MAC Protocols and Mobility Management for Real-time Applications Using Wireless Sensor Networks

Muhsin Atto and Chris Guy
University of Reading, Reading, United Kingdom
{kp003919, c.g.guy}@reading.ac.uk

***Abstract*— Wireless Sensor Networks detect events using one or more sensors, then collect data from detected events using these sensors. This data is aggregated and forwarded to a base station (sink) through wireless communication to provide the required operations. Different kinds of MAC and routing protocols are needed for WSN in order to guarantee data delivery from the source nodes to the sink. GinMAC has been simulated for real-time applications, where energy saving, delay and reliability need to be considered. A Mobility module for GinMAC has been implemented. It has been showed that GinMAC can be applied for real-time applications in both mobile and static scenarios. However, further improvements of its performance are needed. Some future plans to investigate this are given at the end of this paper.**

Keywords*–*WSN; Real-time Applications; GinMAC; Mobility; Castalia.

I. Introduction

Wireless Sensor Networks(WSNs) detect events using one or more sensors, then collect data from detected events using these sensors. This data is aggregated and forwarded to a base station (sink) through wireless communication to provide the required operations. There are a lot of challenges that need to be considered before designing protocols for WSNs. Different kinds of Medium Access Control (MAC) and routing protocols need to be designed for WSN in order to guarantee data delivery from the source nodes to the sink in different applications.

The initial applications supported by WSNs were mostly in environment monitoring, such as temperature monitoring for a specific area, house alarms, and so on. The main objectives in such applications only involved simple data processing. Energy consumption needed to be considered for specific applications, so little attention was taken on data delivery and reliability related issues such as in [1][2][3].

WSNs have been extended and their design have been advanced to support more hard design and complex applications, such as security, military, fire detection and health care related applications. In these applications, data delivery and reliability must be taken as important parameters in addition to energy efficiency, because data must be collected from the sources of events and be forwarded to the sink in real time with high reliability, otherwise the application will be useless.

In this paper, an implementation of GinMAC [4] including the proposed mobility management module is described for real-time applications, where energy saving, delay and reliability for end to end data delivery over multi hop WSNs need to be considered. An example of the proposed real-time application in this paper is healthcare, where data needs to be collected from the body of the patients and then sent to a sink, for example [5].

The rest of the paper is structured as follow. Motivations for the paper will be given in Section II, then the implementation of GinMAC for both static and mobility applications will be described in Section III. Simulation scenarios and required parameters with figures showing the results for GinMAC implementation for the proposed applications will be given in Section IV. The simulation results and some discussion is in Section V. A conclusion and future work are presented in Section VI.

II. Motivations

Most of the recently proposed protocols for WSNs consider either energy saving or reliability for the target applications, none of them have considered both performance metrics at the same time [1]. However, some applications may need to guarantee both energy saving and reliability at the same time without any errors, otherwise the applications may be useless. Therefore, in order to provide this, new and very efficient MAC protocols need to be designed. Previous work has shown that GinMAC is the only protocol, which can be used for real-time applications to provide the required performances as shown in [6]. The novel motivations for this paper are the following:

- Design MAC protocols for real-time applications where the required energy saving, reliability and delay for data delivery need to be considered.
- Design mobility management modules for real-time applications.
- Adapt GinMAC to add new features to improve its applicability to real-time applications which require mobility, such as healthcare applications as described in [5].
- Simulate a GinMAC implementation including the proposed mobility management module for real-time applications.

III. MAC protocols for Real-time Applications

It was concluded in [6] that GinMAC is a possible MAC protocol for use in real-time applications, where reliability, energy saving and delay can be guaranteed. Challenges and requirements that need to be considered before designing any MAC protocols for such applications are also described in the same paper. The implementation of GinMAC including a mobility management module is described in this Section.

A. Implementation of GinMAC for Real-time Applications

GinMAC [7] is a Time Division Multiple Access (TDMA) based MAC protocol, so energy saving and reliability with bounded delay can be achieved. However, an efficient synchronization and slot allocation algorithms need to be designed in order to allocate the required slot time for each node in the network and let the radio of the nodes be turned on only in the allocated time. In this case, each node needs enough slots of time to transmit data toward a sink, including control messages such as messages for slots permission, mobility and topology control related messages. GinMAC has been modified to add new features to improve its applicability to applications which require mobility, such as healthcare applications. The GinMAC implementation in [7] does not support mobility while this once does. Topology management and time synchronization for GinMAC in this implementation are described below.

