Design Considerations for Medium Access Control in Resource Constrained Embedded Wireless Networks

Von der Fakultät für Elektrotechnik und Informationstechnik der Rheinisch-Westfälischen Technischen Hochschule Aachen zur Erlangung des akademischen Grades eines Doktors der Ingenieurwissenschaften genehmigte Dissertation

vorgelegt von

Junaid Ansari, M.Sc.
aus Karlsruhe, Deutschland

Berichter: Univ.-Prof. Dr. Petri Mähönen
Univ.-Prof. Dr.-Ing. Gerd Ascheid

Tag der mündlichen Prüfung: 12 September 2012

ABSTRACT

Recent years have experienced a huge influx of daily life applications based on embedded wireless networks. While new applications with more demanding requirements and challenging deployment conditions are being explored, most of the existing networks suffer from communication deficiencies, inefficient use of resources, and inability to satisfy desired quality of service requirements. In order to carry on exploitation and exploration of embedded wireless networks in different daily life applications, there is a need for bridging the widening gulf between growing application demands and capabilities of the embedded networking solutions.

Medium Access Control (MAC) protocols play a critically important role in embedded wireless communication and heavily influence the performance characteristics of a network. The increasing popularity of embedded wireless networks demand new MAC solutions that best suit to the application requirements and the deployment conditions. Vast literature exists on the fundamentals of MAC designs especially on their theoretical underpinnings. One of the biggest concerns for many researchers and practitioners is that a large number of the proposed MAC schemes when implemented in real applications often fail to provide the behaviour as seen in analytical and simulation studies. The underlying reasons mainly include lack of practical insights due to unavailability of prototype implementation and unrealistic deployment assumptions.

This dissertation aims at bridging the gap between theory and practice of MAC protocols for embedded wireless networks through development of new protocols and analysis of existing solutions. The work mainly targets wireless sensor- and cognitive wireless networks. It investigates MAC design aspects with high practical significance: energy efficiency, spectrum agility, runtime reconfigurability and cross-layer methodologies. A novel solution of combining radio-triggered wake-ups with MAC protocols for extremely low-power operation and a specialized dual-radio based MAC design for highly energy efficient communication has been developed. An innovative MAC protocol is introduced for spectrum agility and distributed dynamic channel access in resource constrained embedded wireless networks. For enabling runtime reconfiguration to MAC solutions and fast prototyping, a component oriented framework is developed. Different cross-layer design approaches are investigated for MAC-routing stack optimization and achieving higher network efficiency. All the concepts introduced in this work have been validated through prototype implementation and extensive performance evaluation on real node testbeds under realistic application conditions. Furthermore, comparative empirical studies of the proposed solutions with existing state-of-the-art designs are conducted under the same traffic, network and spectral conditions. The experimental results indicate that the developed MAC protocols have high relevancy to real-world applications and deployment scenarios.

KURZFASSUNG

Die vergangenen Jahre zeichnen sich durch einen enormen Zuwachs an Applikationen aus die auf eingebetteten Drahtlosnetzwerken basieren. Immer anspruchsvollere Applikationen werden in schwierigen Einsatzszenarien erprobt und enthüllen, dass bestehende Kommunikationsnetze Defizite aufweisen, Ressourcen nur ineffizient nutzen und nicht die gewünschte Service-Qualität liefern können. Um die Ausnutzung und Erforschung von eingebetteten Drahtlosnetzwerken für verschiedene alltägliche Zwecke voranzutreiben, müssen Brücken geschlagen werden, die die Kluft zwischen steigenden Anforderungen an Applikationen und den Möglichkeiten eingebetteter Netzwerklösungen überwinden.

Medienzugriffsprotokolle (MACs) spielen eine kritische und wichtige Rolle für eingebettete drahtlose Kommunikation, da sie einen großen Einfluss auf die Leistungscharakteristik eines Netzwerkes haben. Die Popularität von eingebetteten Drahtlosnetzwerken verlangt nach neuen maßgeschneiderten MAC-Lösungen für den jeweiligen Einsatzzweck und die vorgefundenen Umweltbedingungen. Bereits heute gibt es umfangreiche Erkenntnisse über die grundlegenden Entwurfstechniken für MACs, hierbei insbesondere über deren theoretischen Unterbau. Forscher und Praktiker betrachten allerdings mit Sorge, dass viele der vorgeschlagenen MAC-Lösungen im praktischen Einsatz nicht die in Analysen und Simulationen prognostizierten Verhaltensmuster aufweisen. Grund für diese Diskrepanz sind häufig ein Mangel an praktischen Erfahrungen auf Grund fehlender Prototypen und unrealistische Annahmen über die geplante Nutzung.

Diese Dissertation versucht, eine weitere Spaltung zwischen Theorie und Praxis der MAC-Protokolle für eingebettete Drahtlosnetzwerke zu verhindern, indem sie innovative Entwurfstechniken vorschlägt und existierende Lösungen analysiert. Die Arbeit bezieht sich dabei hauptsächlich auf drahtlose Sensornetzwerke und kognitive Drahtlosnetzwerke. Wir untersuchen Aspekte des MAC-Entwurfs mit hoher praktischer Relevanz: Energieeffizienz, Spektrumagilität, die Fähigkeit zur Laufzeitkonfiguration und ebenenübergreifende Methoden. Wir schlagen eine neuartige Lösung vor, die funkempfängergesteuertes Aufwecken mit MAC-Protokollen für hochgradig stromsparenden Betrieb verbindet und ein spezielles Design für hocheffiziente, energiesparende Kommunikation auf Basis eines dual ausgelegten Funksystems. Ein innovatives MAC-Protokoll für agile Spektrumsnutzung und verteilten dynamischen Kanalzugriff in ressourcenbeschränkten eingebetteten Drahtlosnetzwerken wird vorgestellt. Um eine Konfiguration zur Laufzeit und einen schnellen Prototypenentwurf zu ermöglichen, wird ein komponentenorientiertes Framework entwickelt. Verschiedene ebenenübergreifende Ansätze zur Optimierung von Sicherungs- und Vermittlungsschicht werden im Hinblick auf höhere Netzwerkeffizienz untersucht. Alle in dieser

Arbeit vorgestellten Konzepte wurden durch prototypische Implementationen und umfangreiche Leistungsanalysen in echten Prüfaufbauten unter realistischen Einsatzbedingungen überprüft. Desweiteren wurden vergleichende empirische Studien der vorgeschlagenen Lösungen mit existierenden aktuellen Designs unter gleichen Verkehrs-, Netzwerk- und spektralen Bedingungen durchgeführt. Die experimentellen Ergebnisse deuten darauf hin, dass die von uns vorgeschlagenen MACs eine hohe Relevanz für praktische Applikationen und Einsatzszenarien haben.

ACKNOWLEDGMENTS

I am grateful to Prof. Dr. P. Mähönen for giving me the scientific freedom and encouragement in pursuing my ideas. He dedicated his time and energy to advise, guide and support me throughout this work. He taught me how to analyze problems at a much deeper level and view things in a wider context. Light and friendly discussions with him on the intricacies of research have always been very inspiring and motivating for me. I am also very thankful to him for providing a pleasant research atmosphere.

I would like to thank Prof. Dr.-Ing. G. Ascheid for his interest in co-examining this work. His insightful feedback has certainly been useful.

I would like to especially thank Xi Zhang for a long collaboration and enthusiasm on various research topics. The work would have been considerably less fun without her scientific challenges and useful discussions.

Some of the results could have not been gathered without the support of my students. I would like to thank Dmitry Pankin, Obaid Salikeen, Tobias Ang, Wasif Masood and Arham Muslim for their contributions.

I would like to thank my colleagues and friends who provided an enjoyable and friendly working environment. The international diversity of the institute has always been refreshing to work. I have learnt many new technical skills from Elena Meshkova, Aleksandar Kovacevic and Janne Riihijärvi.

I would also like to thank Gero Schmidt-Kärst for helping me order various hardware components. His technical support as a system administrator and timely preparing work stations for diverse experimental needs had been quite helpful.

I am thankful to RWTH Aachen University, European Union (WASP-IST-034963 and 2PARMA-FP7-ICT-2009-4-248716 projects) and German Research Foundation's UMIC research centre for providing necessary funding grants. Many experimental studies and prototypes developed in this work would have not been possible without the equipment grants. Also these institutions provided the travel budget to various conferences for presenting the achieved results and demonstrating the developed prototypes.

I would also like to thank our family friends, Mrs. & Prof. Dr. M. D. Zeidler for their encouragement during my MSc studies and PhD research.

I am extremely fortunate to have been blessed with a very loving family. My wife Saman has been very supportive to me in finishing this work. My sisters, Sarah and Sadia have always been very encouraging to me throughout my studies. I am grateful to my mother Azra Yasmin and my father Prof. Dr. M. S. Ansari for their love, keen interest and continuous support during my PhD research. They have always been a source of inspiration for me throughout my education and research. I dedicate this dissertation to them.

CONTENTS

1

INTRODUCTION

This dissertation focuses on the design considerations of Medium Access Control (MAC) protocols for resource constrained embedded wireless networks. It presents a number of new MAC protocols including their prototype implementations that are tested on real hardware testbeds. It also describes a new toolchain that has been developed for rapid prototyping and design of MAC protocols.

1.1 MOTIVATION

During the last decade we have witnessed an emergence of many daily life applications and services that are based on embedded wireless networking. We expect that this trend will become even more prominent in this decade. Low-power embedded networks enable us to monitor, control and interact with physical world at a granularity that was never perceived before. This strong technological drive towards self-configurable autonomous ubiquitous embedded networks has certainly broadened the horizon of the low-power applications and services. While new application areas are being explored, most of the existing embedded wireless networks suffer from communication inefficiencies, suboptimal use of resources, and failure to provide the desired performance characteristics. There is an urgent need of bridging the widening gulf between the growing application demands and the capabilities of the low-power communication solutions.

A MAC protocol plays a pivotal role in network communication as it controls the wireless channel access to a node. An inefficient MAC scheme can lead to heavy packet losses, low throughput, high data communication latency, and significant energy and bandwidth wastage. The requirements and deployment conditions vary heavily across different applications and have a direct impact on the volume and pattern of the demanded traffic loads, network size, dynamics, and the spectral conditions in a network. There is no universally optimal MAC solution that suits well to all the applications [1]. Although many different designs [2–5] have been proposed in the past few years, many of them lack insights on real implementation and deployment experience. One of the biggest dilemmas faced by researchers and practitioners is that many of the existing MAC proposals simply fail to provide the expected performance characteristics when implemented and deployed in real world applications [6]. The emerging new application areas and existing network problems demand even more innovative MAC designs. The resource constrained nature of embedded wireless networks makes the MAC design research extremely challenging and exciting.

In order to realize applications and make network deployments viable, not only a better practical understanding of existing MAC protocols is important, but also new solutions are required which address the open research and engineering issues on energy consumption, spectrum agility, reconfigurability and cross-layer design through theoretical and experimental studies. This is necessary to carry on the exploitation pace of embedded wireless networks in different daily life applications and further explore new areas and services.

