Edge Betweenness Centrality: A Novel Algorithm for QoS-based Topology Control over Wireless Sensor Networks

Alfredo Cuzzocrea[a], Alexis Papadimitriou[b], Dimitrios Katsaros[c], Yannis Manolopoulos[b]

[a] *ICAR-CNR and University of Calabria, Cosenza, Italy*
[b] *Department of Informatics, Aristotle University, Thessaloniki, Greece*
[c] *Computer and Communication Engineering Department, University of Thessaly, Volos, Greece*

Abstract

In this paper we propose a novel topology control algorithm, called *Edge Betweenness Centrality* (EBC). EBC is based on the concept of *betweenness centrality*, which has been firstly introduced in the context of *Social Network Analysis* (SNA), and measures the "importance" of each node in the network. This information allows us to achieve high *Quality of Service* (QoS) in wireless sensor networks by evaluating relationships between entities of the network (i.e., edges), and hence identifying different roles among them (e.g., brokers, outliers), thus controlling information flow, message delivery, latency and energy dissipation among nodes. The experimental evaluation and analysis of EBC in comparison to other state-of-the-art topology control algorithms shows that our algorithm outperforms the competitor ones in all observed cases.

Keywords: Betweenness centrality, topology control, wireless sensor networks, graph structure analysis.

Email addresses: `cuzzocrea@si.deis.unical.it` (Alfredo Cuzzocrea), `apapadi@csd.auth.gr` (Alexis Papadimitriou), `dkatsar@inf.uth.gr` (Dimitrios Katsaros), `manolopo@csd.auth.gr` (Yannis Manolopoulos)

1. Introduction

Recent advances in low-power and short-range-radio technology arisen during last few years have enabled a rapid development of wireless sensor networks (WSN). The range of applicability of WSN is very wide, and spans from environmental sensor networks monitoring (environmental) parameters, such as temperature and humidity, to industrial control robotics, from disaster prevention systems to emergency management systems, and so forth. Sensors are tiny, usually battery-operated devices with radio and computing capabilities, which are used to cooperatively monitor physical or environmental conditions.

As regards research issues of sensor networks, several efforts have been done by both the academic and industrial research community, mainly in the context of routing algorithms [1, 2], network coverage aspects [3, 4, 5], storage issues [6, 7] and topology control [8, 9, 10]. The common denominator of all these efforts is represented by the goal of maximizing energy conservation across the network, in order to gain efficacy and efficiency, as maximizing energy conservation corresponds to maximizing network lifetime. For instance, as regards specific data management issues over sensor networks [11], maximizing energy conservation means that multi-step maintenance and query algorithms can be executed over the target sensor network, thus involving in more effective data management capabilities rather than the case of single-step algorithms. Another motivation of the need for energy conservation in sensor networks relies on inherent technological properties of sensors. In fact, sensors are unlikely to be recharged, especially since they may be deployed in unreachable terrains, or, in some cases, they may be disposed after the monitoring application running over the target network ends its execution.

In order to reduce energy consumption, *topology control algorithms* have been proposed in literature [9, 10, 12, 13, 14, 15, 16, 17, 18, 19, 20]. The final goal of these algorithms consists in reasoning-over and managing the topology of the graph modeling the target sensor network in order to reduce energy consumption as much as possible hence increase network lifetime accordingly. A different line of research, which appeared recently, proposes driving sensor network topology control in terms of *Quality of Service* (QoS) requirements [14] over the target sensor network itself. Several QoS-based requirements have been designed and developed in this context, depending on the particular application scenario ranging from real-time video and content provisioning to time-critical control systems, and so forth (see [14] for a complete survey of typical case studies). Given a set of nodes performing a specific task which is critical for the target sensor network application (e.g., sink nodes in environmental sensor networks), the basic idea behind topology control algorithms is to select from the target network appropriate *logical neighbors* of the former nodes, namely a subset of *physical neighbors* of the former nodes that can be used to perform application-specific procedures (e.g., message transmission) without the need of involving the rest of physical neighbors during the execution of these procedures. QoS-based topology control algorithms select the suitable set of logical neighbors such that input QoS requirements can be satisfied.

Inspired by motivations above, in this paper we investigate the problem of QoS-based topology control over homogenous WSN. Given (*i*) a set of wireless nodes in a plane such that nodes have the same transmitting power and bandwidth capacity, and (*ii*) QoS requirements between node pairs, our problem consists in finding a network topology that can simultaneously meet the input QoS requirements and minimize the maximal power utilization ratio of nodes. In particular, in our research QoS requirements are modeled in terms of simple-yet-effective node connectivity, so that message transmission can be ensured (while node connectivity can be preserved in order to ensure correct message delivery), and network lifetime can be increased as much as possible accordingly. In this scenario, avoidance of hotspots also needs to be carefully considered. Therefore, adaptive tasks that depend on the current logical neighbor seem to play the role of most promising strategy to be investigated in order to avoid fast battery depletion.

Looking at deeper details, in our research we propose a *weighted, bidirectional topology control algorithm*, called *Edge Betweenness Centrality* (EBC), and experimentally evaluate this protocol against a set of low complexity, distributed topology control algorithms, namely Gabriel Graph (GG) [21], Relative Neighborhood Graph [22] and Closeness Centrality [23]. Fundamentals of our approach can be found in the conceptual basis drawn by several centrality measures that have been proposed in order to model and evaluate the *importance* of a node in a network [24, 25]. These measures have been initially applied in the context of *Social Network Analysis* (SNA), and later to other areas as well, such as biological networks [26].

Freeman [23, 27] defines the *betweenness* of a node as a possible centrality measure for detecting the importance of that node within the target network, thus achieving the fundamental concept of *betweenness centrality.* This concept founds on the property stating that vertices that occur on many shortest paths between other vertices have higher betweenness than those with lower occurrences. Closeness centrality [23] pinpoints vertices that tend to have short geodesic distances from other vertices with in the network. In network analysis, closeness is preferred over shortest-path length, as closeness gives higher values to more central vertices [23]. Finally, Eigenvector centrality [28] assigns relative scores to all nodes in the network based on the principle that connections to high-scoring nodes provide to the global score of the actual node a higher contribution rather than the one provided by connections to low-scoring nodes. For instance, Google's PageRank [29] is a variant of the eigenvector centrality measure. Our research focuses on a meaningful variation of the betweenness centrality concept, namely *edge betweenness centrality* [24, 30], and its application to the leading context of sensor networks.

