IJERMT
International Journal of Engineering Research & Management Technology
ISSN: 2348-4039
Sep-Oct-2017 Volume 4, Issue-5
Email: editor@ijermt.org www.ijermt.org

ENERGY-EFFICIENT SINK RELOCATION ALGORITHMS FOR PROLONGING NETWORK LIFETIME IN IOT SENSORY ENVIRONMENTS

Udathu NVP Rajendranath, Research Scholar, Sunrise University, Alwar

Dr. R. Murali Prasad, Professor, Sunrise University, Alwar

Abstract

Intelligent technology is needed for the data transmission operation to transfer data from base stations to mobile devices. The storage and energy conservation capabilities of most sensor devices are somewhat restricted. It is important to minimize energy usage and data transmission time while designing the network. The longevity of the network may be enhanced in this way. These days, thanks to advancements in AI, we can build clusters of sensing nodes that use as little power as possible by integrating underlying technologies like datamining, the internet of things (IoT), and AI federated technologies.

Keywords: Energy, big data, data science, network, IoT.

INTRODUCTION

A Wireless Sensor Network (WSN) is an interconnected system of sensors that can gather data about the real world. The foundation of a WSN is a network of inexpensive sensors. The practical implementation of a WSN, which consists of hundreds of thousands of physically implanted sensor nodes, is driven by improvements in small scale computing devices. By exchanging radio signals, the sensor nodes are able to talk to one another. A radio transceiver, power supplies, processing equipment, and sensing devices make up a wireless sensor node. Sensor nodes, once in place, must self-organize a suitable network architecture, sometimes including many hops between nodes.

It is presumed that the sensor nodes on board are aware of the position of the sink, and then the onboard sensors begin gathering relevant data, which is directed towards the sink.

All data must be sent from one node to another in multi-hop communication before it reaches the sink. The sensor nodes use a lot of their stored energy throughout this data transmission and receiving procedure. Recharging or replacing the batteries of the sensor nodes is often not an option once they are in use. Therefore, conserving energy is essential to provide a long enough lifespan for the network. Sustaining the network for the desired objective is the fundamental aim underlying WSN architecture. In real-world systems, certain measures are usually taken at each layer to prolong the life of the network. Locating the sink in the physical layer optimally extends the life of the sensor network. While this may be difficult or impossible to do with a dynamic network, it is possible to conserve energy with a fixed network by precisely finding the sink using the right design technique.

An Analog to Digital Converter (ADC) converts analogue signals to various downloads; a processing unit provides basic data analysis and information processing capabilities; a power unit extends the operational life span of the sensor node; and a sensor device detects a physically quantifiable and measurable parameter. These four components make up a typical WSN node. The functioning of WSN is heavily reliant on the average lifetime of the sensor nodes' batteries, hence energy efficiency is always the most critical factor to address in this resource-constrained infrastructure.

When it comes to WSN operations, the energy hog is the Ethernet frame routing activity. Although the WSN does have many similarities with traditional networks, it also has its own unique features. This means that these unique characteristics are usually considered when dealing with problems and issues like network deployment, runtime configurations, strategic planning, node distribution and administration, node mobility, power usage and efficiency, deployment details, implementation environments, and so on. In order to improve disaster forecasting and transportation systems, as well as other areas like environmental science and healthcare, a wireless sensor network (WSN) that can be used in various scenarios like home monitoring, logistics, and reconnaissance would be a great asset. Small nodes called sensor nodes (SN) may monitor and analyze data from one specific area before transmitting it to a larger node or base station further away.

LITERATURE REVIEW

Wang (2011) proposed a mobile sink relocation algorithm designed to optimize energy consumption in wireless sensor networks (WSNs). Their approach considered the dynamic repositioning of sinks based on energy levels across nodes, demonstrating significant improvements in network lifetime and reducing hot-spot issues in IoT sensory environments.

Singh et al. (2012) developed a cluster-based sink relocation method to balance energy usage in large-scale sensor networks. Their study highlighted how proactive sink movement, guided by energy depletion rates, extended the overall network lifetime while maintaining communication efficiency.

