Mavromatis, A., Papadopoulos, G., Fafoutis, X., Goulianos, A., Oikonomou, G., Chatzimisios, P., & Tryfonas, T. (2017). Link quality and path based clustering in IEEE 802.15.4-2015 TSCH networks. In *2017 IEEE Symposium on Computers and Communications (ISCC)* Institute of Electrical and Electronics Engineers (IEEE). https://doi.org/10.1109/ISCC.2017.8024625

Peer reviewed version

Link to published version (if available):
10.1109/ISCC.2017.8024625

Link Quality and Path Based Clustering in IEEE 802.15.4-2015 TSCH Networks

Alexandros Mavromatis*, Georgios Z. Papadopoulos†, Xenofon Fafoutis*,
Angelos Goulianos*, George Oikonomou*, Periklis Chatzimisios‡ and Theo Tryfonas*

*Faculty of Engineering, University of Bristol, UK
{a.mavromatis, xenofon.fafoutis, agoulianos, g.oikonomou, theo.tryfonas}@bristol.ac.uk
†IRISA, IMT Atlantique, France *georgios.papadopoulos*@telecom-bretagne.eu
‡CSSN Research Lab, Department of Informatics, Alexander TEI of Thessaloniki, Greece *pchatzimisios*@ieee.org

***Abstract*—Advance clustering techniques have been widely used in Wireless Sensor Networks (WSNs) since they can potentially reduce latency, improve scheduling, decrease end-to-end delay and optimise energy consumption within a dense network topology. In this paper, we present a novel clustering algorithm for high density IEEE 802.15.4-2015 Time-Slotted Channel Hopping (TSCH). In particular, the proposed methodology merges a variety of solutions into an integrated clustering design. Assuming an homogeneous network distribution, the proposed configuration deploys a hierarchical down-top approach of equally numbered sub-groups, in which the formation of the separate sub-groups is adapted to the network density and the node selection metric is based on the link quality indicator. The presented algorithm is implemented in Contiki Operating System (OS) and several test vectors have been designed in order to evaluate the performance of the proposed algorithm in a COOJA simulation environment. Performance results demonstrate the capability of the clustering structure since compared to the default scheme it significantly improves the energy efficiency up to 35%, packet drops more than 40% as well the packet retransmission rate. Last but not least, the outcome of this study indicates a major increase in the network lifetime, *i.e.*, up to 50%.**

***Index Terms*—Internet of Things, IEEE 802.15.4-2015, TSCH, Synchronisation, Clustering, Energy Consumption.**

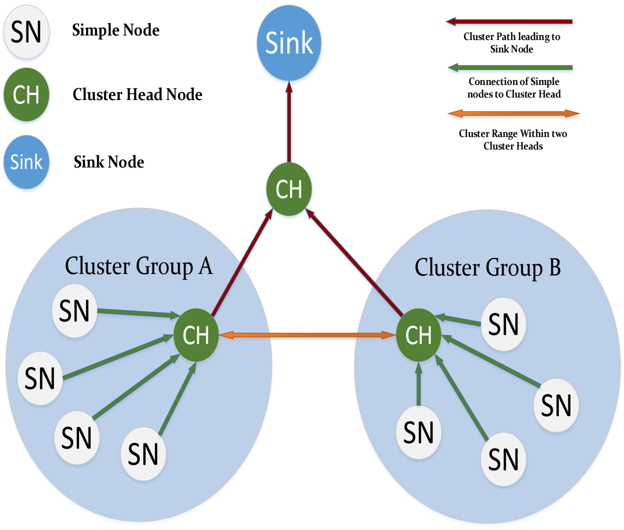


Figure 1. Clustering Algorithm visualisation. Cluster groups A and B contain Simple Nodes (SN) and each of them a Cluster Head (CH). Every cluster group has a Cluster Range and the Cluster Path to the sink node.

