An Efficient MAC Scheme in Wireless Sensor Network with Energy Harvesting (EHWSN) for Cloud Based Applications

Farrukh Shahzad and Tarek R. Sheltami
Computer Engineering Department
King Fahd University of Petroleum and Minerals
Dhahran 31261, Saudi Arabia
Email: farrukhshahzad, tarek@kfupm.edu.sa

***Abstract*— Wireless sensor networks (WSNs) are utilized in various applications and are providing the backbone for the new pervasive Internet, or Internet of Things (IoT). Researchers are working on merging other emerging technologies and services such as cloud computing and Big data with IoT to design and build smart cities. Energy consumption is one the important constraint in the design of WSN based system. The sensor nodes typically rely on power sources like batteries which also makes it bulky. Recent advances in ambient energy harvesting technology have made it a potential alternative energy source for WSNs. However, this energy is not available continuously at the desired level and therefore the node need to utilize the sporadic availability of energy to sense and transmit the data efficiently and as quickly as possible. Energy Harvesting-based WSNs (EHWSNs) are the result of granting nodes the capability to extract energy from ambient sources. Due to limited power, it is imperative to optimize every aspect of the WSN including communication protocols. The heart of all this optimization is the medium access control (MAC) layer. In this paper, we study the performance of different MAC schemes for WSNs. Later we proposed a new scheme for better utilization of EHWSN in certain application and provide simulation and experimental results. The cloud services provide the resources to process the information from hundreds of regional IoT systems at a higher level.**

Keywords—Wireless sensor network, Energy harvesting, MAC protocol, Internet of things, Cloud application, Smart cities, Big data.

I. INTRODUCTION

Wireless Sensor Networks (WSNs) have been employed in many important applications such as intrusion detection, object tracking, industrial/home automation, smart structure and several others. Each node main task is to send sensed/detected data to a gateway (also called sink or base station) that processes the received data. WSNs design has to satisfy some important characteristics including scalability, low energy consumption, fault tolerance, quality of Service (QoS): latency, data integrity, security, and low cost.

A major problem in deploying WSNs is their dependence on battery power. Thus it is vital and imperative to optimize every facet of the communication protocols to utilize power efficiently [3]. The heart of all this optimization is the medium access control (MAC) layer. The prime role of the MAC is to coordinate access to and transmission over a medium common to several nodes. A multitude of MAC protocols for WSNs have been proposed in literature, and an exhaustive list can be found in [1][2][3]. These detailed surveys have classified MAC protocols into groups based on their some performance matrix, design criteria, and/or other aspects.

In future, it is forecast that fewer WSN nodes will rely on batteries and the majority will be powered by energy harvesting sources [4]. This is true for emerging technologies and services such as smart cities which are utilizing state-of-the-art research in *Big data* and Cloud computing with all its challenges [19]. The dream to analyze massive amounts of information and to connect devices (IoT) became a reality due to availability of affordable cloud related services. "*Smaller devices (sensors) produces Big data*" is a new jargon in the cloud world which means smaller, cheaper and lightweight sensors will be installed in billions and generates petabytes of data. This is now possible to handle and process all this data thanks to cloud infrastructure and services. Some of the benefits of utilizing cloud services include:

- Applications can be deployed as web services,
- Solutions run in a highly scalable cloud environment,
- Program and data are maintained in a secure cloud environment.

In this work, our primary focus is to study the performance of energy harvesting WSN based IoT systems in terms of MAC protocols. The processing and analysis of massive data generated by the billions of sensors and IoT devices entail deployment of scalable and elastic cloud-based applications and services. These cloud-based applications must be designed also to guarantee quality of service performance metrics such as latency and request loss. The authors in [20][21] show how to deploy elastic and scalable cloud-based applications with the deployment of an overlay network involving load balancer and cloud virtual nodes.

