Proceedings Paper:

Avdullahi, A., Lagkas, T. orcid.org/0000-0002-0749-9794 and Sarigiannidis, P.G. (2017) QoS Supportive MAC Protocols for WSNs: Review and Evaluation. In: Zdravkova, K., Eleftherakis, G. and Kefalas, P., (eds.) BCI '17 Proceedings of the 8th Balkan Conference in Informatics. The 8th Balkan Conference in Informatics, 20-23 Sep 2017, Skopje, Macedonia. ACM , 22:1-22:1. ISBN 978-1-4503-5285-7

https://doi.org/10.1145/3136273.3136282

QoS Supportive MAC Protocols for WSNs: Review and Evaluation

A. Avdullahi
Computer Science Department
The University of Sheffield
International Faculty, CITY College
Thessaloniki, Greece
aavdullahi@city.academic.gr

T. D. Lagkas
Computer Science Department
The University of Sheffield
International Faculty, CITY College
Thessaloniki, Greece
T.Lagkas@sheffield.ac.uk

P. Sarigiannidis
Department of Informatics and
Telecommunication Engineering
University of Western Macedonia
Kozani, Greece
psarigiannidis@uowm.gr

ABSTRACT

The use of wireless sensor networks technology is growing in different applications of monitoring. Since it is a relatively new technology, the interest of researchers to improve the network performance and behaviour has been enormous. In this context, new resource allocation scheme that takes into account traffic priority and load has been introduced. The evaluation of this scheme is intended to be achieved by implementing a custom simulator. This report discusses and evaluates all the important concerns needed to be considered during the development of this project. Moreover, this work also reviews the related literature in order to afford optimisations to the scheme.

KEYWORDS

Sensors, Wireless Sensor Networks (WSN), MAC, Resource Allocation.

1 INTRODUCTION

The energetic growth rate of networks has had a tremendous effect on the improvement of other technologies. One such technology that has been revolutionised by the introduction of wireless technology are sensors. Prior to wireless, sensors typically relied on a wired network through which they could communicate. However, in many industrial and commercial applications it has been difficult or expensive to monitor the environment using wired sensors [1]. The introduction of wireless sensor networks has prompted the usage of sensors in diverse areas such as military, health, agriculture, etc.

Wireless sensor networks are spatially distributed autonomous sensors to monitor physical or environmental conditions, such as temperature, sound, pressure, etc. and to cooperatively pass their data through the network to a main location [2]. In the past couple of years there have been numerous of diverse research projects regarding Wireless Sensor Networks (WSNs). Rawat et al. [3] argues that WSN is probably the most researched area in the last decade. The motives for research in this area are numerous, however the advantages that WSNs include, such as opportunity for active usage of resources in perilous environments, lower costs, scalability, reliability, accuracy, flexibility, as well as ease of deployment, are considered to be the main ambitions to research this area [4], [5]. Regardless of the vast research that has been conducted, there are still many hindrances in regards to wireless sensor communication that need to be challenged [6]. One particular area of interest is devising and analysing novel communication architectures, in order to provide an efficient and qualitative routing technique. On this ground, a new resource allocation scheme that focuses on queue load and access prioritization has been proposed.

In this report, the relevant literature is reviewed and thoroughly discussed in order to be able to compare the different approaches and primacies of several schemes. The literature reviewed will serve as a basis for introducing optimizations to the examined scheme. Moreover, relevant simulators/emulators have been reviewed as candidate tools to evaluate such a scheme. However, it was the review of the simulators/emulators that demonstrated the need to develop a specialized simulator. The simulator developed is able to create different scenarios in which modern resource allocation algorithms are tested. The simulator measures the key network metrics (i.e. throughput, delay, etc.) and presents them on a comprehensive and useful manner. The developed simulator is utilized in order to evaluate the examined schema. The results of the evaluation will be discussed later in this report.

The rest of this paper is organized as follows. Section II discusses similar relevant schemes, as well as WSN simulators/emulators. Section III discusses thoroughly the examined WSN resource allocation scheme. Section IV presents and discusses the performance results of the scheme. Finally, section V concludes this paper.

2 BACKGROUND AND RELATED WORK

Taking into consideration the diverse applications of Wireless sensor networks, there are many factors to consider upon devising a MAC (Medium Access Control) protocol. The application specific requirements create the need for prioritization of certain factors [7]. Nevertheless, the majority of the proposed protocols mainly focus on quality of service and energy efficiency. Considering this, some of the priority-based, energy efficient, as well as both priority-based and energy efficient protocols have been discussed below.