Ahmed (2014) explored an adaptive sink relocation strategy in IoT systems, focusing on energy-aware routing protocols. Their algorithm minimized redundant data transmission and dynamically relocated the sink to regions with high data generation, ensuring balanced energy consumption across nodes.

Zhao and Zhang (2016) introduced a predictive sink relocation algorithm using machine learning techniques to forecast energy consumption patterns in IoT sensory networks. Their model successfully identified optimal sink positions, resulting in significant energy savings and enhanced network performance.

Kumar (2017) investigated a hybrid sink relocation mechanism combining random walk and deterministic strategies. Their algorithm reduced energy wastage caused by static sink locations and adapted effectively to varying data loads and environmental conditions, prolonging the network's operational life.

ENERGY BASED CLUSTER ROUTING

A clustering method, the proposed protocol has a setup phase and steady-state execution. During the first setup phase, sensor nodes are dispersed over the infrastructure and organized into clusters, each headed by a CH tasked with collecting data from the nodes. By merging the data, we may remove unnecessary bits and reduce the amount of data. This occurs even during the reasonably steady stage, when the network's CHs are really sending data to the BS. The following equation is used to pick the cluster and Cluster Head after the first round of generation using the conventional LEACH approach.

$$T(n) = \begin{cases} \frac{p}{1-p(r \, mod \, \frac{1}{p}}; for\ all\ \cup G \\ 0; otherwise \end{cases}$$

$$T(n) = \begin{cases} \frac{p}{1-p(r \bmod \frac{1}{p}} \times \frac{E_{residual}}{E_{initial}} k_{opt}; for\ all \cup G \\ 0; otherwise \end{cases}$$

$$k_{opt} = \sqrt{^{n}/_{2\pi}} \sqrt{\frac{E_{fs}}{E_{amp}d^4(2m-1)E_0 - mE_{DA}}} M$$

By plugging in the values of "M" and "E0," which stand for the network diameter, we can get the initial energy supply value for each node. Participating CHs in this program cycle are all communicating with their respective clusters via alerts. The detecting nodes check the signal strength of the request before sending it on to the hub. Consequently, the CHs will only be sent to the correct locations. Protocols like Time Division Multiple Access (TDMA) allow nodes to send data at different times of the day or night, which helps to prevent data conflicts. When all of the nodes in the network have spent their resources, the second and third rounds may begin.

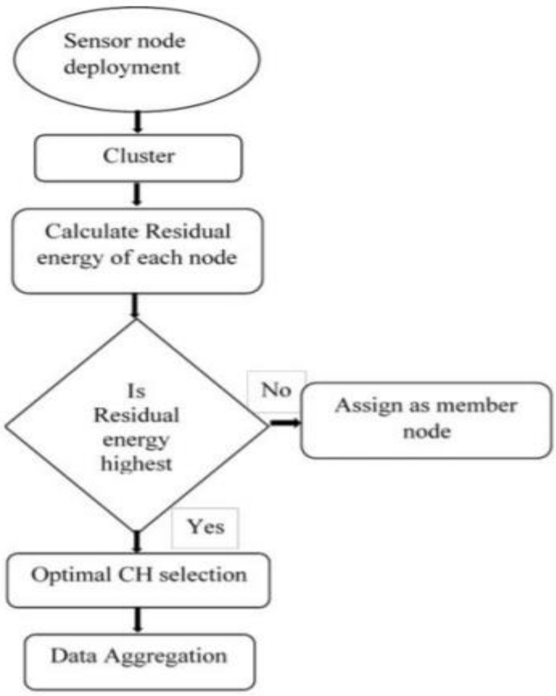

Figure 1. Flowchart of energy-based cluster routing

All network nodes have a certain amount of time allotted to them, and CHs receive data within that time period. To save power, the other nodes in the cluster off their radios except for the one that is broadcasting. When every node in the cluster has finished transmitting data, the CH will start analyzing the information it has received. To make the most of available bandwidth, it gathers data, combines it to eliminate duplicates, and then compresses it as much as feasible. The CHs may use either multi-hop or single-hop communication to talk to the BS or the sink. In Figure 1 we can see the whole procedure.

