



AKADÉMIAI KIADÓ

Gateway placement and traffic load simulation in sensor networks

László Hajdu^{1,2,3}, Balázs Dávid^{2,3} and Miklós Krész^{1,2,3*} 

Pollack Periodica •
An International Journal
for Engineering and
Information Sciences

¹ Department of Applied Information Technology, Institute of Applied Sciences, University of Szeged, H-6720 Szeged, Hungary

² InnoRenew CoE, Livade 6, SI-6310 Izola, Slovenia

³ University of Primorska, SI-6000, Koper, Slovenia

Received: February 01, 2020 • Revised manuscript received: June 07, 2020 • Accepted: July 08, 2020

DOI:

[10.1556/606.2020.00199](https://doi.org/10.1556/606.2020.00199)

© 2020 The Author(s)

ORIGINAL RESEARCH
PAPER



ABSTRACT

Because of the wide variety of possible application fields and the spread of smart devices, the research of wireless sensor networks has become an increasingly important area in the last decade. During the design of these networks, several important aspects have to be considered, for example the lifetime of the network, expected battery usage, or robustness of the installed system. In this paper a simulation environment is introduced that enables the testing of different information spreading methods on the network and provides suggestions for gateway placements with different objectives.

KEYWORDS

wireless sensor network, gateway placement, sensor network simulation, spreading protocols

1. INTRODUCTION

The relevance of the Internet of Things (IoT) and sensors is becoming increasingly important in real-world applications nowadays, as the information they provide can be analyzed and used in several different areas. For this reason, it can be extremely important to deploy these networks in an efficient way, especially in the case of wireless sensor networks, where the sensors need battery power to collect and transmit their data. When its battery dies, a sensor will be unable to communicate with its environment, and as a result, the sensor network might not be able to cover the entire area and collect all the desired information anymore. In a more extreme case, an entire sub-network could be cut off from the network if the dead sensor in question was the only connection linking it to the other parts of the system. These scenarios should be avoided at all costs if possible, as it will prevent data loss on the network. Battery usage mostly depends on the number of data measurements and transmissions done by the sensor. Minimizing the maximum load in the sensor network guarantees that the deployed structure remains intact for the longest possible time.

In this paper, a graph representation of a sensor network is presented, and a framework is proposed that can simulate the transmission of data in the networks, from their origin to the gateway that transmits them to the cloud. This framework measures the load of each sensor node in the network and aims to distribute data transmission as evenly among the nodes as possible. A simulation environment is presented that is able to overview and artificially imitate the operations of a real-world wireless sensor network and propose an optimization method to solve the gateway placement problem with different objective functions. The proposed gateway placement and information spreading methods will be studied with the help of this environment on randomly generated networks.

*Corresponding author.

E-mail: kresz.miklos@szte.hu



2. SENSOR NETWORKS

In most cases, a Wireless Sensor Network (WSN) can be defined as a system that contains multiple individual sensors and gateways to monitor environmental conditions like temperature, humidity, pressure or any environmental effects or properties. The main applications can be grouped by the origin of the collected data, for example transportation [1], healthcare [2, 3] and environmental, structural, or industrial monitoring [4]. Wireless sensor networks have become a huge and well-researched subject in the last decade, since the hardware solutions and the trends along the application areas also changed. However, the main challenge during the design of a wireless sensor network is to provide efficient, robust and scalable solutions. To understand the operation and behavior of a WSN, it is important to note that the term “sensor” mostly refers to the micro-controllers that collect data from the environment through connected sensors or receive and transmit data between each other. In any case, they do not have a direct connection to the Internet. Sensors can communicate with Wi-Fi, radio signal or other technologies and they mostly run on battery. A good overview of the communication protocols for wireless sensor networks can be found in [5]. The other main part of this system is the set of gateways that are connected to the Internet. Their purpose is to collect data measured by the sensors and upload them to a cloud database. The main design issues of a wireless sensor network are the following:

- **Scalability:** The installed system must be easily extendable by additional sensors and gateways to provide high-resolution data. This can be done by using efficient and scalable technologies for communication and data collecting [6];
- **Production cost:** Production cost is an important aspect and highly depends on the price of the used hardware. In general, it is true that the price of the sensors (micro-controllers) is much lower than the price of the gateways, so the placement and the gateway/sensor ratio can have a huge effect on the production cost of the system [7];
- **Robustness:** The robustness provides fault tolerance to the network. If any sensor of the network becomes unable to measure the environment (which can happen because of several issues, like a dead battery or a failure), the sensor network still has to be able to collect and send the data using the remaining part of the system, thus avoiding data losing. A proper sensor placement and strategically located gateways can provide a robust fault-tolerant network [8];
- **Sensor network topology:** Topology is an important structural property that differentiates sensor networks. Examples for this are the star network where a single gateway or base station collects the data from the nodes or the mesh topology where a message can take different paths from the source to the destination where they will be uploaded to the database. Many different topologies exist in the literature, a good overview can be found in [9].