1) Slot allocations in GinMAC: GinMAC is a TDMA based protocol and assumes that data is forwarded hop by hop toward a sink using a tree based topology, consisting of n nodes. Time in GinMAC is divided into a fixed length called *Epoch E*, each E is subdivided by $n*k$ time slots so that each node allocates k slots for transmitting data toward its parent until it reaches a sink. Each node is assigned k exclusive slots with four different types, which are basic slots (*TX,RX*) for data transmitting and receiving, additional slots (*RTX,RRX*) for re transmitting, broadcast slots (*BROD*) for topology control between nodes in the network and unused slots (U) for saving energy (if any). More details about how these slots are used can be found in Figure 1.

Additional slots are used only for re transmission to perform required reliability for the target applications, these slots are used even in the case when no data is available for transmission, as described in [4]. Unused slots are used for saving energy when data can not be delivered using basic and additional slots. This implementation for GinMAC does not contain unused slots, but they may be used in the future for increasing the lifetime of the network. Broadcasting slots are used for topology control. Slots for each node need to be allocated according to the defined topology so that the required performance can be achieved.

2) GinMAC Topology Control Management: GinMAC is a tree based WSN topology so that each node transmits its data toward a sink in its allocated slots and sleeps for the rest of the time. The current static topology that is proposed is a WSN with 13 nodes with static slot allocation, each node has enough slots of time to transmit all data from its children and its own, including control messages toward a sink. GinMAC supports mobility for leaf nodes and this will required to design new topology control and management algorithms to provide connectivity between static and mobile nodes in the network. It is assumed that the Base Station (BS) has adequate power to reach all nodes in the network using down-link slots. However, the sensor nodes cannot always do this because of their limited power supply.

A node added to the network must determine in which slots it must become active before it can transmit or receive data. The steps used to achieve this are described below. After a node is switched on, it must first ensure time synchronization with the rest of the nodes in the network. Both control and data messages transmitted in the network can be used to obtain time synchronization. The node continuously listens to overhear a packet from the sink. After overhearing one message, the node knows when the GinMAC frame starts as each message carries information about the slot in which it was transmitted.

As a next step, the node must find its position in the topology which must stay within the defined topology envelope. For this purpose, the new node listens for packets in all slots. Transmitted data packets from a sink use a header field in which a node that is ready for transmission can find its information and then according to this information starts and stops data transmission toward its parent. A node may be configured with a list of valid nodes or clusters that it is allowed to attach to when mobility is supported. This might be necessary to ensure that a node will only attempt to join the network using known good links as determined by measurements before the deployment to provide the required performance.

3) Synchronization Messages for GinMAC: At the start of each frame, the sink needs to broadcast a synchronization packet which it is denoted as *SYNCH* into the network. This packet holds the start time, end time and slot numbers for each node in the network. When nodes receive a SYNCH packet from the network, they will extract their information from the SYNCH packet and then discard it. In this case, CSMA is used by the sink to synchronize nodes in the network and nodes use TDMA to transmit their data to their parents. After nodes receive their slot information from the sink, they need to ask permission for data transmission from their parents. Then, after slots related information has been received by a node, it has to handshake with its parent and then can start to transmit data. After a node uses its allocated slots, it can go to sleep and wake up at the same time in the next frame. Each node in this case will access the channel using their unique start time, so this will avoid any chance of collision with transmissions from other nodes in the network.

GinMAC lets nodes and their parents be active at the same time so that data can be transmitted between them. This synchronization algorithm for GinMAC synchronizes nodes with their parents so that parents can be active in the same time when one of their children is active. This time synchronization algorithm is good enough to deliver packets with the required performances for the applications described in Section IV.

The core idea behind this GinMAC implementation is to

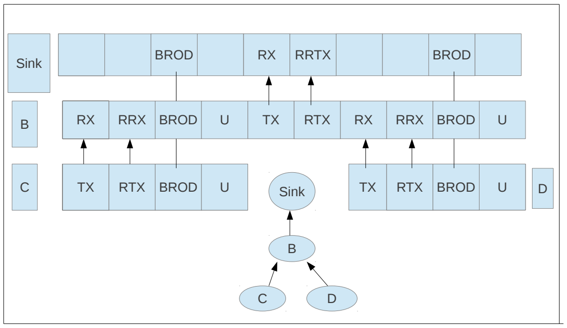

Fig. 1. Slot allocations and Synchronizations for nodes in the GinMAC

let nodes sleep as much as possible without effecting data delivery and required maximum delay, and this can only be performed using a TDMA based technique. The static topology is designed to let nodes have enough slots to transmit their data and in the rest of the frame go to sleep. The slot allocation and synchronization for GinMAC can be found in the Figure 1.

