Evaluation of WSN's Resilience to Challenges in Smart Cities

Sarah Lafi Aljohani*, Mohammed J. F. Alenazi
Department of Computer Engineering, King Saud University, Riyadh 11543, Saudi Arabia.

* Corresponding author. Tel.: +966542337673; email: 437203235@student.ksu.edu.sa
Manuscript submitted April 5, 2020; accepted June 25, 2020.
doi: 10.17706/ijcce.2020.9.4.193-206

Abstract: Smart cities are considered to be one of the most important applications of the IoT notion. Most smart city applications rely fundamentally on ubiquitous sensing, enabled by Wireless Sensor Network (WSN) technologies. These sensor networks are vulnerable to different challenges that cause failures in some parts of the network, which in turn interfere with the availability of network services and weaken the user experience. In this paper, we introduce a graph-theoretic model of wireless sensor networks used in smart cities. Moreover, we present several challenges, such as natural disasters and random failures and evaluate the system's performance in terms of data delivery, end to end delay, and energy consumption. The evaluation results show that fire is the challenge that causes the most damage among the three challenges examined, while random failure has the least effect on network performance. The results also show that the modeled WSN's can cope well with the challenge of random failures.

Key words: IoT, network performance analysis, network resilience, smart city, WSN.

1. Introduction and Motivation

The Internet of Things (IoT) is starting a new wave of the networked computing era. Many objects around us can be incorporated as part of smart systems that collect information and serve us in many ways [1]. The vision of IoT is becoming more realistic over the years due to the exponential increase of devices equipped with networking capabilities. The IoT area of research is gaining more potential, impact, and growth [2]; it promises to change the future of people's lives around the world. It can be used to enhance the quality of life. Smart cities are considered to be one of the most important applications of the notion of IoT [3].

The main goal of smart city initiatives is to improve urban performance by using information and communication technologies in order to provide more efficient services to citizens and to monitor and enhance existing infrastructure [4]. There are a vast number of applications that can be utilized in smart cities to achieve this goal [5]-[8]. These applications include smart streetlights, intelligent traffic management, smart buildings, smart grids, smart water distribution, smart farming, pollution detection, smart surveillance, smart fire control, smart emergency services, and natural disaster alarms. Some of these applications are depicted in Fig. 1.

Most of the smart city applications rely fundamentally on ubiquitous sensing enabled by Wireless Sensor Network (WSN) technologies [5], [8]. These sensor networks are vulnerable to challenges when implemented in real life in smart cities. A challenge is an event that impacts the normal operation of the network [9], [10]. It triggers faults. Failure management is one of the critical concerns of any smart city

development project. Failures can occur during natural challenges, i.e., storms, fires, floods, tornadoes, earthquakes, volcanoes, etc., or they can happen due to system challenges, such as infrastructure breakdown and network unavailability [3]. Challenges may either permanently or temporarily block the reporting of sensor information in smart cities. However, some of the applications provided in smart cities are critical and should maintain their availability during these disasters to monitor the disaster and help in decision-making during the recovery process. The sensory information acquired from the WSN, in this case, might help in controlling the disasters or limiting their danger. That is why, it is extremely vital to work on ensuring that these sensor networks have the resilience to work under challenges.

Modeling smart city networks is an important step to evaluate their performance in the face of challenges. This step can help in studying the best topology design in terms of the number of nodes or node placements. The main contribution of the paper is to model the WSNs of smart cities and to evaluate their resilience against different types of challenges. The network resilience is evaluated based on three metrics: network throughput, end to end delay, and energy consumption.

The rest of the paper is outlined as follows. In Section 2, a brief technical background is presented, and the related work is discussed. After that, Section 3 is dedicated to describing the modeling of the system and challenges. The evaluation details and results are discussed in Section 4. Finally, we conclude our work and discuss future directions in the last section.

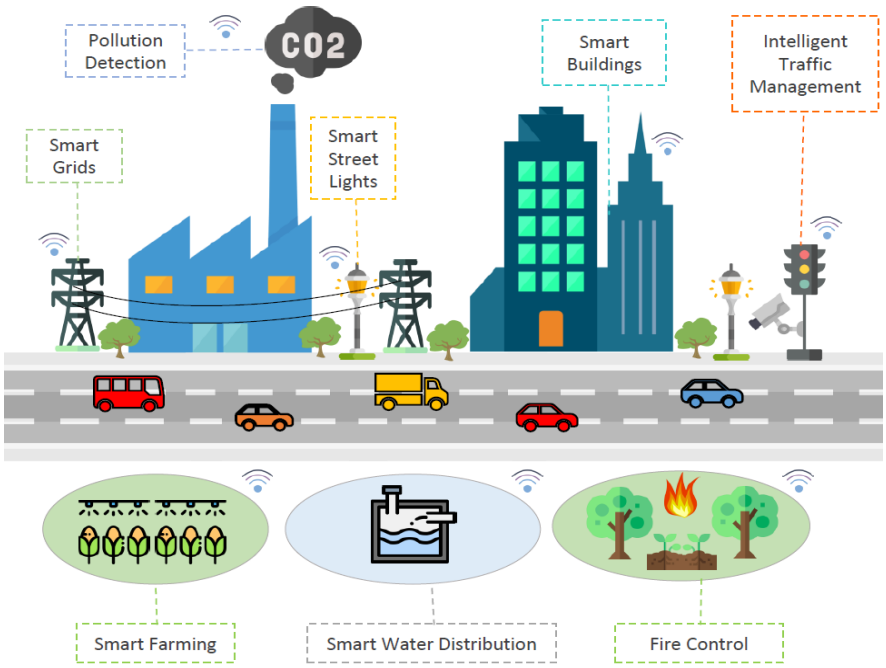

Fig. 1. Smart city applications.

