



A Critical Survey on Overhead Control Traffic Reduction Strategies in Software-Defined Wireless Sensor Networking

Simon Atuah Asakipaam

Department of Telecommunication Engineering, Kwame Nkrumah University of Science and Technology, Kumasi, Ghana.

simonasakipaam@gmail.com

Jerry John Kponyo

Department of Telecommunication Engineering, Kwame Nkrumah University of Science and Technology, Kumasi, Ghana.

jjkponyo@gmail.com

Justice Owusu Agyemang

Department of Telecommunication Engineering, Kwame Nkrumah University of Science and Technology, Kumasi, Ghana.

justiceowusuagyemang@gmail.com

Frederick Egyin Appiah-Twum

Department of Telecommunication Engineering, Kwame Nkrumah University of Science and Technology, Kumasi, Ghana.

f.appiah-twum@mail.com

Received: 03 November 2020 / Revised: 25 November 2020 / Accepted: 20 December 2020 / Published: 23 February 2021

Abstract – The rising interest in the Internet of Things has contributed to the rapid deployment of wireless sensor networks (WSNs). However, as a result of the design of the sensor nodes and networks, WSNs exhibit dynamic challenges in mobile and large-scale applications. The nodes are equipped with limited resources and the networks have static architectures. These problems hinder the effective implementation of WSNs. Software-Defined Networking (SDN) is intended to overcome these problems by removing control logic from the data plane and incorporating programmability to allow dynamic management and control of the nodes. Unfortunately, the gains from incorporating SDN into WSNs are diminished by high overhead control traffic, created to discover and maintain a global network topology view, leading to impaired network performance. This paper provides a systematic overview of the software-defined wireless network sensor literature to identify potential gaps and to provide recommendations for future studies.

Index Terms – Software-Defined Wireless Sensor Networks, Topology Discovery Protocol, Minimal Overhead Control Traffic, Energy Consumption, Software-Defined Networking.

1. INTRODUCTION

The increasing growth of interest in the concept of the Internet of Things (IoT) has led to large deployments of wireless sensor networks (WSNs). WSNs are a group of specialized sensor nodes deployed, often in large numbers and in hard-to-reach areas, to lend valuable support for continuous remote monitoring and data collection applications. The sensor nodes sense the physical environment, perform computation on the data, and communicate the data to a desired location [1][2]. The nodes are typically equipped with resources such as one or more sensors, processing units, memory, power supply, and Radio Frequency (RF) transceiver [3]. These resources enable sensor nodes to detect and measure physical conditions such as temperature, motion, vibrations, pollutant levels and gather the data at the desired location. These resources are limited [4], thus, making WSNs unable to operate efficiently, especially in large and dynamic networks. The extent of limitation depends on the applications of the network. For instance, every node consumes energy to perform the task of sensing, data collection, and



SURVEY ARTICLE

communications [5]. Since the nodes are tiny and are operated on batteries, they have limited processing capability, memory size, communication bandwidth, and range [6]. This constrains the complexity and energy requirement of the software and protocols running on them and the topology to be used to deploy the network. This calls for three basic considerations to be made in any practical WSN implementation: (1). Cost- WSN nodes are deployed in large numbers, therefore, the cost of the sensor nodes and the deployment must be very low to make the deployment

financially feasible. (2). Power consumption- Redeployment, recharging sensor nodes, or engineering traffic in large scale networks is very difficult and costly, therefore, WSNs must be designed to consume minimal power during operation [7]. (3). Network management- WSNs are used for a limitless number of applications including traffic monitoring, object tracking, event detection, and other time-critical applications. These applications require flexible support for data collection and network management. The factors that limit the efficient operation of WSNs are summarized in Figure 1.