One of the most important aspects of sensor network topology is that it can influence the energy efficiency, robustness or the fault tolerance of the network;

- **Power consumption:** Power consumption is an important property of any wireless sensor network, but it is mostly related to other design issues, meaning that power consumption highly depends on the network layout [10], the design of the hardware [11] or the overall energy management schemes developed for the network [12].

However, these are only the most important aspects of the network design, for a more detailed overview about wireless sensor networks, see [13, 14]. The quality of the provided system can depend on different factors, and it can be stated that several properties of the network depend on the quality of the used hardware and protocols.

3. MODELING THE PROBLEM WITH GRAPHS

In this section, a graph data structure is presented to model a general sensor network. This data structure will be used to introduce the different strategies for information spreading on the network, which is applied to simulate the total load of each sensor node. Based on these load values, methods will also be introduced for recommending different sensor placements on the network in order to distribute this load more efficiently among the sensors.

Graph models are widely used to represent wireless sensor networks, as they provide a simplified yet realistic structure that can be used for algorithm development. A good presentation of the most important models of this type can be found in [15]. As problems arising on a WSN usually involve stochastic choices, Markov chains can be used to model these processes [16]. Modeling tools also exist for their simulation; see PRISM [17] as an example of a universal framework for building and analyzing probabilistic models. Studying and characterizing sensor network topology is also important. This characterization can be done in a variety of ways [18], for example using k -hop based density metrics [19]. Besides density, the connectivity of sensor networks should also be studied. One way to determine the above characteristics is shown in [20], where the SAT representation of the graph is used to analyze the connectivity of wireless sensor networks.

3.1. Basic layout

Consider the undirected graph $G = (V, E)$ as an abstract representation of a sensor network. The nodes of node set V can be of two different types; $S \subset V$ represents the set of sensors, while $W \subset V$ gives the set of gateways, $V = S \cup W$, with $S \cap W = \emptyset$. The nodes of the graph correspond to the possible connections between these nodes; considering a sensor node $u \in S$ and any arbitrary node $w \in V$, node $(u, w) \in A$ represents that information can travel between these nodes. As the graph is undirected, the information can travel both ways, and any sensor node can transmit a message to any other connected node. Gateways are considered as sink nodes that are the final destinations of messages,



without the ability to transmit message. To simulate the message sending function for a gateway as well, the above network can easily be modified; a sensor node has to be added to the network with the exact same connections as the gateway in question, while also being connected to the gateway. In this case, the gateway should lose all of its connections, except the one that links it to this sensor.

3.2. Information spreading on the network and simulation environment

To represent the real-life behavior of a wireless sensor network, the simulation of information spreading is realized in an iterative way. As the first step of iteration, every sensor generates a message that contains the measured information about the environment. The objective of each message is to reach a gateway in order to be uploaded and saved into the cloud database. This process is managed through the subsequent steps of the iteration; in each step, every message is transmitted on the network by their current sensor nodes based on a pre-defined strategy. This strategy is used to choose a neighbor of the node that will serve as the next state of the actual message. If a message reaches a gateway node, then it is immediately removed from the network. These transmission steps will be repeated until there are no more messages on the network, and the current iteration ends at this point. As each sensor node generates a single message in every iteration, the total number of iterations will be given by the number of desired measurements. The following message spreading strategies are introduced for the simulation environment:

- *Random*: The sensor node transmits the message to a random neighbor node, so a random sensor will be the next state of the actual message. It is important to note, that in this case no information is needed from the neighbor nodes, so the sensor only has to know the list of the neighbors to which the measurement can be transmitted;
- *Min-load neighbor*: The message is sent to the neighboring node that has the minimal load (i.e. minimal battery usage) so where the battery is less used. In case there are multiple nodes with minimal load, the new position is chosen randomly among them applying a uniform distribution. The information has to be collected and up to date from the neighbors of the actual node, which consumes energy. Naturally the battery information can be transmitted together with the messages;
- *Load-weighted random*: The message is sent to a random neighboring node, but neighbors with smaller loads are more likely to be selected. The new position is chosen using categorical distribution, where each node $u \in N$ in the candidate set is considered with a $1/load(u) + 1$ weight. The additional cost in this case is the same as before, so the load of the neighbors has to be updated.