Summarizing, the contributions of this paper are the following:

- an innovative weighted, bidirectional topology control algorithm, EBC, and its application to the leading context of sensor networks;
- a comprehensive experimental evaluation of algorithm EBC, and its comparison with a very popular topology control algorithm, GG, on top of the well-known simulation environment *JSim* [31];
- critical analysis and discussion on performance of the two comparison topology-control algorithms, EBC and GG.

The rest of the paper is organized as follows. In Section 2 we discuss related work on topology control algorithms over networks. Section 3 describes in detail algorithm EBC. Section 4 focuses on state-of-the-art distributed and low complexity methods for topology control that are related to our research. Section 5 is devoted to the experimental evaluation and analysis of EBC in comparison to other state-of-the-art topology control algorithms. Finally, Section 6 contains conclusions and future work of our research.

2. Related work

There exists considerable related work addressing topology control issues over networks, even focalizing on QoS-based topology control. As regards studies on topology management for energy conservation in networks, it has been demonstrated that both powering off redundant nodes and lowering radio power while maintaining node connections can contribute to efficient power saving. In light of this assumption, Shen et al. [9] introduced the Local Shortest Path (LSP) algorithm. In the LSP approach, each node makes use of link weights in order to compute the shortest paths between itself and neighboring nodes. Then, all second nodes on these shortest paths are selected as logical neighbors. The final step of algorithm LSP involves in adjusting the power transmission of so-selected logical nodes to save energy.

Li et al. [15] instead propose algorithm Localized Minimum Spanning Tree (LMST), which computes a "power-reduced" network topology by constructing a minimum spanning tree over the network in a fully-distributed manner. The aim of this approach relies in the evidence that the power-reduced network is less energy-consuming than the original network.

EasiTPQ [14] is another QoS-based topology control algorithm. EasiTPQ founds on the assumption that each node in the network has different functionalities during data transmission, e.g., some nodes bear more data relay tasks whereas some other nodes only transmit data generated by themselves. In order to achieve the desired QoS, EasiTPQ schedules as active nodes that are more involved in relaying data tasks rather than generating data flows.

Wattenhofer et al. [19] propose a simple-yet-effective distributed algorithm according to which each node makes local decisions about its transmission power, such that these local decisions then collectively guarantee global connectivity of the network. Specifically, based on directional information, a node grows it transmission power until it finds a neighbor node in every possible direction. The resulting network topology increases network lifetime by reducing transmission power, and, in turn, even traffic interference, thanks to the deriving availability of low-degree nodes. Huang et al. [13] further extend [19] to the case of using directional antennas.

Ramanathan and Rosales-Hain [17] describe a centralized spanning tree algorithm for building connected and bi-connected networks with the goal of minimizing the maximum transmission power for each node. Two optimal, centralized algorithms, namely CONNECT and BICONN-AUGMENT, are proposed for the case of static networks. Both are greedy algorithms, and resemble Kruskal's minimum cost spanning tree algorithm [32]. For the case of hoc wireless networks, two distributed heuristics hare proposed, namely LINT and LILT. However, these heuristics do not guarantee network connectivity.

Jia at al. [20] focus the attention on the problem of determining a network topology able to meet input QoS requirements in terms of end-to-end delay and bandwidth. The proposed scheme adopts an optimization criterion whose goal is to minimize the maximum per-node power consumption. In [20], authors demonstrate that, when network traffic is "splittable", a sub-optimal solution can be achieved by means of linear programming techniques.

Finally, alternative approaches to the topology control problem over sensor networks have been proposed recently. [33, 34] are significant instances of these classes of innovative topology control algorithms, where the topology aspect is addressed from a different but relevant perspective. In more detail, [33, 34] basically suggest to exploit the sensor motion to adaptively propagate information based on *local conditions* (such as high placement concentrations), so that the mobile sink gradually "learns" the network and accordingly optimizes its motion as to collect data faster.

3. Edge Betweenness Centrality: a novel topology control protocol for sensor networks

During past years, vertex betweenness has been studied in the vest of a measure of the centrality and influence of nodes in networks [27, 23]. Given a node v_i, vertex betweenness is defined as the number of shortest paths between pairs of nodes that run through v_i. Vertex betweenness is a measure of the influence of a node over the information flow among nodes of the network, especially in scenarios such that information flowing over the target network primarily follows shortest available paths.

In order to compute betweenness centrality, Brandes [35] proposes an efficient backwards algorithm which starts from leaf nodes of a tree of shortest paths and progressively accumulates the leaf-nodes' betweenness values moving bask towards the root node of the tree.

Girvan-Newman algorithm [24] extends the definition of betweenness centrality from network vertices to network edges, via introducing the concept of *Edge Betweenness* (EB). Let $G = \langle V, E \rangle$ be a connected undirected graph, and v_i and v_j two nodes in G, respectively. Let $\sigma_{v_i v_j}$ denote the number of shortest paths between nodes v_i and v_j. Let $\sigma_{v_i v_j}(e)$ denote the number of shortest paths between v_i and v_j which go through $e \in E$. Betweenness centrality of an edge $e \in V$, denoted by $EB(e)$, is defined as follows:

$$EB(e) = \sum_{v_i \epsilon V} \sum_{v_j \epsilon V} \frac{\sigma_{v_i v_j}(e)}{\sigma_{v_i v_j}} \tag{1}$$

In its original implementation [30], which focuses on unweighted, undirected networks, EB analysis makes use of the algorithm breadth-first search (BFS). Girvan-Newman algorithm [24] works in the opposite way. Instead of trying to construct a measure that determines edges that are the "most central" for network communities, it focuses on edges that are the "least central" for network communities, i.e., edges that are "most between" for network communities. Communities are detected by progressively removing edges from the original graph, rather than by adding the strongest edges to an initially empty network. In our research, we do not use the centrality measure to find communities but instead to select the most important edges, energy-wise, to propagate messages.