A. Energy harvesting WSN

Energy Harvesting-based WSNs (EHWSNs) can exploit different sources of energy, such as solar power, wind,

978-1-4673-6773-8/15/$31.00 ©2015 IEEE

mechanical vibrations, temperature variations, magnetic fields, etc. Fig. 1 illustrates the block diagram of an EHWSN. Here the energy harvesting and energy storage devices form the energy harvesting power management system, which enable WSN nodes to last potentially forever or as long as the life of the node hardware itself, by continuously providing energy and/or storing it for future use [5].

B. *Principle of operation*

In a EHWSN node, harvested energy is accumulated in a storage device. A node can only start operating when a certain level of energy is reached. The node can't operate until this minimum energy (E_{min}) is reached and will eventually be inoperable again after few transmissions. The EH system will charge the storage device again. This process is repeated continuously. Since storage devices such as super-capacitors offer virtually unlimited recharge cycles, EHWSN can potentially operate for very long periods of time (years or even decades) without the need to replenish its energy manually.

C. *MAC protocol for EHWSN*

There is huge demand of utilizing EH in WSN which indeed introduce challenges in its implementation from hardware design to communication stack. But with challenges, there are research opportunities in the development of EHWSN. In this work, we will focus on the 'energy aware' communication protocol or MAC protocol designed/adopted for EHWSN [5][6][7][8].

In this research, our main contribution is the design and analysis of a new MAC scheme that specifically exploits the unpredictable nature of the energy harvesting process. Secondly, we adopted a new python based framework for simulation called Pymote and extend its functionality to simulate low level communication protocols.

The rest of the paper is organizes as follows: In section II, we provide some background related to MAC protocols in general and MAC protocols for EHWSN in particular. We present our proposed scheme in section III and provide simulation and experimental results in section IV. We conclude in section V.

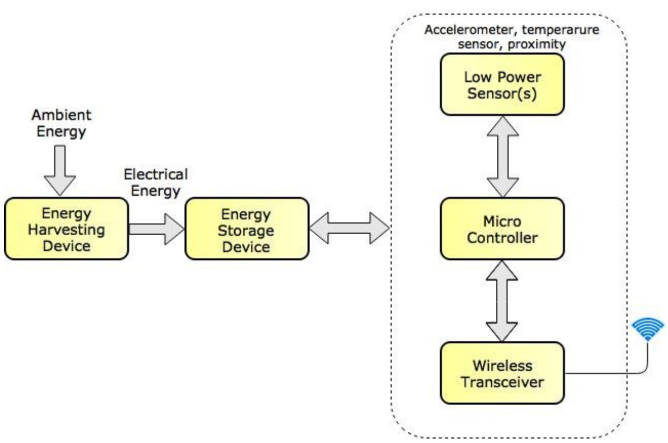

Fig. 1. Block diagram of an EHWSN [adopted from 5]

II. Background and Related Work

The EH technology will benefit other innovations like "Internet Of Things" (IOT). The ideal EH solution for such system is a highly-efficient, eco-friendly, power generation system that can be cycled continuously for the life of the product. The idea is to eliminate the need to replace batteries as well as the need for battery disposal [5]. In literature, EHWSN are also referred to as Wireless Sensor Networks Powered by Ambient Energy Harvesting, or WSN-HEAP. The communication protocol in EHWSN not only concern about the efficient radio communication but need to take into consideration the charging state of EH system in its design and operation.

The challenges in adapting EH in WSN are discussed in [8] in terms of hardware, node placement, middleware and communication protocols. The opportunities and challenges of EHWSNs are also explored in [7]. The authors of [5] studied different MAC protocols suitable for EHWSNs and presented analytical models for the slotted CSMA, identity polling, probabilistic polling and optimal polling MAC schemes. They concluded that probabilistic polling, designed for EHWSN nodes, provides scalability, high throughput and fairness and therefore is suitable to be used in EHWSN deployments.

In [9], authors compared three basic MAC schemes namely synchronization, preamble and beaconing for energy-harvesting systems and concluded that the beaconing paradigm can be tuned to consume less energy whereas the preamble paradigm can provide better performance for delay-sensitive applications if energy input is enough to allow the system to operate at higher duty cycles.