The simulation environment will first consider the input graph, and then determine a placement for the gateways on the network. This will be done either randomly, or by using

one of the methods presented in the next section. Once the gateways are selected, exactly one of the above methods will be chosen by the simulation, and messages will be transmitted on the network based on this spreading strategy.

3.3. Optimizing gateway placement

During the operation of a wireless sensor network, the sensors use energy when they measure the environment or when data is transmitted from a node to its neighbor. The battery usage of a node will be in correlation with its load; those sensors of the network that transmit significantly higher number of messages than others will run out of battery earlier because of being overloaded. The structure of the network and the spreading method of the messages can have a high impact on the load distribution of the network. Moreover, the optimal placement of the gateways can also help with the energy efficiency of the network. Let network G be the same as before, with $V = S$, meaning that every node is a sensor initially and the structure of the connections between them is given. The objective of the optimization problem is to choose k number of sensors as gateways in order to create the W node set that balances and minimizes the load between the sensors. The proposed algorithm is greedy method, that starts with an empty gateway set W , and iteratively increases the size of this until it reaches k ; in every iteration, it examines all the nodes that can be candidate for being a gateway, and chooses the one that is the best from the objective function point of view. This will be added to the set of gateways. The outline of the algorithm can be seen in [Algorithm 1](#).

The proposed algorithm is a greedy method that starts with an empty gateway set W and iteratively increases its size until it reaches k ; in every iteration, it examines all the nodes that can be candidates for being a gateway, and chooses the one that is the best from the objective function point of view. This will be added to the set of gateways. It has to be noted that messages might get lost in the early iterations of the algorithm due to the low number of gateways in the network. In some special network structures, it is also possible that a message gets stuck in an infinite loop and never reaches a gateway. To avoid this scenario, the simulation environment includes a parameter that limits the

```

INPUT: Network  $G$ , Gateway number  $k$ , Spreading method  $p$ ,
objective function  $f$ 
 $W \leftarrow \emptyset$ 
 $j \leftarrow 0$ 
WHILE  $|W| < k$ 
   $best \leftarrow 0$ 
  FOR EACH  $s \in S$ 
     $actual \leftarrow f(W \cup \{s\}, p)$ 
    IF  $actual > f(W \cup \{best\}, p)$ 
       $best \leftarrow s$ 
    END IF
  END FOR
   $W \leftarrow W \cup \{best\}$ 
END WHILE

```

Algorithm 1. Gateway placement optimization



length of the path travelled by a single message. If a message reaches the maximum path length without finding a gateway, the simulator deletes it from the network and the message will be lost. However, since the goal of the gateway placement is to find a solution that is the best from the objective function point of view, even these lost messages will have an impact on the final placement as they also generate load on the network.

The algorithm can work with different objectives depending on the preference of the user or the maintenance company of the wireless sensor network. The different objectives are the following:

- **GLL:** The GLocal Load objective tries to minimize the sum of the loads in the network. If the objective is to save as much energy as possible, it is a good choice to minimize the global number of the messages in the system. Nevertheless, it will not balance the load between the nodes so weak points can be formed during the lifetime, causing dead spots in the network;
- **MIL:** MInimal Load means that the algorithm tries to minimize the load of the most critical sensors that have the highest load in the network. This objective function can be used to maximize the lifetime of the network;
- **BAL:** In the case of the BALanced Load, the optimal scenario occurs when every single sensor has the same load, which can be calculated by dividing the number of the messages by the sensors number. The algorithm tries to minimize the difference between the optimal load and the real load in the case of every node.

It is also important to note that the different spreading methods can work differently with the proposed gateway placements therefore the algorithm is able to choose the best method/placement combination to improve the efficiency of the solution. The next section will introduce the test results of the above methods on different generated wireless sensor networks.

4. ANALYSIS OF THE METHODS

The numerical analysis of the above methods is presented in this section, and their efficiency is shown on different instances. This section first presents the properties of these generated instances, then provides a thorough analysis of the proposed methods on them.