2. Background and Related Work

In this section, we define our notation from a graph-theoretic perspective. Then, we introduce related works and discuss their contributions

2.1. WSN Graph Model

In smart cities, WSN sensor nodes are distributed within a specified area to gather particular information, such as surveillance videos, temperature values, or pollution readings [11], [12]. In general, these nodes can be divided into two groups: sensors and sinks. Sensor nodes are equipped with sensors or cameras to

collect data and send them to a sink. The sink nodes are responsible for gathering the sensed data and process them. Both sensors and sinks are equipped with wireless transceivers, with a certain range, to exchange data with each other. Two sensors can communicate if they are within radio range of each other.

To model this environment using the graph-theoretic approach [13], let $G = \{N, L\}$, where N is a set of WSN nodes and L is a set of links. N is divided into two subsets $N = \{S, T\}$, S represents the set of sensor nodes and T represents the set of sink nodes. L represents the wireless links, such that if the distance between two nodes n_1 and n_2 is less than their transmission range, then the link (n_1, n_2) is in the set L.

2.2. Related Work

The performance evaluation of different communication networks that are commonly used in smart city scenarios has been studied by a number of researchers. In [14], the authors studied the delay that a device may undergo while accessing a Long Term Evolution (LTE) cellular network in the case of a massive number of access requests in real deployment in smart cities. They used the network simulator (ns–3, [15]) to address a Smart City scenario. The results show that if a few hundred smart sensors concurrently require network access, e.g., to report a certain failure, an extremely long delay will be experienced to complete the access procedure; this would be completely unacceptable in critical applications in smart cities, such as in disaster alarms.

In addition, Magrin et al. studied the performance of Long-Range Low-Power Wide Area Networks (LoRa-PWANs) in a typical smart city scenario [16]. They also use the network simulator (ns-3) to simulate a whole network consisting of tens of thousands of end devices. The simulation results show that the network can scale well, achieving packet delivery rates above 95% in the presence of that huge number of end devices. Also, it has been shown that the increase in the number of gateways in the architecture of (LoRa-PWANs) noticeably enhances the coverage and reliability of the uplink.

Other researchers also studied the resilience of different communication networks. The authors of [17] described a methodology to evaluate resilience using a combination of analytic and simulation techniques. They provide a comprehensive framework consisting of a resilience strategy, metrics for quantifying resilience, and evaluation techniques. Also, they added later, in [18], a topology generation and experimental emulation techniques to the framework.

Further, the authors of [19] worked on a simulation-based approach to analyze the effects of perturbations on the normal operation of networks in general. They described how challenges could be categorized, and they presented a framework to evaluate network performance when faced by stationary or evolving challenges. The results show that network performance varies, depending on the type and severity of the challenge applied. Additionally, the authors of [20] used the graph theory to model transportation and communication networks and analyze their resilience in a multilevel framework. They confirmed that dynamic routing helps lighten the impact of perturbations. They also show that adaptive challenges worsen the multilevel network performance more than non-adaptive challenges.

Our work differs from these studies as it targets a different network, the wireless sensor network (WSN), which is recognized as one of the most used technologies in many applications in smart cities [6]. Our work also provides modeling of the network and an analysis of its resilience by evaluating its performance under different challenging scenarios that are commonly faced by networks in real deployment in smart cities. Moreover, it uses three different metrics to evaluate the network's resilience.

3. System Model

In this section, we present our graph-theoretic WSN model in the context of smart cities. Then, we introduce three models to emulate the effect of natural disasters in our WSN model.

3.1. Modeling the Sensor Network

We introduce a system to emulate WSN networks using the graph-theoretic model, discussed in Section 2.1. This model includes several components, as shown in Fig. 2. These components include: system parameters, application data, challenge mode, performance metrics, and tracers. The system parameters are used to define input values for building the WSN such as the area, number of nodes, and transmission range. The application data is used to define the data properties gathered by the sensor, such as the data rate, packet size, and sensing rate. The challenge mode is used to define the type of challenge and the covered area. This component is explained in detail in the next section. The performance metrics component is used to define what metrics are measured during each challenge and to determine the sampling rate. Finally, the tracer component defines the process that is used to obtain the output results.

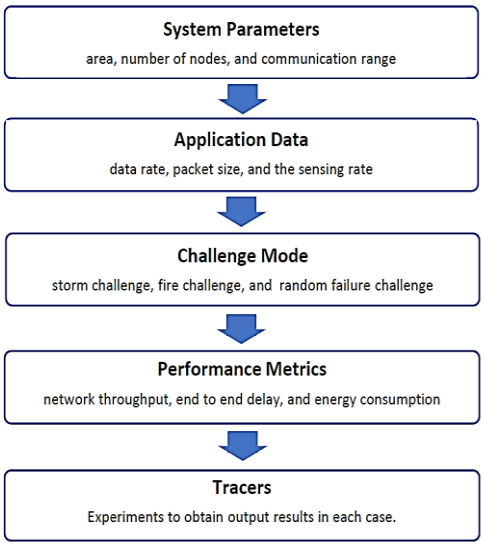

Fig. 2. System model.