B. Mobility for Real-time Applications Using GinMAC

A new challenge is posed when mobility needs to be considered in a WSN for different applications such as healthcare, vehicle and monitoring people related applications. In this case topology control, resource management and performance control need to be designed to provide good connectivity between static and mobile nodes in the network and provide the required performance. Mobility and topology control for critical applications using WSNs are described in [8][9][10]. The proposed mobility management module in this paper follows the same messages and concepts as in the above papers.

1) Mobility Management Module for GinMAC: There may be cases when moving from one location to another in the network effects the connectivity of the network and then reconfiguration algorithms are needed. In order to support mobility for real-time applications, control messages which need to be transferred between static and mobile nodes to find a better attachment have been defined. Some of the possible control messages are Advertisement(*ADV*), join (*JOIN*), and join acknowledgement *JOIN ACK* messages. Static nodes are called *Clusters*.

When clusters switch on their radios, they need to send ADV to the network and then wait some time, then when mobile nodes receive these ADV messages they will ask to join the network. When clusters receive JOIN messages from the mobile nodes they will send back a JOIN ACK message to let the mobile node know that request to join has been accepted. So using these control messages connectivity between mobile nodes and cluster nodes will be established. In the proposed application only leaf nodes are allowed to be mobile nodes and all other nodes are clusters. The mobility module lets nodes move across a line between mobile nodes and a sink.

Mobile nodes may have more than one cluster they could join so they have to decide which cluster will be selected for transferring data toward their parents. In this GinMAC implementation, the cluster with maximum Receiver Signal Strength Indicator (RSSI) is considered the best one to be selected for the new attachment. Cluster nodes send ADV including available positions over time and when mobile nodes receive ADV, they compare the RSSI from their current parents to the received RSSI from the current ADV messages. In the case that a new cluster has better RSSI, mobile nodes need to leave their current parents and attach to this new cluster which is included in the currently received ADV message. When a new attachment is selected then a join request needs to be sent to that cluster. Upon receiving the JOIN request from a mobile node, JOIN ACK needs to be sent by the selected clusters.

Slots in the each frame need to be updated according to the new attachments, mobile nodes need to release the first tree position after it is attached to the second tree address, so in this case slots allocated for the new clusters need to be increased and slots allocated for the old clusters need to be decreased. A new algorithm for updating slots is needed for GinMAC to balance allocated slots for nodes according to the different attachments. A new algorithm has been designed to update channel according to new movements and changes in the topology of the network. Table II shows results from this mobility module describing different attachments between a mobile node and different clusters for transmitting packets toward a sink.

2) Move Detection in GinMAC: There are some cases when nodes can move without being detected. For instance, clusters may be unaware of leaving mobile nodes and then will keep space in the channel for that particular node. This will consume more energy and reduce the reliability of the network. There may be cases when clusters are not available for attachment any more without letting mobile nodes know. So an additional two control messages for this new mobility module for the proposed MAC protocol have been used, which are denoted by *KEEPALIVE* and *NODEALIVE*. *KEEPALIVE* control message is used by clusters to let its currently attached mobile nodes know that this cluster is still available and *NODEALIVE* message is used by mobile nodes to let their attached clusters know that they are still available for attachment. Mobile nodes wait for a specific interval to receive messages from the attached clusters, if they do not receive anything during that interval, a *NODEALIVE* message needs to be sent, to let a cluster know that they still want to use that cluster. If no reply is received then mobile nodes need to search for a new address to make a new attachment.

IV. SIMULATION SCENARIOS FOR GINMAC IMPLEMENTATION

The GinMAC protocol is compared with TMAC [11] in terms of energy saving, reliability and delay for end to end data delivery for WSNs having static and mobile nodes. More details about simulation parameters and scenarios are given below.