I. INTRODUCTION

Due to their enormous capability, Wireless Sensor Networks (WSNs) have attracted significant attention from the wireless communication community during the last decade. WSNs applications are used in various scientific fields ranging from medical to environmental monitoring and rural development. The reduction of the power consumption in WSNs is one of the most important challenges and, thus, ongoing research targets on the optimisation of the parameters that could contribute in the formation of energy efficient network topologies [1], [2]. Furthermore, WSNs aim at increasing network reliability, throughput as well as the lifetime of the active nodes [3], [4].

Clustering is a popular strategy of implementing a parallel processing technique into a group of computing devices. The idea is to define rules and behaviours between a class of resources that have common objectives. Algorithms based on clustering are being widely used in the networking field since they are able to deliver high efficiency, in matter to implement parallel processing, load balancing and fault tolerance, also for better organising dense networks. By orchestrating a network architecture, improving energy efficiency and connectivity are targets that have been pursued in WSNs.

In literature, there are different approaches that could potentially be employed to reduce energy consumption. Among them, efficient clustering techniques have been shown to be a promising solution. Under these grounds, this paper describes in detail and evaluates a novel clustering algorithm that is based on the Time-Slotted Channel Hopping (TSCH) [5]. TSCH is a Medium Access Control (MAC) protocol for ultra low-power operation in Low-Power Lossy Networks (LLNs) [6]. The proposed clustering algorithm identifies that the 6TiSCH minimal configuration is unable to deliver high reliability in high dense networks and consumes energy due to the many retransmissions, therefore we aim to design a clustering model that enables the network to efficiently manage traffic. We implemented the clustering technique in Contiki OS and evaluated it by employing COOJA simulator.

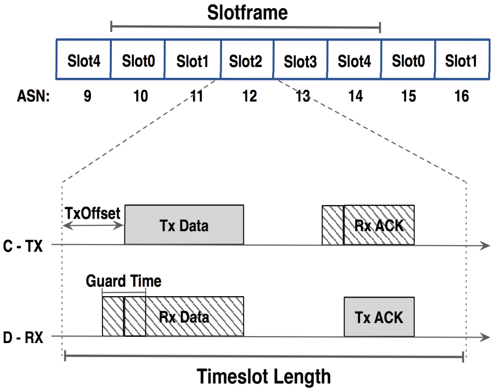


Figure 2. A typical TSCH timeslot template for a transmitter (top) and receiver node (bottom): node C, transmits its data packet after `TxOffset`, while the receiver D.

In a nutshell, the concept of this study is the division of the whole network into clustering groups by evaluating the link quality among the nodes and, thus optimizing the path to the sink node.

Figure 1 illustrates a high level architecture of the proposed algorithm. The implementation and testing is based on IEEE 802.15.4-2015 TSCH network that facilitates the clustering process once the nodes are synchronized, while a time slotted access scheme is employed as depicted in Figure 2. The current work presents simulation results of four network density configurations, each comprised of 40, 60, 80 and 100 nodes. The proposed algorithm aims to subdivide the network into clusters, where each cluster is coordinated by the cluster head. As it is shown in Section IV, by employing the proposed technique, a significant improvement on the number of packet retransmissions is attained, resulting in an enhancing connectivity among the nodes and reducing energy consumption. The clustering algorithm proposed in the paper is build on calculating the link quality among nodes and adjusting the latter on the network density (nodes per network dimensions). Our method utilizes the following benchmarks:

- Link Quality Indicator (LQI)
- Network Density (ND).

The remainder of this paper is organized as follows: Section II presents some general clustering methodologies as well as concepts and it demonstrates related work on WSN clustering algorithms. Section III describes in detail a step by step implementation featuring the Contiki OS. Section IV presents the simulation outcome through the algorithmic implementation in the COOJA environment, whereas the major improvement achieved is highlighted for various traffic network densities. Finally, Section V summarizes the work presented herein, and suggests directions for future work.

II. RELATED WORK

As previously presented, various clustering algorithms have been proposed in the past [7]. Hereafter, we present key works from the literature related with clustering in WSNs.