Moreover, similar emulator/simulators are reviewed and discussed in this paper as well. Taking into consideration the specific requirements of the scheme, it was very important to analyse the existing tools in order to conclude on their applicability.

2.1 WSN MAC Protocols

2.1.1 Priority-based WSN MAC Protocols

QoMOR

As it was mentioned before, many of the schemes discussed in this literature review are designed to meet QoS (Quality of Service) requirements set for WSN based on traffic priorities. QoMOR (QoS-aware MAC protocol using optimal retransmission) achieves this by implementing a scheme that categorizes the network nodes into classes in order to prioritize them, and based on the categorized classes it allocates resources [8]. The algorithm uses nonlinear programming to calculate the bandwidth allocated for each class. Once class values are calculated, the classes can use those values to transmit data. Moreover, in order to ensure that the data gets transmitted, rather than lost because of collisions, this algorithm also defines the number of times each class should retransmit the data. Finally, the scheme allows the deprioritizing of nodes by classifying all of them in one class. In these cases, the bandwidth allocation is calculated based on the number of nodes in the network.

PBSDS

PBSDS (Priority Based Service Differentiation Scheme) favours the higher priority nodes on resource allocation. In order to achieve differentiation, the paper proposes two mechanisms: 1) backoff exponent differentiation (BED) and contention window differentiation (CWD) [9]. The backoff exponent is the integer assigned to a node when the node wants to transmit packets but the network is busy. The backoff differentiation scheme proposed in this paper assigns random backoff exponents to nodes in order to avoid collision. However, nodes of higher priority are serviced primarily by assigning bigger backoff exponents. The content window, on the other hand, is the number of clear channel assessment (CCA) procedures required to confirm that the packet transmission can begin. The CWD scheme assigns lower CW values to higher priority nodes, thus ensuring that higher priority nodes are provided with service first.

CSMA/TDMA

In the track of finding new means of increasing the throughput in WSNs, there are many examples that use hybrid technologies to create innovative allocation schemes. A typical example is the adaptive CSMA/TDMA (Carrier Sense Multiple Access / Time Division Multiple Access) hybrid MAC proposed in [10]. According it, the collision avoidance mechanism provided by traditional CSMA is not efficient in the cases of large scale WSN. Moreover, in high loads, the CSMA protocol has a low throughput. Therefore, a hybrid protocol is proposed that moves towards TDMA because of the higher achieved throughput. Still the proposed solution also supports low rate and low power communication offered by CSMA. The proposed solution calculates the channel 7 utilization of the network and decides for the border between CSMA and TDMA. Once the algorithm decides on the boundary between CSMA and TDMA, all the nodes must inform the central coordinator about the current load state in their queue as an eight-level meter of the fraction of queue currently occupied. All the nodes with a queue state factor higher than 1 will be allocated slots through a TDMA greedy algorithm, while all the nodes with smaller queue state factor will have to compete for the CSMA allocated slots. As mentioned previously, the nodes with a higher queue state fraction than 1 will be assigned bandwidth through the TDMA reserved slots. Through the greedy algorithm, the nodes are provided with bandwidth to transmit data that considers both the queue state and the available TDMA bandwidth.

DMP

In [11], the authors propose DMP (Dynamic Multilevel Priority) to organize the nodes into a hierarchical structure where they can communicate with each other. The hierarchical structure proposed has two kind of nodes, leaf nodes and intermediate nodes, where the leaf nodes are the nodes that are at the last hierarchical level, while intermediate nodes are found in lower hierarchical level. In order to prioritize the traffic traveling through the network, the scheme classifies the data packets in three main types: real time data packets (priority 1), non-real time data packets that are received by the lower level nodes (priority 2) and non-real time data packets that are congregated in the node itself. Consequently, in order to be able to classify the packets appropriately, each node is assigned three queues to temporarily store the data. The data that is located in the first priority queue is transmitted in a non-preemptive first come first serve basis. Moreover, the transmission of real-time data packets cannot be interrupted in order to transmit data of other priorities, but it can preempt all other running tasks. On the other hand, the non-real-time data received by lower level nodes follows a preemptive TDMA manner of data transmission. The transmission of second priority data can be interrupted by the first priority traffic or it will be interrupted after a certain amount of time, in order to allow the transmission of third priority packets. Similarly to second priority packets, the third priority packets are also allocated a certain amount of time to transmit the data. However, the time allocated for the third priority data packets is slightly shorter than second priority data packets.