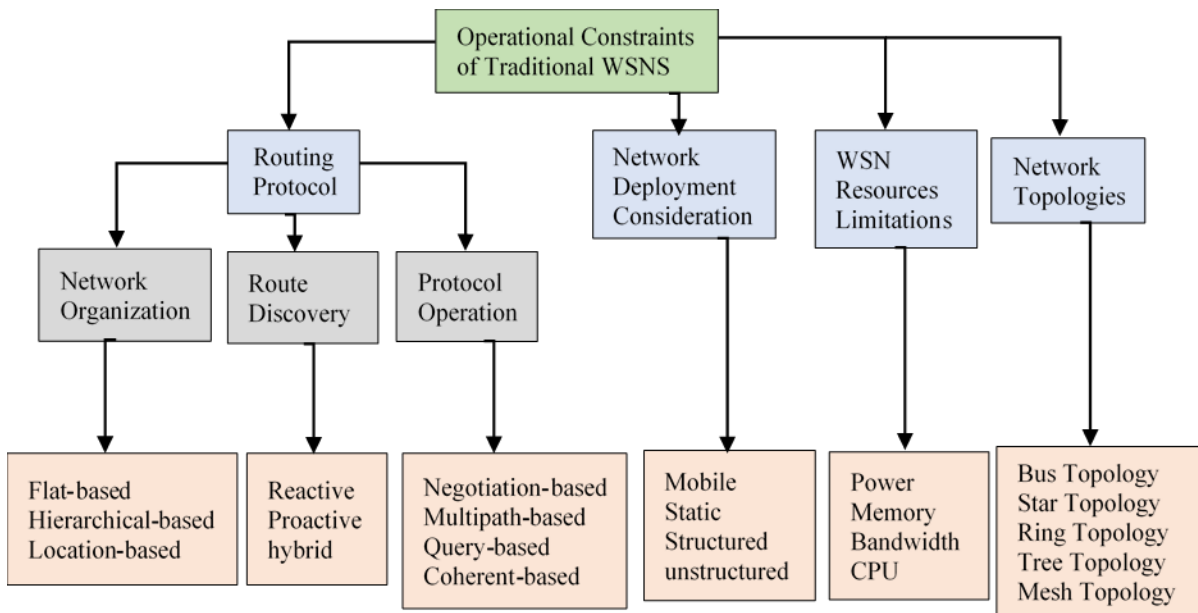


Figure 1 Factors that Affect the Efficient Operation of WSNs

Considerable literature is available regarding efforts to mitigate the impact of these factors on WSNs' lifetime and efficient operation [7] [8]. However, the task of designing innovative techniques [9], protocols [10], and applications [11] to minimize the impact of WSN limitations creates the need for adoption of the Software-Defined Networking (SDN) concept in WSNs. SDN promises to optimize the performance of the limited WSN resources by decoupling the logic of the network from the data plane, introducing programmability [12], and grouping the network into 3 planes: application, control, and data. The application plane is where network policies are defined and administered. These policies are passed to the control plane for evaluation and transformation into routing rules. The data plane generates and forwards traffic using these routing rules and under the supervision of a controller in the control plane. This approach is believed to simplify WSN management and minimize the effects of limited resources [13]. Unfortunately, the gains obtained from applying the SDN concept to WSNs are countered by high overhead control traffic [14], generated as a result of a

collection of topology information to build the network's global topology view and keep the controller constantly updated about it. High overhead control traffic leads to traffic congestion, packet losses, high latency, and inefficient consumption of network resources, thus, looping us back to a shorter network's lifetime and inefficient operation.

Since the concept of SDN and its implementation in WSNs are relatively new, there are only a few available literature review articles ([15], [16], [17]), but none of them focused on finding the relations and gaps in current approaches to minimizing overhead traffic in topology information collection, energy consumption, and operating inefficiency of Software-Defined Wireless Sensor Networks (SDWSN). Thus, this paper presents a detailed literature survey that gives an overview of the SDWSN itself with its architectural design, highlights the issues and shortcomings that emerged in WSN which necessitated the adoption of SDN as well as the imperfections in SDWSN, compares the approaches of existing works of SDWSN regarding the reduction of



SURVEY ARTICLE

overhead control traffic, improvement of network lifetime, and operational efficiency, and provides suggestions for future research. The paper is outlined as follows: Section 2 presents the literature survey. Section 3 evaluates and compares the existing works. Open research issues are presented in section 4, and the paper is concluded in section 5.

2. LITERATURE SURVEY

Recent works in [18][19] have surveyed the possibilities of porting the SDN concept to WSNs [20] to minimize the latter's operational constraints. The authors presented a general architecture of SDWSN, shown in Figure 2, to highlight the novelty that SDN introduces to enhance the efficiency, management, and operations of traditional WSNs [21]. With SDWSN, the network intelligence is moved to a central controller where protocol-specific functions are

introduced in software to optimize the use of generic and commodity hardware and software in both the core and radio access networks. [22]. This approach allows flexible implementation of routing and topology control protocols. It also reduces the computational burden for sensor nodes. An improved network architecture based on SDN to maximize the resource usage of WSN while maintaining secure exchanges of messages among nodes was proposed in [23]. However, as stated in the introduction, topology information collection [24] [25] is the most significant activity of any SDN solution for WSN due to the limitations of WSNs' resources. This, therefore, necessitates the design of efficient Topology Discovery (TD) protocols for SDWSNs that don't reduce the network's throughput, lifespan, and delay and the deployment of energy-efficient networks.

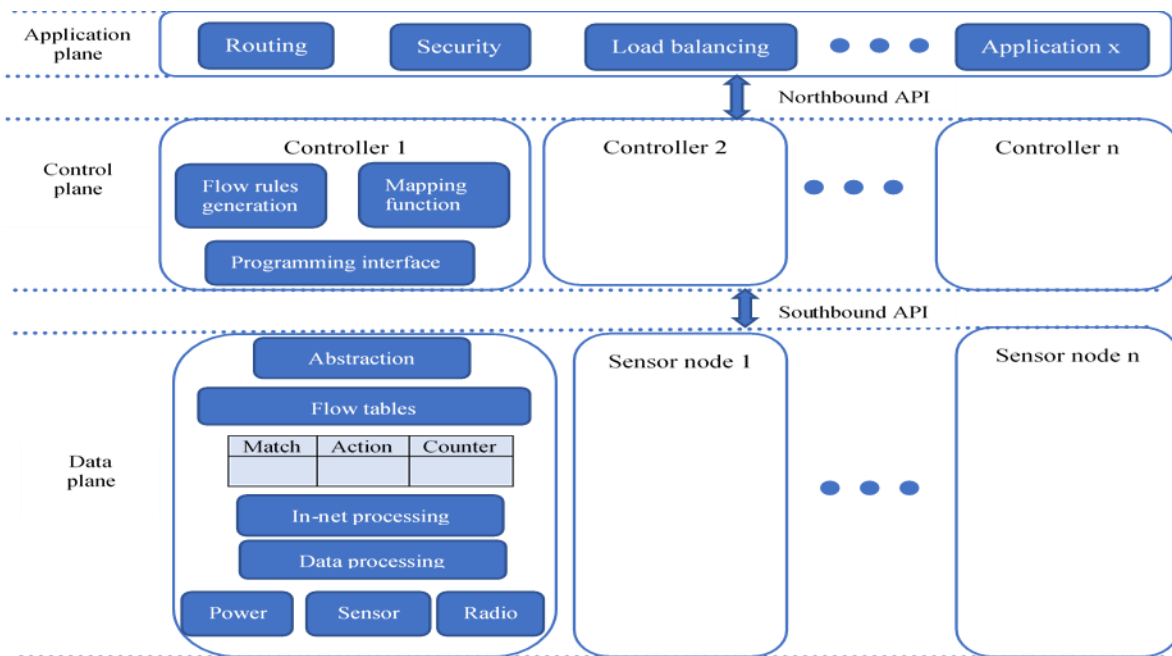


Figure 2 General SDWSN Architecture [26]

