Coordinated Adaptive Power Management (CAPM) Technique for Sensor Network Nodes

by

Gauri Hiren Joshi

A Thesis Submitted in Partial Fulfilment of the Requirements for the Degree of

Doctor of Philosophy

in

Information and Communication Technology

to

Dhirubhai Ambani Institute of Information and Communication Technology,

Gandhinagar, India

January 2013

Declaration

This is to certify that

1. the thesis comprises my original work towards the degree of Doctor of Philosophy in Information and Communication Technology at DA-IICT and has not been submitted elsewhere for a degree,

2. due acknowledgement has been made in the text to all other material used.

Gauri Hiren Joshi

Certificate

This is to certify that the thesis work entitled *Coordinated Adaptive Power Management (CAPM) Technique for Sensor Network Nodes* has been carried out by *Gauri Hiren Joshi* (200521007) for the degree of Doctor of Philosophy in Information and Communication Technology at this institute under my supervision.

Prof. Prabhat Ranjan

Acknowledgments

The completion of my doctoral program thesis has been a long journey. This thesis would not have been possible and complete without the invaluable aid and support of a several people. Without these supporters, I may not have gotten to where I am today.

To the many people I would like to give special thanks, my primary gratitude goes to my academic and research supervisor Professor Dr. Prabhat Ranjan for the constant encouragement and priceless suggestions. These pearls of wisdom not only helped me complete my thesis but also molded me for my career in the future. His leadership, support, attention to detail, hard work, and scholarship have set an example for me. Sincere thanks to Professor Dr. S. C. Sahasrabudhe, Director DAIICT, Gandhinagar for providing the best possible research facilities in the institute. I would also like to mention Dr. T.S. Kumbar, our librarian for delivering top quality resource facilities and making available all the books and papers as and when required. My sincere thanks to the advisory committee members Professor Dr. Sanjay Srivastava and Professor Dr. Jaideep Mulherkar for their thorough interest shown in my work. Their beneficial comments and their sparring valuable time for me is something I will cherish forever.

A very special thanks goes out to Professor Dr. S. Dharmaraja of the Mathematics Department, IIT Delhi for sparring his precious time for both technical discussion and immense moral support. Also, my sincere thanks goes to Professor Kishor S. Trivedi, Duke University, Durham, North Carolina for making software tool SHARPE available to me.

I wish to place on record my candid thanks to Ms. Resham Vinayak (Research scholar, Mathematics Department, IIT Delhi), Mr. Guneshwar Anand (Faculty, CEPT Ahmedabad), Mr. Sudhanshu Dwivedi, Mr. Anshul Goyal and Mr. Hiren Shah (DAIICT) for their direct assistance. Their contribution, time and efforts will remain unforgettable forever.

I express my deepest gratitude to the esteemed colleagues at DAIICT- Shubham Jain, Sunil Jardosh, Ratnik Gandhi, Pratik Shah, A. Purushothaman, Bhavesh Dharmani and Sai Nambiar among others for the instant help they provided and especially for boosting my morale. They created a wonderful work environment which made the thesis easier.

The thesis would not have been possible without the monumental support of my family. My husband, Hiren has been perhaps the most patient and supportive witness to my academic journey over the past years and I thank him for being there all throughout. My children, Mukul and Kaushal are the sunshine of my life and their love as well as affection has made me realize my capabilities even better and taken out the best in me!

My deepest gratitude and thanks to the wonderful parents God has blessed me with. Mr. Sharad Kulkarni and Mrs. Shailaja Kulkarni who provided me with a strong upbringing thus laying the foundation for my academic journey as well as Mr. Jagdish Prasad Joshi and Mrs. Sumitra Joshi who continued to support me in all spheres of life. The support, encouragement and motivation they provided me cannot be thanked just by mere words. This journey, both academic and personal would not have been possible without these four invaluable individuals, I have the privilege to call my parents. Completion of this arduous task is one of their dreams coming true!

(Gauri Hiren Joshi)

Abstract

Small size of wireless sensor nodes limit the size of the battery supported on it which directly limits the capacity of the battery and lifetime of the sensor nodes. Though energy scavenging is one of the solutions, it is not always reliable due to the unavailability of energy to be scavenged (solar, wind, vibrations etc) all the time. Wireless Sensor Networks (WSN) are required to work over longer period as they have been deployed to detect or monitor some rare events or objects. It creates the need for stringent power optimization at all the layers of sensor network. In order to have a WSN alive sufficient number of sensor node needs to be alive covering the sensing field and providing a route to the sink.

Aim of this research is to increase the life time of battery powered wireless sensor nodes along with the Quality of Service (QoS) ensured. QoS parameter considered is data loss due to buffer overflow. Each sensor node can individually and independently make decisions about its operating state depending on the current workload. QoS parameter is kept at utmost priority along with the power optimization. Increasing the lifetime of sensor nodes means decreasing the time averaged power consumption of processing and transmission units. This research creates the need for a dynamically reconfigurable sensor node in which computational as well as communication tasks can be carried out with different speeds and power as per the requirement. DVFS (Dynamic Voltage Frequency Scaling) and DMS (Dynamic Modulation Scaling) techniques help to increase the power efficiency of communication unit and radio unit respectively. In this thesis we are focusing not only

on the simultaneous use of DVFS and DMS but also on coordinated integration of the two technologies what we have termed as Coordinated Adaptive Power Management (CAPM) technique.

We have modeled a wireless sensor node as a tandem queue. Microcontroller has been considered as first server and radio transmitter is the second one. In case of event detection applications, data traffic is not uniform all the time. In the absence of an event, data arrived in a node is very small, while data traffic increases suddenly when an event occurs. We have tried to vary the service rate of processor using DVFS and transmitter using DMS techniques as per the actual workload. Operating with lower service rate consumes less power and with higher service rate, it consumes more power. Though increasing the lifetime of a sensor node is our main objective, equally we are concerned with the data loss due to buffer overflows. So we are suggesting a power optimization technique that trades off between power consumption and buffer overflow probability. During the period of heavy traffic, reducing the data loss due to buffer overflow is of higher priority and during low traffic period, power saving is of higher priority.

Comparison is made for a sensor node with various capabilities- fixed service rate, only DVFS, only DMS, both DVFS and DMS together and coordinated with each other. After analyzing the tandem queue model of sensor node, we have considered a sensor node as a single server having DVFS and DMS coordinated inside it. By changing the service rate of sensor node (μ_1, μ_2,...), specific processing rate and transmission rate are selected internally. $M^{[x]}/M/1/N$ model has been used to capture bulk data arrival and represented with a Markov chain. As it has become very complex to solve Markov chain state equations for performance analysis, we have developed GSPN models of sensor node with various capabilities. Using SHARPE software analytical tool, we have analyzed all these models for various performance parameters. MATLAB simulations carried out and compared with the analytical results. For WSN applications where non uniform data traffic exists, CAPM has shown the lifetime improvement along with reduced buffer overflow probability.

We have analyzed the performance of a wireless sensor node under various data arrival rates and results obtained from analytical tool SHARPE and MATLAB simulations for various models of sensor node have shown that CAPM technique reduces the idle time probability and idle power wastage during normal periods and reduces data loss due to buffer overflow during catastrophic period and increases the lifetime of wireless sensor network node. As compared to fixed service rate sensor node lifetime increase of 15% was seen when only DVFS was implemented on a sensor node while implementing only DMS it was 17.5% but DVFS and DMS together (CAPM) applied on a sensor node resulted in 27.22% lifetime increase. All these results support the concept of CAPM technique.

Abbreviations

- **ACK:** Acknowledgement
- **BER:** Bit Error Rate
- **CAPM:** Coordinated Adaptive Power Mangement
- **cat.:** Catastrophic period
- **DFS:** Dynamic Frequency Scaling
- **DMS:** Dynamic modulation scaling
- **DPM:** Dynamic Power Management
- **DVFS:** Dynamic voltage frequency scaling
- **DVS:** Dynamic Voltage Scaling
- **FIFO:** First In First Out
- **LAS:** Late Arrival System
- **MAC:** Medium Access Control layer
- **MAI:** Multiple Access Interference
- **Mod scaler:** Modulation scaler

- **MQAM:** M-ary Quadrature Amplitude Modulation
- **nor.:** Normal period
- **OV:** Overflow probability
- **PHY:** Phase Shift Keying
- **PSM:** Power Saving Mode
- **QoS:** Quality of Service
- **Rx:** Receiver
- **SNR:** Signal to Noise Ratio
- **Tx/Rx:** Transceiver
- **Tx:** Transmitter
- **WSN:** Wireless Sensor Network

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Wireless Sensor Networks (WSN) has evolved as a much promising technique with applications in almost all the fields of life. WSN applications include environmental monitoring [1] [2], structural monitoring [3] [4], machine condition monitoring [5], surveillance systems [6–8], medical monitoring [9], aircraft monitoring systems [10], body sensing [11] and many more.

Micro sensor nodes are powered by very small batteries which can supply a small amount of energy for the functioning of nodes and nodes are expected to work reliably over a longer period of time which may extend up to few years. Replacing the batteries is not feasible due to remote, random and inaccessible deployment of nodes as well as due to large number of nodes in the network. State of the art sensor nodes are UCLA's Medusa MK-II, Berkeley's Motes, MIT's μAMPs, Rene, Mica-2, Telos, Mica-Z etc. Most of the Wireless sensor networks are application specific having its own design and implementation challenges but a common challenge for all wireless sensor networks is of power management.

Energy scavenging is one of the promising option to overcome limited energy supplied by the batteries. Electrical energy can be scavenged or harvested from environmental sources. Different power scavenging mechanisms are photovoltaic, pressure variations, thermoelectric, human and vibration energy [12]. Wireless sensor networks lifetime can be improved with the help of energy harvesting. Continuous research [13–16] is going on to develop energy harvesting WSN as it seems to overcome the problem of stringent power constraint. Paper [17] discusses the various ambient energy harvesting methods and challenges of designing networking protocols for such WSNs powered by ambient energy harvesting. Harvesting energy for small devices like wireless sensors is challenging as the energy harvesting device has to be comparable in size (i.e. small enough) with rest of the system. For WSN applications the energy demands are large because of the wireless communications but the availability of environmental power can not be guaranteed as node's energy harvesting opportunities may vary from place to place and time to time. Also the energy consumption of the nodes varies due to uneven distribution of workloads or network traffic. All these constraints indicate that sensor nodes with energy scavenging ability also need energy management.

1.1 Energy efficient and low power system design

Energy minimization techniques have been proposed at all levels of the design hierarchy, right from the system behavior to the silicon level [18–21]. Existing techniques include those at technological, circuit, logic, architecture and algorithmic levels.

Techniques based on transistor sizing, floor-planning and wiring have been developed at silicon level [22–25]. At circuit level, Dynamic Voltage Scaling (DVS), dynamic adjustment in supply voltage throughout the execution period has been illustrated to minimize the energy consumption or power dissipation. Pedram and Chang [26] have investigated the use of multiple supply voltages for energy minimization in integrated circuits. Donghwan et al. [27] and others [28] [29] have evaluated the impact of dynamic voltage scaling

(DVS) on the performance - power trade off in context of various application domains. Clock frequency control is applied in tandem with DVS (together it becomes DVFS) to achieve better results [30]. Another popular approach [31] to power minimization at the same level is dynamic power management which aims to reduce power consumption of integrated circuits by selectively shutting down idle components at any instant. Reversible or adiabatic logic [32] [33] is another technique considered. Several circuit level power reduction techniques have been discussed in [34]. Low power design issue has been addressed at logic level through logic minimization techniques [35] while the techniques akin to pipelining [36] [37] and parallel processing [38] have been employed at the architectural level to address the same issue.

There is growing consensus that advances in battery technology and low-power circuit design cannot, by themselves, meet the energy needs of future mobile computers and the higher level techniques must also be involved. Recently the algorithmic or behavioral level techniques are receiving the maximum attention due to the fact that higher the level of the design hierarchy where energy minimization is tackled, higher is the possibility of reduction in energy consumption [19].

As highlighted in [39] power management is required at node level as well as at network level in WSN. Power efficiency improvement needs low-power hardware designs [40] along with energy-efficient protocols [41]. Energy-efficient protocols include scheduling protocols in the MAC layer [42], communication protocols in link layer and routing protocols in the network layer [43]. To increase the lifetime of wireless networks in which nodes are battery operated, various energy management techniques have been proposed in the literature. An overview of these techniques has been given in [44]. Mostly researchers have suggested to reduce the wireless communication power [45] [46]. In [47–49] shutting down processing unit has been suggested in order to optimize topology and communication range. Shutting down results in power saving. Information about network traffic and route is exploited in [50] to decide when to turn the node into the sleeping state from ON state. Dynamic Frequency Scaling (DFS) and Dynamic Voltage Scaling (DVS) techniques are energy efficient when the processor is in idle state [51] to reduce power consumption. We discuss these techniques in brief as:

1. **Power aware routing:** At network level power aware routing is performed using different algorithms which helps to reduce end to end transmission power. Power-aware routing aims at minimizing the per packet transmission power in multihop routing [52–55]. The major limitation of power-aware routing is that it only minimizes the transmission power of nodes and ignores the power consumption in other radio states. As a result, it is only effective for the radio platforms with high transmission power or the networks with high workload where nodes operate in transmission state most of the time.

2. **Duty cycling:** In a wireless sensor networks where a large number of sensor nodes are deployed it is possible to exploit node redundancy to select only a minimum subset of nodes to remain active for maintaining connectivity [56]. Remaining nodes can go to sleep and save energy. Duty cycling serves two fold purpose- topology control and power management.

3. **Topology Control:** Finding the optimal subset of nodes that assures network connectivity is referred to as topology control. In this perspective topology control can be seen as a complimentary technique to power management. Topology control increases the network lifetime by a factor of 2-3, when compared to a network with all nodes always ON [2,57,58]. Topology control aims to preserve the desirable properties of a wireless network (e.g. K-connectivity) through reduced nodal transmission

power. A comprehensive survey on existing topology control schemes can be found in [59].

4. **Sleep scheduling:** In sleep scheduling, each node operates in a duty cycle composed of wake-up/sleep intervals. A sleep schedule with fixed duty cycle is adopted in 802.11 Power Saving Mode (PSM) [60]. Ye et al. propose a MAC protocol called S-MAC for WSNs [61]. Each node in S-MAC operates in synchronous adaptive duty cycles that can be extended based on network activity. T-MAC [62] is a MAC protocol that can mitigate the impact of low node duty cycle on network throughput by an adaptive sleep schedule adjustable based on workload variation. Polastre et al. [63] proposed a MAC protocol called B-MAC for WSNs that supports asynchronous sleep scheduling and adopts a low power listening scheme to wake up nodes. The major limitation of asynchronous sleep scheduling is its high overhead in neighbor discovery as each node in the neighborhood has different sleep schedules. Similar to connectivity maintenance, sleep scheduling only reduces the idle listening power of the network.

1.2 Some other power management techniques

The existing power management approaches only aim at reducing the power consumed in a particular radio state. As a result, they are only effective for certain radio platforms and network conditions. Topology control and power-aware routing protocols only reduce the transmission power of radio, and hence are not suitable for the applications with low workload or the radio platforms with high idle power consumption. Sleep scheduling protocols, on the other hand, only reduce the idle power consumption and hence are not effective when the network workload is high or the idle power consumption of radio is low. Clearly, a WSN needs to reduce the energy consumed in each of the radio states (i.e., transmission, reception, and idle) in order to minimize its total energy consumption, which requires effective application of all the above approaches.

1. **Dynamic Power Management (DPM)**

   Many researchers have studied and suggested DPM for wireless sensor network applications. In order to save the battery energy certain units of a wireless sensor node are allowed to enter in sleep mode and again turned active when required. Radio transceiver is one such unit which consumes maximum power for Tx/Rx (Tx- Transmit, Rx- Receive) and also consumes significant amount of power in the listening (idle) mode. For example, radio used in Mica-Z sensor node consumes 59.1 mW power in Rx mode, 52.2 mW power (maximum Tx output power) in Tx mode [64], ATmega128RFA1 radio module consumes 37.5 mW in Rx mode and 43.5 mW (maximum Tx output power) in Tx mode [65]. So it is always better to have radio in sleep mode rather than in the idle state when there is no data to transmit (idle mode).

2. **Dynamic Rate Selection**

   In wireless sensor networks, multihop communication is preferred because of less power available for data transmission. Lower transmission rates save the battery power but can not handle the large data during catastrophic conditions (event occurrence) and important data gets lost. This can be avoided by transmitting data at higher rates. So here the need arises in which sensor node transmission rates can be selected dynamically. This rate selection depends on the factors like actual workload, energy budget of each node, timing constraint on the packet delivery time and data loss due to buffer overflow. Rate can be selected as a function of one of these factors or a combination of two or more factors. Selected rate also needs to cope up with the channel conditions. All such factors contribute to select the rate dynamically. In [66] authors have suggested to select transmission rates based on the available energy budget in the nodes battery. Rate selection is subject to satisfying a time constraint on the end-to-end packet delivery time. A source node calculates the transmissions rates for other nodes on the route to meet the end-to-end delay. Source node is assumed to have the knowledge about energy budget of other nodes. In [67] [68] [69] objective is to maximize the throughput of transmission over a time-varying channel subject to an average transmit power constraint. Here both the buffer and channel state information is considered for transmission power and rate selection. Markov decision process (MDP) is used to model the situation. In [70] authors have considered the problem of rate selection for different classes of traffic. For each class of traffic an optimized service rate is selected such as to have no packet loss due to buffer overflow. The relation between the power consumption and the transmission rate (in bits per second) is well studied in information theory (Cover and Thomas, 1991). In [71] transmission rate and delay trade off has been analyzed. Scheduling algorithms which adaptively change the transmission power and rate, based on both the transmission queue backlog and the channel conditions are discussed in [72].

3. **Modulation Techniques**

   Effect of modulation techniques on energy efficiency has been analyzed in many papers. In [73] the binary and M-ary modulation schemes have been compared. It shows that M-ary modulation is more energy efficient than binary for a small overhead and transmit-on time. Here start up time is the time required for a node to transit from sleep state to wake up state. **M-ary modulation** is the key to adaptive modulation in which number of bits per symbol (constellation size) can be changed adaptively which **results in variable data rate but with constant symbol rate**. MQAM, MPSK, MFSK are some of the possible variations. In [74], it has been shown that MQAM modulation is efficient for short range communications. In [75] authors have referred energy efficient non complex modulation scheme as green modulation. It is shown that the On-Off Keying (OOK) displays a significant energy saving as compared to the optimized NC-MFSK in dense WSNs with small values of path-loss exponent. In recent years, several energy-efficient modulations have been studied in the physical layer of WSNs. e.g. [76] [77]. In [76] the authors have compared the battery power efficiency of PPM and OOK based on the Bit Error Rate (BER) and the cutoff rate of a WSN over path-loss Additive White Gaussian Noise (AWGN) channels. [77] investigates the energy efficiency of a centralized WSN with an adaptive MQAM scheme. However, adaptive approaches impose some additional system complexity due to the multi-level modulation formats plus the channel state information fed back from the sink node to the sensor node. Most of the pioneering work on energy-efficient modulations, including research in [76], has focused only on minimizing the average energy consumption per information bit, ignoring the effect of the bandwidth and transmission time duration. In a practical WSN, however, it is shown that minimizing the total energy consumption depends strongly on the active mode duration and the channel bandwidth [78]. NC-MFSK and OOK have the advantage of less complexity and cost in implementation than MQAM and Offset-QPSK used in the IEEE 802.15.4 protocol and can be considered as green modulation in WSN applications. In [79] M-ary PSK modulation scheme is analyzed in Rayleigh-Lognormal fading channel and found useful in enhancing the data throughput of the communication system.

1.3 Related work constraints

From the survey of existing literature on the power optimization for WSN at node level we observe that-

- DVFS and DMS are mostly studied in stand alone manner and not studied necessarily for WSN.
- Most of the existing work aims at minimizing the wireless communication energy consumption and ignores the computational energy consumption. Advanced applications like military surveillance require rigorous data processing which consume considerable energy and can not be ignored.
- Mostly existing work applies communication performance scaling to single-hop setups which can not be directly applied to multi-hop communication in WSN.
- Performance scaling techniques (DVFS and DMS) mainly studied for power- latency trade off. (not considered data loss due to buffer overflow).
- Models assume that energy consumption is a continuous function of message modulation level (or correspondingly message transmission latency) but in practice few discrete levels are only available.
- Performance scaling, sleep-wake up scheduling and power adaption have been addressed individually for power optimization.
- Few recent papers have talked about the integrated or joint use of DVFS and DMS on the sensor node but these papers have mainly focused on trade off between computation and communication latencies. Still coordination between DVFS and DMS has not been studied, buffer overflow has not been considered as QoS parameter and also detailed mathematical modeling of sensor node performance has not been found.

1.4 Research problem statement and objectives

With the above mentioned constraints in mind, aim of this research is to increase the life time of battery powered wireless sensor nodes though the final aim is to have a long lived WSN. Increasing the lifetime of individual sensor nodes will ultimately result in a long lived wireless sensor network. Each sensor node can individually and independently make decision about its operating state depending on the current workload. It provides flexibility to the sensor nodes which assures QoS along with the power optimization. QoS parameter is kept at utmost priority along with the power optimization. Increasing the lifetime of sensor node means decreasing the time averaged power consumption of processing and transmission units. This creates the need for a dynamically reconfigurable sensor node in which computational as well as communication tasks can be carried out with various speeds and powers. DVFS and DMS technologies help to increase the power efficiency of communication unit and radio unit respectively. Applying both of these simultaneously on a sensor node is expected to reduce the power wastage during idle period. In this thesis we are focusing not only on the simultaneous use of DVFS and DMS but also on coordinated integration of the two technologies. To achieve this objective, mathematical models for the operation of reconfigurable sensor nodes at the node level were built first. Then model's performance analyzed using GSPN module of software package SHARPE.

The fundamental trade-off between service delay and power consumption is exploited at physical layer which consists of transmitting with the lowest power over the longest feasible duration. At MAC level, many power reduction techniques suggest to transmit as fast as possible and turn the radio off. These two approaches seem contradictory though both results in power saving.

Transmitting with the lowest power (lower transmission rate) over the longest feasible duration (maximum tolerable delay) results in the power saving but seems useful for the applications where data arrival is low and smooth. In the situations like WSN where data arrival is not uniform (sometimes no data or very less data and sometimes suddenly bursty data). Always transmitting with the lowest power would not be able to maintain the delay constraint and will also result in data loss due to buffer overflow if buffer size is small (as in sensor nodes).

So meeting the delay constraint with no data loss (due to buffer overflow) and increasing the lifetime of small battery powered sensor nodes is the real challenge.

We have tried to combine both DVFS and DMS techniques optimally. By having a low duty cycled sensor node (fixed timeout driven schedule), sensor nodes are allowed to enter in the sleep state frequently resulting in power saving. Whenever sensor nodes become active they are allowed to select the service rate as per the actual workload requirement. It results in power saving when there is less workload and also reduces buffer overflow data loss by increasing the service rate during catastrophic conditions. Minimum and maximum service rates are determined by the tolerable delay and symbol error rate by considering MAI (Multiple Access Interference) effects.

Accordingly power and buffer overflow optimization problem at sensor node level can be stated as-

At the beginning of each time slot given the knowledge of input buffer occupancy (workload monitoring), select the coordinated operating states (service rates) of processing and transmitting units so that the average power consumed and data loss due to buffer overflow is minimized within the given latency bounds. Increasing the lifetime is prime objective but not at the cost of data loss.

Figure 1.1 elaborates the scope of this research thesis. We consider a model where individual nodes support both Dynamic Voltage frequency Scaling (DVFS) and Dynamic Modulation Scaling (DMS) power management techniques. Such model has been ana-

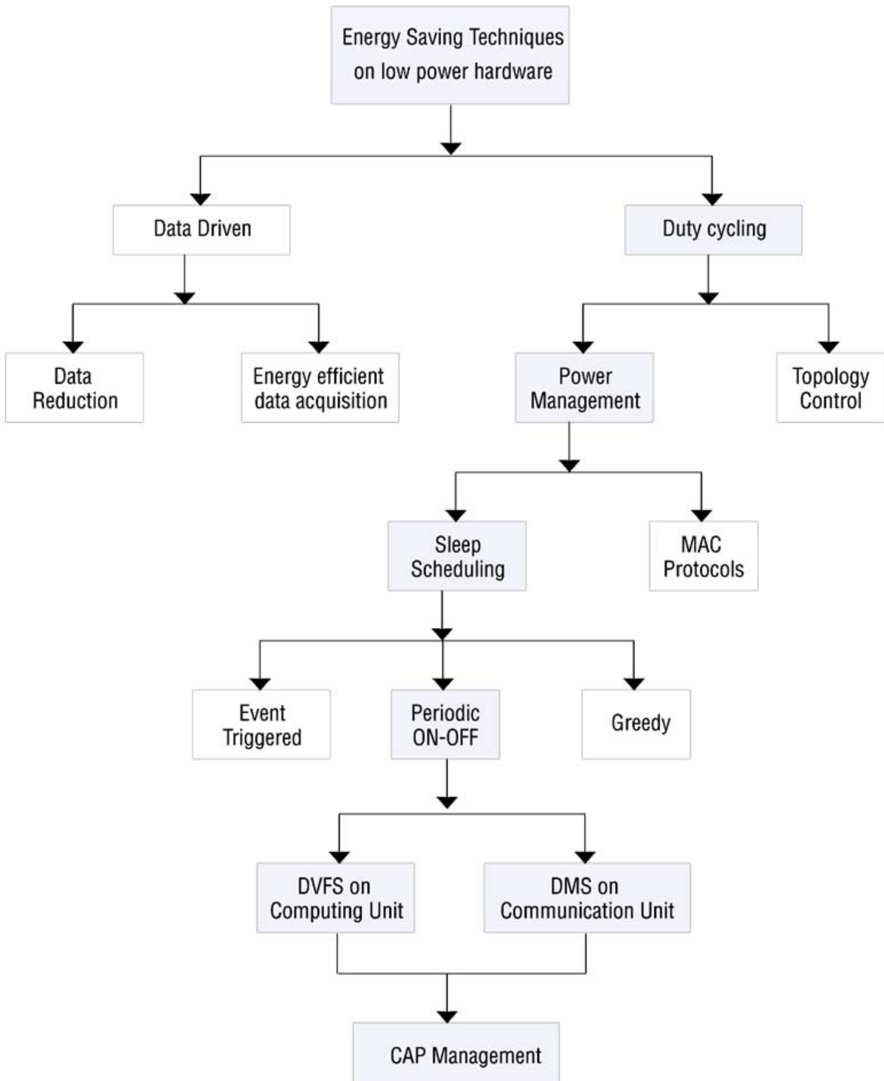


Figure 1.1: Area of research problem

lyzed in [80] in order to explore the energy-time tradeoffs. We not only aim at energy-time tradeoffs but also focus on reducing the data loss due to buffer overflows. So our power management technique is two dimensional as it tries to satisfy two QoS parameters namely latency and buffer overflow. We have coordinated DVFS and DMS techniques in order to have better power management. This coordinated implementation of DVFS and DMS on a sensor node we call as Coordinated Adaptive Power Mangement **(CAPM)** technique.

1.5 How CAPM is different from DPM?

DPM	**CAPM**
Processing and radio units of sensor node can independently enter sleep mode.	Both the units enter together in sleep mode or active mode.
Processing and radio units work with fixed service rate.	Both the units can change service rate adaptively in a coordinated manner.
Switching from sleep to active mode takes considerable time.	Switching from one active state to other does not need much time.
Possibility of data loss during switching period.	Possibility is very less.
Power optimization technique but can not assure network coverage, connectivity and also increases data loss due to buffer overflow probability.	Provides better network coverage, connectivity and power optimization along with reduced buffer overflow probability.

1.6 Contributions of the thesis

Contributions of the doctoral research work presented in the thesis can be outlined as:

- A node level approach increasing the life time of individual sensor node and thereby indirectly enhancing the lifetime of the wireless sensor network.
- A node independently selecting its service rate as per the actual work load at a particular time.
- Development of following models and MATLAB simulations:
  - A tandem queue model of a sensor node to capture the inside details rather than a single entity.
  - A rate adaptive sensor node model using CTMC (Contineous Time Markov Chain) for performance analysis.
  - GSPN (Generalized Stochastic Petri Nets) models for a wireless sensor node with various capabilities.
- Development of coordinated DVFS and DMS technique where processing and transmission rate of a sensor node is dynamically selected by only observing the input buffer.
- A first attempt to coordinate DVFS and DMS to reduce the data loss due to buffer overflow along with power optimization.
- Use of SHARPE software tool for extensive analysis of sensor node performance for bulk data arrival and various data arrival rates using GSPN models.
- Use of CAP management technique leading to energy efficient Time out sleep scheduling algorithm along with better coverage and connectivity.
- Lifetime increase of 27.22% and buffer overflow probability reduction up to 58.33% (during catastrophic period) for a wireless sensor network node with CAPM as compared to that of sensor node with fixed service rate.

1.7 Thesis organization

This thesis consist of eight chapters.

- Chapter 1 has presented wireless sensor networks challenges and constraints of sensor nodes. It highlights the crucial need for power optimization at node level and motivation behind this research thesis.
- Chapter 2 introduces the background theory and concepts required for this thesis. It briefly discusses various sleep scheduling strategies used to achieve low duty cycled sensor nodes. Adaptive power optimization techniques DVFS used for processor and DMS for transmitter has also been discussed.
- Chapter 3 explores the concept of Coordinated Adaptive Power management (CAPM) technique. Effects of coordination between DVFS and DMS techniques as per the actual workload in the input buffer have been discussed and compared with that of having no coordination. This chapter outlines the goals, requirements and components of aforesaid concept.
- Chapter 4 presents tandem queue model of a wireless sensor node. A sensor node with fixed service rate, with only DVFS implemented, with only DMS implemented and both DVFS and DMS implemented together has been considered. MATLAB simulation of these sensor node models has been carried out. Adaptive threshold policy for rate switching is discussed in this chapter.
- Chapter 5 presents the CTMC model for coordinated rate adaptive sensor node. It represents the $M^{[x]}/M/1/N$ model for bulk arrival and batch service given by multi-rate wireless sensor node. Performance of sensor node has been analyzed in normal

workload conditions as well as in catastrophic workload conditions. It presents the over all algorithm for CAP scheduler.

- Chapter 6 presents GSPN models of wireless sensor node with various capabilities and their performance under various workload conditions. Performance analysis carried out with the help of software package SHARPE.

- Chapter 7 discusses the overheads associated with CAPM technique. Rate switching overheads associated with low power ATmega128RFA1 has been discussed.

- Chapter 8 summarizes the thesis and presents the final conclusions. It also provides directions for future research.

Chapter 2

Power Optimization Techniques Used at Sensor Node Level: a survey

We consider a low power hardware selected for Wireless Sensor Nodes. Applying sleep scheduling on these nodes lowers the duty cycles and increases the lifetime of sensor node. Large number of sensor nodes allows some nearby sensor nodes to enter sleep state as some other nodes are available to take care of sensing and connectivity. Our focus is on power consumption reduction during the ON periods of sensor node. Though our aim is to increase the lifetime of sensor nodes but maintaining the QoS is of highest priority even at the cost of increased power consumption. In this chapter we discuss the power saving techniques like sleep scheduling, Dynamic Voltage Frequency Scaling (DVFS) and Dynamic Modulation Scaling (DMS). Sleep scheduling lowers the duty cycle while DVFS and DMS trades off power against latency parameters of computing unit and communication unit respectively. QoS parameter we have considered is data loss due to buffer overflow along with latency constraint.

