ISSN 1816-6075 (Print), 1818-0523 (Online)
Journal of System and Management Sciences
Vol. 12 (2022) No. 2, pp. 221-235
DOI: 10.33168/JSMS.2022.0210

Wireless Sensor Network Performance Enhancement using Software Defined Networking Principle

Basma R. Mohammad, Abdelazeem A. Abdelsalam and Khaled A. Ali

Electrical Engineering Department, Faculty of Engineering, Suez Canal University, Ismailia 41522, Egypt

basma.refaat@eng.suez.edu.eg

Abstract. A great effort has been dedicated by the networks research community to study Wireless Sensor Network (WSN) to overcome its challenges and optimize its performance. Software-Defined Networking (SDN) has been paid attention thanks to its innovative features, which has been deployed to multiple types of networks instead of traditional network architectures. Recently, the researchers are trying to apply SDN technology on WSN to utilize its advantages such as ease of management, being reconfigurable and ease to revolve using various product vendors. Besides the benefit of delegation, a lot of tasks to the base station (controller) which has unlimited power instead of the sensors and that will affect positively the power consumption and the sensors' lifetime. Low-Energy Adaptive Clustering Hierarchy (LEACH) is a routing protocol for hierarchical networks that prolongs the lifetime of sensors. Its proficiency in reducing power consumption has been proved in a lot of WSN projects. In this paper, the LEACH protocol is integrated with SDN technology on WSN so that it could save sensors' power, reduce overhead and give longer time to the network aside from taking benefits of SDN. The proposed strategy met its objectives, as evidenced by studies that indicate a 33% reduction in energy usage and a 56% increase in sensed data over flat SDN networks. It also provides a significant reduction in energy consumption when compared to the LEACH technique.

Keywords: Clustering protocols, Software defined networking, Wireless sensor network.

1. Introduction

In most applications of wireless sensor networks (WSN), the network is deployed without a predefined structure and left unattended to perform defined tasks like monitoring. Thus, the WSN should be able to self-configure its operation and connectivity. A wireless sensor is a low power battery-operated device that is used to sense a physical quantity.

It is capable of wireless communication, data storage, finite computations and signal processing abilities. All these processes cause rapid power dissipation for sensors and that critical problem pushes the scientific community to find solutions to reduce power dissipation by reducing sensors' missions.

Software defined networking (SDN) technology makes it possible to modify network behavior and deploy new applications and network services in real-time (Kreutz et al., 2017; Liu 2021). It overcomes a lot of traditional network problems. The main idea of SDN is to separate the network into a control plane and a data plane.

The control plane is the brain of the network which maintains a global view of the network. It can communicate with all network elements and control them through its flow rules that are changed according to a specific application. In this way, it is not necessary to configure each switch or router individually. Also, it makes easily deploying different vendors' devices on the same network.

Applying the SDN concept on WSN promises to solve a lot of inherited problems such as rigidity to policy changes and difficulty in managing the network (Kobo et al., 2017). It makes each sensor just simply forward the data based on the specified flow table and rules. All those rules can be easily changed through controllers' reprogramming.

Moreover, all computations will be done at the controller that has infinite power, and that will help the sensors to save their energy for just sensing and sending data only. But there is a big shortcoming that is the channel congestion caused by the number of control packets. As all sensors are dummy sensors, they are waiting for the flow rules from the controller to take any action that will influence negatively the control overhead.

Low-energy adaptive clustering hierarchy (LEACH) protocol has a good advantage in that each node has the same probability to be a cluster head (CH), which makes the energy dissipation of each node relatively balanced, nevertheless, sensors spend energy in choosing cluster heads and advertising packets (Dhawan et al., 2014; Silva et al., 2021). The contributions can be pithiness into few points:

Highlighting the idea of implementing SDN principles on a one hop clustered WSN.

Executing the paradigm on an SDN system invented specially for WSN.

Enhancing the paradigm using other additional algorithms to reach optimum results.

In this paper, the LEACH protocol is combined with SDN technology to be applied on WSN. In that situation, sensors are not having to advertise their location, power level or any of their information to neighbors but they will communicate directly with the cluster head and then to the controller. Also, the missions of choosing CHs and advertising will be incurred by the controller. And that will overcome the weaknesses of each method and ensure a longer lifetime for sensors.