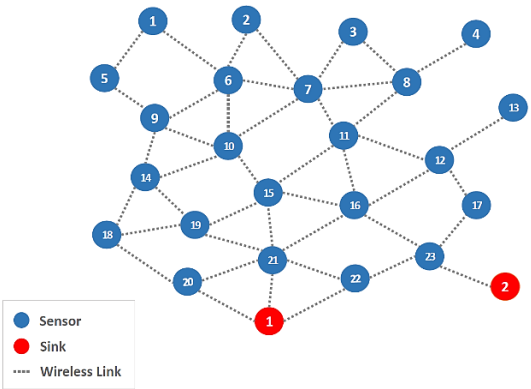

Fig. 3. WSN Graph.

Fig. 3 shows a general, abstract view of the modeled WSN as a graph. The graph consists of a set of nodes and edges that represent sensors and their connections. Sensors are responsible for collecting the sensory information, then reporting it to one of the available sinks. Sinks are responsible for receiving the collected information and processing them to be used later in different smart city applications. The traffic generated by the sensors includes their own sensed information, as well as some forwarded information from other

sensors from different locations. More specifically, a sensor node is not only responsible for sending its own collected sensor information to the nearest sink; it is also responsible for forwarding the sensor information that is collected by other nodes if it falls within the shortest path to the nearest sink.

3.2. Modeling the Challenges

In our system, we target the modeling of three different types of challenges: storm challenge, fire challenge, and random failures. The storm challenge is modeled as a circle with a given radius that specifies a velocity. Nodes that are covered by the circle cannot generate or forward data. We assume that nodes that cease to be covered by the circle can recover their ability to communicate. Fig. 4 shows an example of a storm challenge moving through a WSN. At time 0, the storm challenge starts to enter the sensing area, but only node 20 is affected, as shown in Fig. 4. Therefore, node 20 cannot communicate with other nodes. However, at t=50, the challenge is completely inside the sensing area, covering more sensors. Now, node 20 is uncovered, and other nodes are covered, i.e., nodes 7, 10, 11, and 15. Thus, the sensors of the covered nodes are blocked and cannot communicate with other nodes in the network, as shown in Fig. 4.

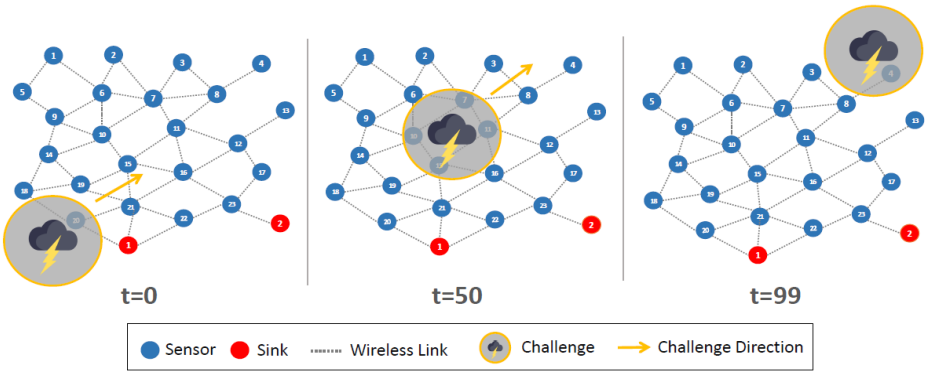

Fig. 4. The storm challenge.

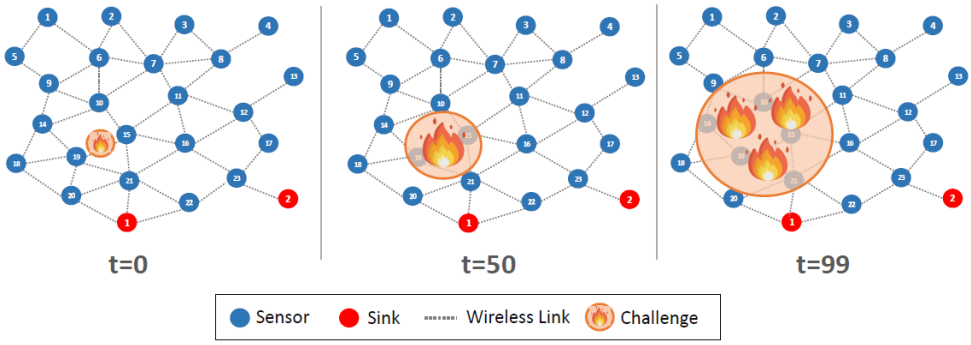

Fig. 5. The fire challenge.

Similarly, the fire challenge has been modeled as a circle, but it keeps expanding with a certain speed, rather than moving. The sensors under the fire challenge are assumed to be completely destroyed by the fire; there is no chance these sensors can work again. Fig. 5. shows how the fire challenge is modeled in detail. At t=0, the fire starts between sensor 15 and sensor 19. At that time, the fire is very small and does not affect any sensor. At t=50, the fire has destroyed sensors 15 and 19, and it starts to hit other sensors. At t=99, about five sensors have already been destroyed.