2.1 Sleep scheduling or Duty cycling

Sleep scheduling protocols are implemented in the MAC layer [42] to increase the lifetime of wireless sensor networks. In a wireless sensor networks, where a large number of sensor nodes are deployed, it is possible to exploit node redundancy to select only a minimum subset of nodes to remain active for maintaining connectivity [56]. Remaining nodes can go to sleep and save energy. Duty cycling serves dual purpose - topology control and power management. Topology control is important in view of network coverage and connectivity issues. It is highly desirable to sense the complete area of interest at all time instants (coverage) so that event occurring any where, at any time will get detected. Another important issue is to send the sensed information to the data sink timely. For this purpose always there should be some path available for data forwarding (connectivity). A coverage and connectivity problem was considered in [81]. Conditions on the sensing radius, the network density and the reliability were derived to achieve asymptotic coverage and connectivity.

Fundamental relationship within the context of providing network coverage using low duty-cycled sensors for surveillance purposes is examined in [82]. In this paper, authors investigated the problem of providing network coverage using low duty cycled sensors. It showed that using coordinated sleep algorithms, one can obtain greater reduction in the duty cycle at the expense of extra control overhead as compared to the random sleep schedules. In sleep scheduling, each node operates in a duty cycle composed of wake-up/sleep intervals. A sleep schedule with fixed duty cycle is adopted in 802.11 Power Saving Mode (PSM) [60]. Ye et al. propose a MAC protocol called S-MAC for WSNs [61]. Each node in S-MAC operates in synchronous adaptive duty cycles that can be extended based on network activity. T-MAC [62] is a MAC protocol that can mitigate the impact of low node duty cycle on network throughput by an adaptive sleep schedule adjustable based on workload variation. Polastre et al. [63] proposed a MAC protocol called B-MAC for WSNs that supports asynchronous sleep scheduling and adopts a low power listening scheme to wake up nodes. The major limitation of asynchronous sleep scheduling is its high overhead in neighbor discovery as each node in the neighborhood has different sleep schedules. Similar to connectivity maintenance, sleep scheduling only reduces the idle listening power of the network.

Optimal sleep scheduling is discussed in [83], where instead of periodically waking up the sensor nodes, sleep period is kept some multiple integer *N* of the ON period. Value of *N* is optimally selected depending on the packets arrived in the queue. For such adaptive sleep schedules, during sleep state only transmitter is turned off but receiver is kept on. Over short distance communication, power consumed for data transmission and reception are comparable and hence not much power saving results. Also it looses the synchronization among the sleep schedules of neighboring nodes and may not ensure good coverage and connectivity. Only the switching cost is saved by not switching on the transmitter frequently. In [84], it has been emphasized that keeping nodes in the deep sleep state results in energy saving but the loss of communication capabilities during the sleep states not only affects network connectivity in the multihop WSN environment but also reduces the overall sensing coverage. Sensed data cannot be timely forwarded to the data sink as the nodes in the routing path have their transceivers turned OFF. Also the data sink won't be able to communicate with the nodes though needs to deliver important instructions.

Many papers consider sleep scheduling of the transceivers only in which during sleep state no transmission takes place but data arrival continues. It results in data transmission latency and also network coverage and connectivity are not considered in the analysis. In [83] an optimal sleep scheduling algorithm is worked out based on sleep duration, arrival rate and packet delay and energy consumption. Various sleep scheduling algorithms can be classified as:

1. **Always ON (No scheduling):** In this case a sensor node always remains ON from the time of its deployment in the sensing field irrespective of whether there is any task to perform or not. Here, sensor node may remain idle over a longer duration and power gets wasted without doing any work. Life time of sensor node and obviously the life time of network is very small. In event detection application sensor network may fail even before the occurance of event due to depletion of batteries, if no energy scavenging means are provided.

2. **Rotational sleep schedule or Time out schedule:** In this case a small percentage of the sensor nodes are active at a time. After a certain predetermined time these nodes are turned OFF (lowest energy consumption mode) and some other sensor nodes are turned ON. A sleep scheduler decides which nodes to wake up so that complete field of interest can be sensed all the time. Better time synchronization between sensor nodes is required in this case. For every node ON time and OFF time are adjusted such that,

$$T_{ON} << T_{OFF}$$

$$\% \ Duty\ cycle = \frac{T_{ON}}{(T_{ON} + T_{OFF})} \cdot 100 \tag{2.1}$$

Power saving can be achieved by having low duty cycle. Periodically nodes are switched ON to check whether there is any task pending to perform. If any pending task, performs it, otherwise remains waiting for the work over a period T_{ON} and then switches to OFF state. In this condition coverage of the sensing field and connectivity between the active nodes at all times is the main issue that sleep scheduler needs to take care. Periodically every node wakes up with start up time T_{wakeup}, and remains ON for a fixed time T_{ON}. Then it returns to sleep until the next wakeup instant. In this schedule, network connectivity can be assured by adjusting the wake up periods of sensor nodes such that whenever a node wakes up, there is at least one neighboring node in the active state. It requires certain neighboring nodes to be synchronized in order to wake up at the same time. Clock synchronization in wireless sensor networks is beyond the scope of this work. Here we simply assume that nodes are synchronized by means of some synchronization protocol. In a Fully Synchronized Pattern, all nodes in the network wake up periodically and remain *ON* for a fixed time duration T_{ON}. Then, they return to sleep until the next wakeup instant as in [85]. This is the simplest wake up scheme and hence used in several practical implementations including TinyDB [86] and TASK [87]. A fully synchronized wakeup scheme is also used in MAC protocols such as S-MAC [88] and T-MAC [89]. The fully synchronized scheme applies equally well to both flat and structured sensor networks. The main drawback is, as neighboring nodes become active at the same time after a long sleep period they try to transmit simultaneously causing a large number of collisions. Also the scheme is not very flexible since the fixed active and sleep periods and does not adapt to variations in the traffic pattern and/or network topology.

In this thesis, we have tried to remove this drawback by offering a multiple service rate capability to the sensor nodes so that sensor nodes can select the service rate as per the traffic requirement and accordingly consume the power. Within the fixed schedule, flexibility is added so as to reduce the power consumption.

3. **Greedy sleep schedule:** It is a modification of rotational sleep scheduling. Frequency of turning ON is predetermined but ON time and OFF times are not fixed. Depending on the workload requirement, T_{ON} and T_{OFF} are adjusted. Whenever a

node is turned ON, it will check for the task to perform, in a certain time period if there is no work to do then instead of remaining idle it turns OFF. It results in to better energy saving but the issue of coverage and connectivity is not properly assured all the times, which may lead to miss some data of interest.

4. **External wake-up or event triggered schedule:** Periodically waking up node consumes lot of energy due to larger start up time. In this scenario a sensor node remains in sleep state and is turned ON when required to perform some task. A node is woken up by some external agent when an event occurs. Different wake up protocols are used to fast wake up the node when required. This scheme results in maximum energy savings and prolonges lifetime of the network. But this needs a very good wake up strategy for fast wake up of the nodes so that event is not missed and also it depends on external wakeup agent.

   It is preferred in all event-driven scenarios like fire detection, surveillance of machine failures etc. In such applications sensor nodes are in the sleep state (only sensing unit active for monitoring the surroundings) for most of the time. As soon as an event is detected, nodes are woken up with the help of low power radio. It needs two different channels, one for normal data communication and another wakeup channel for awakening the nodes are needed. This wake up schedule is most energy efficient as nodes are awaken only when needed so nodes do not remain in idle state and power is saved. This energy efficiency comes with the drawback of additional cost for the second radio and also there is the possible mismatch between the coverage of the two radios.

Because of the significant start up time of transceivers as well as processors it is not a wise decision to reduce ON time and entering the node in the sleep state [90]. As the ratio of

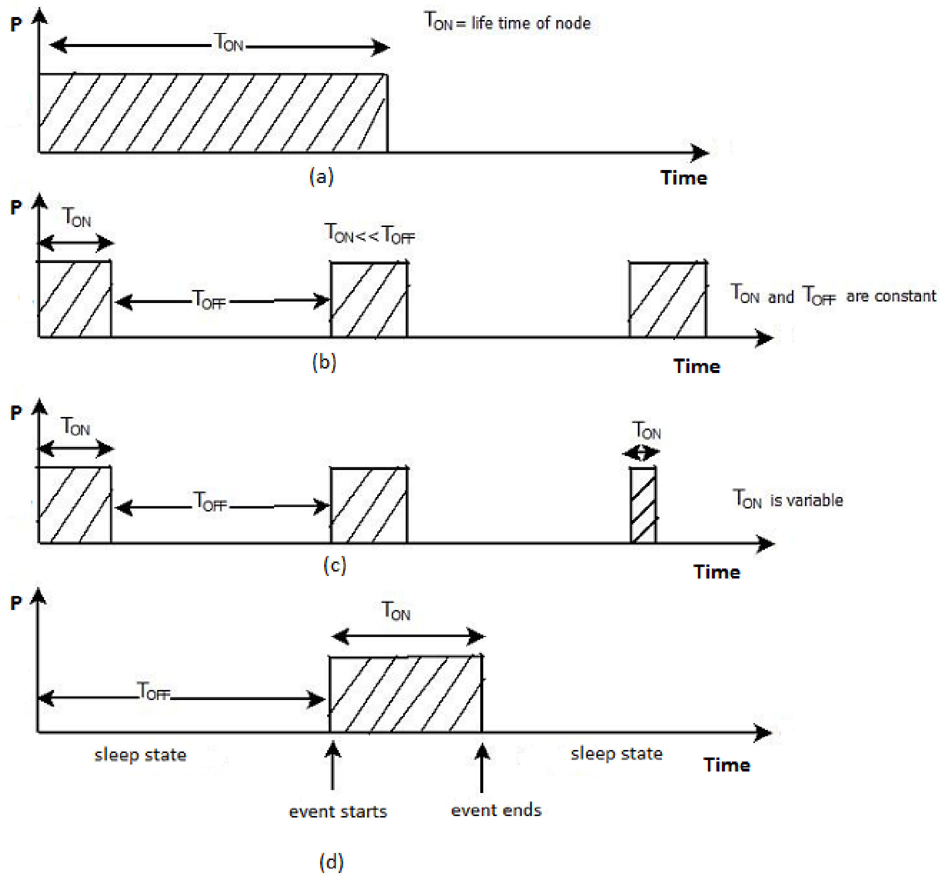


Figure 2.1: Sleep scheduling schemes: (a) Always ON, (b) Rotational, (c) Greedy, (d) Event triggered wakeup

ON time to start up time decreases, power consumed per packet transmission increases. So sensor node must be kept ON for sufficient amount of time.

From these discussions, **rotational sleep schedule** seems the suitable one as it can provide better network coverage and connectivity over all the time periods. In the next sections, we discuss power optimization techniques which help to save the power during ON period of the rotational sleep schedule.

2.2 Optimizing computational energy using Dynamic Voltage Frequency Scaling (DVFS)

Dynamic voltage frequency scaling (DVFS) is a technique that varies the supply voltage and clock frequency based on the computational load to provide the desired performance with the minimal amount of energy consumption. In most of the WSN applications, sensor nodes have a time varying computational load and hence peak system performance is not always required. DVFS exploits this fact by dynamically adapting the processor's supply voltage and operating frequency to satisfy the instantaneous processing requirement.

Here performance of processor is lowered against its energy efficiency. Performance is degraded in the sense that it takes more time for processing and introduces computational delay (latency), which is the cost paid to save computational energy. So this means of computational energy saving can be used only within the latency constraint, which is not going to adversely affect the performance of the network. This latency constraint is different for different applications. Several modern processors such as Intel's Strong-Arm [91], Transmeta's Crusoe [92] and Intel's XScale [93] support dynamic voltage and frequency scaling. There is an inverse relationship between the task processing time and the operating clock frequency. Also there is a cubic relationship between the system power and the operating clock frequency. The system power can be divided as fixed (static) power and variable (dynamic) power. Fixed power remains unchanged during the active period of the system regardless of whether system is processing some task or is idle. Examples of fixed power component includes DC-DC converter power and PLL power as well as leakage power dissipation. Variable power consumption changes with time depending on the state of the system whether busy or idle and also depend on operating clock frequency of the system. Examples of dynamic power include the CPU and memory power dissipations as well as I/O controller power. Standby power is defined as the total of fixed power plus idle power components of the system [94]. Total power dissipation in a CMOS based circuit is given as-

$$P_{Total} = P_{dyn} + P_{stat} \tag{2.2}$$

where P_{dyn} is the dynamic power component due to its dependence on the switching behavior and the frequency of the circuit.

$$P_{dyn} = C_p \cdot f_{clk} \cdot V_{DD}^2 \tag{2.3}$$

Here C_p is the constant of proportionality (switching capacitance) related to the effective parasitic capacitance of the logic gates.

P_{stat} is the static power component also called as leakage component as it results from the leakage current.

$$P_{stat} = S_p \cdot V_{DD}^2 \tag{2.4}$$

Here S_p is the constant of proportionality that depends on the conductivity properties of the circuit. Task processing time or time delay required to complete the task is given as

$$T_d = \frac{V_{DD}}{K(V_{DD} - V_t)^{\alpha}} \tag{2.5}$$

In these equations V_{DD} is the supply voltage and V_t is the gate threshold voltage. Since the clock frequency supported on the processor/ controller is linearly dependent on it's supply voltage, V_{DD} can be written in terms of f_{clk} as,

$$V_{DD} = \frac{f_{clk}}{K} + \varepsilon \tag{2.6}$$

where K and ε are hardware dependent parameters [95]. So by reducing the clock frequency it is possible to reduce the supply voltage which reduces dynamic power component cubically and also the static power component quadratically. On the other hand it affects the performance (T_d) only linearly. If we reduce the operational clock frequency we can get linear energy savings. Additional quadratic energy savings can be obtained if we reduce the power supply voltage to the minimum required for that particular frequency.

DVFS is a combination of both Dynamic Voltage Scaling (DVS) and Dynamic Frequency Scaling (DFS) to reduce power consumption by varying supply voltage and clock frequency [96–101]. Practical implementations of DVFS are: Variable Speed Constant Bandwidth Server (VS-CBS) [102], Genetic List Scheduling Algorithm for Energy-Efficiency (EE-GLSA) [103], General Dynamic Voltage Scaling (GDVS) [104] and Real-Time DVS (RT-DVS) [105–108]. A performance monitoring unit (PMU) based applications have exploited DVFS technique for energy saving with tolerable performance degradation, examples are [109–112]. Many researchers have focused on the CPU power reduction using DVFS techniques [113–118]. Figure 2.2 shows the power-latency trade off that

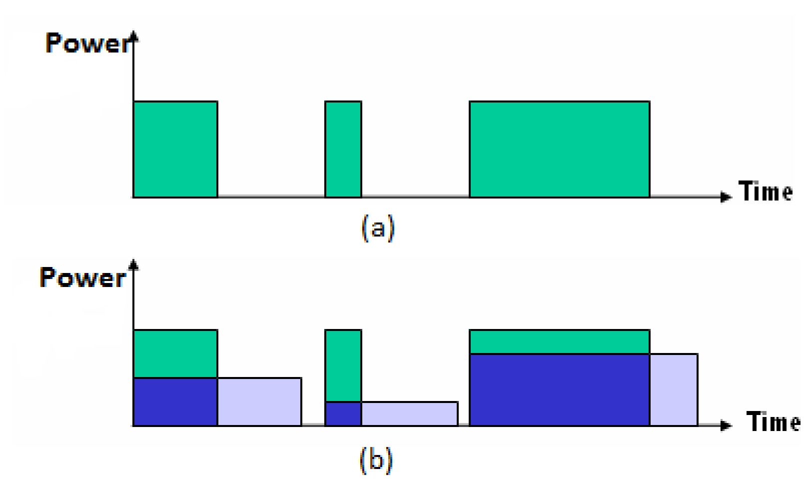


Figure 2.2: Power consumption and idle periods of a processor (a) without DVFS, (b) with DVFS

can be achieved using DVFS. DVFS reduces idle time periods. From this figure, DVFS becomes an energy efficient technique only if the energy saving is more than the energy cost. In the following section, we discuss similar technique applicable to radio transmitter which trades off transmission power and latency.

2.3 Optimizing communication energy using Dynamic Modulation Scaling (DMS)

Apart from monitoring its environment by means of its sensors and transmitting its own data to the rest of the nodes, a wireless sensor network node has to act as router, broadcasting packets received from other nodes. In fact, in a typical sensor network, a great percentage, around 65% , of all the received packets in a node have to be redirected to other destinations [119] . Modulation scaling technique trades off energy consumption against transmission delay (latency). Dynamically varying the modulation index is a concept of M-ary modulation (Refer section 1.2) which is effective for sensor network environment. Figure 2.3 shows the conceptual block schematic of DMS technique. Output buffer is monitored and as per the buffer occupancy a particular modulation scheme (b) is selected and the required transmit power levels are adjusted.

Generally, the energy consumption of the radio consists of two components - an RF energy component that depends on the transmission distance and modulation parameters and an electronics energy component that accounts for the energy consumed by the circuitry that performs frequency synthesis, filtering, up- converting, etc. The energy required for a complete radio transmission can be described as

$$E_{tx} = E_{EL} + E_{RF} \tag{2.7}$$

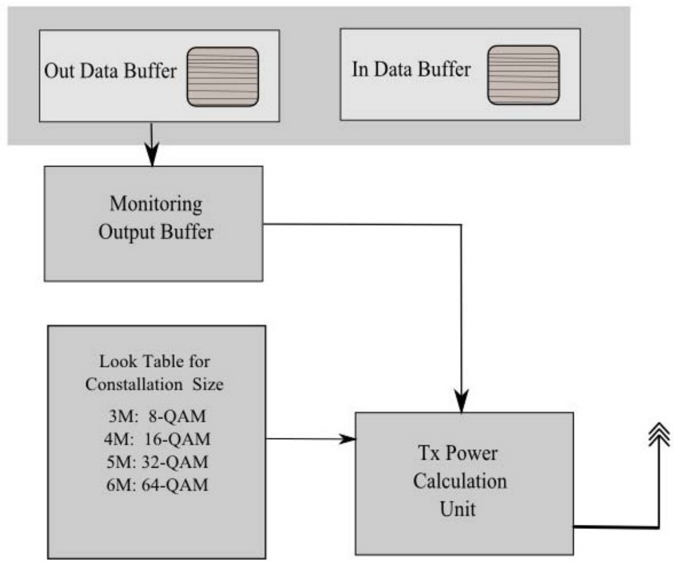


Figure 2.3: DMS concept

where E_{EL} represents energy consumed by electronic circuitry and E_{RF} energy radiated (mainly because of power amplifier).

$$E_{EL} = P_{tx} \cdot (T_{ON} + T_{start}) \tag{2.8}$$

$$E_{RF} = P_{out} \cdot T_{ON} \tag{2.9}$$

where,

- P_{tx} represents the power consumed by the transmitter electronics.
- T_{start} is the startup time.
- T_{ON} is the time duration in which sensor node remains ON.

The most evident way to make communication efficiency higher is reducing transmission time (T_{ON}). This can be achieved by codifying more than one bit per symbol(M-ary modulation). Using several modulation schemes, dynamically selecting the proper one, supplying the radio with a power value that can meet the value of Signal to Noise Ratio (SNR) other than the maximum one, can reduce energy cost of data transmission. The startup time T_{start} is of specific concern. It seems better to reduce the transmission time $T_{transmit}$ in order to reduce the energy consumption, so generally it is desirable to transmit as fast as possible and turn to OFF state. To reduce transmission time, it is desirable to transmit multiple bits per symbol. For today's off- the- shelf transceivers, startup time is much higher (hundreds of microseconds) and it increases the E_{EL} very aggressively as compared to E_{RF}. So switching transmitter ON and OFF frequently is not a wise decision and may not result in significant energy saving.

The transmitter schedules packets out of the queue at the rate of R_t at time t and uses power P_t for transmission. The energy function is monotonically increasing and strictly convex in R_t. It was shown that the average power is exponentially related to the rate [120].

Consider M-QAM modulation, where energy consumption per bit is given by [121]

$$E_{bit} = [C_s \cdot (2^b - 1) + C_E + C_R \cdot \frac{R_{smax}}{R_s}] \cdot \frac{1}{b} \tag{2.10}$$

where C_s, C_E and C_R are hardware dependent factors and have been derived in [121] . Time required to transmit one bit is given as

$$T_{bit} = \frac{1}{b \cdot R_s} \tag{2.11}$$

Main motivation behind adaptive modulation is to maintain fair QoS or to maintain a constant SNR. In wireless communication, proper SNR should be assured for the required QoS. Transmitted power level [122] can be varied as the noise level varies so as to keep SNR constant. But many times it is not the SNR to be maintained constant but it is the QoS parameter like BER (bit error rate) to be maintained over varying SNR. Such objective can be achieved by changing symbol transmission rate [123] as the SNR varies, by varying constellation size (number of bits per symbol) [124–126] or by changing coding rate or coding scheme [127] or any combination of these parameters [128–130] . Without sacrificing bit-error rate (BER), such schemes transmit at high speeds under favorable channel conditions, and reducing throughput as the channel degrades. Spectral efficiency can be increased by optimizing both the transmission rate and transmission power [131]. Slowing down the speed of packet transmission saves energy but results in packet drops as a result of buffer overflow.

In [132] authors have proposed a strategy to bound the number of packet drops. Energy efficient packet scheduling policy with latency constraint using DMS has been proposed in [133]. Curt Schurgers et al [121] studied the dynamic modulation scaling for energy efficiency of communication devices and analyzed the trade off between communication energy and transmission latency. They systematically compared Dynamic voltage scaling with dynamic modulation scaling

In [134] adaptive modulation scaling has been analyzed using queuing theory for checking the energy efficiency, packet delivery latency and packet loss. In [90] authors have discussed the problem of start up energy and have compared binary modulation with multi level modulation. Higher start up time not only increases the start up energy consumption but also reduces the energy efficiency of M-ary modulation as $T_{ON} : T_{start}$ ratio reduces [90]. Centralized and distributed control schemes have been discussed in [135] for selecting optimal value of modulation level, where adaptive modulation scaling has been analyzed using queuing theory.

Our approach is different relative to all of these adaptive techniques in that we optimize both the transmission rate and power to reduce the data loss due to congestion (buffer overflow). In adaptive modulation scheme many parameters can be adjusted according to the channel conditions such as the transmit power, modulation level, symbol rate and coding rate etc.

In the next chapter, we will make use of all these techniques on a sensor node in order to achieve coordinated adaptive power management (CAPM).

Chapter 3

Coordinated Adaptive Power Management (CAPM)

In order to have energy optimized sensor nodes a low power hardware design is the basic requirement. Applying sleep scheduling algorithms further energy savings can be achieved. Coordinated Adaptive power (CAP) management is implied during ON periods of the sensor nodes so that energy consumption is further reduced and lifetime can be increased. CAP management is a technique in which processing rate of a computation unit and transmission rate of a communication unit are changed together as per the actual number of packets waiting in the input buffer for the service. As discussed in Chapter 2, technique used to vary the processing rate of a computation unit is DVFS and technique used to vary the transmission rate of a communication unit is DMS. CAP management not only results in energy optimized processing and transmission but also reduces the packet loss which may result due to limited buffer size in Sensor Nodes. This chapter introduces the concept of CAP management.

3.1 Introduction

Different strategies can be adapted for energy efficiency where main objective is to reduce the power consumption but at the cost of performance degradation. This performance degradation may be in the form of speed of processing as in the case of DVFS, may be in the form of data transmission rate as in the case of DMS or may lead to miss some data of interest as in the case of different sleep scheduling algorithms used for energy saving. If data of interest does not get sensed or sensed data does not get communicated well in time to the server, purpose of sensing is not satisfied.

Data loss in a wireless communication system results in two ways. One possibility is that data is transmitted successfully and traveling on a communication channel data may be lost due to channel impairments, heavy traffic resulting in collisions, network congestion, no connecting node etc. Second possibility of data loss is before data transmission takes place. With a low resource hardware having constraints on its memory and buffer sizes, there is possibility of data loss due to buffer overflow. The tradeoff between packet loss due to buffer overflow and packet loss due to transmission errors has been studied in [136] for the increase in the overall system throughput. Data loss after transmission can be controlled to some extent by using proper modulation technique, channel encoding, error correcting codes etc. Data loss before transmission needs to be taken care by individual node. This kind of data loss is very prominent in the wireless communication networks where each node is highly hardware constrained as well as energy constrained. Wireless sensor network is a good example of this kind of network. Data is lost in the node itself before getting transmitted. Such loss occurs due to the buffer overflow at the output buffer (data loss due to buffer congestion). Data processed by processor comes in the output buffer and waits for getting transmitted. If the channel condition is poor then transmitter will not transmit the data in order to save the retransmissions. If receiver cannot receive the packet due to collision or something else then transmitter does not receive acknowledgement (ACK) from the receiver and needs to keep the transmitted packet stored in the queue for retransmission. In such conditions probability of data loss due to output buffer congestion increases (either tail drop or head drop policy) as the output buffer size is very small in wireless sensor nodes. It not only leads to data loss but also results in CPU power wastage. It is wise to reduce the processing speed and save processor power if the transmission speed has reduced and packets in the output queue are waiting. In [11] effect of network congestion on buffer congestion has been analyzed and shown that reducing the speed of microcontroller during congestion periods can save power. We have discussed the effect of network congestion on buffer congestion resulting in data and power loss. Now consider the situation when there is no network congestion and channel condition is also good. For sensor network applications like monitoring applications most of the time there is no event occurring and there is very less traffic in the network. We call this time period as **normal period**. When event occurs and gets detected by the sensor nodes then the traffic in the network increases. We call this time period as **catastrophic period**. This increased traffic needs to be handled with proper processing and transmission speed otherwise data loss due to buffer overflow will occur. Information about the event of interest though sensed will not reach to the sink node. It may violate the purpose of deploying WSN.

If sensor nodes are always operated in energy saving mode then it may result in data loss due to limited buffer sizes. On the other hand if always operated for worst case condition it will unnecessarily drains the battery energy and the lifetime of the node will be very small. So what is desirable is a delay tolerant but loss intolerant sensor network. This kind of WSN will work with maximum tolerable delay (it will save the power) when there is not much workload. During catastrophic period, sensor nodes have to handle peak workload so work in some other active mode which will consume more energy but ensure no data loss within the node.

With CAP management we are trying to adaptively change the power consumption of sensor nodes. Power consumption of processing unit and communication unit are coordinated together and changed adaptively with respect to the workload. Coordination between processing and communication subunits results in reduced power consumption as well as reduced data loss because of buffer overflow.

3.2 Sensor node architecture

For the purpose of sensing and data forwarding every sensor node is equiped with three main subunits as sensing unit, processing unit and communication unit. All these units are powered using a small battery. Among these units, communication unit is the major energy consuming unit. Different studies have shown that it consumes about 60% of the total energy, processing unit consumes about 30% of the total energy and sensing unit consumes a very little part about 10% of the total energy [137]. So different researchers are working to reduce the energy consumption of communication unit and processing unit in order to save the overall energy consumption of the sensor node and prolong the lifetime of the node. Sensor node architecture have limited size input buffer and output buffer. In input buffer, data arrives from sensors and from receiver. If processing unit does not process this data at a proper rate then input buffer overflow may occur and data loss takes place. Processed data arrives in output buffer which needs to be transmitted. If data arrival rate is higher than the data transmission rate then output buffer overflow occurs resulting in data loss before transmission.

Power consumption varies from one class of sensor nodes to another. Depending on the requirement of application, sensor nodes may contain some additional units like position

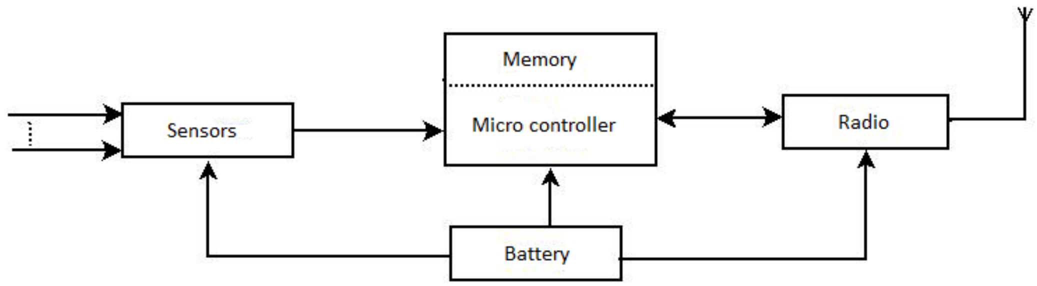


Figure 3.1: Basic blocks of a sensor node

finding system or mobilizer in case of mobile sensor nodes or may have some active sensors like camera. All such additional units consume extra power for their functionality. Mote-class node power characteristics are very much different from that of a Stargate node [138], but if we consider the basic three units of a sensor node (sensors, processor/ controller and radio) following observations hold considerably applicable-

- Wireless communication consumes much higher energy than computation [139].
- Over a short distance communication the radio energy consumption of reception, transmission, and idle states is nearly same. Therefore, the radio should be turned off whenever possible rather than keeping it in idle mode.

Though wireless communication is the major activity which consumes lot of power, power consumption of microcontroller or processor also have significant share in the energy budget of wireless sensor nodes. Emerging real time applications of WSN like body sensor networks need powerful microcontrollers which consume more power [11] [140]. In [141] energy consumption of the widely adopted Mica2 sensor node is studied very accurately and shown that the power consumption of processor ranges from 28% to 86% of the total power consumed and roughly 50% on average. So we can say that-

it is equally important to optimize the power consumption of computing unit along with communication unit.

We are trying to optimize the power consumption of processing unit and wireless communication unit together. The technology that enables to optimize the power consumption of the processing unit dynamically is called DVFS and similar technique for communication unit is DMS. Both these techniques have been discussed in Chapter 2.

3.3 CAP Management

3.3.1 Need

In order to save the overall energy consumption of a sensor node we consider to implement DVFS and DMS on a sensor node. ON time period is divided in to number of time slots of fixed duration.