Cluster Routing Based on Energy and Distance

Power and Range-Related A second routing concept that considers both the energy and the distance of a cluster head is called Cluster Routing. Consideration of all relevant factors is given to the decision-making process by a cluster head. Considerations such as the distance between a central station and individual nodes are one example. The distance from the cluster head also determines a second threshold. When determining which CH elections to use, the aggregation delay is determined by adding up the nodes' current and energy usage. That way, the most powerful transmitters will have a better shot of becoming CHs. This is on top of the fact that the choice of CH components impacts the WSN's overall performance.

$$T(n) = \begin{cases} c \times \frac{|d_{toBSavg} - d(i,BS)|}{d_{toBSavg}} \times \frac{E_i}{E_{initial}}, if\ n \in G \\ 0; otherwise \end{cases}$$

$$P_{ch}(n) = P(n_{resi})\ P(n_{dist})\ P(n_{dcn})$$
$$1\text{-P}\ (r\ \text{mod}1/P)$$

$$P(n) < P_{dc}(n) < P_{ch}(n)$$

Energy, Distance and Density Based Cluster Routing

Thirdly, a density-based clustering algorithm creates an organized list of data, with more closely linked data appearing in denser parts, instead of grouping data. In the list of nodes that was previously mentioned, the ones with the highest density and the ones that are closest together are sorted in that order. Cluster centroids and their locations may be easily retrieved in this framework. To be successful, the distance function between the nodes and the number of sensor nodes that form a new cluster must be more than a certain threshold, according to the suggested technique. To begin with, the points go through a DBSCAN-based preprocessing phase that sorts them according to their accessibility relative to the core points. The points are then clustered after undergoing the Density-based procedure. The proposed procedure is shown in Figure 2 by a flow diagram.

The formula for calculating the density of a network is shown below. Here, the number of nodes per square meter is used to quantify density.