Specifically, steps that are used to compute the edge betweenness centrality index are the following:

1. compute shortest paths through the network by means of Dijkstra's algorithm [36];
2. for each edge, compute the edge betweenness centrality index like in [30], but instead of un-weighted edges use the average energy of the two connecting nodes as edge weight.

Based on the edge betweenness centrality index, our algorithm EBC selects logical neighbors of actual node based on the following rules:

- for each node, logical neighbors must cover the 2-hop node neighborhood;
- 1-hop neighbors with the highest-scoring betweenness centrality index are selected.

Moreover, in order to avoid hotspots, our algorithm recalculates the edge betweenness centrality index based on the corresponding energy levels of each node, therefore selecting different edges to be part of the logical neighborhood of each node.

4. Distributed and low complexity competing methods for topology control

In this section we present in more details some popular, distributed methods for topology control in wireless sensor networks, that comprise the basic competitors of our proposed EBC algorthm.

4.1. Topology control with the Gabriel Graph

Gabriel Graph has been introduced by Gabriel and Sokal in [21]. Formally, given a graph $G = \langle V, E \rangle$ and two vertices v_1 and v_2 in V, we say that v_1 and v_2 are adjacent if the closed disc of diameter v_1v_2 does not contain other vertices of V. In the context of sensor networks, we extend the basic adjacency concept above and we say that a sensor node s_i is connected with a sensor node s_j, who lies within the s_i's transmission range, if there not exist another node s_k which is contained by the closed disc of diameter s_is_j. This simple-yet-effective method is used by algorithm GG to find logical neighbors of a given sensor node.

In more detail, in our JSim-based experimental framework, logical neighbors of a given sensor node are found by algorithm GG according to the following steps:

1. each sensor node broadcasts its location – at the end, every node in the sensor network knows its neighbors and their locations;
2. each sensor node s_i determines its logical neighbor set L_i by computing the closed discs of diameters equal to the distance between the location of s_i and each other physical node belonging to the s_i's physical neighborhood set P_i – for each physical neighbor s_j in P_i, if the disc of diameter s_is_j does not contain other physical neighbors of P_i then s_j becomes a logical neighbor of s_i.

4.2. Topology control with the Relative Neighborhood Graph (RNG)

The relative neighborhood graph (RNG) of a point set is a straight line graph that connects two points from the point set if and only if there is no other point in the set that is closer to both points than they are to each other. A triangu- lation of a point set is a maximal set of nonintersect- ing line segments (called edges) with vertices in the point set.

The relative neighborhood graph of a graph G = (V, E), denoted by RNG(G), is the set of all edges uv ε E such that there is no vertex or point w where $uw \ \varepsilon \ E$, $wv \ \varepsilon \ E$ and $||uw|| < ||uv||$ and $||wv|| < ||uv||$.

4.3. Topology control with the Closeness Centrality (CC)

In graph theory closeness is a centrality measure of a vertex within a graph. Vertices that are 'shallow' to other vertices (that is, those that tend to have short geodesic distances to other vertices with in the graph) have higher closeness. Closeness is preferred in network analysis to mean shortest-path length, as it gives higher values to more central vertices, and so is usually positively associated with other measures such as degree.

5. Experimental evaluation and analysis

In order to evaluate the performance of the proposed EBC topology control protocol, we set up a framework that simulates the basic factors of a wireless environment and implemented in this framework the competitors described in section 4, namely GG, RNG, and CC.

5.1. Simulation model

In our experimental framework, we have developed a simulation model based on JSim, a well-known Java-based simulation environment for numerical analysis [31]. In particular, in our simulation environment, the AODV routing protocol [37] is deployed within the reference WSN. Also, we use IEEE 802.11 as the MAC protocol and the free space model as the radio propagation model. Wireless bandwidth is assumed to be 2 Mbps.

We performed a large number of experiments on top of various sensor network topologies, and by ranging several experimental parameters, but for the interest of space, here we present a subset of our experimental results. Table 1 summarizes the simulation parameters.

Parameter	**Values**
sensor node number	500, 750, 1000
terrain size	400 x 400
radio range	14m, 17m
initial energy charge	10 Joules
transmission energy	0.001 Joules
wireless bandwidth	2 Mbps
λ_e, λ_q	0.128, 0.256, 0.512, 0.768

Table 1: Simulation parameters.

The simulation details are as followed:

- The simulation time was 300 seconds. The records were produced during the first half of the simulation time, whereas the queries were sent during the second half.

- While the record is propagated in the network, its TTL value (measured in hops) is decreased by 1 each time the record is stored at a sensor. The initial TTL value is 10.

- The events and queries are generated according to a Poisson distribution with the rates λ_e and λ_q taking the values $0.128, 0.256, 0.512$ and 0.768.

- The queries originate at sensors whose geographical position follows the Zipfian distribution, i.e., some sensor generate more queries than others.

5.2. Experimental results

As stated in previous sections, topology control algorithms over sensor networks try to minimize the energy consumption of nodes by transmitting data to a subset of a node's physical neighbors. Therefore, given the actual node, the first step deals with the issue of finding node's physical neighbors. Then, topology control algorithms are applied in order to select the subset of logical neighbors that can propagate messages throughout the network without any data loss, neither involving all the effective physical neighbors.

Our experimental analysis focuses on the comparison between algorithms EBC, GG, RNG and CC in terms of logical neighbors found and energy consumption that is needed to propagate messages through logical neighbors. For each algorithm, we also analyze the impact of a change in network density on algorithm's performance.

Figure 1 shows the overall number of physical neighbors that exist in the network for 500, 750 and 1000 nodes, respectively. The increase in the number of physical neighbors is due to the increase in the sensor transmission radius from 14 to 17 meters. This means that each sensor node can communicate with nodes that exist in its wider vicinity. For a radius of $14m$, the number of physical neighbors are $1298, 2640$ and 4488, respectively. For a radius of $17m$, we instead have: $1958, 3984$, and 6797.