Embedded wireless applications require sustainable network communication over extended periods of time. Since wireless nodes are generally battery powered, energy efficient autonomous operation naturally becomes one of the biggest considerations. A number of energy efficient MAC schemes have been proposed during the past five years. Unfortunately, only a few techniques have been prototyped and evaluated on testbeds. Furthermore, most of the studies tend to ignore that besides energy conservation, other metrics such as the offered latency, data reliability and more importantly their trade-offs are of high practical significance from an application perspective. With the ever growing network traffic, longer lifetime demands and the requirements for a minimum affordable latency, there is a need for new and non-classical MAC protocols. This would drastically reduce the energy consumption budget while meeting high data reliability and low latency demands, and thus make application deployments viable.

The number of wireless networks and networked consumer applications is increasing rapidly. Most of the wireless devices operate in ISM frequency bands in a non-cooperative manner. This leads to mutual interference among devices, which eventually causes a detrimental effect on the performance characteristics [7, 8]. The performance degradation becomes even more evident for resource constrained embedded networks, which remain handicapped in competing against less resource constrained systems. Emergence of multiple wireless networks in shared frequency bands naturally requires spectrally efficient operation and coexistence of different technologies and networks. Although some researchers acknowledge the importance of this challenging design aspect, only a little has been done so far towards prototyping of spectrally efficient MAC schemes that enable dynamic channel access for interference mitigation and facilitating symbiotic coexistence of different networks.

One of the key challenges in the design of MAC protocols for embedded wireless networks is the support for flexibility and reconfiguration. Static MAC solutions often remain inefficient to satisfy the changing requirements on network, traffic, and spectral conditions. Runtime reconfiguration and granular control over the protocol parameters is demanded by adaptive wireless systems to adequately address the various quality of service demands and efficiently manage the network resources. Flexible and reconfigurable MAC design for embedded wireless networks still lacks both theoretical and experimental studies.

Although MAC protocols play a critically important role in determining the overall network performance metrics and therefore designing efficient MAC solutions is highly important, looking just at MAC protocols in isolation is incomplete from a practical point of view. Classical OSI-layered design of a protocol stack restricts information sharing and cooperation among different protocols. This leads to inefficient use of network resources and sub-optimal performance characteristics [9]. Cross-layer network design and cooperatively optimizing different protocol layers has therefore received considerable attention in recent years. Only a few schemes have, however, been deployed on real nodes and most of the investigations are confined to analytical and simulation stages. Furthermore, the inter-dependencies of protocols, parameters, and their relationship to application demands have not been empirically studied. Cross-layer design approaches for embedded wireless networks and their comprehensive performance evaluation in realistic deployment conditions are still open research and engineering topics. A thorough investigation of these issues is necessary for optimizing the use of resources and increasing network efficiency.

This dissertation provides insights into medium access techniques and addresses the key open research issues through theoretical as well as practical studies. It introduces new MAC protocols that are aimed for real world applications in order to narrow the gap between theoretical designs and their behaviour in practice. A special focus has therefore been given to prototype implementation of the proposed schemes and their empirical performance evaluation in realistic deployment conditions.

1.2 MAJOR CONTRIBUTIONS

This dissertation provides an extensive study on MAC design for resource constrained wireless networks. Many new techniques and protocols have been introduced in this work. Besides theoretical analysis, the proposed MAC schemes have been prototyped and extensively evaluated on real node testbeds under realistic application conditions in order to obtain meaningful results. In the following, the key design issues addressed in this dissertation are briefly described:

Energy Awareness

Many power-aware MAC protocols have been proposed during the past decade. Different schemes suit better to different network traffic patterns. We have designed a MAC protocol, *TrawMAC*, which combines the strengths of various MAC approaches and offsets their weaknesses according to the traffic conditions in a network. Since TrawMAC benefits from various design techniques and makes an energy efficient selection depending upon the traffic in a network, it satisfies the requirements of a wide range of applications. We have analyzed the activities at a radio interface in embedded wireless networks and proposed a dual radio platform solution for achieving high energy efficiency. One of the radios is tailored to low power channel sensing operation while the other efficiently handles bursty traffic. Furthermore, we have designed a novel protocol, *MR-MAC*, based on the dual radio platform, which is able to significantly lower the energy consumption by effectively handling these two operations at the appropriate radio interfaces. The MAC protocol has been shown to clearly outperform other state-of-the-art single radio based MAC solutions in identical application scenarios. We have designed, *RTWAC*, an innovative scheme of combining radio triggered wakeups with classical low-power MAC protocols for simultaneously achieving extremely energy efficient operation and low-latency response. We have developed a wake-up circuit board, which is attached externally to a wireless node. The on-board radio is completely switched off while the microcontroller is put to a low power mode. Instead of just a tonal signal, data is sent in the wake-up signal including the address of a node, which allows only the destination node to wake up. Having addressing mechanism helps in avoiding significant amount of energy wastage in undesired wake ups of the non-addressed nodes. Rather than using the wake-up radio, normal on-board radio with a low-power MAC protocol is used for data communication to ensure reliability and fast data rates. Hence RTWAC aggregates the benefits of radio triggered wake-ups and low-power MAC protocols. This results in achieving remarkably high energy conservation while providing low latency for data communication. We have carried out extensive performance evaluation and comparative studies of the prototype implementation of our protocols with existing state-of-the-art schemes in identical experimental conditions derived from real application scenarios. Instead of just considering the energy expenditure, we have looked into the trade-offs of energy consumption, latency and packet delivery ratio in realistic application conditions while evaluating our prototypes. The results indicate that our protocols achieve significantly higher performance gains. Our prototype hardware platforms consist of commercially available components, which make them readily useable for technology transfer in existing applications.

Spectrum Agility

This dissertation introduces spectrum agility and coexistence for low-power embedded wireless networks. We have developed *SA-MAC* – a light-weight spectrum agile MAC protocol, which allows nodes to dynamically find interference minimal channels in unregulated crowded frequency bands and enable reliable communication. To the best of our knowledge, our work has been the first attempt of designing, implementing and evaluating a spectral efficient MAC protocol for low-power embedded wireless networks in realistic spectrum conditions [10]. In this dissertation, we will also present an innovative cognitive MAC protocol, *CogMAC*, for infrastructureless networks. CogMAC allows nodes to opportunistically select an available channel for data communication using a distributed channel selection algorithm based on heuristics. Empirical performance evaluation on a commercially available Software-Defined-Radio (SDR) platform indicates that the MAC protocol efficiently utilizes spectrum holes thereby maintaining a high throughput and successful packet delivery ratio even in congested wireless spectrum. Our work gives a better understanding on the practical aspects of cognitive radio MAC protocols while most of the current state-of-the-art designs are limited to a theoretical and simulation stage. Identifying potential interferers is of fundamental importance to any network operating in a crowded uncoordinated frequency band. IEEE 802.11 networks are the major source of interference in the 2.4 GHz ISM frequency band where most of the low-power networks based on the IEEE 802.15.4 radios operate [11]. We have developed an algorithm, *WiSpot*, based on synchronized channel sensing using the IEEE 802.15.4 radios to reliably identify the IEEE 802.11 channels. Empirical performance comparison studies indicate that our scheme outperforms existing approaches in terms of the detection reliability and detection duration.

Reconfigurability

We have introduced a component based framework (*Decomposable MAC Framework*) that allows runtime reconfiguration of MAC protocols. The key MAC functional components are identified as building blocks so that a particular MAC solution can be realized by simply binding the components together. We advocate software-hardware co-design approach to meet the timing requirements while achieving the desired level of flexibility for spectrum agile and cognitive MAC protocols. A toolchain is designed to allow composition of MAC protocols at runtime. As part of our MAC toolchain, a domain specific language and a corresponding meta-compiler is developed, which allows expressing a MAC protocol with just a few lines, and enables its autonomous realization and execution. The toolchain assisted approach gives a new dimension to MAC optimization by composing a suitable MAC solution at runtime as per the application requirements and the network conditions. We have shown in this work that this concept is realizable as a real-time framework on commercially available SDR platforms and enables runtime MAC optimization.

Fast Prototyping

With the growing popularity of applications based on embedded wireless networks, there is a high demand of customized MAC solutions and fast prototyping. Following the component oriented design methodology of Decomposable MAC Framework, we have developed a tool, *MAC-PD*, which allows users to design a MAC protocol in the form of flowcharts – without requiring prior knowledge of programming languages and underlying hardware platform. The code for a selected target platform is auto-generated. The resulting MAC protocols exhibit similar performance characteristics compared to monolithic hand-coded implementations. We have confirmed this by comparing results for well-known MAC protocols. MAC-PD makes MAC development easier and facilitates fast prototyping of MAC protocols.

Cross-Layer Design and Optimization

We have developed *CONFab* (Component based Optimization for Networks), a framework, which assists optimal selection of a protocol stack at the pre-deployment phase for a particular application scenario. It uses comprehensive empirical basis and user experience to provide the optimal selection of a MAC-routing stack and protocol parameters settings according to the application requirements. It has the ability to improve its knowledge-base based on the performance feedback extracted from an already deployed network. Cross validation results from habitat monitoring application [12] show that CONFab is able to provide an appropriate selection of a MAC-routing stack and protocol parameters settings. Inspired by our earlier work on Decomposable MAC Framework, we have extended the component based implementation methodology to routing protocols. We have realized a scheme for cross-layer design and optimization of MAC-routing stack at runtime. In particular, our approach combines common functionalities among protocols and aims at composing the best possible combination of MAC-routing components depending upon the subjected traffic and network conditions. In order to apply optimization decisions, extensive performance measurements of over 700 hours are conducted on a large scale testbed [13] to comprehensively study the characteristics of well-known MAC and routing protocols. Instead of monolithic protocols, component based composite MAC-routing stack is composed. This allows the possibility of adapting the MAC-routing behaviour at runtime through appropriate combinations of components in a cross-layer fashion. Our approach of optimizing the behaviour of a network through composite MAC-routing stack shows an increased lifetime efficiency of over 40 % while maintaining similar packet delivery ratio as compared to a classical layered protocol stack implementation.

Most of the contributions and results reported in this dissertation have already been published in book chapters, international journals, and peer-reviewed international conferences. The list of relevant publications can be found separately at the end of the dissertation. Some of the MAC implementations and related tools for TelosB nodes [14] and WARP SDR platform [15] have been made available to different partner institutions as listed in Appendix C.

