*Embedded Self Organizing Systems (Vol 9. No 2. 2022) (pp. 3-8)*

Embedded Selforganizing Systems

Issue Topic: "Future Challenges in Automotive Research and Development "

# An Energy-aware Ad-hoc Routing Strategy for Queriable Wireless Sensor Networks

René Bergelt
Chemnitz University of Technology
Professorship of Computer Engineering
E-mail: rene.bergelt@informatik.tu-chemnitz.de

**Abstract — The data volume handled by wireless sensor networks (WSN) is ever growing due to increasing node counts and node complexity – be it in traditional WSN applications or for Car2X or Internet-of-Things. Queriable WSN are a concept to handle the large data volumes in such networks by abstracting the network as a virtual database table to which users can pose queries. This declarative approach enables networks which can flexibly adapt to changing application requirements. In addition they possess a flat learning curve since users do not need to have a high technological understanding of the sensor node firmware. Upon executing a query it is first propagated through the network and once it has reached the desired nodes, results are collected and send back through the query-posing node (usually the sink). The routing which is used for the data aggregation step plays a major role in the energy efficiency in networks with increasing node and sensor value counts as represented by Car2X networks for instance. In this paper, an ad-hoc routing strategy for queriable WSN is proposed which is both energy-aware and application-specific. It will be shown that this routing can contribute greatly towards decreasing the energy consumption needed for data aggregation and thus helps increasing the network's lifespan.**

***Keywords— queriable wireless sensor networks, energy-aware routing, ad-hoc routing, data aggregation***

## I. Introduction

Wireless Sensor Networks (WSN) have been in the focus of distributed networking research for many years. A WSN consists of a number of distributed sensing platforms (sensor nodes) which communicate wirelessly. Nodes are equipped with sensor hardware to collect data which is then sent to a special node – the so-called sink – where it is processed or made available to the user or other systems, respectively. Usually, these nodes operate on battery power and therefore they are very limited in processing capability and energy reserves. This basic structure of a WSN is illustrated in Figure 1. By now, industrial applications are used in the field which utilize WSN technology to observe environmental phenomena and detect specified events or – when coupled with actor hardware – directly influence their surroundings through feedback control systems. Usually, wireless nodes are deployed when the area of

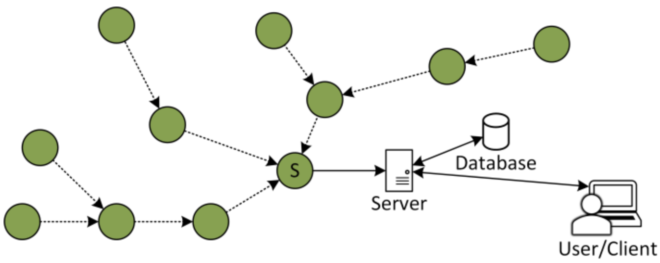



**Figure 1 Basic structure of a wireless sensor network with a downstream processing system**

interest does not provide infrastructure for communication or power supply, or is very large for instance with agricultural applications. In addition, WSN come into consideration if the introduction of cable connections is not desired or possible, for instance in setting up house automation systems in older buildings. A central problem statement in WSN applications is to maximize the networks lifespan by using energy on the node level efficiently. This can be mainly achieved by reducing the need for wireless communication as well as reducing the volume of information which is transmitted. This is due to the fact that an active radio module usually contributes the greatest portion towards the energy consumption of a node [1, 2]. In addition, for multi-hop networks where not every node can directly reach the sink, the routing, i.e. the path data takes through the network from a node to reach the sink, has a great impact on energy consumption as longer paths need more communication processes. The objective is that the network can be operated using its full node set for as long as possible instead of having a perpetually decreasing node count due to specific nodes having a higher load and depleting their energy resources much faster than others. Thus, for many WSN sensing tasks it is desirable that the energy consumption is evenly distributed between nodes as best as possible. Consequently, this leads to a network which is easier to maintain and whose performance can be better predicted [3].

With the advent of Internet-of-Things and Car2X scenarios these problem statements traditionally occurring in WSN are becoming more and more prevalent in the realm of end-user and consumer electronics [4, 5, 6].

## II. Related Work

Since routing has a major impact on the energy efficiency of data aggregation and thus on the lifespan of wireless sensor networks, a multitude of routing approaches and protocols has been devised.

