

Methodology for Simulation of IoT-Based WSN Flood Data Collection Infrastructure

Azimah Abdul Ghapar¹, Salman Yussof²

¹College of Computing and Informatics, Universiti Tenaga Nasional, Putrajaya Campus, Selangor, Malaysia

²Institute of Informatics and Computing in Energy, Universiti Tenaga Nasional, Putrajaya Campus, Selangor, Malaysia
Email: azimah@uniten.edu.my, salman@uniten.edu.my

Abstract – Internet of Things (IoT) is a potential technology to be used for data collection tasks in real-world environments. However, due to the difficulty of deploying and testing a real IoT implementation, many researchers end up having to use software simulation to evaluate their proposed techniques. This paper focuses on the use of IoT-based WSN for collecting flood-related data, which would then use by flood-related applications such as flood prediction applications and flood early warning systems. This paper proposed a methodology for simulating the IoT system used for flood data collection. The proposed methodology consists of four main steps which are identifying the flood environment, defining the architecture for flood data collection, simulating the IoT-based WSN flood data collection infrastructure and analyzing the results. The activities for each step are described in detail as to guide other researchers in the same area to adapt the methodology to their research work.

Keywords – Simulation methodology; WSN; IoT; flood data

I. INTRODUCTION

Internet of things (IoT) is increasingly being deployed in many areas such as health, environment, transportation, smart city, environment monitoring and disaster. These IoT deployments are used to collect data from the real-world using things, where a ‘thing’ can be a device or a human. The data collected by things will be processed or/and transferred to a central location. The data collected by the thing will be used by an information system to help in making decision on a specific problem or a situation. Of particular interest to this paper is the use of IoT to assist in making decisions related to flood disaster.

Wireless Sensor Network (WSN) is one of the driver technologies for IoT [1]. In flood disaster, WSN is employed to collect flood data that is used for monitoring and recording the environmental conditions. The flood data collected will be transferred to the central location before it can be used by flood application. WSN consists of a group of spatially dispersed-self-powered sensing devices called nodes. The nodes will collect information such as water level, rainfall measurement, wind speed sensor and flow rate sensor. All these nodes are of low-cost, low-power, resource-constrained multi-functioning sensor nodes, often operating in an unattended, hostile environment, with limited computational and sensing capabilities. Due to these factors, WSN for flood is susceptible to a large number of security attacks that can cause network disruption, access or modify information

that being transferred and attacks on node that will modify the node behaviour.

The research on Wireless Sensor Networks (WSN) and Internet of Things (IoT) often resorts to use a software simulation platform to evaluate their work. The simulation is needed for research purposes because most of the studies cannot be implemented physically. Based on the literature review conducted, there are quite a number of simulations works done by researchers that involve the simulation of data collection in an IoT-based WSN environment. However, none of the studies have provided a detailed description of the methodology that they utilised to simulate their IoT-based WSN flood data collection setting.

The aim of this paper is to propose a methodology in simulating the IoT-based WSN flood data collection environment.

II. IOT-BASED WSN IN FLOOD

The use of IoT-based WSN in collecting flood data provides significant benefits over traditional approaches. IoT-based WSN is the integration of WSNs and the Internet Protocol (IP) that connect the nodes to the Internet. This configuration allows flood data collected by WSN nodes to be transferred to the centralized storage or cloud storage. These data become an input to various flood application systems [2]–[9]. Flood prediction is one of the most critical application that can help to predict flood occurrences and once a flood is predicted to occur, an alert can be sent to the people living in the respective area [12]. An accurate and early flood detection is crucial and has the potential to save lives. The accuracy and the timeliness of the warning depends a lot on the flood-related data collected by the IoT-based WSN system.

WSN is a collection of nodes which function cooperatively to form a decentralized network system. In flood data collection environment, a sensing area will be identified and a group of nodes will be installed in the area. Each sensor in this group has its own function within the WSN. For example, in an IoT system deployed to collect river data, the nodes installed are used to collect data such as water level, sediment level and water velocity. The sensing area will be along the river and the WSN nodes will form a network for each sensing area as shown in Figure 1 below.