Two well-known clustering algorithms are the Linked Cluster Algorithm (LCA) [8] and the LCA2 [9]. Under LCA the cluster head node is selected by calculating the higher identity in one hop and setting that node as the cluster head. To improve this algorithm the LCA2 algorithm was proposed that creates smaller groups of cluster teams and selects as cluster head the node with the lowest identity. The cluster head proposed in LCA2 is among all the network nodes and not among the nodes inside one hop.

Another nature of clustering algorithm is the Weighted Clustering Algorithm (WCA) [10]. The authors propose an algorithm that selects the cluster head based on the number of neighbors, transmission power and battery life. Moreover, the algorithm restricts the number of nodes in a cluster in order to keep the MAC protocol as much as reliable possible.

Number of clustering algorithms use a distributed logic where weights are used to select cluster heads. In [11] the authors present the Distributed Clustering Algorithm for Ad-Hoc networks (DCA). This algorithm defines weights based on the applications itself and elects the cluster head node with the highest weight inside a one hop. Distributed algorithms about clustering have been also proposed for mobile networks. In [12] the Distributed and Mobility Adaptive Clustering algorithm is proposed (DMAC) that is the modification of the DCA algorithm for mobile WSNs.

All previously presented algorithms aim to the clustering of small networks and specifically one hop networks. We further present related work on clustering algorithms that aim to improve the high traffic WSN. The proposed algoritm in [13] considers the cluster head as known and requires that all the nodes know the topology of the network. Finally, the algorithm aims to find the optimal cluster size and the cluster head assignment aiming at extending network lifetime.

In [14] the algorithm operates with an hierarchical logic where the nodes are distributed based to Poisson point processes and divided to levels in which every level is connected to another one. By this distributions, they calculated the length of segments connected between the hierarchical levels. In [15], this work has been extended, and enables more than two levels in the hierarchical communication. They authors consider a poisson-based distributions and stochastic geometry to calculate the best possible connection between the node levels.

In this paper, we propose a software defined clustering algorithm that profess that computing is more efficient than communication. We implement and test our algorithm on high traffic under a TSCH network. We aim on reducing energy consumption, improving reliability and reducing packet drops, as well as extending network lifetime.

The proposed clustering algorithm described in detail in the following Section, aims at improving the performance of a contention-based 6TiSCH minimal configuration network. Our simulation study demonstrated that although 6TiSCH Minimal Schedule high traffic, dense topologies, delivering high reliability and network connectivity is compromised, therefore the proposed model is improving these issues.

III. Proposed Clustering Algorithm

In this paper, we propose a deterministic clustering algorithm based on the link quality among the nodes under TSCH and 6TiSCH minimal configuration. Once the topology is initialized by Routing Protocol for Low power and Lossy Networks (RPL) [16] the algorithm runs procedures that are constantly repeated over the network lifetime. The link quality measurements and the creation of Cluster Groups within a common range is the first part of the operation. The second part is the management of the cluster groups and the selection of a Cluster Head. Running the algorithm will determine a Cluster Path through which data packets will be delivered to the sink node. An example of the proposed algorithm can be seen in Figure 1 in which the cluster group attributes and the connection between the Cluster Heads are demonstrated.

A. *Link Quality Approach to a Clustering Algorithm*

High density networks are prone to increase packet drops which in turn has a negative impact on network reliability. Therefore, 6TiSCH Minimal Configuration does not perform as expected when the network density exceeds a certain threshold. The network nodes simultaneously transmit both application data and control packets in broadcast mode, thus increasing the interference among the nodes. To overcome this limitation, the algorithm is adjusted accordingly and runs an application specific management model, which improves network performance under high traffic conditions. The Clustering Algorithm divides the network into clusters and coordinates the communication within such a cluster group. Moreover the algorithm reduces the number of broadcast control packets that are now defined by the ”keep alive” packets.