Many protocols deal with energy-aware routing where the goal is to optimize the energy efficiency of a data aggregation task by distributing the load between nodes. The most prevalent examples for this type of routing approaches are LEACH ((Low-Energy Adaptive Clustering Hierarchy) [7] and PEGASIS (Power Efficient Gathering in Sensor Information Systems) [8]. Both approaches are based on clustering the node set. For this, special nodes are selected which function as so-called cluster heads. These cluster heads are the only nodes allowed to directly communicate with the sink and as such are bearing the highest communication load. The remaining nodes are assigned to a cluster and are only allowed to communicate with the respective cluster head based on a timed schedule. On a regular basis, new nodes will be assigned to be cluster heads so that energy consumption is evenly distributed between all nodes. In their originally proposed form, both algorithms do not take the remaining energy level of nodes into account when choosing cluster heads and as such unfavorable selections might be made where a cluster head dies shortly after it has been selected resulting in a necessary restructuring dissipating additional energy.

Other, newer protocols which are often based at least in parts on LEACH or PEGASIS are VCH-ECCR (Vice-Cluster-Head-Enabled Centralized Cluster-based Routing) [9], EELBR (Energy-efficient Load Balancing and Reliable Routing) [10] and SMR (Energy-efficient Static Multi-hop Routing) [11]. While often being energy-aware, i.e. taking the remaining energy levels into account, these routings are usually oblivious of the actual task the network should execute. As such they are only able to distribute energy evenly between nodes if there is a steady sensor data flow with the same probability or rate from all nodes. However, if the sensing task comprises only specific nodes routes are not be established with regards to retrieving application-specific information from these nodes as fast as possible resulting in undesired detours of data on its way through the network. Furthermore, for centralized algorithms such as VCH-ECCR the routing is determined at the sink node which makes a separate network exploration step necessary and results in an increasing management overhead the larger the network gets.

In contrast, routing approaches which provide users the possibility to formulate task descriptions which are then executed by the network form the foundation of queriable WSN [12]. One such approach is Directed Diffusion [13]. It specifies a routing and data aggregation approach using so-called interests. Basically, an interest is a task description for the network which defines ranges for sensor values the user is interested in [13, 14]. These interests are sent into the network through the sink node and distributed to the further nodes. If a

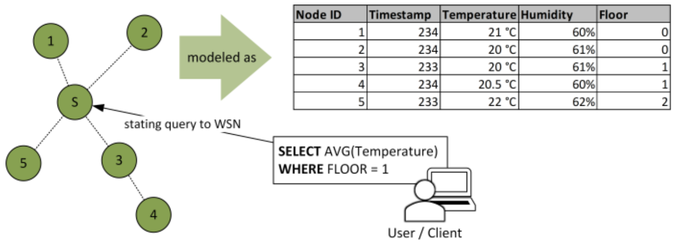



**Figure 2 Modeling of a WSN as virtual database table which can be queried using a query language**

node measures data which lies in the range of the interest it then sends it to the sink. The paths from a node to the sink via neighboring nodes are called gradients [14]. After some rounds of aggregation, gradients which have been used more often than others will be established as the only aggregation routes for the interest. As such, Directed Diffusion realizes an application-specific routing since the routing is built towards the interest it tries to satisfy. However, it is not energy-aware since the energy level of nodes is not taken into account when forming routes and thus the even distribution of energy consumption in the network is not handled. Consequently, an interest might establish a gradient which will fail after a short time if it accidently selected nodes whose energy is already almost depleted. Furthermore, the actual data structure and implementation of interest as well as interest caches are not elaborated. As such, there is still a high effort needed to operate an actual WSN using this approach. In addition, the protocol favors aggregation tasks where results from a single node are transmitted on a shortest path to the sink. Sensing tasks which are distributed between nodes and in-network processing where data is filtered and coalesced along the aggregation path are not handled by it.