2.1.2 Energy Efficient WSN MAC Protocols

DCLA

Taking into consideration that the sensor nodes used in WSNs have a very limited battery, it has been a constant strive to create protocols that are energy efficient. In a similar manner, the authors of [12] try to tackle this issue by proposing a new Duty Cycle Learning Algorithm (DCLA) that tries to optimize the duty cycle of the beacon enabled WSNs. The duty cycle learning algorithm uses a formula to estimate the traffic. The proposed algorithm uses the Q-learning methodology to determine the optimal duty cycle, while through the traffic estimation the algorithm uses, it derives the length of the super frame that will be used to transmit traffic.

TDMA-ASAP

As mentioned earlier, TDMA is a very popular MAC protocol for WSNs. It is very popular because of its efficiency in dealing with bandwidth in highly loaded networks. However, the problem with this protocol is that it conserves too much energy in order to calculate and synchronize the bandwidth allocation between the nodes. The protocol TDMA-ASAP (TDMA – Adaptive Slot-Stealing and Parallelism) proposed in [13] presents modifications to the TDMA protocol in order to make it more energy efficient during low load transmissions. It does so by adding the following: transmission parallelism based on a level-by-level localized graph-colouring, appropriate sleeping between transmissions ("napping"), judicious and controlled TDMA slot stealing to avoid empty slots and intelligent scheduling/ordering transmissions. The main idea behind these modifications is to create "coloured zones", so that nodes in different zones can transmit in parallel, as long as the packets will be transmitted to a node that is in the same zone.

CA-MAC

The Context Adaptive MAC (CA-MAC) protocol tries to conserve energy by introducing a scheme that aims to allocate the bandwidth efficiently [14]. The specific protocol achieves that by letting each node stay in sleep mode if the number of packets in the buffer is smaller than the threshold. The threshold value is decided according to the distance of the node to the sink node. Nevertheless, a node can terminate sleeping mode if it receives a packet that has a high grade of context. The protocol may cause increased latency because of the threshold. In order to tackle this, the proposed protocol initially transmits the high priority packets.

PW-MAC

In the PW-MAC (Predictive-Wakeup MAC) protocol the nodes wake up at randomized, asynchronous times [15]. However, contrary to other proposals, in this protocol the sender node predicts the wake-up time of the receiver node. In order to accurately predict the wake-up time of nodes, this protocol proposes an on-demand prediction error correction mechanism that effectively addresses timing challenges, such as clock drift and operating system delays. Moreover, PW-MAC also introduces an efficient prediction based retransmission mechanism that achieves high energy efficiency even when packets must be retransmitted because of the collisions in the network.

2.1.3 Priority-based and Energy Efficient MAC Protocols

Z-MAC

Z-MAC (Zebra MAC) is a hybrid protocol that tries to combine the strengths of two protocols, TDMA and CSMA, while offsetting their weaknesses [16]. The CSMA protocol is a very flexible protocol, as it does not require clock synchronization nor calculations to distribute bandwidth. However, the problem with CSMA is that is has a high collision rate. TDMA, on the other hand, has a low collision rate, but it requires more power to calculate bandwidth allocation and clock synchronization. Z-MAC achieves to combine these two by proposing two modes of transmission, the Low Contention level (LCL) and the High Contention Level (HCL). In the LCL mode, all the nodes can acquire the frame slot and use it to transmit data. While in the HCL model only the node and its one-hop neighbouring nodes can transmit. A node moves into the high contention level by transmitting a message called Explicit Content Notification (ECN). Finally, in order to avoid exorbitant operations during transmission, the algorithm has a setup phase in which it runs the following operations: neighbour discovery, slot assignment, local frame exchange, and global time synchronization.