The first step of the algorithm is to adjust number of the Cluster Groups according to the network density. This is quite important, as the number the cluster groups depends on the total number of nodes within a specific area. The software calculates the network density which is defined as follows:

$$ND = TNN/a^2, \tag{1}$$

where α is the side of the square area where the nodes are positioned, TNN denotes the Total Number of Nodes and ND determines the number of clusters that are being formed after the algorithm is executed.

Furthermore, we defined another metric; RD is expressed as follows:

$$RD = (NTR \times TNN)/a^2, \tag{2}$$

where NTR is the Nodes Transmission Range and in our simulations was set to 35 meters. RD actually sets the number of nodes within a cluster group.

Based on the above models, and given the network density, the algorithm implements an empirical ”look-up” table that determines both the number of cluster groups and the number of nodes to be formed. Figure 3 illustrates an exemplary clustering formation where 40 nodes are split into seven groups and each group has a variable number of nodes.

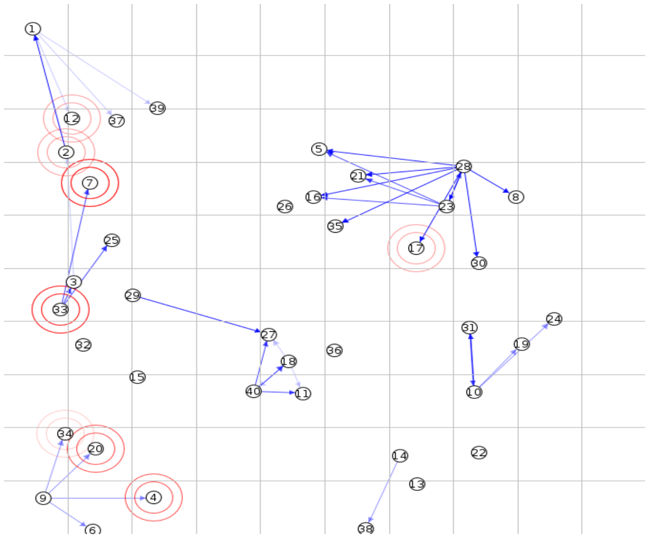


Figure 3. An example of the proposed clustering algorithm visualised in COOJA Simulator 40 nodes including the sink node.

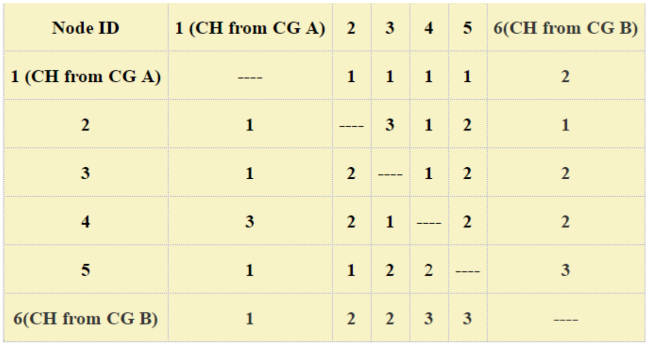
Once cluster groups are defined, the link quality information is being stored into a two-dimensional array. After the data quality information is stored the clustering process initiates and stores the node details under an unique id for each of the cluster groups. Such an example is depicted in Figure 4.

The link quality is evaluated with the discrete values 1, 2 and 3 that essentially categorise the LQI as provided by an inherent function implemented in Contiki OS. Based on this ranking process the cluster head is selected as the ”Best” node within a cluster group. As an example of this scenario is illustrated in Figure 4. The cluster head of group A is node 1, since it has the highest ranking among the rest of the nodes.

B. *Algorithmic features and improvements*

An important feature implemented in the proposed algorithm is the elimination of any communication between the nodes belonging to different cluster groups. This is a significant challenge since it reduces the number of transmissions and consequently interference within the network. In order to achieve this functionality, the software implements a function that prevents a node from choosing a neighboring time source node if it does not belong to the same cluster group. In the case of a cluster group failure, there is a timer running every 10 minutes that regulates the connectivity and ”keep alive” packets of every node.