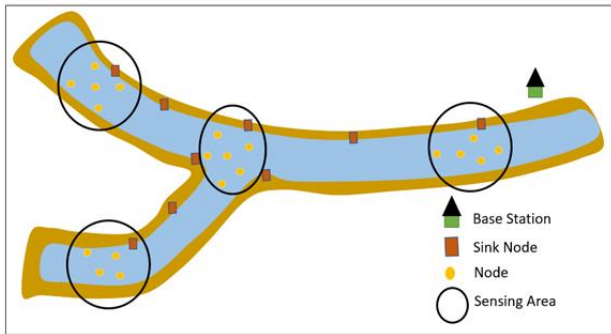


Figure 1. WSN installed at the river

Each sensing area consists of sensor nodes and a sink node. Data collected by the sensor nodes will be transmitted to the sink node and from the sink node the data will be further transmitted through multi-hop techniques to the base station. The base station will send the data to the central server or cloud-based storage to be stored. The communication technologies used for each data transmission may differ according to the distance of each node and whether it is a wired or wireless communication. Zigbee wireless technology can be used for data transmission between sensor nodes to sink node because of the limited transmission range of less than 100 meter [10]. For a wireless transmission that requires a longer distance, which may involve transmission from a sink node to another sink node and transmission from a sink node to a base station, technology such as Low Power Wide Area Network (LPWAN) can be used for data transmission. Data from the base station can be transferred to the central server or cloud-based storage using satellite or cellular technology such as global system for mobile (GSM), 4G, or 5G communication link. The data stored in the storage server would later be retrieved by flood management application. Simulation of the IoT implementation in flood data collection must be done because it can help in planning the infrastructure needed before the infrastructure is actually implemented. Furthermore, by simulating the flood data collection, it will help in evaluating the performance and effectiveness of the WSN in collecting the flood data.

III. FLOOD SIMULATION METHODOLOGIES

WSN has been the main technology used for collecting flood data to be used for forecasting and monitoring flood event. This section discusses existing works that have been done by researchers in simulating the WSN data collection process.

Abdullah [11] developed a flood model and simulated the flood model using OMNET++ simulator. The performance of the flood model is evaluated using parameters such as power consumption, end-to-end delay and throughput to evaluate the availability and scalability of the network. From the simulation done, the author identified the best model for the flood monitoring system for specific location in Saudi Arabia. The author summarizes the simulation steps into two. The first step is to set up the model which consists of the monitored area, number of base stations, number of sensors and sensor installation. The second step is to run the simulation on the

model to be evaluated and the performance of the model is measured based on performance metrics, availability and scalability of the model.

Flood/Drought Forecasting System (FDFS) [12] is developed to help the authorities in generating early warnings for flood or drought. WSN is used in collecting data as input for the system. The author explained the FDFS work starting from defining the WSN architecture for the data collection and using the NS2 simulator to simulate and evaluate the WSN architecture based on energy consumption, memory and processing overhead for a set of sensor nodes involved in the sensing area.

The use of IoT and machine learning based embedded system has been proposed to predict the probability of floods in a river basin [13]. However, the author did not highlight how the simulation was done. NS2 is used to simulate a simplified version of WSN network topology used and implement machine learning model for flood prediction. The author used NS2 simulator in simulating the WSN flood prediction process. The author identified the simulation parameters that will be configured into the experiment before the simulations takes place. These parameters include the size of area coverage, number of sensor nodes involve, packet size, transmission protocol, simulation time, routing protocol and initial energy for each node. Once the simulation is done, MATLAB is used to visualize the result.

The following research works also discussed about the use of WSN in collecting flood data but the detailed description on how the simulation was conducted is not highlighted in their work. Lisangan et al. [14] designed a prototype in detecting flood in the city of Makassar using WSN. The design of the WSN in terms of system infrastructure, communication path between sensor nodes, and the technique used to visualize the data are explained. The author concluded that the data collected by WSN can help in monitoring the network congestion on the wireless public network and flood in Makassar. Kei Hiroi et al. [15] work focuses on early prediction of flash floods in using water level sensor for urban complex water flow (CWF). The result of monitoring the flash flood monitoring is compared to the actual height of the water level. Thekkil et al. [16] developed an early flood detection and control system that captures real time images using WSN. The image is compared with existing data before a warning is generated. The detail of the system implementation is presented and the result of the actual implementation is discussed. Nuhu et al. [17] simulated an energy efficient and accurate flood monitoring system that used 6LoWPAN as a communication protocol. The actual small-scale system is implemented and tested using the emulator. Alfarra et al. [18] proposed a system in detecting occurrence of flood by measuring water level and streamflow of water in rivers. The simulation is focusing on the size of data being transferred to base station. Lukic et al. [19] proposed a system for river monitoring based on WSN. WSN is used for collecting environment data such as flow rate, water level, rainfall and pollution level. All these environmental parameters are collected and measured for a month. Simulation of the river monitoring is done using Matlab. The results show that the environmental parameters were accurately measured

during the observation period using the proposed simulation model.