1.3 DISSERTATION OUTLINE

Each chapter is dedicated to a separate practical MAC design aspect in embedded wireless networks. We present the importance of the studied design issue, related research work, proposed solutions and their performance evaluation, and finally their empirical comparison with existing proposals in each chapter. The dissertation is structured as follows. Chapter 2 discusses the energy efficiency aspect for embedded wireless networks. It presents our developed energy efficient medium access schemes, *TrawMAC*, *MR-MAC* and *RTWAC* and the performance comparison with other state-of-the-art solutions. Chapter 3 focuses on spectral efficiency and describes the need for symbiotic coexistence of embedded wireless networks in a shared frequency band. We introduce a spectrum agile MAC protocol, *SA-MAC*, which allows nodes to dynamically find interference minimal channels in congested wireless spectrum. We also present a decentralized cognitive MAC protocol, *CogMAC*, which uses a distributed algorithm based on heuristics for dynamic channel access. Finally, we present *WiSpot*, a dual IEEE 802.15.4 based platform, which uses a synchronized channel sensing algorithm for reliable and fast detection of IEEE 802.11b/g based networks. Chapter 4 describes *Decomposable MAC Framework*, which is based on component oriented design philosophy. Using reusable MAC components as building blocks, the framework allows composition of a MAC solution at runtime. We also describe the developed interactive graphical user interface based tool, *MAC-PD*, which facilitates designing a MAC protocol by simply connecting components in the form of flowcharts in a user-friendly graphical environment. The MAC protocol code for a selected target platform is auto-generated and downloaded. Empirical evaluation indicates that while allowing fast prototyping, the auto-generated codes for MAC protocols developed through MAC-PD show similar performance characteristics as their monolithic hand-coded counterparts. Chapter 5 describes different cross-layer design approaches for PHY-MAC and MAC-routing optimization. In particular, *CONFab* framework is presented, which allows the selection of an appropriate MAC-routing stack and protocol parameters settings at a pre-deployment phase for a user-specified application scenario. Its cross-validation through a detailed empirical case-study is described. Furthermore, a component based MAC-routing cross-layer design is investigated and a runtime optimization methodology of MAC-routing stack is applied. We present comprehensive performance evaluation of our approach using well-known MAC and routing protocols. Chapter 6 summarizes the key results of this dissertation and outlines some of the future work directions.

2

ENERGY AWARENESS

Low power MAC protocols have received considerable attention in the past decade. This is owing to the fact that most of the embedded wireless networks consist of battery powered nodes. Majority of the existing MAC proposals for energy efficient operation lack prototype implementations and hence insights into the implementation behaviour in realistic traffic and network conditions are missing. The simulations and analytical models often fail to capture the effect of wireless channel characteristics and make unrealistic assumptions on node synchronization, processing bottlenecks and hardware failures [16]. In order to bridge this gap between theory and practice and to provide a deeper understanding of the low-power MAC designs, we have focused on the practical issues in real-world applications. We have given special attention to the prototype implementation of our energy efficient protocols on real node testbeds. We have also carried out a comprehensive empirical performance evaluation in realistic deployment scenarios to study the trade-offs among different metrics. Section 2.1 discusses the key characteristics of resource constrained embedded networks, sources of energy wastage and the design philosophy followed in low-power MAC protocols. Section 2.2 gives a short overview on the design trends for low-power MAC protocols and provides comparative analysis of different approaches. Section 2.4 discusses the design and performance evaluation of our multi-mode MAC protocol. Section 2.5 presents our multi-radio MAC protocol design, its prototype implementation and performance evaluation. Section 2.6 describes the prototype implementation and performance evaluation of our radio triggered wake-up scheme. Finally, a short summary of the chapter is given in Section 2.7. This chapter is mainly based on our articles [4, 17–20], which were published during this work.

2.1 CONCEPT AND IMPORTANCE

MAC protocols and Link Layer Control (LLC) protocols are responsible for reliable and efficient transfer of information across physical links. MAC protocols manage the activities at a radio interface and coordinate different nodes for accessing the shared communication medium. Since wireless medium is inherently broadcast, lack of coordination among different nodes can lead to concurrent data transmissions, which eventually cause packet collisions and loss of data. The lost packets are needed to be retransmitted thereby causing energy wastage, increased latency, reduced effective throughput, and overall lower channel utilization. MAC protocols therefore play an extremely important role for ensuring reliable data delivery and an increased overall network communication efficiency. In resource constrained networks and battery operated embedded Wireless Sensor Networks (WSNs), radio communication typically bears the highest energy consumption budget, while the cost for sensing[1] and computation generally remain low [24]. Since MAC protocols directly control a radio transceiver, they become pivotal in determining energy consumption at a node and hence the lifetime of a network.

In embedded wireless networks, a MAC protocol design is heavily influenced by network characteristics and application requirements. The broad range of applications include military domain and battlefield surveillance [25], environment and habitat monitoring [12], health care [26], home automation [27], traffic control [28], etc. In most of the cases, nodes are battery powered and can be deployed at remote locations in large numbers. This makes battery service or replacement unpractical. Embedded wireless networks are often expected to be in autonomous operation for a few years in order to make them economically viable. Due to the limited energy supply at nodes and the requirement for a long lifetime, power efficient operation becomes one of the primary objectives for MAC designers [4]. Although, energy conservation aspect is dominant, the application requirements on latency, throughput, and reliability still needs to be satisfied. This makes the design of MAC protocols for low-power embedded networks a unique and challenging task.

Before going into the details of different energy efficient MAC design approaches, we list the typical characteristics of embedded wireless networks and the major sources of energy wastage.

2.1.1 Characteristics of Resource Constrained Embedded Wireless Networks

Resource constrained embedded wireless networks exhibit following characteristics.

- Networks consist of a large number of battery powered nodes geographically dispersed in an ad hoc fashion with unstructured topologies. Owing to the battery constraints, nodes use low transmit power levels resulting in short communication range and a high density deployment of nodes in a certain geographical area.

- The average data generation rate is typically low, which leads to long idle duration of radio. Also, some of the applications (for instance, monitoring applications, etc.) have event driven nature, i.e., when an event of interest occurs, high data rates are generated, which result in intermittent traffic bursts.

- Networks are dynamic because of the time varying characteristics of radio environment, mobility in a network, existing nodes dying out or new nodes joining into the network.

[1]The energy consumption related to sensing activities is independent of the network protocol stack. We have looked into the information theoretical aspects of the sensed data and MAC designs [21] but this topic is beyond the scope of this chapter. We refer the reader to [22, 23] for power consumption aspects of different sensors.

- Most of the low-power embedded networks are dedicated to a specific task, which means that different nodes in a particular network cooperate with each other for servicing a single application. Application level performance metrics therefore become more important than the performance characteristics of an individual node.

- Low transmit power in resource constrained embedded networks results in short-ranged links and multihop communication. The end-to-end packet delivery ratio decreases as increasing number of hops in a network. This not only increases the overall latency for data transport but also wastes high amount of energy. Therefore, guaranteed data delivery from a MAC protocol is desired.

- Long idle periods in many low-power embedded applications give opportunities for energy conservation by putting the nodes in dormant state when they are idle. This operation of periodically putting a radio in sleep mode is known as the *duty cycled*[2] behaviour. Since nodes in a network collaboratively work for a single dedicated application, the overall network lifetime becomes the prime consideration instead of the battery lifetime of individual nodes. The duty cycle of a MAC protocol is defined as the ratio of the time when the radio is in *active* state to the sum of the durations when the radio is active and when it is *inactive*. A low duty cycle value implies higher energy savings while a higher value means longer latency for data communication. This implies that a trade-off between energy consumption and latency for data communication exist in duty cycling based MAC protocols.

- Duty cycling requires that nodes have their active periods aligned to each other for data communication. Different strategies have been designed to coordinate the active periods of nodes in different MAC protocols. Each duty cycling MAC scheme has its unique associated overhead for control and signaling.

2.1.2 Sources of Energy Wastage

Although sensing and data processing also contribute to the overall power expenditure, radio communication heavily dominates the energy consumption budget. Ye *et al.* have identified that the major sources of energy wastage in MAC protocols are idle listening, packet collisions, packet overhearing, and control packet overhead [29].

---

[2]The duty cycle can be expressed as $D = \frac{\tau}{T}$, where D is the duty cycle, and τ is the time duration for which the radio is in the listening/active mode. The symbol T denotes the period after which the listen/sleep cycle is repeated and is also known as the channel polling duration or the channel check interval. In order to quantify a duty cycle value, either T or τ should be known. The duty cycle is generally expressed in percentage. The active state is also known as *ON* or *listening* state of a radio. A node is able to transmit and receive packets only in this state. Active state is characterized by significantly high power consumption as compared to the inactive state. the inactive state is also known as *OFF*, *power-down* and *sleep* state. A node is unable to transmit and receive packets in this state. In the inactive state, a radio consumes only a negligible amount of power.

Idle Listening is the most significant source of energy wastage. It refers to the situation when a receiver listens to an idle channel in anticipation of upcoming packets. The most effective approach to reduce energy wastage in idle listening is by turning off the radio during the idle duration. Three main approaches are used to achieve this goal: 1) Time Division Multiple Access (TDMA), 2) scheduled contention, and 3) channel polling. TDMA avoids idle listening by assigning different time slots to different nodes. Nodes are active only in their assigned time slots and are put to sleep during the irrelevant slots. Scheduled contention shares some similarities with the TDMA principle. All the nodes are scheduled to wake up at the same time in order to listen and/or contend for channel. Idle listening is reduced since nodes know when other nodes transmit. Channel polling, on the other hand, refers to the practice where nodes wake up asynchronously for a short period of time and sense the channel for an activity. If the channel is found to be idle, nodes are put back to sleep.

Packet Collisions occur when two or more nodes transmit simultaneously. This can result in interference of the simultaneously transmitted packets on the receiver so that none of the packets can be decoded (received) correctly. Packet collisions waste energy and introduce extra latency because the collided packets need to be retransmitted. Collision avoidance is usually one of the requirements for MAC protocols that work with the random access principle. TDMA can however guarantee a collision free network since all the nodes are assigned different time slots for packet transmission and reception. TDMA, on the other hand, has its own practical limitations such as the scalability problem and the need for a controlling mechanism to govern schedule assignments. Even in a collision free network, retransmissions are inevitable due to interference and fading in a wireless channel. Forward error correction (FEC) based channel coding schemes can be used to lower the residual bit error rate. However, selecting an appropriate FEC scheme requires careful consideration since decoding has a high computation cost. Also channel coding requires transmitting extra bits and therefore adds additional overhead.

Packet Overhearing refers to a node receiving packets that are not destined for it. Since wireless medium is inherently based on broadcasting, nodes may overhear packets which are not destined to them. Receiving data packets and decoding them waste battery power for overhearing nodes. A node usually overhears when it cannot determine if the data packet is addressed to it. The destination address is therefore typically transmitted before the data bytes begin, so that a node is able to quickly decide whether or not to receive the upcoming data bytes.

Control Packet Overhead refers to the control information that is needed to be transmitted in addition to the actual payload. Besides the header, there are potentially many other overheads for duty cycled networks, e.g., information regarding the coordination of nodes for data exchange, synchronization messages for slot assignments, node wake-up schedules, etc. Although control packet overhead is inevitable, there are ways to minimize it. Designing compact protocol data structures, piggy-backing control information with data packets or merging more data packets together with the same control information are a few examples of minimizing the overhead.

2.2 TRENDS FOR LOW-POWER MAC DESIGN

In this section, we will provide background information on the trends of energy efficient MAC protocols and their analysis from a practical perspective. We will give a short chronological literature review of some of the representative MAC protocols used in practice that inspired the design of many of the existing low-power MAC protocols. We will briefly describe the strength and weaknesses of these protocols and discuss how they influenced other MAC schemes. This section does not aim at providing a comprehensive survey of the low-power MAC protocols. Instead the reader is referred to [2–4, 30] for a detailed classification and comparison of different protocols.

