stored and move to State 0.

Relationship to energy-consuming transmitter and receiver:

Power consumed in transmission: the power consumed by the node in transmitting (n) Bits for a distance (d):

Table 3: Relationship to energy-consuming transmitter and receiver

Matching rule					Matching rule					Action Statistics					
Op	Size	S	Ad	value	Op	Size	S	Ad	value	Type	M	S	Ad	value	TT L
=	-	1	-	.10	=	-	1	-	0	Store	1	1	-	Data	100
=	-	1	-	.10	=	-	1	-	0	Modify	0	1	-	1	100
=	-	1	-	.20	=	-	1	-	0	Store	1	1	-	Data	80
=	-	1	-	.20	=	-	1	-	0	Modify	0	1	-	2	80
=	-	1	-	.10	=	-	1	-	1	Send to INPP	1	1	-	Agg	100
=	-	1	-	.10	=	-	1	-	1	Store	0	1	-	Data	100
=	-	1	-	.20	=	-	1	-	2	Send to INPP	1	1	-	Agg	80
=	-	1	-	.20	=	-	1	-	2	Store	0	1	-	Data	80
=	-	1	-	.20	=	-	1	-	1	Send to INPP	1	1	-	Agg	80
=	-	1	-	.20	=	-	1	-	1	Modify	0	1	-	0	80
=	-	1	-	.10	=	-	1	-	2	Send to INPP	1	1	-	Agg	100
=	-	1	-	.10	=	-	1	-	2	Modify	0	1	-	0	100

$$e_{tx}(n, d) = e_{elec}n + e_{amp}nd^2 \quad (3)$$

Energy consumed at reception: the energy consumed by the node when receiving (n):

$$e_{rx}(n) = e_{elec}n \quad (4)$$

$$e_{amp} = 10Pj/bit/m^2 \text{ and } e_{elec} = 50nJ/bit$$

The power consumed at the node is therefore the sum of the power consumed at transmission and reception.

The work-flow scenario

We implemented the simulation in the network simulator COOJA 2.7 and the two models of two different networks with a size of 11 nodes for the first network, and 31 nodes for the second network. One complex node is allocated in each network, as these nodes were distributed differently at each execution of the simulation.

The transmission of data packets was made randomly, where 100 data packets were sent to each node within the network and a total power was given for each node equal to the sum of the transmitting and receiving a power of 100 data packets multiplied by the number of quantum nodes within the network to calculate the power as a percentage relative to the quantum energy.

Report messages were sent from the nodes to the controller at each event in the network, which in turn formed rules based on the incoming messages and sent them in the form of routing tables to the nodes.

The following hypotheses were relied on:

- [1] the sink node is fixed
- [2] hold the sensor each time simulating fixed locations
- [3] a single controller gives you an overview of the network
- [4] the nodes have sufficient storage space to store the collected data, the network operation parameters have been adjusted according to the values shown in table (4).

Table 4: the network operation parameters

Value	Parameter
200*200, 100*100	Network space
31, 11	Number of sensor nodes
10pJ/bit/m ²	e_{elec}
50nJ/bit	e_{amp}
%20	Aggregation rate
60 sec	Network stability time
100 packets	The number of data packets drawn from each node
180 sec	Simulation time
47 Byte	The size of the sent packet header
24 byte	The size of the transmitted packet load

4. Results and Discussion

The performance of the proposed algorithm was evaluated by changing the number of nodes and their locations. Moreover, simulations were performed in the COOJA environment, and the results were compared in terms of energy consumed before and after applying the clustering algorithm.

Power consumed in the node: this criterion was developed, which in our article expresses the total power consumed in the transmission and reception based on relationships (3) and (4), where the simulation was performed with a change in the number and locations of nodes (11 nodes, 31 nodes) and randomly distributed in both cases, where we note from the simulation results a significant reduction in the power consumed after assembly (EN) compared with the power consumed before assembly(OEN). It also turns out that increasing the number of nodes did not lead to an increase in power consumption as in traditional networks due to technology.

Through the form it turns out for us to determine the type of node whether it is grouped type=1 or ungrouped Type=0.in some cases, we observe the node moving from grouped to ungrouped due to the receipt of Report Messages on the controller by event, and thus sending the rule based on these events from the controller to the nodes. nxhCT represents the number of nodes that depend on the selected node within its path towards the LINK, then we set a grouping rate of 20% to take the results according to it.

Table 5: results of the first network simulation

EN%	OEN%	Node's type	Node's number
15	25	0	3
15	25	0	11
31	62	1	5
15	25	0	4

Table 6: results of the second network simulation

%EN	%OEN	Node's type	Node's number
12	25	1	19
9	24	0	21
9	18	0	6
9	18	0	2
5	9	0	22
6	12	0	29

5. Conclusions and Recommendations

The results showed that the application of the assembly algorithm using SDN technology significantly improved energy consumption. It turned out through the results that the increase in the number of nodes did not affect the energy conservation after the application of the algorithm, especially in the nodes near the complex, which were suffering from very large energy consumption in traditional wireless sensor networks. Additionally, we note that the proposed algorithm reduced power consumption not only in clustered the nodes but also in the nodes of the entire network due to the reliance on Software-Defined Networking Technology (SDN). In future studies, we propose to apply operations to the data when collected relying on several applications in one network to show the obvious impact of SDN technology.