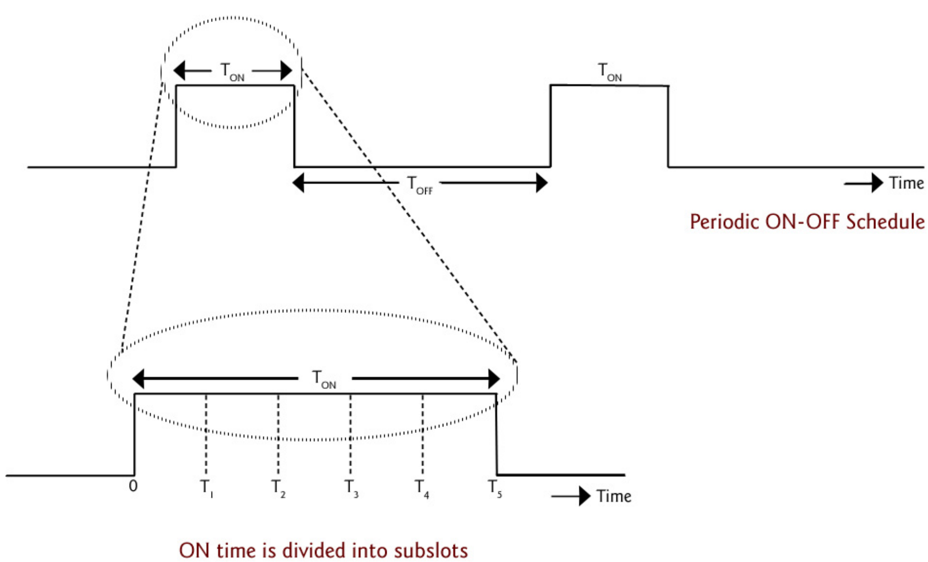


Figure 3.2: ON period divided in to slots

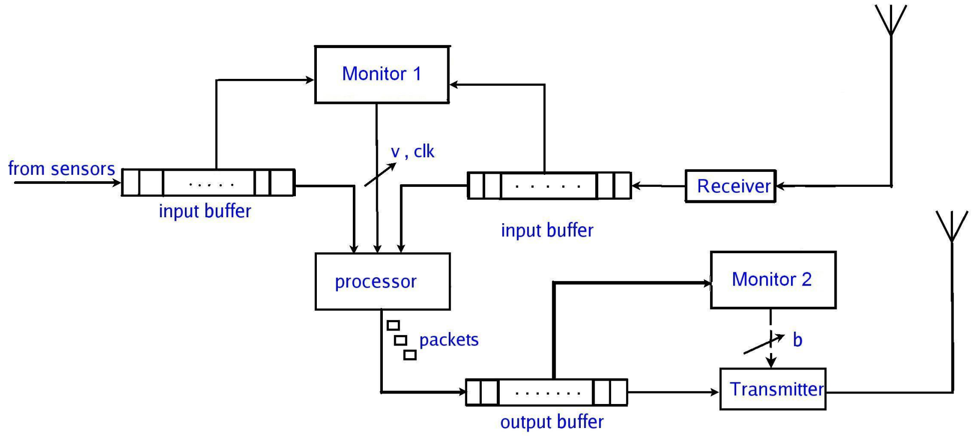


Figure 3.3: Wireless sensor node having DVFS and DMS without coordination

In Figure 3.3 at the start of every time slot status of the input buffer and output buffer are checked independently and accordingly voltage-frequency settings of the processor and modulation index of transmitter are set. In this figure for the ease of understanding we have shown Transmitter and Receiver as two seperate blocks but practically it is the same Tranceiver unit which either acts as transmitter or as receiver at a time. Here as DVFS and DMS work independently on the same sensor node some of the problems that arise are:

- Two different power managers are required one for processing unit and other for communication unit.
- Input buffer and output buffer are separately monitored hence more interfacing with hardware is required.
- More software, more iterations and more energy required.
- DVFS has predictive approach but DMS checks the actual status of the output buffer, so DMS monitor is not aware of what will be the status of output buffer in next slot.
- Most of the time sensor node simply has to forward the data, it does not consume any time for processing and suddenly the data in the output buffer increases.

As DMS modulator is not aware of this fact in advance so monitoring the output buffer, deciding and then adjusting the modulation level for transmission in the same time slot takes time. Meanwhile there is possibility of data loss due to limited buffer size. To overcome above mentioned problems coordination between DVFS and DMS is required. If operating state of communication unit is selected based on the operating state of the processor then both units will work together for power optimization as well as possibility of data loss before transmission gets removed.

3.3.2 Concept

CAP management is a technique to coordinate active operating states of processor and transmitter in a particular time slot adaptively with workload. Dynamic Voltage Scaling (DVS) [142] and Dynamic Modulation Scaling (DMS) [143] techniques are integrated on the node and works in coordination. Idea of coordinated power management was floated by Vijay Raghunathan et al [138] in 2002 which is depicted with the help of figure in the paper as below. We have tried to exploit this concept for coordinating power management

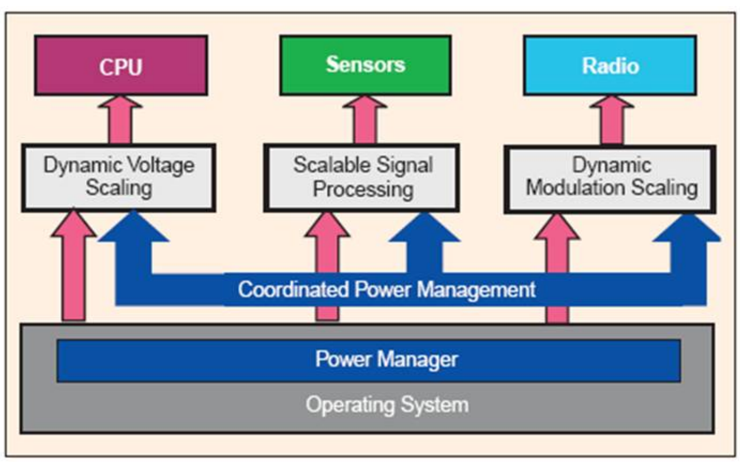


Figure 3.4: Coordinated power management concept as given in [138]

of computing and communication unit with the objective of data loss reduction due to buffer over flow. Hardware limitations and power scarcity are the two main constraints arise because of small size and low cost of the wireless sensor nodes. Making optimum use of these two resources is the main objective of CAP Management which will give the desired QoS. We have made the following assumptions:

- Due to limited energy availability, a short haul multi hop communication is preferred.
- Other than sensing its environment each node acts as a router and simply forwards the received data to other nodes.
- Percentage of data to be forwarded is much greater than the percentage of data actually sensed (border nodes are exception).
- Number of border nodes is very less as compared to the total number of nodes in the network.

Using the fact that if workload monitor observes a heavy workload and selects higher supply voltage and clock frequency for processing predicted heavy load, then packet arrival rate in output buffer will be quite high and if packets are not transmitted quickly then some of the data packets may get lost even before transmission due to limited output buffer size. In this situation modulation scaler selects higher constellation size i.e. selects multiple bits per symbol and fast data transmission takes place, but at the cost of increased energy consumption. If DVFS is selecting smaller value of supply voltage and clock frequency then optimum constellation size b is selected, which results in to comparatively slow data transmission but reduces power consumption.

In the block diagram shown in Figure 3.5 a common workload monitor and predictor controls voltage and frequency scalers for processor and also modulation (Mod) scalar for transmitter. As workload is predicted for next slot, where first data will be processed with

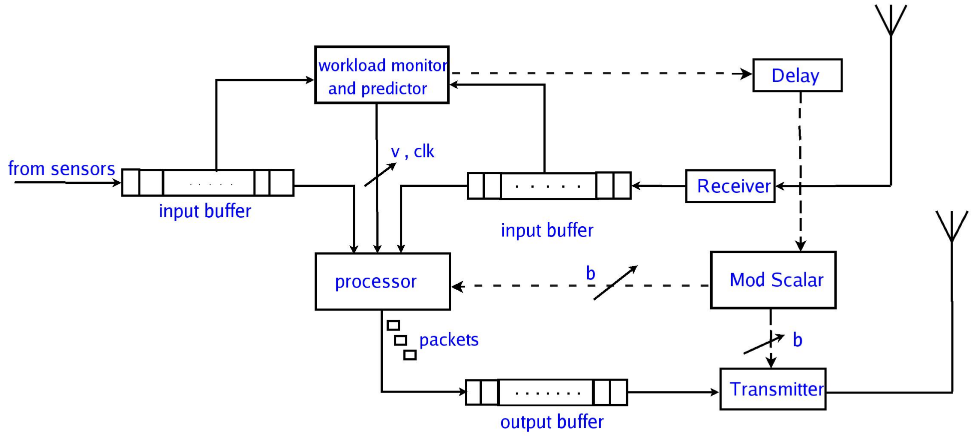


Figure 3.5: Concept of CAP management on Wireless Sensor Node

certain processing rate and then will be made available in output buffer for transmission so a sufficient delay is introduced before giving control signal to modulator. Also constellation size used for modulation is required to be specified in the packet header for the purpose of demodulation at the receiver side.

Use of DVFS and DMS together has been explored in [144] and [145] for minimizing energy consumption. In [144], Kumar et al. addressed a resource allocation problem. simulation results for static and dynamic slack allocation algorithm are given and compared with baseline algorithms and checked for single hop networks. In [145], genetic algorithm is used to solve the convex optimization problem of the energy management. This paper has mainly focused on trade off between computation and communication latencies. Still coordination between DVFS and DMS has not been studied, buffer overflow has not been considered as QoS parameter and also detailed mathematical modeling of sensor node performance has not been found. In [146] [147] authors have combined DVFS and DMS techniques to maximize the battery energy levels of individual nodes at the same time meeting the end to end latency requirements.

From various sleep scheduling algorithms discussed in Chapter 2, it reveals that CAP management has maximum scope of power saving in the scenario where sensor nodes remain ON from its deployment and also in the applications where sensor nodes use time out (periodic) sleep schedule. We have modeled both the scenarios of normal period and catastrophic period using queuing theory and have discussed in next two chapters.

Chapter 4

Tandem Queue Model and Simulation of Sensor Node

In this chapter we will analyze the performance of a wireless sensor node as a tandem queue. Instead of directly modeling sensor node as a single entity, we have tried to capture inside functioning of a sensor node with the help of tandem queue. Two main power consuming units of a sensor node- computation unit and communication unit we have considered as two servers connected in tandem. Here output of computation unit acts as an input for the communication unit. A wireless sensor node with various capabilities like fixed service rate, only DVFS, only DMS, both DVFS and DMS and both coordinated together is considered and performance analysis carried out. MATLAB simulation of a wireless sensor network node is carried out to check and compare the performance of sensor node with various capabilities.

4.1 Tandem queue model of sensor node

Single server, exponential queuing system with finite buffer capacity ($M/M/1/K$ model) has been well established in literature [148] [149]. From the sensor node architecture (see Figure 3.1) it is possible to model individual sensor node as two servers with finite queues, connected in tandem (output of the first becomes input for the second). Equivalent tandem queue model is shown in Figure 4.1. In this model first server refers to the computing

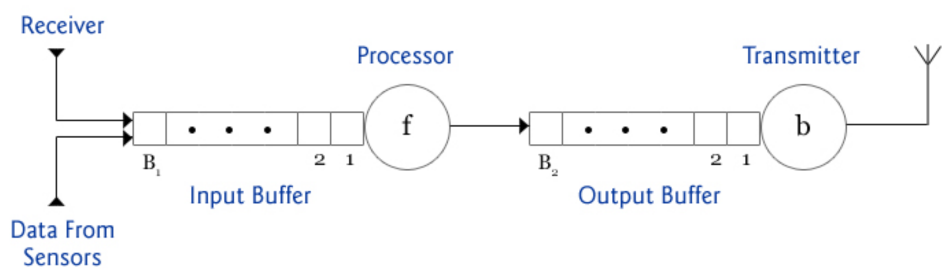


Figure 4.1: Tandem queue model of Wireless Sensor Node

unit (processor or microcontroller) and second server is a radio transmitter. Data sensed by various sensors and data received by the receiver comes in at the input buffer and gets processed by the computation unit. Processed data is temporarily stored in output queue before transmission. Input and output buffers are very small in size as wireless sensor nodes are very small in size. We have considered periodic rotational sleep schedule (described in Chapter 2) for sensor nodes as it provides better synchronization, field coverage and network connectivity. Sensor nodes wake up (ON state) for a fixed duration and then go to sleep (power saving state) for a fixed duration. We refer to sleep state as OFF state. Duration of ON state and OFF state are predefined such that required low duty cycle operation can be achieved. Sensor nodes keep on changing between OFF state and ON state. Duty cycle varies application to application and also depends on the sensor node density in the network. Higher the duty cycle more stringent is the need for power optimization.

Let

A_n = number of packets arrived during n^{th} slot $\quad 0 \leq A_n \leq 6$

B_1 = maximum input buffer size $\quad B_1 = 6$ packets

B_2 = maximum output buffer size $\quad B_2 = 6$ packets

f = service rate of first server (depends on clock frequency)

b = service rate of second server (depends on transmission bit rate of transmitter)

Here service rate implies the maximum number of packets that can be served in one time slot.

M_n = input buffer occupancy at the end of n^{th} slot $\quad 0 \leq M_n \leq B_1$ packets

N_n = output buffer occupancy at the end of n^{th} slot $\quad 0 \leq N_n \leq B_2$ packets

We have considered Late Arrival System (LAS) where data packets are allowed to enter in the system just before the slot ends. These packets get service in the next time slot. Input buffer occupancy at the start of a time slot is dependent on buffer occupancy at the start of previous slot, number of packets served and number of packets arrived during previous slot.

$$M_n = min\{max\{(M_{n-1} - f),\ 0\} + A_{n-1}, \quad B_1\} \tag{4.1}$$

Similarly output buffer occupancy can be written as,

$$N_n = min\{max\{(N_{n-1} - b),\ 0\} + min\{M_{n-1},\ f\}, \quad B_2\} \tag{4.2}$$

As discussed in chapter 2, DVFS and DMS are power optimization techniques used for processor and transmitter respectively. Here we will implement these techniques one by one and then together on a sensor node tandem queue model to study effect on buffer congestion (overflow) and power consumption.

4.1.1 Wireless sensor node with fixed service rates (No DVFS, No DMS)

In most of the WSN applications during normal periods very less workload needs to be handled. When the number of packets in the buffer is less than the number of packets that can be served in one time slot of duration Δt, server remains idle for some period. We can define I_{1n} as the Idle period of first processor in the n^{th} time slot and can be given as,

$$I_{1n} = max\{(f - M_{n-1})/f, 0\} \cdot \Delta t \tag{4.3}$$

Similarly, we can define I_{2n} as the Idle period of second processor in the n^{th} time slot and can be given as,

$$I_{2n} = max\{(b - N_{n-1})/b, 0\} \cdot \Delta t \tag{4.4}$$

For a fixed service rate sensor node, M_n is a function of service rate and arrival rate. As service rate is fixed M_n depends on arrival rate only. During normal period, arrival rate is very small which keeps the value of M_n also small and as a result processor remains idle over a longer period. On the contrary during catastrophic conditions the arrival (number of packets arrived in one slot) increases but as the service rate is fixed and buffer size is small possibility of data loss due to buffer congestion (buffer overflow) occurs as per the head drop or tail drop scheme. Input buffer overflow (OV_1) occurs when $M_n = B_1$ and output buffer overflow (OV_2) occurs when N_n = B_2.

$$OV_{1n} = max\{(max\{(M_{n-1} - f),\ 0\} + A_{n-1} - B_1),\ 0\} \tag{4.5}$$

$$OV_{2n} = max\{(max\{(N_{n-1} - b),\ 0\} + min\{M_{n-1}, f\} - B_2),\ 0\} \tag{4.6}$$

For example, consider a sensor node with fixed serice rate f= 0.5 f_{max} and b= 0.5 b_{max} i.e. f = b = 3 packets per time slot. $B-1$ = B_2 = 6 packets. During normal time period, packet arrival rate is very small, if it is just 1 packet per time slot then from above equations we can find out that sensor node remains idle for 66.66% of the time slot and there is no buffer overflow. But when we consider catastrophic time period (A_n = 5 packets per time slot), only 3 packets can be served by the first processor and the same 3 packets can be handled by the second processor. So there is no output buffer overflow but packets get rejected at input buffer resulting in 3 packets overflow in each time slot.

4.1.2 Wireless sensor node with only DVFS, No DMS (variable f and fixed b)

In this case input buffer status depends on the arrival rate as well as service rate f. Arrival rate cannot be controlled but service rate can be used as a control knob to reduce idle period during normal times and buffer overflow during catastrophic period. In this case

$$M_n = min\{max\{(M_{n-1} - f_{n-1}),\ 0\} + A_{n-1},\ B_1\} \tag{4.7}$$

This value of M_n will decide the value of service rate f_n. Smaller the value of M_n smaller value of f_n will be selected. Reduction in service rate will reduce the power consumption (smaller value of f will need lesser supply voltage and reduce power consumption) and will take more time to complete the service (DVFS). It helps to reduce the idle power wastage. Idle period of first processor in n^{th} time slot can be given as,

$$I_{1n} = max\{(f_n - M_{n-1})/f_{max},\ 0\} \cdot \Delta t \tag{4.8}$$

By reducing the value of f_n idle time will be reduced and will save the power. Buffer overflow is given as,

$$OV_{1n} = max\{(max\{(M_{n-1} - f_{n-1}),\ 0\} + A_{n-1} - B_1),\ 0\} \tag{4.9}$$

During catastrophic condition as the arrival rate increases value of M_n will be more. Data loss due to input buffer overflow can be reduced by increasing the value of f_n. In order to make service rate buffer adaptive we need to scale f_n in terms of M_n.

$$f_n = (M_{n-1} \cdot f_{max})/B_1 \tag{4.10}$$

f_{max} is the maximum supported service rate. But as the second server in the tandem queue (transmitter) works with fixed service rate, there is possibility of data loss during catastrophe and more power wastage during idle period. Output buffer occupancy is-

$$N_n = min\{max\{(N_{n-1} - b),\ 0\} + min\{M_{n-1},\ f_n\}, B_2\} \tag{4.11}$$

Output buffer overflow can be written as,

$$OV_{2n} = max\{(max\{(N_{n-1} - b),\ 0\} + min\{M_{n-1},\ f_n\} - B_2),\ 0\} \tag{4.12}$$

In this equation f_n is varying at the first server but there is no control knob at the second server to control the overflow. During catastrophe as A_n increases, f_n at the first server will increase resulting in increased N_n. As b is constant and B_2 is fixed, output buffer overflow increases. It not only results in data loss but as the processed data gets lost, processing power used for that also goes waste. Similarly during normal conditions as A_n reduces, f_n will be reduced. It will reduce the packet arrival rate in the output buffer but as second server works with fixed rate (which is high enough to handle worst case condition), it will remain idle over longer duration and more power will be wasted. Idle period of transceiver in n^{th} time slot can be given as,

$$I_{2n} = max\{(b - N_{n-1})/b,\ 0\} \cdot \Delta t \tag{4.13}$$

N_{n-1} becomes smaller due to reduced f_n but b is constant and moderately high hence I_{2n} increases. Implementation of only DVFS is not enough as it increases the processed data loss and processing power loss during catastrophe and more idle power wastage during normal period.

For example, consider a sensor node with variable f_n (DVFS) and fixed b= 0.5 b_{max} (no DMS). During normal time period, packet arrival rate is very small, if it is just 1 packet per time slot then from above equations, value of f_n selected will be 1 packet per time slot and hence I_1 = 0 but second server remains idle for 66.66% of the time slot and there is no buffer overflow. But when we consider catastrophic time period (A_n = 5 packets per time slot),f_n will be set to 5 packets per time slot and hence OV_1 = 0 but second server works with constant service rate and hence can not handle the increased value of N_n and results in the buffer overflow.

4.1.3 Wireless sensor node with only DMS, No DVFS (variable b and fixed f)

In this case service rate of processor (f) is fixed and kept considerably high in order to handle sufficient number of packets during catastrophic conditions. Transmission rate b can be varied as per the number of packets in the output buffer. During normal conditions f will be much higher than the arrival rate A_n so the first server- processor remains idle over a longer duration. Idle power wastage is more likely in the first server. Number of packets entering in the output buffer is also small during normal conditions. Second server-transmitter will select lower service rate b for the transmission of packets. Lowering the transmission rate will reduce the power consumption (RF power with DMS). Similarly during catastrophic conditions A_n will increase and if f is not sufficiently high then the data loss may occur at input buffer. At the output buffer data loss due to buffer congestion is reduced by increasing the transmission rate (more number of bits per symbol).

$$OV_{2n} = max\{(max\{(N_{n-1} - b_n),\ 0\} + min\{M_{n-1},\ f\} - B_2),\ 0\} \tag{4.14}$$

Here though b_n is changing, OV_{2n} gets limited by f which is fixed. So implementation of only DMS is also not enough.

For example, consider a sensor node with fixed f = 0.5 f_{max} (no DVFS) and variable b_n (DMS). During normal time period, packet arrival rate is small but value of f is fixed, 3 packets per time slot and hence I_1 = 0.66 of total time slot but second server can adust it's serice rate to 1 packet per slot and idle time period can be reduced to zero and also there is no buffer overflow. But when we consider catastrophic time period (A_n = 5 packets per time slot), fixed value of f will make the number of packets arriving in the output buffer constant and hence though second server is capable of increasing the service rate, it will work with the fixed rate (equal to the service rate of first server) and hence $OV_2 = 0$ but all packets will not be allowed to enter in the sensor node due to input buffer overflow.

4.1.4 Both DVFS and DMS integration (variable b and f)

From the discussions above it is highly desirable to have both f and b changing w.r.t. the number of packets in the buffer waiting for the service. Figure 4.2 shows the sensor node architecture with variable processing rate and variable transmission rate. Here a monitor checks the queue length and the probability of buffer overflow. Processing rate of the processor is varied as per the principle of dynamic voltage/frequency scaling (DVFS) and the data transmission rate is varied using dynamic modulation scaling (DMS). For DVFS input buffer is monitored while for DMS output buffer is monitored.

For example a sensor node with both DVFS and DMS capabilities can reduce the service rate of both the servers during normal period and will reduce idle time periods and will increase the service rates during catastrophic period and will reduce data loss due to buffer overflows. Same results are checked with MATLAB simulation and presented in the next section.

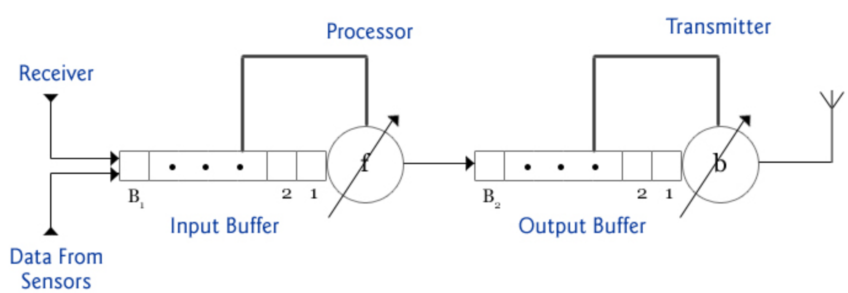


Figure 4.2: Sensor node model with DVFS and DMS

As seen earlier input and output buffer occupancies can be given as,

$$M_n = F\{A_n,\ f\} \qquad 0 \leq M_n \leq B_1 \tag{4.15}$$

$$N_n = F\{f,\ b\} \qquad 0 \leq N_n \leq B_2 \tag{4.16}$$

for the stability of the system it is required to have $f \geq A_n$, so that the departure rate of the first server is nothing but its arrival rate A_n. So we can approximate,

$$N_n = F\{A_n,\ b\} \qquad 0 \leq N_n \leq B_2 \tag{4.17}$$

This shows that the occupancy of input buffer as well as output buffer is a function of arrival rate A_n. Implementing DVFS (on processor) and DMS (on transmitter) together on a sensor node makes the service rates f and b to change w.r.t. input and output buffer occupancy respectively. This will save power during normal periods and will reduce data loss due to buffer overflow during catastrophic periods. Also as both the buffer occupancies are function of arrival rate A_n (directly proportional) there is no need to monitor input and output buffers separately. By monitoring input buffer only it is possible to select required f and b. Now we can say that f and b are changing in coordination.
We implemented both DVFS and DMS together on a sensor node and carried out MATLAB similation to study the input and output buffer lengths. Figure 4.3 and Figure 4.4 shows input buffer length (q1) and output buffer length (q2) at various time slots.

These figures show that whenever input queue length increases, output queue length also increases (not necessarily by same amount) but after some time interval (time delay). This time delay may be of one time slot or the processing time required for the task. These figures support the idea of monitoring only input buffer and varying service rates of both the buffers with some delay in between. It removes the need for observing output buffer

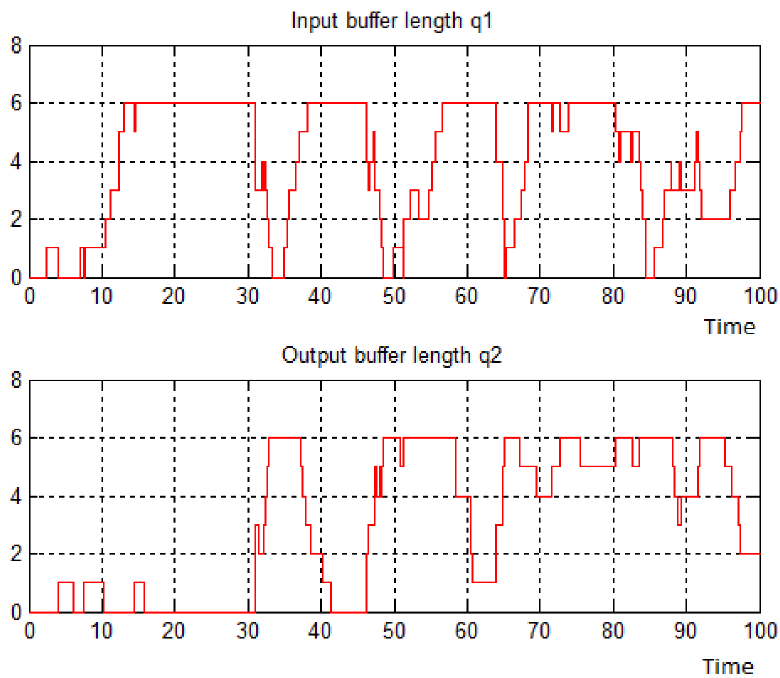


Figure 4.3: Input and output buffer lengths during time slots 1 to 100

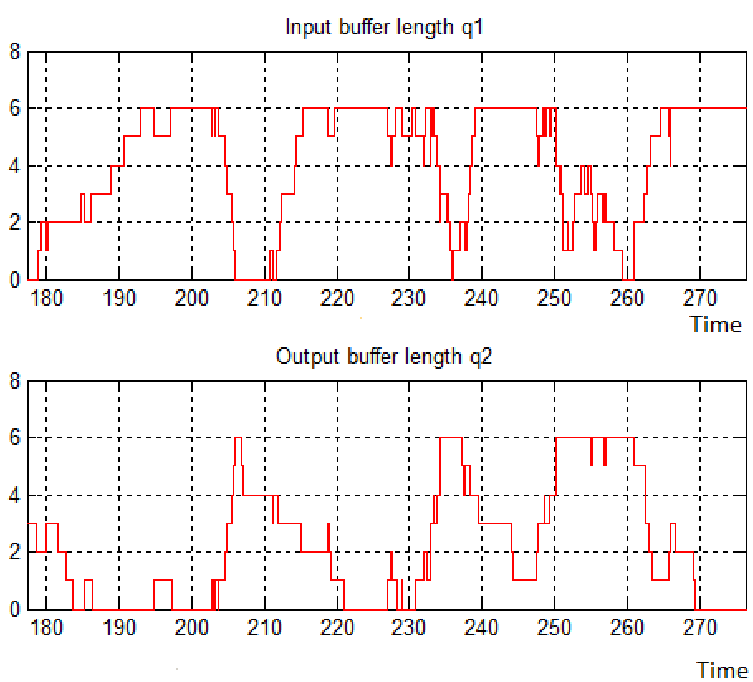


Figure 4.4: Input and output buffer lengths during time slots 170 to 270

seperately. It results in coordinated adaptive power (CAP) management giving extended lifetime to the sensor nodes and indirectly contributing to the lifetime extension of WSN. Figure 4.5 shows the concept of CAP management where processing rate and transmission rate are varied by monitoring input buffer only and without the need to monitor output buffer. It also ensures QoS by reducing the buffer overflow and data loss because of it.

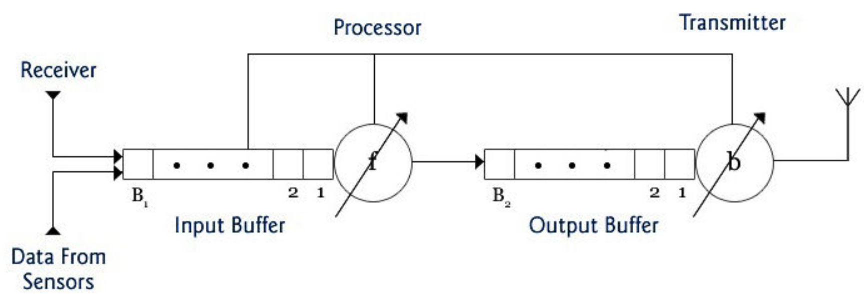


Figure 4.5: Coordinating DVFS and DMS

4.2 MATLAB simulation of tandem queue model

Tandem queue equivalent model of wireless sensor node we simulated in MATLAB 6.5. Each server is having a limited capacity buffer with capacity $B1$ and $B2$ respectively. f_1 and b_1 are the service rates of server1 and server2 respectively in active low state while f_2 and b_2 are the service rates in active high state. λ is the arrival rate of requests in the system. It is considered that the arrival and service rates are Poisson, capacity of buffers is finite and here considered 6 packets. We simulated the models of sensor node with fixed service rate, sensor node with only DVFS capacity, sensor node with only DMS capacity and sensor node with DVFS and DMS applied together. Figure 4.6, Figure 4.7 and Figure 4.8 shows the comparison of a sensor node with fixed service rate, having only DVFS implemented, with only DMS implemented and finally both DVFS and DMS integrated together on a wireless sensor node.

4.2.1 Simulation results for tandem queue model of sensor node

A sensor node is desired to be capable of handling heavy traffic without much loss of information as well as the power optimized one in order to have longer life. A compromizing service rate of 0.5 ($f_1 = f_2 = 0.5$ and $b_1 = b_2 = 0.5$) is selected for both the servers. Effect of varying data arrival rate is analyzed in terms of overflow probability and expected idle time period. Results obtained with Matlab simulation are as in Table 4.1.

Table 4.1: Performance of fixed service rate sensor node for various data arrival rates

Arrival rate	**Idle period**		**Overflow Prob.**		**Queue length**		**Lifetime**
λ	I_1	I_2	OV_1	OV_2	q_1	q_2	**(units)**
0.2	0.58	0.61	0.0015	0.0022	0.88	0.7204	29162
0.3	0.4010	0.4211	0.0125	0.0106	1.2	1.3	20283
0.4	0.2527	0.2596	0.033	0.042	2.2	2.01	19307
0.5	0.1306	0.1953	0.081	0.075	2.8	2.8	15324
0.6	0.0609	0.1780	0.25	0.06	4.02	2.6	15120
0.7	0.0421	0.1740	0.324	0.09	4.6	2.55	15080
0.8	0.0232	0.1561	0.4040	0.1117	4.7998	2.72	15016
0.9	0.0127	0.1341	0.436	0.107	5.15	3.16	14906

From the observation table 4.1, we can divide the traffic in two categories—normal (λ less than 0.5) and catastrophe (λ is 0.5 and above).

During normal data arrival:

- Sensor node remains idle over large period (25% to 58% of ON time). Obviously idle power wastage is more.

- Overflow probabilities (OV_1 and OV_2) remains well within tolerance limit (below 10%).
- Average queue length below 2 (one third of maximum buffer capacity of 6).

During catastrophic data arrival:

- Idle period minimized (6% and below).
- Overflow probability at input buffer (OV_1) increases much beyond the tolerance limit (above 25%) but OV_2 is within limit.
- Average queue length q_1 increases sharply.

From these findings we can think of reducing the service rate during normal period to reduce the power consumption and idle period power wastage. It will save the power and also maintains the QoS paramter (overflow probability). Similarly during catastrophic period, need to decrease data loss by increasing the service rate. It will consume more power but QoS will be maintained.