4.1. Test instances

Test instances were randomly generated with the forest fire model [21]. The properties of the random network are the following:

- The number of the nodes are 250, 500, 750, 1,000, and 1,500 so the networks was generated in 5 different sizes;
- The number of the edges are generated from 873 to 5,205;
- The forward burning probability was 0.34;
- The ratio of gateways is 5% of the number of nodes in every case.

Table 1 shows the detailed properties of the test instances.

4.2. Results of the methods

The efficiency of the proposed method is shown through a series of simulations. Each network is tested for all possible combinations of information spreading method and gateway placement objective. An instance will be represented by the following tuple: $\langle network, spreading\ method, placement\ objective \rangle$. In addition to the gateway placement objectives presented in Section 3.3, the random placement of gateways using a uniform distribution is also considered. The initial input network consists only of the nodes and connections between them. The following steps are executed for each instance:

1. Assignment of the appropriate number of gateways (see Table 1) to the initial network. Depending on the placement objective, this is either realized by running Algorithm 1, or by choosing random gateways on the network with a uniform distribution;
2. Simulation of information spreading on the resulting network of (1), as described in Section 3.2. The number of iterations (number of messages generated in each node) is 1,000 for each instance and the simulation ends once all messages will reach a gateway. Iteration ends when all generated messages reach a gateway, and there are no messages lost.

Using the results of the above steps, the maximum and average load of the sensor nodes in the network can be shown for each instance, as well as the total load of the network. As it is stated in (1), the efficiency of the proposed placement objectives is compared to an arbitrary gateway placement (random) for each instance.

As these simulations involve a large number of random choices at all levels, they were executed 100 times for each instance. Tables 2–4 present the aggregated results of these simulations. Each row of a given table defines the instance size and the method for spreading information (*Random – R*, *Min-Load neighbor – ML*, *Load-weighted Random – LR*), while the columns will show the loads for the four different sensor placement methods from Section 3.3 (*random*, *GLL*, *MIL*, *BAL*). All values are the averages of the 100 simulations, rounded to integer value. The tables also present the ratio of the load given by the proposed gateway placements compared to the random placement. The tables only consider the load of the sensor nodes of the system, and the

Table 1. Properties of the input graphs

Network	Number of nodes	Number of edges	Number of gateways
WSN_250	250	873	12
WSN_500	500	1,552	25
WSN_750	750	3,452	37
WSN_1000	1,000	3,708	50
WSN_1500	1,500	5,205	75



Table 2. Average total network load

Instance	Average total load after gateway placement				New load ratio (%)		
	Random	GLL	MIL	BAL	GLL	MIL	BAL
250_R	8,081,777	1,557,493	1,646,392	1,607,363	19.27	20.37	19.89
250_LR	1,255,454	244,577	390,510	259,629	19.48	31.11	20.68
250_ML	5,453,787	4,647,462	5,046,937	5,228,541	85.22	92.54	95.87
500_R	17,593,294	2,894,198	4,424,666	3,057,585	16.45	25.15	17.38
500_LR	4,698,854	778,916	1,447,049	960,168	16.58	30.80	20.43
500_ML	13,254,218	9,239,819	11,443,381	10,651,296	69.71	86.34	80.36
750_R	20,003,556	3,369,555	5,432,409	4,217,637	16.84	27.16	21.08
750_LR	2,855,514	726,228	1,615,482	844,388	25.43	56.57	29.57
750_ML	16,283,432	13,441,927	14,512,352	14,576,725	82.55	89.12	89.52
1000_R	31,271,245	6,323,079	9,720,546	6,394,088	20.22	31.08	20.45
1000_LR	7,108,488	1,442,847	3,028,112	1,690,771	20.30	42.60	23.79
1000_ML	22,879,301	18,044,404	20,189,745	19,843,386	78.87	88.24	86.73
1500_R	48,155,340	10,110,836	21,356,748	13,166,814	21.00	44.35	27.34
1500_LR	10,163,597	2,515,110	4,876,187	3,087,132	24.75	47.98	30.37
1500_ML	35,747,053	28,271,097	30,767,634	33,024,288	79.09	86.07	92.38