Other queriable WSN realizations are database-oriented protocols which model the sensor network as a database table [12]. As such, the available sensors of a node are treated as columns of a virtual table and each row represents a measurement at a node at a specific point in time as illustrated in Figure 2. Using this model, users can formulate queries in a query language often resembling SQL which are then executed by the network. Mainly, this approach has three advantages: First, users do not need to have a deep technical understanding to extract sensor data from the network. Should they already possess previous knowledge regarding database systems the usage of a query language has an even lower entrance barrier. Second, users can state that not all measured data should be sent to the sink but instead filter data by conditions. Third, by using so-called aggregate functions (e.g. SUM, MIN, MAX) in the queries, data can already be processed and reduced along the routing tree. This allows users to effectively reduce network traffic and volume by only extracting information which is necessary for the actual application of the WSN [15]. Furthermore, by replacing the active queries, the network can easily adapt to changed application requirements. Examples for such database-oriented aggregation systems are TinyDB, Cougar, Planetary and SINA (Sensor Information Networking Architecture) [15, 16]. Query-based systems enable the WSN to use cross-layer optimization for routing in the network since users state intents on what should happen with data. Unfortunately, routing in existing, queriable WSN usually is not done in an energy-aware way, i.e. node energy is not taken into account or is not query-specific, i.e. the underlying routing tree is not built for a specific query. Furthermore, many systems require nodes to maintain a knowledge base with information about their neighbors and the network in general or the routing is decided in a centralized fashion at the sink so that perpetual network exploration is required.

In this paper, we propose a non-centralized approach for ad-hoc routing in queriable WSN which builds a specific routing for each query and takes node energy levels into account to balance the load within the network.

## III. Routing Concept

The desired routing tree for a query in a queriable wireless sensor network is one that includes all nodes which satisfy the conditions of the query and excludes all nodes which are not necessarily needed. A node is not needed if and only if it does not fulfill the query conditions and is not needed by other included nodes to reach the sink. As such, we want to find a routing tree in the network which connects all nodes which satisfy the query conditions while including as few as possible unrelated nodes. This problem can be modeled using the Steiner tree problem. Let $G = (V, E)$ be a graph with vertex set $V$ and the edge set $E$ including edge weight values and the subset of vertices $R \subseteq V$ which are called terminals. The Steiner tree of this graph is a subgraph of $G$ which contains all vertices of $R$ and has the minimal total edge weight. For a WSN we can define an undirected graph where each sensor node is represented by a vertex and where an edge between two vertices exists if the corresponding sensor nodes are able to communicate with each other (i.e. they are within reach). We now want to determine the Steiner tree of this graph for a given query $Q$ with condition set $Q_c$ where the terminals in $R$ are the sensor nodes which fulfill the query conditions. We define the weight of the edge between two nodes $a$ and $b$ with respect to the query $Q$, where the query gets propagated from node $a$ to node $b$, as follows:

$$\omega_{a,b}(Q) = P_a(Q) * \frac{1}{E(a)} * {h_a}^2 \qquad (1)$$

with

$$P_x(Q) = \begin{cases} 10, & Q_c \text{ not satisfied by node } x \\ & 1, \text{else} \end{cases} \qquad (2)$$

In addition, $E(x)$ is a metric for the remaining energy level of node $x$ in the range (0, 1] with 1 representing the maximum capacity. The value of $h_x$ represents the minimum distance of the node to the sink (hop count). Consequently, the routing algorithm should favor connections to nodes with higher energy levels, which are nearer to the sink and which are also part of the query. Figure 3 shows a simple sample network where the nodes highlighted in green are terminals. The label of each node is its energy metric value and the edge weights have been calculated accordingly. By using a Steiner tree for the aggregation of a query in the network, we can ensure that all necessary nodes are included in the tree (application specificity) and that load is balanced since the more often a node is chosen to be included in the routing the lower its energy level becomes and thus the higher its node weight gets (energy-awareness).

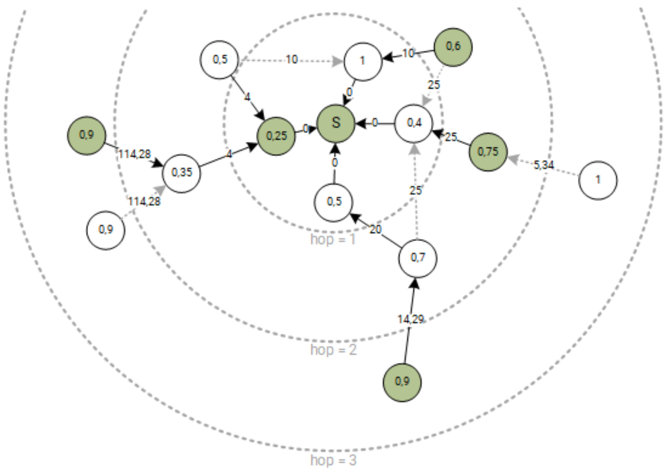



**Figure 3 Sample network with highlighted terminals (node energy level as node label) and corresponding edge weight; resulting Steiner tree represented by solid lines**