A. Simulation Scenarios and parameters

Castalia has been used in this work, because of its capabilities for simulating protocols for WSNs based on the real data, as shown in [12]. Both MAC protocols were simulated according to the application requirements given in the following sections, using different packets rates. We define packet rate by Rp = packets per second, so as we can see from the graphs in our simulation results that *R1* means nodes send 1 packet per second, *R2* means nodes send 2 packets per second and so on. More details about the topology of the deployed WSN, MAC protocols and other parameters can be found in Table I.

TABLE I
SIMULATION PARAMETERS

Paremeter	Value
MAC Protocols	GinMAC and TMAC
Network Dimensions(in meters)	90 X 90
Distance Between pair of nodes	20 meters
Simulation Duration	10 minutes
Measurement Metrics	Life time, delay and reliability
Number of Nodes	13
Packet Rates (packet(s) per second)	1-5
mobility speed(meters in seconds)	5
mobility interval(in minutes)	1
Advertisement interval (in seconds)	15
Initial Energy(in Joules)	18720
Real Radio	CC2420

B. Proposed Applications for GinMAC Simulation

A simple application where all nodes send data towards a sink using different packet rates is used. The proposed application in this paper is healthcare related, where data needs to be collected from the body of patients and then sent back to a base station. Some threshold values have been selected for the proposed application, which need to be achieved before deciding if these MAC protocols can be used for the target application or not.

*1) **Reliability**:* The reliability of given protocols is defined as the ratio between total packets generated and sent by source nodes and total packets received by a sink or a final destination. So Reliability = (Received Packets/Sent Packets). The threshold value for reliability needs to be high enough in order to achieve the requirements for the proposed application. When mobility is supported this threshold may be reduced, therefore, reliability is highly dependent on the applications. Information from the patients needs to be accurate and delivered to a sink as soon as possible to make the required operations. Based on this, it has been assumed that at least a reliability of 0.99 for the static scenarios and a reliability of 0.98 for the mobility scenarios must be achieved.

*2) **Energy Saving and Lifetime**:* Energy saving is one of the most important challenges needed to be considered in any applications when designing WSNs. Each node has a small battery with limited power which cannot be recharged very often, so each protocol needs to consider consumed energy for each node in order to extend the life time of the entire network. The lifetime of the networks is the maximum days that a WSN can survive, whilst spending energy at the given rate. Let consumed energy by each node be denoted by C joules, initial energy by E joules and current simulation time by T seconds, then the lifetime of given MAC protocols for each node in the network has been calculated as follow:

TABLE II
DIFFERENT ATTACHMENTS USING THE PROPOSED MOBILITY MODULE FOR GINMAC SEE SECTION III-B1

Time(sec)	RSSI(dBm)	Cluster	current RSSI(dBm)	Required Action
0.0356	-76.10	10	-100	new attachment
15.0051	-86.05	1	-76.10	no action
15.0081	-88.02	5	-76.10	no action
15.0224	-84.24	9	-76.10	no action
15.02667	-75.05	6	-76.10	new attachment

$$LifeTime(indays) = ((E/C) * T)/86400 \qquad (1)$$

where *86400* is number of seconds in each day. We define life time as how many days each node will be alive consuming C joules as described in the above equation. It has been assumed that nodes in the proposed healthcare application can be recharged every week.

*3) **Delay Calculation**:* Delay is defined as the difference between the time when each packet is sent from its source node to the time when the same packet is received by its final destination. Delay in real-time applications needs to be measured so that all data must be delivered within bounded delay, i.e., each packet that is delivered after this delay is considered to be lost and will be ignored. All data needs to be collected from the source nodes and then delivered to the sink within a bounded delay. It was decided that a delay of more than 10 seconds for delivering data to a sink may risk the health of the patients.

C. Simulation for Static and Mobility Scenarios

GinMAC and TMAC have been simulated with different packets rates as shown in each graph, the WSN topology is also given below. The results graphs from running simulation for both static and mobility scenarios are shown below.

V. SIMULATION RESULTS AND CONCLUSION

1) Packets Delivery and Reliability: It is shown in the Figure 3 that GinMAC can offer the applications requirements in term of reliability (as defined in Section IV) at various packet rates. From packet rates 1 to 4, GinMAC delivers more than 0.99 of packets from source nodes to a sink. Thus, it can be said that GinMAC can be used for the proposed applications. However, TMAC cannot offer a reliability of more than 0.91 in both high and low packet rates using the same parameters as shown in the Figure 3.