A significant improvement of the proposed algorithm is the increase of network reliability and the efficient regulation of network traffic. This is due to the fact that in case the cluster head is within a one hop link distance from the sink node, the data will be delivered directly from the cluster head to the sink node. In the case of multi-hop topologies, packets will be forwarded from one hop to another by forming the optimum (link quality based) Cluster Path (CP). Since



Node ID	1 (CH from CG A)	2	3	4	5	6(CH from CG B)
1 (CH from CG A)	----	1	1	1	1	2
2	1	----	3	1	2	1
3	1	2	----	1	2	2
4	3	2	1	----	2	2
5	1	1	2	2	----	3
6(CH from CG B)	1	2	2	3	3	----

Figure 4. The visualization of the two dimensional array that is being used for the link quality based clustering. The ranking is calculated from both side of the nodes and is grades in 1, 2 and 3 one being the best.

every cluster head is the node that collects all the data, the algorithm suggests that a path comprised of Cluster Heads forwarding packets to the sink node can significantly improve connectivity and reduce the latency of the network. This has been validated through simulations that are presented in Section IV. Even though the cluster path method reduces latency and increases packet delivery rate, simulation results indicate a performance degradation in cases where the packet transmission rate increases. In such conditions the dynamic structure of the proposed algorithm allows the formations of additional cluster head nodes within the same cluster group.

C. Software Implementation and Validation

As mentioned in Section I (Introduction), the proposed clustering algorithm is implemented in Contiki OS and tested by employing the COOJA simulator. In this sub-Section the methodology and functions behind the presented algorithm will be explained in detail. The sequential execution of the proposed design includes the following algorithmic steps (functions):

1) `Controller():`
2) `RangeControl():`
3) `Clustering():`
4) `ClusterHeadUpdate():`
5) `ClusterPath():`

The Controller function is employed once the topology is initialized and its main functionality is to orchestrate the rest of the other functions and keeping information regarding the timetables. Furthermore, the controller initiates timers that are used in turn for the purpose of calling the ClusterHeadUpdate and ClusterPath fuctions. The RangeControl functions adjusts the range of each node within the area defined by the cluster group which contains the node. Running this function facilitates interference suppression with other cluster groups and collision avoidance. The main function of the algorithm is the clustering function which calculates and stores data that are being used for the formation of the individual clustering groups. In addition, the Clustering functions selects the Cluster Head among the nodes as well as determines the optimum Cluster Path. Occasionally, under certain conditions, the cluster path to the sink and the cluster head have to be modified, in such cases the ClusterPath and the ClusterHeadUpdate functions customize the network accordingly.

Table I
SIMULATION SETUP.

Topology	**Value**
Topology	Random Multi-hop
Number of nodes	40, 60, 80, 100
Simulation	**Value**
Duration	10 hours
Traffic Pattern	1 pkt / 60 sec , 10 pkt/ 60sec , 20 pkt/ 60sec
Data packet size	102 $bytes$ (77 $bytes\ payload$)
Routing model	RPL [16]
MAC model	TSCH (6TiSCH minimal schedule)
TSCH	**Value**
EB frequency	3.42 s
Slotframe length	7
Timeslot length	15 ms
Guard Time	(1800) μs
Transmition Range	35meters(dynamic)
Hardware	**Value**
Antenna model	CC2420
Radio propagation	2.4 GHz
Transmission power	0 dBm

The proposed algorithm has been evaluated under various testing scenarios. More specifically, the performance of the algorithm has been assessed for a range of network densities and packet transmission frequencies. The following Section demonstrates the capabilities of the proposed algorithm as its clearly illustrates that the performance of the presented clustering scheme outperforms the 6TiSCH minimal schedule in terms of energy efficiency, end-to-end delay and the network lifetime.