**input**: node n, Query Q, set of nodes from which Q was received S

***function*** $choose\_parent(n, Q, S)$
  $h_n = 1 + \min(h_s)$ *for* $s \in S$
  $\omega_n(Q) = P_n(Q) * \frac{1}{E(n)} * h_n^2$
  $children = propagate(Q, \omega_n(Q), h_n)$
  ***if*** $children = \emptyset$ ***and*** $P_n(Q) = 10$ ***then***
    ***return*** $\emptyset$
  **else**
    $p \in S$ *where* $\omega_p$ *is minimal*
    ***return*** $p$
  **end if**

**Listing 1 Proposed parent selection algorithm**

However, since the Steiner tree calculation is an NP-complete problem [17] it is not feasible to compute the exact Steiner tree for routing in larger networks. In addition, most graph algorithms for calculating the Steiner tree use global knowledge about the graph which we do not possess at the node level since we want to create an ad-hoc routing scheme where the aggregation route is determined during the propagation phase of an individual query. On that account, we propose a heuristic method for nodes to select their parent node for a propagated query $Q$ which is geared to the Steiner tree as shown in Listing 1. Basically, we employ a local greedy algorithm where a node selects the neighbor node as its parent which has the minimal edge weight calculated using equation (1). Since the edge weight between two nodes $a$ and $b$ is only dependent on node a, we can calculate the weight of an edge $\omega_{a,b} = \omega_a$ for any node $a$ already on the node itself without the need to know which nodes will receive the propagation. As such, the edge weight can be included as a value in the re-propagated query. A node will only select a parent in case it is a terminal node or if it has been selected by any child node as parent. Nodes which are neither terminals nor have been selected as parents will be excluded from the aggregation routing tree for the query. As it is the case with many hop-based gradient routing approaches, nodes can never choose sibling nodes (i.e. nodes which have the same hop count) to avoid circles and sending duplicate data in the routing. However, since the algorithm possesses application-specific knowledge, it might be beneficial to choose a route over sibling nodes, especially if the siblings themselves are also terminals. Therefore, we propose an extension which allows same-level links (SLL). For this, the algorithm faces two challenges: First, it must not produce any circles as the resulting routing graph would be not a valid tree anymore. Second, it must not result in disconnected sub-graphs which cannot reach the sink anymore by creating connections in a way that no node has a connection to a node with smaller hop count. As every node can only decide on its parent and has no knowledge of the parent choice of its neighbors, these two requirements are ensured by imposing the rule that a node which has been selected as parent by a sibling node must select a parent with a smaller hop-count (so it might be required to switch the already selected parent).

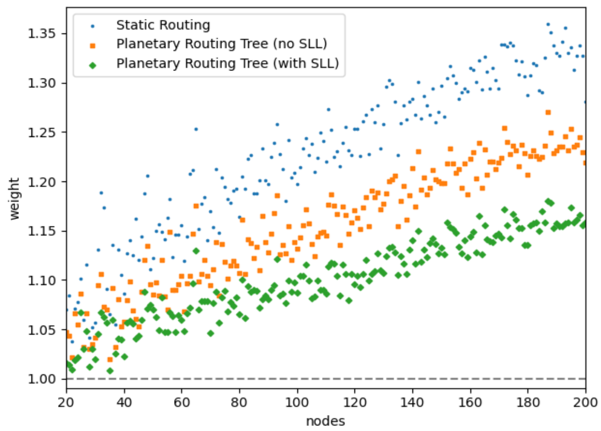
## IV. Evaluation

In this section, the routing algorithm proposed in this paper will be compared to other routing strategies in queriable WSN which are neither application-specific nor energy-aware.