Herein, the structure of the paper summarizing the submitted effort is arranged as follows. section II clarifies the notion of SDN in traditional networks and the innovation of merging the concept of SDN within WSN using different controllers. Also, it states the attempts of clustering wireless sensor networks using the SDN paradigm.

Section III illustrates the proposed scheme of applying the LEACH concept, ContikiMAC radio duty cycling and merged transmitted data (MTD) algorithm on a clustered base one-hop distance SDWSN which contains many homogenous internet of things (IoT) devices.

The simulated methods and the results are discussed in section VI. different comparisons are held between our clustered software defined wireless sensor network (SDWSN) versus the flat IT-SDN network and LEACH network. Moreover, we prove the feasibility of each additional step used in the proposal via graphs.

Our conclusions are summarized in Section VII. It defines the main findings of the study indicating how the research objectives have been achieved. Also, the directions for future work are presented.

2. Related Work

2.1. Concept of SDWSN

The SDN concept in traditional wired networks has been confirmed to give a lot of great benefits that gain the trust of great internet content providers like Facebook, Google, etc. OpenFlow is the first protocol deployed in the SDN paradigm in traditional networks (Mckeown et al., 2008). It is the protocol that enabled the SDN controller to directly interact with the forwarding plane of network devices such as switches and routers, so it makes the network more adaptive to any required changes.

If all elements of the network support OpenFlow protocols, the SDN controller will be enabled to push down changes to the dummy switch/router flow-table allowing network administrators to partition traffic, control flows for optimal performance and start testing new configurations and applications. Also, there are many types of controllers that the most intelligence of the network is centralized such as Open Daylight, POX, Flood Light, etc. The severe benefits and applications that SDN makes it applicable and easy to perform encourage the scientific community to pay attention to deploying the same idea on WSNs.

Whereas WSN has a lot of quandary according to the nature of sensors that have a limited amount of energy, the impossibility of reconfiguration, the difficulty of managing or the hardness of applying multitasking, SDN promises to solve all these problems and covers all its flaws.

Many researchers propose an SDN-based architecture that proposes to centralize network management and enables the running of different applications on a single WSN. The control plane is decoupled from the data plane that runs on the sensor nodes. The centralized controllers mostly use a customized version of OpenFlow to interact with the nodes. The nodes are modified to enable this centralized control of their flow tables. Unfortunately, using SDN controllers and protocols for traditional networks does not trustable to prove the effects of SDN on WSN. Therefore, inventing controllers specialized in SDWSN became indispensable.

Many software-defined wireless sensor network (SDWSN) architecture became usable such as TinySDN (Oliveira et al., 2014), SDN-WISE (Galluccio et al., 2015), IT-SDN (Alves et al., 2017) etc., but still the SDWSN in its infancy stages, it also needs unremitting efforts to be more reliable and applicable. The basic architecture of SDWSN leans on three piers: application plane, control plane and data plane as shown in Fig.1.

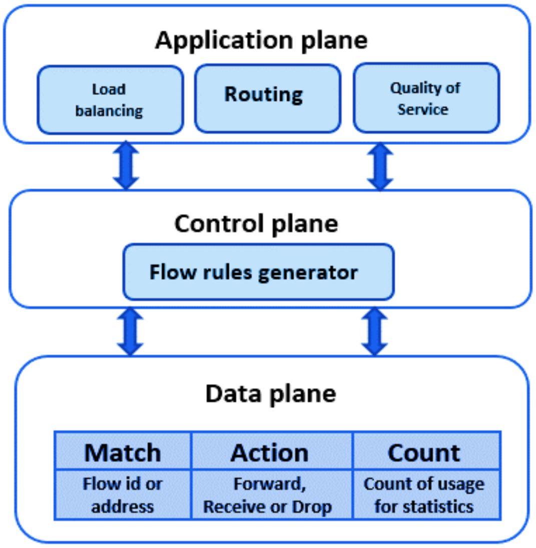


Fig. 1: Software-defined wireless sensor network (SDWSN) overall architecture

A controller is an unlimited-sources unit in comparison with the other components of SDWSN. It has unlimited power, memory, and processing capability, which suits its essential role in the network as it has the whole intelligence of the network. The controller composes a flow-based packet that contains a rule that obliges the sensors with specific actions according to a certain application like load balancing. Every sensor has a flow table that is altered dynamically according to the controller perspective, so it can perform actions like forwarding and receiving or even dropping packets.