GinMAC cannot offer reliability of more than 0.97 for packets rates more than 4 and if it is needed to increase this reliability for such packet rates, the number of slots for nodes in the static topology may be increased. But, GinMAC performs better than TMAC in both high and low packet rates

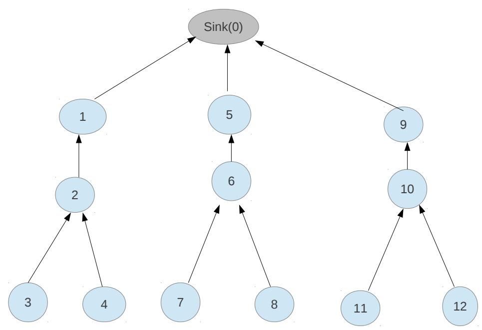

Fig. 2. Topology of the WSN in our Simulation

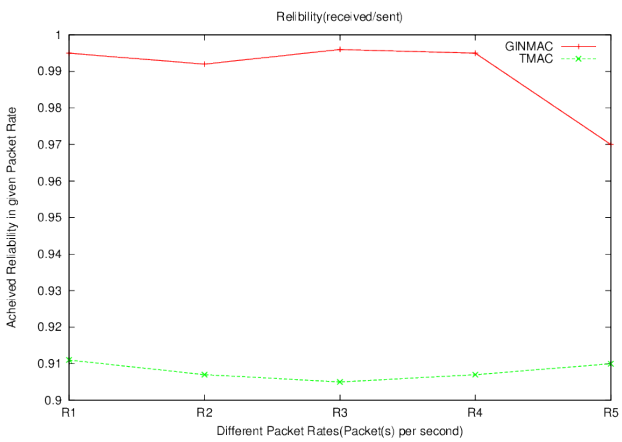

Fig. 3. Performances in term of Reliability using TMAC and GinMAC for Static scenario using different packet rates see section IV-A

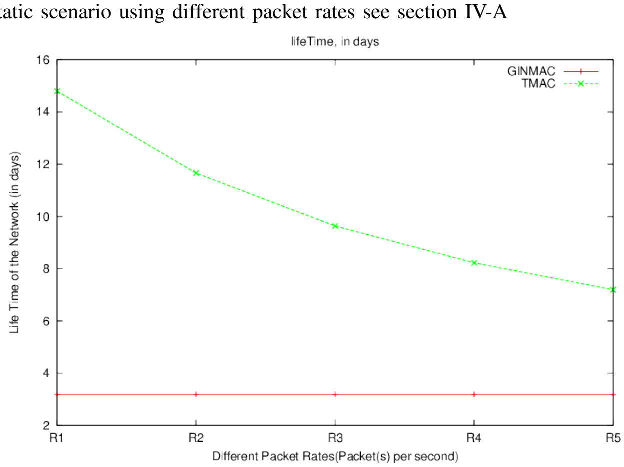

Fig. 4. Performances in term of life time of the nodes in networks using GinMAC and TMAC for Static scenario using different packet rates see section IV-A

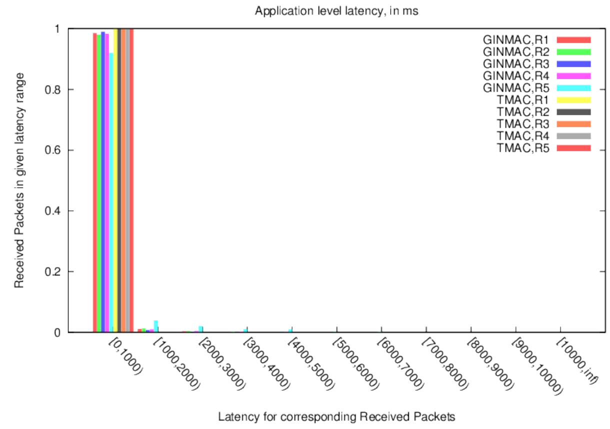

Fig. 5. Latency for delivered packets using TMAC and GinMAC for Static scenario using different packet rates see section IV-A