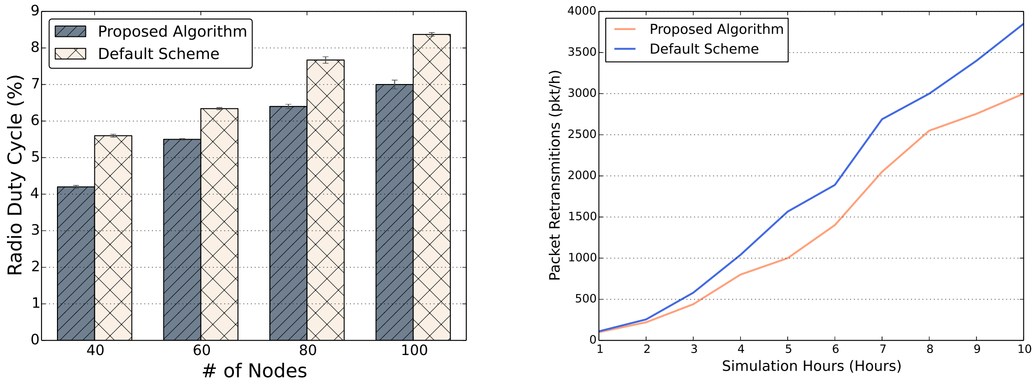
IV. PERFORMANCE EVALUATION

A. Simulation Setup

In order to test the performance of the algorithm the software was implemented in Contiki OS, and simulations took place in the COOJA simulator environment. Various scenarios have been evaluated; more specific simulations have been run by using Z1 nodes having hardware characteristics specified in Table 2. The figures of merit that have been employed to test the validity of the algorithm are the Radio Duty Cycle (RDC) spent during the network lifetime, and the traffic which was measured based on the total number of retransmissions within the simulation run-time. The reliability of the network was evaluated with repsect to the number of the packet drops and the presence of ”dead” nodes within the network. The parameters that have been used during simulations are depicted in Table I.

B. Simulation Results

Figure 5a illustrates the RDC percentage per node for each of the deployed simulation scenarios. It is shown that the proposed clustering algorithm reduces the RDC more than 20% respectively of the active network nodes. An important feature exhibited by the presented design is that the algorithm can dynamically be adjusted in order to incorporate additional



(a) The Radio Duty Cycle comparison between the default TSCH scheme and the Proposed Algorithm

(b) A visualization of the network traffic in terms of packet retransmissions, also another way to express the energy efficiency by less radio on time.

Figure 5. Radio Duty Cycle Results in Network Densities of 40, 60, 80 and 100 nodes, the same results refer to the 5b figure where we show the performance of the algorithm by reducing the total packet retransmissions.

cluster groups in case that the network requirements impose saw. Figure 5b demonstrates energy efficiency performance. As opposed to the 6TiSCH minimal schedule the proposed model provides high reliability and better network connectivity, improving thus this problematic behaviour of the default scheme.

Furthermore, the algorithm keeps retransmissions at lower levels and performs better than the default scheme, especially when the simulations run-time increases. Such an exemplary scenario is illustrated in Figure 5b for simulation run-time equal to 10 hours. As shown, the proposed algorithm presents a 25% reduction in packet retransmissions.

As mentioned earlier an important performance metric of the network reliability is the number of packet drops during the simulation. The packet drops are usually caused due to the Time to Live (TTL) limitations as well as due to the link congestion that could appear at a specific receiver. Another reason for the packet loss is the network inability to deliver throughout the full multi-hop approach. This is due to the fact that in TSCH networks, the nodes will store the packet that has to be forwarded, and may drop the packet if no ”next hop” is available. Another reason could be due to the node management as far as concern the transmission and reception time intervals. Finally, the broadcast control packets can cause significant congestion especially in highly dense populated networks.

The proposed algorithm significantly improves the default scheme of 6TiSCH minimal schedule, since it reduces the packet drops and performs better, even for high network densities. As shown in Figure 6a, simulation results indicate that the packet loss is reduced for the case of 40, 60, 80 and 100 nodes. As explained in Section III, the algorithm adjusts the clustering procedure according to the network density. Such an example is depicted in Figure 6a where it is shown that implementing this technique allows for a significant reduction in a number of ”dead” nodes, especially at high network densities.