SDN paradigm corrects a lot of manifest imperfections of WSN, even so, the energy consumption issue still exists. Also, frequent messages between the control plane and data plane affect badly on the limited bandwidth. These messages are keeping the network robust and fine-grained. So, overhead is a critical issue in applying the SDN paradigm on WSN.

2.1. Clustered SDWSN

Energy consumption optimization has recently become the most essential performance goal in wireless sensor networks. Clustering techniques are particularly effective in extending the life of sensors and integrating such approaches with SDN should do the same.

On the one hand, clustering reduces the power waste caused by communication between the controller and the sensors by directing it solely through the cluster heads. It will aid in the avoidance of possible interference between control and data flows that use the same bandwidth. Clustering also aids in the avoidance of congestion issues and the reduction of overhead by idling the link between the controller and the majority of the sensors.

Using SDN, on the other hand, reduces the tasks that sensors must perform such as sending advertising packets and selecting the cluster head, resulting in an increase in sensor lifetime, in addition to the flexibility in multiple tasking, routing, and vendor variation that the SDN paradigm provides. Many academics are trying to figure out how clustering affects SDWSN.

Olivier et al., (2020) proposed a clustered network in which the cluster head acts as both a base station and a controller. The controller or CH can use this strategy to create configuration parameters, store data, and aggregate the acquired data in the domain, or transfer data to the sink or another CH. To play the part of sensor nodes, they employed the OpenFlow protocol and open Vswitches.

Then, they reveal their results about deploying 1000 IoT devices distributed on five open Vswitches and managed by one SDN controller in an ad-hoc fashion network. The results monitored the CPU usage and memory used by the controller process (Olivier et al., 2020).

An idea of using a two-levels network was presented, the lower level composed of clustered sensors which use Routing Protocol for Low-Power and Lossy Networks (RPL) based on the multi-hop clustering technique (MHC-RPL) (Abdallah 2020). The upper level for large scale managment uses an SDN controller with a Q-routing algorithm that controls a collection of SDN switches. They proved the efficiency of their work at energy saving, packet delivery rate and end-to-end delay.

The authors used the same concept in a dynamic network by placing the controller in the base station as a master node and placing it in the middle of the area (Han et al., 2014). They used a typical network's NOX controller, OpenFlow protocol, and switch. They used OpenFlow switches as cluster heads to construct a load balancing application on WSN utilizing Elman neural networks in each cluster and compared the results to LEACH (Cui et al., 2019).

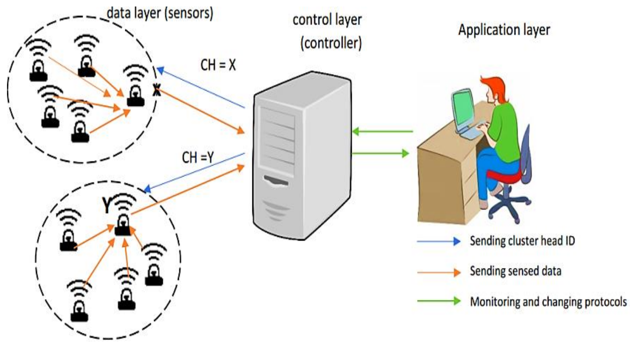


Fig. 2: Clustered software defined wireless sensor network

CLUFLOW splits its SDWSNs into clusters with the fewest cluster border nodes possible (Liu et al., 2019). CH is used to split the clusters, and cluster border nodes are solely used as gateways to other clusters in their system. The number of nodes and messages engaged in flow control inside an SDWSN has decreased, according to the findings. The controller used in that work is IT-SDN (Alves et al., 2017), which is specialized in SDWSN.

IT- SDN is an open software defined sensor network tool and is being used in its most recent upgraded version (Alves et al., 2019). Instead of using typical network methods, this will make the project more realistic.

3. Proposed Scheme

The IoT became not just an invention or new technology fashion, it became an existing part of our daily life (Dunkels 2011). The IoT means the connection via the internet between things, the internet provides fully and remotely control of things. A thing means an Internet-enabled physical and virtual object that possesses intelligence and embedded capabilities to communicate with real-life applications.