In synchronous MAC schemes, the nodes that share a common sleeping schedule, form virtual clusters, to wake up simultaneously. The benefit of this approach is that the only overhead is to keep the nodes finely synchronized by establishing, maintaining and distributing sleeping schedules between the nodes. This approach is unsuitable for EHWSNs, since it is not guaranteed that individual nodes will be available during the awake period as they may be in charging state. Examples of MAC protocols based on synchronization are S-MAC, DSMAC, T-MAC, RMAC and DW-MAC [2].

On the other hand asynchronous schemes can be classified into preamble or beaconing schemes. In preamble approaches, the senders transmit a preamble lasting as long as the sleeping period of the receiver before the actual data transmission. The receiver periodically wakes up and if it detects the preamble, it stays awake waiting for the data transmission. The overhead includes: the transmitter consumes energy transmitting the preamble for each packet transmission and, each node periodically needs to wake up and consume energy to listen for preambles. Examples includes B-MAC, X-MAC and WiseMAC [3]. In the beaconing approach, the communication is initiated by the receivers which periodically broadcast beacons that indicate their availability to receive data. Again the two sources of overhead are: the transmitter consumes energy listening to the channel for a beacon in each packet transmission and, each node periodically consumes energy to transmit the beacons. Some examples are RI-MAC, ODMAC and OriNoCo [1][2][3].

The authors of [10] suggested that the MAC protocols that are actually implemented and designed for energy efficiency, will operate efficiently enough (or could be configured to do so) for energy-harvesting systems.

III. PROPOSED SCHEME

The proposed scheme is illustrated in Fig. 2. The EHWSN nodes form a cluster (in virtual star topology) around a coordinator node (or cluster head). EHWSN nodes can only communicate to its own coordinator (when they have enough energy). Coordinators are special wireless nodes which have sufficient power available and can send data to base station directly or via other coordinators (multi-hop) in a typical converge-cast application. For MAC, we can utilize different scheme for within cluster communication and for cluster-cluster-base station transmission. Furthermore, different channel can be employed for each type of communication to avoid any possible interference.

At a higher level, there could be several of such independent sub-systems each with a base station. The cloud services provide the resources to efficiently process and present the information from these sub-systems. As shown in Fig. 2, base station is connected to two cloud data centers; primary and the secondary or backup datacenter to provide fault tolerant and redundant operation.

We assume that the coordinator has sufficient power available which is a fair assumption for various applications as we will discuss later. Coordinators and base station can therefore use standard contention-based protocols like IEE802.11 with CSMA/CA because of its simplicity and robustness as energy consumption is not an issue here.