Figure 2 illustrates the average number of physical neighbors of each node in the network, for different sizes of the sensor network. In the first case, i.e., a network with 500 nodes, the average number of physical nodes per-sensor-node is 2.4 for a radius of $14m$ and 3.7 for a radius of $17m$. The respective numbers for a network with 750 nodes are: 3.5 ($14m$ radius) and 5.3 ($17m$ radius). Finally, for a network with 1000 nodes, retrieved numbers

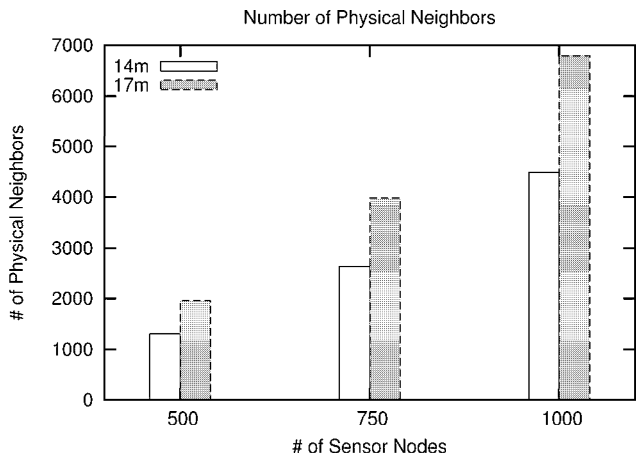

Figure 1: Number of physical neighbors.

are: 4.4 (14m radius) and 6.7 (17m radius). Notice that in all the cases retrieved numbers are the same for all algorithms since they are not applied to the initial step that finds the physical neighbors of each node.

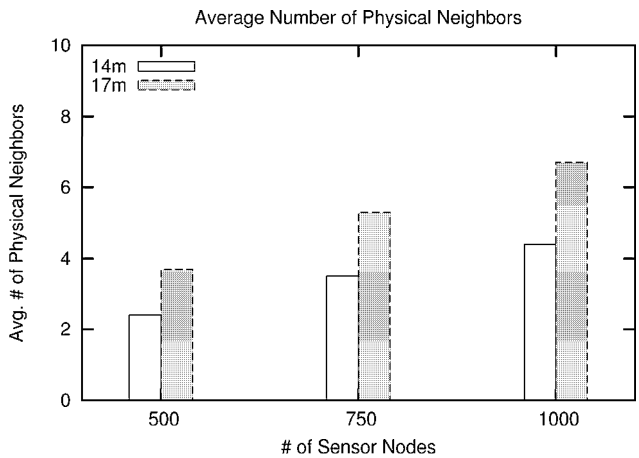

Figure 2: Average number of physical neighbors per-sensor-node.

Moving the attention on the proper experimental comparison of the four investigated topology-control algorithms (i.e., EBC, GG, RNG and CC), Figure 3 shows the overall number of logical neighbors found after each algorithm has been applied to each network setting with different size (500, 750 and 1000 nodes) when the radius is set to 14m. As shown in the Figure, GG and RNG find the most logical neighbors, starting from 1086 and 951 nodes for the 500 nodes setting and reaching 3742 and 3145 for the 1000 nodes setting respectively. EBC and CC on the other hand perform similarly but EBC finds the least amount of logical neighbors between the two. The difference between the two sets of algorithms increases as the number of sensor nodes in the network increases. For 1000 nodes, algorithm GG found 3742 logical neighbors, whereas algorithm EBC 1513 logical neighbors only.

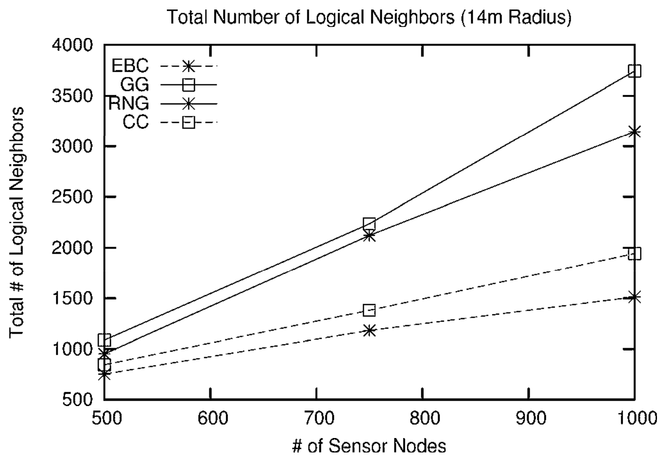

Figure 3: Number of logical neighbors found (radius = $14m$).

Figure 4 shows the performance of the algorithms in terms of average logical neighbors found per-sensor-node, still with a radius equal to $14m$. As clearly follows from Figure 4, algorithm EBC delivers about the same average number of logical neighbors per-sensor-node, i.e., about 1.5, irrespectively of the size of the sensor network. On the other hand, algorithms GG and RNG do not perform as well, since the average number of logical neighbors per-sensor-node ranges from 2 up to 3.7 for GG and 1.9 to 3.1 for RNG. CC does follow the same pattern as EBC but it still finds a smaller average of logical neighbors than EBC, at a range of 1.7 to 1.9.

It should be noted that, in our experimental analysis, we overall consider two sets of algorithms that perform very differently. Algorithms GG and RNG, which belong to the first set, expose some limitations in maintaining the number of logical neighbors small and, as a consequence, the average number of logical neighbors increases significantly. The main reason of this phenomenon is that performance of algorithms GG and RNG strongly depends on the geodesic placement of sensors. Increasing the number of sensors in an $400m \times 400m$ area will lead to an increase of physical neighbors, which, in turn, will lead to a relatively smaller increase of logical neighbors. Also, EBC is actually a subset of GG, hence it is reasonable for the two algorithms to perform similarly. On the other hand, algorithms CC and EBC, which belong to the second set, retrieve the significance of a sensor node in a two-hop neighborhood. As a consequence, even if the number of sensors in the terrain increases, the average number of logical sensors in the newly created two-hop neighborhood roughly ranges on the same interval values.

