Implementation of an Effective Power-aware Sensor Node Model for Wireless Sensor Networks

Rym Che´our, Mohamed Wassim Jmal, and Mohamed Abid,

National School of Engineers of Sfax, University of Sfax, Computer and Embedded System Laboratory, Sfax, Tunisia

Abstract

The Wireless Sensor Networks are facing an important rise which is reflected by their omnipresence in var- ious fields of application. Indeed, they require new features, better performances, and numerous challenges like the power constraints, the need to adapt to different conditions and the limited computation capabilities. So, minimizing the overall energy consumption meanwhile avoiding the deadline violations is crucial to achieve high performances and to enhance the reliability of the network. In this paper, we will provide a model of energy management for WSN that will be simulated and validated by the STORM Simulator. Unlike traditional WSN energy management systems, our model reduces the energy consumption through a dual approach: a global and dynamic approach using the analysis of the behavior of the network and a local strategy applied at the node level. We have proposed an implementation of the Global Earliest Deadline First (G-EDF) scheduling algorithm, augmented with energy optimization techniques to yield extensive lifetime for every nodes battery. An interplay between DPM (Dynamic Power Management) and DVFS (Dynamic Voltage and Frequency Scaling) that are appropriate for the WSN has been also adopted. A simulation set-up as well as some results are given to illustrate the gain obtained. These different strategies are validated through different simulations.

Keywords

Wireless Sensor Networks, Modeling, Simulation, Energy, Power, DVFS, DPM, EDF, Scheduling, QoS

I. Introduction

Wireless Sensor Networks (WSNs) are made of a set of cooperating sensors nodes spatially distributed so that they can monitor the environment where they are deployed. Sensor networks provide endless opportunities as well as numerous challenges. WSNs are massively deployed in many fields such as the monitoring and preservation of the environment, the industrial manufacturing, the automation in the transport [1], the health sectors [2], the agriculture [3], the telematics and the logistics which has led to a higher consumption of energy [4]. WSNs are made up of low-cost, low-power, multifunc- tional, and small sensor nodes interconnected to accomplish a common task [5]. Most of the nodes are powered by batteries that store a limited amount of energy [6]. That’s why, the energy conservation should be more rigorously considered; as it represents actually an overriding constraint for the design and operation of the network. Reducing the energy in WSN is scientifically investigated in several research studies [7]. The computation power is the sum of both dynamic and static power. In the CMOS technology the dynamic power exceeds the static power [8]. So the power dissipation is equal to P = CV 2 f where C is the total capacitance, V is the supply voltage, f is the clock frequency. As a result, DVFS can lead to cubic improvement in power dissipation. Several experiments of low-power circuit design [9], energy management [10] [11] and real-time scheduling [12] are conducted. However, despite those energy-efficient architectures, few tools are available to test them [13]. Besides, most of the studies targeted the radio transmission consumption and few of them addressed decreasing the power of the computing unity.

This paper proposes a generic power-aware model that exploits power management techniques and scheduling to reduce energy consumption, to increase the operational lifetime of the network and to minimize the delays for Wireless Sensor Networks. We have chosen to apply a G-EDF ”Global Earliest Deadline First” scheduler that will help to reduce the consumption at the network level when the combined DPM/DVFS strategies will yield a significant energy saving at the local node level. More specifically, at the local level, we will apply the DPM technique during the idle interval, and when the node is active the strategy DVFS will be used. At the global level, the G-EDF intend to avoid missing deadlines and to provide schedulability guarantees. The scheduler will check whether the task execution should be carried out or not, depending on the available energy and the current time. We have validated the software infrastructure and algorithms with The STORM simulator. Another major contribution of our work is to enhance and extend the features of the simulator by a module that addresses energy. The paper is organized in the following manner. Section 2 states previous work in energy and power management. It includes an overview and a comparison of the existing techniques. In the next section, we will outline the”Sensor Energy Model” developed and describe the advantages of such a model. Section 4, will introduce the setup used in the experiments and the different parameters considered. Then, we give a detailed results description.