Table II

APPROXIMATE CURRENT CONSUMPTION OF Z1 MOTE.

IC	Notes	Current Consumption
CC2420	TX Mode @ 0 dBm	17.4 mA
	RX Mode	18.8 mA
MSP430f2617	Active Mode @ 8 MHz	4 mA
	Low-power Mode	0.5 uA

Table III

SUMMARY OF RESULTS.

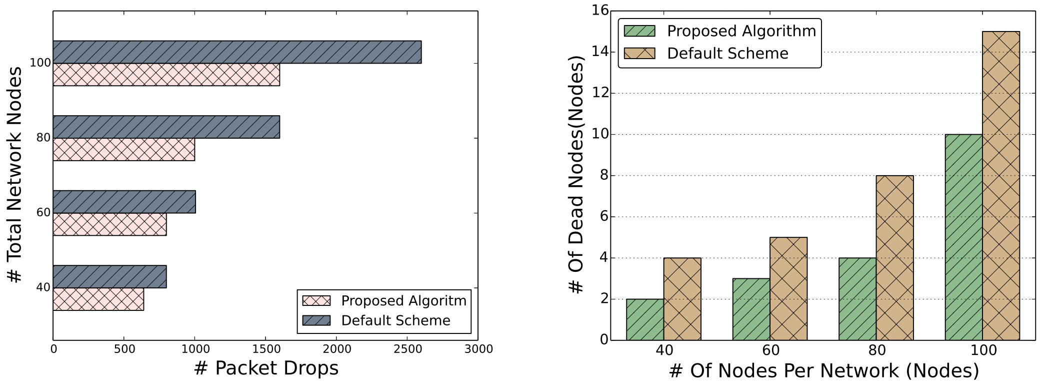
Scheme	Energy	Reliability	Traffic
TSCH with Proposed Algorithm	<7%	>70%	<65%
6TiSCH minimal schedule	>8%	>55%	>80%

In order measure the lifetime of the network we investigate the number of ”dead” nodes during the simulation run-time. ”Dead” nodes are defined as the nodes that experience more than 90% packet loss. Figure 6b highlights the decrease in the number of the ”dead” nodes after implementing the proposed algorithm. Figure 6b illustrates that in contrary to a 6TiSCH minimal schedule, the proposed clustering algorithm enables more nodes to be active in all four scenarios presented here in. Therefore the number of ”alive” nodes increases and this results in a more reliable and energy efficient network.

Table 3 summarises the performance of the presented algorithm and compares it to the 6TiSCH minimal schedule, in terms of energy efficiency, network reliability and traffic. As demonstrated, the proposed algorithm outperforms the default scheme, thus, improving the overall network functionality.

V. CONCLUSIONS

This paper demonstrates a novel clustering methodology that could potentially be applied in highly dense WSNs. The proposed algorithm is based on the link quality among nodes as well as the network density. More specifically, the suggested software based technique is implemented in Contiki OS and has been evaluated in COOJA environment. Simulation results



(a) Packet drops tested in various network densities.

(b) Inactive nodes during the lifetime of the network.

Figure 6. Packet drops per network and total number of inactive dead nodes.

designates the efficiency of the proposed scheme in terms of network reliability, traffic reduction and energy consumption of the network. Finally, simulation results depict an increase of the network lifetime since, as the proposed methodology allows for more active nodes within a defined network architecture.

Future work will investigate a possible extension of the algorithm to mobile WSNs. Furthermore, ongoing work involves the design of real hardware testbeds that will assess the proposed software techniques presented in the current paper. In the authors view, an interesting research topic would be the development of efficient clustering algorithms for none-homogeneous node distributions in future WSNs.

ACKNOWLEDGEMENTS

This work was partially performed under the SPHERE IRC funded by the UK Engineering and Physical Sciences Research Council (EPSRC), Grant EP/K031910/1.