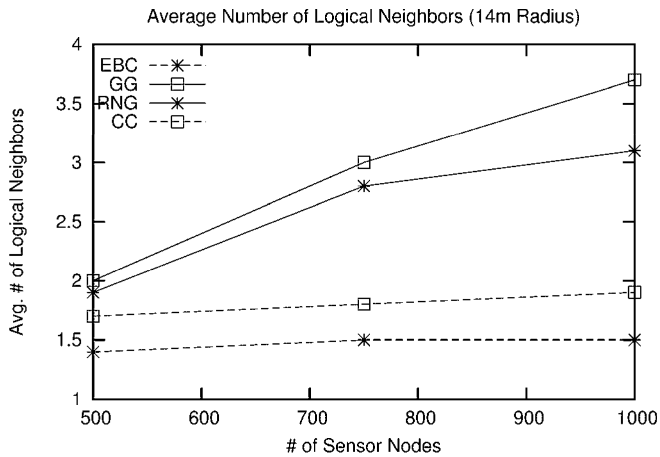

Figure 4: Average number of logical neighbors per-sensor-node (radius = $14m$).

The same experiment is performed for a radius of $17m$. Figure 5 shows the results obtained for this setting. As shown in the figure, when radius increases the difference between the two algorithms' performance is even more noticeable. In fact, the number of logical neighbors found by algorithm GG ranges from 1639 (500 nodes) to 5648 (1000 nodes). The respective numbers for algorithm EBC range from 1014 (500 nodes) to 2052 (1000 nodes). RNG and CC values lie in between the previous values. Therefore, it clearly follows that EBC outperforms the other algorithms even under this experimental analysis perspective.

Figure 6 confirms the superiority of algorithm EBC over the rest of the algorithms in terms of the average number of logical neighbors found per-sensor-node, still with a radius equals to $17m$. It should be noticed again that algorithm EBC remains practically insensitive to the increase in the number

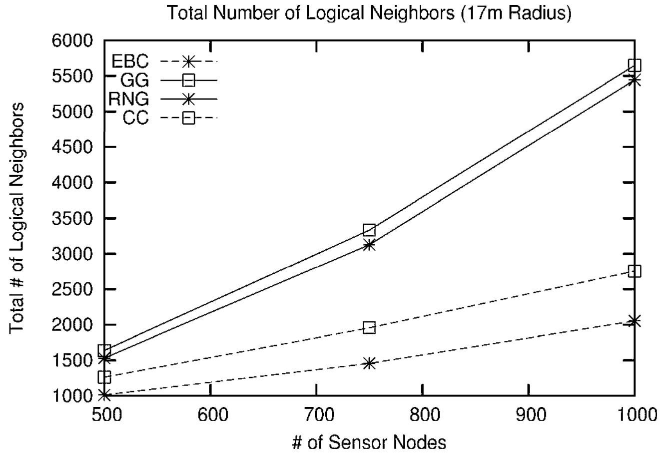

Figure 5: Number of logical neighbors found (radius = $17m$).

of sensor nodes and provides an average number of 2 logical neighbors per-sensor-node throughout the simulation. CC performs similarly but still finds more logical neighbors on average than EBC. On the other hand, algorithms GG and RNG perform poorly with an average number of logical neighbors found per-sensor-node ranging from 3.1 to 5.6 for GG and 3 to 5.4 for RNG.

Similarly to the previous analysis, we again notice that the two sets of algorithms (i.e., GG and RNG, and CC and EBC) perform differently, with the evidence that GG and RNG are unable to effectively and efficiently cope with the increase of the number of sensors placed in the terrain. Contrary to this, CC and EBC focus on the two-hop neighborhood of each sensor node which remains fairly stable throughout the experiment.

Looking at energy consumption minimization, the main goal of topology control algorithms, Figure 7 shows the energy consumption per-node needed to propagate a message to logical neighbors, when the radius is set to $14m$. Again, algorithm EBC requires an almost unchanged amount of energy to this goal, i.e., about 0.0015 Joules, whereas algorithm GG requires an amount of energy ranging from 0.0020 (500 nodes) to 0.0037 (1000 nodes) Joules to perform the same operation. RNG is also not effective at all at reserving energy, just like GG, while CC performs better but not as good as EBC.

Figure 8 shows the results for the same experiment when the radius is set to $17m$. Even in this experimental analysis, algorithm EBC outperforms the other algorithms with a transmission energy consumption per-node equal

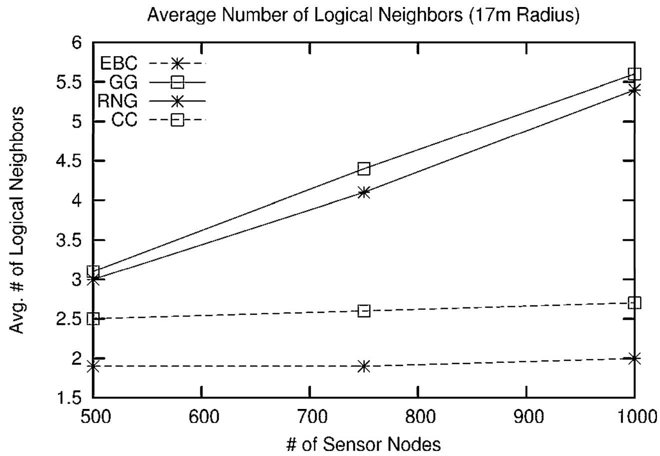

Figure 6: Average number of logical neighbors per-sensor-node (radius = 17m).