The field of IoT covers many subjects from technical issues such as routing protocols to a combination of social and technical problems such as security. Sensing data is an important capability of IoT, So, the IoT network is similar to WSN at this stage. The more usage of smart things, the more data generated within the network. Thus, the data must be better managed and controlled to provide better real-time data delivery. Also, saving energy is a critical issue that must be taken into concern in designing an IoT network.

In light of the LEACH protocol principles, the effect of clustering on an SDWSN is looked in. There are a few clusters of one-hop homogenous sensors. The controller's task would entail electing one cluster head for each cluster, who would be in charge of collecting, gathering, and delivering data to the base station or the controller, as well as cluster members with a single goal of sensing data. All calculations would be the controller's responsibility. This proposed scheme is going to be promising in the field of enhancing IoT networks' performance.

3.1. Clustered SDWSN

The basic idea behind the presented technology is to decouple the controller and data plane using the SDN paradigm. Simultaneously, the sensors are grouped according to LEACH protocol assumptions. The proposed SDWSN concept is illustrated in Fig. 2.

The controller, the network's brain, a set of homogeneous sensors with equal resources, and two flows to exchange packets between those parts are the system's major components. A control flow and a data flow exist in every cluster.

Control flow is the connection between the controller and all network nodes that allows the controller to see the entire network and have complete control over the sensors while circumventing the flow regulations. Data flow is the flow of information between sensors such that data packets can be sent and received without interfering with control flow.

Because the controller has a much larger specification than the sensor nodes, all of the CH election computations are put on the controller server in a round robin manner. The idea of evenly rotating the cluster head with round robin algorithm was proposed and its Simulations result achievements and good energy saving compared to LEACH (Sharma et al., 2015). According to (1), a load balancing method is chosen to distribute the load evenly across all nodes.

$$H = (r \ mod \ n) + n * c \tag{1}$$

The round count is r, the number of candidates in the cluster is n and the order of cluster is c starting with zero. For example, if there is a network with four clusters, the number of nodes within a cluster is 10 nodes and the thirteenth round begins, the corresponding cluster heads at this moment are ID = 3 at the first cluster, ID=13 at the second cluster, ID=23 at the third , cluster ID=33 at the fourth cluster and so on. When the controller selects the CHs, it sends the number of the chosen CH to the cluster control flow, regardless of the election criteria.

Aside from having a global view of the network, creating flows, determining the optimum routes, and defining the sensors' roles according to the application layers, it also has other capabilities. The sensors simply listen to the control flow to determine their role in the next round. The processes carried out by the nodes are summarized in Fig. 3.

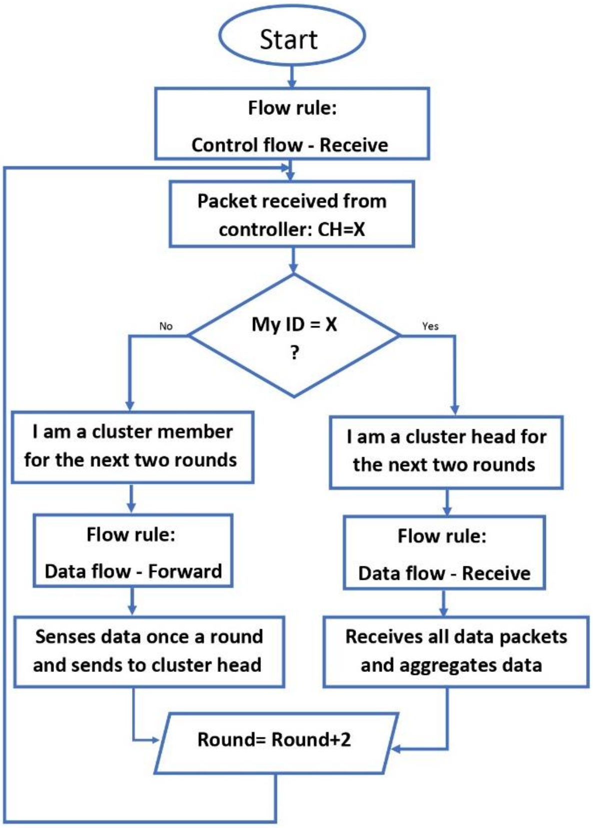


Fig. 3: Clustered SDWSN node behavior

3.2. Using leach in clustered sdwsn

Since it has been published 20 years ago, LEACH has been the focus of many studies forasmuch its simplicity, efficiency and reliability, encouraging the researchers to modify, upgrade and invent new versions of this protocol. The nodes of the cluster elect a CH according to (2).

$$T(n) = \begin{cases} \dfrac{T}{1 - p\left(r \, mod \, \frac{1}{p}\right)} & if \; n \in G \\ 0 & otherwise \end{cases} \tag{2}$$

where r is the current round number, p is the probability for each node to become cluster head and G is the set of nodes that have not been cluster-head in the last 1/p round as presented (Heinzelman et al., 2000).