From the work done by the researchers described above, we can conclude that WSN is one of the main technologies used in collecting flood data. The simulation is done to analyze the performance of the WSN before it can be deployed in real environment. The performance of the WSN is measured using performance parameters. The parameters are identified first before the simulation begins. The simulation is used to test the functionality of an IoT system and to evaluate the developed flood models. The functionality of an IoT system is simulated to evaluate whether the system will function well in the intended sensing environment. The result from the simulation helps in identifying the best flood model that can be implemented in real environment. Although the studies above have shown that simulation is the main practical approach to analyze the deployment of WSN in real environment, none of them has come out with a clear description of methodology that should be used to develop and run the simulation. For this reason, this paper will propose a simulation methodology for simulating the IoT-based WSN flood data collection.

IV. PROPOSED IoT-BASED FLOOD DATA COLLECTION SIMULATION METHODOLOGY

This section will describe the proposed simulation methodology that consists of four main steps. Each of the steps will be described as generic as possible to make it easier to be applied in many applications. Figure 2 shows the four main steps of the proposed flood data collection simulation.

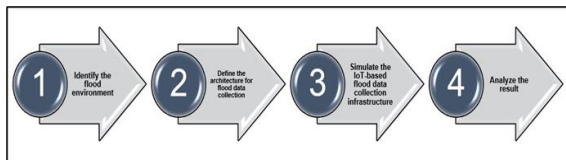


Figure 2. The Four Steps of the Proposed IoT-based WSN Flood Data Collection Simulation

A. Identify the Flood Environment

Flood environment refers to the situation or environment that needs to be simulated. The environment may include river flood, snowmelt flood, or flash flood that can occur at the underground parking area or flood that happens in city area due to blocked drainage. The type of flood environment is essential to be identified because it will affect the next step in this simulation methodology

B. Define the Architecture for Flood Data Collection

Defining the architecture for flood data collection is related to the type of flood environment that is identified in step 1. In this step, the environment detail will be defined and this may include the size of the area to be affected by the flood, the types of data that need to be collected, the method by which the data is to be transmitted and the data storage method. The area size will define the size of the sensing areas to be included in the architecture. Sensing

area refers to the area where the data will be collected. If the architecture consists of more than one sensing area, it needs to consider the method by which the collected data is to be transferred to the central server. This can be done either by transmitting the data through an infrastructure-based network such as the cellular network or by sending the data from one node to another until it arrives at a sink node that will then transmit the data to the central server. Furthermore, a sensor or a group of sensors will be configured in the sensing area according to the types of data that need to be collected.

Referring to Figure 1 in Section II, the configuration may include three types of devices which are sensor node, sink node and base station. Table 1 below shows the three types of nodes and their definition.

Table 1: Device and its function

Device	Functions
Sensor Node (N)	Collect water level, sediment level and water velocity from the river
Sink Node (SN)	Collect data from the sensor node and transfer the data to another sink node or to the base station
Base Station (BS)	Receive data from the Sink Node and transmit it to central server or cloud-based storage

In each sensing area, each sensor node will transfer data to a sink node. Then, the sink node will transfer the data to another sink node if the sink node is far away from the base station. The last sink node will transfer the data to the base station and the base station will transfer it to the central or cloud storage. For each stage of the data transfer, the communication link needs to be configured with a specific wired or wireless communication technology. Table 2 below show comparison of several wireless communication technologies according to their transmission distance, network nodes supported, communication frequency and transmission rate.

Table 2: Comparison of Several Wireless Communication Technologies [20]

Parameter	Wireless Communication Technology			
	Bluetooth	Wi-Fi	ZigBee	LoRa
Transmission distance	10m	100m	75m	20,000m
Network Node	7	255	65535	65535
Communication frequency	2.4GHz	2.4GHz	2.4GHz	433/470/868/915 MHz
Transmission rate	2Mbps	54Mbps	250kbps	37.5kbps

C. Simulate the IoT-Based Flood Data Collection Infrastructure

This step is the primary process in simulating the flood data collection. This step will be further broken up to several sub-processes, as depicted in Figure 3.