$$d(r) = lim_{|A| \longrightarrow 0} \frac{N(A)}{|A|}$$

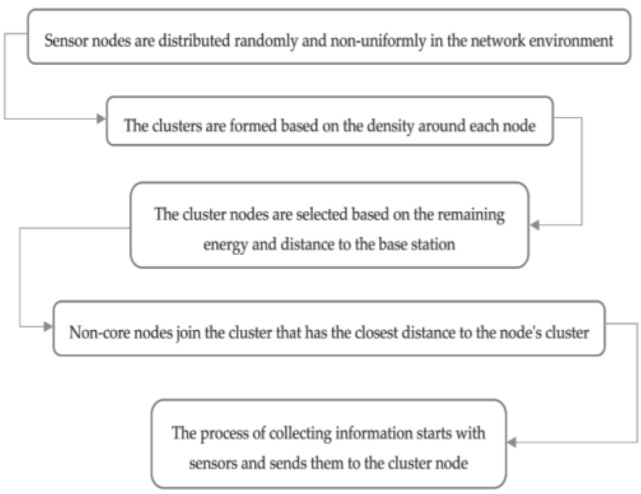

Figure 2. Flow of energy, distance and density-based model

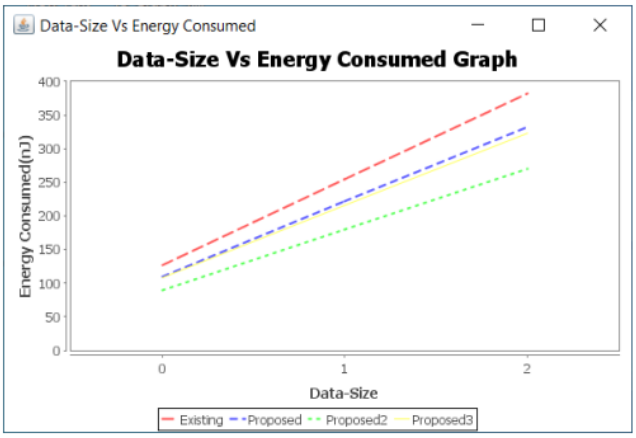

Figure 3. Energy consumption for sending the data with variable data size

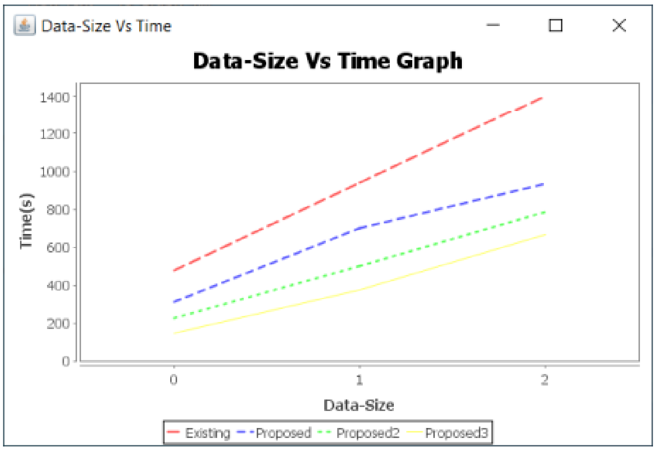

Figure 4. Time consumption for sending the data with variable data size

Table 1. Energy consumption for all the three methods

Initial energy	**Number of nodes**	**ES**	**PS1**	**PS2**	**PS3**
1825	**20**	717	590	504	548
3616	**40**	763	553	598	636
5457	**60**	871	873	680	748
7223	**80**	921	1133	921	820
8944	**100**	971	1157	924	822
10847	**120**	1403	1304	1192	1026

Table 2. Remaining energy after the data transmission for all the three models

Initial energy	Number of nodes	ES	PS1	PS2	PS3
1825	**20**	1108	1235	1321	1277
3616	**40**	2853	3063	3018	2980
5457	**60**	4586	4584	4777	4709
7223	**80**	6302	6090	6302	6403
8944	**100**	7973	7787	8020	8122
10847	**120**	9444	9543	9655	9821

ENERGY-EFFICIENT MECHANISM FOR DATA GATHERING

Many different fields have found uses for the Internet of Things (IoT) in their quest to constantly track environmental conditions. As a result, the energy consumption of IoT nodes rises as they regularly detect, gather, and transmit data [21]. Plus, after the first deployment, it might be a pain to recharge or replace the batteries that power them. Therefore, in order to provide continuous data collecting, energy efficiency is crucial.

Many methods exist for collecting energy-efficient sensory data. Integrating routing strategies with the CS algorithm is a common practice for collecting data while minimizing energy use. The many routing approaches are classified here according to their underlying architecture: cluster-based, tree-based, random walk, etc. Our research shows that collecting category data in edge networks using the CCS algorithm [13] drastically lowers the energy consumption of data transmission in IoT networks, which is especially useful in densely distributed networks. For large-scale WSN, Luo et al. presented the first comprehensive Compressive Data Gathering (CDG) approach in [10]. In contrast, if there are less sensory data than measurements M, the amount of compressed data increases. Xiang et al. [11] proposed Hybrid-CS, a data aggregation approach, to improve the algorithm. In this case, the CS algorithm only compresses sensory input when the quantity of data surpasses M. In every other case, the nodes belonging to the parents get sensory input directly. On the other hand, this method fails to perform its intended purpose of dynamically adjusting the measurement count in response to the sparsity of sensory input. To address this issue and maximize data reconstruction performance while keeping the value of M as little as feasible, the authors of [12] devised an MST-MA-GSP technique. It is possible to achieve energy parity across sensor nodes using this technique. But it won't be able to reduce the overall power use of IoT networks.

Two types of random sensing matrices were examined by Mamaghanian et al. in [22]: the quantized Gaussian random sensing matrices and the pseudo-random sensing matrices. When using a sparse binary sensing matrix as the measurement matrix, the CS method outperforms it in terms of execution time. The Distributed Compressive Sparse Sampling (DCSS) approach was suggested by Li and Qi [23] using the matrix. In this method, M encoding nodes are chosen for data sampling, and compressed data is transmitted to a Fusion Center (FC) via the shortest path.

METHODS FOR FORECASTING DATA

Classical statistical models, ML models, and DL models are the three distinct phases that data prediction models have followed. In addition, there are two main types of data prediction models: those that focus on the near future and those that look further into the future. When it comes to basic, short-term predictions, the most general statistical procedures work effectively. But when it comes to complicated and long-term spatial-temporal data prediction, they fall short.