REFERENCES

[1] G. Z. Papadopoulos, A. Mavromatis, X. Fafoutis, N. Montavont, R. Piechocki, T. Tryfonas, and G. Oikonomou, “Guard Time Optimisation and Adaptation for Energy Efficient Multi-hop TSCH Networks,” in *Proceedings of the IEEE 3rd World Forum on Internet of Things (WF-IoT)*, 2016, pp. 301–306.

[2] G. Z. Papadopoulos, A. Mavromatis, X. Fafoutis, R. Piechocki, T. Tryfonas, and G. Oikonomou, “Guard Time Optimisation for Energy Efficiency in IEEE 802.15.4-2015 TSCH Links,” in *Proceedings of the EAI International Conference on Interoperability in IoT (InterIoT)*, 2016.

[3] G. Z. Papadopoulos, T. Matsui, P. Thubert, G. Texier, T. Watteyne, and N. Montavont, “Leapfrog Collaboration: Toward Determinism and Predictability in Industrial-IoT applications,” in *Proceedings of the IEEE International Conference on Communications (ICC)*, 2017.

[4] G. Z. Papadopoulos, J. Beaudaux, A. Gallais, P. Chatzimisios, and T. Noel, “Toward a Packet Duplication Control for Opportunistic Routing in WSNs,” in *Proceedings of the IEEE Global Communications Conference (GLOBECOM)*, 2014, pp. 94–99.

[5] P. Thubert, “An Architecture for IPv6 over the TSCH mode of IEEE 802.15.4,” draft-ietf-6tisch-architecture-10, June 2016.

[6] A. Mavromatis, G. Z. Papadopoulos, X. Fafoutis, A. Elsts, G. Oikonomou, and T. Tryfonas, “Impact of Guard Time Length on IEEE 802.15.4e TSCH Energy Consumption,” in *Proceedings of the IEEE International Conference on Sensing, Communication and Networking (SECON)*, 2016.

[7] A. A. Abbasi and M. Younis, “A survey on clustering algorithms for wireless sensor networks,” *Comput. Commun.*, vol. 30, no. 14-15, pp. 2826–2841, 2007.

[8] D. Baker and A. Ephremides, “The architectural organization of a mobile radio network via a distributed algorithm,” *IEEE Transactions on Communications*, vol. 29, no. 11, pp. 1694–1701, 1981.

[9] A. Ephremides, J. E. Wieselthier, and D. J. Baker, “A design concept for reliable mobile radio networks with frequency hopping signaling,” *Proceedings of the IEEE*, vol. 75, no. 1, pp. 56–73, 1987.

[10] M. Chatterjee, S. K. Das, and D. Turgut, “Wca: A weighted clustering algorithm for mobile ad hoc networks,” *Cluster Computing*, vol. 5, no. 2, pp. 193–204, 2002.

[11] S. Basagni, “Distributed clustering algorithm for adhoc networks,” *in Proceedings of International Symposium on Parallel Architectures, Algorithms and Networks*, pp. 310–315, 1999.

[12] S. Basagni, “Distributed and mobility-adaptive clustering for multimedia support in multi-hop wireless networks, in proceedings of vehicular technology conference.” 1999, pp. 889–893 vol.2.

[13] C.-F. Chiasserini, I. Chlamtac, P. Monti, and A. Nucci, “Energy efficient design of wireless ad hoc networks.” Springer-Verlag, 2002, pp. 376–386.

[14] S. G. Foss and S. A. Zuyev, “On a voronoi aggregative process related to a bivariate poisson process,” in *Adv. in Appl. Probab*, 1996, pp. 965–981.

[15] F. Baccelli and S. Zuyev, “Poisson-Voronoi Spanning Trees with Applications to the Optimization of Communication Networks,” INRIA, Tech. Rep. RR-3040, 1996.

[16] T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J. Vasseur, and A. R., “RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks,” RFC 6550, 2012.