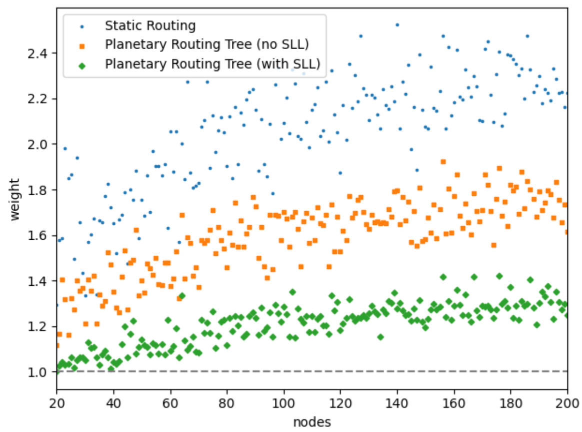
The first routing algorithm to compare with is called static routing. There a pre-established routing tree is to be used for the propagation and aggregation, i.e. each node has a fixed parent which never changes. The second algorithm is called dynamic routing. There the query is propagated through the network and a node which receives the query from a number of nodes with smaller hop-count chooses any of these nodes as parent. This aggregation approach is used for instance by TinyDB for its data aggregation algorithm (TAG, Tiny Aggregation) [17]. Since we have implemented the routing for the Planetary WSN aggregation system, the routing we proposed in this paper is called Planetary Routing.

The first evaluation is a comparison between the resulting routing tree weight and the weight of the (approximated) Steiner tree for a given query. The energy levels for the nodes are assumed to be 1 in this evaluation, since only the impact on the routing tree for a single query is examined. For this, networks with a node count between 20 and 200 have been simulated. In each network, 10 % of nodes have been chosen as terminals, i.e. nodes which fulfill the query conditions. The tests were run both with a random terminal set and with a clustered terminal set, where the terminal nodes are all neighbors, as this better represents the actual use case of querying an area in a WSN (e.g. temperature data from floor 1, cf. Figure 2). For each node count 100 random networks have been generated and the weight of each routing tree was averaged for each algorithm. As reference, the Steiner tree was approximated using the Python library NetworkX [19]. As in this scenario static and dynamic routing are basically the same since only a single query is observed, the dynamic routing is omitted from the evaluation.

The plots in Figure 4 show the simulation results in relation to the approximated Steiner tree's weight with non-clustered terminals (Figure 4a) and clustered terminals (Figure 4b), respectively. Naturally, the lower the weight the better suited the resulting routing tree. For both cases it can be seen that the Planetary routing variants consistently outperform the static routing approach. In addition, with increasing network sizes the




a) Using 10% non-clustered terminal nodes




b) Using 10% clustered terminal nodes

**Figure 4 Comparison of the weight of the resulting routing tree for static routing and two variants of Planetary routing, for networks of different node counts; weights were normalized with Steiner tree weight (dashed line for reference)**

weight of the static routing's trees increases more than for the Planetary routings. Comparing the two variants of the Planetary routing it can be seen that there is a significant improvement when allowing same-level links. As such, this version should be preferred. Furthermore, the routing trees for the simulations with clustered terminals show a larger improvement and less dependency on network size as well as a lower variance when compared to the static routing.

In order to determine the actual impact of the routing algorithm on the lifespan of a queriable WSN, we conducted a follow-up evaluation where a running sensor network was simulated. In this simulation, each node is equipped with a 90 mAh energy reservoir, i.e. a small cell battery like CR2016 (3V). The energy consumption of the wireless communication is simulated using a modeled IEEE 802.15.4 radio module as shown in [12] to mimic a realistic node operation. After starting, randomized queries with 10% clustered terminals are continuously stated to the network. When a node is part of the query it sends a single sensor value to the sink (via other nodes if necessary). After the execution of each query, the number of active nodes, i.e. nodes which have not depleted their energy and can reach the sink

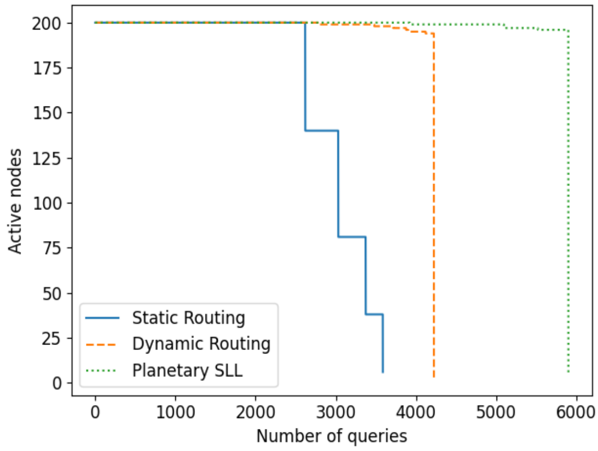



**Figure 5 Comparison of network lifespan measured in executed queries for a network with 200 nodes and 10% clustered terminals**