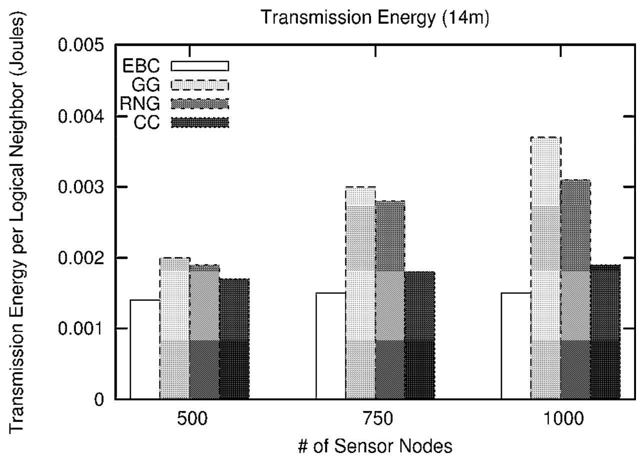

Figure 7: Transmission energy consumption per-node (radius = 14m).

to 0.002 Joules. Indeed, algorithms GG and RNG significantly increase the energy requirement ranging from about 0.0031 to 0.0056 Joules. CC provides better energy conservation but still not better than EBC.

The high energy consumption of GG and RNG can be explained by the fact that sensors have more logical neighbors when these algorithms are employed. At a practical level, this means that packets must be sent to a larger number of sensor nodes, hence leading to significant energy consumption. On the other hand, CC and EBC do not impose such a burden to the transmission of packets, and, as a consequence, expose a better energy efficient.

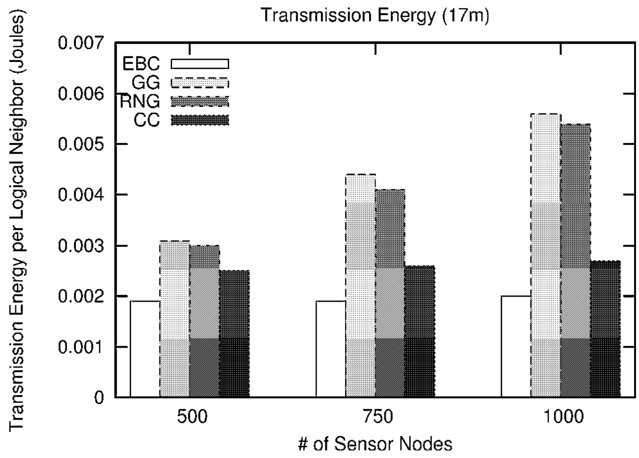

Figure 8: Transmission energy consumption per-node (radius = 17m).

Apart from the number of logical neighbors and the transmission energy, two other metrics that play an important role in the evaluation of the topology control algorithms are latency and hit-ratio. Latency is considered to be the time passed between issuing a query and receiving an answer to it. Obviously, the lesser the latency the better the network response to queries. Hit-ratio on the other hand, is considered to be the ratio of answers received over the total number of queries that were produced.

The first experiment is performed for a setting of 500 nodes and a radius of $17m$. As shown in Figure 9, GG and RNG perform similarly with CC providing smaller latency values than both of them. EBC outperforms the other algorithms, producing latency values ranging from 81 milliseconds to 90 milliseconds. The same results occur at the second experiment where the number of sensors inside the network is increased to 1000. Results are shown in Figure 10. Again, EBC shows its superiority by producing latency values ranging from 26 milliseconds to 55 milliseconds. Observe that the average latency is increased when we increase the number of sensors inside the network. This is because collisions occur inside the network when multiple sensors try to communicate at the same time and because it takes longer for the record to be propagated through a denser sensor network.

The same principle applies when we increase the number of packets inside the network. The more packets occur, the more (packet) collisions occur, hence packets flow throughout the network more problematically. Another critical evidence that is related to the previous phenomenon concerns with the number of logical neighbors inside the network. When the target sensor network employs GG or RNG as topology control algorithm, nodes expose a significantly-larger number of logical neighbors, as shown by previous experiments, and, as a consequence, an higher packet collision probability is observed. Contrary to this, CC and EBC expose a fairly-stable number of logical neighbors and, as a consequence, the packet collision probability reduces significantly.

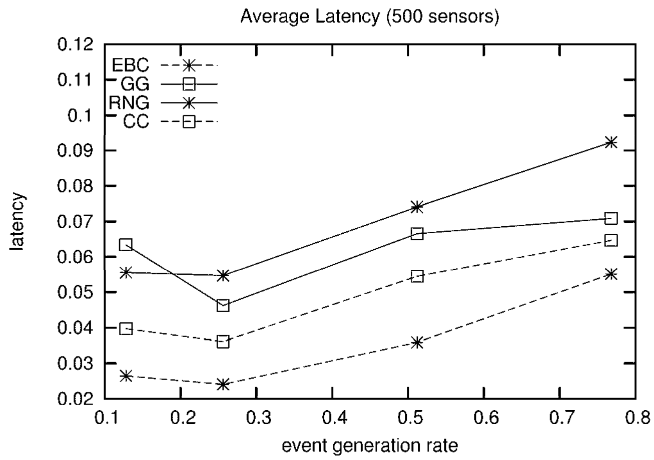

Figure 9: Average latency per-node (500 nodes, radius = 17m).

Finally, the hit-ratio percentage metric measurements are essential to application scenarios, such as a forest fire for example, where the need to obtain answers to our queries is imperative. Figure 11 shows the hit-ratio percentage obtained for all four algorithms when the sensor network consists of 500 nodes. EBC performs the best with a lowest hit-ratio of 83% and a highest of 94%. CC obtains the second best results, while GG and RNG algorithms perform the worst with percentages ranging from about 50% to 65%.