Different researchers have adopted different approaches to address the TD issues. Joseph et al [27] discussed how overhead control traffic can be minimized to improve efficiency. In a full literature review of various methods or algorithms, the authors identified the drawbacks, strengths, open issues, and future research directions. Babedi et al [28] proposed a Request For Comments (RFC) 7567 based QoS resource-aware scheme to minimize overhead traffic in highly dynamic and large-scale SDWSNs. The proposed scheme improved data acquisition and network topology collection time as well as network throughput, latency, and packet loss ratio when compared with other similar schemes. The authors in [29] introduced programmable controller logic in the sensor

nodes so that when the nodes receive new packets, they can take decisions without often sending flow requests to the controller. Besides, a new topology management layer was introduced to collect the local topology information. The authors in [30] expanded the forwarding rules to reduce the frequency of requests for flow setup rules. Nasim et al [31] proposed and implemented a "Fuzzy TD protocol" to increase packet delivery ratio, reduce the loss of packets, and conserve the energy [32] of the network to further optimize SDWSN performance. The protocol uses node cost calculated from energy, queue length, and all neighbors of each node to build the flow table and distribute it to all the nodes. Though it ensures a fair distribution of energy consumption, it is



SURVEY ARTICLE

computationally expensive and turns to generate high overhead control traffic. Theodorou et al [33] proposed a separate control channel in SDWSN to minimize the performance issues of high control messages associated with the in-band control channel of SDWSNs. However, this additional network interface and dedicated channel increase the hardware complexity and cost of the sensor nodes. Tomovic et al [34] suggested the coexistence of SDN enabled

and conventional sensor nodes to exploit the SDN enabled nodes to maximize energy usage in the network while reducing overhead control traffic, as not all nodes are managed by the central controller. This approach, however, does not exploit the full benefits of the SDN concept.

Table 1 summarizes the strengths and weaknesses of the existing SDN solutions for WSNs.

Objective	Approach	Merits	Demerits
To prolong WSN lifetime [34]	Intermixed WSN nodes and SDN enable nodes and proposed a new routing protocol	Simultaneously balances load and enhance the network's lifetime	Not all nodes are supervised by the controller, hence, flexibility cannot be achieved
To demonstrate major features of SDWSN [30]	Introduced two novel operations in the SDWSN forwarding mechanism	Reduces latency in the network	The introduction of new rules slows down performance and also increases control overhead
To reduce overhead control traffic [29]	Defined a stateful SDN solution for SDWSNs. Introduce PLC in sensor nodes to enable them to take local decisions	PCL in sensor nodes could offload controller from decisions based on local states and reduce latency and flow processing tasks	The involvement of sensor nodes in decision making drains batteries faster due to extra computation
To improve performance of SDWSN [31] [35]	Applied fuzzy theory in TD and routing decision modules to take the best forwarding decisions	Decreases the number of lost packets and retransmissions	The approach is computationally intensive and contributes greatly to the inefficient utilization of network resources
To address issues of congestion, reliability, and scalability [36]	Investigated the feasibility of installing multiple controllers for SDWSNs	Increases network's robustness and performance	Multiple controllers reduce network efficiency and the optimal positions of the controllers were not defined or proposed
To extend the lifetime of SDWSN [13]	Designed two energy-efficient routing protocols for SDWSNs and introduced an additional layer between the base station and cluster	Additional layer reduces transmission distance and energy consumption	The introduction of an additional layer increases latency



SURVEY ARTICLE

To increase the network's robustness [33]	Introduced a separate control channel to minimize high control traffic	Reduces control overhead and improves performance	Additional network interface and channel increase hardware complexity and cost of sensor nodes
To improve network efficiency and lifetime [37] [38]	Proposed optimal control node selection algorithms based on residual energy	The algorithm ensures load balancing and extension of the network's lifetime	Algorithms are computationally expensive and contribute to high overhead control traffic
To prolong the network's lifetime [39]	Suggested node duty-cycling	This approach ensures efficient energy consumption	Altering the operation of sensor nodes reduces the reliability of the controller's decisions
To improve routing efficiency [40] [41]	Proposed multiple routing algorithms to optimize route selections based on network condition	Changing routing protocols dynamically improves the network's performance	Changing routing protocols in real-time takes a long time, thus increasing delay and loss of packets
To increase the network's robustness against attacks and failures [42] [43]	Introduced multiple controllers to address WSN's perpetual limitations	Increases network's robustness and performance	Multiple controllers reduce network efficiency and tend to generate more control traffic
To reduce energy consumption and increase network lifetime [44]	Implemented an optimized CGSR using PSO with a fuzzy algorithm to balance energy consumption in SDMANET	The protocol ensures load balancing and uniform lifespan of the network	It is computationally expensive and requires extra memory

Table 1 Summary of Literature Review

Antony et al [35] implemented a fuzzy and particle swarm optimization-based energy-efficient clustering and routing algorithm to improve the network's lifespan in a software-defined manet. The proposed algorithm consistently spreads the rate of energy usage of all nodes in the network. The algorithm was analyzed arithmetically, and the authors reported a significant reduction in energy consumption. However, this work was merely an exploration since it did not provide implementation details and actual performance results. Maria et al [44] introduced and implemented an optimized cluster-head switch routing protocol using particle swarm optimization with fuzzy rules to balance energy consumption in a software-defined mobile ad hoc network. The cluster routing algorithm based on particle swarm

optimization was used to perform clustering processes for data transmission, and high energy mobile nodes were identified and chosen for optimal routing of data with the fuzzy rules. The algorithm was reported to outperform conventional cluster-head switch routing protocol when evaluated under network lifetime, packet delivery ratio, and network latency in NS2. The battery level of all mobile nodes is calculated by fuzzy rules to achieve an optimum next-hop mobile node and this information is used to create a network topology dependent on residual energy, power level, reachability, and reliability of the routing node. When the optimal routing node moves or fails, the fuzzy system works to recreate the topology and optimal routing node, placing a high computation burden on the sensor nodes in a highly

SURVEY ARTICLE

dynamic network. Besides, the nodes maintained two tables each for routing and storing neighbor nodes, thus, requiring extra memory.