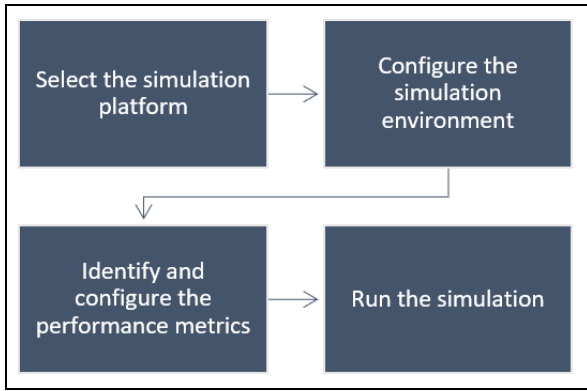


Figure 3. Sub steps for Simulate the IoT-based flood data collection infrastructure

1) *Select the Simulation Platform*

Many types of simulators can be used to simulate an IoT-based environment. These include Cooja, OMNET++, NS3, TOSSIM and Avrora. The choice of platform depends on the objectives of the simulation, the number of WSN nodes to be deployed, the types of language supported by the platform, the support for GUI based simulation and whether the simulator is open-source or commercial software. Table 3 shows the comparison between the simulators.

Table 3: General Comparison of The Simulators [13], [21]

Simulators	Operating System	License	Language Supported	Scalability
Cooja	Linux	Open Source	Java	200 – 500 nodes
OMNET++	Linux , Mac Windows	Academic Public License	C++	>1000 nodes
NS2/NS3	FreeBSD Linux / Windows	Open Source	C++, Python	400 nodes
Tossim	Linux	Open Source	NesC, Python, C++	1000 nodes
Avrora	Linux	Open Source	Java	10000 nodes
Netsim	Windows	Commercial	C, Java	Academic (100) Standard (500) Pro version 100,000

2) *Configure the Simulation Environment*

After identifying the simulation platform, this step will focus on configuring the simulation environment. It will start by configuring the nodes involved in the stated flood environment. In configuring this, the output from step B above can be used as a guide. The nodes need to be configured according to their purpose. The node can be a sensor node where the task is just collecting the data. The node can also be a sink node which is responsible for receiving data from the sensor nodes and then transmit them to the base station. Next, the base station will gather the data and will transfer the data to the central server. The communication links between the nodes also need to be configured. This may include the communication technology used and the frequency of data transmission.

Table 4 shows examples of simulation parameters that are used in configuring the simulation environment.

Table 4: Simulation parameter [21]

Parameter	Details
Area	10000m x 10000m
Nodes	50
Packet Size	512bytes
Transmission protocol	UDP
Simulation time	100sec
Traffic Type	CBR
Antenna Model	Omni directional area
Routing Protocol	AODV
Queue Type	Drop Tail
Propagation Model	Two Ray Ground
Initial energy	100joules

3) *Identify and Configure the Performance Metrics*

Performance metrics are identified according to the objective of the simulation. The performance metrics to be chosen may vary according to what type of information is needed. Performance metrics that can be used in flood simulation are data loss, power consumption, end-to-end delay and throughput [22].

Power consumption is about measuring the sensing node’s battery life. This is related to how frequent the data need to be collected and transferred. The more frequent the activity of the sensor node in collecting and transferring data, the faster the power will be drained. If this happens, it may result in data loss [21].

End-to-end delay refers to the time taken for a single packet to be transmitted across the a network from source to destination. The measurement of end-to-end delay for WSN flood can be defined as the time taken to transfer sensor data from the sensor node to the sink node[23]. Minimizing the end-to-end delay is important for flood prediction applications.

Throughput refers to the rate at which data is received by the receiver that can be measured in bit per second (bps). Since WSN uses wireless transmission, throughput can be affected by multiple factors such as signal strength, number of nodes in the area, or even the weather condition. The number of nodes can affect the throughput because by having more nodes in the same area, there will be overlapping signals which may cause signal interference and collision. This would then result in lower throughput.