Broadly categorizing, there exist two major principles of MAC design for low-power embedded wireless networks, namely the TDMA and the contention based approach. The contention based protocols can further be sub-divided into two main classes: The first class is the IEEE 802.11 inspired MAC protocols, which use the common rendezvous principle with RTS/CTS/DATA/ACK handshake. This class of protocols enforces a common active/sleep schedule in order to achieve low duty cycle operation. The second subclass includes preamble sampling or channel polling protocols. Besides these types of MAC schemes, a few hybrid TDMA and CSMA, multi-channel, and receiver-initiated protocols have also been implemented.

2.2.1 TDMA based Protocols

TDMA based protocols divide a channel into time slots. Nodes are assigned to explicit slots for transmission and reception, while they are put to sleep during the rest of the slots. The number of time slots is usually fixed since managing variable number of time slots typically requires complex signaling overhead. TDMA based protocols typically suit to networks with a base station or a cluster head, which is able to coordinate the slot assignments. TDMA based MAC protocols can allow collision free systems since nodes only transmit in their respective slots. In the following, we describe the key design features of the well-known protocols, TRAMA [31], LMAC [32] and Bit-MAC [33], which have inspired the design of many other TDMA MAC protocols [2,3].

TRAMA [31] creates schedules which allow nodes to access a single channel in a collision-free manner. Time is divided into random- and scheduled access periods. The random access period is used by nodes to regularly broadcast neighbouring node identities and traffic information. This is used for building up two-hop neighbour information and computing a collision free schedule. Since the access to channel in the random access period is based on contention, it is prone to collisions. On the contrary, collision free scheduled access periods are used for sending data packets and schedule information.

LMAC [32] aims at reducing the design and implementation complexity for TDMA based MAC protocols. Each node controls only one time slot, which is reused at a non-interfering distance. Unlike in most of the traditional TDMA protocols, time slots are not assigned by a central controller – rather nodes select their time-slots based on the two-hop neighbourhood slot occupancy information. On one hand, LMAC reduces the need for having a central manager for slot assignment while on the other, every node needs to maintain neighbourhood information. During the initial network setup phase, nodes suffer from potential packet collisions since they are unaware of their neighbours and need to contend for controlling the time slots. When the setup phase is complete, nodes can communicate in a collision free fashion. Each message transmitted by a node consists of control information and data payload. All the nodes in a neighbourhood first listen to the control information part and based on the destination address decide whether to continue receiving the data portion or not.

BitMAC [33] is a deterministic and collision-free MAC protocol tailored to support high density node deployments. It is designed for radios supporting On-Off-Keying (OOK) modulation, multi-channel communication and synchronized bit transmission. It assumes a static spanning tree topology for data-collection networks. BitMAC eliminates the preamble of reply messages from a synchronized node. This technique can only be used on radios which provide a bit-level interface. Unlike LMAC where every node controls a dedicated time-slot, BitMAC allocates slots only to those nodes which need to send data. The sink node sends out beacon messages, collects transmit requests from all the nodes and assigns time-slots based on the requests. This optimizes channel utilization but adds a significant control overhead. In a case, where receiver is interested in data aggregation from different transmitters, the transmitting nodes access the channel concurrently in a bit-synchronized fashion without collisions.

TDMA based MAC protocols result in better collision avoidance performance than contention based-protocols. However, TRAMA and LMAC both need to keep two-hop neighbourhood information and compute the schedule and priorities, which add to the computational and communication workloads for a node. Since all the nodes need to be awake during the random access period for exchanging control information, it is difficult to achieve low duty cycles and therefore energy spent on the control overhead becomes significant. BitMAC is restricted only to one type of network topology and application. It also imposes restrictions on the hardware platform, which makes it less suitable for general usage. TDMA based MAC protocols usually require centralized or cluster based control, which is not guaranteed in all the low-power embedded networks. It is also not easy to change the slot assignments dynamically, which leads to scalability problems. Furthermore, there are some unused slots in a TDMA frame where none of the nodes transmit. As a consequence, time-slots remain under utilized. Some schemes steal the unutilized slots from other nodes to increase the throughput and channel utilization. However, scalability and efficient handling of network dynamics remain the biggest practical constraint for TDMA based protocols.

2.2.2 Contention based Protocols

In contention based protocols, nodes contend to access the medium for data communication. Since two or more nodes may simultaneously attempt to use the medium, it may lead to packet collisions unlike in TDMA based protocols. Contention based protocols are therefore designed to avoid packet collisions. They are scalable to a variable network size and are able to tolerate network dynamics in a better way as compared to TDMA based protocols.

S-MAC [29] is in part similar to IEEE 802.11 MAC protocol. It uses RTS/CTS/DATA /ACK handshake for unicast packets in order to reduce collisions. Furthermore, the overhearing nodes are put to sleep after receiving RTS/CTS packets. Scheduling active periods reduces energy consumption but the use of explicit SYNC frames for synchronization increases the control overhead. S-MAC trades off per-hop fairness and latency in favour of energy conservation.

nanoMAC [34] shares many design similarities to IEEE 802.11 MAC and allows duty cycle operation for energy conservation. Instead of using explicit SYNC frames as in S-MAC, nanoMAC uses RTS/CTS control frames to coordinate the rendezvous sleep schedules among nodes. While S-MAC allows a fixed duty cycle throughout the network, our nanoMAC implementation [34] allows on-the-fly selection of different predefined duty cycles on nodes depending upon the traffic load.

T-MAC [35] is also an IEEE 802.11 inspired MAC protocol. The key breakthrough of this protocol is that it introduced the idea of adaptive duty cycling. It automatically adapts the duty cycle to the observed traffic. Instead of using a fixed-length active duration, T-MAC uses a time-out mechanism to dynamically determine the length of the active duration. If a node does not detect any activity within a certain pre-defined time-out interval, it goes to sleep. This scheme reduces the amount of energy wasted in idle listening. However, exercising this time-out mechanism might lead to an early sleeping problem, where a potential receiver goes to sleep while another node still has a message intended for it.

S-MAC, nanoMAC and T-MAC, as representative examples of the contention based protocols with common rendezvous, possess certain degree of resemblance to TDMA based protocols as the nodes are forced to synchronize their periodic wake-up schedules. These protocols also inherit the problem from TDMA based protocols where the overhead of maintaining synchronization is high. However, contention based protocols with common rendezvous are efficient for broadcasting data packets since all the nodes follow the same schedule unlike the case in TDMA based protocols where nodes wake up at different time-slots. Although TDMA based protocols typically have a broadcast time-slot where all the nodes wake up for synchronization, this slot is mainly used for control information instead of data transmission. Contention based scheduling protocols are also easier to implement and deploy in ad hoc environments as they do not require nodes to form communication clusters, thus eliminating the need for inter-cluster communication and complex slot management. Another category of contention based MAC protocols is the preamble sampling MAC protocols, which do not require explicit scheduling on common rendezvous.

B-MAC [36] is a CSMA/CA based preamble sampling protocol, which allows nodes to follow asynchronous sleep and wake-up cycles. B-MAC requires nodes to wake up periodically for a short duration and listen to a transmission activity in the medium. If no activity is detected, nodes go to sleep and wake up on the next cycle. Before sending a data packet, the transmitter sends a long preamble (at least equal to the periodic channel polling interval) to ensure that the destination nodes detect the transmission. Upon detecting channel activity, a node extends its active period until it receives the data packet following the preamble sequence. B-MAC contains four main functionalities: Clear Channel Assessment (CCA) for determining the occupancy status of the channel, packet backoffs, link layer ACKs and low power listening. These functions are provided through the flexible Application Programming Interfaces (APIs) and can be enabled or disabled according to the application needs. B-MAC uses a specialized CCA mechanism that allows estimating the noise floor when the channel is free. The signal strength is sampled and added to a FIFO queue. Only the median sample from the queue is taken to estimate the noise floor using an exponentially weighted moving average with variable coefficient [37].

B-MAC works well in both static and mobile scenarios since all the nodes exercise the same fixed duty cycle in an asynchronous fashion without knowing the neighbourhood sleep schedule information. A long preamble ensures that all the nodes in the neighbourhood receive data packet that follows the preamble sequence. However, retransmissions in B-MAC are very costly because not just the data packet, a long preamble is also needed to be retransmitted. Retransmissions not only cause large energy wastage but also impart high latency. Preamble sampling MAC protocols do not use RTS/CTS control frames, which effectively reduces the control overhead. However, it leads MAC protocols to potentially suffer from hidden terminal problem and lower data reliability as we have shown on a testbed [34].

WiseMAC [38] is a preamble sampling MAC protocol, which aims at reducing the length of a transmitted preamble for unicast transmissions by making use of the sleep schedule of a destination. In case of a naïve preamble sampling MAC protocol such as B-MAC, the length of the transmitted preamble is directly related to the duty cycle of the receiver. For a network operating at a low duty cycle, the required preamble length is very long and is therefore highly energy consuming for a transmitter. WiseMAC maintains sleep schedule information of all the neighbouring nodes in a network while the nodes follow asynchronous channel sampling schedules. Upon receiving a data frame, the destination sends back an acknowledgement frame containing its sleep schedule information. This information is used by a transmitter for implicit synchronization with a receiver for next transmissions. In particular, the transmitter tries to send a data packet right at the time when the receiver is scheduled to wake up. Thus both transmitter and receiver are able to save energy by avoiding unwanted preamble transmission/reception. The preamble length cannot be reduced to zero because of the imprecise timings resulting from inaccuracies in crystal oscillators. WiseMAC estimates the length of a preamble based on the crystal jitters accumulated over two subsequent packet transmissions, and the offsets between the sleep schedules of the transmitter and the receiver. WiseMAC though optimizes the preamble length, adds an extra complexity to the network. The optimization applies only when the packets are unicast and when the traffic load is high in order to allow better estimation of the sleep schedule of a receiver. For broadcast transmissions, a preamble sequence equal to the channel check interval is sent as in B-MAC. If the neighbours of a node change very frequently, the optimization gains are low. Furthermore, in case of imprecise sleep schedule estimation, the preamble is overheard by neighbours.

2.2.3 Hybrid Protocols

Depending upon the application scenarios, both TDMA and contention based protocols have their strengths and weaknesses. Thus it is natural to combine the two approaches. We describe Z-MAC [39] as a representative example.

Z-MAC adapts itself to the level of contention in the network and combines features of both CSMA and TDMA principles. Under low contention, it behaves like CSMA and under high contention, it acts as a TDMA protocol. The switching between the two behaviours is dynamic while time-slot assignment is performed at the deployment time. Unlike classical TDMA, a Z-MAC node may transmit during any time slot and therefore carrier sensing is required before transmission. The owner of the slot is given higher priority over other contending nodes through a smaller initial contention window size. When a network is overloaded with traffic, i.e., there is high contention for slots, nodes always transmit in their own slots as is the normal behaviour in TDMA protocols. Under low contention, time slots can be used by all the nodes and this constitutes a CSMA like behaviour. The main drawback of hybrid protocols is their fairly complex signaling overhead.