Table 4.2 shows the simulation results for a sensor node. First row shows the result for a fixed service rate sensor node as the service rates during normal and catastrophic period are same. Other observation rows show the results for a sensor node with different service rates in normal and catastrophic period. We have checked the performance with various service rates by keeping the normal arrival rate 0.2 and catastrophic arrival rate 0.8. Also we ensured to have the same catastrophic duration in all cases.

Figure 4.6 shows the overflow probabilities of input and output buffers (OV_1 and OV_2) under catastrophic conditions. During catastrophic period data arrival rate suddenly increases. If this increased data arrival is not handled with proper service rate then data loss occurs due to buffer overflow.

Table 4.2: Performance parameters observed by simulation

Normal		Catastrophic		Idle period		Normal		Catastrophic		Normal		Catastrophic		Lifetime
f_1	b_1	f_2	b_2	I_1	I_2	OV_1	OV_2	OV_1	OV_2	P_1	P_2	P_1	P_2	(units)
0.55	0.45	0.55	0.45	0.64	0.58	0.002	0.002	0.44	0.24	0.06	0.03	0.16	0.08	34620
0.3	0.3	0.8	0.8	0.37	0.40	0.03	0.02	0.32	0.14	0.032	0.032	0.41	0.35	48260
0.3	0.3	0.9	0.9	0.36	0.40	0.02	0.01	0.30	0.17	0.032	0.029	0.44	0.33	44083
0.3	0.3	1	0.8	0.36	0.40	0.02	0.02	0.24	0.13	0.033	0.027	0.61	0.33	39483
0.35	0.35	0.85	0.85	0.45	0.47	0.01	0.01	0.28	0.16	0.032	0.030	0.39	0.035	44044
0.4	0.4	0.9	0.9	0.48	0.50	0.01	0.04	0.23	0.16	0.034	0.035	0.43	0.42	38504
0.4	0.4	1	0.9	0.48	0.50	0.01	0.00	0.21	0.13	0.037	0.037	0.66	0.46	31401

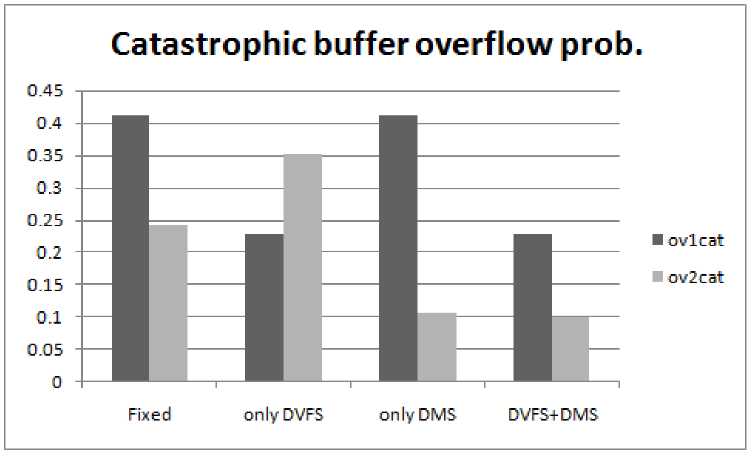


Figure 4.6: Buffer overflow probabilities during catastrophe

A sensor node with fixed service rate do not have any control knob and hence it is not possible to control buffer overflow. A sensor node with only DVFS facility implemented on it can control the overflow probability of input buffer by increasing the clock frequency of the first server (processor). Increased service rate of the first server increases the data arrival rate in the output buffer but since second server works with fixed service rate overflow probability of the output queue (OV_2) increases. This situation is highly undesirable as the processed data gets lost, power used for processing that data also goes waste, which a highly power constrained sensor node can not afford. It is not advisable to have only DVFS implemented on sensor node. Similarly when only DMS is implemented, it results in decreasing the output buffer overflow probability at the cost of increased power but as the first server works with fixed service rate, input buffer overflow can not be controlled. Integrating both DVFS and DMS on a sensor node results in controlling the overflow probabilities of both the buffers. Though more power is consumed by DVFS and DMS during catastrophe by working with higher service rates data loss due to buffer overflow is reduced and can be kept within the tolerance limit which is the highest priority QoS parameter during catastrophe.

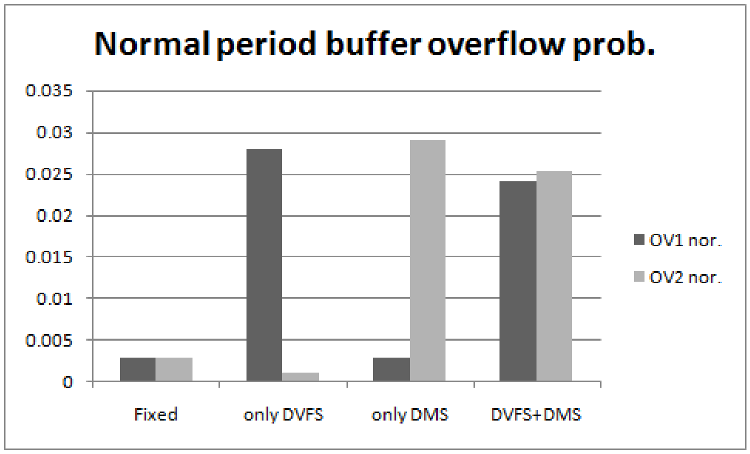


Figure 4.7: Buffer overflow probabilities during normal period

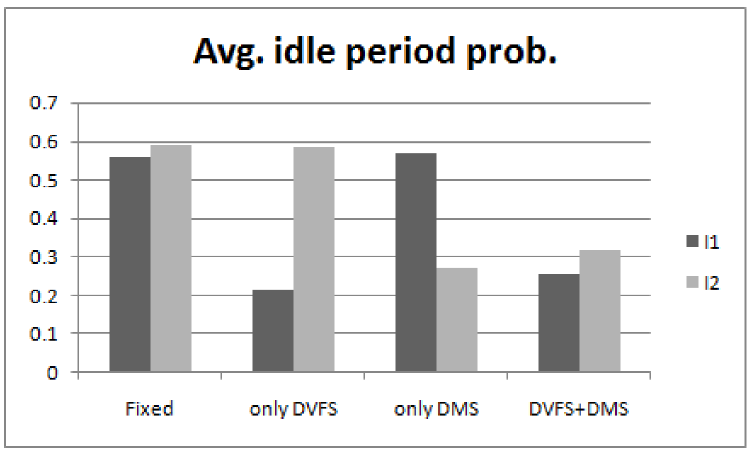


Figure 4.8: Average idle time probabilities

In Figure 4.7 overflow probabilities of input and output buffers are compared during normal period and in Figure 4.8 average idle time probabilities of both the servers are shown. As the data arrival rate is very small during normal periods, buffer overflow possibilities are negligible but possibilities of servers remaining idle are more. More the idle period, more is the power wastage. So for power constrained wireless sensor nodes power saving becomes highest priority QoS parameter during normal periods by reducing idle period.It is achieved by reducing the service rates of the servers. Fixed service rate sensor nodes are designed to handle worst case conditions and hence their service rates are set quite high. During normal periods these servers remain idle most of the time and large amount of power is wasted but there is negligible chance of buffer overflow. When only DVFS is implemented then first server is able to reduce its service rate (processing speed) so that it consumes less power and also reduces idle period by working over a longer period. It slightly increases the overflow probability of the input buffer but which is fairly within the tolerance limit. Similarly having only DMS will reduce the idle time and power consumption of second server with little increase in the overflow probability, which is acceptable. Having both DVFS and DMS reduces the power consumption of both the servers also reduces their idle periods and hence results in power saving. This saved power can used during catastrophe periods to reduce data loss due to buffer overflows. As compared to fixed service rate sensor node lifetime increase of 15% was seen when only DVFS was implemented on a sensor node while implementing only DMS it was 17.5% but DVFS and DMS together applied on a sensor node resulted in 27.22%.

Figures 4.9 and 4.10 shows the snapshots of the screen showing MATLAB simulation graphs under normal and catastrophic periods.

4.3 Threshold determination

As few discrete service rates are supported by the servers, it becomes important to select a particular service rate by comparing the data in the buffer with the threshold. For convenience we consider a node which has two ON states- ON_1 and ON_2. In ON_1 state service rate is higher and the power consumed is more while in ON_2 state service rate is lower as well as power consumed is less but more service time is required. For switching the state between ON_1 and ON_2 a threshold is set and comparing the number of packets in the buffer with set threshold a decision is made whether to serve packets with ON_1 or

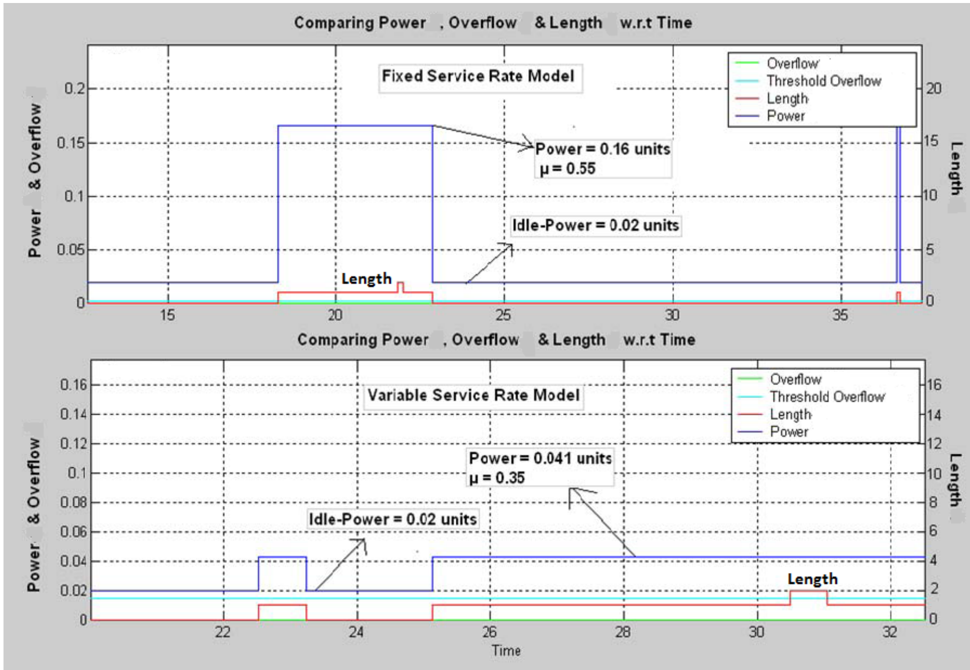


Figure 4.9: Comparison between fixed service rate and variable service rate sensor node in normal time period

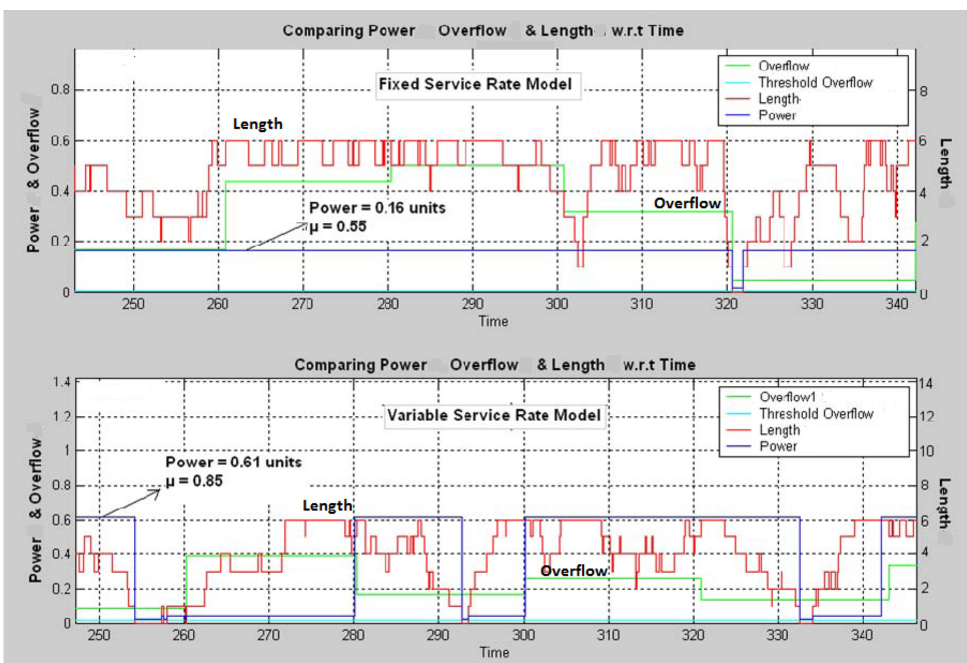


Figure 4.10: Comparison between fixed service rate and variable service rate sensor node in catastrophe period

ON_2 state. Selection of buffer threshold affects the performance of sensor node with finite buffer size. Here performance metric used is lifetime of sensor node and data loss due to buffer overflow. Various threshold policies have breen studied in the literature. Threshold policy is used to choose the better one between two or more possibilities.
Threshold policies are used for selecting a particular data rate at wireless switch or router in order to increase the throughput [150] [151]. Adaptive threshold policy has been used in [152] but the aim of using adaptive threshold policy is to change the transmission rate with time variable channel conditions. Threshold policies are also used for memory sharing between various sources as per their priority. In this paper we consider buffer threshold policy which enables the sensor node to vary its service rate w.r.t. buffer occupancy so that data loss due to buffer overflow can be minimized along with power optimization resulting in the increased lifetime of the node.

4.3.1 Threshold policies

Various threshold policies used for switching between two states are as below:

1. **Single threshold policy:** For a finite sized buffer, a threshold value is set such that whenever the number of packets in the buffer is more than the threshold k, server switches to the high service rate consuming more power but reduces the possibility of data loss due to buffer overflow. On the other hand if the number of packets in the buffer are less than or equal to the threshold k, server continues in the low power state consuming less power and working with low service rate. It reduces the idle period and power consumption but increases the possibility of data loss due to buffer overflow.

2. **Dual threshold policy:** Single threshold policy is suitable when there is not much variation in the traffic arrival. For random traffic arrival we may need to switch the service rates in consecutive time slots. Switching the state consumes extra time and extra energy which may over weigh the power saved. It is better to have some margin before switching to another state. It helps in reducing switching energy and switching time overheads. In dual threshold policy two thresholds k1 and k2 have been defined such that k2 $>$ k1. At the start of a new slot if number of packets in the buffer is more than k1 and less than k2 then server will continue with the service rate as in the previous slot. Service rate will change only when buffer occupancy is outside the margin.

3. **Adaptive threshold policy:** In this policy the threshold level is varied w.r.t. the value of buffer overflow in previous slot along with the buffer occupancy. Buffer overflow value is stored in a overflow register. If the buffer overflow is approaching near the tolerable value then the threshold is decreased so that server can enter in high service state and buffer overflow possibility is reduced. On the other hand if the buffer overflow in the previous slot is negligible then the threshold value can be raised so that server remains in low power state and power is saved.

Figure 4.11 elaborates various threshold policies along with the pseudo codes.

Table 4.3: Threshold effect on lifetime and buffer overflow with fixed threshold

Threshold		**Avg. idle time**		**Normal**		**Catastrophe**		**Lifetime**
k1	**k2**	I_1	I_2	OV_1	OV_2	OV_1	OV_2	**Time units**
1	1	0.4267	0.5428	0.01	0.0098	0.285	0.105	15000
2	2	0.43	0.47	0.0112	0.0113	0.3	0.2233	19900
3	3	0.42	0.44	0.03	0.02	0.41	0.35	24400
4	4	0.3789	0.4313	0.018	0.02	0.3828	0.315	33641
5	5	0.3692	0.3928	0.017	0.02	0.35	0.18	43900

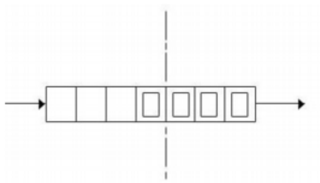
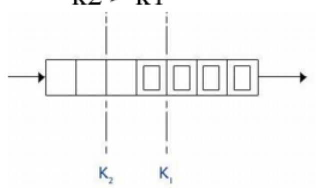
<table>
<tr><th>Single threshold policy</th><th>Dual threshold policy</th><th>Adaptive threshold policy</th></tr>
<tr>
<td>if buff > k
serve_new = mu2;
else
serve_new = mu1;
end

</td>
<td>if buff > k2
serve_new = mu2;
else if buff < k1
serve_new = mu1;
else
serve_new = serve_prev;
end
k2 > k1

</td>
<td>if ov1>=0.02
k1=k1-1;
else if ov1<=0.01
k1=k1+1;
else
k1=k1;
end
if ov2>=0.02
k2=k2-1;
else if ov2<=0.01
k2=k2+1;
else
k2=k2;
end</td>
</tr>
</table>

Figure 4.11: Various threshold policies

Table 4.3 lists the effect of varying threshold levels on the life time of sensor node as well as buffer overflow probability during normal period as well as during catastrophic period. It has been observed that for higher threshold levels lifetime of the sensor node is more than that for lower threshold levels. This increase in lifetime has been achieved at the cost of data loss during catastrophic periods which is not at all desirable. With higher threshold levels sensor node remains in low power state for a longer duration and saves the power. On the other hand with small threshold levels very frequently sensor node switches to high power state giving service with increased speed so more power is consumed but data loss gets reduced. selection of threshold levels is a trade off between lifetime and data loss.

In the Table 4.4, Fixed (3,3) indicates the threshold values for input and output buffer are 3 and 3 and remains fixed. Adaptive (3,3) indicates the starting threshold values are 3 and 3 but adaptively changes with buffer overflow and buffer occupancy.

Figure 4.12 shows comparison of various performance parameters of a sensor node using fixed threshold policy and adaptive threshold policy.

Table 4.4: Comparison of fixed threshold effect with adaptive threshold effect

Threshold	**Avg.idle time**		**Normal**		**Catastrophe**		**Lifetime**
(k1,k2)	I_1	I_2	OV_1	OV_2	OV_1	OV_2	(Time units)
Fixed (3,3)	0.3906	0.4351	0.0186	0.0261	0.3048	0.3245	24660
Adaptive (3,3)	0.3344	0.3470	0.0382	0.0257	0.2852	0.1989	58361
Fixed (4,4)	0.3919	0.4041	0.0167	0.0159	0.4170	0.3133	34163
Adaptive (4,4)	0.3350	0.3709	0.0276	0.0162	0.3043	0.1813	63481

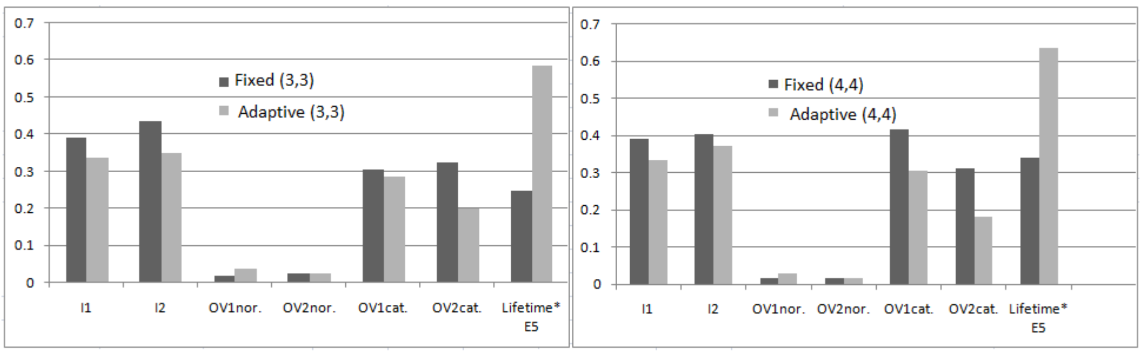


Figure 4.12: Comparison of a sensor node using fixed threshold policy and adaptive threshold policy

Simulation results show that the lifetime of sensor node can be increased by using adaptive threshold policy as compared to that with fixed threshold policy. It also reduces idle time period and power loss during it. Overflow probability during catastrophic periods can be reduced further by selecting next higher service rate along with adaptive threshold.

In the next chapter, we will discuss the bulk arrival Markov chain model of a sensor node with two service rates. There, sensor node will be a single entity having DVFS and DMS coordinated within it.

Chapter 5

Rate Adaptive Wireless Sensor Node

In Chapter 4 we have considered a tandem queue model of wireless sensor node and analyzed it for various capabilities. Results show that implementation of DVFS and DMS together gives better power opimization with QoS ensured. Also it has been shown that there is no need to monitor input buffer and output buffer seperately as both the buffers are correlated with each other. Now in this chapter we will consider a wireless sensor node as a single entity with input buffer. Inside sensor node, microcontroller and transmitter work in a coordinated manner. Changing the service rate of sensor node will change the processing rate as well as transmission rate. A Markov chain model of rate adaptive sensor node is presented in this chapter.

5.1 Concept of working vacation

In standard queuing models with vacation policy a server is allowed to take single or multiple vacations. During vacation period server does not provide the service. Vacation policies are considered in order to save power if there is no data to be served or in order to control the topology. Sleep scheduling policies discussed in Chapter 2 are nothing but the vacation policies. We have considered time periodic rotational sleep schedule in order to have better network coverage and network connectivity at all time instants. Here our aim of introducing vacation policy is to save the power by allowing sensor node to enter sleep (vacation) state. Further we are trying to save the power by introducing working vacations during ON time of the sensor node. During working vacation, a node do not enter the sleep state but it works slowly (lower service rate) and consumes less power. This is exactly what we want to do with a rate adaptive sensor node. The concept of working vacation was introduced by Servi and Finn [153] for M/M/1 queue. Later many other researchers analyzed various Markovian and non Markovian queuing models with multiple working vacations [154–157]. The bulk input queue with multiple working vacations have been analyzed in [158–160].

5.2 Bulk arrival Markov chain model of state dependent rate adaptive wireless sensor node

We have considered the fact that workload (number of packets in the buffer) for wireless sensor nodes is not uniform over the time period. Here it is to be noted that workload is a function of data arrival rate. For rare event detection applications, during normal period data arrival rate is very small but during catastrophic (event occurance) period suddenly the data arrival rate increases. It is very important to save power during normal period in order to increase the lifetime of the sensor node and to reduce the data loss during catastrophic period in order to maintain QoS. A state dependent service rate adaptive sensor node has been modeled using discrete time Markov decision process. We have considered a time out rotational sleep schedule. It operates sensor nodes with low duty cycle in order to save power and extends the lifetime of the sensor nodes. Here the fact exploited is that there are number of sensor nodes deployed in close proximity so whenever a node enters sleep node there are other nodes available to take care of sensing and data forwarding. Our problem is focused on power optimization and QoS provided during ON time of wireless sensor nodes. We have divided the ON time interval in to discrete number of equal time slots. A LAS (Late arrival system) has been considered so the data arrival in the buffer takes place at the end of time slot as shown in Figure 5.1. At the beginning of new slot, the

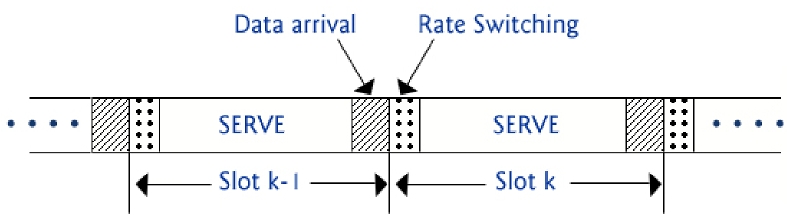


Figure 5.1: Late arrival system

state of FIFO (First In First Out) buffer is checked and depending on the buffer occupancy a service rate decision is made for the next slot. During switching time and arrival time no service has been provided. Now we will consider only input buffer and sensor node as single server inside which processor and transmitter works in a coordinated manner as in Figure 5.2. Service rate of sensor node is changed as per the input buffer occupancy.

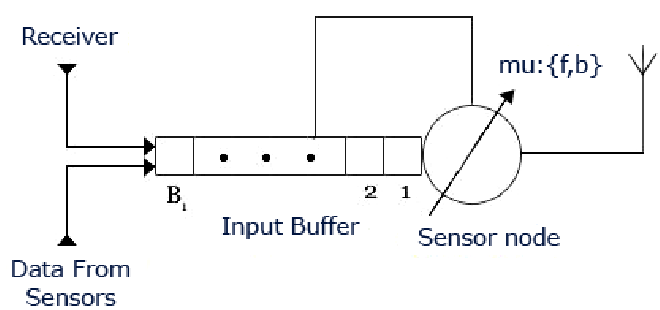


Figure 5.2: Coordinated Rate Adaptive model of Wireless Sensor Node

Here by the service rate μ (mu) of sensor node we mean the coordinated service rates of processor and transmitter. For every value of μ there exists a pair of f and b or we can say that there exists a look up table which gives the values of μ and corresponding f and b.

Figure 5.3 is the flowchart which elaborates the rate switching process for a sensor node with two active states (two service rates). A sensor node is capable to work only if sufficient battery energy is available otherwise a sensor node is considered as dead. When a sensor node is turned ON, it sets the slot counter, buffer threshold and buffer overflow probability threshold. ON time and OFF (sleep) time of a sensor node are divided in to number of equal time slots. As time out sleep schedule is used, a sensor node remains in ON state for a predefined number (M) of time slots and after that enters in to sleep state. For a adaptive buffer threshold policy, initially buffer threshold (K) is set to half the maximum buffer size (N), for a specific application it's tolarable buffer overflow probability (probability of data loss due to packet drop) is given ($OV_allowed$) and we have set the buffer overflow threshold (OV_TH) to half the $OV_allowed$. Here we have considered the overflow probability of the output buffer. Data arrival takes place at the end of time slot (LAS) and the buffer occupancy (Q_length) is checked at the start of next time slot, if it is less than K then Active Low state (service rate R_1, power consumption P_1) is selected otherwise Acitve High state (service rate R_2, power consumption P_2) is selected and packets stored in the buffer get served accordingly, buffer overflow probability is checked over the slot. If the current slot duration is over, slot counter is decremented by one and if not equal to zero then based on the overflow probabilty in the previous slot new buffer threshold is set and again comparing the Q_length with new threshold K node enters either in active ow or active high state and continues in the loop till the slot counter becomes zero and enters in the sleep state.

Figure 5.4 shows a Markov chain which elaborates the rate switching decision process by monitoring the input buffer. Here we have considered two active states: active low (L)

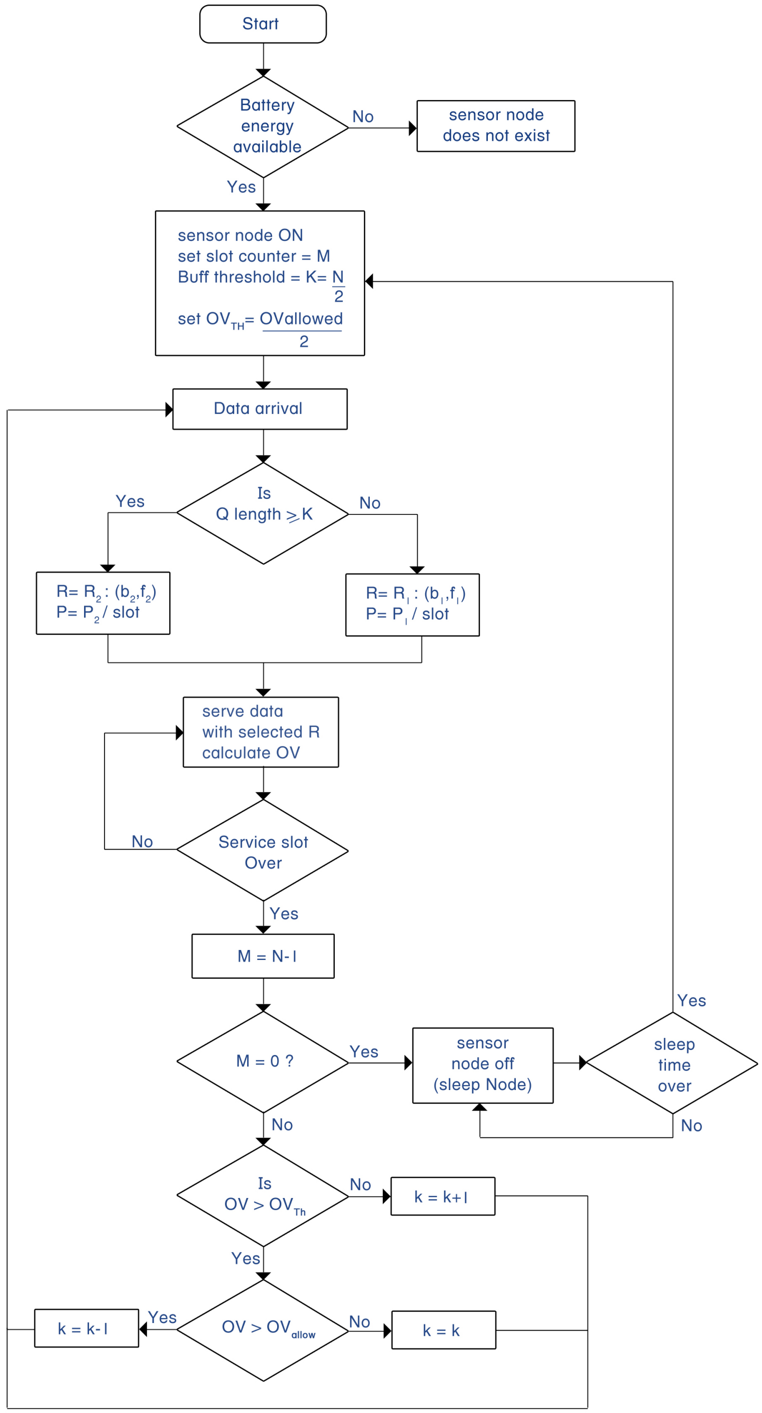


Figure 5.3: Flowchart for a two service rate capability sensor node

and active high (H). In Active Low state, service rate and power consumption is lower but service time is more while in active high state, service rate and power consumption is higher but service time is less. Also we have considered rate switching states (state when a node is changing it's state from active low to active high and vice versa) in between two active states. During switching (transition) states no service has been provided but power is consumed. We have considered a bulk arrival system where the number of packets in the bulk varies from 0 to K. Maximum K number of packets can arrive in each arrival time and K is an integer. The state of a sensor node is depicted by a tuple $\{n, T, R\}$. Where–

n = number of packets available in the system $\quad 0 \leq n \leq N$

N = maximum number of packets that can be stored (size of buffer)

T =Transition state

$$T = 0 \qquad \textit{no transition}$$

$$T = 1 \qquad \textit{state transition}$$

R =service rate

$$R = 0 \qquad \textit{low service rate}$$

$$R = 1 \qquad \textit{high service rate}$$

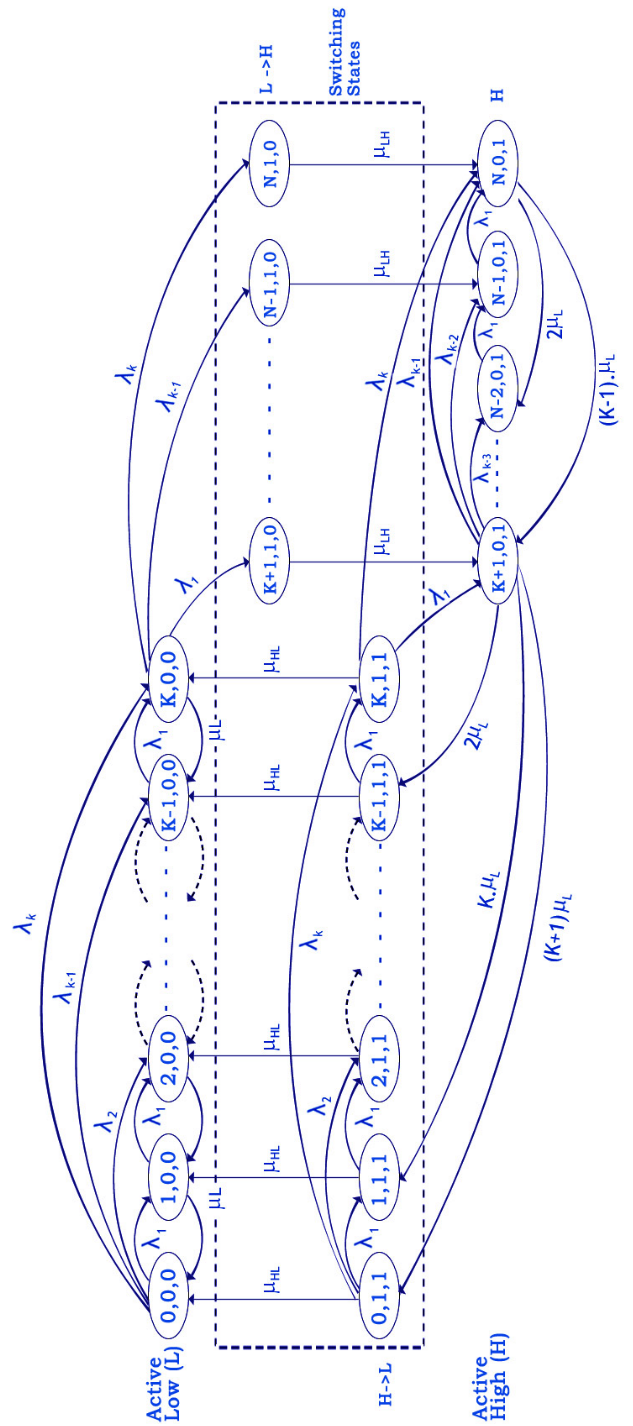


Figure 5.4: Markov chain model of a rate adaptive wireless sensor node

$\{n,0,0\}$ represents the node is in active low state $\quad 0 \leq n \leq K$

$\{n,0,1\}$ represents the node is in active high state $\quad K+1 \leq n \leq N$

$\{n,1,0\}$ represents the low rate to high rate switching states $\quad K+1 \leq n \leq N$

$\{n,1,1\}$ represents the high rate to low rate switching states $\quad 0 \leq n \leq K$

where K is the buffer threshold. Generally K=$N/2$. Also here we have considered $K = W$ i.e. maximum batch size.