The random challenge represents failures that occur due to power outage in some regions of the city, or battery shortage when wireless sensors are used or any physical problems in the sensors. To represent this

challenge, it has been assumed that there is a chance of failure at any time. Four scenarios were modeled. The first scenario is when none of the sensors fail. The second scenario is when a single sensor fails for any reason. It has been modeled such that any sensor in the sensor networks has a probability of failing. The third scenario is when any two sensors fail at the same time. All sensors in the sensor network have an equal chance to be one of the two failed sensors. The fourth scenario represents an extreme random challenge when three sensors randomly fail at the same time. Fig. 6 illustrates some of these scenarios.

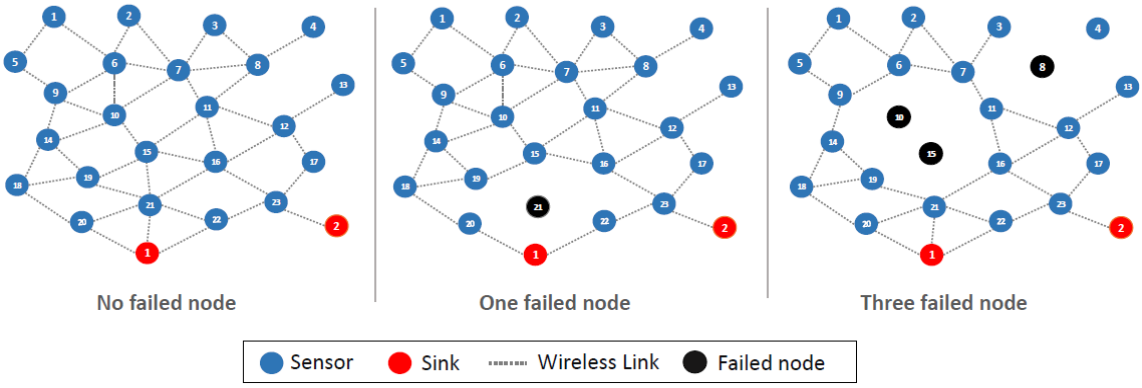

Fig. 6. The random failures challenge.

4. Evaluation

In this section, the details of the implementation and evaluation of the modeled system are presented. Also, the performance metrics that are used to measure the system resilience under challenges are stated and the ways they have been obtained are described. Most importantly, the results of the evaluation of the system under the different types of challenges are illustrated and discussed.

4.1. Environmental Setup

Table 1. The Modeling Parameters

Parameter	Value
Sensing Area Length	1000 m
Sensing Area Width	1000 m
Storm Challenge Radius	280 m
Storm Challenge Speed	10 m/s
Fire Challenge Expanding rate	2 m/s
Random Challenge Probability	0% to 10%
Number of Sensors	24
Number of Sinks	2
Communication Range	300 m

Python programming language is used to implement and evaluate the system. NetworkX [13] library is utilized to create and process graphs. In addition, Shapely library is used to model geometric objects for geometrical modeling and analysis [21]. The sensing area has been assumed to be a 2D square of 1000 by 1000 meters and the challenge is assumed to be a circle with a radius of 280 m. At time 0, the storm challenge is centered at the point (0,0) and it moves towards the top right of the area at a speed of 10 m/s until the challenge center reaches the point (1000,1000). The fire challenge starts at point (300,400). It

expands at a rate of 2 m/s. For the random challenge, it has been assumed that the sensor network has a probability of node failures that varies between 0% of its nodes fail to 10% of its nodes fail. The number of sensors used in the experiment is 24, all of which report to any of two available sinks. The communication range between any two sensors is assumed to be 300m. Table 1 lists all the parameters that are used to perform the experiment.

4.2. Performance Metrics

Three performance metrics have been used to evaluate the resilience of the network: network throughput, end to end delay, and energy consumption.

4.2.1. Network throughput

The network throughput is the most important metric when examining the resilience of a network, as it shows the amount of data delivered per unit of time. In our experimental evaluation, the network throughput is calculated by averaging the throughput of all alive nodes. It is assumed that each node produces a Constant Bit Rate (CBR) of 16 Kbps. Sensors under the challenge are assumed to be out of service, generating no traffic.

4.2.2. End to end delay

In many critical smart city applications, the sensed information should reach the control center with the minimum possible delay [22]. The end to end delay is considered an important parameter for quality of service (QoS) guarantees [23]. In this paper, it has been used as one of the metrics to evaluate the network's ability to continue to work under challenges. The end to end delay is defined as the amount of time needed to deliver a packet of data from a sensor to the nearest sink. This delay is affected by four factors: queuing delay, processing delay, transmission delay, and propagation delay.

The queuing delay is the waiting time of packets in the buffer of the sensor node before transmitting, while the processing delay is the time needed to process a packet at each node to prepare it for transmission. In turn, the transmission delay is defined as the time needed to transmit a complete packet from the first bit to the last bit over the communication link. Finally, the propagation delay is defined as the time needed to propagate a bit through the link. It is determined by the travel time of the electromagnetic wave through the physical channel of the communication path [24].