2.2.4 Multi-channel Protocols

Multi-channel MAC protocols are well investigated for the classical wireless ad hoc networks since they offer higher wireless spectrum utilization and can provide better performance in terms of throughput and latency. In low-power applications, use of multiple channels can lead to power savings. Using a signaling channel independent of the data channel allows a receiving node to transmit a busy tone over the control channel and let other nodes become aware of an on-going transmission. This separate signaling channel also enables nodes to determine when and for how long they can power themselves off. Real implementations of multi-channel MAC protocols are however not very common because of a higher economic price of nodes with multiple radio interfaces and limited possibilities of dedicated access to multiple channels.

DCMA/AP [40] uses two separate channels simultaneously. The data channel is used for preamble and data packet transmissions while the control channel is used for indicating Receive-In-Progress (RIP) to avoid hidden and exposed terminal problems. When the control channel is free, i.e., neighbouring nodes are all idle, a node may start transmitting preamble and data. The length of the preamble is greater than the inactive period of the node so that it is detected by receiving nodes. When a receiver detects preamble, it sends a RIP frame on the control channel, which contains information of the duration for data reception. When the neighbouring nodes wake up and detect the message, they go to sleep for the entire duration as indicated in the RIP frame. The major advantage of using a dual channel transceiver is that while receiving data, a node can send RIP that prevents the neighbouring nodes from wasting energy on overhearing in data channel.

2.2.5 Receiver Initiated Protocols

Unlike the usual MAC designs, where a sender initiates data communication, in receiver-initiated MAC protocols a receiver triggers data communication by first transmitting a probe or a chirp signal [41]. Recently, Dutta *et al.* have carried out an implementation of A-MAC, which heavily leverages from the popular CC2420 [42] radio chip features such as hardware address recognition, transmission of automatic acknowledgement frames and the Link Quality Indicator (LQI). While A-MAC is certainly a way forward for combining low-power probing with the support for unicast and broadcast data transmission, receiver initiated MAC protocols generally suffer from synchronization problems, scalability issues and poor channel utilization. This makes receiver initiated MAC protocols unfit for a wide range of applications especially with low-traffic requirements and high density deployments.

2.3 COMPARISON OF ENERGY EFFICIENT MAC PROTOCOLS

In order to provide reliable data communication and enhance the lifetime of a low-power embedded network, several different power aware MAC designs have been explored by the research community [2–4]. Protocols based on the preamble sampling principle are popular as they do not enforce explicit synchronization among nodes. Nodes are implicitly synchronized through a preamble sequence and this takes place only when data communication is initiated. Since no extra maintenance overhead is required and nodes can follow asynchronous sleep schedules, preamble sampling protocols suit better to the typical characteristics of the resource constrained embedded wireless networks with low traffic and dynamic nature [3]. A transmitting node sends a preamble sequence for a length equal to the periodic channel polling interval. Upon detecting the preamble sequence, the receiving nodes keep on listening to the medium and eventually receive the data packet, which follows the preamble. This way a preamble sequence implicitly synchronizes the asynchronously waking up nodes to receive data, which is always transmitted immediately after the preamble sequence. Since typically wireless nodes operate in low duty cycles, the corresponding preamble sequence has to be long. This leads to significant amount of energy consumption, not only at the transmitter but also at the addressed and the non-addressed nodes in receiving and overhearing the preamble sequence.

Many techniques have been proposed to shorten the length of the transmitted and received preamble sequence for conserving battery consumption. MFP-MAC [43] divides a long monolithic preamble into tiny frames – each containing the destination address and the data payload information. An asynchronously waking up node upon receiving a preamble frame is able to decide if the upcoming data is intended for it or not, and therefore can avoid listening to the rest of the redundant preamble sequence. If a particular node happens to be a destination, it wakes up again in order to receive the data frame following the preamble frames. The timing information for the transmission of data frame is also sent inside the preamble frame. MFP-MAC allows saving energy consumption for receivers, while the transmitting node is still required to send a preamble sequence for a duration of the periodic check interval. X-MAC [44] divides the monolithic preamble into small frames, each containing the destination address. For unicast transmission, preamble strobing technique is used where after transmitting a frame, the transmitter waits for its acknowledgement from the destination. Subsequent preamble frames are sent if the preamble frame remains unacknowledged within a certain timeout duration. After receiving an acknowledgement, the transmitter immediately sends out data. Preamble strobing operation saves energy for transmitters in unicast transmission by avoiding transmission of extra preamble frames once a preamble frame has been acknowledged. The required length of the preamble sequence depends upon the relative offset between the start of a preamble sequence and the wake up schedule of the receiver.

B-MAC+ [45] transmits the actual data frame back to back to form the preamble sequence. The receiving nodes need to receive only one data frame without needing to receive the useless preamble sequence and can go to sleep for the rest of the preamble transmission duration. B-MAC+ exercises preamble strobing for unicast transmission and thus the transmitter as well as the receiver save energy by avoiding the need for transmitting and receiving the useless preamble sequence. Having no information about the sleep schedules of the destination(s), a significant amount of energy is spent for preamble transmission and reception despite the use of the above preamble shortening techniques.

WiseMAC [46] makes use of neighbourhood sleep schedule in order to shorten the preamble length for unicast transmission. Each node explicitly announces its wake-up schedule in the acknowledgement packet. Having the knowledge of wake-up information, a transmitter delays the data transmission until the destination is scheduled to wake up. WiseMAC also adjusts the length of the preamble based on the jitter offset developed over time between the transmitter and the receiver clocks. In a follow-up article [38], the authors of WiseMAC devised a scheme of repeating the data packet for broadcast transmissions where the preamble lengths are long. Relying heavily on the neighbourhood sleep schedule, WiseMAC has shortcomings in a dynamic network where sleep schedules can be easily outdated or incomplete. Furthermore, repeating data packets in preamble introduces energy saving only when data size is small [19].

SCP-MAC [47] uses synchronized channel polling and combines the features of scheduled based protocols with preamble sampling. This approach is suitable for networks operating in very low duty cycles with static topologies.

Unlike the preamble sampling approach, IEEE 802.11 inspired MAC protocols such as T-MAC [35] aim at adapting the effective duty cycle of the network by early timeout scheme in order to lower the energy wastage in idle listening. nanoMAC [34] provides a means to select one of the three supported fixed duty cycles based on the application traffic requirements in order to save the idle mode power consumption. Z-MAC [39] uses a hybrid CSMA/TDMA based medium access principle. In low traffic conditions, it behaves as a CSMA protocol while in the case of higher traffic volumes, it behaves like a TDMA protocol. Z-MAC implementation on a testbed has shown to out-perform B-MAC in high traffic load scenarios. However, Z-MAC has a fairly complex signaling mechanism and in low traffic conditions, its control overhead starts to dominate. Funneling-MAC [48] also uses the hybrid CSMA/TDMA principle to effectively handle the traffic load near sink node(s). It has a complex signaling mechanism and suffers from network dynamics.

In addition to the contention based protocols, conflict free TDMA based protocols such as BMA [49] and LMAC [32], are also used in practice. Time-slotting inherently has a duty cycling behaviour and conserves energy as the nodes are active only in the assigned slots. Slot assignment and maintenance has a high control overhead, which makes contention free protocols a less attractive choice. Furthermore, contention free protocols suffer from scalability and mobility problems since slots need to be updated when the network topology alters. BurstMAC [50] uses the TDMA principle to efficiently handle the correlated traffic bursts in low-power network applications. A sender indicates the transmission of an upcoming traffic burst. Using an on-demand channel allocation scheme based on a two hop coloring algorithm, concurrent transmissions can take place resulting in higher throughput. BurstMAC requires strict time-synchronization among nodes and is not efficient for broadcast transmission and in dynamic network conditions.

MaxMAC [51] offers a smart scheme to adaptively increase the wake-up rates by reacting to the offered traffic load. MaxMAC keeps track of the incoming traffic at a node and if the traffic rate increases over a pre-defined threshold, extra wake-ups are introduced. The simulation results show higher throughput and lower latency as compared to T-MAC and WiseMAC. The dynamic duty cycling of MaxMAC has an edge over static duty cycling schemes such as in A-MAC [52, 53] where a node doubles the active periods when higher traffic volumes are experienced. Bac *et al.* [54] propose a tree based scheme with adjustments of duty cycles based on the traffic loads in the network. The nodes are synchronized and use a super-frame structure at each level with distinct topology. This scheme has its limitations in the presence of high network dynamics and is therefore not suitable to many low-power embedded network deployments. StrawMAC [55] is a contention based protocol and exercises RTS/CTS backoff strategy. It is designed to operate in low duty cycles, however in case of sporadic traffic surges, it tries to adapt its duty cycle by estimating the length of the data packet through control frames.

All the above mentioned protocols are designed to use single channel in a particular frequency band. Schurgers *et al.* have proposed STEM [56], which uses two radios operating in separate frequency bands to completely separate data transfer from wake-up signaling. In the tone based approach of STEM, a long wake-up tone is sent to make sure that the destination has awoken once. The use of a wake-up tone instead of an addressing mechanism results in energy wastage for the non-addressed nodes. DCMA/AP [40] is a dual channel MAC protocol and uses two separate channels in the same frequency band. The main idea is to conserve energy consumption by avoiding RTS/CTS control frames. The data channel is used for preamble and data packet transmissions while the control channel is used for indicating reception in progress in order to avoid hidden terminals and packet overhearing.

2.4 ENERGY EFFICIENCY THROUGH MULTI-MODE MAC DESIGN

The biggest challenge for low-power MAC protocols is that there exists no universal solution, which is optimal for all the diversified applications [1]. A particular MAC protocol works well for a specific network and traffic conditions but remains inefficient for others. Furthermore, even for a particular application the requirements may change over time and switching to the closest to the optimal MAC solution from a pool of different MAC protocols at runtime [57] but this approach is unpractical for resource constrained nodes. In this context, we have designed a Traffic aware MAC (TrawMAC) protocol, which is a multi-mode MAC scheme and adapts its behaviour to the changing traffic and network conditions. TrawMAC aims primarily to optimize the energy consumption by exploiting the shared traffic information across routing and MAC protocol layers with minimum compromises in latency and packet delivery ratio. It is based on the preamble sampling principle and combines the various types of preamble optimization techniques as were discussed in Section 2.3. The protocol includes additional features to efficiently handle the varying traffic loads and types.

Section 2.4.1 describes the design details of TrawMAC protocol and various energy conservation techniques. Section 2.4.2 presents an analytical model of TrawMAC and gives an expression for the optimum energy consumption at different sampling periods and network traffic loads. Section 2.4.4 presents the evaluation of TrawMAC using both simulation and testbed implementation studies.

2.4.1 Protocol Design Details

A preamble transmission coordinates different nodes for data exchange while allowing them to poll the medium asynchronously. Different types of optimization techniques are applied to shorten the required preamble length to be transmitted and received to conserve the available energy. Since none of the preamble shortening technique is universally optimal in all the traffic conditions [1, 58], TrawMAC makes intelligent use of different preamble optimization methods depending upon the network traffic conditions. In the following, we briefly discuss the different aspects of the protocol design.