OGTS

Similar to other hybrid protocols discussed in this paper, the scheme proposed by the author in [17] also tries to optimize an already existing protocol. However, the OGTS (Optimization-Based GTS) scheme focuses mainly on allocating the Guaranteed Time Slots (GTS) in the Contention Access Period (CAP), while the contention free period is not a concern of this scheme, hence, it is handled by the IEEE 802.15.4 MAC protocol. Similar to other approaches surveyed by our work, this scheme also has a coordinator that accepts GTS requests and handles those requests by allocating time slots to transmit data. As mentioned earlier, the connected devices are able to transmit data through pre-allocated slots as well as through contesting for slots during the contention free period. Nevertheless, in order for the nodes to be able to transmit during CAP, a request must be sent to the coordinator. The request contains the number of packets that the node needs to transmit granted that the load exceeds a certain threshold set by the network. The coordinator, on the other hand, receives all the GTS requests and allocates time slots following the knapsack algorithm. The specific algorithm follows a round-robin methodology to allocate the slots. However, the knapsack algorithm is not strictly round-robin, as it favours the nodes that have higher number of packets in the queue.

A comparison of all aforementioned QoS supportive MAC protocols for WSNs is presented in Table 1, based on their abilities related with energy efficiency and data prioritization.

Table 1: QoS supporting MAC protocols comparison

Name	Energy Efficiency	Data Prioritization
QoMOR	Low	High
PBSDS	Low	High
CSMA/TDMA	Low	High
DMPPS	Medium	High
DCLA	High	Low
TDMA-ASAP	High	Low
CA-MAC	High	Medium
PW-MAC	High	Medium
Z-MAC	Medium	Medium
OGTS	Medium	Low

2.2 WSN Simulation/Emulation Approaches

2.2.1 WSN Simulation/Emulation Tools

NS2

NS2 is a network simulator built in Object-Oriented extension of Tool Command Language and C++ [18]. This simulator can be run on Linux Operating Systems or on Cygwin, which is a Unix-like environment and command-line interface running on Windows. It is a generic open source network simulator and provides sufficient online documentation.

TOSSIM

TOSSIM is an emulator specifically designed for WSNs running on TinyOS, which is an open source operating system targeting embedded systems. Users can execute TOSSIM on Linux Operating Systems or on Cygwin on Windows. It also provides sufficient online documentation [19].

EmStar

EmStar is an emulator specifically designed for WSN built in C, and it was first developed by University of California, Los Angeles. This emulator runs on Linux operating system. Besides libraries, tools and services, an extension of Linux microkernel is included in EmStar emulator [20].

OMNeT++

OMNeT++ is a network simulator built in C++. OMNeT++ provides a non-commercial license for academic institutions or non-profit research organizations, and a commercial license, for "for-profit" environments. Most of frameworks and simulation models in OMNeT++ are open sources [21].

J-Sim

J-Sim is a discrete event network simulator built in Java. It offers open source models and online documentation. This simulator is frequently used in physiology and biomedicine areas, but it can also be used in WSN simulation. In addition, J-Sim can simulate real-time processes [22].

ATEMU

ATEMU is an emulator of an AVR processor for WSNs built in C. It runs on Solaris and Linux operating systems. ATEMU is an explicit emulator for WSNs. It supports execution on TinyOS in MICA2 hardware. ATEMU can imitate not only the communication among the sensors, but also every instruction implemented in each sensor [23].

AVRORA

AVRORA is a simulator built in Java that is specifically designed for WSNs. It supports a wide range of tools that can be used to simulate WSNs as well as energy consumption simulation. This simulator is provided in open source and offers sufficient online documentation [24].

EmSim

EmSim is a program that simulates networks of 32-bit nodes running on the Linux-based EmStar software environment. Through EmTOS, EmSim enables using nesC applications as modules of the simulation. Additionally, the EmView visualization and analysis tool presents the topology and current status of the network [25].

2.2.2 Custom Simulation/Emulation Tools

Event-to-Sink Reliable Transport in Wireless Sensor Networks (ESRL)

The simulator was developed to create an evaluation environment and it was built in ns-2 [26].

Rate-Controlled Reliable Transport (RCRT)

The RCRT schema has been implemented in TinyOS 1.x for the motes and in C for a PC-class sink device running Linux. The RCRT module as implemented provides a transport layer interface that a sensor application can use to initiate a flow to the sink and send data packets [27].

DMP Scheme Simulator

The simulation model has been implemented in C. It evaluates the performance of the proposed DMP packet scheduling scheme while drawing comparisons against the FCFS (First Come First Served), and Multilevel Queue scheduling schemes [11]. The comparison is made in terms of average packet waiting time, and end-to-end data transmission delay.