Manuscript received May 5, 2016
Manuscript revised May 20, 2016

The impact of this model is treated in the discussion section. Finally, a conclusion and future work are given.

II. Related Works on Energy Optimization in WSNS

Power dissipation and energy consumption play a key role in high performance computing and in the embedded systems [14]. The management of energy has even become synonymous with improved performance for resource-constrained systems [8]. The optimization of energy in WSN is widely investigated as high reactivity applications usually lead to higher energy consumption that will drain rapidly the battery lifetime and impede the operation of the whole network [15]. Overviews and surveys have toured the main techniques of energy like [9], [16] and [17]. They focused in the Dynamic Power Management like [18] [19] [20] [21] [22] or the DPM with Scheduled Switching [23]. The use of Dynamic Voltage and Frequency Scaling has been discussed in [24] and an interplay of both of DPM and DVFS has been considered in. Also, scheduling has been considered in [12] [25] and [26].

[27] Proposes to reduce the power consumption of the processing unit through”undervolting” by powering the electric circuits below the specified voltage levels. However, this technique depends on the environment and has an operating limit. This threshold is unpredictable because for the same platform it gives different results and is not portable to all platforms. Besides, the risk of error and propagation delays grow with higher clock rates.

[9] Is an overview of the energy optimization techniques at each level of the node (architecture, capture unit, MAC layers). They focus on application changes over time and environ- mental conditions to provide a new MDP(Markov Decision Process). This technique relies on tunable parameters other than voltage and frequency which are sensing frequency and transmission power to retrieve the new operating state of the node in accordance with the changing environment stimuli.

In [28], authors present a theoretical approach to estimate the yield and the ideal usage of DVS on a wireless sensor node. They derived modDVS combining both DVS and DPM. However, non-compliance with time constraints may affect the efficacy of the network. As a result, not relying on scheduling can lead to significant loss of performance criteria.

In [29] authors propose a WSN energy model based on reducing the redundancy of working sensor nodes by defining minimal number of active nodes in a sensing area through the deployment of scheduling. Nodes switch between working and sleeping states based on remaining power and guaranteed coverage throughout the network. This method shows some disadvantages in terms of complexity and optimality.

The related works presented are valuable because they reveal the potential of additional power saving by using power saving techniques. However, they have a few shortcomings such as not exploring the possibility of reducing the system energy by scheduling. Not taking into account the set of tasks settings considered and how they react to the use of the latest techniques mentioned, especially the impact of these technologies on the deadlines may affect the overall performance of the network. In the next section, we will present a DPM and DVFS techniques that can potentially minimize the power consumption when scheduling a set of tasks based on G-EDF.

III. The Sensor Node Model

This paper highlights an advanced model for WSN energy saving. Rather than designing a new power management policies, our contribution is to gather several well-known techniques which have proved their effectiveness separately and apply the best-performing policy at run-time for any given workload and at any point in time. Those techniques are an interplay of the DPM and DVFS with a global EDF scheduler. The figure 1, shows the different internal interactions between the application layer and the hardware layer of every sensor through the intermediate layer that manages the resources. In our case, it provides the facilities to manage the CPU activity. The sensor node model (SNM) defines the scheduler and the energy management technique through a software command. As shown, in figure 1, the model is made of four basic components. The first phase is to assign tasks to the nodes through an XML file. The inputs of the model are the data related to time settings, number of tasks, number of nodes, etc. In the next step, tasks are scheduled with G-EDF. The third phase aims to adapt the choice of the energy strategy according to the needs for the application. The energy management components try to control the resource usage by selecting DPM or DVFS. Finally, the power evaluation component aims to give a feedback about the performance criteria such as energy saving, respecting time constraints. The SNM performance evaluation component will be done and validated by STORM. When we apply the DVFS strategy we reduce the frequency and as a result we prolong the task execution time. At the same time, the idle intervals are shortened but remain less than the time beyond which we switch to low-power mode of the DPM technique.

The aim of our model is also to integrate both DPM and DVFS with the best trade-off to reduce the total energy consumption. Besides, the interplay of DPM/DVFS with the global EDF is not exploited yet in

WSN, so we implement the SNM to perform better energy gain.