A resource-efficient algorithm was developed and proposed in [37][38] to improve network efficiency and lifetime using game theory and fork and join adaptive particle swarm optimization. The authors argued that the approach automatically optimizes the number of controllers and their best-suited locations to minimize the exchange of control packets. The approach, though, ensures a fair distribution of energy consumption and load balancing, but it is computationally expensive and contributes to high overhead control traffic. A proposal to alter the activity of the nodes to increase the lifetime of the network was suggested in [39]. However, altering the operation of sensor nodes in SDWSN tends to reduce the reliability of the controller's decisions [45]. This is because the sensor nodes constantly update the central controller about the network's global view. Ali et al [19] presented a study of the various contributors to energy consumption in WSN and the introduction of a miscellaneous form of SDWSN on managing the energy consumption. The authors also outlined the research gaps and suggested an SDN architectural implementation to handle efficient energy consumption. Further studies in [40] have proposed energy-efficient routing algorithms for optimizing route selections. The proposed algorithms considered the remaining energy of all nodes before selecting energy-efficient routes to balance energy usage. Sudip et al [41] suggested a situation-aware SDWSN protocol switching method, using an information routing strategy to improve network efficiency. The proposed method adopts supervised learning algorithms that enable the controller to make decisions in real-time based on network conditions and application-specific requirements. However, changing routing protocols in each sensor node takes a long time, thus increasing delay and loss of packets. The works in [42], [43], and [36] investigated the potential for distributing the control system to minimize congestion and address WSN's perpetual limitations by bringing the controller closer to the sensor nodes. The authors [42] proposed distributed autonomous controllers to monitor and effectively secure the domains to prevent external and internal attacks. To ensure that the change in one controller complies with the network rules, and the change is accessible to the other controllers so that the malfunction in one section of the network will not bring down the entire network, controllers need to continuously synchronize data resulting in high overhead control traffic and inefficient use of network resources.

It can be noted in all the reviewed literature that the processes of topology information collection and update remain relatively the same. All the sensor nodes periodically transmit a TD and update packets over the broadcast channel and install flow tables for the entire network, so while these recent works have made great strides at minimizing the volume of

overhead control traffic, maximizing energy usage, and improving the network's efficiency, it can be observed that more work needs to be done in the topology information collection and update processes.

3. EVALUATION AND COMPARISON

In this section, we evaluate and compare the promising algorithms, suggested to address the TD issue and improve SDWSN performance. The authors in [29] introduced new topology management (TM) layer on top of the controller and dedicated it to collect topology information and execute topology related tasks. The authors also introduced programmable control logic in each sensor node to enable it to take local forwarding decisions and avoid frequent requests for flow setup. This offloads the logically centralized controller from decisions based solely on local states, thus, enabling such decisions to be handled at high speed by the sensor nodes. It also reduces the amount of flow processing activities of the controller, thus, enabling it to respond quickly to more critical requests (requests that cannot be handled by the TM layer) and a minimal number of requests from the data plane. The authors have also included a comprehensive overview of the proposed protocol and evaluated its performance under round trip time, network efficiency, and the controller responsivity at different payload sizes and distances using a physical testbed and Omnet++ and observed that the protocol can potentially reduce overhead, congestion, latency, and packet losses and improve network's performance and lifetime. However, it can be observed that the proposed solution induces high overhead packet exchange between the sensor nodes and the control plane in highly dynamic networks, leading to a substantial decrease in the performance and lifetime of the network.

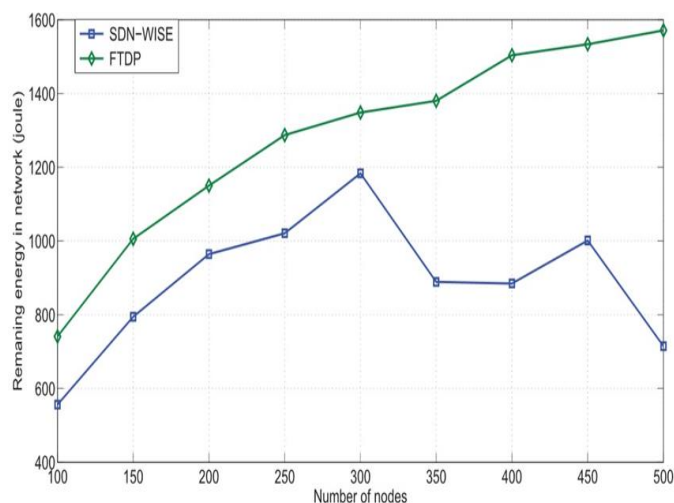


Figure 3 Energy Usage against Node [31]

The authors in [31] attempted to address this shortcoming to optimize the performance of SDWSNs by applying fuzzy

SURVEY ARTICLE

logic to design a load balancing TD and routing decision module. The module considers the cost of nodes to send packets to all other nodes to take the best local forwarding decisions, thus, balancing the battery power consumption of all sensor nodes. In this module, the cost of nodes was computed from the energy of the nodes, queue lengths, and several neighbors. The authors evaluated the energy utilization of the proposed protocol among other parameters in NS2 and reported an improved network performance as shown in Figure 3. However, due to the dynamic nature of SDWSN, this approach is computationally expensive and contributes greatly to the inefficient utilization of network resources even though it ensures load balancing.