In order to measure performance in a bigger network, we increase the number of sensor nodes to 1000. Once again, EBC outperforms the other algorithms, even though the hit-ratio values are decreased compared to the values obtained in the 500 nodes setting. This is because more sensor nodes exist inside the network and therefore more collisions occur, making it difficult for the messages to reach their destination. EBC performs at an average

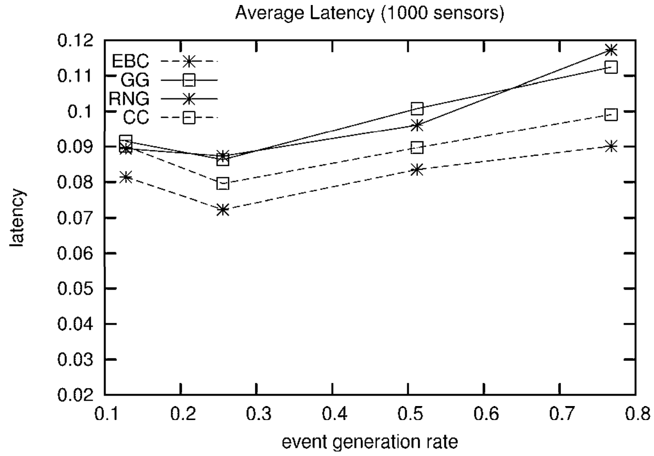

Figure 10: Average latency per-node (1000 nodes, radius = 17m).

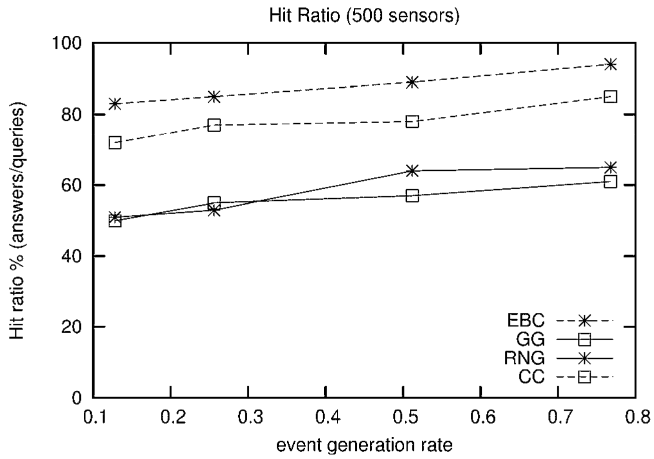

Figure 11: Hit Ratio percentage, i.e., answers/queries (500 nodes, radius = 17m).

of about 70% while GG, RNG and CC perform at an average of 33%, 41% and 60%, respectively. The TTL value plays an important role in this case, since it is decreased at each hop. Therefore the larger the number of hops that the message travels, the less possible it is to reach its destination. The experimental results are thus convergent in the sense that both algorithms GG and RNG create more logical neighbors inside the network than CC and EBC. As a consequence, packets must flow longer distances throughout the network, hence an higher (packet) collision probability is observed.

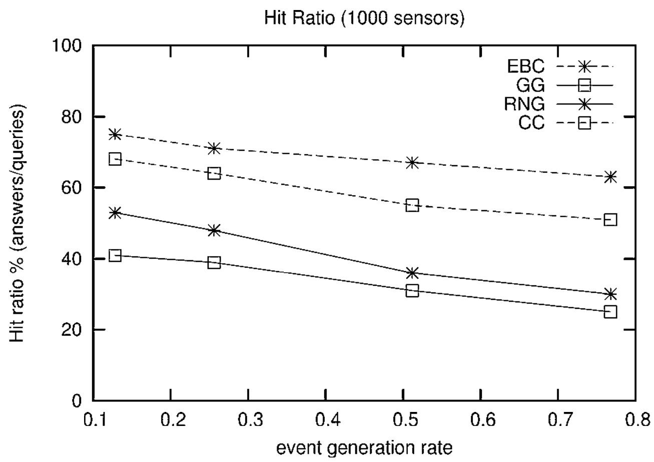

Figure 12: Hit Ratio percentage, i.e., answers/queries (1000 nodes, radius = $17m$).

6. Conclusions and future work

Betweenness is a centrality measure for networks that has been initially studied in the context of SNA. This measure states that vertices that occur on many shortest paths between other vertices have higher betweenness than those with lower occurrences. Therefore, nodes with high betweenness are selected as nodes able to control the overall information flow within the network. Topology control algorithms aim at providing high QoS by maximizing network lifetime and ensuring message delivery. Inspired by these motivations, in this paper we have proposed a novel topology control algorithm for sensor networks, EBC, which exploits the edge betweenness centrality concept to ensure high QoS throughout the network. Our scheme can be effectively combined with load balancing techniques such as those described in [38]. Also, we performed a comprehensive campaign of experiments where we compared the performance of algorithm EBC with the performance of algorithms GG, RNG and CC under several perspectives of analysis. Experimental results have clearly demonstrated the superiority of algorithm EBC over the other algorithms, in terms of logical neighbors found, energy consumption, latency and hit-ratio.

As future work, we plan to devise alternative centrality measures for networks, looking at the wide literature available on the topic, and experimentally compare these novel measures to edge betweenness centrality. Apart from number of logical neighbors found, transmission energy consumption and scalability, which have been investigated in this paper, in the future experimental analysis we will focus on other interesting experimental parameters that need more research efforts, such as message latency and message delivery.

References

[1] W. Heinzelman, A. Chandrakasan, H. Balakrishnan, Energy-efficient communication protocol for wireless microsensor networks, in: Proceedings of the Hawaii International Conference on System Sciences (HICSS), Vol. 8, 2000, pp. 8020–8029.

[2] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, F. Silva, Directed diffusion for wireless sensor networking, IEEE/ACM Transactions on Networking 11 (1) (2003) 2–16.

[3] M. T. Thai, F. Wang, D. H. Du, X. Jia, Coverage problems in wireless sensor networks designs and analysis, International Journal on Sensor Networks 3 (3) (2008) 191–200.

[4] B. Liu, P. Brass, O. Dousse, P. Nain, D. Towsley, Mobility improves coverage of sensor networks, in: Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC), 2005, pp. 300–308.

[5] K.-P. Shih, H.-C. Chen, C.-M. Chou, B.-J. Liu, On target coverage in wireless heterogeneous sensor networks with multiple sensing units, Journal of Network and Computer Applications 32 (4) (2009) 866–877.

[6] G. Mathur, P. Desnoyers, D. Ganesan, P. Shenoy, Ultra-low power data storage for sensor networks, in: Proceedings of the ACM International Conference on Information Processing in Sensor Networks (IPSN), 2006, pp. 374–381.