Table 3. Average load per node

Instance	Average load per node after gateway placement				New load ratio (%)		
	Random	GLL	MIL	BAL	GLL	MIL	BAL
250_R	32,957	5,544	5,917	5,753	16.82	17.95	17.46
250_LR	5,275	1,027	1,640	1,090	19.47	31.10	20.67
250_ML	21,915	18,527	20,205	20,968	84.54	92.20	95.68
500_R	36,038	5,093	8,315	5,437	14.13	23.07	15.09
500_LR	9,892	1,639	3,046	2,021	16.57	30.79	20.43
500_ML	26,903	18,452	23,091	21,423	68.59	85.83	79.63
750_R	27,055	3,725	6,619	4,915	13.77	24.46	18.17
750_LR	4,004	1,018	2,265	1,184	25.42	56.57	29.56
750_ML	21,837	17,852	19,353	19,444	81.75	88.62	89.04
1000_R	31,917	5,655	9,232	5,730	17.72	28.92	17.95
1000_LR	7,482	1,518	3,187	1,779	20.29	42.59	23.78
1000_ML	23,083	17,994	20,252	19,887	77.95	87.74	86.15
1500_R	32,793	6,095	13,987	8,239	18.59	42.65	25.13
1500_LR	7,132	1,764	3,421	2,166	24.74	47.97	30.37
1500_ML	24,085	18,839	20,591	22,174	78.22	85.49	92.07

Table 4. Average maximum load

Instance	Average max. load after gateway placement				New load ratio (%)		
	Random	GLL	MIL	BAL	GLL	MIL	BAL
250_R	226,543	46,721	22,100	22,975	20.62	9.76	10.14
250_LR	38,331	6,245	7,307	7,260	16.29	19.06	18.94
250_ML	64,766	21,107	35,997	65,412	32.59	55.58	101.00
500_R	312,589	26,382	42,510	34,149	8.44	13.60	10.92
500_LR	111,740	9,472	13,157	9,229	8.48	11.77	8.26
500_ML	119,677	40,088	56,687	62,634	33.50	47.37	52.34
750_R	370,875	21,546	48,527	42,612	5.81	13.08	11.49
750_LR	33,917	12,023	12,882	10,234	35.45	37.98	30.17
750_ML	62,993	40,540	40,486	40,544	64.36	64.27	64.36
1000_R	392,303	47,000	65,886	46,931	11.98	16.79	11.96
1000_LR	93,675	9,714	20,100	9,390	10.37	21.46	10.02
1000_ML	73,494	45,088	41,802	41,656	61.35	56.88	56.68
1500_R	511,659	50,816	132,564	101,177	9.93	25.91	19.77
1500_LR	86,522	12,737	25,587	16,683	14.72	29.57	19.28
1500_ML	109,829	60,178	61,049	67,166	54.79	55.59	61.15



gateways are not present in the statistics. The decrease achieved by the specific gateway placement objective compared to the random one is shown, as well as the load values at the end of the simulation, where both the placement objective and the information spreading strategy had an influence.

Table 2 presents the total load on the network after all messages reached a gateway. This is important for measuring the total energy consumption of the network. It can be seen that there is a load decrease in almost every test instance. The exceptions to this are the instances where the *Min-Load Neighbor* information spreading method was used. This is a deterministic method that aims to maintain a network where the load of every node is more or less the same. This can also be seen for the instances where gateway placement was done using the *Minimal Load* objective function. In general, it can be said that both the *GLL* and *BAL* gateway placement methods improved the initial total load compared to a random placement (with *GLL* being slightly better than *BAL* in every case). As for the method of information spreading on the network, the *Random* and *Load-Weighted Random* strategies are both good combinations with these two gateway placements. If only the final maximum load values are considered on the network, it can be clearly seen that the combination of the *GLL* placement objective combined with the *Load-Weighted Random* strategy has by far the lowest total load in every case.

Table 3 shows the average load of a single node after the simulation. This measure could be useful for giving and estimate of the required battery capacity of sensors in the network.

The observations are the same as in the case of the total load of the network. The *MIL* gateway placement and the *Min-Load Neighbor* information spreading method both perform poorly, while all the other placement objectives and spreading methods are efficient. Again, the *GLL* placement objective provides better results than the *BAL*, and the *Random* spreading provides a bigger total decrease in load, than the *Load-Weighted Random*. However, the *Load-Weighted Random* spreading strategy in combination with the *GLL* placement objective outperforms every other option. These results should not come as a surprise, because the average load of a single node is strongly connected to the total average load of the network.