LEACH has two phases: the cluster formation phase in which the cluster candidates will perform the above equation to decide thier role whether a cluster head or cluster member. The second phase is the steady-state phase when all member node data packets are delivered to the cluster head to collect and aggregate.

In the presented system, the controller will be the only element in the network that has permission to choose the cluster head. The unlimited resources of the controller make changing of the cluster head election algorithms do not affect the performance of the sensors. whether the algorithm is complicated or needs a special specification it will not be a problem since the whole computations will be performed at the controller. Furthermore, it will reduce the time of set up time interval. Set-up time in leach protocol is the time of cluster formation, cluster head calculation and cluster head election and then the cluster head generates a time division multiple access (TDMA) schedule to the cluster. At last, it reduces the broadcast packets which the nodes frequently transmit to advertise themselves at the cluster formation phase. Also, the join-request messages transmitted by the cluster members will no longer exist.

Merged transmitted data LEACH (LEACH-MTD) is a developed method of LEACH that decreases the consumed energy in WSNs by combining two successive rounds with the same cluster head. Instead of replacing the cluster head in every round, including calculations and advertising packets in the cluster formation state, it keeps the same CH for the next round (Badr et al., 2021). The same approach is included in the tested system by keeping the same CH at its position for two consecutive rounds.

3.3. It-sdn controller and network

The IT-SDN controller, which is an open-source SDWSN controller that extends the TinySDN controller, is utilized. It is concerned with delivery rate, delay, control overhead, and energy in order to meet the goals in terms of energy consumption monitoring. Unlike TinySDN, which allows examination of multiple techniques for neighbor and controller discovery, IT- SDN clearly separates the protocols needed to establish Southbound Communication (SB), Neighbor Discovery (ND), and ControllerDiscovery (CD). It runs on the Contiki operating system, which is designed for networked, memory-constrained systems with a focus on low-power wireless IoT devices.

It specifies a control plane architecture that is logically centralized in a single controller or multiple controllers. There is also an architecture of enabled nodes, whose sole purpose is to collect data. Aside from the sink node, which gathers data packets from enabled nodes and subsequently aggregates them, there is a data aggregation node. The suggested technique is compared to a network with one controller, one drain in the center, and enabled nodes in this comparison.

The node behaves similarly to any software-defined sensor. If the arriving packet meets any flow rule in the sensor's flow table, the sensor will take the specified action.

Otherwise, it sends a flow request to the controller, asking it to deliver a suitable action for the sensor. The controller specified the type of flow set up to be used, as well as routing policies and sensor query metrics such as energy.

The ContikiMAC radio duty cycling technique used is adopted because power consumption is the concern (Dunkels 2011). To lower the energy footprint, RDC is used, which employs a power-efficient wake-up mechanism with a set of temporal limitations to allow the sensors to turn off their transceivers for the majority of the time. This method employs a precise timing strategy to ensure that the wake-up mechanism is extremely power-efficient, as well as a phase-lock mechanism to ensure efficient transmissions. As seen in the results, ContikiMAC offers advantages and disadvantages when compared to Carrier-sense multiple access with collision avoidance (CSMA/CA).

Table 1: Simulation parameters

Simulation parameters	
Simulation method	Contiki COOJA simulation
SDWSN Controller	IT-SDN using C++
MAC protocol	CSMA or ContikiMAC
Clustering protocol	MTD-LEACH
Data payload size	10 bytes
Duration of simulation	60 minutes
Max. number of rounds	3500 rounds
Energy consumption parameters [14]	
Processing current consumption	2.33 mA
Receiving current consumption	22.0 mA
Transmission current consumption	21.7 mA
Initial energy	27000 Joule - (2 AA battery)
Voltage	3 V

4. Simulation Results and Discussion

In this section, COOJA simulator is used for simulations. COOJA is a network simulator that allows for the emulation of genuine hardware platforms (Aleksandar et al., 2016). The tested data is compared twice to better understand the crossbred system First; the proposed approach is compared with LEACH. Second, a comparison with the IT-SDN network is conducted. In addition, the two MAC protocols; CSMA and ContikiMAC are compared.

