The Impact of Clustering on the Average Path Length in Wireless Sensor Networks

Azrina Abd Aziz† Y. Ahmet Şekercioğlu*
*Department of Electrical and Computer Systems Engineering, Monash University, Australia
†Department of Electrical and Electronic Engineering, Universiti Teknologi Petronas, Malaysia

***Abstract*—Clustering algorithms have been widely used in wireless sensor networks for virtual backbone construction. They organize the nodes into smaller groups and form a structured topology allowing more efficient bandwidth usage and battery consumption. As the clustering algorithms are usually used for routing, it is crucial to measure the efficiency of the generated backbone in information transport. Failure to do so will impact the routing performance and reduce the reliability of the system.**

This paper investigates whether the backbone formed by clustering algorithms is able to preserve the routing paths of the network. This property is evaluated by comparing the performance of several clustering algorithms with respect to the average path length. In order to obtain accurate results, the performance is investigated under different network sizes as well as network densities.

I. Introduction

Wireless sensor networks (WSNs) are self-organizing and autonomous networks composed of sensor nodes typically designed for event monitoring. These sensor nodes are equipped with data processing, sensing, storage and communication capabilities which make them suitable for numerous applications ranging from military, civil, industrial to health. The nodes are built to be low-cost and small-size; allowing them to be used in large numbers to accomodate for any failures during deployment. The networks do not require physical infrastructure and operate wirelessly, thus resulting in minimum set-up and maintenance costs.

WSNs have dynamic network topologies which need frequent updates in the case of hardware failures, mobility or energy depletion. Also, the small-size nodes imply that their energy, processing and storage capabilities are severely constrained. Due to these characteristics, efficient techniques that require careful resource management are crucial to support the networks. Clustering technique is among the most proposed solution for this problem.

In WSNs, network clustering is used to construct a temporary infrastructure to support various tasks including routing. It organizes the network into a hierarchical structure to achieve ease of network management, minimum network maintenance and reduction in communication overheads. The basic concept of clustering is to organize sensor nodes into groups thereby offering the network with a logical organization. Each group must contain at least one leader called clusterhead node which is assigned to special tasks while the remaining nodes become the non-clusterhead nodes. The non-clusterhead nodes utilize the clusterhead nodes for data forwarding. This reduces the energy associated with transmission and improves the overall network lifetime. As the clusterheads are loaded with various tasks such as data processing and aggregation, they are most likely to experience energy depletion. This problem can be solved by rotating the role of the clusterhead among nodes.

In order to support routing, the backbone formed by the clustering algorithms must preserve the average path length property of the original topology. In a WSN, average path length is defined as the average shortest path between a node and sink. The shortest path length between a sensor node and a sink node indicates that nodes can relay packets via a shorter path, hence less energy is consumed. The absence of a path between a node and the sink indicates that the network is partitioned.WSNs are commonly known to be highly exposed to radio-links failures such as signal attentuation, radio interference and fading. If a link failure breaks the shortest path, a new path should be available to support the routing or else the network will be disconnected. However, a new path that is longer than the shortest path will result in additional energy consumption.

Minimizing the backbone size to reduce the network overhead is a common objective of the clustering algorithms. Links that initially exist in the original topology prior to the application of clustering might be removed hence, limiting the number of paths to traverse. In this paper, we make an effort to investigate the existence of the average path length in the backbone and study how much it is affected by the clustering. The ability of the algorithms to maintain this property will greatly impact the routing performance as well as the energy consumption. It is interesting to find out whether the investigated path length is in close agreement with the path length of the original topology. In this paper, a number of leading clustering algorithms [1]–[3] were chosen to collect this information and they were simulated under various realistic network topologies using the OMNeT++ simulator [4].

The rest of this paper is organized as follows. Section II presents the background on the clustering techniques. Section III describes the assumptions and implementation of the work. Section V provides the simulation results. Finally, Section VI concludes our work and discusses our future work.