Table 4 presents the average maximum load of the network, which is an important measure for showing the worst-case metric for energy usage, as this illustrates the load of the most used sensor node. The given results are not as straightforward as before with regard to the amount of decrease in load. What remains consistent is that the *ML* spreading strategy is the worst of the three in most cases, while there are test cases where the *MIL* gateway placement objective actually performs well compared to the other two. The amount of decrease given by the *GLL* placement and *Random* spreading strategy does not dominate the *BAL* and *Load-Weighted Random* that clearly as before, and their performance differs based on the underlying graphs, and the combination of placement objective and spreading strategy. What remains consistent, however, is that the *Load-*

Weighted Random provides the lowest load values again, and depending on the instance, this can be decreased further by choosing either the *GLL* or the *BAL* placement objective.

It can be seen from the above test results that deterministically distributing the load over the network with the *Min-Load Neighbor* spreading strategy is never a good option, as it performed poorer on all test instances than either of the other methods. It is useful to have randomness in the transmission decisions of the network and introducing a bit of control over this (using loads as weights) will greatly improve the efficiency, which can be seen from the lower values provided by the *Load-Weighted Random* spreading strategy in every case. As for gateway placement objectives, *MIL* also underperformed, and *GLL* seems to be the best strategy from most aspects, with *BAL* is better only in the case of decreasing the maximum load of certain networks.

5. CONCLUSIONS

With the significant increase in the importance and availability of wireless sensor networks over the past decade, studying the structure of the networks and identifying possible bottlenecks can provide useful information for the design of these systems. In this paper, a simulation environment has been proposed that could be used to study the transmission of messages among the nodes of a wireless sensor network, and monitor the loads of the entire network, as well as individual nodes. This is important as high load decreases the battery life of a sensor, and as a result, dead sensors can cause data loss and problems in the connectivity of the network. By minimizing the maximum load of a node can ensure the complete connectivity of the network for a longer period of time. An abstract graph structure to represent the network has been introduced and different strategies both for choosing the placement of gateways on the network and for the transmission of messages among the nodes were proposed.

The above methods were tested in the simulation environment for a large number of instances and the results were analyzed concerning the maximum load of the network, the average load of a node and the sum of loads in the entire network. While some methods proved to be more useful than others, it was shown that deterministic methods performed poorly in comparison with stochastic approaches. It could also be seen that considering the load of neighboring nodes when making a random decision improves the spreading process.

However, there is still room to improve the simulation process. In the future, the simulation will be tested on real-world networks and results will also be analyzed in connection with the structure of the network.

ACKNOWLEDGMENTS

The Authors gratefully acknowledge the European Commission for funding the InnoRenew CoE project (Grant Agreement 739574) under the Horizon2020 Widespread-Teaming



program, and the Republic of Slovenia (Investment funding of the Republic of Slovenia and the European Union of the European Regional Development Fund). The Authors are also grateful for the support of the Slovenian Academy of Sciences and Arts (project title: ‘Deployment and analysis of sensor networks in buildings’), and for the support of the Slovenian ARRS grants N1-0093 and J7-9404.

The Authors would also like to acknowledge the work of Ferenc Kálmán, who was helping the research with the implementation of the simulation environment. He was supported by “Integrated program for training new generation of scientists in the fields of computer science”, no. EFOP-3.6.3-VEKOP-16-2017-0002 (supported by the European Union and co-funded by the European Social Fund).