The COOJA simulator has a major flaw: it has limited memory, which limits the simulation length (approximately 60 minutes), but the curves can be extended by applying the linear regression equation. To illustrate energy usage, the average power of all nodes is used. CPU processing, low power mode, transmitting and receiving will all be taken into account in the system.

Equation (3) is used by each node to determine its own energy usage.

$$E = \sum_{i=4}^{i=1} I_i * T_i * V \qquad (3)$$

where I represents the current, V for the voltage, T for the time division and i for the node's state (MEMSIC 2012). The suggested method is implemented in a 2-d virtual area on various numbers of nodes depending on the presented comparisons. Network's nodes have been deployed in a 100 x 100-meter area. The simulation assumption for both systems equals as shown in Table 1

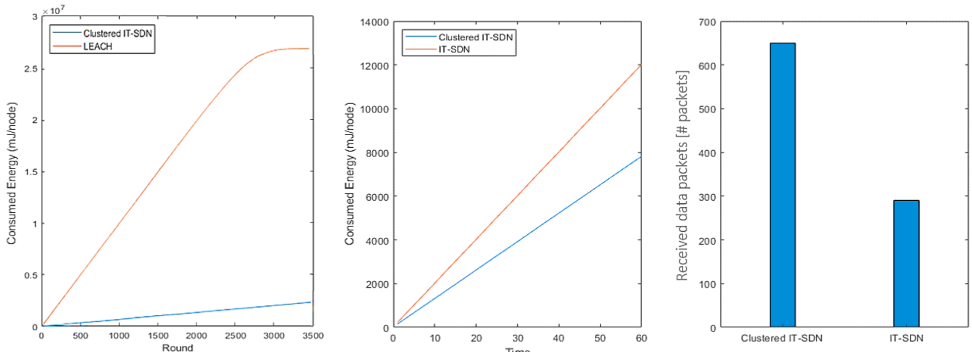
4.1. Clustered sdwsn versus leach

The LEACH methodology is compared to the proposed method. A total of 100 nodes have been separated into four clusters, with each member being only one hop away from the controller. The maximum number of rounds allowed is 3500.

Fig. 4a demonstrates a considerable reduction in the average amount of energy consumed. The CSMA Mac protocol is used, which ensures a high delivery rate. The first dead node in the LEACH network occurs in round 1744. The system continues to function with its finished nodes until the 3500th round is done.

4.2. Clustered sdwsn versus it-sdn

The IT-SDN system network and the proposed clustered SDWSN are compared. In a 100 x 100-meter region, forty nodes have been placed. In Table 1, the simulation assumptions for both systems are the same, and the simulation can last up to 60 minutes. Clustered SDWSN has four equally dispersed clusters, whereas IT-SDN network nodes are spread arbitrarily, with one sink node.



(a) Energy consumption of clustered SDWSN vs LEACH

(b) Energy consumption of clustered SDWSN vs IT-SDN

Fig. 4 The comparisons of the energy consumption and data throughput between clustered SDWSN versus LEACH protocol and IT-SDN

The simulation results shown in Fig. 4b demonstrate a 33 % reduction in average energy consumption. the ContikiMAC protocol is used which reduces energy consumption significantly.

Aside from the positive impact of clustering on energy consumption, it also improves data packet performance by 56 % compared to the standard IT-SDN network as illustrated in Fig. 4c.

4.3. Radio duty cycling

The following comparison is showing the reasons to choose RDC or CSMA methods in the proposed paradigm. Radio duty cycling has a considerable impact on energy usage; it saves roughly 96 % more energy than pure CSMA, which is to be expected that the sensors' transceivers are turned off as will be seen in Fig.5a.

However, there is a flaw: the controller takes longer to recognize the network and orders the nodes to drop data packets until a legitimate path is discovered. This will result in a significant reduction in network data packet throughput. ContikiMAC drops up to 88% of data packets within one hour of duty, as illustrated in Fig. 5b.