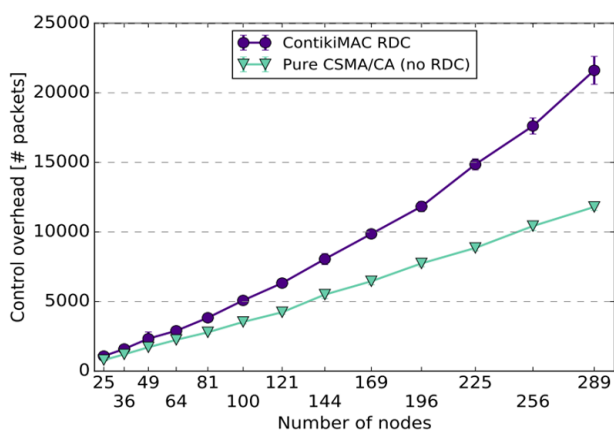


Figure 4 Control Overhead [46]

The authors in [30] expanded the forwarding rules of SDWSNs to potentially minimize the need for sensor nodes to frequently request flow setup rules from the controller since there are more "if-else conditions" to fall back on in forwarding local packets. The approach also provides higher flexibility in defining the flows and controlling the duty-cycling of the nodes' radios to minimize energy usage. However, the additional rules create memory overhead and excessive latency for considering all the conditions before making a forwarding decision. This eventually leads to poor network performance. Theodorou et al [33] have suggested a dedicated control channel to minimize the issue of control channel failure due to data plane links and sensor nodes outages and high overhead control traffic associated with the in-band control channel of SDWSNs. The authors proposed and detailed a framework for reliable and efficient SDWSN to mitigate the high overhead control packets in SDWSN due to the frequent interaction between the data plane and central controller. The proposed solution outperformed RPL for traffic delivery, overhead control traffic, and latency when it was evaluated in CORAL-SDN. But the extra network interfaces increase complexity and costs; the approach is very impractical to accomplish given the nature of sensor nodes. Besides, requests from remote sensor nodes using an out-of-

band control channel may not be able to reach the controller resulting in high packet loss. The authors, in another paper, evaluated the effectiveness of duty cycling to conserve energy in SDWSN and observed that the control overhead increase substantially as shown in Figure 4.

Muhammad et al [47] exploited SDN capabilities to conserve energy usage in WSN and implemented an energy-aware routing algorithm in a real testbed. The authors reported a significantly improved packet delivery ratio of the proposed protocol over AODV due to minimal congestion as shown in Figure 5. The collection of neighbor information for topology view and the approach for controller updates, however, increase overhead cost on sensor nodes and resulted in deteriorated network performance when it was compared with existing SDN protocol.

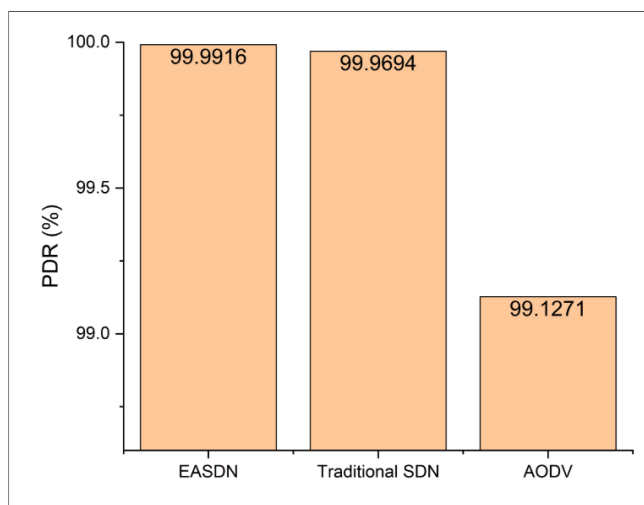


Figure 5 Packet Delivery Ratio of Proposed Protocol (EASDN) Against Two Other Protocols

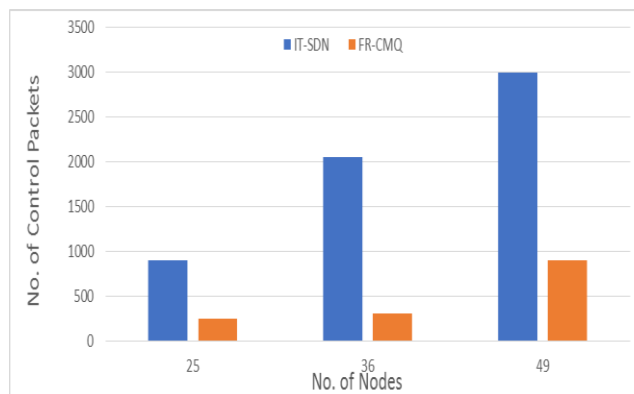


Figure 6 Control Messages Generated for Proposed Algorithm (FR-CMQ) and IT-SDN

Musa et al [48] proposed and implemented a control traffic reduction algorithm that prevents the transmission of duplicate requests for the flow set up from the controller. The



SURVEY ARTICLE

algorithm creates a source-destination pair reference list to keep track of all flow setup requests sent to the control and crosscheck all calls for flow requests. The algorithm was evaluated in it-sdn and the authors reported a substantial reduction in the overhead message (Figure 6) and an increase in inefficient utilization of energy. This algorithm, however,

introduces an extra delay which may lead to traffic congestion and poor network performance; besides, it requires additional memory to store the reference list. Table 2 grades the analyzed approaches' strengths and weaknesses in terms of minimizing overhead control traffic, energy consumption, and network operation inefficiency.

Reference	Overhead Traffic	Energy Consumption	Operating Inefficiency	Comments
[29]	High	Medium	Low	Local decisions tend to be inaccurate resulting in loss of packets and high transmission of flow requests to the controller in dynamic networks
[30]	Medium	Medium	High	Memory overhead and excessive latency results in poor network performance
[31]	Medium	Low	High	The algorithm is computationally expensive, and this contributes greatly to the inefficient utilization of network resources
[33]	Low	Medium	Medium	Sensor nodes farther from the controller experience high loss of packets
[46]	high	Low	Medium	The efficiency of the controller's decisions reduces resulting in repeated flow requests
[47]	high	Low	Medium	Topology information collection and updates use broadcast leading high overhead traffic in the network
[48]	Low	Low	Medium	The algorithm introduces memory overhead and excessive latency

Table 2 Strengths and Weaknesses of Analysed Approaches

4. OPEN RESEARCH ISSUES

It is obvious, from the available literature, that integrating SDN to WSN does not completely address the various limitations of WSNs due to high overhead control traffic, generated as a result of topology discovery and maintenance. Consequently, there are various opportunities for future research efforts to mitigate topology discovery and maintenance challenges and maximize network performance efficiency.