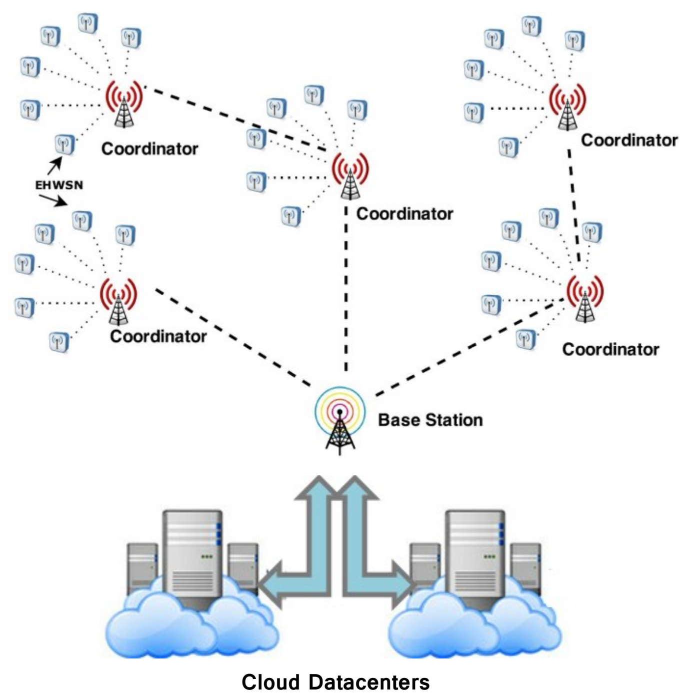

Fig. 2. Proposed deployment scheme for EHWSN system

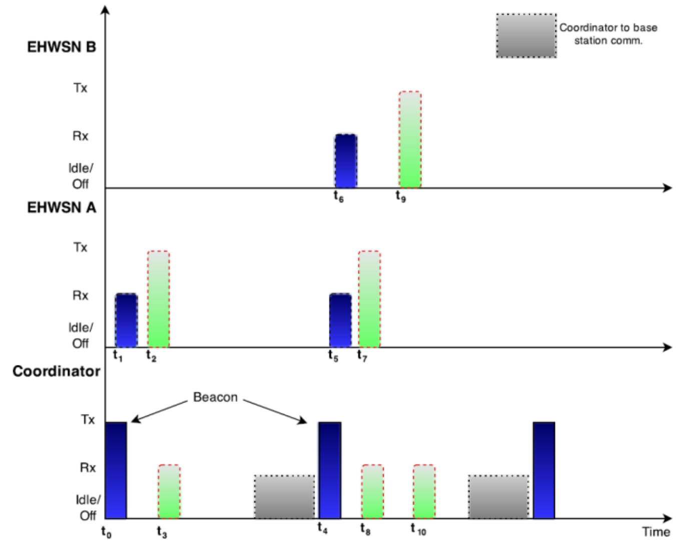

Fig. 3. Timing diagram of coordinator's beaconing and node's transmission

On the other hand, the communication between EHWSN and coordinator can adopt any suitable MAC protocol designed for EHWSN. One of the proposed protocols is probabilistic polling [4] which is based on beacon approach as discuss in the previous section. We will explain our proposed MAC scheme later in this section. This communication can happen in a different channel (frequency range) then the high-powered coordinator to base station transmission. As not all the EHWSN nodes can operate at any given time, the protocol should be able to self-configured and allow node to be added or removed from the neighborhood (cluster) as needed. EHWSN can be mobile as long as they can at least talk to one of the coordinators at any given time whereas coordinator assumed to be fixed. Also coordinator can either aggregate the data from its children EHWSN nodes or forward the data packets as is to the base station based on the application requirements.

A. MAC scheme

As we mentioned that EHWSN nodes formed a virtual cluster around a special coordinator node, which is similar to LEACH [18] or TEEN except that in our scheme, the coordinator is fixed and helps in forwarding data from its child EHWSN nodes. Fig. 3 illustrates the scheme on time scale and can be summarized as follows:

- The coordinator periodically (period = $T_c = t_4 - t_0$) broadcasts a beacon pulse with 10% duty cycle. The pulse contains the MAC address (48 or 64 bytes) of the radio, which is universally unique, and the number of registered nodes. After transmitting the beacon, it goes in the listening mode to receive messages from any EHWSN mode which have any packets to send.
- The neighboring EHWSN nodes (which have enough power to operate) periodically wake up (period = $T_n > T_b$) with a certain duty cycle to receive the beacon pulse from the coordinator (t_1 in Fig. 3). If the received coordinator's MAC address is different than the last communicated coordinator or the node has not communicated recently, then the node will send a

registration message containing its MAC address and the power status (t_2 in Fig. 3); otherwise node will go back to sleep or send the data packet if any. The nodes will use a random back-off time before transmission to avoid any collision with other sibling node's transmission during the same wakeup. The destination address will be set to the coordinator address so that any other neighboring nodes which are listening will ignore it.

- On receiving the node' registration message, the coordinator record the registration information and increment the number of registered nodes (t_3 in Fig. 3). If coordinator receives a data message then it will buffer it for future transmission to base station or for aggregation. The coordinator will acknowledge the received packet in both cases.
- The case of multiple child nodes transmitting in response to the same beacon pulse is also shown in Fig. 5 during the second beacon period. The contention is avoided by using a random back off time before transmission. The wining node will transmit first (node A transmits at t_7 in Fig. 3) and other nodes will wait for the channel (e.g. node B transmits at t_9 in Fig. 3).