II. Background of Clustering Algorithms

Various clustering algorithms have been surveyed in [5]. These algorithms can be classified into two approaches; dis-

tributed or centralized. Distributed approaches can deal with the dynamic nature of the networks as supposed to centralized approaches, in which they can quickly adapt to any changes in the links between nodes. Centralized approaches on the other hand rely on the assumption that the global information is available when gathering information from all nodes in the network. As a result, they require a significant amount of message overhead.

There are various cluster-based algorithms such as LEACH [6], HEED [7], DSBCA [8] and PEGASIS [9] which are designed for balancing the load and extending the network lifetime. Another type of cluster-based algorithms is based on the dominating set concept which is the focus of this paper. We give the definition of the dominating set concept in Section II-A before discussing the leading dominating set algorithms.

A. Dominating Set (DS) and Connected Dominating Set (CDS) Terminologies

A dominating set (DS) is defined as a subset $S \subseteq V$ if and only if every node in the graph $G = (V, E)$ is either in S or at least one-hop away from a node in S. A dominating set of G which induces a connected subgraph of G forms a connected dominating set (CDS). Figure 1 illustrates this concept. In a network, a set of nodes is defined as a dominating set (DS) if all nodes are either in the set or have a neighboring node in the set. To create the network backbone, the dominating set must remain connected in the network. This connected DS is referred to a CDS.

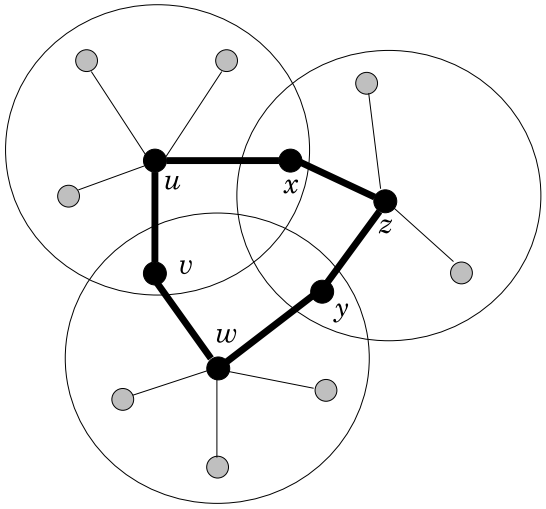


Fig. 1. Cluster formation using the dominating set and connected dominating set concepst.

B. Related CDS based Algorithms

Wu et al. [10] proposes a simple distributed and localized connected dominating set (CDS) algorithm. This algorithm operates in two phases using a marking process. At the beginning of the first phase, all nodes are unmarked. If these nodes have two unconnected neighbors they are then marked as dominators to form the CDS. Since this rule is a greedy method, the size of the generated CDS is typically not minimized. To reduce the CDS size, two pruning rules are introduced in the second phase to remove the redundant CDS. The pruned nodes which used to be the CDS nodes will become the non-CDS nodes. The applied pruning rules are (i) eliminate node u if its neighbor has higher ID and can cover all of its neighbors or (ii) remove node u if it has two connected neighbors with higher ID which can cover all of its neighbors. The algorithm is popular due to its simplicity and quick formation of the CDS. PACDS outperforms classic CDS algorithm in [11]. This algorithm is later expanded to power aware CDS (PACDS) [1] algorithm to extend the network lifetime and further minimize the CDS size. It is common for the CDS nodes to be overloaded with various tasks and they usually are the first to experience energy depletion. To solve this limitation, the role of the CDS is distributed fairly among nodes with higher residual energy. PACDS operates in two phases. The first phase involves the marking rule similar to the one in [10]. The second phase on the other hand introduces several pruning rules that consider new parameters such as number of neighbors(node degree) and energy level when forming the CDS. Although both algorithms have the ability to converge fast, the amount of exchanged messages involved is significant, thus it may quickly deplete the energy reserves in the network.

Similar work to the PACDS is conducted by Yuanyuan et al. [2]. The authors introduce energy-efficient CDS (ECDS) algorithm to address the energy constraints and the size of the backbone. Unlike the algorithm in [10], the ECDS uses a coloring technique to differentiate various role of the nodes. The formation of the CDS involves two phases. The first phase builds a dominating set called a maximal independent set (MIS) while the second phase identifies gateway nodes to join the MIS. The selection of the MIS and gateways is based on the node degree and node's energy reserve. Our work in [12] shows that in dense network, the ECDS can generate a smaller CDS size compared to PACDS at the expense of high message complexity associated with acquiring neighbor's weight and updating nodes' status.