The following research directions would allow researchers to examine various methods and suggest a better way to minimize overhead traffic and optimize network performance in SDWSN.

- The volume of traffic in the network can be reduced significantly by minimizing the exchange of neighbor discovery packets to build the network topology. This can be accomplished by a network architecture that ensures that before the flow table is obtained, a sensor node needs not to be aware of all other sensor nodes in the network. This will minimize broadcasts in the network and reduce the extent of the topology information collection process. However, a question arises as to how nodes unaware of all other nodes in the network will update the paths to the controller to request a flow set-up using the shortest path.
- The distance between the controller and sensor nodes needs to be considered in future works as it contributes significantly to high overhead traffic in the network since excessive delay causes some nodes to send repeated requests or packets. Besides, as the hop count increases for large networks, the intermediate nodes generate flow requests to set up paths, resulting in heavy traffic congestion, packet losses, and inefficient use of sensor node resources.
- Another major issue that needs serious attention is the logical centralization of the controller. In addition to scalability and reliability issues, the controller could become overburdened with numerous flow requests resulting in the network being flooded with several repeated requests. This could potentially increase the overhead control traffic and cause traffic congestion. Optimal ways of distributing the control intelligence that do not contribute to increasing overhead control traffic could be explored in future studies.
- The efficiency of the forwarding rules also needs to be investigated for the possibility of making nodes to share or synchronize the forwarding rules in the flow table of other nodes instead of each node requesting flow rules separately when a nearby node may have the required flow rules at that particular instance.



SURVEY ARTICLE

5. CONCLUSION

In this paper, the major factors that hinder the efficient operation of traditional WSNs were discussed. SDN was identified to offer an optimized solution to mitigate the impact of these constraints. Porting the SDN concept to WSNs simplifies the latter's management and improves its operational efficiency. These gains, however, are countered by high overhead control traffic, generated during topology discovery and maintenance. High overhead control traffic causes traffic congestion, loss of packets, inefficient use of network resources, and poor network performance. Existing literature works have proposed various methods to resolve this issue and improve efficiency in SDWSNs. This paper has discussed the strengths and weaknesses of these methods and future research opportunities. The objective of a future work of this paper will focus on combining the various techniques and methods, suggested in this review, to design a resource-efficient topology discovery, maintenance, and data forwarding protocol for SDWSN. The protocol will enable sensor nodes to operate efficiently under stringent computational and energy constraints requirements.