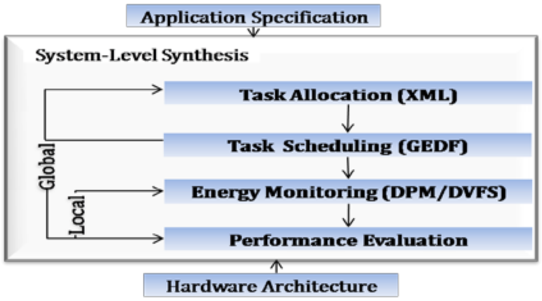


Fig. 1: The Sensor Node Model

IV. Implementation of the SNM with Storm

To simulate and evaluate the considered energy model in the simulation environment STORM, we have selected a Mica2 WSN node [30]. The STORM" Simulation Tool for Real time Multiprocessor scheduling" simulator is intended to analyze the behavior and to evaluate the performance of the policies of scheduling while taking into account the algorithms of energy management. So, it can describe both the considered scheduling algorithm, the application part (the sets of tasks) and a large grain description of the hardware architecture represented by the CPU cores. The results of the simulation are a set of diagrams. To characterize the power architecture processor cores, we introduce a power model. Thus, the objective is to transmit at every event of scheduling, the power dissipated by each core, according to the activity of cores obtained from scheduling. In case of a processor with DVFS and DPM capabilities, the corresponding entity owns additional properties such as power consumption mode, operating voltage, and frequency etc.

This simulator requires a set of information and settings so that it will be representative of a real system as shown in both table I and II. We must provide the task characteristics, the description of the architecture and define the scheduling algorithm used. The set of the tasks represents the software architecture. Each node is represented separately. To map system timing requirements, the table I gives a glimpse about the run time parameters of a task important to estimate the power consumption. We consider a set of n periodic and preemptive task that can be interrupted during the execution and resumed later from the same point (n=5 in table I). For each task we define the following attributes: its period, its deadline, its activation date, its WCET and BCET. The WCET stands for the “Worst Case Execution Time” and the BCET stands for “Best Case Execution Time”. The scheduler sorts the list of ready-tasks according to the deadlines, it gives orders to the kernel to carry out or preempt the ready tasks (Running On (), preempt ()). The task with the earliest scheduling deadline is selected for execution. We assume also that the deadline of each task is equal to the corresponding period.

TABLE I: System timing requirements

Parameters	T1	T2	T3	T4	T5
Period	80	100	120	150	200
WCET	50	55	60	60	85
BCET	30	35	40	30	45
Deadline	80	100	120	150	200

As we have shown earlier the maximum response time of a process cannot exceed the value of the deadline. We applied several modification during the simulations to the number of tasks, the number and the different characteristics of processors as shown in table II. We consider that the number of CPU represents the number of sensor nodes used. Our work is based on a set of assumptions that contribute to retrieve better results. First, all the sensors are the same. They are homogeneous provided the same power and the same scope. Therefore, we can consider a mote as a set of tasks and time constraints we absolutely must respect. When selecting the CPU speed, we also need to ensure that during this interval the energy consumed is at a minimum and that all tasks are completed prior to/or at the time of deadline. As in classical real-time scheduling problem, the relative deadline is assumed to be equal to period. Task must complete its execution before the next release. The consumption can be estimated if we know the time spent at each state. During our work, we are not going to consider renewable energy and we will use non-rechargeable battery. These assumptions allow the power manager to decide when switching to a lower power mode.