The aims of SPSI [3] are twofold: form a small set of CDS and minimize the message overhead. Contradictory to PACDS and ECDS, SPSI forms a CDS in a single phase. SPSI classifies nodes into two categories; dominators and dominatees. Dominators represent the CDS while the dominatees represent the non-CDS nodes in the network. Each dominator uses a greedy approach when selecting the CDS nodes among its one-hop neighbors. The chosen dominator neighbors then continue the search of dominators among their one-hop neighbors. This process is repeated until all nodes become dominators or have been covered by at least one dominator. The small size CDS is obtained by minimizing the number of chosen dominators. By utilising the two-hop neighborhood information, the enhanced greedy MPR algorithm in [13] is used to achieve this. Two-hop neighbor information as supposed to one-hop neighbor information allows the dominators to eliminate the redundant coverage of its one-hop neighbors resulting in minimum CDS

size. For example, the dominators can choose its one-hop neighbors that can cover the largest number of two-hop neighbors to become dominators. The advantage of SPSI is that it can build a CDS quickly using minimum energy resources and low communication overhead.

III. Assumptions and Implementation

In this section, the description on the assumptions and the detailed implementation of the study are presented.

A. Network Model

An *undirected graph* $G = (V, E)$ is used to represent the WSN, where V is a set of sensor nodes in the network, called vertices and E is a set of a communication link between a pair of sensor nodes, called edges usually denoted as $(u, v) \in E$. Two vertices u and v are neighbors if (1) they are within their maximum transmission range R_{max} and (2) the communication links between them are symmetrical. We assume all sensor nodes have same hardware capabilities. The sink node on the other hand has more powerful communication and processing features.

B. Average Path Length

Average path length is one of important measures of clustering given that the network topology generated is often used for routing. A path P in a graph G is a sequence of vertices connected by edges. To find the average shortest path length in the graph $G = (V, E)$, the greedy method based on classical Dijktra's algorithm is applied on the network. The main objective is to find the shortest path from each node i in the graph to a sink node j in which the edge weight is the distance between the vertices. It is assumed that the graph G is connected if the shortest path length between these two nodes is a finite number and it is disconnected if there is no connecting edge between the nodes.

In this paper, Dijkstra's technique was adopted to obtain the shortest path length measure. A total of 150 topologies were used to obtain the shortest path length measure. These topologies were referred to the original topologies. The clustering algorithms also utilize these topologies to form the backbone or clusters. Recall that the backbone topology is derived from the original topology in which the number of nodes remain the same except that redundant links in the backbone topology might be removed. Thus, it is possible for the backbone topology to have longer path between nodes. In order to investigate the performance of the average path length in the original topologies against the topologies of the backbone, we compute the average shortest length on both (i) original topologies and (ii) topologies constructed by the clustering algorithms (known as backbones).

IV. Performance of Algorithms

SPSI algorithm has the following performance:

Theorem 1. *SPSI has $O(n)$ message complexity and $O(3\Delta C + 3\Delta)$ time complexity, where n is defined as the overall number of nodes, Δ is the maximum node degree while C is the number of selected connectors.*

Proof. SPSI has $O(n)$ message complexity because each node u exchanges exactly one message when building the CDS.

The time complexity of the SPSI algorithm refers to the amount of time required for computing the connector set. SPSI takes the advantage of MPR property [13] when determining the connector, where the time complexity of the SPSI algorithm is $O(3\Delta C + 3\Delta)$.

ECDS algorithm has the following performance:

Theorem 2. *ECDS has $O(n)$ message complexity and $O(n)$ time complexity, where n is defined as the total number of nodes.*

Proof. Each node sends at most one message during the first and second phase of the algorithm. Hence, its message complexity is is $O(n)$.

The time complexity of ECDS is bounded by the MIS construction in which node requires at most $O(n)$ time complexity.