Table 2 provides an overview comparison of the simulation/emulation tools presented in the current subsection. The comparison criterial are: Programming Language, Operating System, Open Source.

Table 2: Simulation/Emulation tools comparison

Name	Programming Language	Operating System	Open Source
NS2	TCL/C++	Linux/Cygwin	Yes
TOSSIM	-	Linux/Cygwin/ TinyOS	Yes
EmStar	C	Linux	Yes
OMNet++	C++	-	Yes*
J-Sim	Java	-	Yes
ATEMU	C++	Solaris/Linux	-
AVRORA	Java	-	Yes
EmSim	-	EmSim	-
ESRL	Built on Ns2	Linux/Cygwin	-
RCRT	C	Linux	-
DMPPS	C	-	-

3 THE EXAMINED WSN RESOURCE ALLOCATION SCHEME

The schemes and protocols intended for WSN MAC have a variety of approaches as well as priorities on creating an effective and reliable manner of managing the network. As it has been already shown, there are algorithms that focus on prioritizing the data, others focus on load levels while there are others that follow completely different approaches to achieve fast and efficient network architecture. From the research conducted for this project, there have not been found established schemes that prioritize the data, while still allowing the load to have an effect on time slot allocation. The scheme proposed in [28] does exactly that. In this section, we briefly present it and in the next one we evaluate it through simulation.

The proposed method assumes that there are different levels of queues, each corresponding to a particular traffic priority. The amount of bandwidth allocated is related to the load as well as traffic priority of that particular queue [28]. The specific resource allocation scheme introduces and uses formula (1) in order to derive a queue indication (Q_i) that is used to distribute the available bandwidth in terms of time division. Specifically, the capacity of the considered WSN is shared among all packet buffer proportionally to their Q_i values. The proposed formula affords discrepancies in the prioritization of data through factor z, a preset priority factor expressing the fraction of the channel access probabilities among two consecutive priority levels of traffic flows. On the other hand, the load (L_i) which denotes the current load of a particular priority queue at a given time (t), maintains a fixed impact on the queue indication.

$$Q_i(t) = z^{p_i} L_i(t) \quad (1)$$

Furthermore, this procedure is set in a way that allows the network to eradicate the effect that priority queues may have in the time slot allocation. The former can be achieved by setting the priority factor (z) of the formula to one. Similarly, the scheme can be arranged to diminish the effect that the load of a priority queue has in bandwidth allocation. The latter is achieved by setting an increased value for the priority factor (z).

4 RESULTS AND DISCUSSION

4.1 The Implemented Simulator

As mentioned earlier, this work targets to evaluate a particular resource allocation scheme by developing a specialized simulator. The simulator created models different scenarios that a resource allocation algorithm may encounter. In addition, the simulator measures the exported network metrics (such as throughput) and presents them in a convenient and complete manner. As a result, the simulator allows the researchers to derive recommendations and evaluate the impact of different factors on network performance. The simulator that has been created supports a wide range of different scenarios, and provides metrics that are directly affected by the scheme. It is important to note that the software is an abstract representation of the actual network. Most of the technical details that have been deemed surplus for this simulation are neglected.

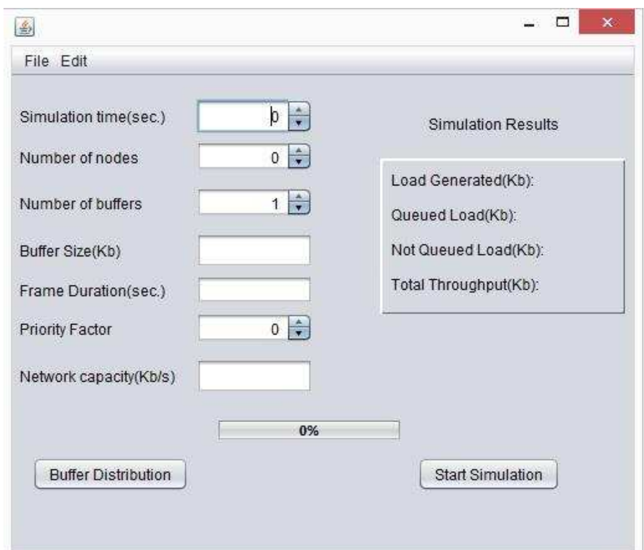

Figure 1. Main view of the developed simulator

The simulator created consists a network of nodes and a base station. All the nodes in the simulator communicate directly with the base station. Moreover, each node may contain up to eight buffers (queues) with each of them containing a specific amount of load. The simulator also supports super frames used by the evaluated resource allocation scheme in order to allocate slots of transmission for each buffer. Finally, the nodes use the allocated slots in order to transmit their data, by adding them to the super frame. Once the duration of a super frame is finished, another frame is created.