We have achieved the discrete time policy in this model by not changing the service rate as soon as the buffer length crosses the threshold on either side till the start of next time slot. No arrival occurs during service time. If node is working in active low state then it will be in the same state till the end of service interval but at the end of arrival interval there is possibility that it may remain in the same state or go to transition state if the number of packets after arrival period are greater than K. From low state, a node enters in “low-to -high“ transition state only after arrival period and there is no possibility of any arrival or service during transition state. If node is found in the transition state during switching interval then state transition takes place with rate μ_{LH}. Now server works in the active high state where the service rate is some integer multiple of the service rate in active low state. Depending on the service rate selected at the end of service interval a node may be in the active high state or may be in high to low transition state. Service interval is followed by arrival interval so there is possibility of data arrival during high to low transition states but not during low to high transition states. Data arrival may change the state from high to low transition back to the active high state. After arrival interval if node remains in transition state the state is changed from active high to active low with transition rate μ_{HL}. Here we have considered that the service interval is sufficient to transmit one packet with low service rate so with high service rate at least two packets (integer multiple) will be served in the same service duration. As we have considered bulk arrival system the probability of arriving k packets during arrival interval is given by f_k where $0 \leq\ k \leq K$.

Now we are going to model a coordinated rate adaptive wireless sensor node. As we have considered a separate service interval and arrival interval in each time slot though the data arrives serially packet by packet, sensor node see the data arrival as bursty arrival as it only checks the data arrived at the end of arrival interval. Here size of the burst (X) is variable. During normal period, burst size X is very small and during catastrophe it will be high. We consider that during normal period, no burst is there (as X is very small) and data burst occurs only during catastrophe. Bursty data traffic arrival at a node can be modeled using a batch $M^{[x]}/M/1/N$ queue where X is the distribution of packets burst sizes. It assumes that packets arrive in bursts of size X = W packets with probability B(W). The bursts are assumed to have an exponential inter arrival time which is valid when there are large number of sources. We also assume that the service time of each packet is exponentially distributed. If λ_a is the arrival rate of bursts, the expected arrival rate of a burst of size W is simply the probability of the burst B(W) multiplied by the arrival rate.

$$\lambda_b(W) = B(W) \cdot \lambda_a \tag{5.1}$$

The well known $M^{[x]}/M/1/N$ model is not suitable for modeling packet loss in a queue as it assumes that an entire burst of packet arrive at the buffer instantaneously. Practically packets of a burst arrive serially one after the other. For this reason we consider the modified $M^{[x]}/M/1/N$ model similar to [160], [161]. We model the buffer using a Markov process V_k. The state space S of the process is same as that of $M^{[x]}/M/1/N$ where n is the current buffer size and N is the maximum buffer size.

$$n \in \{0, 1, 2, \cdots, N\} \tag{5.2}$$

Each state of the Markov process represents the number of packets in the buffer. When the buffer is in the state V_k = q, a burst arrival and/ or service results in a transition to state V_{k+1}. Equation below defines the transitions and their corresponding rates. If the overall compound arrival rate of bursts of size w is λ then

$$\lambda = \sum_{w=1}^{W} \lambda_b(w) \tag{5.3}$$

where $\lambda_b(w)$ is the arrival rate of a poisson process of batches of size w. Then probability of arrival of batches of size w is given as,

$$C_w = \frac{\lambda_b(w)}{\lambda} \tag{5.4}$$

Here further we will use λ_w to represent $\lambda_b(w)$. Now we will analyse the FSM of Figure 5.4 in two parts–in low state and in high state separately.

1. **In active low state:**

$$\lambda \cdot P_{0,0} = \mu_L \cdot P_{1,0} + \mu_{HL} \cdot P_{0,1} \qquad n = 0 \tag{5.5}$$

$$(\lambda + \mu_L) \cdot P_{n,0} = \mu_L \cdot P_{(n+1),0} + \lambda \cdot \sum_{k=1}^{n} P_{(n-k),0} \cdot C_k$$
$$+ \mu_{HL} \cdot P_{n,1} \qquad \text{for} 1 \le n \le W \tag{5.6}$$

$$\mu_{LH} \cdot P_{n,0} = \lambda \cdot \sum_{k=1}^{W} P_{(n-k),0} \cdot C_k \qquad n > W \tag{5.7}$$

Here we note that probability of being in low to high transition state after threshold (W) is nothing but the probability of being in low state.

$$P_{n,0} = Pr\{n,0,0\} + Pr\{n,1,0\} \tag{5.8}$$

for $n \leq W$, $Pr\{n,1,0\} = 0$ and for $W \leq n \leq N$, $Pr\{n,0,0\} = 0$

$$P_{n,0} = \begin{cases} Pr\{n,0,0\} & \text{if } 1 \leq n \leq W \\ Pr\{n,1,0\} & \text{if } W \leq n \leq N \end{cases} \tag{5.9}$$

2. **In active high state:**

   Here it is to be noted that

$$P_{n,1} = Pr\{n,0,1\} + Pr\{n,1,1\} \tag{5.10}$$

for $W < n$, $Pr\{n,1,1\} = 0$ and for $n \leq W$, $Pr\{n,0,1\} = 0$

$$P_{n,1} = \begin{cases} Pr\{n,0,1\} & \text{if } n > W \\ Pr\{n,1,1\} & \text{if } n \leq T \end{cases} \tag{5.11}$$

It is considered that in active high state packet serving rate μ_H is some integer multiple of serving rate in low state μ_L.

$$\mu_H = X \cdot \mu_L \qquad X \in \{2,3,4,5\} \tag{5.12}$$

Only one value of X being selected at a time which depends on the queue lenght as well as overflow probability in the previous time slot.

$$(\lambda + \mu_H) \cdot P_{n,1} = \mu_{LH} \cdot P_{n,0} + \lambda \cdot \sum_{k=1}^{W} P_{(n-k),1} \cdot C_k$$
$$+ \mu_H \cdot P_{(n+X),1} \qquad W < n < N \tag{5.13}$$
$$\mu_H \cdot P_{N,1} = \mu_{LH} \cdot P_{N,0} + \lambda \cdot \sum_{k=1}^{W} P_{(N-k),1} \cdot C_k \qquad n \leq W \tag{5.14}$$
$$(\lambda + \mu_H L) \cdot P_{n,1} = \lambda \cdot \sum_{k=1}^{W} P_{(n-k),1} \cdot C_k + \mu_H \cdot P_{(n+X),1} \tag{5.15}$$
$$n + X > W \quad and \quad n \leq W$$

μ_L = service rate in active low state

μ_H = service rate in active high state

μ_{LH} = switching rate from low to high state

μ_{HL} = switching rate from high to low state

Switching rate is inverse of switching time required to switch the state. Energy is consumed during switching times resulting in energy overheads so higher switching rates are desired. Switching rates are hardware dependent and vary from one platform to another. Practically values of μ_{LH} and μ_{HL} may differ but we have considered μ_{LH} = μ_{HL} for the purpose of simulation and analysis.

From this $M^{[x]}/M/1/N$ model of rate adaptive wireless sensor node we can analyze the following QoS parameters as,

1. **Power savings:** It gives the energy efficiency of a rate adaptive sensor node with

CAP management scheduling.

$$P_{sav} = \frac{P_H - P_{cons}}{P_H} \cdot 100\% \tag{5.16}$$

Total power consumption P_{cons} of a sensor node is given as-

$$P_{cons} = P_L \cdot \sum_{n=0}^{W} \Pr\{n,0,0\} + P_H \cdot \sum_{n=W+1}^{N} \Pr\{n,0,1\} + P_S \cdot [\sum_{n=W+1}^{N} \Pr\{n,1,0\} + \sum_{n=0}^{K-1} \Pr\{n,1,1\}] \tag{5.17}$$

where

P_L = power consumed in active low state

P_H = power consumed in active high state

P_S = switching power consumed during state transition. $P_S = P_{LH} = P_{HL}$.

2. **Packet loss probability/ buffer overflow probability:** It is nothing but the buffer full probability as the packet loss takes place only when a packet arrives but buffer is already full.

$$P_{loss} = P_{fullbuff} = Pr\{n = N\} = \sum_{T=0}^{1} \sum_{R=0}^{1} \Pr\{N,T,R\} \tag{5.18}$$

3. **Mean packet delay:**From Little's formula

$$\mathbf{E}\{N_u\} = \lambda \cdot \mathbf{E}\{D\} \tag{5.19}$$

$$\lambda_a = \lambda_0 \cdot (1 - P_{loss}) \tag{5.20}$$

Where

$\mathbf{E}\{N_u\}$ = avarage number of packets in the buffer

λ_a = average accepted packet arrival rate

$\mathbf{E}\{D\}$ = average packet delay

λ_0 = average requested packet arrival rate

So average packet delay can be computed using

$$\mathbf{E}\{D\} = \frac{\mathbf{E}\{N_u\}}{\lambda} = \frac{\sum_{n=0}^{N} \sum_{T=0}^{1} \sum_{R=0}^{1} n \cdot \Pr\{n,T,R\}}{\lambda_0 \cdot (1 - P_{loss})} \tag{5.21}$$

4. **Number of state oscillations (State Switching rate):** Total number of state switching during ON time period is defined as number of state oscillations. Though it is a number but measured over the ON time period we can call this as state switching rate. Here state oscillations is nothing but the rate switching as each state has it's specific service rate.

$$N_{osc} = \mu_{HL} \cdot \sum_{n=0}^{K} \Pr\{n,1,1\} + \mu_{LH} \cdot \sum_{n=K+1}^{N} \Pr\{n,1,0\} \tag{5.22}$$

Table 5.1 lists the effect of variable service rate. We carried out discrete time bulk arrival and bulk service model simulation in Virtual lab. Queuing networks and modeling simulation package in virtual lab has been developed by Mathematics Department, IIT Delhi. Here symbol transmission rate is kept constant but by varying the number of bits transmitted per symbol we achieve variable transmission rate. We analyzed catastrophic condition with packets arriving in batches with probability 0.7 and with service probability of 0.9. System capacity is considered to be 6 and service policy used is first in first out (FIFO). Here service probability or departure probability depends on symbol transmission rate and departure batch size is nothing but the number of bits getting transmitted per symbol i.e. constellation size.

Table 5.1: Effect of bulk service on system parameters

Parameter	**batch size=1**	**batch size=2**	**batch size=3**
Mean number of packets in the system	3.488	2.914	2.609
Mean number of packets in the queue	2.631	1.411	1.163
Mean waiting time in the queue	10.00	4.235	3.459
Mean sojourn time in the system	13.257	8.823	7.756
Utilization	0.851	0.764	0.481
Throughput	0.24	0.313	0.3
Blocking probability	0.363	0.190	0.075

From the simulation results we see that with increased constellation size number of packets in the system as well as in the queue decreases and hence the blocking (overflow) probability decreases. It reduces the data loss. Data obtained during catastrophe is the main purpose of deploying a WSN and hence transmitting the data successfully to the sink is very important. This is achieved with extra transmission power. Increased batch size reduces the waiting time and helps to satisfy latency constraint of the application.

Analyzing the Markov chain model of a rate adaptive wireless sensor node, needs to solve number of equations (equations 5.5 to 5.15) for every arrival rate. So analyzing the performance on paper was found very difficult (may be solved using some tool). We changed our approach and developed Generalized Stochastic Petrinet models with all the parameters of rate adaptive sensor node considered in this chapter and using the software tool SHARPE, we analysed the performance of a multirate sensor node, which is discussed in the next chapter.

Chapter 6

Performance Evaluation with Generalized Stochastic Petri Net Model

Continuous Time Markov Chains (CTMCs) may prove tedious, error prone and expensive for complex communication systems. The Generalized Stochastic Petri Net (GSPN) framework [162], [163] provides a powerful set of building blocks to deal with the computational complexities of the state-transition mechanism and event-scheduling mechanism of a discrete event stochastic system. They integrate the essential features of concurrency, synchronization and a sequenced presentation of complex systems. A GSPN is a tool for representing complex CTMC in a concise way, thereby facilitating a graphical visualization of a multi-tasking scenario and a dynamic behavior of packet transmission. We have analyzed various models of wireless sensor node using software analytical tool SHARPE [164]. GSPN is useful to analyse the underlying large Continuous Time Markov Chain model of a sensor node. It helps us to measure performance parameters under various scenarios for solving the actual equations, which otherwise would be a very tedious job.

6.1 Introduction to Petri Nets

- **Petri Net (PN):** A Petri net (PN) is a bi-partite directed graph with two types of nodes called places and transitions, that are connected by directed edges or directed arcs [149] . Arcs exist only between places and transitions, i.e. there is no arc between two places or two transitions. In communication systems, a token is considered as one packet. [165] and [166] are the fundamental references for study of Petrinets.

- **Stochastic Petri Net (SPN):** It is an extension of Petri net (PN) [167], which is a high-level description language for formally specifying complex systems. SPN has been used as a powerful modeling tool in performance, availability and reliability analysis in communication systems [168] as it provides the means for introducing randomness (stochasticity) in the model. Timed transitions are used for random arrival and service provided. Timed transitions provide exponentially distributed service rate. Detailed study of SPN can be found in [169] and [170].

- **Generalized Stochastic Petri Net (GSPN):** It is more powerful version of PN. It can use both immediate transitions as well as timed transitions. It is more powerful as it provides inhibitor arcs as well as guard functions.

The general components of a GSPN are shown in Figure 6.1. Each place may contain an arbitrary number of tokens. The number of tokens in a place is a non-negative integer. For a pictorial presentation, places are depicted as circles, transitions are represented by bars and tokens are represented by dots or integers in the places. Each directed arc in the bi-partite graph is assigned a weight or a multiplicity, which is a natural number. If the multiplicity of an arc is not specified, then it is taken to be unity. For any transition in the PN, arcs directed out of the transition are called as output arcs of the transition. The corresponding places

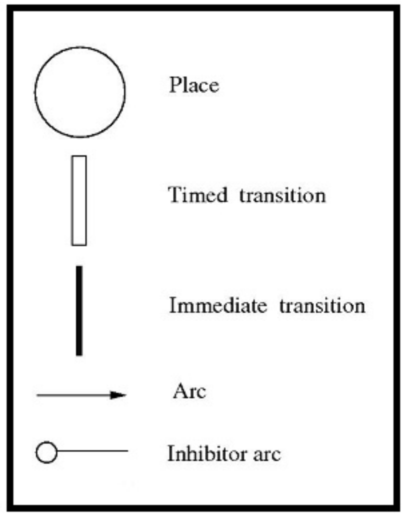


Figure 6.1: Components of GSPN

are called as output places of the transition. For any transition in the PN, arcs directed into the transition are called input arcs of the transition. The corresponding places are called as input places of the transition.

A transition is said to be enabled if all of its input places have at least as many tokens as the multiplicities of the corresponding input arcs.

An inhibitor arc is an undirected arc (between a place and a transition) with multiplicity. If there exists an inhibitor arc with multiplicity k between a place and a transition and if the place has k or more number of tokens, then the transition is inhibited even if it is enabled. When enabled (and not inhibited), a transition can fire. When a transition fires, it removes from each input place, the number of tokens given by multiplicities of the corresponding input arcs, and adds to each output place, the number of tokens given by the multiplicities of the corresponding output arcs.

6.2 Generalized Stochastic Petri Net(GSPN) Model of Wireless Sensor Node

Table 6.1 describes various places and transitions used in the GSPN models of a WSN.

Table 6.2 defines various functions and guard functions used in the GSPN models considered in this section.

6.2.1 GSPN model of a wireless sensor node with fixed service rate

SPN models has been developed and analyzed for energy optimization or energy estimation of energy constrained devices in various papers like [171–176].

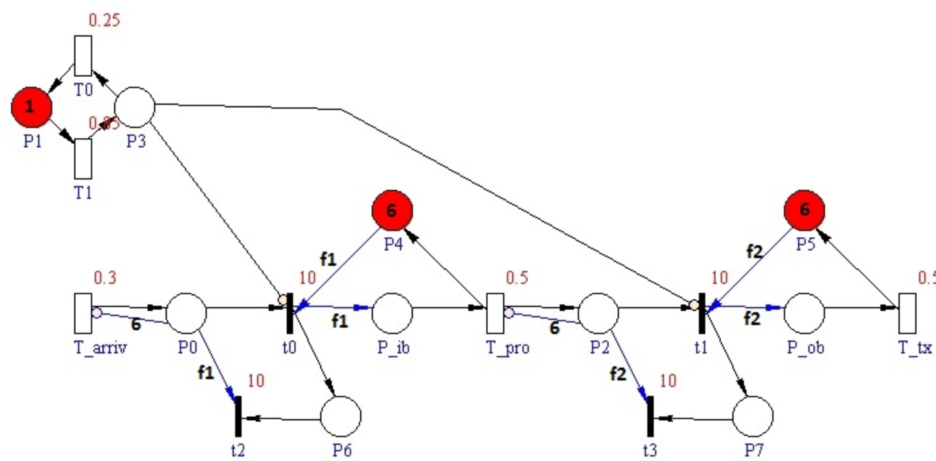


Figure 6.2: Wireless Sensor Node with fixed service rate

Figure 6.2 shows the GSPN model of a sensor node with fixed service rate. It is a tandem queue model of wireless sensor node. We consider data arrives with exponential arrival distribution and arrival rate λ at transition T_arriv. Arrival rate varies from time to time (0.2 to 0.9). Intermediate buffer P0 temporarily holds the arrived data and transfers the bulk of data at the end of time slot to input buffer P_ib. Input buffer and output buffer

Item	**Description**
T_arriv	Exponential data arrival with rate lambda
T_pro	Exponential processing rate (1.0)
T_tx	Exponential transmission rate (1.0)
T_pro1	Exponential low rate processing (0.2)
T_pro2	Exponential high rate processing (0.6)
T_tx1	Exponential low rate transmission (0.5)
T_tx2	Exponential high rate transmission(1.0)
T0	Exponential timed transition to detect end of the slot when data arrival takes place
T1	Exponential timed transition to detect the start of a slot
T2	Exponential timed transition by which sensor node turns ON
T3,T4	Exponential timed transition by which sensor node enters in sleep mode
t0	Immediate transition to move a bulk of arrived data in to input buffer
t1	Immediate transition to move processed data in output buffer
t2	Immediate transition to empty the intermediate buffer when bulk of data gets transferred to input buffer
t3	Immediate transition to empty the intermediate buffer when bulk of data gets transferred to outputput buffer
P0	Intermediate place to hold the arrived data which stores the bulk of data before entering in to input buffer
P1	place to initialise the arrival of data in input buffer
P2	Intermediate place to hold the processed data
P3	place to stop the arrival of data in input buffer
P4	Place to indicate input buffer vacancy, initial number of tokens equals inbuffer size
P5	Place to indicate output buffer vacancy, initial number of tokens equals outbuffer size
P6	place to indicate the arrival of data in input buffer
P7	place to indicate the arrival of data in output buffer
P_ib	Input buffer of WSN, sensed data as well as received data arrives here, finite buffer
P_ob	Output buffer of WSN ,Processed data comes here and needs to be transmitted

Table 6.1: Description of GSPN model items

Function	**Used with transition**	**Description**

f1		if(#(P0) <#((P4))
		#(P0)
		else
		#(P4)

f2		if(#(P2) <#((P5))
		#(P2)
		else
		#(P5)

[g1]	T_pro1	if(#(P_ib) <4)
		1
		else
		0

[g2]	T_pro2	if(#(P_ib) >3)
		1
		else
		0

[g3]	T_tx1	if(#(P_ob) <4)
		1
		else
		0

[g4]	T_tx2	if(#(P_ob) >3)
		1
		else
		0

[g7]	T2	if(#(P1) >0 or #(P3) >0)
		0
		else
		1

Table 6.2: Various functions used in GSPN model

capacity considered is of six tokens. Data is transferred to input buffer only when there is vacancy in it. Place P4 indicates the vacancies in input buffer and similarly P5 indicates vacancies in output buffer. We are using these places to measure overflow probability. Initially 6 tokens are deposited in P4 and P5 which indicates initially both the input and output buffers are empty. Data in input buffer gets exponential service. In case of fixed service rate model, there is only one processing transition T_pro and only one transmission transition T_tx, both are with transition rates of 0.5.

Functions f1 and f2 are used to check the number of vacancies in input buffer and output buffer respectively and accordingly transfer the tokens from intermediate buffers P0 and P2. It captures the bulk arrival of data as considered in the previous chapter. Place P6 is used as an indicator that data batch has been transferred to input buffer and initiates the immediate transition t2 to remove the same batch of tokens from place P0. Same function is performed at output buffer using P7 and t3.

Places P1, P3 along with transitions T0 and T3 forms ON-OFF timer as we have selected a sensor node to work with periodic sleep schedule. A token is initially deposited in place P1, when transition T1 gets fired, token from P1 is removed and deposited in place P3. As soon as a token comes in P3, no data arrival or service takes place and sensor node enters in to sleep state. When transition T0 fires, token is removed from P3 and comes in P1 and sensor node is again turned ON. Transition rates of T0 and T1 determines the sleep time and ON time duration. Transition rates of T0 and T1 can be adjusted to get required duty cycle.

6.2.2 GSPN model of a wireless sensor node with only DVFS

The method is composed of several activities Figure 6.3 shows the GSPN model of sensor node with only DVFS capability but with fixed transmission rate. In this model depending on the number of tokens in P_ib service is given by either T_pro1 or T_pro2. We have used guard functions [g1] and [g2] for this purpose. Description of guard functions is given in Table 6.2. For the purpose of monitoring P_ib and accordingly selecting the transition T_pro1 or T_pro2, ON time period is divided in equal time slots. At the start of every slot, input buffer is monitored and a particular transition selected for service. For this purpose, we need two timers in this model- one ON-OFF timer and other slot timer. Places P1, P3 along with timed transition T0, T1 forms a timer which divides the ON time duration

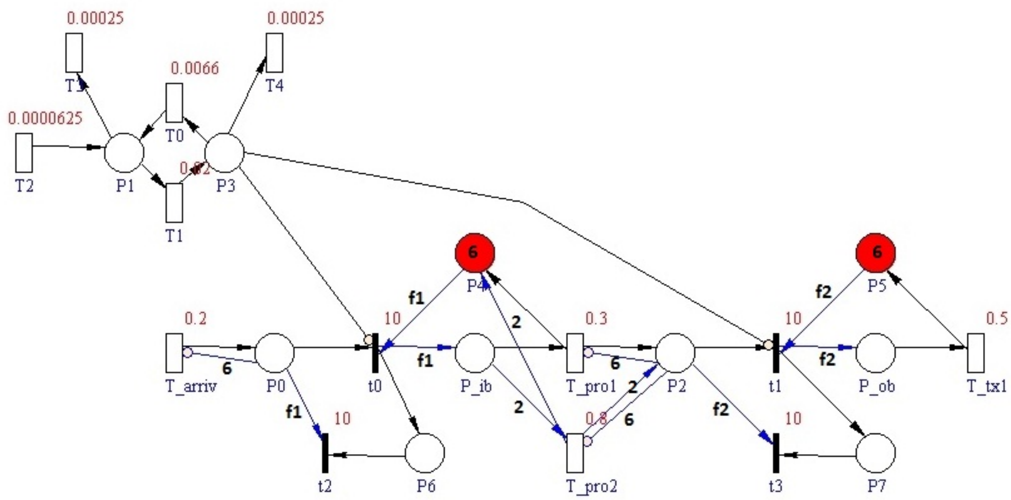


Figure 6.3: Wireless Sensor Node with only DVFS

into number of time slots. Similarly timed transitions T2, T3 and T4 constitute ON-OFF timer. Though timers are used with deterministic time settings, for the purpose of analysis we have considered it exponential. Whenever transition T2 fires a token arrives at P1 and enables the bulk data deposition in input and output buffer. When transition T1 fires, token is removed from P1 and deposited in P3. It disables the immediate transitions t0 and t1 so that now data can not enter in input and output buffers but the data earlier entered, gets serviced. It helps to capture the LAS (Late Arrival System) model discussed earlier. Whenever ON time duration is over, token is removed from the timer using T3 and T4 and sensor node enters in sleep state. If sleep period is over then again a token arrives in P1 and sensor node turns ON. Arrival of token in P1 is controlled using guard function [g7] implemented on transition T2. Transition rates of T0, T1, T2, T3 and T4 are adjusted such that ON:OFF ratio is 1:4 for low duty cycling and ON period divided in 20 time slots. Insertion of ON-OFF timer enables the working of sensor node during ON period and disabled during OFF period. It helps to increase the lifetime of WSN by saving energy during OFF period.

6.2.3 GSPN model of a wireless sensor node with only DMS

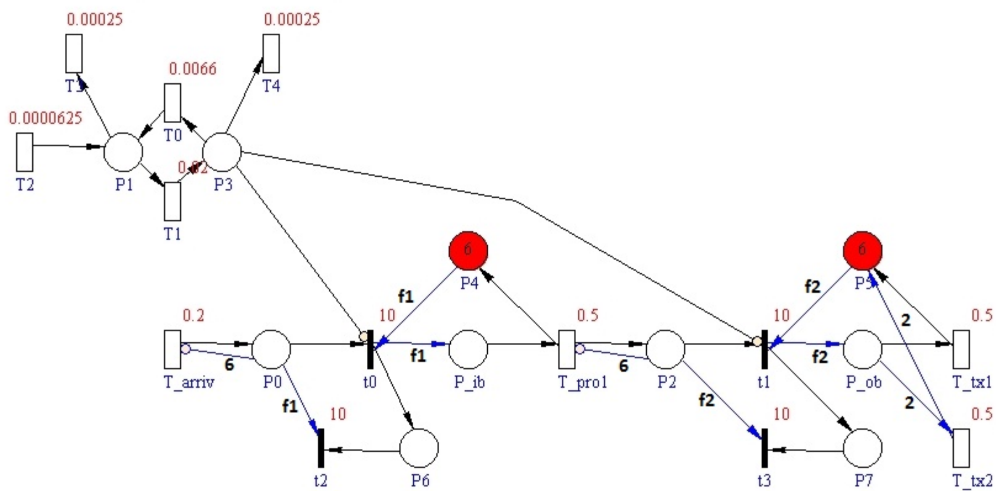


Figure 6.4: Wireless Sensor Node with only DMS

Figure 6.4 shows the GSPN model of sensor node with only DMS capability but with fixed processing rate.

This model is same as that with only DVFS but with the difference of having only transition after P_ib and having two transitions after P_ob. Two transitions T_tx1 and T_tx2 provides two transsmission rates. Though the transition rates of both T_tx1 and T_tx2 are same (0.5), input arc multiplicities are different. It captures the DMS concept of having constant symbol rate but different number of bits per symbol, resulting in variable data transmission rate. In this model depending on the number of tokens in P_ob, service is given by either T_tx1 or T_tx2. We have used guard functions [g3] and [g4] for this purpose. Description of guard functions is giveen in Table 6.2. ON-OFF timer and slot timers used here are same as in the previous model for only DVFS.

6.2.4 GSPN model of a wireless sensor node with DVFS and DMS coordinated (CAPM)

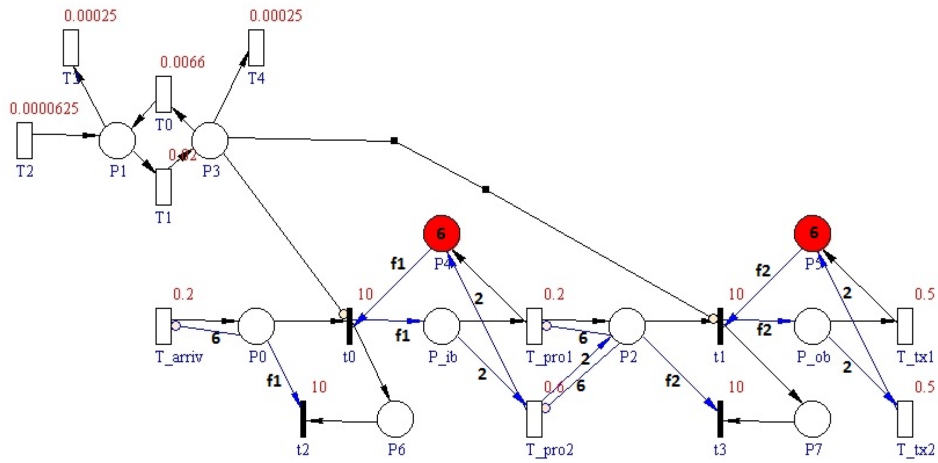


Figure 6.5: Wireless Sensor Node with coordinated DVFS and DMS

Figure 6.5 shows the GSPN model of sensor node with multirate (both DVFS and DMS implemented) capabilities. Here sensor node is capable to process the data at two different rates as well as to transmit the data at two different rates. This model is combination of two models described earlier- with only DVFs and with only DMS. Functionality of all the components has been discussed earlier. Concepts of bulk arrival, LAS system, guard functions, ON-OFF timer and slot timer are same as we discussed earlier. Now we want to add a battery of finite capacity in this model in order to measure the lifetime of sensor node.