[7] B. Sheng, Q. Li, W. Mao, Data storage placement in sensor networks, in: Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC), 2006, pp. 344–355.

[8] Y. Manolopoulos, D. Katsaros, A. Papadimitriou, Topology control algorithms for wireless sensor networks: A critical survey, in: Proceedings

of the International Conference on Computer Systems and Technologies (CompSysTech), 2010.

[9] Y. Shen, Y. Cai, X. Xu, A shortest-path-based topology control algorithm in wireless multihop networks, ACM SIGCOMM Computer Communications Review 37 (5) (2007) 29–38.

[10] J. Liu, B. Li, Distributed topology control in wireless sensor networks with asymmetric links, in: Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM), 2003, pp. 1257–1262.

[11] A. Cuzzocrea (Ed.), Intelligent Techniques for Warehousing and Mining Sensor Network Data, IGI Global, 2009.

[12] G. Hackmann, O. Chipara, C. Lu, Robust topology control for indoor wireless sensor networks, in: Proceedings of the ACM Conference on Embedded Network Sensor Systems (SenSys), 2008, pp. 57–70.

[13] Z. Huang, C. chung Shen, C. Srisathapornphat, C. Jaikaeo, Topology control for ad hoc networks with directional antennas, in: Proceedings of the IEEE International Conference on Computer Communications and Networks (ICCN), 2002, pp. 16–21.

[14] W. Liu, L. Cui, X. Niu, W. Liu, EasiTPQ: QoS-based topology control in wireless sensor network, Journal of Signal Processing Systems 51 (2) (2008) 173–181.

[15] N. Li, J. C. Hou, L. Sha, Design and analysis of an MST-based topology control algorithm, IEEE Transactions on Wireless Communications 4 (3) (2005) 1195–1206.

[16] J. Pan, Y. T. Hou, L. Cai, Y. Shi, S. X. Shen, Topology control for wireless sensor networks, in: Proceedings of the IEEE/ACM International Conference on Mobile Computing and Networking (MOBICOM), 2003, pp. 286–299.

[17] R. Ramanathan, R. Rosales-Hain, Topology control of multihop wireless networks using transmit power adjustment, in: Proceedings of the IEEE Conference on Computer Communications (INFOOCOM), 2000, pp. 404–413.

[18] Y.-C. Tseng, Y.-N. Chang, B.-H. Tzeng, Energy-efficient topology control for wireless ad hoc sensor networks, in: Proceedings of the International Computer Symposium, 2002, pp. 27–37.

[19] R. Wattenhofer, L. Li, P. Bahl, Y.-M. Wang, Distributed topology control for power efficient operation in multihop wireless ad hoc networks, in: Proceedings of the IEEE Conference on Computer Communications (INFOOCOM), 2001, pp. 1388–1397.

[20] X. Jia, D. Li, D.-Z. Du, QoS topology control in ad hoc wireless networks, in: Proceedings of the IEEE Conference on Computer Communications (INFOCOM), 2004, pp. 1264–1272.

[21] R. K. Gabriel, R. R. Sokal, A new statistical approach to geographic variation analysis, Systematic Zoology 18 (3) (1969) 259–278.

[22] G. Toussaint, The Relative Neighborhood Graph of a finite planar set, Pattern Recognition 12 (1980) 261–268.

[23] L. C. Freeman, Centrality in social networks: Conceptual clarification, Social Networks 1 (3) (1979) 215–239.

[24] M. Girvan, M. E. J. Newman, Community structure in social and biological networks., Proceedings of the National Academy of Sciences U.S.A. 99 (12) (2002) 7821–7826.

[25] U. Brandes, On variants of shortest-path betweenness centrality and their generic computation, Social Networks 30 (2) (2008) 136–145.

[26] J. Yoon, A. Blumer, K. Lee, An algorithm for modularity analysis of directed and weighted biological networks based on edge-betweenness centrality, Bioinformatics 22 (24) (2006) 3106–3108.

[27] L. C. Freeman, A set of measures of centrality based on betweenness, Sociometry 40 (1) (1977) 35–41.

[28] P. Bonacich, Factoring and weighting approaches to status scores and clique identification, Journal of Mathematical Sociology 2 (1) (1972) 113–120.

[29] S. Brin, L. Page, R. Motwani, T. Winograd, PageRank citation ranking: Bringing order to the Web, Tech. Rep. 1999-66, Computer Science Department, Stanford University (1999).

[30] M. E. J. Newman, M. Girvan, Finding and evaluating community structure in networks, Physical Review E 69 (2).

[31] A. Sobeih, J. C. Hou, L.-C. Kung, N. Li, H. Zhang, W.-P. Chen, H.-Y. Tyan, H. Lim, J-Sim: A simulation and emulation environment for wireless sensor networks, IEEE Wireless Communications magazine 13 (4) (2006) 104–119.

[32] J. B. Kruskal, On the shortest spanning subtree of a graph and the Traveling Salesman Problem, Proceedings of the American Mathematical Society 7 (1) (1956) 48–50.

[33] C. M. Angelopoulos, S. Nikoletseas, Fast sensory data collection by mobility-based topology exploration, in: Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM), 2009, pp. 1–6.

[34] C. M. Angelopoulos, S. Nikoletseas, Accelerated sensory data collection by greedy or aggregate mobility-based topology ranks, in: Proceedings of the ACM Symposium on Performance Evaluation of Wireless Ad hoc, Sensor, and Ubiquitous Networks (PE-WASUN), 2009, pp. 63–70.

[35] U. Brandes, A faster algorithm for betweenness centrality, Journal of Mathematical Sociology 25 (2001) 163–177.

[36] E. W. Dijkstra, A note on two problems in connexion with graphs, Numerische Mathematik 1 (1) (1959) 269–271.

[37] C. E. Perkins, E. Royer, Ad hoc On-demand Distance Vector routing, in: Proceedings of the IEEE Workshop on Mobile Computing Systems and Applications, 1999, pp. 90–100.

[38] P. H. Pathak, R. Dutta, Using centrality-based power control for hotspot mitigation in wireless networks, in: Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM), 2010.