As it can be seen in Fig. 1, the simulator allows the user to choose the number of nodes, number of buffers (queues), size of the buffer, frame duration and network capacity, among others. As mentioned previously, the scheme focuses on the queue load and prioritization access of each queue. Moreover, it affords changes in the significance of queue priorities depending on specific preferences. The simulator supports that feature by allowing the user to choose the priority factor. As the priority factor increases, the priority of the queue is becoming more significant for the allocation decision.

Furthermore, the simulator supports different random number distributions (normal, uniform, Poisson and constant) of the data inter-arrival time as well as the load bursts length. Therefore, the simulator can support diverse settings of the arrival time as well as size of data, thus contributing towards a more realistic simulation. Finally, a summary of the metric results produced is presented on the main view of the simulator, while more comprehensive results about each buffer of specific priority are exported to a spreadsheet and presented through graphs. The source code of the simulator is available and provided upon request.

4.2 Performance Results

After running numerous simulations, while analysing and comparing the results, it was verified that the scheme performs just as it was expected and in accordance to the mathematical analysis presented in [28]. It can be adjusted to ignore priority and arrange resources according to queue load only, or it can be arranged to highpoint the priority, but still consider the load of each buffer (queue).

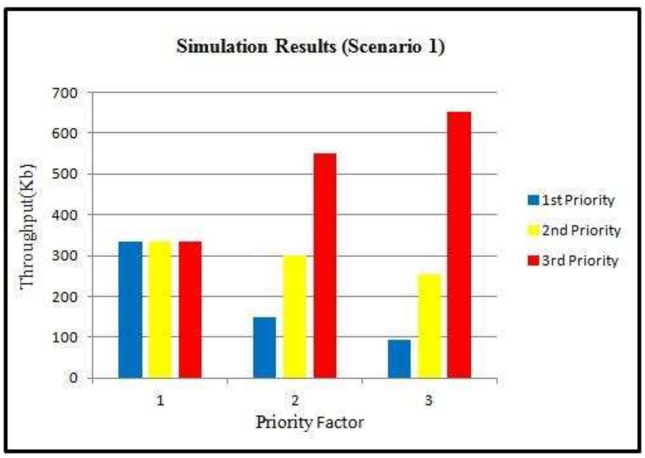

Figure 2. Network Throughput against Priority Factor for the examined resource allocation scheme

Fig. 2 illustrates the significance of the priority in resource allocation of the available bandwidth by performing three simulations in which the load is the same for all the buffers, while the priority factor of the scheme changes. Through this graph it can be clearly seen that despite the fact that the load of queues is similar, the throughput of higher priority buffers increases in a rapid rate analogously to the increase of the priority factor.

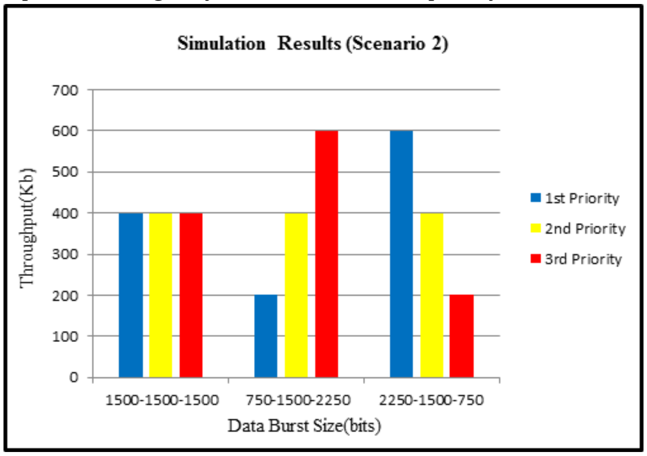

Figure 3. Network Throughput against Data Burst Size of the generated load for the examined resource allocation scheme

On the other hand, load also stands as an important variable to the resource scheme allocation. The executed simulations that disregard the impact of queue priority, presented in the Fig. 3, depict the growth of throughput corresponding to the burst of load (data inter-arrival time is set to be fixed in all executions). However, the impact of the load is slightly less significant, due to the linear relationship with the queue indication, and that could potentially raise issues in cases that a relatively low priority buffer has loads of data to transfer.