TABLE II: Experimental Parameters

Parameter	Value
Number of tasks	5, 10, and 20
Number of CPU	2,5
CPU type	ATMega128L

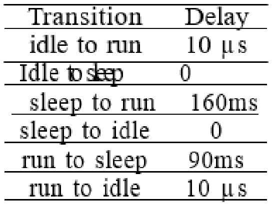
A. EDF-DPM Experiment Setup

We assign for each task one of these four states: Running, Ready for execution, Waiting or Unexisting. When we move the running task to another state and to give control of the CPU to a new task, a context switch should be performed. Each time a task enters the ”Ready” state (its methods onActivate() and onUnBlock()), it has to be added to the end of this list by calling its addLast() method. Each time a task leaves the ”Running” state (its methods onTerminated() and onBlock()), it has to be removed from this list by calling its remove() method. Besides, the first activation (onActivate) and the following activations (onUnblock) will add the corresponding task to the ready queue, whereas the events of termination of jobs (onBlock) or of task (onTerminate) correspond to a rejection of the corresponding task from the ready tasks queue. When a state changes from ”Ready” to ”Running” for such a task, it simply requires calling the onRunning method of its equivalent object. The ATmega128L is a low- power microcontroller which consumes P_{run} = 8mW in the RUN mode or P_{idle} = 98.0 µW in the IDLE mode, $P_{slee\ p}$ = 15 µW in the SLEEP mode. Every node has at its disposal a E_{bat} = 21 battery. Many interruptions take place at t= iT when i∈ 0, 1, 2, ... with the aim of informing the CPU that new tasks arrived. The treatment of each task requires a given time The treatment of each task requires a given time. To switch from the RUN mode to the IDLE mode lasts we require a time t equal to t = 10µ s. However, when switching from the RUN mode to the SLEEP mode we need t = 90µ s. During the transitions from one power mode to another the CPU is supposed to be idle and as a corollary, a linear and continuous variation must take place at the power level.

B. EDF-DPM Experiment Results

The figure 2 illustrates the experiment results. The time is represented in units of 50 seconds (x-axis). We have noticed that it is not desirable to keep nodes inactive for too long, because it can impact the network Quality-of-Service. When we had applied the DPM policy, major improvement had been seen such as the elimination of both dynamic and static power dissipation. Besides, the transition delays had been set up to avoid the potential impact of missing the execution of any interesting task. Moreover, the transition between the different power configurations showed an extra energy and latency costs. However, we have noticed that the levels of energy consumption of the different modes, the costs of transition between modes but also the time spent by the CPU in each mode had a significant impact on the total consumption of energy of a sensor node.

TABLE III: Transition states and latency supported by AT- mega128L



Transition	Delay
idle to run	10 µs
Idle to sleep	0
sleep to run	160ms
sleep to idle	0
run to sleep	90ms
run to idle	10 µs

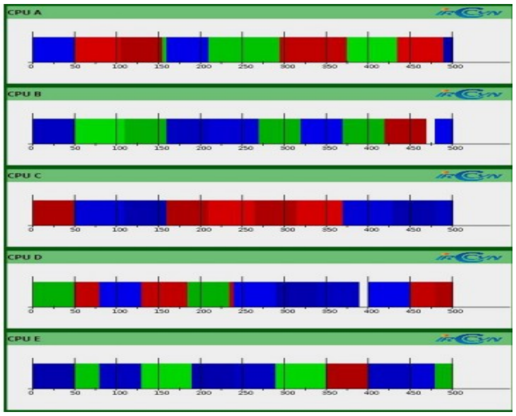
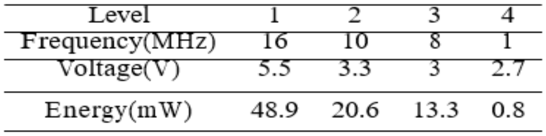


Fig. 2: EDF-DPM Experiment Results

C. EDF-DVFS Experiment Setup

Our aim is decreasing the power consumption of the CPUs through picking out the smallest available frequency able to finish a task in a given time frame. The overhead of changing DVFS settings are assumed to be negligible. To address this issue, only 3 values of supply voltage and corresponding operating frequency are selected according to the type of received events. We define n nodes at fixed frequencies fi = 16MH z; 10MHz; 8MHz; 1MHz voltages V_i = 5.5V ; 3.3V ; 3V ; 2.7V as shown in table IV. Choosing a fre- quency lower than $fmin$ = 1MH z would necessarily lead to overruns of deadlines. The maximum frequency available is f_{max} = 16MH z , when the minimum frequency f_{min}) is equal to 1 MHz. The transition among the different frequencies generates negligible overheads. WSNs use ISM (industrial, sci- entific and medical) radio bands for applications at 443MHz, 886/916MHz and 2.4GHz. To retrieve more energy saving, the DPM strategy will be used so that the processor is put off or put in the sleep mode such as 0 kJ will be deployed.