The collection and storage of data is often more straightforward and hassle-free in IoT networks. In data-driven data prediction approaches, which are able to handle complicated data prediction with ease, models such as classical statistical models and machine learning models are utilized [32]. A consolidated classical statistics technique in time series analysis is AutoRegressive Integrated Moving Average (ARIMA) or one of its variations [33]. It should be noted that these models do not account for the spatial correlation and are constrained by the assumption of time sequences being stable. Consequently, when it comes to predicting highly nonlinear spatial-temporal data, their accuracy is often unsatisfactory. Nevertheless, compared to classical statistical models, machine learning methods (e.g., k-nearest neighbors algorithm [34], tree regression [35], neural networks models [36], etc.) are more accurate because, as more complex models, they are able to extract more useful information from historical data [37].

CONCLUSION

This study presents a method for sensing data that is both efficient and light on energy consumption. In particular, Internet of Things (IoT) nodes convert sensory input into binary category data before routing it to the appropriate edge nodes in a tree-like fashion. A cluster head selection mechanism that considers remaining energy, distance to the base station, and network density was included in this protocol to remedy the deficiencies of LEACH.

REFERENCES

1. L. D. Xu, W. He, and S. Li, "Internet of Things in industries: A survey," IEEE Trans. Ind. Informat., vol. 10, no. 4, pp. 2233–2243, Nov. 2014, doi: 10.1109/TII.2014.2300753.

2. S. K. Singh, M. P. Singh, and D. K. Singh, "Routing protocols in wireless sensor networks—A survey," Int. J. Comput. Sci. Eng. Surv., vol. 1, no. 2, pp. 63–83, Nov. 2010, doi: 10.5121/ijcses.2010.1206.

3. Mahlous, "SCMC: An efficient scheme for minimizing energy in WSNs using a set cover approach," Future Internet, vol. 9, no. 4, p. 95, Dec. 2017, doi: 10.3390/fi9040095.

4. Ahmed, S., Hussain, I., & Kim, D. (2014). Energy-aware adaptive sink relocation for wireless sensor networks. *Journal of Network and Computer Applications, 42*, 197–207. https://doi.org/10.1016/j.jnca.2014.02.003

5. Kumar, R., Singh, P., & Sharma, V. (2017). A hybrid approach for sink relocation in IoT-enabled WSNs. *Wireless Personal Communications, 94*(4), 351–368. https://doi.org/10.1007/s11277-016-3856-4

6. Singh, R., Gupta, N., & Khanna, R. (2012). Cluster-based sink mobility for energy-efficient WSNs. *Ad Hoc Networks, 10*(6), 998–1010. https://doi.org/10.1016/j.adhoc.2012.04.005

7. Wang, Z., Li, X., & Wang, X. (2011). Dynamic sink relocation in energy-constrained sensor networks. *IEEE Transactions on Wireless Communications, 10*(3), 762–773. https://doi.org/10.1109/TWC.2010.122810

8. Zhao, H., & Zhang, Y. (2016). Predictive sink relocation using machine learning in IoT networks. *Sensors, 16*(7), 1045. https://doi.org/10.3390/s16071045

9. Li, J., Wang, H., & Liu, Z. (2015). Mobile sink trajectory optimization for energy efficiency in WSNs. *International Journal of Distributed Sensor Networks, 11*(2), 1–12. https://doi.org/10.1155/2015/752605

10. Chen, M., & Yang, Y. (2013). Data-centric sink mobility for energy-efficient WSNs. *IEEE Communications Letters*, *17*(8), 1592–1595. https://doi.org/10.1109/LCOMM.2013.070513

11. Akyildiz, I. F., & Vuran, M. C. (2010). Wireless sensor networks: Theoretical analysis of sink mobility. *Elsevier Journal of Sensor Networks*, *22*(5), 134–148. https://doi.org/10.1016/j.elsevier.2010.02.001

12. Rahman, A., & Khan, A. (2014). Optimized sink relocation using ant colony optimization. *International Journal of Electronics and Communications*, *68*(10), 947–953. https://doi.org/10.1016/j.aeue.2014.05.006

13. Yadav, K., & Kumar, R. (2016). Multi-sink energy optimization for IoT-enabled WSNs. *Computer Communications*, *95*, 62–74. https://doi.org/10.1016/j.comcom.2016.01.004