4.4. MTD clustered SDWSN

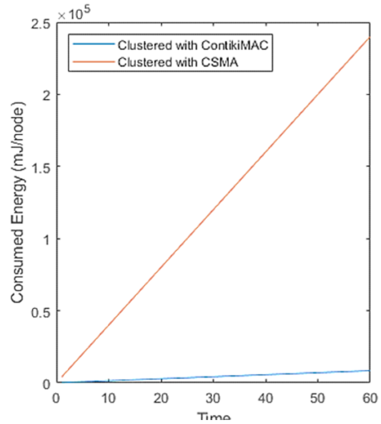
Changing the flow rules to accommodate the new cluster head at the start of each round takes time, resulting in a large number of packets being dropped.

A significant impact on delivered data packets is discovered after using the merged transmitted data technique to the clustering paradigm, as illustrated in Fig. 5d. This great impact overcomes the shortage of using ContikiMAC, so that it is motivation for applying this algorithm in the tested system.

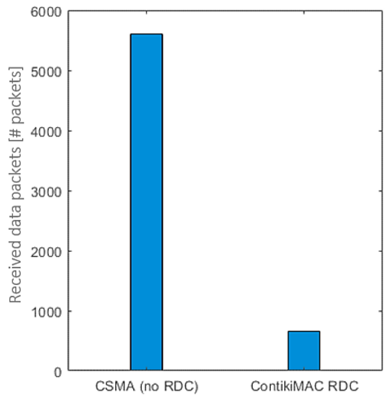
In contrast, it has a minor influence on the energy usage, reducing it by about 8 % when compared to clustered SDWSNs without MTD, as shown in Fig 5c.

5. Conclusions

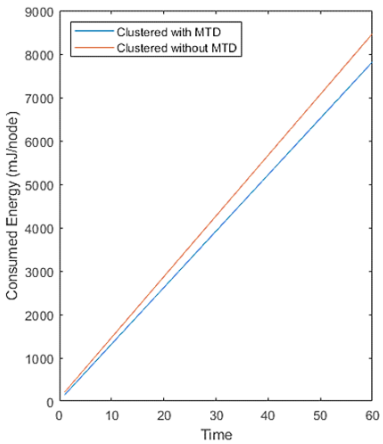
In comparison to both the LEACH protocol and the SDN paradigm, a study of a clustering algorithm on software-defined sensor networks is provided, and it improves the energy consumption behavior of the sensors. It also has good data packet throughput leveraging. Furthermore, it characterizes the tested network with the desired flexibility, scalability, and redundancy which improve the overall functioning of IoT networks. The proposed paradigm has been proven in small IoT networks with up to 50 nodes and a regular data stream, which is ideal for critical applications such as military applications. In the future, this proposed network can be grown by putting it to the test on a real testbed and experimenting with different clustering techniques.



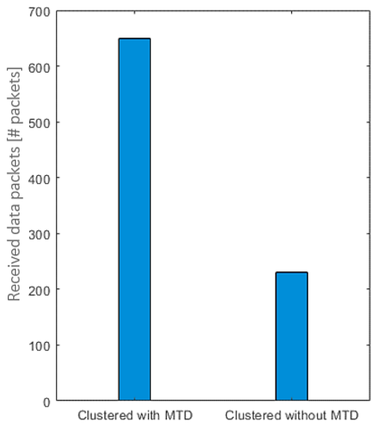
(a) Energy consumption of adopting ContikiMAC vs CSMA



(b) Data throughput of adopting ContikiMAC vs CSMA



(c) Energy consumption after and before using clustered MTD



(d) Data throughput after and before using clustered MTD

Fig. 5: The comparison of the energy consumption and data throughput between ContikiMAC versus CSMA protocols and the impact of using MTD on a Clustered SDWSN

6. References

Abdallah, O. (2020). Energy-efficient clustering and routing algorithm for large-scale SDN-based IoT monitoring, *IEEE International Conference on Communications (ICC), Virtual Conference*, 1-6, 7-11.

Aleksandar, V. & Mileva, A. (2016). Running and testing applications for Contiki OS using Cooja simulator. 279-285.