REFERENCES

- [1] T. M. C. Nguyen, D. B. Hoang, and Z. Chaczko, "Can SDN Technology Be Transported to Software-Defined WSN/IoT?," Proc. - 2016 IEEE Int. Conf. Internet Things; IEEE Green Comput. Commun. IEEE Cyber. Phys. Soc. Comput. IEEE Smart Data, iThings-GreenCom-CPSCom-Smart Data 2016, pp. 234–239, 2017.
- [2] K. M. Modieginyane, B. B. Letswamotse, R. Malekian, and A. M. Abu-Mahfouz, "Software-defined wireless sensor networks application opportunities for efficient network management: A survey," Comput. Electr. Eng., vol. 66, pp. 274–287, 2018.
- [3] N. Sabor, S. Sasaki, M. Abo-Zahhad, and S. M. Ahmed, "A comprehensive survey on hierarchical-based routing protocols for mobile wireless sensor networks: Review, taxonomy, and future directions," Wirel. Commun. Mob. Comput., p. 24, 2017.
- [4] T. Bala, V. Bhatia, S. Kumawat, and V. Jaglan, "A survey: Issues and challenges in wireless sensor network," Int. J. Eng. Technol., vol. 7, no. 2, pp. 53–55, 2018.
- [5] L. K. Ketshabetswe, A. M. Zungeru, M. Mangwala, J. M. Chuma, and B. Sigweni, "Heliyon Communication protocols for wireless sensor networks : A survey and comparison," Heliyon, vol. 5, no. July 2018, p. e01591, 2019.
- [6] R. C. A. Alves, D. A. G. Oliveira, G. C. C. F. Pereira, B. C. Albertini, and C. B. Margi, "WS 3 N: Wireless Secure SDN-Based Communication for Sensor Networks," Secure. Commun. Networks, vol. 2018, 2018.
- [7] M. Razzaq, D. Devi Ningombam, and S. Shin, "Energy-efficient K-means clustering-based routing protocol for WSN using optimal packet size," Int. Conf. Inf. Netw., vol. 2018-Janua, no. 1, pp. 632–635, 2018.
- [8] M. J. McGrath, C. N. Scanaill, M. J. McGrath, and C. N. Scanaill, "Sensor Network Topologies and Design Considerations," in Sensor Technologies, 2013, pp. 79–95.
- [9] S. Kaur and R. N. Mir, "Energy Efficiency Optimization in Wireless Sensor Network Using Proposed Load Balancing Approach," Int. J. Comput. Networks Appl., vol. 3, no. 5, p. 1, Oct. 2016.
- [10] N. Benaouda and A. Lahlouhi, "Ant-based delay-bounded and power-efficient data aggregation in wireless sensor networks," Int. J. Pervasive Comput. Commun., vol. 15, no. 2, pp. 97–119, Jun. 2019.
- [11] J. Prajapati and S. C. Jain, "Machine Learning Techniques and Challenges in Wireless Sensor Networks," in 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT), 2018, no. Icicct, pp. 233–238.
- [12] T. Bakhshi, "State of the art and recent research advances in software-defined networking," Wirel. Commun. Mob. Comput., vol. 2017, p. 36, 2017.
- [13] M. S. Azizi and M. L. Hasnaoui, "Software-defined networking for energy-efficient wireless sensor network," Proc. - 2019 Int. Conf. Adv. Commun. Technol. Networking, CommNet 2019, p. 7, 2019.
- [14] F. F. Jurado-lasso, G. S. Member, and K. Clarke, "Performance Analysis of Software-Defined Multihop Wireless Sensor Networks," IEEE Syst. J., pp. 1–10, 2019.
- [15] T. Tony and L. Hiryanto, "Software-Defined Wireless Sensor Networks: A Systematic Review, Architecture, and Challenges," IOP Conf. Ser. Mater. Sci. Eng., vol. 852, no. 1, p. 012136, Jul. 2020.
- [16] O. P. Cloete, A. M. Abu-Mahfouz, and G. P. Hancke, "A review of wireless sensor network localization based on software-defined networking," Proc. IEEE Int. Conf. Ind. Technol., vol. 2019-February, no. February, pp. 1731–1736, 2019.
- [17] S. Abdipoor, "Software-Defined Wireless Sensor Networks: A Survey," Spec. J. Electron. Comput. Sci., vol. 5, no. 3, pp. 62–64, 2019.
- [18] H. I. Kobo, A. M. Abu-Mahfouz, and G. P. Hancke, "A Survey on Software-Defined Wireless Sensor Networks: Challenges and Design Requirements," IEEE Access, vol. 5, no. c, pp. 1872–1899, 2017.
- [19] N. F. Ali, A. M. Said, K. Nisar, and I. A. Aziz, "A Survey on Software Defined Network Approaches for Achieving Energy Efficiency in Wireless Sensor Network," 2017 IEEE Conf. Wirel. Sensors, vol. 2018-Janua, pp. 28–33, 2017.
- [20] D. Sinh, L. V. Le, B. S. P. Lin, and L. P. Tung, "SDN/NFV - A new approach of deploying network infrastructure for IoT," 2018 27th Wirel. Opt. Commun. Conf. WOCC 2018, pp. 1–5, 2018.
- [21] M. Ndiaye, G. P. Hancke, and A. M. Abu-Mahfouz, "Software-defined networking for improved wireless sensor network management: A survey," Sensors (Switzerland), vol. 17, no. 5, pp. 1–32, 2017.
- [22] J. Dalal et al., "A Survey on Software-Defined Networking," IEEE Commun. Surv. TUTORIALS, vol. 7, no. 2, pp. 72–78, 2018.
- [23] O. Flauzac, C. Javier Gonzalez Santamaria, F. Nolot, and I. Woungang, "An SDN approach to route massive data flow of sensor networks," Int. J. Commun. Syst., vol. 33, no. 7, p. e4309, May 2020.
- [24] C. L-system, "Distributed Learning Fractal Algorithm for Optimizing a Centralized Control Topology of Wireless Sensor Network Based on the Hilbert," Sensors, pp. 1–26, 2019.
- [25] M. Ndiaye, G. P. Hancke, and A. M. Abu-Mahfouz, "Towards Control Message Quenching for SDWSN: A State of the Art Overview," South. Africa Telecommun. Networks Appl. Conf. 2019, pp. 360–364, 2019.
- [26] H. I. Kobo et al., "A Survey on Software-Defined Wireless Sensor Networks: Challenges and Design Requirements," IEEE Access, vol. 5, no. c, pp. 1872–1899, 2017.
- [27] J. Kipongo, T. O. Olwal, and A. M. Abu-Mahfouz, "Topology Discovery Protocol for Software-Defined Wireless Sensor Network: Solutions and Open Issues," IEEE Int. Symp. Ind. Electron., vol. 2018-June, pp. 1282–1287, 2018.
- [28] B. B. Letswamotse, R. Malekian, C. Y. Chen, and K. M. Modieginyane, "Software-defined wireless sensor networks and efficient congestion control," IET Networks, vol. 7, no. 6, pp. 460–464, 2018.
- [29] L. Galluccio, S. Milardo, G. Morabito, and S. Palazzo, "SDN-WISE: Design, prototyping, and experimentation of a stateful SDN solution for Wireless Sensor networks," in 2015 IEEE Conference on Computer Communications (INFOCOM), 2015, vol. 26, pp. 513–521.
- [30] A. Anadiotis et al., "SD-WISE : A Software-Defined Wireless Sensor network ," Elsevier, vol. 159, pp. 84–95, Aug. 2019.
- [31] S. M. Nasim Abdolmaleki, Mahmood Ahmadi, Hadi Tabatabaee Malazi, "Fuzzy topology discovery protocol for SDN-based wireless sensor networks," Elsevier, vol. 79, pp. 54–68, 2017.