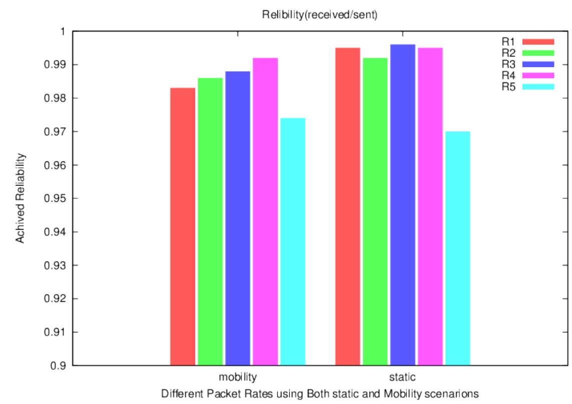

Fig. 6. Performances in term of Reliability using both mobility and static scenarios for GinMAC using different packet rates see section IV-A

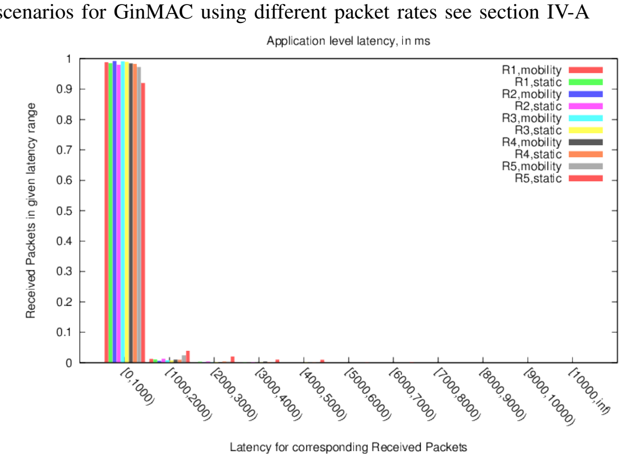

Fig. 7. Latency for delivered packets using both mobility and static scenarios for GinMAC using different packet rates see section IV-A

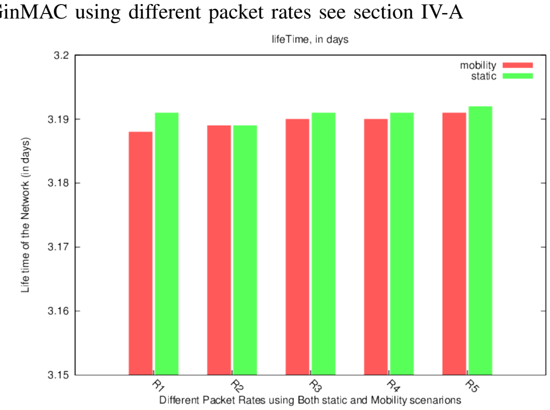

Fig. 8. Life time of the nodes in the network using both mobility and static scenarios for GinMAC using different packet rates see section IV-A

 ISBN: 978-1-61208-284-4

as shown in the Figure 3. Therefore, the conclusion is that TMAC can not be applied for applications where reliability of more than 0.91 needs to be guaranteed at any packet rates. This is expected to be so since GinMAC uses static slots allocation and no interference can occur. TMAC uses contention based techniques and hence collisions are more likely to happen.

2) Energy Saving and Lifetime: Figure 4 shows the average life time of the nodes in the network and shows that TMAC performs better than GinMAC in terms of energy saving in both low and high packet rates. This is expected to be so, because of the adaptive related techniques for TMAC, which depend on the on-going traffic in the network nodes which need to be active. In this case, TMAC lets nodes sleep a lot at low packet rates, however, GinMAC does not consider traffics for sleeping nodes. GinMAC consumes the same amount of energy for all given packet rates. This is expected to be so, because according to the GinMAC design specification given in [4] GinMAC does not consider traffic for sleeping nodes and nodes are active in their allocated slots even in the case where no packets are available for transmission to achieve the required reliability and bounded delay. Hence, nodes will consume nearly the same amount of energy at both high and low packet rates. A WSN using GinMAC can survive 3.19 days which is not enough to be used for these applications without improvement.

3) Delay for Data Delivery: In Figure 5, *MaxLatency* defines the bounded latency that all packets need to be delivered, which represents the threshold for latency in the proposed applications and *MaxColumns* defines the number of columns to be used for measuring the latency for given MAC protocols. Any delivered packets after the last column are considered to be lost and may be discarded. GinMAC does not perform better than TMAC in term of latency as shown in the Figure 5, however this performance is enough to be used for the proposed applications. According to the results from Figure 5, most of the packets (which is about more than 0.98 of received packets) are received before first 5 seconds from packet rates 1 to 5, and the rest of packets are received within the last 5 seconds and hence all packets are received within 10 seconds.