6.2.5 GSPN model of wireless sensor node with battery

In order to measure the lifetime of a sensor node, a battery of fixed capacity has been considered, supplying power for the operation of a sensor node.

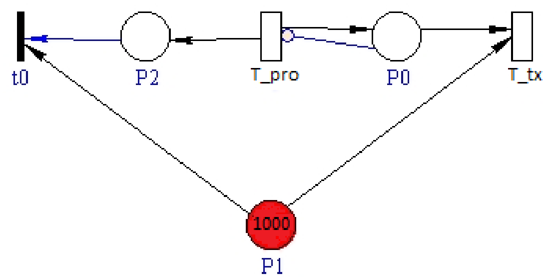


Figure 6.6: Wireless Sensor Node with fixed battery capacity and fixed service rate

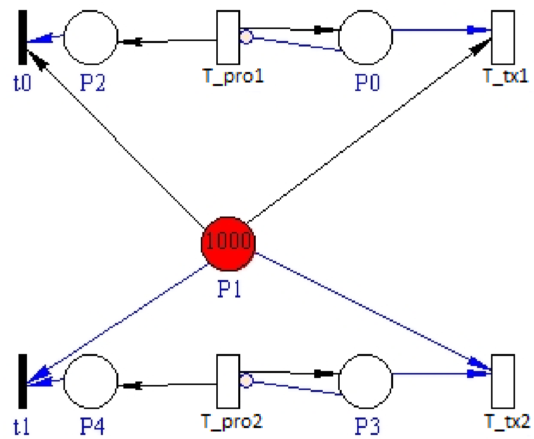


Figure 6.7: Wireless Sensor Node with fixed battery capacity and multiple service rates

Using hierarchical modeling technique, we used the parameters (throughput) obtained from above models and used as inputs for the battery model to compute the lifetime of sensor node. Figure 6.6 and Figure 6.7 shows the GSPN models with fixed capacity battery for a sensor node with fixed service rate and with multiple service rates respectively. Place P_battery is used as a battery in which initially 1000 tokens have been deposited. It indicates initially 1000 power units (say mW) are available for the operation of wireless sensor node. Different amount of power is consumed for serving a packet (here token) with different service rates. Also the power consumed for processing and for transmission is different. In this model whenever a token is being processed or transmitted, it draws a certain number of tokens from the battery. During sleep state tokens are not removed from the battery. Here battery considered is non rechargable. The time, at which all tokens have been removed from the battery, is called the lifetime of the sensor node.

In fixed service rate model, whenever a token is processed and transmitted, it withdraws fixed number of tokens from the battery. In case of multirate model, the number of tokens drawn from the battery depends on the service rate selected. For the processing, power consumed is proportional to the cube of processing rate (as per DVFS) and for transmission, power consumed is proportional to the square of the number of symbols used for transmission(M^2). In this model, number of bits per symbol (b) is given by the cardinality of the arc, which gives us value of $M = 2^b$. For fixed rate model, one power unit consumed for processing of ten tokens and one power unit consumed for transmission of three tokens. In multi rate model, with low processing rate (0.3) power consumption decreases, it becomes 1 power unit for 350 tokens processed and 1 power unit for 20 packets processed with high processing rate (0.8). Accordingly, transmission power also varies.

6.3 Performance measures

1. **Overflow probability (OV_1 and OV_2):** Buffer overflow occurs when buffer is full and another token (packet) arrives. OV_1 is the steady state average probability that input buffer is full and another token arrives. Similarly OV_2 is the steady state average probability that output buffer is full and another token arrives . We have used places P4 and P5 to indicate the number of vacancies in input buffer (P_ib) and output buffer (P_ob) respectively. It means whenever input buffer is full, number of tokens in P0 will be zero. So the probability that P4 is empty gives OV_1. Similarly the probability that P5 is empty gives OV_2.

2. **Idle time probability (I_1 and I_2):** Idle time probability is the probability of having

the server in idle state (there are no tokens to be served). I_1 is the idle probability of first server (processor) and I_2 is the idle probability of second server (transmitter). Probability that P_ib is empty gives I_1 and Probability that P_ob is empty gives I_2.

3. **Average buffer length (q_1 and q_2):** This is the steady state average number of tokens in inbuffer (q_1) and outbuffer (q_2).

4. **Lifetime:** A battery with fixed capacity is used for the functioning of a sensor node. Data processing and radio activity consumes power. This power consumption is a function of service rate provided. Time at which there is no token left in the battery place (battery fully discharged) is the lifetime of WSN.

5. **Average throughput of the sensor node (η):** It gives the average number of tokens served by a transition over the ON time period of a sensor node.

6. **Rate Oscillations:** It is the total number of times rate switching takes place (from low rate to high rate and vice versa) over the ON period of a sensor node.

7. **Average delay:** It is the average delay experienced by a packet, from it's arrival in the input buffer till getting served by the transmitter.

$$Total\ delay\ (D) = \frac{\#(P\_ib)}{\eta_{T\_arriv}} + \frac{\#(P_{\_ob)}}{\eta_{T\_pro1} + \eta_{T\_pro2}} + \frac{1}{\eta_{T\_tx1} + \eta_{T\_tx2}} \tag{6.1}$$

where, #(P_ib) indicates the number of packets in the input buffer and #(P_ob) indicates the number of packets in the output buffer

6.4 Sensitivity analysis

Using GSPN model of Wireless Sensor Node, we are able to measure the overflow probability (OV_1 and OV_2), idle time probability (I_1 and I_2), average buffer length (q_1 and q_2), lifetime, average throughput of the sensor node, average delay etc. We have considered input buffer and output buffer of capacity six, data arrival rate follows Poisson distribution and service time distribution is exponential.

In all GSPN models discussed in this chapter, dark circles indicate that initially some fixed number of tokens are deposited in those places. If these places have input arc as well as output arc, then tokens deposited in that place keeps on rotating and generally such places are used to indicate the buffer occupancy or to fire some other transitions at certain time intervals. If a dark circle has only output arc, then that place indicates a kind of reservoir, in our model a battery with fixed capacity. Monitoring of this place is important as the functioning of remaining model depends on the availability of tokens in this place.

Table 6.3 shows the performance measures of a sensor node with fixed service rate (see Figure 6.2).

Table 6.4 and Table 6.5 shows the analytical results obtained for a wireless sensor node with only DVFS capability (Figure 6.3) and with only DMS capability (Figure 6.4) respectively.

Table 6.6 shows the analytical results obtained for a wireless sensor node with CAP management technique (coordinated DVFS and DMS) as shown in Figure 6.5.

Table 6.3: Performance parameters of fixed service rate sensor node observed by SHARPE analytical tool

Arrival	**Queue length**		**Idle prob.**		**Overflow prob.**		**Avg.Throughput**		**Avg. Utilization**		**Delay**	**Lifetime**
λ	q_1	q_2	I_1	I_2	OV_1	OV_2	n_{T_pro}	n_{T_tx}	U_{T_pro}	U_{T_tx}	D	(units)
0.2	0.237	0.229	0.8203	0.8203	0.00096	0.00015	0.1796	0.1796	0.1755	0.1754	7.0099	11000
0.3	0.3938	0.3846	0.7329	0.7329	0.00153	0.00078	0.2605	0.2605	0.26	0.26	6.6358	7350
0.4	0.5828	0.576	0.6554	0.6551	0.0036	0.00349	0.3448	0.3448	0.345	0.344	6.01223	5800
0.5	0.83	0.832	0.5727	0.572	0.00841	0.0116	0.4272	0.4272	0.4272	0.424	5.947	5300
0.6	1.131	1.156	0.4943	0.4943	0.0182	0.0305	0.5056	0.5056	0.505	0.498	6.143	3400
0.7	1.477	1.55	0.422	0.422	0.0348	0.0663	0.5773	0.5773	0.577	0.562	6.5271	1320
0.8	2.612	2.504	0.298	0.316	0.0733	0.0479	0.6839	0.6839	0.64	0.613	8.388	860
0.9	3.597	3.254	0.198	0.25	0.09	0.106	0.7449	0.7449	0.692	0.651	9.684	520

Table 6.4: Performance parameters observed by SHARPE analytical tool for a sensor node with only DVFS

Arrival	**Queue length**		**Idle prob.**		**Overflow prob.**		**Avg.Throughput**			**Avg. Utilization**		
λ	q_1	q_2	I_1	I_2	OV_1	OV_2	n_{T_pro1}	n_{T_pro2}	n_{T_tx}	U_{T_pro1}	U_{T_pro2}	U_{T_tx}
0.2	1.1528	1.340	0.4482	0.5166	0.0185	0.0414	0.1461	0.0387	0.241	0.4871	0.0645	0.4833
0.3	1.645	2.164	0.322	0.3470	0.0435	0.0922	0.1638	0.0791	0.3260	0.5460	0.1318	0.6251
0.4	2.015	2.749	0.2516	0.2504	0.0723	0.1381	0.1653	0.1182	0.3747	0.5513	0.1969	0.7495
0.5	2.290	3.123	0.2106	0.1972	0.1008	0.1757	0.1607	0.1519	0.4013	0.5359	0.2533	0.8027
0.6	2.50	3.359	0.1854	0.1674	0.1270	0.2032	0.1543	0.1800	0.4163	0.5145	0.3000	0.8325
0.7	2.663	3.511	0.1690	0.1497	0.1505	0.2231	0.1478	0.2029	0.4251	0.4926	0.3382	0.8502
0.8	2.793	3.614	0.1578	0.1386	0.1712	0.2376	0.1417	0.2217	0.4306	0.4725	0.3695	0.8613
0.9	2.897	3.687	0.1499	0.1312	0.1894	0.2485	0.1364	0.2371	0.4343	0.4547	0.3953	0.8687

Table 6.5: Performance parameters observed by SHARPE analytical tool for a sensor node with only DMS

Arrival	**Queue length**		**Idle prob.**		**Overflow prob.**		**Avg.Throughput**			**Avg. Utilization**		
λ	q_1	q_2	I_1	I_2	OV_1	OV_2	n_{T_pro}	n_{T_tx1}	n_{T_tx2}	U_{T_pro}	U_{T_tx1}	U_{T_tx2}
0.2	1.121	0.908	0.557	0.5879	0.0278	0.0232	0.2216	0.1698	0.0361	0.4432	0.3397	0.0722
0.3	1.938	1.368	0.379	0.4643	0.07282	0.0487	0.3101	0.1992	0.0686	0.6203	0.3985	0.1371
0.4	2.639	1.673	0.2613	0.3971	0.1275	0.0709	0.3693	0.2077	0.0936	0.7386	0.4155	0.1872
0.5	3.154	1.848	0.1900	0.3635	0.1795	0.086	0.4050	0.2087	0.1094	0.8099	0.4174	0.2189
0.6	3.508	1.943	0.1485	0.3470	0.2232	0.096	0.4257	0.2077	0.1187	0.8515	0.4155	0.2373
0.7	3.748	1.996	0.1238	0.3386	0.2582	0.1018	0.4380	0.2066	0.1240	0.8761	0.4132	0.4280
0.8	3.913	2.026	0.1088	0.3342	0.2858	0.1054	0.4455	0.2057	0.1271	0.8911	0.4115	0.2540
0.9	4.030	2.043	0.0991	0.3318	0.3077	0.1077	0.4504	0.2050	0.1290	0.9008	0.4101	0.2580

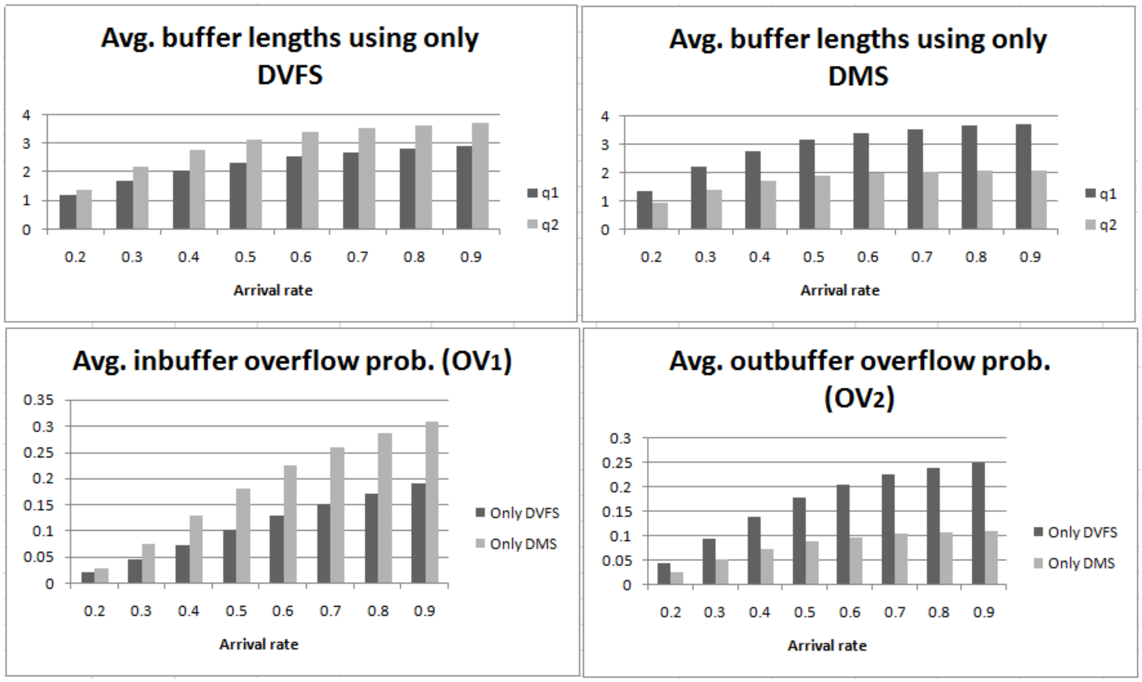


Figure 6.8: Performance of a sensor node with only DVFS and only DMS

From Figure 6.8, it is observed that only DVFS results in average input buffer length (q_1) less than 3. Hence the input buffer overflow probability is also within the limit of tolerance considered (15%). With only DVFS, output buffer length is always higher than input buffer length and results in higher data loss due to buffer overflow probability (more than 0.15). It is highly undesirable as it not only results in data loss but also the power used for processing the lost data goes waste. With only DMS capability, output buffer overflow probability is smaller because of the fixed service at the input and variable service at the output. If service rate of the processor is of some moderate value then input buffer overflow probabilty is more during catastrophic period. This moderate service rate of processor also restricts the use of higher transmission rate. This higher transmission rate can be utilized during catastrophic period by making processor to work with the highest possible service rate. It increases the idle time probabilty of the processor and idle power wastage during normal period. So a sensor node with only DMS capability is also not a good choice.

Table 6.6: Performance parameters of multirate sensor node observed by SHARPE analytical tool

Arrival Rate	**Queue length**		**Idle prob.**		**Overflow prob.**		**Avg.Delay**	**Lifetime**
λ	$q1$	$q2$	I_1	I_2	OV_1	OV_2	D	(units)
0.2	1.398	1.4	0.3687	0.3703	0.00817	0.00853	16.367	23100
0.3	2.45	2.4	0.1725	0.1778	0.0398	0.041	14.257	12000
0.4	3.282	3.2492	0.07688	0.0840	0.09466	0.0963	12.006	9500
0.5	3.856	3.792	0.03529	0.04200	0.1619	0.16355	11.441	8500
0.6	4.259	4.173	0.01711	0.0227	0.2347	0.23600	10.494	7100
0.7	4.559	4.4625	0.008755	0.01314	0.309	0.3107	8.082	6600
0.8	4.798	4.6970	0.00468	0.00807	0.3856	0.3864	8.808	5200
0.9	4.999	4.899	0.00258	0.00521	0.46213	0.46274	8.690	4820

Arrival Rate	**Avg.Throughput**				**Avg.Utilization**			
λ	n_{T_pro1}	n_{T_pro2}	n_{T_tx1}	n_{T_tx2}	U_{T_pro1}	U_{T_pro2}	U_{T_tx1}	U_{T_tx2}
0.2	0.1172	0.0576	0.1569	0.0306	0.586	0.0480	0.3138	0.0306
0.3	0.1270	0.1398	0.1195	0.0824	0.6353	0.1083	0.3990	0.0823
0.4	0.1241	0.2104	0.2166	0.1490	0.6205	0.1753	0.4331	0.1490
0.5	0.1162	0.2900	0.2172	0.2180	0.5813	0.2413	0.4344	0.2177
0.6	0.1069	0.3636	0.2095	0.2800	0.5345	0.3030	0.4190	0.2801
0.7	0.09745	0.3408	0.1987	0.3340	0.4873	0.3590	0.3975	0.3332
0.8	0.0885	0.4908	0.1876	0.3760	0.3769	0.3753	0.4091	0.4426
0.9	0.08033	0.5442	0.1774	0.4120	0.4016	0.4535	0.3548	0.4121

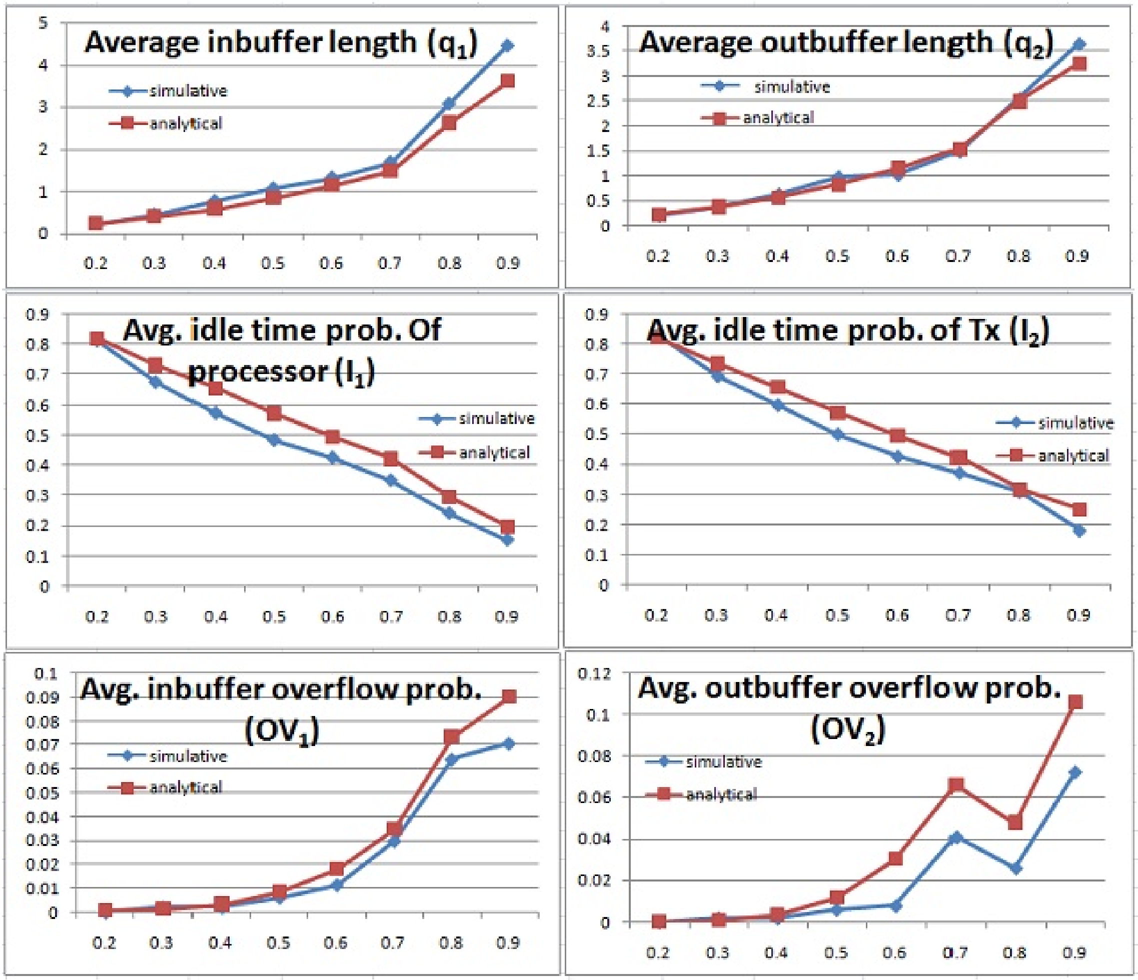


Figure 6.9: Performance of a sensor node with fixed service rate

Figure 6.9 shows performance measures of a wireless sensor node with fixed service rate. Performance measures obtained by SHARPE analysis and obtained from MATLAB simulation have been compared and found consistant with each other.

Similarly Figure 6.10 shows performance measures of a wireless sensor node with multiple service rates both analytical and simulative.

Figure 6.11 compares the various performance measures of a sensor node with fixed service rate with that of a sensor node with multiple service rates. Overflow probability OV_2 is one of the QoS parameter of interest and Lifetime of sensor node is another. With the use of

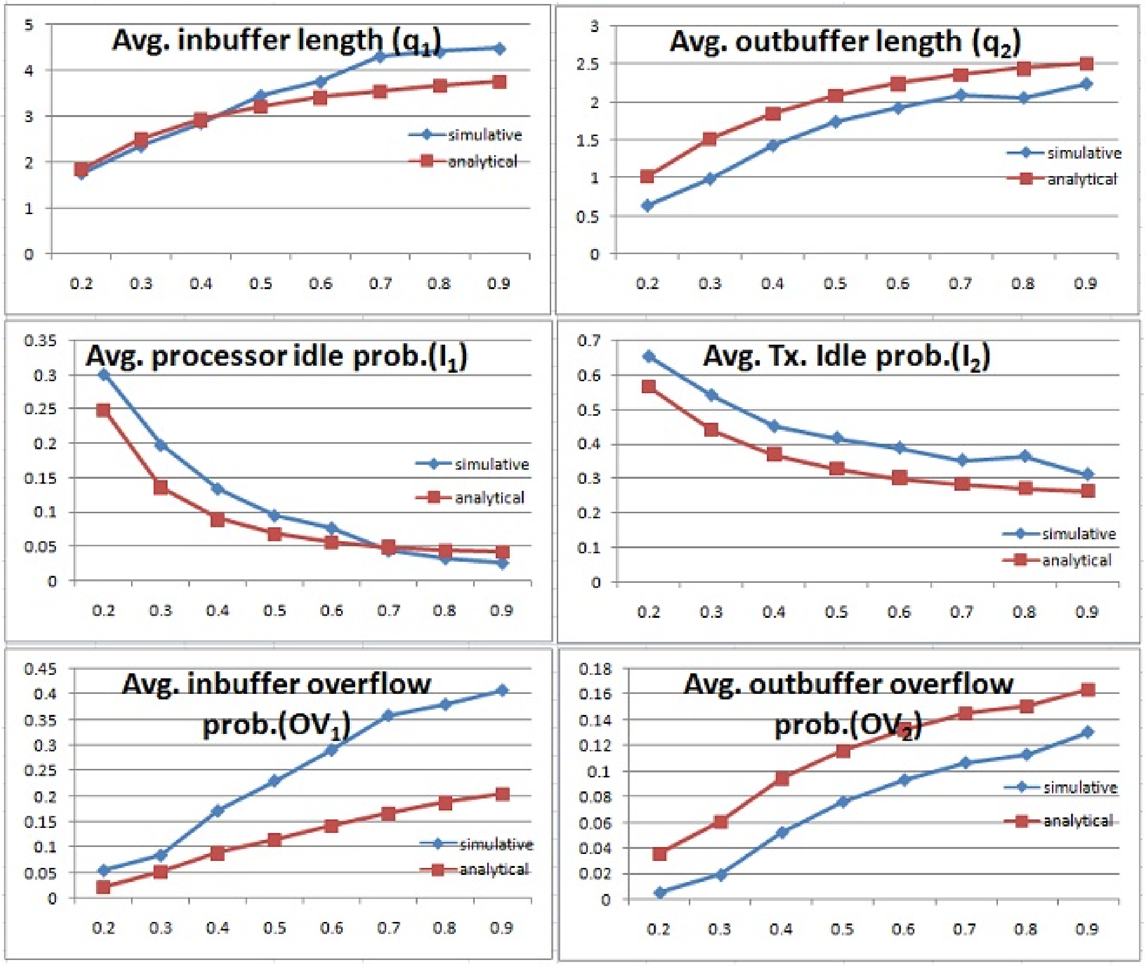


Figure 6.10: Performance of a sensor node with multiple service rates

CAP management technique it is seen possible to reduce OV_2 and increase the lifetime as seen from Figure 6.12.

From all the above comparison graphs a rate adaptive (multi rate) sensor node seems to outperform the one with fixed service rate in view of lifetime, buffer overflow probability, latency, throughput etc.

Next chapter summarizes the thesis, its conclusion and the future scope of research.

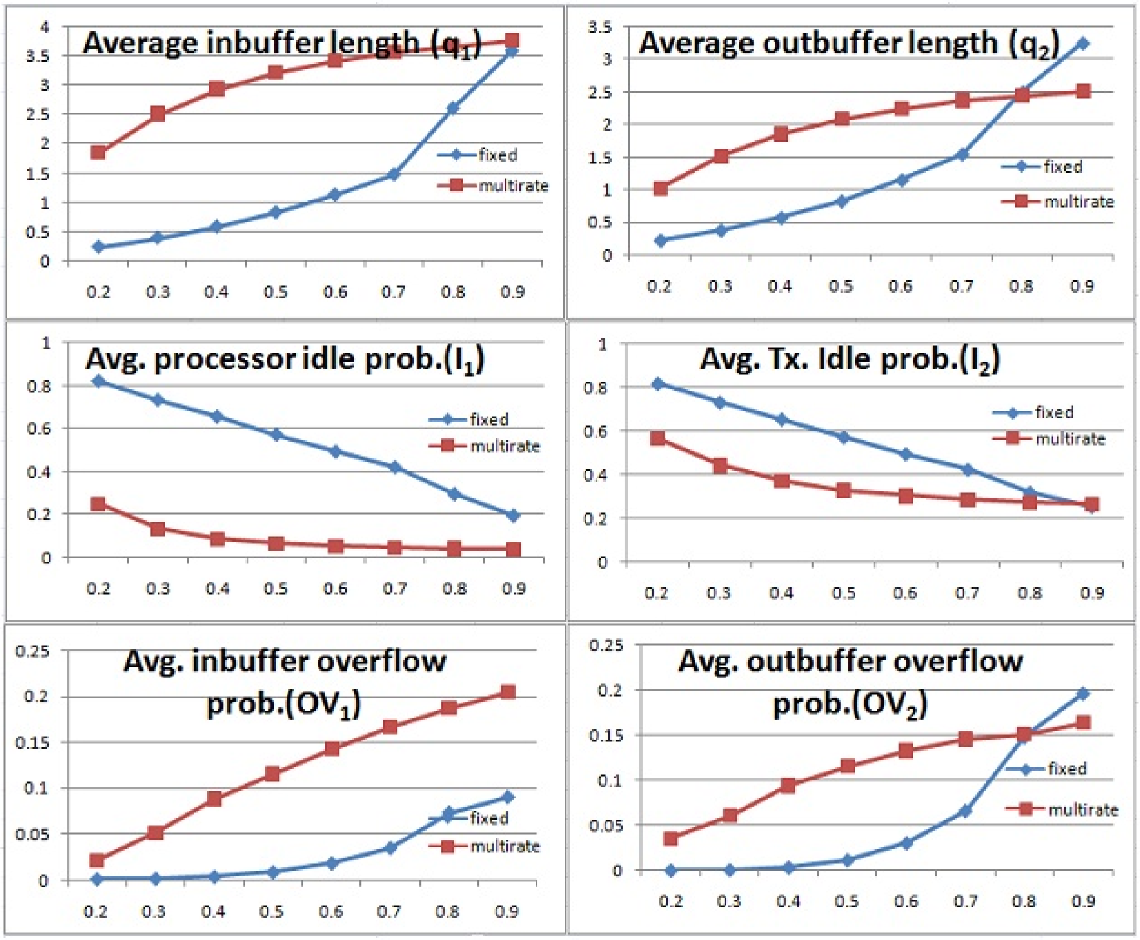


Figure 6.11: Comparison between fixed service rate and multiple service rates

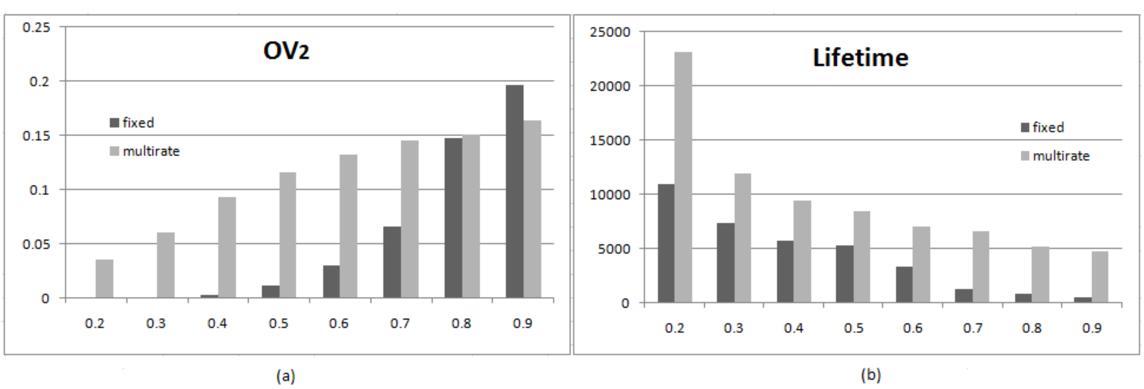


Figure 6.12: Comparison between fixed service rate and multiple service rates for (a) buffer overflow prob. and (b) lifetime

Chapter 7

CAPM Overheads

Analytical and simulation results show the lifetime improvement of a sensor node along with the reduction in buffer overflow probability. Now we discuss the additional energy cost required to achieve this. Due to large number of sensor nodes, price has to be very less. Hardware with DVFS and DMS capability may increase the price. With advancement in technology this price will reduce in future. But other than price overhead, there are some overheads associated with CAPM technology. Startup time and startup energy overheads are always there even if CAPM is not implemented. With the help of periodic sleep scheduling, we have restricted the very frequent wakeup of sensor nodes.

Overheads are associated with CAPM technique mainly because of its rate switching capability. Switching time and switching energy are two main factors which contribute to the performance and energy cost of sensor node respectively. For CAPM enabled sensor node, we need a processor/ microcontroller with DVFS capability and a radio transceiver with DMS capability. Currently no radio is available with DMS capability but DVFS is available with a number of high end processors but not siutable for WSN applications due to high power consumptions.

7.1 Overheads with low end devices

Now a days, low power hardware, compatible with IEEE 802.15.4 wireless standard are available and widely used in wireless Sensor Network. ATmega128RFA1 is having 8-bit AVR Microcontroller with Low Power 2.4GHz Transceiver for ZigBee and IEEE 802.15.4. It's microcontroller speed grade is 0-16 MHz 1.8-3.6 V. It's transceiver supports 4 different data rates of 250 kb/s, 500kb/s, 1Mb/s and 2Mb/s [65].