Preamble Framelets

TrawMAC divides a monolithic preamble sequence into small preamble frames, each containing the destination address, the source address, the message type and some optional fields depending on the message type. The composition of preamble sequence varies depending upon the number and size of data packets to be transmitted and the destination address. There are mainly two types of preamble frames: data-frame and micro-frame. When the size of the data packet is smaller than a certain empirically found threshold, data-frame is used instead of micro-frame. Data-frame, as the name suggests, contains the data payload inside the preamble frames. A train of data-frames forms Data Frame Preamble (DFP), which serves the purpose of transmitting both preamble and data. Micro Frame Preamble (MFP), on the other hand, consists of micro-frames which contain only the control information. By using a preamble which consists of a number of small preamble frames instead of one single long frame, energy consumption can be reduced at the receiver as well as at the transmitter. In case of broadcasting data-frame preambles, a receiver goes to sleep immediately after receiving a single data-frame. In case of broadcasting micro-frame preambles, all the receivers go to sleep asynchronously after receiving a micro-frame, and switch back to receive mode synchronously at the start of the scheduled data packet transmission as illustrated in Figure 2.1. In unicast transmission, additional optimization techniques on the preamble length have been applied as discussed later in this section.

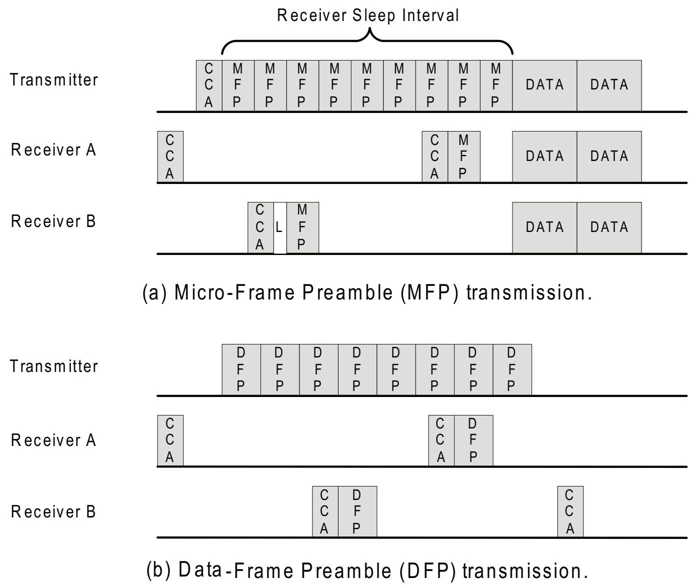


Figure 2.1: TrawMAC operation for broadcast transmission. In (a), data size is larger than an empirical threshold and therefore MFP is used. Receivers in the neighbourhood wake up asynchronously, perform CCA and receive MFP frames, which synchronizes nodes to receive data frames. In (b), receivers wake up asynchronously, perform CCA and receive DFP frames, which contain data.

Neighbourhood Information

TrawMAC maintains sleep schedules of neighbours in a similar way as WiseMAC [38]. However, unlike WiseMAC, TrawMAC announces the sleep schedule of the transmitting nodes in the preamble frames, which has the advantage that non-addressed nodes are also able to update the sleep schedule information. The sleep schedule is announced in unicast as well as for broadcast transmission. Preamble length shortening based on the gathered sleep schedules of the neighbouring nodes is applied for unicast transmissions. A transmitter looks up the gathered sleep schedule information of its neighbours and delays the transmission of the preamble till the destination is scheduled to wake up. The preamble length is shortened by the delayed duration. If the sleep schedule of a receiver is not known, a transmitter starts transmission immediately and sets the maximum length of preamble to be the same as for broadcast transmission. It is important to note that a transmitter does not skip channel sensing at its scheduled wake-up instant even if it is in the process of delaying a preamble transmission since other nodes in the neighbourhood may potentially transmit a data packet at its scheduled wake-up.

Preamble Strobing

Preamble strobing technique [44] is used in unicast transmissions. After transmitting a preamble frame (DFP/MFP), the transmitter waits for its acknowledgement from the receiver. When an acknowledgement is received, the transmitter stops sending subsequent micro-frames. In case of an MFP, data packet(s) are sent immediately after an acknowledged micro-frame while in case of a DFP, an acknowledgment frame marks the completion of data transmission. Combining neighbourhood wake-up schedule information with preamble strobing in TrawMAC requires only one preamble frame to be transmitted in the best case. However, due to clock drifts over time, the estimated wake-up schedule of a neighbour can be inaccurate [38], and hence more than one preamble frames might be needed to be transmitted as illustrated in Figure 2.2. Having no acknowledgement received, the transmitter keeps on sending subsequent preamble frames after a time-out interval. In the worst case, the required length of a preamble transmission becomes the same as that for a broadcast transmission, i.e., equal to the periodic channel check interval. Since each data frame is acknowledged as well, a retransmission of only the unacknowledged data frames is ensued upon the failure to receive acknowledgements after a certain timeout interval. Non-addressed nodes might also wake up during data packet transmission and in order to avoid further overhearing, receiving nodes are only allowed to listen to the medium for a maximum duration of twice the maximum sized micro-frame transmission when a channel activity is detected.

Preamble Substitution

With the preamble optimization methods described above, nodes can save significant amount of energy compared to the naïve preamble sampling approach of B-MAC [36]. However, in case of broadcast transmissions, a transmitter is required to send a long preamble of length equal to the channel check interval, which results in high energy

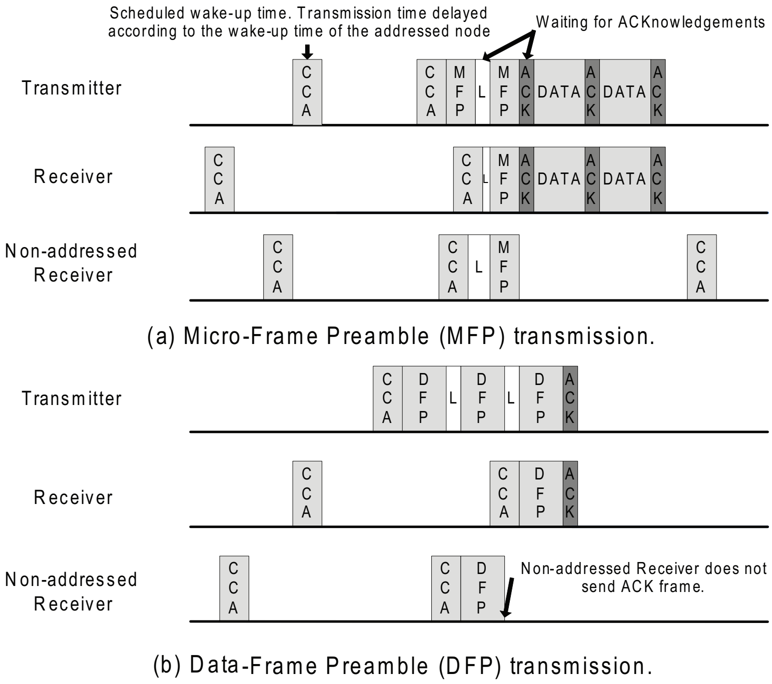


Figure 2.2: TrawMAC operation for unicast transmission. (a) When an MFP is received by an addressed node, it sends an acknowledgement frame and receives data frames transmitted immediately afterwards. (b) A DFP transmission stops immediately when an acknowledgement is received from the destination.

wastage. As a consequence of frequent broadcast transmissions not only can the transmitter be depleted with energy, congestion is also caused in the network because of longer channel occupancies. When network information is unavailable to nodes, preamble length optimization cannot be applied in broadcast packet transmissions since the sleep schedule of the potential receivers is unknown. However, since TrawMAC collects neighbourhood information, it is possible to shorten the transmitted preamble length even in broadcast transmission. In a static network neighbourhood, nodes can have complete information about their neighbours after certain number of message exchanges. Based on the gathered wake-up schedules of all the neighbours, a transmitter can shorten the length of a preamble for broadcast packets using a novel technique described below.

A simple broadcast can either be substituted with multiple unicasts, a shortened broadcast, or a combination of these, depending on the sleep schedules of the neighbouring nodes. TrawMAC estimates the energy consumption of transmitting a broadcast message and also transmitting the possible substitutions. The energy spent in a packet transmission is directly proportional to the size of the packet at a given transmit power level since the radio bit rate remains constant. The relationship of energy consumption versus data length can therefore easily be expressed as a linear equation. The linear relationship has also been shown by Panthachai [59] for the case of the popular CC2420 [42] radio at different transmit power levels. When a broadcast preamble is substituted, the resulting sequence includes transmitting a certain number of bytes at different intermittent intervals while switching off/on the radio. Radio switching and CCA operation take different time durations and energy consumption on different types of radio. This may lead to different decisions for broadcast preamble substitution on different radio transceivers. These timing and power consumption figures are hardware dependent and are defined as parameters in TrawMAC design. Since the overall energy expenditure for a broadcast substitution is linearly related to individual operations, TrawMAC is able to make an estimation for all the realizable possibilities beforehand and subsequently exercise the most energy optimal substitution. The parameters required for describing the energy spent in radio communication are listed in Table 2.1:

Table 2.1: List of parameters describing energy consumption in radio communication.

Parameter	*Description*
P_{tx}	Power consumption of a radio in transmission (TX) state [W] It depends on the transmission power level. For TelosB [14] which is used in our evaluation, the exact values can be found in [59]
P_{rx}	Power consumption of radio in reception (RX) state [W]
$t_{\text{poll\_once}}$	Time required for performing one channel polling operation[s]
$E_{\text{poll\_once}}$	Energy spent in performing one channel polling operation[J]
t_{rxsetup}	Time required for the radio to switch on to RX state [s]
E_{rxsetup}	Energy spent in switching on radio to RX state [J]
t_{txrx}	Time required for the radio to switch from TX to RX state [s]
E_{txrx}	Energy spent in switching radio from TX to RX state [J]
t_{rxtx}	Time required for the radio to switch from RX state to TX state [s]
$l_{\text{preamble\_frame}}$	Length of the frame to be transmitted in the preamble [bit]
r_{b}	Data transmission rate [bps]
E_{frame}	Energy spent in transmitting one frame [J]

Before substituting a broadcast preamble, TrawMAC estimates the energy consumption for the substitution using the following rules:

1. If the offset between sleep schedules of any two neighbours is smaller than *GAP* $= l_{\text{preamble\_frame}}/r_{\text{b}} + t_{\text{rxsetup}} + t_{\text{poll\_once}} + t_{\text{rxtx}}$, the only possibility would be to do a shorter preamble length broadcast for the two nodes.

2. If the offset between sleep schedules of any two neighbours is greater than *GAP*, choose smaller of the estimated energy consumption for a shorter broadcast (E_{bcast}) and two unicasts (E_{ucast}). E_{bcast} and E_{ucast} can be expressed as:

$$E_{\text{bcast}} = \left\lceil \frac{\text{offset}}{\frac{l_{\text{preamble\_frame}}}{r_{\text{b}}}} \right\rceil \times E_{\text{frame}} \tag{2.1}$$

and

$$E_{\text{ucast}} = E_{\text{rxsetup}} + E_{\text{poll\_once}} + E_{\text{txrx}} + 2 \times E_{\text{frame}} \quad (2.2)$$

We have developed an algorithm for substitution of long broadcast preambles based on the sleep schedule of the neighbours. The pseudo-code of the algorithm is given in Algorithm 2.4.1. First we define a data structure called *Neighbour* with three fields: *offset_startpa* which is the offset to the beginning of the preamble, *offset_next*, which is the offset to the next node, and *tx_method* which indicates whether it is to be sent as a broadcast or a unicast. The field *offset_next* contains the wake up offset between two nodes. If *offset_next* is bigger than the minimum *GAP*, a separate transmission is more energy efficient than continuing broadcasting back-to-back preamble frames. *GAP* is the time duration for a node to switch to receive mode, perform one CCA operation, switch to transmit mode and transmit one preamble frame. Generally, the time required to switch off a radio chip is negligible and is therefore not modeled in our algorithm.