As mentioned earlier that the cluster-cluster-base station communication happens on a different channel. The buffered data (either packets which were received earlier from child nodes or aggregated information from those packets) from each coordinator will be transmitted to base station either directly or via other coordinators (multi-hop) as shown in Fig. 2. This could be implemented in one of the two ways:

- A separate radio in each coordinator which communicates on a different channel (frequency) using robust IEE802.11 with CSMA/CA protocol can be utilized to transmit buffered data in parallel; or
- During each beacon cycle, after receiving data packets from nodes, the coordinator can switch temporarily to a preset channel to send all data from the buffer, if any. This was indicated in Fig. 3 also.

The EHWSN nodes are expected to report to coordinator (either via registration, data or heart-beat message) once a day (a configurable parameter), otherwise coordinator will assume that a child node is dead (or currently out of energy). The heart-beat could be a registration message sent at least once a day if there is no data message sent. This will allow auto deregistration of node and assist in node mobility from one coordinator to another. We called this scheme Coordinator Assisted MAC (CA-MAC).

B. *Suitable Applications*

Our scheme is suitable for following application scenarios:

- Industrial asset monitoring where EHWSN are deployed around a section of the facility or on mission-critical machinery. Nodes can be installed within the asset in hard-to-reach places. Once installed these nodes need to work for years and can utilize thermal energy generated by the machinery or other ambient sources for energy harvesting. This group of nodes is monitored by a coordinator (cluster head) which is powered from AC mains (which is usually available to power the facility or the machinery anyway). The data from other such clusters within the facility is gathered at a base station in the control room.
- Building/Office security/automation where EHWSN are deployed in a high traffic area for monitoring temperature, motion, etc. Again these sensors may be installed behind the wall or under the floor and need to work for long periods without any maintenance. They harvest energy from the ambient sources like light and pressure. The coordinator is powered from the readily available AC power. Such independent network/cluster can be installed on different parts of the building or on different floors and is managed by a base station in the security room.
- Internet of Things (IoT) where EHWSN are installed/embedded within the 'Thing' (object that need to be monitored). These objects with EHWSN formed a cluster around a coordinator (cluster head). These objects are mobile and can move around its neighborhood or move to another neighborhood (within the range of a different coordinator). The coordinators are installed at strategic fixed locations throughout the facility.

If this deployment is in a remote location then satellite technology can be utilized at base station to route data to a back office system [11]. For mission critical applications, where certain EHWSN node need to be available all the time, a battery backup system can be incorporated with EH system in the node's hardware. These nodes can switch to backup battery power when EH storage device doesn't have energy to power the node.

IV. Simulation and Experiment

Simulation has always been very popular among network-related research. Several simulators have been developed to implement and study algorithms for wireless networks. Some are general purpose while others are design for specific purpose and vary in features and the level of complexity. They support certain hardware and communication layers assumptions, and provide set of tools for deployment scenarios, modeling, analysis, and visualization. Classical simulation tools include NS-2/3, OPNET, OMNeT++, J-Sim, and TOSSIM [12][13][14][15].

After some research, we concluded that Python-based tools completely fulfill our requirements. We decided to use Pymote, which is a high level Python library specifically designed for wireless networks to perform event based simulation of distributed algorithms [16]. The user can implement their ideas in Python; which has become popular in academia and industry. The library is developed without much abstraction and therefore can be used or extended using Python's highly expressive native syntax. The library particularly focuses on fast and accurate implementation of ideas at algorithm level using formally defined distributed computing environment.

A. Simulation setup

We only need to simulate the communication performance of the cluster formed by a coordinator and its children EHWSN nodes. The coordinator is placed in middle of n randomly deployed EHWSN nodes over a 600 m by 600 m area. We simulate the charging rate by message rate (higher message rate implies higher charging rate). We consider beacon, registration and data packet sizes of 100 bytes while the acknowledgment packet size of 15 bytes. Some other parameters are shown in Table I.

Fig. 4 shows a simple topology generated for simulation using the Pymote. The center node (#1) acts as the coordinator for the EHWSN nodes (numbered 2 to 51). First we evaluate the performance by changing the number of nodes (from 5 to 100) to see if our proposed scheme is scalable. In this simulation, we kept the fixed data rate of one message every 5 seconds. Secondly, we vary the data rate from one message every 5 seconds to a message every second and keep the number of nodes fixed at 10.