PACDS algorithm has the following performance:

Theorem 3. *PACDS has $O(m)$ message complexity and $O(\Delta^3)$ time complexity, where m is the number of edges and Δ is the maximum nodal degree.*

Proof. The message complexity of PACDS is contributed by the number of messages sent to each edge, in this case are two messages. Whereas the time complexity refers to the $O(\Delta^3)$ time required for constructing and pruning the CDS.

V. Simulations and Discussions

In this section, we present our simulation results. We evaluated the presence of the average path length with respect to network density (or average node degree) and network sizes to investigate the scalability of the algorithm. We also compare the performance of all algorithms to study how well they preserve the average path length.

The discrete event simulator OMNeT++ (version 4.1) [4] was used for simulating various algorithms. In order to have more realistic model, MiXiM [14] which is an extension of OMNeT++ framework was used to provide models of radio wave propagation and MAC protocols. Realistic topologies that were extended from a small network calibrated using the testbed in our department were created. We generated different types of topologies to represent various network sizes and densities. These topologies are categorized into three densities; sparse network (node degree 4), medium network (node degree 8) and dense network (node degree 12). To obtain various network sizes we varied the number of nodes N in each density from 100 to 500 with an interval of 100. To achieve a 95% confidence interval, the simulations were repeated with 10 runs. We assumed that the nodes were deployed in a 2D dimensional space and their transmission range varied.

A. The Effect of Network Densities on the Average Path Length

We first studied the average path length over various network densities ranging from sparse, medium to dense. It is evident from Figures 2(a), 2(b) and 2(c), the average path length decreases as the topologies become denser. As the network becomes denser, the number of available paths from a source node to a destination node also increases. This allows nodes to route through a shorter path, resulting in a shorter path length value. Contradictory to dense networks, the average path length of sparse networks yields the largest value due to limited route choices.

Among the three algorithms, the ECDS provides a significant improvement in the path length as the topologies are varied from sparse to dense, in which the path length of the dense topology is 50% lower than the path length of the sparse topology. Whereas the path length of the PACDS and SPSI drops to 44% as the topologies were varied from sparse to dense.

B. Performance Comparison of Algorithms

We also compared the average path length gained for backbone and original topologies against the network sizes. In general, the average path length increases as the network size increases from 100 to 500 as shown in Figures 3(a), 3(b) and 3(c). This can be explained by the fact that the number of paths to traverse from a source to a destination node increases.

It is clear that PACDS yields the lowest average path length followed by SPSI and ECDS. For sparse network, the path length of the PACDS is 13% lower than the path length of the ECDS, followed by 38% and 20% lower than the path length of the ECDS in medium and dense networks respectively. PACDS creates a larger backbone size than the SPSI and ECDS due to its less efficient rule in constructing the backbone. Larger backbone means that there are more available paths to traverse resulting in better path length. ECDS on the other hand has smaller backbone size thus limiting the choice of routing through shorter links.

The PACDS and SPSI yield about the same path length values in all three network densities.The figures indicate the average path length of the original topologies which were used to build the backbone. It is obvious that ECDS, SPSI and PACDS are able to preserve the path length property since their path length values are in close agreement with the path length of the original topologies.

VI. Conclusion

In this paper, we have compared the ability of clustering algorithms in preserving the path length property. We showed that the average path length increased with the network sizes but decreased with the network densities. The simulation results confirmed that the path length calculated over the original topologies is in close agreement with the path length computed on the backbone topologies. In the future, we would like to investigate the performance of routing algorithms on the backbone topologies. It would be interesting to study the effect of mobility on these clustered networks.

References

[1] J. Wu, M. Gao, and I. Stojmenovic, "On calculating power-aware connected dominating sets for efficient routing in ad hoc wireless networks," in *International Conference on Parallel Processing*, Sep. 2001, pp. 346 – 354.

[2] Z. Yuanyuan, X. Jia, and H. Yanxiang, "Energy efficient distributed connected dominating sets construction in wireless sensor networks," in *Proceedings of the 2006 International Conference on Wireless Communications and Mobile Computing*. ACM, 2006, p. 802.