SURVEY ARTICLE

- [32] J. Kipongo and E. Esenogho, "Efficient Topology Discovery Protocol for Software-Defined Wireless Sensor Network," *Int. J. Electr. Comput. Eng.*, vol. 9, no. September, p. 19, 2020.
- [33] T. Theodorou and L. Mamatas, "A Versatile Out-of-Band Software-Defined Networking Solution for the Internet of Things," *IEEE Access*, vol. 8, pp. 103710–103733, 2020.
- [34] S. Tomovic and I. Radusinovic, "Performance analysis of a new SDN-based WSN architecture," 2015 23rd Telecommun. Forum Telford, pp. 99–102, 2017.
- [35] R. P. Maria Anthony Sahaya, "Improvement of battery lifetime in the software-defined network using particle swarm optimization based cluster-head gateway switch routing protocol with fuzzy rules," *Comput. Intell.*, p. 22, 2019.
- [36] H. I. Kobo, G. P. Hancke, A. M. Abu-Mahfouz, and G. P. Hancke, "Towards a distributed control system for software-defined Wireless Sensor Networks," *Proc. IECON 2017 - 43rd Annu. Conf. IEEE Ind. Electron. Soc.*, vol. 2017-Janua, pp. 6125–6130, 2017.
- [37] L. Peizhe, W. Muqing, L. Wenxing, and Z. Min, "A Game-Theoretic and Energy-Efficient Algorithm in an Improved Software-Defined Wireless Sensor Network," *IEEE Access*, vol. 5, pp. 13430–13445, 2017.
- [38] D. P. V. Neetesh Kumar, "A Green Routing Algorithm for IoT-Enabled Software-Defined Wireless Sensor Network," *IEEE Sens. J.*, vol. 18, no. 22, p. 12, 2018.
- [39] J. Long and O. Büyüköztürk, "Collaborative duty cycling strategies in energy harvesting sensor networks," *Comput. Civ. Infrastruct. Eng.*, vol. 35, no. 6, pp. 534–548, Jun. 2020.
- [40] M. Masood, M. M. Fouad, S. Seyedzadeh, and I. Glesk, "Energy-Efficient Software Defined Networking Algorithm for Wireless Sensor Networks," *Transp. Res. Procedia*, vol. 40, pp. 1481–1488, 2019.
- [41] S. Misra, S. Bera, A. M. P., S. K. Pal, and M. S. Obaidat, "Situation-Aware Protocol Switching in Software-Defined Wireless Sensor Network Systems," *IEEE Syst. J.*, vol. 12, no. 3, pp. 2353–2360, Sep. 2018.
- [42] B. T. de Oliveira and C. B. Margi, "Distributed control plane architecture for software-defined Wireless Sensor Networks," in 2016 IEEE International Symposium on Consumer Electronics (ISCE), 2016, pp. 85–86.
- [43] O. Flauzac, C. Gonzalez, and F. Nolot, "Developing a Distributed Software Defined Networking Testbed for IoT," *Procedia Comput. Sci.*, vol. 83, pp. 680–684, 2016.
- [44] M. A. Sahaya Sheela and R. Prabakaran, "Improvement of battery lifetime in the software-defined network using particle swarm optimization based cluster-head gateway switch routing protocol with fuzzy rules," *Comput. Intell.*, vol. 36, no. 2, pp. 813–823, May 2020.
- [45] R. C. A. Alves, D. A. G. Oliveira, G. A. Nunez Segura, and C. B. Margi, "The Cost of Software-Defining Things: A Scalability Study of Software-Defined Sensor Networks," *IEEE Access*, vol. 7, pp. 115093–115108, 2019.
- [46] T. Theodorou and L. Mamatas, "Software-defined topology control strategies for the Internet of Things," in 2017 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN), 2017, vol. 2017-Janua, no. November, pp. 236–241.
- [47] M. U. Younus, S. U. Islam, and S. W. Kim, "Proposition and Real-Time Implementation of an Energy-Aware Routing Protocol for a Software-Defined Wireless Sensor Network," *Sensors*, vol. 19, no. 12, p. 2739, Jun. 2019.
- [48] M. Ndiaye, A. M. Abu-Mahfouz, G. P. Hancke, and B. Silva, "Exploring Control-Message Quenching in SDN-based Management of 6LoWPANs," in 2019 IEEE 17th International Conference on Industrial Informatics (INDIN), 2019, vol. 2019-July, pp. 890–983.

Authors



Simon Atuah Asakipaam is a graduate assistant in the Department of Telecommunication Engineering. He received his BSc. degree in Telecommunication Engineering from the Kwame Nkrumah University of Science and Technology, Ghana, in 2017 and is currently pursuing his MPhil. Degree in Telecommunication Engineering from the same university. His areas of interest include Software-Defined Networking, Wireless sensor networks, 5G technology, Artificial Intelligence, etc. Simon's present research topic is "Design of Resource-Efficient Topology Discovery and Data Forwarding Protocol for Software-Defined Wireless Sensor Networks". He is also working on a multidisciplinary project on renewable energy in Sub-Sahara Africa where he is investigating the application of computer vision to design energy-efficient and low-cost tracking systems for Concentrating Solar Power (CSP) Plants.



Jerry John Kponyo is a Senior Lecturer of the Department of Telecommunication Engineering and the current head of the Quality Assurance and Planning Unit of KNUST. His research focuses on applying intelligent systems and computer models in the area of wireless communication to solve real-life problems. He has researched in applying artificial intelligence to Vehicular Ad hoc Networks, where knowledge of the behavior of ants as they search for food was applied to solve a traffic problem. In the area of information dissemination in Vehicular Networks, he has developed a Minimum Spanning Tree clustering algorithm which substantially reduces the traffic overhead as far as information sharing in vehicles is concerned. He has worked on an emergency call routing scheme for disaster management in which he proposes an algorithm that gives priority to distressed calls from emergencies and as a result, reducing the response time in emergencies.



Justice Owusu Agyemang is currently a Ph.D. student at Kwame Nkrumah University of Science and Technology (KNUST), Ghana. He holds a BSc and MPhil in Telecommunication Engineering from the same University. His research areas include the Internet of Things (IoT), Network and Application Security, Artificial Intelligence, and Applied Cryptography. He is a Huawei Certified Network Associate (HCNA) and a Huawei Certified Academy Instructor (HCAI). He is also an ICSI Certified Network Security Specialist (CNSS). His current research work focuses on developing a Cyber-secure Infrastructure for Decentralized Energy Resources.



Frederick Egyin Appiah-Twum received his BSc in Telecommunication Engineering from the Ghana Technology University College in Accra, Ghana. He also received his MPhil in Telecommunication Engineering from KNUST where he is currently a Ph.D. student. His research interests are in traffic management, Smart Systems, Artificial Intelligence, and Wireless Sensor Networks. Frederick is a member of the Ghana Institution of Engineers.