Figure 7.1 shows the current drawn by microcontroller when operated with various speeds (operated with supply voltage of 3 volts). It is important to note that idle current consumption is different for different speeds. As seen in Chapter 2, reducing the speed (processing rate) will reduce the power consumption and idle time of sensor node. With reduced speed even if sensor node remains idle for some duration, it will consume less idle power than that with higher speed.

Symbol	Parameter	Condition / AVR mode	Min	Typ	Max	Units
I_{SUPPLY}	Power Supply Current (PRR0=0xFF, PRR1=0x3F, 16MHz RC Oscillator selected)	Standby mode		0.31		mA
		Idle 1MHz		0.45		mA
		Idle 8MHz		0.8		mA
		Idle 16MHz		1.1		mA
		Active 1MHz		0.8		mA
		Active 8MHz		2.5		mA
		Active 16MHz		3.7		mA
	Power Supply Current (PRR0=0x00, PRR1=0x00)	Active, 16MHz RC Oscillator		4.0		mA
		Active, 16MHz Crystal Oscillator		4.5		mA
		Active, external 16MHz clock on CLKI		4.5		mA

Figure 7.1: Current consumption of microcontroller in ATmega128RFA1 [162]

As wakeup time for low power devices is in microseconds (less than 250 microseconds for ATmega128RFA1 radio) we may get switching times just few nano seconds. Switching energy is a function of switching time, have smaller switching time will result in less switching energy overheads. ATmega128RFA1 neither have Dynamic Voltage nor Dynamic Frequency scaling capability inbuilt within it. But application can change the CPU freqnency in the application program at run time by configuring the Prescalar register. When we tried to achieve DFS by configuring the Prescalar register, we observed the frequency switching time for various frequencies between 19 μ sec to 28 μ sec. For rough estimation of switching energy, we will consider a constant frequency switching time of approx. 20 μ sec. Figure 7.2 and Figure 7.3 show the clock pulses with two different frequencies (1 MHz and 8 MHz) and switching time period in between.

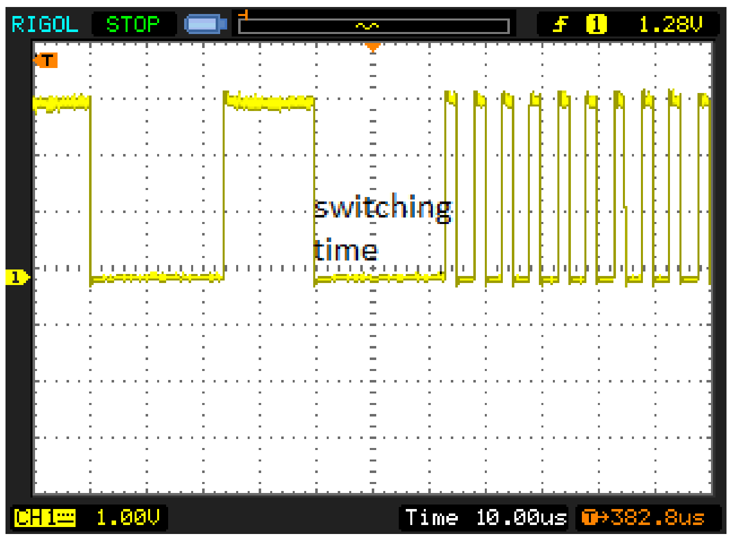


Figure 7.2: Clock frequency switching from 8 MHz to 1 MHz

Only DFS does not give any energy reduction. In order to exploit the advantage of DVFS we consider two clock frequencies at two different supply voltages (note that voltage range 1.8V to 3.6V supports frequency range from 1 MHz to 16 MHz). Let us consider maximum clock frequency of 16 MHz, supported at 3.0V supply voltage and other clock frequency of 8 MHz.

$$active\ power\ with\ 16MHz\ freq. = (3.7mA) \cdot (3V) = 11.1\ mW \tag{7.1}$$

$$idle\ power\ with\ 16MHz\ freq. = (1.1mA) \cdot (3V) = 3.3\ mW \tag{7.2}$$

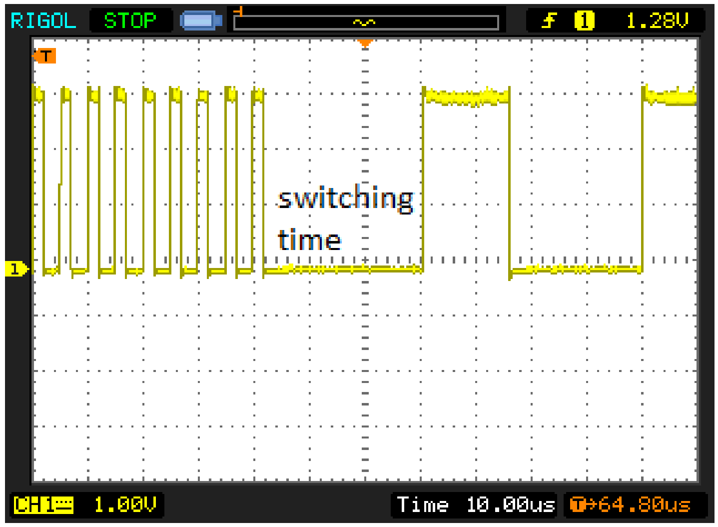


Figure 7.3: Clock frequency switching from 1 MHz to 8 MHz

Now, when operated with 8 MHz clock frequency, power consumption will be just $(8/16)^3 = 0.125$ times that with 16 MHz frequency.

$$active\ power\ with\ 8MHz\ freq. = (0.125) \cdot (11.1\ mW) = 1.3875\ mW \tag{7.3}$$

If we consider 8 MHz frequency supported at 1.8 V, active current drawn will be 0.7708 mA. Idle current drawn with 8 MHz frequency at 1.8 V is 0.75 mA.

$$idle\ power\ with\ 8MHz\ freq. = (0.75mA) \cdot (1.8V) = 1.35\ mW \tag{7.4}$$

We executed a simple program on ATmega128RFA1, where 200 buffer characters were stored and added together, first 100 characters were added with 16 MHz clock frequency and remaining 100 characters added with 8 MHz clock frequency. It is similar to DVFS where buffer threshold is set to the 100, if number of characters in buffer are more than 100, serve with higher rate (16 MHz) and if less than 100, serve with lower rate (8 MHZ). We have suggested the same policy for CAPM.

We observed that 32 clock cycles are required to add one character every time. Clock period with 16 MHz clock frequency is 0.0625 μsec and with 8 MHz freqnency it is 0.125 μsec. For adding first 100 characters, time taken is,

$$\begin{aligned} T_{16} &= (No.\ of\ characters\ to\ be\ added) \cdot (clk\ cycles\ per\ character\ addition) \cdot (clock\ period) \\ &= (100) \cdot (32) \cdot (0.0625) \\ &= 200\ \mu sec \end{aligned} \tag{7.5}$$

Similarly, adding next 100 characters with 8 MHz clock frequency will take,

$$T_8 = 400\ \mu sec \tag{7.6}$$

So, total time reqired for adding 200 characters should be,

$$\begin{aligned} T_{Total} &= T_{16} + T_8 \\ &= 600\ \mu sec \end{aligned} \tag{7.7}$$

Experimentally, we observed this time is 720 μsec i.e. 120μsec extra time is required. This time consists of frequency switching time and time required for execution of software program which monitors the buffer and accordingly make switching decision. In our experiment frequency switching occurs only once frome 16 MHz to 8 MHz. As observed, switching time is approx. 20 μsec. Remaining 100 μsecs are consumed for buffer monitoring. In our experiment, buffer is monitored 200 times. 100 times buffer was monitored with 16 MHZ frequency and 100 times with 8 MHz frequency. As working with 8 MHz frequency takes double time than that with 16 MHz, buffer monitoring with 8 MHz has consumed total 66.66 μsecs and with 16 MHz it has consumed 33.33 μsecs. Here, we can note that buffer monitoring program execution consumes 5 clock cycles every time. This is about the latency (total switching time) overhead. Now, we calculate total switching enery overheads.

Frequency switching occured only once, from 16 MHz to 8 MHz, so during switching period power in the initial state will be consumed.

$$\begin{aligned} freq.\ switching\ energy\ (16 \rightarrow 8) &= (switching\ time) \cdot (P_{16}) \\ &= (20\mu secs) \cdot (11.1mW) \\ &= 222\ nJ \end{aligned} \tag{7.8}$$

Total buffer monitoring energy can be calculated as,

$$\begin{aligned} buffer\ moni.\ energy &= (moni.\ time\ with\ 16\ MHz) \cdot (P_{16} + (moni.\ time\ with\ 8MHz) \cdot (P_8) \\ &= (33.33\mu secs) \cdot (11.1mW) + (66.66\mu secs) \cdot (1.3875mW) \\ &= 461.5\ nJ \end{aligned} \tag{7.9}$$

Total energy cost is,

$$\begin{aligned} Total\ switching\ energy\ (energy\ cost) &= (freq.\ switching\ energy) + (buffer\ monitoring\ energy) \\ &= (222nJ) + (461.5nJ) \\ &= 683.5\ nJ \end{aligned} \tag{7.10}$$

$$\begin{aligned}
Total\ energy\ consumption\ (E_{var}) &= (P_{16} \cdot 200) + (P_8 \cdot 400) + (Total\ switching\ energy) \\
&= (2220nJ) + (555nJ) + (683.5nJ) \\
&= 3458.5\ nJ \qquad (7.11)
\end{aligned}$$

Out of 3458.5 nJ of total energy consumption, 2775 nJ is consumed for adding of characters and 683.5 nJ is the extra energy cost used for frequency switching.

If we have not performed frequency switching and have operated with fixed 16 MHz frequency only, total enery required is,

$$\begin{aligned}
E_{16} &= (T_{16}) \cdot (P_{16}) \\
&= (400\mu secs) \cdot (11.1mW) \\
&= 4440\ nJ \qquad (7.12)
\end{aligned}$$

$$\begin{aligned}
Energy\ saved\ for\ adding\ of\ characters &= 4440 - 2775 \\
&= 1665\ nJ \qquad (7.13)
\end{aligned}$$

We can see, 1665 nJ energy is saved at the extra energy cost of 683.5 nJ. So, net energy saving is 981.5 nJ.

$$\begin{aligned}
Net\ energy\ saving &= (E_{16}) - (E_{var}) \\
&= (4440nJ) - (3458.5nJ) \\
&= 981.5\ nJ \qquad (7.14)
\end{aligned}$$

From above equations, it becomes clear that DVFS not only reduces power but also reduces energy consumption. Here, it must be noted that the actual power consumption depends on number of times frequency switches from higher value to lower value and vice versa. In above analysis, if frequency switching takes place from 8 MHz to 16 MHz then frequency switching energy consumption will be different though we consider switching time to be constant. It will be,

$$\begin{aligned} freq.\ switching\ energy\ (8 \rightarrow 16) &= (switching\ time) \cdot (P_8) \\ &= (20 \mu secs) \cdot (1.3875 mW) \\ &= 27.75\ nJ \end{aligned} \tag{7.15}$$

which is much smaller than 222 nJ energy consumed while switching frequency from 16 MHz to 8 MHz.

Now consider radio transceiver used in ATmega128RFA1. It does not support DMS but support four data rates of 250 Kbps, 500 Kbps, 1 Mbps and 2 Mbps. It also supports variable transmit power outputs. Using DMS we are able to vary data rate as well as transmit power together. We try to map these variable output power with DMS. Here symbol rate is considered 250 Kbps, supply voltage is 3.0 V.

Figure 7.4 shows the current drawn by radio transceiver used in ATmega128RFA1. We consider, when number of bits transmitted per symbol is 1 then bit rate is same as symbol rate (250 Kbps). With this data rate, RF output power is -16.5 dBm and transmitter power consumption is 24 mW. When number of bits transmitted per symbol are 2, then effective bit rate becomes 500 Kbps,RF output power is 3.5 dBm and transmitter power consumption is 43.5 mW.

As ATmega128RFA1 is Zigbee compliant, we consider the same packet size of maximum 128 bytes, packet header of 28 bytes and maximum payload of 100 bytes (please note that

Symbol	Parameter	Condition	Min.	Typ.	Max.	Units
I_{BUSY_TX}	Supply current transmit state	P_{TX} = 3.5 dBm P_{TX} = 1.5 dBm P_{TX} = -2.5 dBm P_{TX} = -16.5 dBm (current consumption is reduced at V_{DD} = 1.8V for each output power level)		14.5 10 9 8		mA mA mA mA
I_{RX_ON}	Supply current RX_ON state	RX_ON state		12.5		mA
$I_{RX\_ON\_P}$	Supply current RX_ON state	RX_ON state, with register setting RX_PDT_LEVEL > 0[1]		12.0		mA
I_{PLL_ON}	Supply current PLL_ON state	PLL_ON state		5.7		mA
I_{TRX_OFF}	Supply current TRX_OFF state	TRX_OFF state		0.4		mA
I_{SLEEP}	Supply current SLEEP state	SLEEP state		0.02		µA

Figure 7.4: Current consumption of radio transceiver in ATmega128RFA1 [162]

for ATmega128RFA1, these specifications may differ). 4 msec time is required to transmit a packet of 128 bytes using 250 Kbps data rate (i.e. 1 bit per symbol).

$$\begin{aligned} Energy\ consumed &= (transmission\ time) \cdot (power) \\ &= (4\ msec) \cdot (24mW) \\ &= 96\ \mu J \end{aligned} \tag{7.16}$$

Now if we select 2 bits per symbol for transmission then effective bit rate will be two times more than symbol rate (i.e. 500 Kbps). Effective packet size will be,

$$\begin{aligned} effective\ packet\ size &= header\ size + effective payload \\ &= 28 + 50 \\ &= 78\ bytes \end{aligned} \tag{7.17}$$

Time required for transmission of one packet will be, 2.5 msec.

$$\begin{aligned} Energy\ consumed &= (transmission\ time) \cdot (power) \\ &= (2.5\ msec) \cdot (43.5 mW) \\ &= 108.75\ \mu J \end{aligned} \tag{7.18}$$

From above analysis, we can see that increasing the number of bits per symbol increases the effect data rate and reduces the transmission time. This time saving is obtained at the cost of extra energy consumption. Here 1.5 msec time is saved but at the cost of extra 12.75 μ J of energy.

ATmega128RFA1 packet header does not contain any information regarding data rate. So, on the fly data rate can not be changed. Both transmitter and receiver are required to be synchronized for the same data rate. Here it is not possible to check rate switching times but as the radio wakeup time is less than 250 μsec,rate switching times are expected to be less than 250 μsec.

Chapter 8

Conclusion and Future Scope

At the end of the thesis, before reaching to the conclusion, we will first summarize the thesis. Future scope gives further possibilities of research in this dirction.

8.1 Summary

In this thesis, we have suggested a coordinated adaptive power management (CAPM) technique for wireless sensor nodes. Power optimization technique DVFS for processor and DMS for transmitter are coordinted together w.r.t. the input buffer occupancy (workload). With CAPM, we are trying to increase the lifetime of individual sensor nodes, which in turn improves the lifetime of Wireless Sensor Network (WSN). For the rare event monitoring and detection applications, data traffic in the network is non uniform and unpredictable. In this scenario, it is important to have WSN alive when the event of interest occurs. In order to have WSN alive, sufficiently high number of sensor nodes must be alive in the network. So, increasing the lifetime of individual sensor node is important in view of a long lived WSN. Low power hardware and sleep scheduling schemes are helpful in increasing lifetime of sensor nodes. Further power saving can be achieved by applying CAPM during ON period of sensor nodes.

We are aiming to increase the lifetime by power optimization at senor node level but at the same time we are concerned with the QoS. Here QoS parameter considered is data loss due to buffer overflow. Due to very small size and limited hardware capabilities sensor nodes are highly prone to this kind of data loss. Purpose of CAP management technique is to use the optimum power to handle the workload with QoS. Wireless sensor networks are mainly deployed for special event detection and monitoring of such events and hence need to handle a very non uniform traffic pattern. Sensor nodes use multihop communication due to limited battery power available for transmission. Hence when an event gets detected at one corner of the sensing field, other sensor nodes far away in the field need to forward the detected data towards the sink. So, aim of this research is two fold- Saving the power during no event period (normal period) and reducing the data loss due to buffer congestion at the cost of increased power consumption when event occurs (catastrophic period).

A wireless sensor node with discrete rates has been considered as practically it is not possible to have a continuous rate adaptation due to hardware constraints. Rate adaptation helps to adjust the power consumption of a sensor node as per the actual workload requirement. Figure 3.5 shows the conceptual block schematic for CAPM technique. In Chapter 4, tandem queue model of sensor node has been considered with various capabilities and MATLAB simulation results have shown. We have modeled, analyzed and simulated a sensor node with fixed service rate, with only DVFS, with only DMS technique and with both DVFS and DMS implemented together. Table 4.1 shows the simulation results for a sensor node with fixed service rate. It shows that, during normal period, sensor node remains idle over large period (25% to 58% of ON time). So, obviously idle power wastage is more. During catastrophic period, idle period has minimized (6% and below) but overflow probability at input buffer (OV_1) is very large (above 25%). It means increased data traffic is dropped before entering the node, so power is saved but data of interest has lost. Figure 4.6, Figure 4.7 and Figure 4.8 show the comparison between sensor nodes with various capabilities. A sensor node with only DVFS facility implemented on it can control the overflow probability of input buffer by increasing the clock frequency of the processor. Increased service rate of the processor increases the data arrival rate in the output buffer but since transmitter works with fixed service rate, overflow probability of the output queue (OV_2) increases. This situation is highly undesirable as the processed data gets lost and power used for processing that data also goes waste. Similarly, when only DMS is implemented, it results in decreasing the output buffer overflow probability at the cost of increased power but as the first server works with fixed service rate, input buffer overflow can not be controlled. Integrating both DVFS and DMS on a sensor node results in controlling the overflow probabilities of both the buffers. Though more power is consumed by DVFS and DMS during catastrophe by working with higher service rates data loss due to buffer overflow is reduced and can be kept within the tolerence limit which is the highest priority QoS parameter during catastrophe.

As the data arrival rate is very small during normal periods, buffer overflow possibilities are negligible but possibilities of servers remaining idle are more. More the idle period, more is the power wastage. So, for power constrained wireless sensor nodes power saving becomes highest priority QoS parameter during normal periods is reducing the idle period. It is achieved by reducing the service rates of the servers. Fixed service rate sensor nodes are designed to handle worst case conditions and hence their service rates are set quite high. During normal periods these servers remain idle most of the time and large amount of power is wasted but there is negligible chance of buffer overflow. Power saved during normal periods can be used during catastrophic periods to reduce data loss due to buffer overflows. As compared to fixed service rate sensor node lifetime increase of 15% was seen when only DVFS was implemented on a sensor node while implementing only DMS it was 17.5% but DVFS and DMS together applied on a sensor node resulted in 27.22% lifetime increase.

After analyzing a sensor node as a tandem queue, CAPM found giving better results in terms of lifetime and overflow probability. We have considered a sensor node as a single entity (server) inside which DVFS and DMS works in a coordinated manner. Changing the service rate of sensor node will internally select a specific processing rate and a specific number of bits per symbol in a look up table manner. This single server model of sensor node we have tried to capture using Markov chain (Refer Figure 5.4. This is a bulk arrival and batch service $M^{[x]}/M/1/N$ model. Flow chart for a sensor node with CAPM and having two active states (active Low and active High) is shown in Figure 5.3.

Analysis of Markov chain model on paper has become a tedious job due to large number of equations involved in it. So, we have carried out the analysis with the GSPN models using software analysis tool, SHARPE. Chapter 6, gives all the GSPN models and their sensitivity analysis. Figures 6.11 and Figure 6.12 shows the comparison between a sensor node with fixed service rate and with multiple service rates (CAPM). From all the above comparison graphs, a multi rate (CAPM) sensor node seems to outperform the one with fixed service rate in view of lifetime, buffer overflow probability, latency, throughput etc.

8.2 Conclusion

Non uniformity of traffic in WSN can be exploited to increase the lifetime of sensor nodes. Sensor node with only DVFS or only DMS capability does not give required QoS. DVFS and DMS implementation on a sensor node with coordinated manner can give us better results. Selected service rates are considered to satisfy the latency constraint. Data loss due to buffer overflows can be reduced with CAPM technique along with lifetime improvement.

Both DVFS and DMS integrated together on a sensor node can effectively suffice the purpose of power optimization as well as data loss reduction. DVFS and DMS techniques coordinated with input buffer makes the makes the sensor node operation more effective. Markov model of sensor node, Matlab simulation results and Generalized Stochastic Petri Net (GSPN) model of sensor node simulated using SHARPE, support the concept of CAP management. We have analyzed a sensor node as a tandem queue model to capture the internal functioning of a sensor node and effect of DVFS and DMS on various parameters. After coordinating DVFS and DMS together with actual workload to be handled, we prepared a look up table which gives values of processing frequency and transmission rate for a specific value of data arrival rate which reduces data loss. Each pair of processing frequency and constellation size (transmission rate) is assigned with a specific value of service rate of a sensor node.

8.3 Future scope

This thesis provides the performance analysis of a sensor node which supports only two service rates (active high and active low). With the availability of hardware, such N number of discrete service rates can be possible. Study of CAPM with such N number of service rates may result in better power optimization and better QoS, where switching from one rate to the next rate will be much faster and with minimal overheads. Along with performance analysis, reliability analysis of the model can be carried out.

Another thing that can be done further is a WSN can be simulated with every sensor node with CAPM capability. For some real life application effect of CAPM on the lifetime of WSN can be studied. This thesis has studied the effect of CAPM on the lifetime of a sensor node. When lifetime of all the sensor nodes in a WSN will be increased using CAPM technique, what is the aggregate effect on the lifetime of WSN can be studied.

This thesis presents modeling and simulation of wireless sensor node with CAPM. It will be interesting to implement on real hardware. Currently no radio hardware available in the market supports DMS technique. Design of a low power radio with DMS capability will ensure the actual gain with CAPM.

List of Publications

1. Gauri Joshi and Prabhat Ranjan: CAP (Coordinated Adaptive Power) Management Technique with Adaptive Threshold Policy for Wireless Sensor Nodes, *In the proceedings of* 32*nd* *Asia Pacific Advanced Network (APAN-NRW) meet*, New Delhi, August 2011

2. Gauri Joshi and Prabhat Ranjan: Optimizing Power and Buffer Congestion on Wireless Sensor Nodes using CAP (Coordinated Adaptive Power) Management Technique, *International Journal of Wireless and Mobile Networks (IJWMN)*, Volume 3, No. 2, April 2011, pp-225-241

3. Gauri Joshi, Sudhanshu Dwivedi, Anshul Goel, Jaideep Mulherkar and Prabhat Ranjan: Power and Buffer Overflow Optimization in Wireless Sensor Nodes, *In the proceedings of Advanced Computing Communications in Computer and Information Science 2011* , Volume 133, Part 5, pp-450-458

4. Gauri Joshi, Guneshwar Anand, R.B.Lenin, Prabhat Ranjan: Performance evaluation of wireless sensor nodes using queuing model, *In the proceedings of IEEE International workshop on mobile systems (WoMS)*, West Bengal Technical University, Kolkata, July 2008

5. Gauri Joshi, Sunil Jardosh, Prabhat Ranjan: Dynamic Modulation Scaling for Energy Efficient Topology Control of Wireless Sensor Networks, *In the proceedings*

of IEEE International workshop on mobile systems (WoMS), West Bengal Technical University, Kolkata, July 2008.

6. Gauri Joshi, Sunil Jardosh, Prabhat Ranjan: Multiple Access Interference (MAI) and reduction techniques for Wireless Sensor Networks, *In the Proceedings of International Conference on Radio Science (ICRS)*, Jodhpur, Feb.2008

7. Gauri Joshi, Sunil Jardosh, Prabhat Ranjan: Bounds on Dynamic Modulation Scaling for Wireless Sensor Networks, *In the Proceedings of IEEE Third international conference on Wireless Communications and Sensor Networks (WCSN) 2007*, IIIT Allahabad, India, December 2007, pp.90-93

8. Gauri Joshi, Prabhat Ranjan: Co-Ordinated Adaptive Power (CAP) Management for Wireless Sensor Networks, *In the proceedings of International Conference on Sensor Technologies and Applications (SENSORCOMM)2007* , Valencia, Spain, October 2007, pp. 554-559 IEEEComputerSociety.

Bibliography

[1] Zenon Chaczko and Fady Ahmad. Wireless sensor network based system for fire endangered areas. In *Proceedings of the Third International Conference on Information Technology and Applications (ICITA'05) Volume 2 - Volume 02*, ICITA '05, pages 203–207, July 2005.

[2] Alan M. Mainwaring, David E. Culler, Joseph Polastre, Robert Szewczyk, and John Anderson. Wireless sensor networks for habitat monitoring. In *WSNA*, pages 88–97, September 2002.

[3] N.Xu S.Rangwala K.Chintalapudi D.Ganesan, A.Broad, R.Govindan, and D.Estrin. A wireless sensor network for structural monitoring. In *In Proceedings of the ACM Conference on Embedded Networked Sensor Systems(Sensys04) Baltimore, Maryland, USA*, November 2004.

[4] Yang Wang A, Jerome P. Lynch B, and Kincho H. Law. Design of a low-power wireless structural monitoring system for collaborative computational algorithms, March, 2005.

[5] R. X. Gao, A. Deshmukh, R. Yan, and Z. Fan. Energy efficient wireless sensor network for dynamic system monitoring. In B. Gopalakrishnan, editor, *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, volume 5999

of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, pages 36–45, November Nov. 2005.

[6] A. Arora, P. Dutta, S. Bapat, V. Kulathumani, H. Zhang, V. Naik, V. Mittal, H. Cao, M. Gouda, Y. Choi, T. Herman, S. Kulkarni, U. Arumugam, M. Nesterenko, A. Vora, and M. Miyashita. A line in the sand: A wireless sensor network for target detection, classification, and tracking. *Computer Networks (Elsevier*, 46, Issue 5:605–634, 2004.

[7] Marco F. Duarte and Yu Hen Hu. Vehicle classification in distributed sensor networks. *J. Parallel Distrib. Comput.*, 64:826–838, July 2004.

[8] Dan Li, Kerry D. Wong, Yu H. Hu, and Akbar M. Sayeed. Detection, classification and tracking of targets in distributed sensor networks. In *IEEE Signal Processing Magazine*, volume 19, Issue 2, pages 17–29, 2002.

[9] Victor Shnayder, Bor-rong Chen, Konrad Lorincz, Thaddeus R. F. Fulford Jones, and Matt Welsh. Sensor networks for medical care. In *Proceedings of the 3rd international conference on Embedded networked sensor systems*, SenSys '05, pages 314–314, New York, NY, USA, 2005. ACM.

[10] I. Y. Tumer and A. Bajwa. A survey of aircraft engine health monitoring systems. In *The 1999 AIAA/ASME/SAE/ASEE Joint Propulsion Conference*, Los Angeles, CA, June 1999.

[11] Fabrizio Mulas, Andrea Acquaviva, Salvatore Carta, Gianni Fenu, Davide Quaglia, and Franco Fummi. Network-adaptive management of computation energy in wireless sensor networks. In *Proceedings of the 2010 ACM Symposium on Applied Computing*, SAC '10, pages 756–763, New York, NY, USA, 2010. ACM.

[12] Deborah Estrin, David Culler, Kris Pister, and Sukhatme Gaurav. Connecting the physical world with pervasive networks. *IEEE Pervasive Computing*, 1:59–69, January 2002.

[13] Yu Gu, Ting Zhu, and Tian He. Esc: Energy synchronized communication in sustainable sensor networks, October, 2009.

[14] Aman Kansal, Jason Hsu, Sadaf Zahedi, and Mani B. Srivastava. Power management in energy harvesting sensor networks. *ACM Trans. Embed. Comput. Syst.*, 6, September 2007.

[15] Shaobo Liu, Qinru Qiu, and Qing Wu. Energy aware dynamic voltage and frequency selection for real-time systems with energy harvesting. In *Proceedings of the conference on Design, automation and test in Europe*, DATE '08, pages 236–241, New York, NY, USA, 2008. ACM.

[16] C. Moser, D. Brunelli, L. Thiele, and L. Benini. Real-time scheduling with regenerative energy. In *In Proc. of the 18th Euromicro Conference on Real-Time Systems (ECRTS 06*, pages 261–270. IEEE Computer Society Press, 2006.

[17] Winston K. G. Seah, Zhi Ang Eu, and Hwee pink Tan. Wireless sensor networks powered by ambient energy harvesting (wsn-heap) survey and challenges. In *1st International Conference on Wireless Communication Vehicular Technology, Information Theory and Aerospace Electronic Systems Technology*, pages 1–5, 2009.

[18] Massoud Pedram. Power minimization in IC design: Principles and applications. *ACM Transactions on Design Automation of Electronic Systems*, 1(1):3–56, January 1996.

[19] Benini, L. Bogliolo, and G. De Micheli. A survey of design techniques for system-level dynamic power management. *IEEE transactions on Very Large Scale Integrated Systems*, 8:299–316, June 2000.

[20] Srinivas Devadas and Sharad Malik. A survey of optimization techniques targeting low power vlsi circuits. In *DAC '95: Proceedings of the 32nd ACM/IEEE conference on Design Automation*, pages 242–247, New York, NY, USA, 1995. ACM Press.

[21] Massoud. *Power Aware Design Methodologies*. Kluwer Academic Publishers, Norwell, MA, USA, 2002.

[22] Chandrakasan, Sheng S., and Brodersen R. W. Low-power cmos design. *IEEE Journal of Solid-State Circuits*, pages 472–484, 1992.

[23] Davari B., Dennard R.H., and Shahidi G.G. Cmos scaling for high performance and low power-the next ten years. *Proceedings of the IEEE*, 83(4):595–606, Apr 1995.

[24] Manjit Borah, Robert Michael Owens, and Mary Jane Irwin. Transistor sizing for minimizing power consumption of cmos circuits under delay constraint. In *ISLPED '95: Proceedings of the 1995 International Symposium on Low Power Electronics and Design*, pages 167–172, New York, NY, USA, 1995. ACM Press.

[25] K.Y. Chao and D. F. Wong. Low power considerations in floorplan design. In *Proceedings of the 1999 International Workshop on Logic Power Design*, pages 45–50, Apr 1999.

[26] J. Chang and M. Pedram. Energy minimization using multiple supply voltages. In *ISLPED '96: Proceedings of the 1996 International Symposium on Low Power Electronics and Design*, pages 157–162, Piscataway, NJ, USA, 1996. IEEE Press.

[27] Donghwan Son. Dynamic voltage scaling on MPEG decoding. In *ICPADS '01: Proceedings of the Eighth International Conference on Parallel and Distributed Systems*, page 633, Washington, DC, USA, 2001. IEEE Computer Society.

[28] Malena Mesarina and Yoshio Turner. Reduced energy decoding of MPEG streams. *Multimedia Systems*, 9(2):202–213, 2003.

[29] Padmanabhan Pillai and Kang G. Shin. Real-time dynamic voltage scaling for low-power embedded operating systems. In *SOSP '01: Proceedings of the Eighteenth ACM Symposium on Operating Systems Principles*, pages 89–102, New York, NY, USA, 2001. ACM Press.

[30] Mark Weiser, Brent Welch, Alan Demers, and Scott Shenker. Scheduling for reduced CPU energy. In *OSDI '94: Proceedings of the 1st USENIX conference on Operating Systems Design and Implementation*, page 2, Berkeley, CA, USA, 1994. USENIX Association.