TABLE IV: Voltage/frequency couples supported by AT- mega128



Level	1	2	3	4
Frequency(MHz)	16	10	8	1
Voltage(V)	5.5	3.3	3	2.7
Energy(mW)	48.9	20.6	13.3	0.8

To simplify the discussion, we consider that the voltage and the frequency are always adjusted together. When switching the voltage, we assume that the overhead associated with the scheduling of tension is negligible.

D. EDF-DVFS Experiment Results

A gain of energy was observed every time we have changed the frequency and mainly when the sleep mode is not applica- ble. Also, we have noticed that applying the scheduling when taking into account the current energy level and the priorities of the tasks (done through the EDF scheduler) had enabled graceful degradation. The results are scaled between 0.0 and 1.0 respecting the values that are not optimized. The proposed scheme is generic in the sense that it can work with other global scheduling algorithms as well rather than the EDF that is used. Our assessments indicate a gain is obtained using a DVFS algorithm in order to increase the autonomy of the mote. When the CPU frequency is lowered down from fmax to 8Mhz (2.7V) as shown in figure 3 with the DVFS strategy, indicates that the CPU energy consumption was reduced by about 77%. As a result while applying the DVFS we can obtain a gain in the overall power consumption ranging from 50% to 80%. The previous figures illustrated that changing frequency causes a significant delay without however exceeding the deadlines defined of each task. This latency, although small, is the result of scaling both the frequency and the voltage and is closely related to the application and the choice of the scheduler.

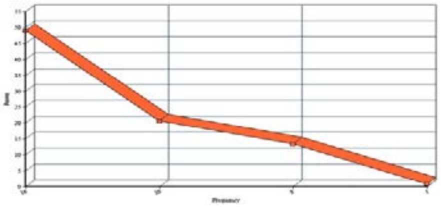


Fig. 3: Power consumption gain with DVFS

V. Results Discussion

The simulation is so prevalent and suggests a fundamental preference over hardware prototyping for significant cost savings, compressed validation time, greater system scale and improved component analysis. The energy consumption in several of the CPU states is computed with the help of the current consumption available in the manufacturer data sheet and the time spent in these states during the simulation time. Its done through the use of the command”calcpower”. During this simulation phase, we have observed that STORM provides performance criteria such as deadlines, consumption, etc very close to the values obtained from real platform and applied in the same experimental conditions seen in the data-sheet. In the Task Scheduling phase of the SNM, the G-EDF algorithm selects the task with the earliest scheduling deadline as the one to be executed next. Thanks to this feature, we have noticed that it performs well when scheduling the considered periodic task sets because it minimized the value of context switches, preemptions, and at some level the response time was also minimized. We have doubled and tripled the number of tasks. Results depict that whenever we rise the number of tasks the energy consumption rises in parallel. Besides, we note that when increasing the number of tasks (Task Allocation) and keeping the same number of node that the rate of CPU utilization increases jointly equaling 100%. This is due to the fact that excessive request of the CPU had caused an overload. During the simulation, we have changed the number of CPUs (number of nodes) to study its impact on the whole system. As a result, we have noted that the more the number of processors increases the more the processing time and the makespan(i.e. the date of completion of the last task scheduled) are reduced. However, it creates an additional energy costs by rising the slack time. The simulation demonstrate also how scaling further the frequencies can lead to more energy saving at the global level. The decision toward the length of an upcoming idle period in the DPM algorithm is not trivial in the Energy Management phase of the SNM. Here, we have chosen to put the task in the idle state when t>300ms. If the idle period is so short that the powering-up costs are greater than the energy saved in the sleep state, we cannot have the amount of energy saving expected. Besides, if this value is too long to power-down may not achieve the best-possible energy reductions either. During the simulation, we have changed the number of CPUs (number of nodes) to study its impact on the system. As a result, we have noted that the more the number of processors increases the more processing time and the makespan(i.e. the date of completion of the last task sequenced) are reduced. However, it creates an additional energy costs by increasing the slack time. The simulator has the potential to improve its performances and mainly the accuracy criteria. As the measured values are close to the real ones obtained from the data-sheet (almost 95%), to this end, we can adjust those results by imposing the optimal power mode to the unused devices.