Algorithm 2.4.1: BROADCASTTOUNICAST()

```
Struct_Neighbour
{offset_startpa; offset_next; tx_method;
comment: 1) Sort neighbourlist based on the offset in ascending order.
comment: 2) Calculate offset_next.
for i ← 0 to totalneighbour − 1
  do neighbour[i].offset_next ← neighbour[i + 1].offset_startpa−
neighbour[i].offset_startpa
comment: 3) Decide on transmission method for each node.
local broadcast ← false
for i ← 0 to totalneighbour
  do if neighbour[i].offset_next < GAP or
E_bcast(neighbour[i].offset_next) < E_ucast and neighbour[i].offset_next! = empty
       ⎧ if broadcast = false
  then ⎨   then ⎧ neighbour[i].tx_method ← broadcast_start
       ⎪        ⎩ broadcast ← true
       ⎩   else neighbour[i].tx_method ← broadcast
  else if neighbour[i].offset_next! = empty
       ⎧ if broadcast = true
       ⎪   then neighbour[i].tx_method ← broadcast_end
  then ⎨
       ⎪   else neighbour[i].tx_method ← unicast
       ⎩ broadcast ← false
       ⎧ comment: last node in the list
  else ⎨ if broadcast = true
       ⎪   then neighbour[i].tx_method ← broadcast_end;
       ⎩   else neighbour[i].tx_method ← unicast;
```

Figure 2.3 shows the preamble sequence with and without substitution operation for the case of DFP as well as for the case of MFP. In the preamble substitution, though some of the transmissions are for specific receivers, the destination address field remains broadcast and no acknowledgements are expected at the transmitter. Therefore, these transmissions can more accurately be termed as pseudo-unicast. The transmitter sends individual preamble frames for each of its neighbouring nodes according to the wake-up schedules of the neighbours. If nodes wake up in closer instances with each other, the scheme can be simplified through a broadcast transmission with a shorter

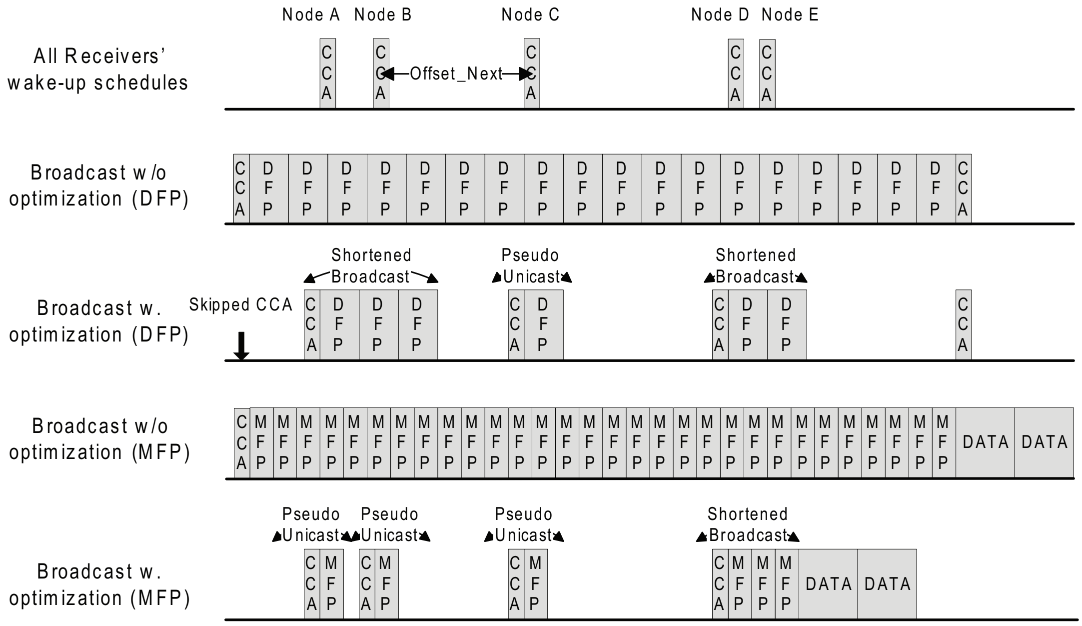


Figure 2.3: Optimization of broadcast preamble in TrawMAC for both DFP and MFP transmissions when the sleep schedule of the receivers is known at the transmitter.

preamble. For example in Figure 2.3, the wake up offset between nodes A and B, and D and E, are shorter than the *GAP* calculated for DFP transmission. Therefore, the transmitter sends two short broadcast preambles with back-to-back data frames for nodes A and B, and D and E. Node C is transmitted a pseudo-unicast packet. In the case of an MFP transmission, instead of transmitting data packets immediately after the preamble frame transmission for each destination, data packets are only transmitted after the preamble frames for all the nodes are sent. The transmission time of data packets is pre-determined depending upon the neighbour with the largest offset before the first preamble frame can be transmitted so that the timing information can be included in all of the preamble frames. In this case, the operation can be appropriately described as a selective transmission of micro-frame preamble. Similar preamble shortening principles are applied to multicast transmissions as that to broadcast transmissions, e.g., a multicast transmission can be divided into multiple unicasts if the schedule information of the destined nodes is known.

In order to validate the effectiveness of broadcast optimization described above, we have carried out a simple set of simulations in OMNET++ v4.0 [60] (cf. Section 2.4.4) for power consumption analysis. A one-hop network is simulated starting with each node sending a packet to the sink so that the sink gathers complete neighbourhood timing information. After the neighbourhood timing information is gathered at the sink node, it broadcasts 100 packets at a rate of 0.1 packets/second. Power consumption measurements are started when the sink starts broadcasting packets. In order to observe the advantage of broadcast preamble substitution, we repeated the experiments

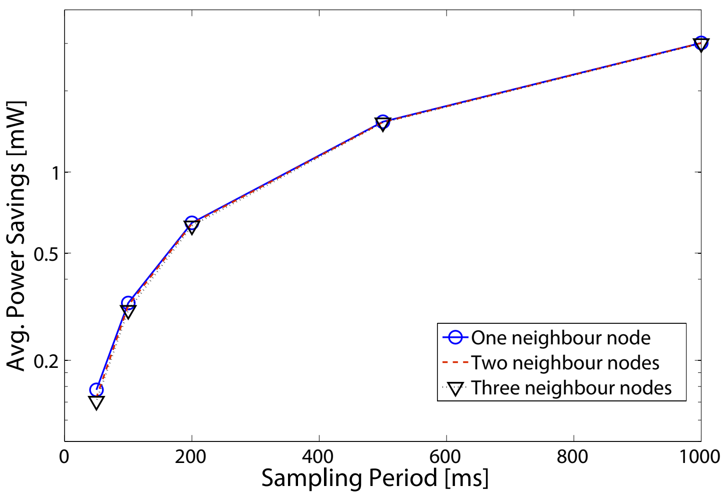


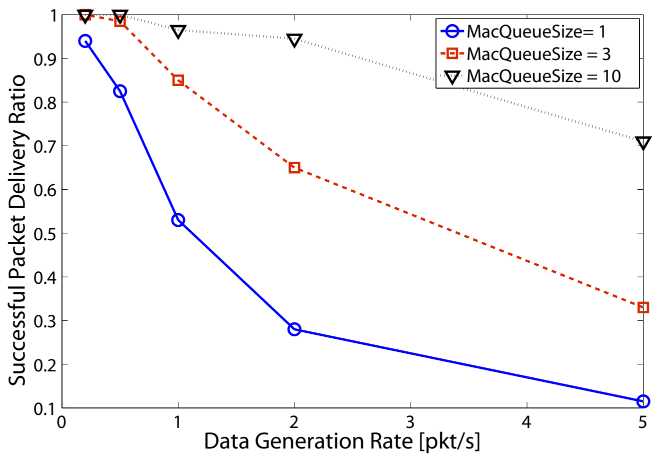
Figure 2.4: Average power consumption savings at different sampling periods through broadcast preamble optimization in TrawMAC when 100 packets are transmitted in a network at a rate of 0.1 packets/second.

without the broadcast preamble substitution. Figure 2.4 shows the average power consumption savings when the broadcast preamble substitution is applied compared to the case without the broadcast substitution. Although the power consumption is shown for the whole network, savings occur only at the transmitter. The transmitter is referred to as the sink node in our simulations. As the sampling period increases, the energy savings also become high. This is because the required length of the transmitted preamble becomes larger for longer sampling periods in the case without preamble substitution and hence results in high energy wastage. An increasing number of nodes in the neighbourhood also show a slight decrease in the total amount of savings since the sink node is required to transmit for a relatively longer period of time in order to reach more nodes.

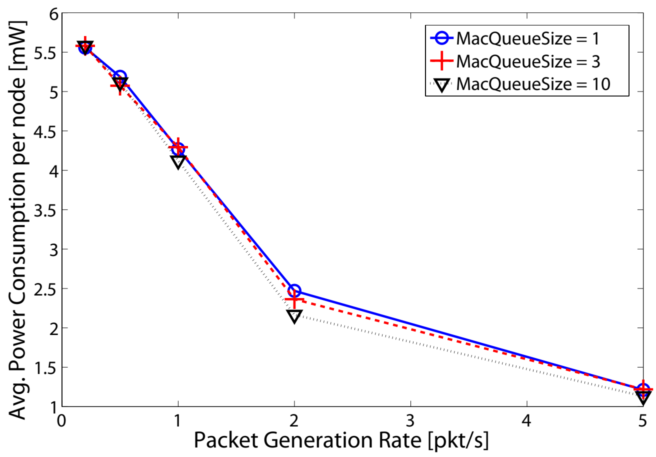
Data Aggregation

A natural latency is introduced in data communication for duty cycling MAC protocols since data packets can be received only when the active periods of the transmitter and the receiver(s) are aligned. Since packets to be transmitted can be scheduled at any point of time, they cannot always be processed immediately especially when the channel is unavailable or when the transmitter is forced to delay the transmission having a prior knowledge on the sleep schedule of the receiver. Therefore, often more than one data packets is scheduled for transmission at the transmitter before it can find the channel free and when the receiver is in active state to be able to receive data. In this situation, most of the MAC implementations simply refuse to accept more packets after a packet has been scheduled for transmission at the MAC layer. When the packet generation rate becomes higher than the channel polling rate or the rate of seizing a free channel at a node, the network becomes overloaded and as a result MAC protocols start to drop packets, report high latency, show low delivery ratio and obviously waste a significant amount of energy. In this context, we have proposed MAC queue as part of the TrawMAC design. We see a high potential for minimizing control overhead (essentially the length of the preamble sequence) through data aggregation using MAC queue. In majority of the low-power embedded applications, data aggregation is traditionally carried out at the routing layer, for instance, in the case of data-centric routing protocols to reduce energy consumption at the expense of latency [61]. Keeping a packet queue at the routing layer allows aggregating multiple packets arriving at different time instants. Additionally, the MAC queue enables aggregating packets which are scheduled for transmission but have not yet been transmitted because of the inherent duty cycling delays. TrawMAC not only aggregates data packets which are scheduled for unicast transmission to the same destination or broadcast packets but may also aggregate packets destined to different nodes using multicast addresses. As a result, significant energy savings can be made by avoiding the transmission of multiple long preamble sequences (cf. Figure 2.5). The size of the queue can be flexibly set depending upon the perceived traffic load fluctuations.