[31] G. A. Paleologo, L. Benini, A. Bogliolo, and G. De Micheli. Policy optimization for dynamic power management. In *DAC '98: Proceedings of the 35th annual conference on Design Automation*, pages 182–187, New York, NY, USA, 1998. ACM Press.

[32] Merkle R.C. Reversible electronic logic using switches. *IEEE Journal of Nanotechnology*, 4:21–40, 1993.

[33] William C. Athas, Lars J. Svensson, Jeffrey G. Koller, Nestoras Tzartzanis, and Eric Ying-Chin Chou. Low-power digital systems based on adiabatic-switching principles. *IEEE Transactions on Very Large Scale Integrated Systems*, 2(4):398–407, 1994.

[34] Paul J.M. Havinga and Gerard J.M. Smit. Design techniques for low-power systems. *Journal of System Architecture*, 46(1):1–21, 2000.

[35] A. P. Potkonjak, M. Mehra, R. Rabaey, and J. Brodersen R. W. Optimizing power using transformation. *IEEE transactions on Computer Aided Design of Integrated Circuits and Systems*, 14:12–31, Jan 1995.

[36] Renu Mehra and Jan Rabaey. Exploiting regularity for low-power design. In *ICCAD '96: Proceedings of the 1996 IEEE/ACM international conference on Computer-aided design*, pages 166–172, Washington, DC, USA, 1996. IEEE Computer Society.

[37] H. H. Loomis and B. Sinha. High speed recursive digital filter realization. *IEEE Journal of Circuit, Systems, and Signal Processing,*, 3(3):267–294, 1984.

[38] K. K. Parhi and D. G. Messerschmitt. Pipeline interleaving and parallelism in recursive digital filters - Part I and II. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 37:1099–1134, July 1989.

[39] Guoliang Xing. *Unified Power Management in Wireless Sensor Networks*. PhD thesis, *School* of Engineering and Applied Science, Washington university in St. Louice, October 2006.

[40] Jason Hill, Mike Horton, Ralph Kling, and Lakshman Krishnamurthy. The platforms enabling wireless sensor networks. *Commun. ACM*, 47:41–46, June 2004.

[41] Ian F. Akyildiz, Weiiian Su, Yogesh Sankarasubramaniam, and Erdal Cayirci. A survey on sensor networks. *IEEE Communications Magazine*, 40(8):102 – 114, August 2002.

[42] Mac protocols for wireless sensor networks: a survey. *IEEE Communications Magazine*, 44(4):115–121, April 2006.

[43] Jamal N. Al-karaki and Ahmed E. Kamal. Routing techniques in wireless sensor networks: A survey. *IEEE Wireless Communications*, 11:6–28, December 2004.

[44] Xiaodong Zhang Xiaoyan Cui and Yongkai Shang. Energy-saving strategies of wireless sensor networks. In *Proceedings of the international Symposium on Microwave, Antenna, Propagation and EMC Technologies for Wireless Communications, 2007*, pages 178 – 181, August 2007, Hangzhou.

[45] Ines Slama, Badii Jouaber, and Djamal Zeghlache. Optimal power management scheme for heterogeneous wireless sensor networks: Lifetime maximization under qos and energy constraints. In *Proceedings of the Third International Conference on Networking and Services*, pages 69–, Washington, DC, USA, 2007. IEEE Computer Society.

[46] A.S.Zahmati N. M. Moghadam and B. Abolhassani. Epmplcs: An efficient power management protocol with limited cluster size for wireless sensor networks. In *Proceedings of the 27th International Conference on Distributed Computing Systems Workshops*, (ICDCSW'07), pages 69–72, 2007.

[47] Hung-Chin Jang and Hon-Chung Lee. Efficient energy management to prolong wireless sensor network lifetime. In *Proceedings of ICI 2007. 3rd IEEE/IFIP International Conference in Central Asia on Internet, 2007*, pages 1 – 4, Sept. 2007.

[48] C.Lin, Y.-X. He, and N. Xiong. An energy-effcient dynamic power management in wireless sensor networks. In *Proceedings of The Fifth International Symposium on Parallel and Distributed Computing*, ISPDC'06, 2006.

[49] Xue Wang, Junjie Ma, and Sheng Wang. Collaborative deployment optimization and dynamic power management in wireless sensor networks. In *Proceedings of the Fifth International Conference on Grid and Cooperative Computing*, GCC '06, pages 121–128, Washington, DC, USA, 2006. IEEE Computer Society.

[50] H. Wang, W. Wang, D. Peng, and H. Sharif. Optimal power management scheme for heterogeneous wireless sensor networks: Lifetime maximization under qos and energy constraints. In *Proceedings of the International Conference on Communication Systems, ICCS*, pages 1–5, 2006.

[51] A. Sinha and A. Chandrakasan. Dynamic power management in wireless sensor networks. *Design & Test of Computers, IEEE*, 18(2):62–74, August 2002.

[52] Sheetalkumar Doshi, Shweta Bhandare, and Timothy X Brown. An on-demand minimum energy routing protocol for a wireless ad hoc network. *Mobile Computing and Communications Review*, 6:50–66, 2002.

[53] Sheetalkumar Doshi and Timothy X Brown. Minimum energy routing schemes for a wireless ad hoc network, 2002.

[54] Qun Li, Javed Aslam, and Daniela Rus. Online power-aware routing in wireless ad-hoc networks, 2001.

[55] Suresh Singh, Mike Woo, and C. S. Raghavendra. Power-aware routing in mobile ad hoc networks. In *MobiCom '98: Proceedings of the 4th annual ACM/IEEE international conference on Mobile computing and networking*, pages 181–190, New York, NY, USA, 1998. ACM.

[56] Giuseppe Anastasi, Marco Conti, Mario Di Francesco, and Andrea Passarella. Energy conservation in wireless sensor networks: A survey. *Ad Hoc Networks*, 7(3):537–568, May 2009.

[57] Deepak Ganesan, Alberto Cerpa, Yan Yu, Deborah Estrin, Wei Ye, and Jerry Zhao. Networking issues in wireless sensor networks. *Journal of Parallel and Distributed Computing*, 64:799–814, 2003.

[58] S.Park J. Mina A. Warrier and I. Rheea. How much energy saving does topology control offer for wireless sensor networks? a practical study. *Elsevier/ACM Computer Communications*, 30 (14-15):2867–2879, October 2007.

[59] P. Santi. Topology control in wireless ad hoc and sensor networks, 2003.

[60] IEEE. Wireless lan medium access control (mac) and physical layer (phy) specifications, 1999.

[61] Wei Ye, John Heidemann, and Deborah Estrin. Medium access control with coordinated, adaptive sleeping for wireless sensor networks. *IEEE/ACM Transactions on Networking*, 12:493–506, 2003.

[62] Tijs van Dam and Koen Langendoen. An adaptive energy-efficient mac protocol for wireless sensor networks, 2003.

[63] Joseph Polastre, Jason Hill, and David Culler. Versatile low power media access for wireless sensor networks. In *SenSys '04: Proceedings of the 2nd international conference on Embedded networked sensor systems*, pages 95–107, New York, NY, USA, 2004. ACM.

[64] Crossbow Technologies Inc, San Jose, USA. *Datasheet of Mica-Z Wireless Measurement System*, 2005.

[65] ATMEL AVR. *ATmega128RFA1 8-bit Microcontroller with Low Power 2.4GHz Transceiver for ZigBee and IEEE 802.15.4 data sheet*.

[66] Nevine Aboughazaleh, Patrick Lanigan, Sameh Gobriel, Daniel Moss, and Rami Melhem. Dynamic rate-selection for extending the lifetime of energy-constrained networks, 2004.

[67] Anh Tuan Hoang and Mehul Motani. Buffer and channel adaptive modulation for transmission over fading channels. In *ICC'03 /IEEE Communications society*, 2003.

[68] Anh Tuan Hoang and Mehul Motani. Buffer and channel adaptive transmission over fading channels with imperfect channel state information. In *WCNC/IEEE Communications society*, pages 1891–1896, 2004.

[69] Anh Tuan Hoang and Mehul Motani. Cross-layer adaptive transmission: Optimal strategies in fading channels. *IEEE TRANSACTIONS ON COMMUNICATIONS*, 56(5), May 2008.

[70] Nicolas Christin and Jrg Liebeherr. A scalable service architecture for providing strong service guarantees. In *Proceedings of SPIE ITCOM Workshop on Scalability and Traffic Control in IP Networks II, Boston,USA*, 2002.

[71] Reu-Ching Chen and Chen-Sung Chang. Energy cost optimization by adequate transmission rate dividing in wireless communication system. *Informatica*, 19(2):191–200, 2008.

[72] Behnaam Aazhang and Ashutosh Sabharwal. Algorithms for wireless communication systems: A power-efficiency perspective. *Wirel. Pers. Commun.*, 26(2-3):217–226, 2003.

[73] Eugene Shih, Seong-Hwan Cho, Nathan Ickes, Rex Min, Amit Sinha, Alice Wang, and Anantha Chandrakasan. Physical layer driven protocol and algorithm design for energy-efficient wireless sensor networks. In *MobiCom '01: Proceedings of the 7th annual international conference on Mobile computing and networking*, pages 272–287, New York, NY, USA, 2001. ACM.

[74] Muddassir Iqbal Shakya Mukesh, Zhang Jianhual, Zhang Ping, and Inam-Ur-Rehman. Comparative analysis of m-ary modulation techniques for wireless ad-hoc networks. In *SAS, IEEE Sensors Applications Symposium,*, February 2007.

[75] Jamshid Abouei, Konstantinos N. Plataniotis, and Subbarayan Pasupathy. Green modulations in energy-constrained wireless sensor networks. *CoRR*, abs/1008.3776, 2010.

[76] Fengzhong Qu, Dongliang Duan, Liuqing Yang, and Ananthram Swami. Signaling with imperfect channel state information: A battery power efficiency comparison. *IEEE Transactions on Signal Processing*, 56(9):4486–4495, 2008.

[77] J. Joaquí Escudero Garzás, n Calzón, Carlos Bouso and Ana García Armada. An energy-efficient adaptive modulation suitable for wireless sensor networks with ser and throughput constraints. *EURASIP J. Wirel. Commun. Netw.*, 2007(1):23–23, 2007.

[78] Shuguang Cui, Andrea J. Goldsmith, and Ahmad Bahai. Energy-constrained modulation optimization. *IEEE Transactions on Wireless Communications*, 4:2349–2360, 2005.

[79] Ming-Chien Tseng Hsin-Piao Lin and Ding-Bing Lin. Performance analysis of m-ary psk adaptive modulation system over rayleigh-lognormal fading channel. *Vehicular Technology Conference, VTC 2005*, 1:576 – 580, 2005.

[80] G. Sudha Anil Kumar, Govindarasu Manimaran, and Zhengdao Wang. End-to-end energy management in networked real-time embedded systems. *IEEE Transactions on Parallel and Distributed Systems*, 19:1498–1510, 2008.

[81] Sanjay Shakkottai, R. Srikant, and Ness Shroff. Unreliable sensor grids: Coverage, connectivity and diameter. In *In Proceedings of IEEE INFOCOM*, pages 1073–1083, 2003.

[82] Chih fan Hsin and Mingyan Liu. Network coverage using low duty-cycled sensors: Random and coordinated sleep algorithms. In *abc*, pages 433–442, 2004.

[83] David Shuman and Mingyan Liu. Optimal sleep scheduling for a wireless sensor network node. In *Fortieth Asilomar Conference on Signals, Systems and Computers*, pages 1337 – 1341, 2006.

[84] Rick W. Ha, Pin-Han Ho, and Xuemin Shen. Optimal sleep scheduling with transmission range assignment in application-specific wireless sensor networks. *IJSNet*, (1/2):72–88, 2006.

[85] Abtin Keshavarzian, Huang Lee, and Lakshmi Venkatraman. Wakeup scheduling in wireless sensor networks. In *Proceedings of the 7th ACM international symposium on Mobile ad hoc networking and computing*, MobiHoc ’06, pages 322–333, New York, NY, USA, 2006. ACM.

[86] Samuel R. Madden, Michael J. Franklin, Joseph M. Hellerstein, and Wei Hong. Tinydb: An acquisitional query processing system for sensor networks. *ACM Trans. Database Syst*, 30:122–173, 2005.

[87] Phil Buonadonna, David Gay, Joseph M. Hellerstein, Wei Hong, and Samuel Madden. Task: Sensor network in a box. In *In Proceedings of European Workshop on Sensor Networks (EWSN 2005)*, January 2005.

[88] Wei Ye, John Heidemann, and Deborah Estrin. Medium access control with coordinated adaptive sleeping for wireless sensor networks. *IEEE/ACM Transactions on Networking*, 12,Issue:3:493–506, June 2004.

[89] Tijs Van Dam and Koen Langendoen. An adaptive energy-efficient mac protocol for wireless sensor networks. In *Proceedings of the 1st international conference on Embedded Networked Sensor Systems*, SenSys '03, pages 171–180, New York, NY, USA, November 2003. ACM.

[90] Andrew Wang, Seong-Hwan Cho, Charles Sodini, and Anantha Chandrakasan. Energy efficient modulation and mac for asymmetric rf microsensor systems. In *ISLPED*, pages 106–111, 2001.

[91] Intel. *Intel StrongARM SA-1110 Microprocessor, ADVANCE INFORMATION Brief Datasheet*, July 1999.

[92] Transmeta-corporation. *Tm5400 Crusoe processor specifications*.

[93] Intel. *Intel Xscale Microarchitecture*.

[94] Kihwan Choi, Wonbok Lee, Ramakrishna Soma, and Massoud Pedram. Dynamic voltage and frequency scaling under a precise energy model considering variable and fixed components of the system power dissipation. In *In Proceedings of the International Conference on Computer Aided Design*, pages 29–34, 2004.

[95] Vadim Gutnik and Anantha P. Chandrakasan. Embedded power supply for low-power dsp. *IEEE Trans. Very Large Scale Integr. Syst.*, 5:425–435, December 1997.

[96] Woonseok Kim, Dongkun Shin, Han-Saem Yun, Jihong Kim, and Sang Lyul Min. Performance comparison of dynamic voltage scaling algorithms for hard real-time systems. In *IEEE Real Time Technology and Applications Symposium*, pages 219–228, 2002.

[97] Johan Pouwelse, Koen Langendoen, and Henk Sips. Dynamic voltage scaling on a low-power microprocessor. In *Proceedings of the ACM International Conference on Mobile Computing and Networking, MobiCom*, pages 251–259, 2001.

[98] Ravindra Jejurikar, Cristiano Pereira, and Rajesh Gupta. Leakage aware dynamic voltage scaling for real-time embedded systems. In *In Proceedings of the Design Automation Conference*, pages 275–280, 2004.

[99] Ian Brynjolfson and Zeljko Zilic. Dynamic clock management for low power applications in fpgas. In *In Proc. of IEEE Custom Integrated Circuits Conference*, pages 139–142, 2000.

[100] Saraju Mohanty Ranganathan, N. Ranganathan, and V. Krishna. Datapath scheduling using dynamic frequency clocking. In *In Proceedings of the IEEE Computer Society Annual Symposium on VLSI*, pages 65–70, April 2002.

[101] Saraju P. Mohanty, N. Ranganathan, and Sunil K. Chappidi. Peak power minimization through datapath scheduling. In *In Proceedings of the IEEE Computer Society Annual Symposium on VLSI, Feb 2003*, pages 121–126, February 2003.

[102] Wanghong Yuan and Klara Nahrstedt. Integration of dynamic voltage scaling and soft real-time scheduling for open mobile systems, 2002.

[103] Marcus T. Schmitz and Bashir M. Al-hashimi. Energy-efficient mapping and scheduling for dvs enabled distributed embedded systems. In *Proceedings of the conference on Design, Automation and Test, DATE02*, pages 514–521, 2002.

[104] Ala Qadi and Steve Goddard. A dynamic voltage scaling algorithm for sporadic tasks. In *Proceedings of the IEEE International Real-Time Systems Symposium, RTSS*, pages 52–62, 2003.

[105] Padmanabhan Pillai and Kang G. Shin. Real-time dynamic voltage scaling for low-power embedded operating systems. In *Proceedings of the eighteenth ACM symposium on Operating systems principles*, pages 89–102, 2001.

[106] Ravindra Jejurikar. Abstract dynamic voltage scaling for systemwide energy minimization in real-time embedded systems, 2004.

[107] Marcus T. Schmitz and Bashir M. Al-Hashimi. Considering power variations of dvs processing elements for energy minimisation in distributed systems, 2001.

[108] Woonseok Kim, Jihong Kim, and Sang Lyul Min. Preemption-aware dynamic voltage scaling in hard real-time systems. In *ISLPED*, pages 393–398, 2004.

[109] Soraya Ghiasi, Jason Casmira, and Dirk Grunwald. Using ipc variation in workloads with externally specified rates to reduce power consumption. In *In Workshop on Complexity Effective Design*, 2000.

[110] Andreas Weissel and Frank Bellosa. Process cruise control: event-driven clock scaling for dynamic power management. In *Proceedings of the 2002 international conference on Compilers, architecture, and synthesis for embedded systems*, CASES '02, pages 238–246, New York, NY, USA, 2002. ACM.

[111] Kihwan Choi, Ramakrishna Soma, and Massoud Pedram. Fine-grained dynamic voltage and frequency scaling for precise energy and performance trade-off based on the ratio of off-chip access to on-chip computation times. In *In DATE 04: Proceedings of the conference on Design, automation and test in Europe*, page 10004. IEEE Computer Society, 2005.

[112] Kihwan Choi, Ramakrishna Soma, and Massoud Pedram. Off-chip latency-driven dynamic voltage and frequency scaling for an mpeg decoding. In *In DAC 04: Proceedings of the 41st annual conference on Design automation*, pages 544–549. ACM Press, 2004.

[113] F. Yao, A. Demers, and S. Shenker. A scheduling model for reduced cpu energy. In *Proceedings of the 36th Annual Symposium on Foundations of Computer Science*, FOCS '95, pages 374–, Washington, DC, USA, 1995. IEEE Computer Society.

[114] Youngsoo Shin and Kiyoung Choi. Power conscious fixed priority scheduling for hard real-time systems. In *Proceedings of the 36th annual ACM/IEEE Design Automation Conference*, DAC '99, pages 134–139, New York, NY, USA, 1999. ACM.

[115] Inki Hong, Gang Qu, Miodrag Potkonjak, and Mani B. Srivastava. Synthesis techniques for low-power hard real-time systems on variable voltage processors, 1998.

[116] Dongkun Shin, Jihong Kim, and Seongsoo Lee. Low-energy intra-task voltage scheduling using static timing analysis. In *Proceedings of the 38th annual Design Automation Conference*, DAC '01, pages 438–443, New York, NY, USA, 2001. ACM.

[117] Seongsoo Lee and Takayasu Sakurai. Run-time power control scheme using software feedback loop for low-power real-time applications. In *Proc. Asia-Pacific Design Automation Conference*, pages 381–386, 2000.

[118] Mark Weiser, Brent Welch, Alan Demers, and Scott Shenker. Scheduling for reduced cpu energy. In *Proceedings of the 1st USENIX conference on Operating Systems Design and Implementation*, OSDI '94, Monterey,California, pages 13–23, Berkeley, CA, USA, 1994. USENIX Association.

[119] Vlasios Tsiatsis, Scott Zimbeck, and Mani Srivastava. Architecture strategies for energy-efficient packet forwarding in wireless sensor networks. In *Proceedings of the 2001 international symposium on Low power electronics and design*, ISLPED '01, pages 92–95, New York, NY, USA, 2001. ACM.

[120] Mohammad Ali Khojastepour and Ashutosh Sabharwal. Delay-constrained scheduling: power efficiency, filter design, and bounds. In *IEEE INFOCOM, Hong Kong*, pages 7–11, 2004.

[121] Curt Schurgers, Olivier Aberthorne, and Mani B. Srivastava. Modulation scaling for energy aware communication systems. In *International Symposium on Low Power Electronics and Design*, pages 96–99, 2001.

[122] J. F. Hayes. Adaptive feedback communications. *IEEE TRANSACTIONS ON COMMUNICATIONS*, COM-16:2934, Feb. 1968.

[123] J. K. Cavers. Variable-rate transmission for rayleigh fading channels. *IEEE TRANSACTIONS ON COMMUNICATIONS*, COM-20:15–22, Feb. 1972.

[124] S. Sampei S.Otsuki and N. Morinaga. Square-qam adaptive modulation/ tdma/tdd systems using modulation level estimation with walsh function. *Electron. Lett.*, 31:169171, Feb. 1995.

[125] W. T. Webb and R. Steele. Variable rate qam for mobile radio. *IEEE Trans. Commun.*, 43:22232230, July 1995.

[126] H. Sasaoka Y. Kamio, S. Sampei and N. Morinaga. Performance of modulation-level-controlled adaptive-modulation under limited transmission delay time for land mobile communications. In *in Proc. IEEE VTC95,*, pages 221–225, July 1995.

[127] B. Vucetic. An adaptive coding scheme for time-varying channels. *IEEE Trans. Commun.*, 39:221225, May 1991.

[128] S. M. Alamouti and S. Kallel. Adaptive trellis-coded multiple-phasedshift keying for rayleigh fading channels. *IEEE Trans. Commun.*, 42:23052314, June 1994.

[129] S.Sampei T.Ue and N. Morinaga. Symbol rate and modulation level controlled adaptive modulation/tdma/tdd for personal communication systems. In *in Proc. IEEE VTC95,*, page 306310, July 1995.

[130] N. Morinaga H. Matsuoka, S. Sampei and Y. Kamio. Symbol rate and modulation level controlled adaptive modulation/tdma/tdd for personal communication systems. In *in Proc. IEEE VTC96,*, page 487491, Apr. 1996.

[131] Andrea J. Goldsmith. Variable-rate variable-power mqam for fading channels. *IEEE TRANSACTIONS ON COMMUNICATIONS*, 45(10):493–506, October 1997.

[132] Xiliang Zhong and Cheng-Zhong Xu. Energy-efficient wireless packet scheduling with quality of service control. *IEEE Transactions on Mobile Computing*, 6(10):1158–1170, 2007.

[133] Xiliang Zhong and Cheng-Zhong Xu. Delay-constrined energy-efficient wireless packet scheduling with quality of service guarantee. In *IEEE GLOBECOM '05. Saint Louis,MO*, pages 33–40, Dec. 2005.

[134] Yong Yuan, Zongkai Yang, and Jianhua He. An adaptive modulation scaling scheme for quality of services ensurance in wireless sensor netwoks, 2005.

[135] Zongkai Yang, Yong Yuan, Jianhua He, and Wenqing Chen. Adaptive modulation scaling scheme for wireless sensor networks. *IEICE Transactions*, 88-B(3):882–889, 2005.

[136] Anh Tuan Hoang and Mehul Motani. Cross-layer adaptive transmission with incomplete system state information. *IEEE Transactions on Communications*, 56(11):1961–1971, 2008.

[137] Action Nechibvute, Albert Chawanda, and Pearson Luhanga. Piezoelectric energy harvesting devices: An alternative energy. *Smart Materials Research, Hindawi Publishing Corporation*, 2012:13.

[138] Vijay Raghunathan, Curt Schurgers, Sung Park, Mani Srivastava, and Barclay Shaw. Energy-aware wireless microsensor networks. In *IEEE Signal Processing Magazine*, pages 40–50, 2002.

[139] Gregory J. Pottie and William J. Kaiser. Wireless integrated network sensors. *Commun. ACM*, 43(5):51–58, 2000.

[140] Mark Hempstead, Nikhil Tripathi, Patrick Mauro, Gu-Yeon Wei, and David Brooks. An ultra low power system architecture for sensor network applications. *SIGARCH Comput. Archit. News*, 33:208–219, May 2005.

[141] Victor Shnayder, Mark Hempstead, Bor-rong Chen, Geoff Werner Allen, and Matt Welsh. Simulating the power consumption of large-scale sensor network applications. In *Proceedings of the 2nd international conference on Embedded networked sensor systems*, SenSys '04, pages 188–200, New York, NY, USA, 2004. ACM.

[142] Hakan Aydin, Rami Melhem, Daniel Mossé, and Pedro Mejía-Alvarez. Power-aware scheduling for periodic real-time tasks. *IEEE Trans. Comput.*, 53:584–600, May 2004.

[143] Yang Yu, Bhaskar Krishnamachari, and Viktor K. Prasanna. Energy-latency trade-offs for data gathering in wireless sensor networks. In *In IEEE Infocom*, 2004.

[144] G. Sudha Anil Kumar, Govindarasu Manimaran, and Zhengdao Wang. End-to-end energy management in networked real-time embedded systems. *IEEE Trans. Parallel Distrib. Syst.*, 19(11):1498–1510, 2008.

[145] C.Yeh Z. Fan and R.X. Gao. Energy-aware data acquisition in wireless sensor networks. In *IEEE Instrumentation and Measurement Technology Conference*, 2007.

[146] Bo Zhang, Robert Simon, and Hakan Aydin. Energy management for time-critical energy harvesting wireless sensor networks. In *SSS*, pages 236–251, 2010.

[147] R. Simon B.Zhang and H. Aydin. Joint voltage and modulation scaling for energy harvesting sensor networks. In *Proceedings of the First International Workshop on Energy Aware Design and Analysis of Cyber Physical Systems (WEA-CPS'10), Stockholm, Sweden*, April 2010.

[148] Sheldon Ross. *Queueing Theory an Introduction to Probability models, 8th ed.*, pages 475–501. Elsvier, San Diego,CA, 2002.

[149] Kishor S. Trivedi. *Probability and Statistics with Reliability, Queuing and Computer Science Applications, second edition*, pages 405–554. John Wiley & Sons, Inc., 605 Third Avenue, New York, 2003.

[150] C. Callegari, R. Garroppo, S. Giordano, and G. Nencioni. A new markov model for evaluating the alr dual-threshold policy. In *GLOBECOM Workshops, 2009 IEEE*, pages 1 –5, 30 2009-dec. 4 2009.

[151] Chamara Gunaratne, Kenneth J. Christensen, and Stephen Suen. Ethernet adaptive link rate (alr): Analysis of a buffer threshold policy. In *GLOBECOM*, 2006.

[152] Xiao-Hui Lin and Yu-Kwong Kwok. On channel adaptive energy management in wireless sensor networks. In *Proceedings of ICPP Workshops*, pages 397–404, 2005.

[153] L. D. Servi and S. G. Finn. M/m/1 queues with working vacations (m/m/1/wv). *Performance Evaluation*, 50(1):41 – 52, 2002.

[154] Yutaka Baba. Analysis of a gi/m/1 queue with multiple working vacations. *Operations Research Letters*, 33(2):201 209, 2005.

[155] Chae.K.C. Kim.J.D.Choi.D.W. Analysis of queue length distributions of the m/g/1 queue with working vacations. In *Proceedings of the International Conference on Statistics and Related fields*, (ICDCSW'07), 2003.

[156] X. Tian N. Zhao and K. Wang. The m/m/1 queue with single working vacation. *International Journal Of Information and Management Sciences*, 4:621–634, 2008.

[157] De-An Wu and Hideaki Takagi. M/g/1 queue with multiple working vacations. *Perform. Eval.*, 63:654–681, July 2006.

[158] Julia Rose Mary.K. and Afthab Begum.M.I. M[x]/m/1 queue with working vacations. *Acta Ciencia India*, XXXVI(3):429 439, 2010.

[159] M.I. Afthab Begum M.Jemila Parveen. Analysis of the batch arrival mx/g/1 queue with exponentially distributed multiple working vacations. *International Journal of Mathematical Sciences and Applications*, 1:865–879, May 2011.

[160] Peter Dimopoulos, Panlop Zeephongsekul, and Zahir Tari. Modeling the burstiness of tcp. In *Proceedings of the The IEEE Computer Society's 12th Annual International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems*, pages 175–183, Washington, DC, USA, 2004. IEEE Computer Society.

[161] Michele Garetto and Don Towsley. Modeling, simulation and measurements of queuing delay under long-tail internet traffic. In *Proc. of ACM SIGMETRICS 2003*, pages 47–57, 2003.

[162] Gianfranco Ciardo, Jogesh Muppala, and Kishor S. Trivedi. On the solution of gspn reward models. *Performance Evaluation*, 12(4):237–253, Jul,1991.

[163] M. Ajmone Marsan, G. Balbo, G. Conte, S. Donatelli, and G. Franceschinis. Modelling with generalized stochastic petri nets. *SIGMETRICS Perform. Eval. Rev.*, 26(2):2–, August 1998.

[164] Robin Sahner, Antonio Puliafito, and Kishor S Trivedi. *Performance and reliability analysis of computer systems:an example-based approach using the SHARPE software package*. Kluwer, Boston, MA, 1996.

[165] Carl Adam Petri. Kommunikation mit automaten. *New York: Griffiss Air Force Base, Technical Report RADC-TR-65-377*, 1:1–Suppl.1, 1966. English translation.

[166] Tadao Murata. Petri nets: Properties, analysis and applications. In *Proceedings of the IEEE*, volume 77, pages 541–580, Apr. 1989.

[167] James Lyle Peterson. *Petri Net Theory and the Modeling of Systems*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1981.

[168] Yue Ma, James J. Han, and Kishor S. Trivedx. Channel allocation with recovery strategy in wireless networks. *European Transactions on Telecommunications*, 11(4):395–406, 2000.

[169] Reinhard German. *Performance Analysis of Communication Systems with Non-Markovian Stochastic Petri Nets*. John Wiley & Sons, Inc., New York, NY, USA, 2000.

[170] M.K.Molloy. *On the integration of Delay and Throughput Measures in Distributed Processing Models*. PhD thesis, UCLA, Los Angeles,CA,USA., 1981.

[171] E. Tavares, P. Maciel, P. Dallegrave, B. Silva, T. Falco, B. Nogueira, G. Callou, and P. Cunha. Model-driven software synthesis for hard real-time applications with energy constraints. *Design Automation for Embedded Systems*, 14:327–366, 2010.

[172] E. Tavares, P. Maciel, and B. Silva. Modeling hard real-time systems considering inter-task relations, dynamic voltage scaling and overheads. *Microprocessors and Microsystems*, 32(8):460–473, 2008.

[173] R. Barreto, P. Maciel, E. Tavares, R.D. Freitas, M. Oliveira, and R.M. Lima. A time petri net-based method for embedded hard real-time software synthesis. *Design Automation for Embedded Systems*, 12(1):31–62, 2008.

[174] B. Nogueira, P. Maciel, E. Tavares, E. Andrade, R. Massa, G. Callou, and R. Ferraz. A formal model for performance and energy evaluation of embedded systems. *EURASIP Journal on Embedded Systems*, 2011:2, 2011.

[175] G. Callou, P. Maciel, E. Andrade, B. Nogueira, and E. Tavares. Estimation of energy consumption and execution time in early phases of design lifecycle: an application to biomedical systems. *Electronics Letters*, 44(23):1343–1344, 2008.

[176] M. Junior, S. Neto, P. Maciel, R. Lima, A. Ribeiro, R. Barreto, E. Tavares, and F. Braga. Analyzing software performance and energy consumption of embedded systems by probabilistic modeling: An approach based on coloured petri nets. *Petri Nets and Other Models of Concurrency-ICATPN 2006*, pages 261–281, 2006.

Index