In this experiment, we assume that queuing, processing, and transmission delays are constant for all sensor nodes and for all packets; thus, the main factor in calculating the end to end delay is the prorogation delay, which is a function of the distance between the source and destination nodes; the longer the distance is between them, the longer is the delay. The propagation delay is obtained by tracking one packet that is to be sent from node 0 to the nearest available sink. The packet is supposed to follow the shortest possible path. After determining the path, the path distance is calculated. Then, the delay is determined by multiplying the path distance by the time the packet takes to be transferred for one meter, which is assumed to be 0.5 ms for each meter.

4.2.3. Energy consumption

Energy consumption is also a significant metric when evaluating the resilience of WSN under challenges in smart cities. It is considered one of the key factors in WSN [25]. Energy consumption significantly affects the lifetime of the sensors and the whole network. Lower energy consumption leads to a longer period of network survival under a challenge. This survival is extremely important, especially in the case of natural disaster control and alarm applications in smart cities. In the experiment, the energy consumption for a given pair of source and destination nodes located at multihop h, using the path L, is calculated according to this equation:

$$E(l) = \sum_{k=1}^{h-1} E_{k,ckt}^{rx} + \sum_{k=1}^{h} \left(E_{k,ckt}^{tx} + \frac{\epsilon}{\eta} d_h^{\sigma} \right) \quad (1)$$

where:

$E(l)$ = energy for the complete path
h = number of hops
$E_{k,ckt}^{rx}$ = circuitry energy consumption in transmission
$E_{k,ckt}^{tx}$ = circuitry energy consumption in reception
d_h = distance between two nodes
η = drain efficiency
Є = constant energy
σ = path loss exponent

The number of hops that are considered when calculating the circuitry energy consumption in reception is $(h-1)$ hops. This is because usually, the destination node is powered by an external source [26]. For a single node, the circuitry energy consumption in reception is assumed to be 1 μW. Similarly, the circuitry energy consumption in the transmission is assumed to be 1 μW.

The value of the path loss exponent (σ) usually varies between 2 for free space to 4 or 6 for obstructed areas in building propagation [27]. For simplicity in the experiment, it has been assumed to be always constant and equal to 3. The drain or rectifier efficiency is the ratio of the output radio frequency (RF) power to the input direct current (DC) power [28]. It is a measure of how much DC power is converted to RF power. The value of the drain efficiency (η) is always less than or equal to 1, as the maximum drainage efficiency is 100. For the experiment, it has been chosen to be 0.80. Finally, the constant energy (Є) is assumed to be 0.1 μW.

4.3. Results and Discussions

In this section, we apply the three challenges presented in Section 3.2 to the WSN network explained in Section 4.1 to study its network resilience against such challenges. The results are discussed based on the performance metrics: network throughput, end to end delay, and energy consumption.

4.3.1. Network throughput

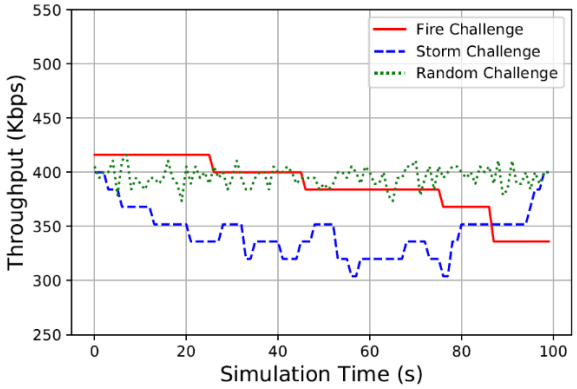

Fig. 7. The change in the network throughput over time under the three challenges.

Fig. 7 shows the results of measuring data traffic throughput during the three challenges. For the storm challenge, the network throughput begins high as the storm starts to hit the sensing area, and most of the sensors are not affected by the challenge yet. As the storm goes inside the sensing area, more sensors are affected. This is reflected as a gradual decrease in network throughput until it reaches its minimum value at t=57 and t=77 when the storm covers the maximum number of sensors. Then, we observe that the throughput recovers to its normal high value at the end of the experiment as the storm is about to leave the sensing area. Similarly, the network throughput under the fire challenge starts with its highest value at t=0, when the fire is too small to affect any sensor. After that, the throughput keeps decreasing as time passes,

without returning to its normal high value. This is because the fire keeps expanding and affecting more sensors, and the affected sensors do not recover to their normal operation. The network throughput under the random challenge decreases when more sensors are affected by the challenge. When there is no failure, the throughput reaches its maximum.

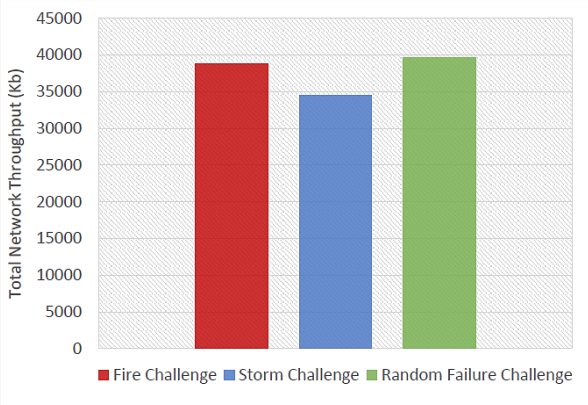

Fig. 8. Total network throughput under the three challenges.