directly or through other nodes, is recorded. Naturally, the more queries that can be executed before the network fails (no remaining active nodes) the better. In addition, it is desirable that nodes fail gradually towards the end of the network's lifespan so that the complete network stays operable as long as possible. The result for a single simulation of a network with 200 nodes is shown in Figure 5 comparing Planetary SLL with the static and dynamic routing approaches. It can be seen that the static routing results in the shortest network lifetime among the compared approaches. In addition, nodes fail en bloc so that large parts of the network become inoperable at the same time which is not desirable. The dynamic routing approach is able to distribute the load better and shows a more desirable failing curve where the major part of the network stays operable during its whole lifetime. This is due to the fact that a parent with depleted energy resources gets replaced by another node during propagation. By using Planetary SLL, the lifespan of the same network can be extended even more since it shows a much better load distribution and almost all nodes of the network stay active over the course of the network's operation time. This is also due to the fact that mostly only nodes relevant to the query are selected as part of the routing.

To get a general impression of the lifespan improvement of Planetary SLL routing, 1000 such simulations have been run. Figure 6 shows a histogram plot of these simulations for the change to the simulated network's lifespan using dynamic routing or Planetary SLL when compared to static routing. It can be seen that Planetary SLL results in a larger lifetime improvement much more often than dynamic routing. Comparing the network lifespan for each simulation, the dynamic routing approach improves the lifespan by around 23% compared to the static routing on average, whereas Planetary SLL increases the lifespan by 56%. When directly compared to the dynamic routing, the improvement of the Planetary SLL approach is still around 27%. In addition, while there are cases where the dynamic routing performs worse than the static routing (by a reduction in lifespan of up to 6%), using Planetary SLL always increased the lifetime of the network by at least 9%.

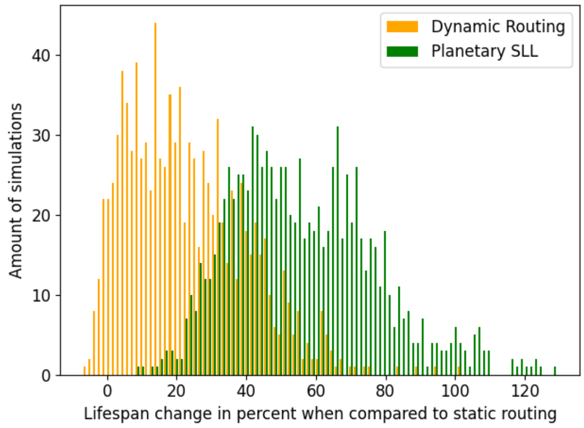



**Figure 6 Histogram of the network's lifespan change (100 bins have been used)**

In the non-clustered scenario, the network's lifespan improvement when using Planetary SLL is around 38% compared to static and 6% compared to dynamic routing on average.

## V. Conclusion

In this paper, we have shown that using an energy-aware and application-specific ad-hoc routing can improve the network lifetime of a queriable wireless sensor network. Since no prior knowledge of the network topology is required, a separate network exploration step is not necessary. As a consequence of this, scenarios with node mobility, e.g. Car2X communication applications, could also be investigated.

In addition, using different routing trees for queries contributes to an improved load sharing within the network in the sense that energy consumption is distributed more evenly between nodes. In this work, we have conducted simulations for networks of up to 200 nodes in size. Future research should evaluate whether the observed improvements are also present for larger networks. Additionally, it would be of interest if the increase in lifetime for the non-clustered scenario becomes more significant with increasing network size.

In the proposed form, the routing tree is only suitable for one-shot queries, i.e. queries which are run once and where data is aggregated immediately. This is due to the fact, that the query conditions usually contain comparisons to measured sensor data. Naturally, measurements can change during the continuous execution of a query so that the established routing may not contain all nodes which satisfy the conditions for future aggregation phases (since the query is not re-propagated). Future simulations could be geared towards finding a good trade-off point between re-propagating and result accuracy for continuous queries.

## References


[1] S. Madden, R. Szewczyk, M. J. Franklin and D. Culler, "Supporting Aggregate Queries Over Ad-Hoc Wireless

Sensor Networks," in *Proceedings Fourth IEEE Workshop on Mobile Computing Systems and Applications*, 2002.