[3] A. A. Aziz and Y. A. Sekercioglu, "A distributed energy aware connected dominating set technique for wireless sensor networks," in *Proceedings of the 4th International Conference on Intelligent and Advanced Systems (ICIAS 2012)*, vol. 1. IEEE, 2012, pp. 241–246.

[4] "OMNeT++ discrete event simulator," http://www.omnetpp.org/.

[5] A. A. Aziz, Y. A. Sekercioglu, P. Fitzpatrick, and M. Ivanovich, "A survey on distributed topology control techniques for extending the lifetime of battery powered wireless sensor networks," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 1, pp. 121–144, 2013.

[6] 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*, vol. 8. Citeseer, 2000, p. 8020.

[7] O. Younis and S. Fahmy, "HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks," *IEEE Transactions on Mobile Computing*, vol. 3, no. 4, pp. 366–379, 2004.

[8] Y. Liao, H. Qi, and W. Li, "Load-balanced clustering algorithm with distributed self-organization for wireless sensor networks," *IEEE Sensors Journal*, vol. 13, no. 5, pp. 1498–1506, 2013.

[9] S. Lindsey and C. S. Raghavendra, "PEGASIS: Power-efficient gathering in sensor information systems," in *Aerospace conference proceedings, 2002*, vol. 3. IEEE, 2002, pp. 3–1125.

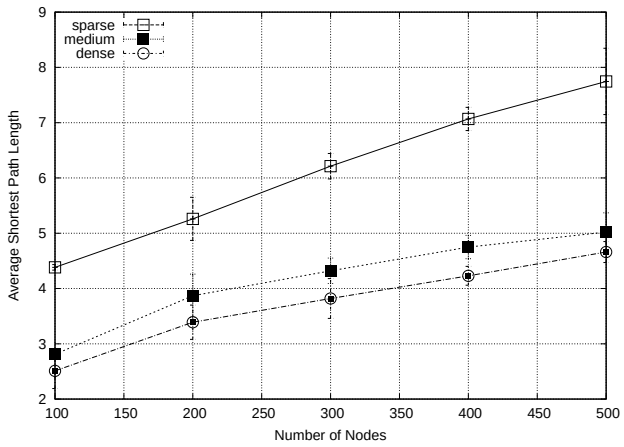
[10] J. Wu and H. Li, "On calculating connected dominating set for efficient routing in ad hoc wireless networks," in *Proceedings of the 3rd International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications*. ACM, 1999, p. 14.

[11] S. Guha and S. Khuller, "Approximation algorithms for connected dominating sets," *Algorithmica*, vol. 20, no. 4, pp. 374–387, 1998.

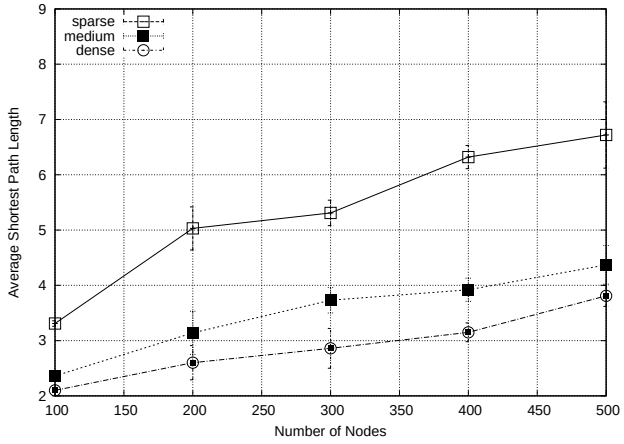
[12] A. A. Aziz and Y. A. Sekercioglu, "A distributed energy aware connected dominating set technique for wireless sensor networks," in *Proceedings of the 2012 International Conference on Intelligent and Advanced Systems (ICIAS)*. IEEE, 2012, pp. 241–246.

[13] A. Qayyum, L. Viennot, and A. Laouiti, "Multipoint relaying for flooding broadcast messages in mobile wireless networks," in *Proceedings of the 35th Annual Hawaii International Conference on System Sciences (HICSS)*. IEEE, 2002, pp. 3866–3875.