Fig. 8 shows the total data delivered by the network through the whole simulation time under the three challenges. By studying all challenges, we observe that the storm challenge is the most destructive since it covers the largest number of nodes. The fire challenge ranks second in lowering the throughput since nodes do not recover after they are burned. The random challenge has the least effect on the network resilience since failed nodes are not necessarily close to each other. Thus, there exist alternative paths for delivering data to sinks as one or two affected nodes fail.

4.3.2. End to end delay

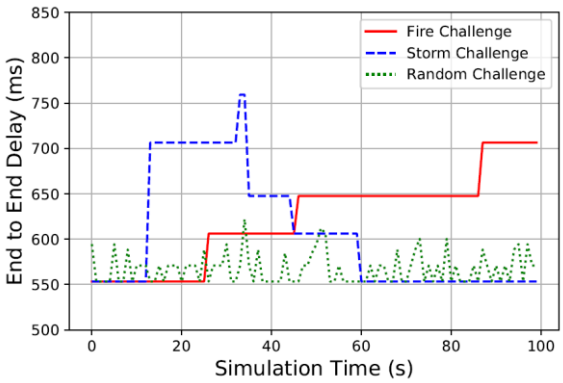

Fig. 9. The change in the end to end delay over time under the three challenges.

Fig. 9 shows the result of measuring the end to end delay over time under the three challenges. Under the storm challenge, the end to end delay starts at its minimum value at t=0, as the storm does not yet affect the shortest path between node 0 and the nearest sink. When the storm reaches the shortest path area, the delay is dramatically increased from 550 ms to 710 ms at t=17. Then, it goes to its maximum when the storm challenge fully covers the sensing area. This is because the network starts to use alternative paths that are longer than the shortest path, which is temporarily blocked by the challenge. As the storm leaves the sensing area, the blocked sensors return to work and the delay decreases again. Similarly, the end to end delay under the fire challenge starts with its minimum value at the beginning as the fire does not reach the

shortest path region yet. As time passes, the fire expands and keeps affecting more sensors, resulting in increasing the delay. This is because the network searches for alternative longer paths to be used instead of the shortest path affected by the fire. The delay keeps increasing each time the network has been forced to use a longer path. Under the random challenge, the end to end delay increases when more nodes fail. When no node fails, the delay is at its minimum value.

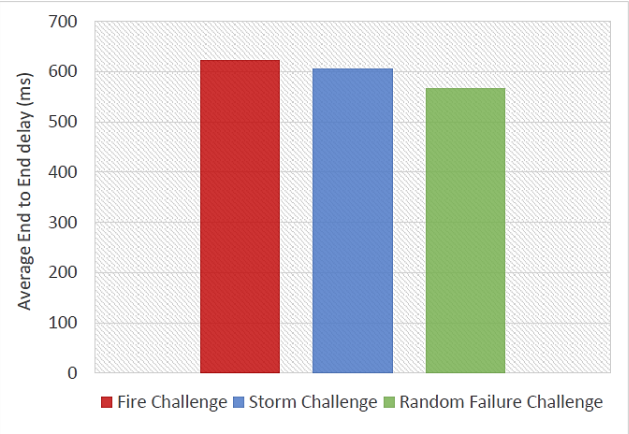

Fig. 10. Average end to end delay under the challenges.

Fig. 10 shows the average end to end delay during the whole simulation time under the three challenges. The fire challenge has the worst effect on the network as it has the highest average end to end delay among the three challenges. The storm challenge comes in second place. The calculated average end to end delay under the storm challenge is less than the average end to end delay under the fire challenge but it is higher than the average end to end delay under the random failure challenge. The random failure challenge has the least effect on the network end to end delay.

4.3.3. Energy consumption

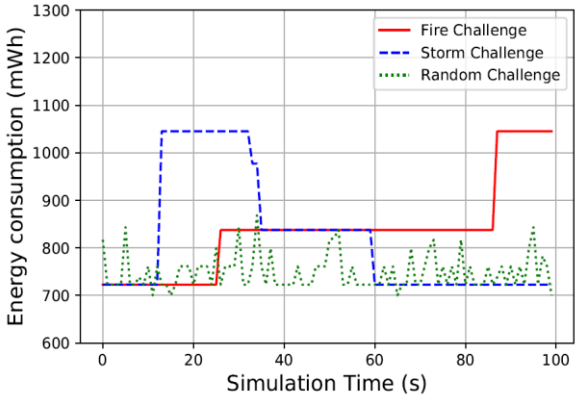

Fig. 11. The change in energy consumption over time under the three challenges.

Fig 11. shows the result of measuring the energy consumption over time under the three challenges. Under the storm challenge, the energy consumption starts at its minimum value as the storm is not entirely inside the sensing area. However, when the challenge starts cutting the shortest path to the nearest sink, the energy consumption increases. This is because, in this case, the network should find an alternative longer path that needs more energy to transmit packets through it. Once the challenge leaves the shortest path area, the energy consumption returns to its standard value. In the case of the fire challenge, the energy

consumption increases each time the network is being forced to find alternative longer paths to be used instead of the shorter ones. However, the shortest paths are never available again once the fire reaches them. This why energy consumption under the fire challenge keeps increasing without returning to normal values, in contrast to the energy consumption under the storm challenge. The energy consumption under the random challenge increases as more nodes fail, following a random pattern as the node failure is modeled to be random.