[2] N. Trigoni, A. Guitton and A. Skordylis, "Querying of Sensor Data," in *Learning from Data Streams*, Springer Berlin Heidelberg, 2007, pp. 73-86.

[3] I. Raicu, "Efficient Even Distribution of Power Consumption in Wireless Sensor Networks," in *Proceedings of ISCA 18th International Conference on Computers and Their Applications*, 2003.

[4] B. Kundaliya, "Challenges of WSNs in IoT," in *Wireless Sensor Networks - Design, Deployment and Applications*, 2020.

[5] A. Thakur and R. Malekian, "Internet of Vehicles Communication Technologies for Trafc," *Wireless Personal Communications,* no. 109, pp. 31-49, November 2019.

[6] F. Kargl, P. Papadimitratos, L. Buttyan, M. Muter, E. Schoch, B. Wiedersheim and V. T. Ta, "Secure Vehicular Communication Systems: Implementation, Performance, and Research Challenges," *IEEE Communications Magazine* , vol. 46, no. 11, December 2008.

[7] W. R. Heinzelman, A. Chandrakasan and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks," in *Proceedings of the 33rd Hawaii International Conference on System Sciences* , 2000.

[8] S. Lindsey and C. S. Raghavendra, "PEGASIS: Power-efficient gathering in sensor information systems," in *Proceedings of the IEEE Aerospace Conference*, 2002.

[9] R. Pachlor and D. Shrimankar, "VCH-ECCR: A Centralized Routing Protocol for Wireless Sensor Networks," *Journal of Sensors,* 2017.

[10] K. Samara, *An Energy Efficient, Load Balancing, and Reliable Routing Protocol for Wireless Sensor Networks,* University of Wisconsin-Milwaukee, 2016.

[11] E. Alnawafa and M. Ion, "New Energy Efficient Multi-Hop Routing Techniques for Wireless Sensor Networks: Static and Dynamic Techniques," *Sensors,* June 2018.

[12] R. Bergelt and W. Hardt, "An Event-based Local Action Paradigm to Improve Energy Efficiency in Queriable Wireless Sensor Actuator Networks," *Journal of Communications Software and Systems,* vol. 1, no. 16, pp. 66-74, März 2020.

[13] C. Intanagonwiwat, G. Ramesh and E. Deborah, "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks," in *Proceedings of the 6th Annual ACM International Conference on Mobile Computing and Networking*, 2000.

[14] C. Intanagonwiwat, R. Govindan, D. Estfin, J. Heidemann and F. Silva, "Directed diffusion for wireless sensor networking," *IEEE/ACM TRANSACTIONS ON NETWORKING,* vol. 11, no. 1, February 2003.

[15] R. Bergelt, M. Vodel and W. Hardt, "Energy efficient handling of big data in embedded, wireless sensor networks," in *Proceedings of 2014 IEEE Sensors Applications Symposium (SAS)*, 2014.

[16] A. Belfkih, C. Duvallet and B. Sadeg, "A survey on wireless sensor network databases," *Wireless Networks,* no. 8, pp. 4921-4946, 2019.

[17] C. Gröpl, S. Hougardy, T. Nierhoff and H. J. Prömel, "Lower Bounds for Approximation Algorithms for the Steiner Tree Problem," in *Graph-Theoretic Concepts in Computer Science, 27th International Workshop*, 2001.

[18] S. Madden, M. J. Franklin, J. M. Hellerstein and W. Hong, "TAG: a Tiny AGgregation service for ad-hoc sensor networks," *ACM SIGOPS Operating Systems Review,* pp. 131-146, Dezember 2002.

[19] A. Hagberg, D. Schult and P. Swart, " Approximations and Heuristics - Steiner Tree," in *NetworkX Reference*, 2022, pp. 179-181.

[20] S. R. Madden, F. Michael J., J. M. Hellerstein and W. Hong, "TinyDB: An Acquisitional Query Processing System for Sensor Networks," *ACM Transactions on Database Systems,* vol. 30, no. 1, pp. 122-173, März 2005.

[21] Y. Yao and J. Gehrke, "The Cougar Approach to In-Network Query Processing in Sensor Networks," *SIGMOD Record,* no. 31, pp. 9-18, 2002.