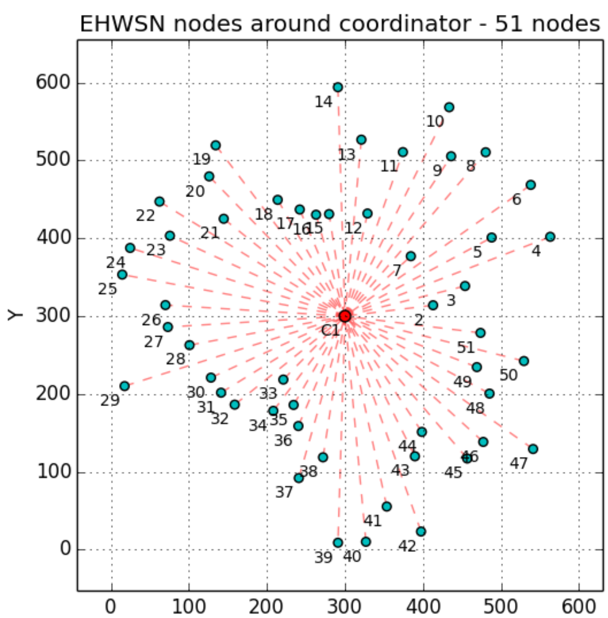

Fig. 4. 50 EWSN nodes around a corrdinator

TABLE I. SIMULATION PARAMETERS

Parameter	Name	Value
T_b	Beacon period	1 sec
Δ_b	Beacon duty-cycle	10%
n	No. of nodes	5 - 100
S_d	Data packet size	100 bytes
S_a	Ack packet size	15 bytes

B. Experiment

We also utilize a WSN professional kit provided by MEMSIC to build a proof of concept system. The kit provides an end-to-end platform for the creation of wireless sensor networks [17]. A windows interface, *MoteView,* is provided as a client tier between the user and the deployed network of wireless sensors. This tool not only simplifies deployment and monitoring but also provides connection to a database, have option to analyze and plot sensor readings. It also provides node health statistics in terms of transmission quality, percentage of drop packets, number of retries, etc.

Finally, we performed a simple proof of concept experiment with 2 clusters and a base station (connected to PC). We used 2 nodes emulating EHWSN nodes around a coordinator (i.e. 3 nodes per cluster). The EHWSN nodes are placed such a way that base station can't see them and they can only communicate to one of the coordinator. The packets from EHWSN nodes are forwarded through coordinators to the base station in a simple converge-cast scenario as shown in Fig. 5.

We run the experiment for about one hour (62 minutes) with message rate of 5 seconds at KFUPM stadium. We record all the packets arriving from coordinators at the base station. Fig. 6 shows the approximate location of the sensors for one of the clusters.

$$\text{Number of expected messages per node/min} = 60/5 = 12$$
$$\text{Total expected messages per node} = 12 \times 62 = 744$$

Table II summarizes the results. Due to retries, we received about 2.3% more packets than expected.

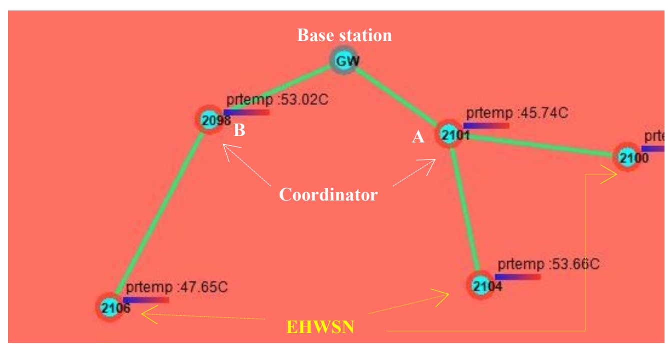

Fig. 5. Two coordinators, a base station and few EHWSN nodes

TABLE II. EXPERIMENT RESULTS

Node	Number of packets received	extra packets
Coordinator A	762	18
Node A-1	764	20
Node A-2	761	17
Coordinator B	761	17
Node B-1	766	22