VI. Conclusion

Power source evolution in batteries, as well as external source of energy such as solar or vibration, are expanding. However, they didn’t reach maturity yet. So, it’s necessary to test new methods and protocols in order to improve the efficiency of WSN. The contributions of our work include modeling of a “Sensor Node Model” for WSN. This model has been implemented in STORM including all modes of operation and the transitions between the different modes. We have also tried to include a global EDF scheduler, to improve the performances of WSN and to increase their lifetime. The DPM correlated to EDF traced the history of task executions to predict the idle periods and to turn hardware to a low- power state to reduce the energy consumption. The DVFS was applied also to tune a processor clock speed and its corresponding voltage according to requirements such as the workload (expected or actual) or the battery charge. This model has succeeded to offer a reduction in the total energy consumption ranging from 50% to 80%. For future work, we will explore in greater detail the impact of implementing our work on a real conditions.

REFERENCES

[1] H. Kopka and P. W. Daly, A Guide to LATEX, 3rd ed. Harlow, England: Addison-Wesley, 1999.

[2] H.-C. Lee and H.-B. Huang, "A low-cost and noninvasive system for the measurement and detection of faulty streetlights," IEEE Trans. Instrum. Meas., vol. 64, no. 4, pp. 1019–1031, 2015.

[3] Y. Lim and J. Park, "Networking strategies for structural health mon- itoring in wireless sensor networks," International Journal of Energy, Information and Communication, vol. 6, no. 3, pp. 11–18, 2015.

[4] S. Khriji, D. El Houssaini, M. W. Jmal, C. Viehweger, M. Abid, and O. Kanoun, "Precision irrigation based on wireless sensor network," Science, Measurement & Technology, IET, vol. 8, no. 3, pp. 98–106, 2014.

[5] F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," Computer networks, vol. 38, no. 4, pp. 393–422, 2002.

[6] F. Akyildiz and I. H. Kasimoglu, "Wireless sensor and actor networks: research challenges," Ad hoc networks, vol. 2, no. 4, pp. 351–367, 2004.

[7] Keerthivenkateshkumar and E. Baburaj, "A confined strategy to analyze the lifetime of sensor nodes for multihop wsn," International Journal of Computer Science and Network Security (IJCSNS), vol. 16, no. 2, p. 1, 2016.

[8] J. A. Khan, H. K. Qureshi, and A. Iqbal, "Energy management in wire- less sensor networks: A survey," Computers & Electrical Engineering, vol. 41, pp. 159–176, 2015.

[9] T. Yuki and S. Rajopadhye, "Folklore confirmed: Compiling for speed= compiling for energy," in Languages and Compilers for Parallel Computing, pp. 169–184, Springer, 2014.

[10] Munir and A. Gordon-Ross, "Optimization approaches in wireless sensor networks," Sustainable Wireless Sensor Networks, pp. 313–338, 2010.

[11] T. Rault, A. Bouabdallah, and Y. Challal, "Energy efficiency in wireless sensor networks: a top-down survey," Computer Networks, vol. 67, pp. 104–122, 2014.

[12] J. A. Khan, H. K. Qureshi, and A. Iqbal, "Energy management in wire- less sensor networks: A survey," Computers & Electrical Engineering, 2014.

[13] D. Harkut, M. Ali, and P. Lohiya, "Scheduling task of wireless sensor network using earliest deadline first algorithm," International Journal of Scientific Research in Computer Science and Engineering, vol. 2, no. 02, pp. 1–6, 2014.