Fig. 12 shows the total energy consumed by all sensors during the whole simulation time under the three different challenges. The fire challenge caused the most destructive effect on energy consumption among the three challenges as the network consumed the most energy under this challenge in comparison to the other two challenges. The storm challenge ranks second in increasing energy consumption, while the random challenge has the least effect on energy consumption.

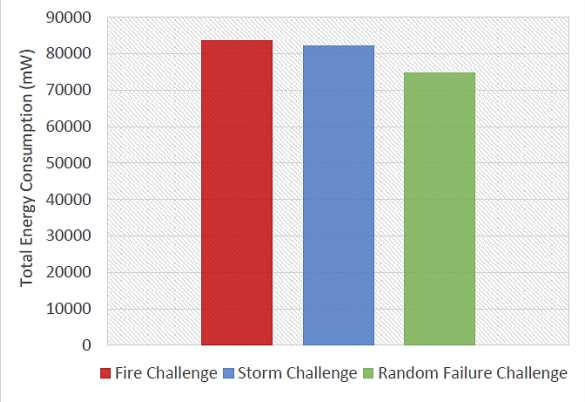

Fig. 12. Total energy consumption under the three challenges.

5. Conclusions and Future Work

Applications of the IoT notion are gaining more importance over the years. Smart cities are supposed to bring together critical applications of IoT. Most smart city applications strongly depend on ubiquitous sensing that is enabled by sensor networks, such as Wireless Sensor Networks (WSN). Modeling smart city networks is an important step to evaluate their performance in the face of the unavoidable challenges that always exist in real implementations. This step can help in studying the best topology design in terms of the number of nodes or node placements. In this research, the resilience of sensor networks has been evaluated under different types of challenges, based on network throughput, end-to-end delay, and energy consumption.

The results show that the storm challenge has the most destructive effect on network throughput, while the random failure challenge has the least. The effect on the network throughput that is caused by the fire challenge is less than that of the storm challenge but more than the effect of the random failure challenge. However, the worst effect on both end-to-end delay and energy consumption is caused by the fire challenge. The storm challenge ranks second in increasing energy consumption and end-to-end delay. Under all cases, the random failure challenge has the least effect on the performance of the sensor networks.

For future work, we plan to use our model with different topologies to show the effect on small-, medium-, and large-scale networks. Moreover, we will use network simulators such as ns-3 to simulate the studied scenarios and compare modeling and simulation results. Besides, other performance metrics can be considered, such as the packet delivery ratio. Additionally, an efficient routing protocol in case of the challenges could be proposed.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

This paper is a partial result of the MS degree research conducted by the first author under the supervision of the second author. The first author wrote the code and implemented the idea provided by the second author. The second author guided the first author throughout the research work.

References

[1] Jayavardhana, G., Rajkumar, B., Slaven, M., & Marimuthu, P. (2013). Internet of things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems 29(7)*, 1645–1660.

[2] Minhaj, A. K., & Khaled, S. (2018). IoT security: Review, blockchain solutions, and open challenges. *Future Generation Computer Systems, 82(2018)*, 395–411.

[3] Bhagya, N. S., Murad, K., & Kijun, H. (2018). Towards sustainable smart cities: A review of trends, architectures, components, and open challenges in smart cities. *Sustainable Cities and Society, 38(2018)*, 697–713.

[4] Vito, A., Umberto, B., & Rosa, M. D. (2015). Smart cities: Definitions, dimensions, performance, and initiatives. *Journal of Urban Technology, 22(1)*, 3–21.

[5] Du, R., Santi, P., Xiao, M., Vasilakos, A. V., & Fischione, C. (2019). The sensable city: A survey on the deployment and management for smart city monitoring. *IEEE Communications Surveys Tutorials, 21(2)*, 1533–1560.

[6] Gharaibeh, A., Salahuddin, M. A., Hussini, S. J., Khreishah, A., Khalil, I., Guizani, M., & Al-Fuqaha, A. (2017). Smart cities: A Survey on data management, security, and enabling technologies. *IEEE Communications Surveys Tutorials, 19(4)*, 2456–2501.

[7] Morello, R., Mukhopadhyay, S. C., Liu, Z., Slomovitz, D., & Samantaray, S. R. (2017). Advances on sensing technologies for smart cities and power grids: A review. *IEEE Sensors Journal, 17(23)*, 7596-7610.

[8] Nandury, S. V., & Begum, B. A. (2015). Smart WSN-based ubiquitous architecture for smart cities. *Proceedings of the 2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*.

[9] James, P. G., Sterbenz, D. H., Egemen, K., Çetinkaya, A. J., Justin, P., Rohrer, M. S., & Paul, S. (2010). Resilience and survivability in communication networks: Strategies, principles, and survey of disciplines. *Computer Networks, 54(8)*, 1245–1265.

[10] Sterbenz, J. P. G. (2017). Smart city and IoT resilience, survivability, and disruption tolerance: Challenges, modelling, and a survey of research opportunities. *Proceedings of the 2017 9th International Workshop on Resilient Networks Design and Modeling (RNDM)*.