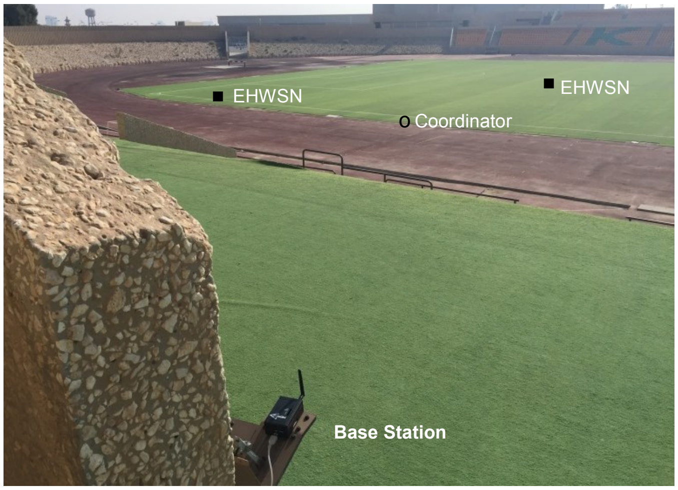

Fig. 6. Experiment site – Base station with a coordinator and couple of EHWSN nodes

V. CONCLUSION

In this work, we studied different MAC protocols that can be utilized in WSN system based on energy harvesting. We analyzed analytical models and performance metrics for some proposed MAC schemes. Most existing MAC protocols are inefficient or non-optimal for EHWSN since the charging times are unpredictable. We then proposed our deployment and MAC scheme for efficient utilization of EHWSN in certain applications. We validated our scheme using simulations developed on the interactive Python based Pymote framework. We also performed a proof of concept experiment using a PC-based WSN development kit.

As future work, we are further extending the Pymote framework to include propagation, energy consumption, mobility and other models. The extended framework will also provide interactive graphing and logging facilities for improved performance evaluation. Also, we plan to design, deploy, and test scalable and elastic cloud-based applications and services with the capabilities to process and analysis the massive data to be generated by the billions of sensors and IoT devices.

ACKNOWLEDGMENT

We would like to acknowledge the support provided by College of Computer Science and Engineering (CCSE) and the Deanship of Research (DSR) at King Fahd University of Petroleum and Minerals (KFUPM).

REFERENCES

[1] P. Suriyachai, U. Roedig, and A. Scott., "A survey of MAC protocols for mission-critical applications in wireless sensor networks." *IEEE Communications Surveys & Tutorials,* vol. 14, no. 2 (2012): 240-264.Fukumoto, M. and Tonomura, Y. Body coupled Fin-geRing. In Proc of CHI 1997, 147-154.

[2] P. Huang, L. Xiao, S. Soltani, M. Mutka, and N. Xi., "The evolution of MAC protocols in wireless sensor networks: A survey." *IEEE Communications Surveys & Tutorials*, vol.15, no.1, pp.101-120, First Quarter 2013.

[3] A. Bachir, M. Dohler, T. Watteyne, and K. K. Leung. "MAC essentials for wireless sensor networks." *IEEE Communications Surveys & Tutorials*, vol. 12, no. 2 (2010): 222-248.

[4] Steve Grady and Jeff Mullin Cymbet, "The Design Secrets for Commercially Successful EH-Powered Wireless Sensors." Energy Harvesting for Powering WSN Symposia, 2014

[5] Zhi Ang Eu, Hwee-Pink Tan, Winston K.G. Seah, "Design and performance analysis of MAC schemes for Wireless Sensor Networks Powered by Ambient Energy Harvesting," *Ad Hoc Networks*, Volume 9, Issue 3, May 2011, Pages 300-323, ISSN 1570-8705, http://dx.doi.org/10.1016/j.adhoc.2010.07.014.