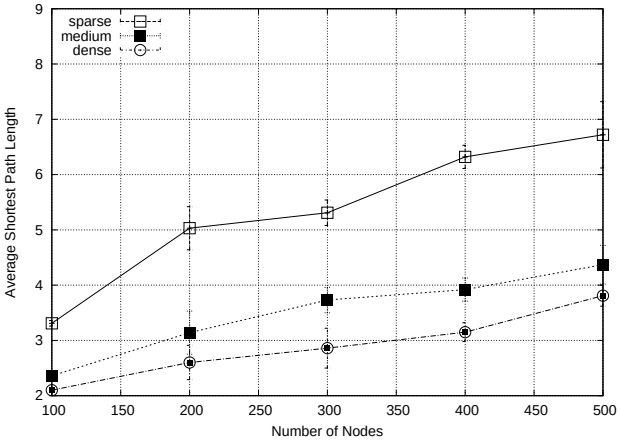
[14] "MiXiM simulator for wireless and mobile simulations," http://mixim.sourceforge.net/.



(a) Comparison of average path length against network densities for ECDS.

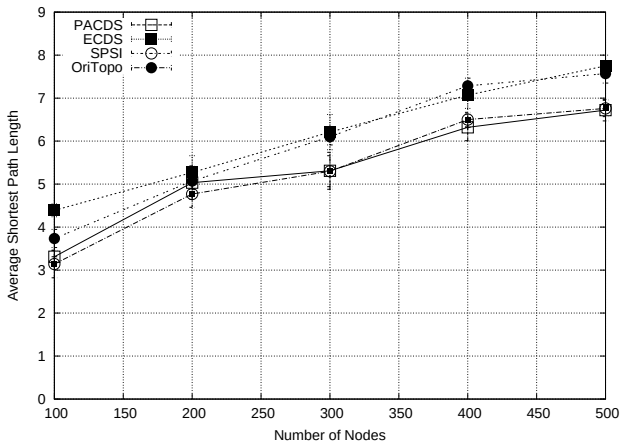


(b) Comparison of average path length against network densities for PACDS.

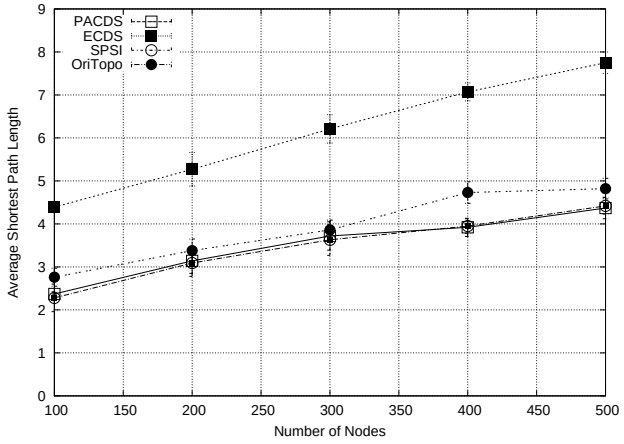


(c) Comparison of average path length against network densities for SPSI.

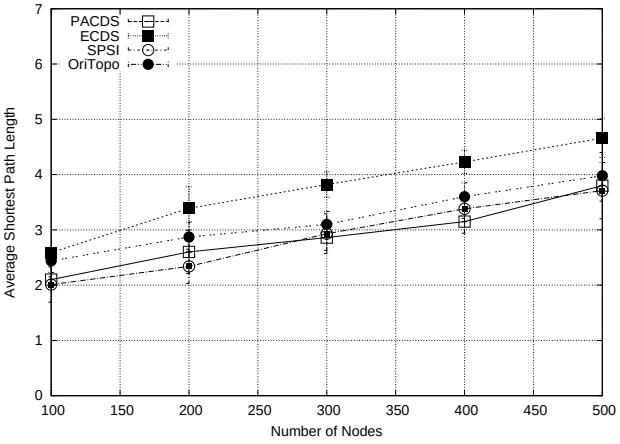
Fig. 2. Average path length versus various network densities.



(a) Average path length versus network size for sparse network.



(b) Average path length versus network size for medium network.



(c) Average path length versus network size for dense network.

Fig. 3. Average path length comparison against network size.