Availability of the data and how big the sensing area are important for flood data collection. In the case where the flood data is used for predicting flood occurrence, flood data must be available and must be received on time, so that the warning can be issued earlier. The bigger the sensing area the more flood data can be collected and the more data can be analysed and this hopefully would enable for more accurate flood prediction.

4) *Run the Simulation*

Once the simulation environment is ready a list of experiments needs to be constructed in order to make sure that useful information is collected from the simulation. The experiment constructed must ensure a solid foundation to answer the questions addressed by the research being conducted. Hence, experiments can be conducted by involving performance parameters and using multiple different scenarios. Each experiment may consist of

objectives, a scenario of the events, simulation parameters and performance metrics as explained in the previous steps.

In the simulation software, there are events or parameters that are randomly generated during simulation such as when a node transmits data and how much data is transmitted. This is done using a Random Number Generator (RNG). The RNG uses a pseudo-random number algorithm to produce a list of integers that are seemingly random. The pseudo-random number algorithm is initialized using an integer value, and this integer value is called a seed. Different seed values will give a different list of random numbers. To get a reliable simulation result, an experiment must be conducted multiple times, where each time the simulation runs, a different seed value is used for the RNG so that a unique simulation event is generated. The final result should be averaged from the result of each run. In deciding whether the simulations run for the specific performance parameter is enough or not, confidence level interval needs to be calculated. The confidence level interval can be used to determine whether or not the result is significant based on the configuration done in each experiment. Authors in [24] define confidence interval as an estimated range of values that is likely to include an unknown population parameter. Authors in [25], [26] shows the methods on calculating the confidence interval.

D. Analyze the Result

The result from each of the experiments can be presented according to performance parameter as identified in Step 2 above. The result can be visualized using data visualization techniques. Among the common techniques used to present the results include table, bar chart, scatter plot, correlation matrices and histogram plot.

Analyzing the result is one of the critical steps that need to be documented. The analysis that can be done are on the effect of power consumption with regards to data transfer rate on every node, end-to-end delay experienced while transferring data from source to destination that involve many intermediate nodes and on the percentage of data loss that could happen according to the setting done for each experiment.

V. CONCLUSION

In this paper, we have proposed a 4-step methodology for simulating the IoT infrastructure used for collecting flood-related data. The methodology can be applied to other research work related to flood that uses an IoT technology such as WSN. It is hoped that the methodology can become a reference in simulating and evaluating IoT-related techniques proposed by researchers in the area of flood disaster.

REFERENCES

[1] M. Kocakulak and I. Butun, "An overview of Wireless Sensor Networks towards internet of things," in *2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC)*, 2017, pp. 1–6.

[2] A. Prafanto and E. Budiman, "A Water Level Detection: IoT Platform Based on Wireless Sensor Network," in *2018 2nd East Indonesia Conference on Computer and Information Technology (EIConCIT)*, 2018, pp. 46–49.

[3] S. Bande and V. V. Shete, "Smart flood disaster prediction system using IoT & neural networks," *Proc. 2017 Int. Conf. Smart Technol. Smart Nation, SmartTechCon 2017*, pp. 189–194, 2018.

[4] M. A. M. Sahagun, J. C. Dela Cruz, and R. G. Garcia, "Wireless sensor nodes for flood forecasting using artificial neural network," in *2017 IEEE 9th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM)*, 2017, vol. 2018-Janua, pp. 1–6.

[5] N. A. Z. M. Noar and M. M. Kamal, "The development of smart flood monitoring system using ultrasonic sensor with blynk applications," in *2017 IEEE 4th International Conference on Smart Instrumentation, Measurement and Application (ICSIMA)*, 2017, vol. 2017-Novem, no. November, pp. 1–6.

[6] D. K. C. Nalavade, P. D. S. Shingate, and P. P. C. Patil, "Real Time Rainfall Monitoring and Flood Control System Using Wireless Sensor Network," *SSRN Electron. J.*, 2019.

[7] U. B. Iykekepolo, F. E. Idachaba, and S. I. Popoola, "Early flood detection and monitoring system based on Wireless Sensor Network," *Proc. Int. Conf. Ind. Eng. Oper. Manag.*, vol. 2018, no. SEP, pp. 1381–1394, 2018.

[8] E. Samikwa, "Flood Prediction Using IoT and Artificial Neural Networks with Edge Computing," in *2020 International Conferences on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData) and IEEE Congress on Cybermatics (Cybermatics)*, 2020, pp. 234–240.