A. Results from the Mobility Scenario

Mobility module for GinMAC uses the RSSI to select a better attachment. It has been assumed that there is only one mobile node however, more than one node can be mobile for the leaf nodes in the proposed mobility management module. Figures 8 and 6 show that GinMAC offers nearly the same reliability and lifetime for both static and mobility scenarios. Figure 7 shows that latency is also the same as in static topology and the required delay performance is offered using the mobility module.

VI. Conclusion and Future Work

An implementation of GinMAC including the proposed mobility module for real-time applications were described in this paper, where the required performance needs to be guaranteed. It has been shown that GinMAC can be used for the target applications, except that the life time needs to be improved. A mobility module has been designed and simulated for GinMAC. The results from the mobility module have also showed that GinMAC implementation can give nearly the same performance in both mobility and static scenarios for the proposed applications. GinMAC can not provide the required routing for the proposed applications, hence efficient routing protocols need to be designed in the future to cooperate with GinMAC in order to provide the required routing and extend the life time of the network for the proposed applications.

Acknowledgement

The authors would like to thank Mr James Brown from the University of Lancaster in UK, for his help for providing the basic ideas about the GinMAC and its specification. The authors would also like to thanks Dr Athanssions Boulis for information about Castalia simulator which helped a lot.

References

[1] P. Suriyachai, U. Roedig, and A. Scott, “A survey of mac protocols for mission-critical applications in wireless sensor networks,” Communications Surveys Tutorials, IEEE, vol. 14, no. 2, 2012, pp. 240 – 264.

[2] J. Lotf and S. Ghazani, “Overview on routing protocols in wireless sensor networks,” in Computer Engineering and Technology (ICCET), 2010 2nd International Conference on, vol. 3, April 2010, pp. 610 – 614.

[3] M. Soyturk and D. Altilar, “Reliable real-time data acquisition for rapidly deployable mission-critical wireless sensor networks,” in INFOCOM Workshops 2008, IEEE, April 2008, pp. 1 –6.

[4] P. Suriyachai, J. Brown, and U. Roedig, “Time-critical data delivery in wireless sensor networks,” in 6th IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS ’10). IEEE, June 2010, pp. 216–229.

[5] H. Huo, Y. Xu, H. Yan, S. Mubeen, and H. Zhang, “An elderly health care system using wireless sensor networks at home,” in Sensor Technologies and Applications, 2009. SENSORCOMM ’09. Third International Conference on, June 2009, pp. 158 –163.

[6] M. Atto and C. G. Guy, “Wireless sensor networks: MAC protocols and real time applications,” in The 13th Annual Post Graduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting (PGNet2012) (PGNet2012), Liverpool, UK, United Kingdom, June 2012, pp. 1 – 6.

[7] J. Brown and U. Roedig, “Demo abstract: Ginlite - a mac protocol for real-time sensor networks,” in In Proceedings of 9th European Conference on Wireless Sensor Networks (EWSN’12), Feb 2012.

[8] Z. Zinonos, R. Silva, V. Vassiliou, and J. Silva, “Mobility solutions for wireless sensor and actuator networks with performance guarantees,” in Telecommunications (ICT), 2011 18th International Conference on, May 2011, pp. 406–411.

[9] Z. Zinonos and V. Vassiliou, “Inter-mobility support in controlled 6lowpan networks,” in IEEE Globecom 2010 Workshop on Ubiquitous Computing and Networks (UbiCoNet2010), Miami, Florida, USA, Dec 2010, pp. 1718–1723.

[10] R. Silva, Z. Zinonos, J. Sa Silva, and V. Vassiliou, “Mobility in wsns for critical applications,” in Computers and Communications (ISCC), 2011 IEEE Symposium on, July 2011, pp. 451–456.

[11] Y. Tselishchev, A. Boulis, and L. Libman, “Experiences and lessons from implementing a wireless sensor network mac protocol in the castalia simulator,” in Wireless Communications and Networking Conference (WCNC), 2010 IEEE, April 2010, pp. 1–6.

[12] A. Boulis, “Castalia simulator: A simulator for wireless sensor networks and body area networks,” in http://castalia.npc.nicta.com.au, NICTA, [retrieved: April, 2013], pp. 1 – 120.