[14] R. Cheour, M. W. Jmal, A. Lay-Ekuakille, F. Derbel, O. Kanoun, and M. Abid, "Choice of efficient simulator tool for wireless sensor networks," in International Workshop on Measurements and Networking Proceedings (MN), IEEE, 2013.

[15] Rountree, D. K. Lownenthal, B. R. de Supinski, M. Schulz, V. W. Freeh, and T. Bletsch, "Adagio: Making dvs practical for complex hpc applications," in Proceedings of the 23rd International Conference on Supercomputing, ICS '09, (New York, NY, USA), pp. 460–469, ACM, 2009.

[16] E. Fadel, V. Gungor, L. Nassef, N. Akkari, M. A. Maik, S. Almasri, and I. F. Akyildiz, "A survey on wireless sensor networks for smart grid," Computer Communications, vol. 71, pp. 22–33, 2015.

[17] X. Jiang, J. Taneja, J. Ortiz, A. Tavakoli, P. Dutta, J. Jeong, D. Culler, P. Levis, and S. Shenker, "An architecture for energy management in wireless sensor networks," ACM SIGBED Review, vol. 4, no. 3, pp. 31– 36, 2007.

[18] Boudhir, M. Bouhorma, and M. Benahmed, "Energy optimization approaches in wireless sensor networks: A survey," International Jour- nal of Networks and Systems, vol. 1, no. 1, pp. 43–51, 2012.

[19] W. Dargie, "Dynamic power management in wireless sensor networks: State-of-the-art," IEEE Sensors Journal, vol. 12, no. 5, pp. 1518–1528, 2012.

[20] Paul, "Dynamic power management for ubiquitous network devices," Advanced Science Letters, vol. 19, no. 7, pp. 2046–2049, 2013.

[21] Bogliolo, E. Lattanzi, and V. Freschi, "Idleness as a resource in energy-neutral wsns," in Proceedings of the 1st International Workshop on Energy Neutral Sensing Systems, p. 12, ACM, 2013.

[22] Lin, Y.-X. He, and N. Xiong, "An energy-efficient dynamic power management in wireless sensor networks," in Parallel and Distributed Computing, 2006. ISPDC'06. The Fifth International Symposium on, pp. 148–154, IEEE, 2006.

[23] Sinha and A. Chandrakasan, "Dynamic power management in wireless sensor networks," Design & Test of Computers, IEEE, vol. 18, no. 2, pp. 62–74, 2001.

[24] P. S. Sausen, J. R. de Brito Sousa, M. A. Spohn, A. Perkusich, and A. M. N. Lima, "Dynamic power management with scheduled switching modes," Computer Communications, vol. 31, no. 15, pp. 3625–3637, 2008.

[25] U. Kulau, F. Busching, and L. Wolf, "A node's life: Increasing wsn lifetime by dynamic voltage scaling," in IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS), 2013, pp. 241–248, IEEE, 2013.

[26] M. Dgharkut and P. Lohiya, "Real-time scheduling algorithms for wire- less sensor network," Circuits and Systems: An International Journal (CSIJ), 2014.

[27] J. Wu and S. Yang, "Energy-efficient node scheduling models in sensor networks with adjustable ranges," International Journal of Foundations of Computer Science, vol. 16, no. 01, pp. 3–17, 2005.

[28] U. Kulau, F. Busching, and L. Wolf, "Undervolting in wsns : A feasibility analysis," in IEEE World Forum on Internet of Things (WF- IoT), 2014, pp. 553–558, IEEE, 2014.

[29] V. T. Hoang, N. Julien, and P. Berruet, "Increasing the autonomy of wireless sensor node by effective use of both dpm and dvfs methods," in Faible Tension Faible Consommation (FTFC), 2013 IEEE, pp. 1–4, IEEE, 2013.

[30] N.-T. Le and Y. M. Jang, "Energy-efficient coverage guarantees schedul- ing and routing strategy for wireless sensor networks," International Journal of Distributed Sensor Networks, vol. 501, p. 612383, 2015.

[31] IRCCYN, "Presentation de storm." http://storm.rts- software.org/doku.php?id=start=, 2015.