[11] Ba-Cuong, H., Thanh-Hieu, N., Thanh-Duong, V., Nguyen-Son, V., & Trung, Q. D. (2019). Natural disaster and environmental threat monitoring system: Design and implementation. *Industrial Networks and Intelligent Systems*.

[12] Wei, Y. Y., Kin, M. L., Terrence, M., Kwong, S. L., Yee, L., & Mei, L, M. (2015). A survey of wireless sensor network based air pollution monitoring systems. *Sensors, 15(12)*, 31392–31427.

[13] Frutuoso, G. M. S., Quoc, T. N., Acácio, F. P. P., Correia, F. M. C., & Fernando, M. L. M. (2019). *Network Analysis Tools*. Springer International Publishing.

[14] Polese, M., Centenaro, M., Zanella, A., & Zorzi, M. (2016). M2M massive access in LTE: RACH performance evaluation in a smart city scenario. *Proceedings of the 2016 IEEE International Conference on Communications (ICC)*.

[15] George, F. R., & Thomas, R. H. (2010). *The ns-3 Network Simulator*. Springer Berlin Heidelberg

[16] Magrin, D., Centenaro, M., & Vangelista, L. (2017). Performance evaluation of LoRa networks in a smart city scenario. *Proceedings of the 2017 IEEE International Conference on Communications (ICC)*.

[17] Sterbenz, P. G., Egemen, K. C., Hameed, M. A., Jabbar, A., & Rohrer, J. P. (2011). Modelling and analysis of network resilience. *Proceedings of the 2011 Third International Conference on Communication Systems and Networks (COMSNETS 2011).*

[18] James, P. G., Sterbenz, E. K., Çetinkaya, M. A., Hameed, A. J., Shi, Q., & Justin, P. R. (2013). Evaluation of network resilience, survivability, and disruption tolerance: Analysis, topology generation, simulation, and experimentation. *Telecommunication Systems, 52(2)*, 705–736.

[19] Egemen, K., Çetinkaya, D. B., Amit, D., Sripriya, S., & James, P. G. S. (2011). Modelling communication network challenges for Future Internet resilience, survivability, and disruption tolerance: A simulation-based approach. *Telecommunication Systems*.

[20] Egemen, K., Çetinkaya, M. J. F., Alenazi, A. M. P., Justin, P. R., & James, P. G. S. (2015). Multilevel resilience analysis of transportation and communication networks. *Telecommunication Systems, 60(4)*, 515–537.

[21] Erik, W. (2015). *Python Geospatial Analysis Essentials*. Birmingham, UK: Packt Publishing.

[22] Pinto, P., Pinto, A., & Ricardo, M. (2013). End-to-end delay estimation using RPL metrics in WSN. *IFIP Wireless Days (WD)*, 1–6.

[23] Min, X., & Martin, H. (2009). Towards an end-to-end delay analysis of wireless multihop networks. *Ad Hoc Networks, 7(5)*, 849–861.

[24] Bovy, C. J. (2002). Analysis of end-to-end delay measurements in internet. *Proceedings of the Passive Active Measurement Workshop*.

[25] Bera, S., Misra, S., Roy, S. K., & Obaidat, M. S. (2018). Soft-WSN: Software-defined WSN management system for IoT applications. *IEEE Systems Journal, 12(3)*, 2074–2081.

[26] Misra, S., Bera, A. M. P., Pal, S. K., & Obaidat, M. S. (2018). Situation-aware protocol switching in software-defined wireless sensor network systems. *IEEE Systems Journal, 12(3)*, 2353–2360.

[27] Miranda, J., Abrishambaf, R., Gomes, T., Gonçalves, P., Cabral, J., Tavares, A., & Monteiro, J. (2013). Path loss exponent analysis in wireless sensor networks: Experimental evaluation. *Proceedings of the 2013 11th IEEE International Conference on Industrial Informatics (INDIN)*.

[28] Zulkifli, F. F., Sampe, J., Islam, M. S., Mohamed, M. A., & Wahab, S. A. (2015). Optimization of RF- DC converter in micro energy harvester using voltage boosting network and bulk modulation technique for biomedical devices. *Proceedings of the 2015 IEEE Regional Symposium on Micro and Nanoelectronics (RSM)*.

Copyright © 2020 by the authors. This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited (CC BY 4.0).

Sarah Lafi Aljohani obtained her BS. degree in computer engineering with honors from Yanbu University College, Saudi Arabia in 2013. She is doing her MS. degree in computer engineering at King Saud University, Saudi Arabia. She had worked as a computer trainer and IT technical support engineer. She got the best presentation award in the 12th International Conference on Computer and Electrical Engineering (ICCEE2019) held at Delf University, Netherlands. Her research interests include embedded systems, mechatronics, wireless sensor networks, the internet of things, and software-defined networks (SDN).

Mohammed J. F. Alenazi obtained his BS. and MS. degrees in computer engineering with honors from the University of Kansas, USA, in 2010 and 2012 respectively. In 2015, he obtained his Ph.D. in computer

science, with honors, from the same university. Currently, he is an assistant professor at the Computer Engineering Department at the College of Computer and Information Sciences in King Saud University, Saudi Arabia. His research interests include software defined networks, internet of things, resilient networks, routing algorithms, multipath transport protocols, mobile ad hoc network (MANET), network topology modeling and design, sensor networks. He published more than 20 research in these areas until now.