Alves, R. C. A., Oliveira, D. A. G., Segura, G. A. N. & Margi, C. B. (2017). IT-SDN: Improved architecture for SDWSN. *35th Simpósio Brasileiro Redes Computadores Sistemas Distribuídos*, 1143-1150.

Alves, R. C. A., Oliveira, D. A. G., Nunez Segura, G. A. & Margi, C. B. (2019). The cost of software-defining things: A scalability study of software-defined sensor networks. *IEEE Access*, 7, 115093-115108.

Badr, S., Aly, K. & Ghuniem, A. (2021). A novel improvement in LEACH protocol for decreasing energy loss in wireless sensor networks. *Port-Said Engineering Research Journal*, 25(2), 106-111.

Cui, X., Huang, X., Ma, Y. And Meng, Q. (2019). A load balancing routing mechanism based on SDWSN in Smart City. *Electronics*, 8(3), 273.

De Oliveira, B. T., Margi, C. B. & Gabriel, L. B. (2014). TinySDN: Enabling multiple controllers forsoftware-defined wireless sensor networks. *In Proc. IEEE LATINCOM*, 1-6.

Dhawan, H. And Waraich, S. (2014). A comparative study on LEACH routing protocol and its variants inwireless sensor networks: A survey. *Int. J. Comput. Appl.*, 95(8), 21-27.

Dunkels, A. (2011). The ContikiMAC radio duty cycling protocol. *Swedish Inst. Comput. Sci., Stockholm*, Sweden, Rep. T2011:13.

Galluccio, L., Milardo, S., Morabito, G. & Palazzo, S. (2015). SDN-WISE: Design, prototyping and experimentation of a stateful SDN solution for wireless sensor networks. *IEEE INFOCOM*, 513-521.

Ghasempour, A. (2019). Internet of things in smart grid: Architecture, applications, services, key technologies, and challenges. *Inventions Journal*, 4(1), 1-12.

Han, Z. & Ren, W. A novel wireless sensor networks structure based on the SDN. *Int. J. Distrib. Sens. Networks*, 10(3), 1–8.

Heinzelman, W. R., Chandrakasan, A., Balakrishnan, H. (2000). Energy-efficient communication protocol for wireless microsensor networks. *33rd Annual Hawaii International Conference on System Sciences*, 2, 10.

Kreutz, D., Ramos, F., Veríssimo, P. E., Rothenberg, C. E., Azodolmolky, S. and Uhlig, S. (2015). Software-defined networking: A comprehensive survey. *Proc. IEEE*, 103, 14-76.

Kobo, H. I., Abu-Mahfouz, A. M. And Hancke, G. P. (2017). A survey on software-defined wireless sensornetworks: Challenges and design requirements. *IEEE Access*, 5, 1872-1899.

Liu, L. (2021). Dynamic matrix clustering method based on time series. *Journal of Smart Technology Applications*, 2(1), 9-20, doi:10.21742/JSTA.2021.2.1.02.

Liu, Q., Cheng, L., Alves, R., Ozcelebi, T., Kuipers, F., Xu, G., Lukkien, J. & Chen, S. (2021). Cluster-based flow control in hybrid software-defined wireless sensor networks. *Computer Networks*, 187.

McKeown, N., Anderson, T., Balakrishnan, H., Parulkar, G., Peterson, L., Rexford, J., Shenker, S. & Turner, J. (2008). OpenFlow: Enabling innovation in campus networks. *ACM Computer Communication Review*, 38(2), 69-74.

MEMSIC, Andover, MA, USA. (2012). TelosB Product Details. Accessed: Sep. 2016. [Online]. Available:http://www.memsic.com/userfiles/files/Datasheets/WSN/telosb datasheet.pdf

Olivier, F., Carlos, G. & Florent, N. SDN based architecture for clustered WSN. *9th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, 342-347.

Olivier, F. (2020). An SDN approach to route massive data flows of sensor networks. *International Journal of Communication Systems*, 33(7), e4309.

Sharma, R., Jain, G. & Gupta, S. (2015). Enhanced cluster-head selection using round robin technique in WSN. *International Conference on Communication Networks (ICCN)*, 37-42. doi: 10.1109/ICCN.2015.8.

Silva, A, Luo, L, & Karunasekera, S. (2021). Research on the application of an improved LEACH algorithm in smart home. *International Journal of Smartcare Home*, 1(1), 1-8.