REFERENCES

- [1] M. Franceschinis, L. Gioanola, M. Messere, R. Tomasi, M. A. Spirito, and P. Civera, “Wireless sensor networks for intelligent transportation systems,” in *IEEE 69th Vehicular Technology Conference*, Barcelona, Spain, Apr. 26–29, 2009, 2009, pp. 1–5.
- [2] A. Darwish and A. E. Hassanien, “Wearable and implantable wireless sensor network solutions for healthcare monitoring,” *Sensors*, vol. 11, no. 6, pp. 5561–5595, 2011.
- [3] Á. Garai, I. Péntek, A. Adamkó, and Á. Németh, “Methodology for clinical integration of e-Health sensor-based smart device technology with cloud architecture,” *Pollack Period.*, vol. 12, no. 1, pp. 69–80, 2017.
- [4] N. Xu, S. Rangwala, K. K. Chintalapudi, D. Ganesan, A. Broad, R. Govindan, and D. Estrin, “A wireless sensor network for structural monitoring,” in *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems*, Baltimore, Maryland, USA, Nov. 3–5, 2004, 2004, pp. 13–24.
- [5] L. K. Ketshabetswe, A. M. Zungeru, M. Mangwala, J. M. Chuma, and B. Sigweni, “Communication protocols for wireless sensor networks: A survey and comparison,” *Heliyon*, vol. 5, no. 5, Paper no. e01591, 2019.
- [6] L. K. Alazzawi, A. M. Elkateeb, A. Ramesh, and W. Aljuhar, “Scalability analysis for wireless sensor networks routing protocols,” in *22nd International Conference on Advanced Information Networking and Applications – Workshops*, Okinawa, Japan, Mar. 25–28, 2008, 2008, pp. 139–144.
- [7] D. Zorbas, P. Raveneau, and Y. Ghamri-Doudane, “Assessing the cost of deploying and maintaining indoor wireless sensor networks with RF-power harvesting properties,” *Pervasive Mobile Comput.*, vol. 43, pp. 64–77, 2018.
- [8] M. Bhushan, S. Narasimhan, and R. Rengaswamy, “Robust sensor network design for fault diagnosis,” *Comput. Chem. Eng.*, vol. 32, no. 4–5, pp. 1067–1084, 2008.
- [9] S. Sharma, D. Kumar, and K. Kishore, “Wireless sensor networks - A review on topologies and node architecture,” *Int. J. Comput. Sci. Eng.*, vol. 1, no. 2, pp. 19–25, 2013.
- [10] C. Zhou, M. Wang, W. Qu, and Z. Lu, “A wireless sensor network model considering energy consumption balance,” *Math. Probl. Eng.*, vol. 2018, pp. 1–8, 2018.
- [11] P. Györke and B. Pataki, “Power supply design for solar powered harvesting-aware sensors,” *Pollack Period.*, vol. 7, no. 7, pp. 45–54, 2012.
- [12] F. Engmann, F. A. Katsriku, J. D. Abdulai, K. S. Adu-Manu, and F. K. Banaseka, “Prolonging the lifetime of wireless sensor networks: A review of current techniques,” *Wireless Commun. Mobile Comput.*, vol. 2018, pp. 1–23, 2018.
- [13] C. Buratti, A. Conti, D. Dardari, and R. Verdone, “An overview on wireless sensor networks technology and evolution,” *Sensors*, vol. 9, no. 9, pp. 6869–6896, 2009.
- [14] M. A. Matin and M. M. Islam, “Overview of wireless sensor network,” in *Wireless Sensor Networks – Technology and Protocols*, M. A. Matin, ed., Brunei, Darussalam: Intech Open Access Publisher, Ch. 1, pp. 3–24, 2012.
- [15] S. Schmid and R. Wattenhoffer, “Algorithmic models for sensor networks,” in *Proceedings 20th IEEE International Parallel & Distributed Processing Symposium*, Rhodes Island, Greece, Apr. 25–29, 2006, 2006, pp. 1–11.
- [16] L. Ding and Z. H. Guan, “Modeling wireless sensor networks using random graph theory,” *Physica A: Stat. Mech. Appl.*, vol. 387, no. 12, pp. 3008–3016, 2008.
- [17] M. Kwiatkowska, G. Norman, and D. Parker, “PRISM 4.0: Verification of probabilistic real-time systems”, in *Proc. 23rd International Conference on Computer Aided Verification*, vol. 6806, G. Gopalakrishnan, S. Qadeer, eds, Springer, pp. 585–591, 2011.
- [18] J. M. Hernández and P. Van Mieghem, “Classification of graph metrics,” Report, Delft University of Technology, pp. 1–20, 2011.
- [19] Cs. Bíró and G. Kusper, “Some k-hop based graph metrics and node ranking in wireless sensor networks,” *Ann. Mathemat. Inform.*, vol. 50, pp. 19–37, 2019.
- [20] Cs. Bíró and G. Kusper, “Equivalence of strongly connected graphs and black-and-white 2-SAT Problems”, *Miskolc Math. Notes*, vol. 19, no. 2, pp. 755–768, 2018.
- [21] J. Leskovec, J. Kleinberg, and C. Faloutsos, “Graphs over time: Densification laws, shrinking diameters and possible explanations,” in *Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Chicago, Illinois, USA, Aug. 21–24, 2005, 2005, pp. 177–187.