[6] Nintanavongsa, P.; Naderi, M.Y.; Chowdhury, K.R., "Medium access control protocol design for sensors powered by wireless energy transfer," *INFOCOM, 2013 Proceedings IEEE* , vol., no., pp.150,154, 14-19 April 2013. doi: 10.1109/INFCOM.2013.6566753

[7] Basagni, Stefano, et al. "Wireless sensor networks with energy harvesting," *Mobile Ad Hoc Networking*: The Cutting Edge Directions (2013): 701-736.

[8] Seah, Winston KG, Y. K. Tan, and Alvin TS Chan, "Research in energy harvesting wireless sensor networks and the challenges ahead," Springer Berlin Heidelberg, 2013.

[9] Fafoutis, X.; Dragoni, N., "Analytical comparison of MAC schemes for Energy Harvesting — Wireless Sensor Networks," *Networked Sensing Systems (INSS)*, 2012 Ninth International Conference on , vol., no., pp.1,6, 11-14 June 2012. doi: 10.1109/INSS.2012.6240580

[10] Tan, W.M.; Jarvis, S.A., "Self-determination of maximum supportable receiver wakeup intervals in Energy Harvesting WSN nodes," Wireless Days (WD), 2013 IFIP , vol., no., pp.1,7, 13-15 Nov. 2013. doi: 10.1109/WD.2013.6686440

[11] F. Shahzad, "Satellite monitoring of wireless sensor networks (WSNs)," *Procedia Computer Science*, vol. 21, no. 0, pp. 479 – 484, 2013.

[12] Sobeih, A., Chen, W., Hou, J., Kung, L., "J-sim: A simulation environment for wireless sensor networks," in Proceedings of the 38th annual Symposium on Simulation. IEEE Computer Society Washington, DC, USA, 2005, pp. 175–187.

[13] Ali, Q., Abdulmaojod, A., Ahmed, H., "Simulation & Performance Study of Wireless Sensor Network (WSN) Using MATLAB," *IJEEE Journal*, 2010.

[14] Qutaiba I. Ali (2012). Simulation Framework of Wireless Sensor Network (WSN) Using MATLAB/SIMULINK Software, MATLAB - A Fundamental Tool for Scientific Computing and Engineering Applications - Volume 2, Prof. Vasilios Katsikis (Ed.), ISBN: 978-953-51-0751-4, InTech, DOI: 10.5772/46467.

[15] Tselishchev, Y.; Boulis, A.; Libman, L., "Experiences and Lessons from Implementing a Wireless Sensor Network MAC Protocol in the Castalia Simulator," *Wireless Communications and Networking Conference (WCNC)*, 2010 IEEE , vol., no., pp.1,6, 18-21 April 2010. doi: 10.1109/WCNC.2010.5506096

[16] Damir Arbula and Kristijan Lenac, "Pymote: High Level Python Library for Event-Based Simulation and Evaluation of Distributed Algorithms," *International Journal of Distributed Sensor Networks*, vol. 2013, Article ID 797354, 12 pages, 2013. doi:10.1155/2013/797354

[17] Memsic, "MoteView Users Manual", Revision B, May 2012, PN: 7430-0008-05

[18] Heinzelman, W.R.; Chandrakasan, A.; Balakrishnan, H., "Energy-efficient communication protocol for wireless microsensor networks," *System Sciences*, 2000. Proceedings of the 33rd Annual Hawaii International Conference on , vol., no., pp.10 pp. vol.2,, 4-7 Jan. 2000

[19] F. Shahzad, "State-of-the-art Survey on Cloud Computing Security Challenges, Approaches and Solutions," *Procedia Computer Science*., vol. 37, pp. 357–362, 2014.

[20] Salah, K., and Boutaba, R. "Estimating Service Response Time for Elastic Cloud Applications," Proceedings of 1st IEEE International Conference on Cloud Networking (IEEE CloudNet 2012). Paris, France, November 28-30, 2012

[21] Salah, K., Alcaraz Calero, J., Zeadally, S., Almulla, S., and Alzaabi, M., "Using Cloud Computing to Implement a Security Overlay Network," *IEEE Security and Privacy*, Vol. 11, No. 1, January 2013, pp. 44-53.