References

- [1] Mohammad Hossein Shafiabadi, Zohre Ahmadi, Mohammad Reza Esfandyari, Solving the Problem of Target k-Coverage in WSNs Using Fuzzy Clustering Algorithm, Journal of Journal of Intelligent Systems and Internet of Things, Vol. 2 , No. 2 , (2021) : 55-76 (Doi : <https://doi.org/10.54216/JISIoT.020203>)
- [2] Harith Yas, Manal M. Nasir, Quality of Service in Mobile Adhoc Networks with Non-Saturation Conditions, Journal of International Journal of Wireless and Ad Hoc Communication, Vol. 5 , No. 2 , (2022) : 64-76 (Doi : <https://doi.org/10.54216/IJWAC.050205>)
- [3] Tie Luo, Hwee-Pink Tan, and Tony Q. S. Quek. Sensor OpenFlow: Enabling SoftwareDefined Wireless Sensor Networks .2012
- [4] Habib Mostafaei and Michael Menth. Software-Defined Wireless Sensor Networks: A Survey.2018
- [5] Duc Tai Le, Taewoo Lee and Hyunseung Choo. Delay-aware tree construction and scheduling for data aggregation in duty-cycled wireless sensor networks.2018
- [6] T. -Y. Tsai, W.-C. Lan, C. Liu, and M.-T. Sun, "Distributed compressive data aggregation in large-scale wireless sensor networks," J. Adv. Comput. Netw., vol. 1, no. 4, pp. 295–300, Dec. 2013
- [7] Ankita Srivastava, P. K. Mishra, Fuzzy Logic Based Load Balanced Clustering for Network Lifetime Enhancement in WSN, Journal of International Journal of Wireless and Ad Hoc Communication, Vol. 7 , No. 1 , (2023) : 08-17 (Doi : <https://doi.org/10.54216/IJWAC.070101>).
- [8] S. Luo, H. Yu, and L. Li, "Practical flow table aggregation in SDN," Computer Networks, vol. 92, pp. 72-88, 2015.
- [9] X. Zhang, Z. Cheng, R. Lin, L. He, S. Yu, and H. Luo, "Local Fast Reroute With Flow Aggregation in Software Defined Networks," IEEE Communications Letters, vol. 21, no. 4, pp. 785-788, 2017.
- [10] M. Malboubi, L. Wang, C.-N. Chuah, and P. Sharma, "Intelligent sdn based traffic (de) aggregation and

- measurement paradigm (istamp)," 2014.
- [11] Dasha Stablichenkova. (2023). A Study of the Accuracy of Positioning in a Network of Wireless Underwater Sensors Depending on the Genetic Algorithm. *Galoitica: Journal of Mathematical Structures and Applications*, 9 (2), 22-30 (**Doi** : <https://doi.org/10.54216/GJMSA.090203>).
 - [12] R. Friedman and D. Sainz, "An Architecture for SDN Based Sensor Networks," in *Proceedings of the 18th International Conference on Distributed Computing and Networking*, 2017.
 - [13] <https://sdnwiselab.github.io/docs/guides/Core.html>.
 - [14] Laura Galluccio, Sebastiano Milardo†, Giacomo Morabito*, Sergio Palazzo. *SDNWISE: Design, prototyping and experimentation of a stateful SDN solution for Wireless Sensor networks*.2015.
 - [15] Hamza M. Ridha Al-Khafaji, Refed Adnan Jaleel, Design of High-Performance Intelligent WSN based-IoT using Time Synchronized Channel Hopping and Spatial Correlation Model, *Journal of Fusion: Practice and Applications*, Vol. 13 , No. 1 , (2023) : 49-58 (Doi : <https://doi.org/10.54216/FPA.130104>)
 - [16] Barbara Charchekhandra. (2023). The XENVMC communication protocol to improve the efficiency of live migration for co-resident virtual machines. *Pure Mathematics for Theoretical Computer Science*, 1 (1), 32-46 (**Doi** : <https://doi.org/10.54216/PMTCS.010103>)
 - [17] Taoufik Bouguera 1,2,* , Jean-François Diouris 1 , Jean-Jacques Chaillout 2, Randa Jaouadi 3 and Guillaume Andrieux Energy Consumption Model for Sensor Nodes Based on LoRa and LoRaWAN.2018.
 - [18] Zayood, K., " On Novel Public-key Cryptosystem Using MDS Code", *Neoma Journal Of Mathematics and Computer Science*, 2023.
 - [19] Zayood, K., " On A Novel Generalization of The RSA Crypto-System", *Neoma Journal Of Mathematics and Computer Science*, 2023.
 - [20] Suber Halder. Sipra Dasbit. Amira GHosal. A Probability Density Function for EnergyBalanced Lifetime-Enhancing Node Deployment in WSN. 2011