5 CONCLUSIONS

In this paper, we have presented and thoroughly discussed different schemes that deal with resource allocation in WSNs and provide support for QoS. As it has been shown, there are different criteria considered upon designing a bandwidth distribution algorithm. Therefore, the schemes discussed have been categorized into three main groups: priority-based protocols, energy efficient protocols, and hybrid (both).

Moreover, this work presents and evaluates a new scheme that had been developed by authors of this paper. In order to conduct a more precise evaluation, a new simulator has been built. The results provided by the simulator support the mathematical analysis that had been conducted for this scheme [28].

Nevertheless, this work provides important reassurances for the future work that is to be conducted. In the effort to provide complete QoS support for different types of contemporary monitoring applications [29], we aim to improve the simulator by adopting some of the resource allocation concepts discussed in this paper and make appropriate adjustments. The modelling and comparison against other cutting-edge schemes will be critical on affording recommendations aiming to optimize the proposed scheme.

ACKNOWLEDGMENTS

This work was partially supported by the Sheffield Research Undergraduate Experience (SURE) 2016.

REFERENCES

[1] X. Li, Wireless ad hoc and sensor networks, 1st ed. Cambridge: Cambridge University Press, 2008.

[2] I. Akyildiz and I. Kasimoglu, "Wireless sensor and actor networks: research challenges", Ad Hoc Networks, vol. 2, no. 4, pp. 351-367, 2004.

[3] P. Rawat, K. Singh, H. Chaouchi and J. Bonnin, "Wireless sensor networks: a survey on recent developments and potential synergies", The Journal of Supercomputing, vol. 68, no. 1, pp. 1-48, 2013.

[4] "Wireless Sensor Networks (WSN) 2014-2024: Forecasts, Technologies, Players: IDTechEx", Idtechex.com, 2016.

[5] Z. Jaffri, "A Survey on "Energy Efficient Routing Techniques in Wireless Sensor Networks Focusing On Hierarchical Network Routing Protocols"", International Journal of Scientific and Research Publications, vol. 4, no. 2, 2014.

[6] S. Abdollahzadeh and N. Navimipour, "Deployment strategies in the wireless sensor network: A comprehensive review", Computer Communications, vol. 91-92, pp. 1-16, 2016.

[7] T. D. Lagkas, P. Angelidis and L. Georgiadis, "Wireless Network Traffic and Quality of Service Support: Trends and Standards", IGI Global, Hershey, PA, 2010.

[8] R. Sudhaakar, S. Yoon, J. Zhao and C. Qiao, "A novel Qos-aware MAC scheme using optimal retransmission for wireless networks", IEEE Transactions on Wireless Communications, vol. 8, no. 5, pp. 2230-2235, 2009.

[9] M. Kim and C. Kang, "Priority-Based Service-Differentiation Scheme for IEEE 802.15.4 Sensor Networks in Nonsaturation Environments", IEEE Transactions on Vehicular Technology, vol. 59, no. 7, pp. 3524-3535, 2010.

[10] M. Gilani, I. Sarrafi and M. Abbaspour, "An adaptive CSMA/TDMA hybrid MAC for energy and throughput improvement of wireless sensor networks", Ad Hoc Networks, vol. 11, no. 4, pp. 1297-1304, 2013.

[11] N. Nasser, L. Karim and T. Taleb, "Dynamic multilevel priority packet scheduling scheme for wireless sensor network", IEEE Transactions on Wireless Communications, vol. 12, no. 4, pp. 1448-1459, 2013.

[12] R. de Paz Alberola and D. Pesch, "Duty cycle learning algorithm (DCLA) for IEEE 802.15.4 beacon-enabled wireless sensor networks", Ad Hoc Networks, vol. 10, no. 4, pp. 664-679, 2012.

[13] S. Gobriel, D. Mosse and R. Cleric, "TDMA-ASAP: Sensor Network TDMA Scheduling with Adaptive Slot-Stealing and Parallelism", 2009 29th IEEE International Conference on Distributed Computing Systems, 2009.