[9] O. Mendoza-Cano *et al.*, "Experiments of an IoT-based wireless sensor network for flood monitoring in Colima, Mexico," *J. Hydroinformatics*, vol. 23, no. 3, pp. 385–401, May 2021.

[10] T. Rault, A. Bouabdallah, and Y. Challal, "Energy efficiency in wireless sensor networks: A top-down survey," *Comput. Networks*, vol. 67, pp. 104–122, Jul. 2014.

[11] M. Abdullah, "Simulation of Wireless Sensor Network for Flood Monitoring System," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 8519 LNCS, no. PART 3, Springer Verlag, 2014, pp. 255–264.

[12] F. Khan, S. Memon, I. A. Jokhio, and S. H. Jokhio, "Wireless sensor network based flood/drought forecasting system," in *2015 IEEE SENSORS*, 2015, pp. 1–4.

[13] P. Mitra *et al.*, "Flood forecasting using Internet of

- things and artificial neural networks,” in *2016 IEEE 7th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, 2016, pp. 1–5.
- [14] E. A. Lisangan and S. C. Sumarta, “Proposed Prototype and Simulation of Wireless Smart City: Wireless Sensor Network for Congestion and Flood Detection in Makassar,” in *2018 2nd East Indonesia Conference on Computer and Information Technology (EIconCIT)*, 2018, no. 2016, pp. 72–77.
- [15] K. Hiroi and N. Kawaguchi, “FloodEye: Real-time flash flood prediction system for urban complex water flow,” in *2016 IEEE SENSORS*, 2016, pp. 1–3.
- [16] T. M. Thekkil and N. Prabakaran, “Real-time WSN based early flood detection and control monitoring system,” in *2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICT)*, 2017, vol. 2018-Janua, pp. 1709–1713.
- [17] N. B. K., A. O. T., A. I. A., and A. I. M., “Wireless Sensor Network for Real-time Flood Monitoring Based on 6LoWPAN Communication Standard,” *APTIKOM J. Comput. Sci. Inf. Technol.*, vol. 1, no. 1, pp. 12–22, Mar. 2016.
- [18] H. A. Alfarra and M. H. Alsibai, “A Wireless Smart Sensor Network for Flood Management Optimization,” *Int. J. Eng. Technol. Sci.*, vol. 4, no. 1, pp. 36–42, 2017.
- [19] D. Lukić and B. Lukić, “Application of Wireless Sensor Networks in Flood Detection and River Pollution Monitoring,” *Forum Geogr.*, vol. XVII, no. 2, pp. 118–122, Dec. 2018.
- [20] H. Sun, L. Chen, Y. Chen, C. Li, and Y. Zhou, “Review on Key Technologies of Wireless Monitoring of Pump Group Based on Internet of Things,” in *2019 Prognostics and System Health Management Conference (PHM-Qingdao)*, 2019, pp. 1–5.
- [21] I. Priyadarshinee, K. Sahoo, and S. Mallick, “A model for flood prediction and prevention using wireless sensor network,” *Int. J. Comput. Appl.*, no. May 2015, pp. 22–30, 2016.
- [22] K. P. Naik and U. R. Joshi, “Performance analysis of constrained application protocol using Cooja simulator in Contiki OS,” in *2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICT)*, 2017, vol. 2018-Janua, pp. 547–550.
- [23] O. Azuaje and A. Aguiar, “End-to-End Delay Analysis of a Wireless Sensor Network Using Stochastic Network Calculus,” *IFIP Wirel. Days*, vol. 2019-April, pp. 1–8, 2019.
- [24] M. O. A. Abu-Shawiesh, S. Banik, and B. M. Golam Kibria, “A simulation study on some confidence intervals for the population standard deviation,” *Sort*, vol. 35, no. 2, pp. 83–102, 2011.
- [25] R. C. H. Cheng and W. Holland, “Calculation of confidence intervals for simulation output,” *ACM Trans. Model. Comput. Simul.*, vol. 14, no. 4, pp. 344–362, Oct. 2004.
- [26] M. B. Swift, “A simulation study comparing methods for calculating confidence intervals for directly standardized rates,” *Comput. Stat. Data Anal.*, vol. 54, no. 4, pp. 1103–1108, Apr. 2010.