[14] K. Kim, W. Choi, M. Whang and H. Youn, "CA-MAC: Context Adaptive MAC Protocol for Wireless Sensor Networks", 2009 International Conference on Computational Science and Engineering, 2009.

[15] L. Tang, Y. Sun, O. Gurewitz and D. Johnson, "PW-MAC: An energy-efficient predictive-wakeup MAC protocol for wireless sensor networks", 2011 Proceedings IEEE INFOCOM, 2011.

[16] Injong Rhee, A. Warrier, M. Aia, Jeongki Min and M. Sichitiu, "Z-MAC: A Hybrid MAC for Wireless Sensor Networks", IEEE/ACM Transactions on Networking, vol. 16, no. 3, pp. 511-524, 2008.

[17] B. Srestha, E. Hossain, S. Camorlinga and R. Krishanmoorthy, "An Optimization-Based GTS Allocation Scheme for IEEE 802.15.4 MAC with Application to Wireless Body-Area Sensor Networks", IEEE ICC, 2010.

[18] "The Network Simulator, NS–2" [Online]. Available: http://www.isi.edu/nsnam/ns/. [Accessed: 17-Mar-2017].

[19] P. Levis, N. Lee, M. Welsg and D. Culler, "TOSSIM: Accurate and Scalable Simulation of Entire TinyOS Applications." In Proc. 1st ACM Int. Conf. Embedded Networked Sensor Systems (SenSys), Los Angeles, CA, pp. 126–137, 2003.

[20] L. Girod, J. Elson, A. Cerpa, T. Stathopoulos, N. Ramanathan, and D. Estrin, "EmStar: A software Environment for Developing and Deploying Wireless Sensor Networks." In Proceedings of USENIX 2004, Boston, MA, pp. 283–296, 2004.

[21] OMNET++ discrete event simulator. [Online]. Available: http://www.omnetpp.org. [Accessed: 29-Dec-2016].

[22] A. Sobeih, J. C. Hou, L-C. Kung, N. Li, H. Zhang, W-P Chen, H-Y. Tyan and H. Lim, "J-Sim: A Simulation and emulation environment for wireless sensor networks", IEEE Wireless Communications, vol. 13, no. 4, pp. 104-119, 2006.

J. Polley, D. Blazakis, J. McGee, D. Rusk, J. S. Baras and M. Karir, "ATEMU:

A Fine-grained Sensor Network Simulator." In Proc. 1st IEEE Int. Conf. Sensor
[23] and Adhoc Communication Networks (SECON'04), Santa Clara, CA, October
2004.
"Avrora - The AVR Simulation and Analysis Framework",
Compilers.cs.ucla.edu, 2017. [Online]. Available:
[24] http://compilers.cs.ucla.edu/avrora/. [Accessed: 13- Nov- 2016].
L. Girod, T. Stathopoulos, N. Ramanathan, J. Elson, D. Estrin, E. Osterweil and
T. Schoellhamme,r "A System for Simulation, Emulation and Deployment of
[25] Heterogeneous Sensor Networks." In Proc. 2nd ACM Int. Conf. Embedded
Networked Sensor Systems (SenSys), Baltimore, MD, pp. 201–213, 2004.
O. Akan and I. Akyildiz, "Event-to-sink reliable transport in wireless sensor
networks", IEEE/ACM Transactions on Networking, vol. 13, no. 5, pp. 1003-
1016, 2005.
[26] J. Paek and R. Govindan, "RCRT", ACM Transactions on Sensor Networks,
vol. 7, no. 3, pp. 1-45, 2010.
T. D. Lagkas, P. Angelidis, D. G. Stratogiannis and G. I. Tsiropoulos,
[27] "Analysis of Queue Load Effect on Channel Access Prioritization in Wireless
Sensor Networks," in Distributed Computing in Sensor Systems Workshops
[28] (DCOSSW), 2010 6th IEEE International Conference on, Santa Barbara, CA,
2010.
T. D. Lagkas and G. Eleftherakis, "An Overview of Wireless Sensor Networks:
Towards the Realization of Cooperative Healthcare and Environmental
Monitoring," Handbook of Research on Progressive Trends in Wireless
[29] Communications and Networking, IGI Global, p. 317-339, 2014.