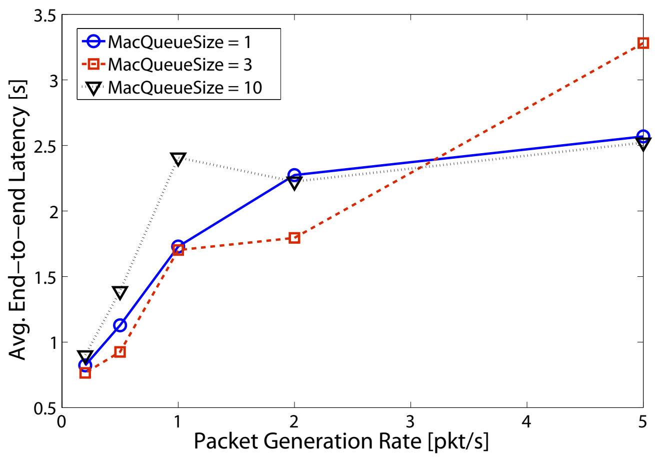
Before transmitting data packets, TrawMAC first looks through the data packets queued for transmission and sends the ones with the same destination address using a single preamble reservation in the form of a frame train. Transmitting multiple data packets with a single preamble certainly saves high amount of energy, which otherwise requires transmission/reception of multiple preamble sequences. The transmission of data packets is based on a first-come-first-served basis, i.e., if the destination address of the packets in the queue is `a`, `a`, `b`, `c`, `a`, only the first two with address `a` will be transmitted together. This not only conserves energy but also lowers the latency and improves the channel bandwidth utilization. Furthermore, it allows the routing infrastructure to interact with the queued packets. For instance, in one of the integration experiences of TrawMAC protocol with a content based routing protocol in publish/subscribe paradigm [62], the routing protocol is able to indicate the cancelation of the scheduled queued packets, when necessary (e.g., after timing out or receiving a signal). In order to show the effect of a MAC queue and data aggregation mechanism in TrawMAC, we have carried out a set of simple simulation runs with a fixed sampling period value, variable queue size, and different packet generation rates. A one-hop three node network is considered with two transmitters sending packets. All the nodes operate with a sampling period of 1000 ms. Since the metrics such as power consumption, offered latency and the packet delivery ratio, are interrelated, we studied them together. The simulation results are shown in Figure 2.5. Packet delivery ratio increases significantly with larger queue sizes. The power consumption is not heavily affected with the size of the queue although more data packets are delivered when the MAC queue is larger. It is because multiple data packets are sent with one preamble for channel reservation and thus packet drops out the queue can be reduced. Exercising data aggregation based on MAC queue brings improvements in network performance but affects the fairness among nodes in a network. According to many researchers, fairness is regarded as a secondary MAC performance metric [29].



(a) Average Delivery Ratio.



(b) Average Power Consumption.



(c) Average End-to-end Latency.

Figure 2.5: Performance characteristics of TrawMAC with different MAC queue sizes at different packet generation rates. A network of three nodes is considered and the sampling period of all the nodes is set to 1000 ms.

2.4.2 Analysis of Optimum Energy Consumption

We have derived an expression for the optimum sampling period of TrawMAC, which gives the minimum energy consumption required for handling a given traffic load in a network. Our model is simplified with several assumptions but helps in understanding the relationship between network traffic load and duty cycle, and their effect on the energy consumption at a node. The channel is assumed to be ideal, i.e., all the transmitted packets are successfully received at the receivers. Packet collisions, retransmissions and data aggregation are however not modeled. The channel is considered to be uncongested without having the need for backoffs at the MAC protocol. Since commercially available low-power radios have very fast mode switching durations, the energy consumed in radio mode switching, such as, transmit-to-receive and sleep-to-receive, is ignored.

Models and Parameters

Consider a network consisting of n nodes within the vicinity of each other. Each node transmits r_{data} data packets per second. A data packet takes a duration t_{data} to be transmitted. If DFP is used, data packets are included in preamble and $t_{\text{data}} = 0$. Each node consumes power in operations: Carrier sensing before transmission, data transmission, data reception, channel polling at periodic wake-up and sleep mode, which are denoted by P_{cs}, P_{tx}, P_{rx}, P_{poll} and P_{sleep}, respectively. Table 2.2 lists the terms used in our model.

Table 2.2: Notation for different terms used in the analytical model of TrawMAC.

Notation	*Meanings*
$t_{\text{poll\_once}}$	single channel polling duration [s]
$t_{\text{samp\_period}}$	channel sampling period [s]
$l_{\text{preamble\_frame}}$	length of one preamble frame [bit]
$t_{\text{cs\_once}}$	single channel carrier sensing duration [s]
t_{b}	bit duration corresponding to radio data rate [s]
t_{ack}	duration of one acknowledgement packet transmission [s]

Broadcast

The overall energy consumption at a node is the sum of energy spent in each operation and is given by,

$$E = E_{\text{poll}} + E_{\text{rx}} + E_{\text{cs}} + E_{\text{tx}} + E_{\text{sleep}}. \tag{2.3}$$

Since energy can be expressed as the product of power consumption and time duration, the equation can be expressed as :

$$E = P_{\text{poll}}t_{\text{poll}} + P_{\text{rx}}t_{\text{rx}} + P_{\text{cs}}t_{\text{cs}} + P_{\text{tx}}t_{\text{tx}} + P_{\text{sleep}}t_{\text{sleep}}. \tag{2.4}$$

The values of the power terms are based on the hardware specifications while the timing expressions are given as follows:

$$
\begin{aligned}
t_{\text{poll}} &= \frac{t_{\text{poll\_once}}}{t_{\text{samp\_period}}}, \\
t_{\text{rx}} &= (n-1)r_{\text{data}}(1.5 l_{\text{preamble\_frame}} t_{\text{b}}) + (n-1) r_{\text{data}} t_{\text{data}}, \\
t_{\text{cs}} &= r_{\text{data}} t_{\text{cs\_once}}, \\
t_{\text{tx}} &= r_{\text{data}} t_{\text{samp\_period}} + r_{\text{data}} t_{\text{data}}, \\
t_{\text{sleep}} &= 1 - t_{\text{poll}} - t_{\text{rx}} - t_{\text{cs}} - t_{\text{tx}}.
\end{aligned}
$$

In order to know the destination address and timing information for the data frame, receivers are required to listen to the medium for an average of 1.5 micro-frame transmission duration. Ideally, nodes are required to listen for a complete micro-frame duration but since a frame boundary can be misaligned due to the imprecise estimated wake up schedule, nodes may have to receive one to two micro-frames. This gives an average of 1.5 micro-frames reception duration. The length of the preamble sequence to be transmitted before the data frame(s) equals to the periodic sampling interval. Since our target is to find the sampling period value which leads to the minimum energy consumption, we plug-in the terms defined above into Equation (2.4) and take the derivative w.r.t. $t_{\text{samp\_period}}$:

$$
\frac{dE}{dt_{\text{samp\_period}}} = -\frac{P_{\text{poll\_once}} t_{\text{poll}}}{t_{\text{samp\_period}}^{2}} + r_{\text{data}} P_{\text{tx}} + \frac{P_{\text{sleep}} t_{\text{poll\_once}}}{t_{\text{samp\_period}}^{2}} - r_{\text{data}} P_{\text{sleep}}. \tag{2.5}
$$

Putting $\frac{dE}{dt_{\text{samp\_period}}} = 0$ and simplifying the terms gives the optimum sampling period,

$$
t_{\text{samp\_period}} = \sqrt{\frac{t_{\text{poll\_once}}(P_{\text{poll}} - P_{\text{sleep}})}{r_{\text{data}}(P_{\text{tx}} - P_{\text{sleep}})}}. \tag{2.6}
$$

It should be noted that the $t_{\text{samp\_period}}$ expression is independent of n, the number of nodes contending for the same channel. It is due to the fact that in TrawMAC, energy dissipated in receive mode (which is dependent on n) is independent of the sampling period as we can see from the expression for t_{rx}. However, since our model is only applicable for uncongested network, n has an upper bound which is imposed by the equation,

$$
n \leq \frac{1}{r_{\text{data}}(t_{\text{samp\_period}} + t_{\text{data}})}.
$$

If n is greater than this bound, the network is in congestion and the optimum duty cycle derived using this model does not offer the optimum performance.

Unicast

The total energy consumption at a node takes the same form as Equation (2.4) in the case of unicast transmission. Instead of being the destination for all the packets transmitted in the neighbourhood, a particular node is destination for k packets out of the total $(n-1)$ packets transmitted by its neighbours. Having a perfect knowledge of the neighbourhood timing schedules for unicast transmission in TrawMAC, only one preamble frame is required to be transmitted regardless of the sampling period. TrawMAC has an inherent ability to optimize the preamble length. In the absence of any timing information, the preamble length depends on the offset between the sleep schedules of the transmitting and receiving nodes. In the worst case, the minimum preamble length is the same as in broadcast transmission and is given by Equation (2.6). However, the timing expressions for unicast transmission are different than those for broadcast transmission and are given by:

$$\begin{aligned} t_{\text{poll}} &= \frac{t_{\text{poll\_once}}}{t_{\text{samp\_period}}}, \\ t_{\text{rx}} &= (n-1)r_{\text{data}}(1.5 l_{\text{preamble\_frame}} t_{\text{b}}) + k r_{\text{data}} t_{\text{data}} + 2 r_{\text{data}} t_{\text{ack}}, \\ t_{\text{cs}} &= r_{\text{data}} t_{\text{cs\_once}}, \\ t_{\text{tx}} &= r_{\text{data}} t_{\text{samp\_period}} + r_{\text{data}} t_{\text{data}} + 2k r_{\text{data}} t_{\text{ack}}, \\ t_{\text{sleep}} &= 1 - t_{\text{poll}} - t_{\text{rx}} - t_{\text{cs}} - t_{\text{tx}}. \end{aligned}$$

A coefficient 2 is used for the time duration for acknowledgement because a node needs to transmit/receive acknowledgement for both the preamble frame and the data packet in the case of MFP transmission. If DFP is used, the coefficient equals to one. Similarly, the upper bound of n for MFP transmission is governed by the equation,

$$n \leq \frac{1}{r_{\text{data}}(t_{\text{samp\_period}} + t_{\text{data}} + 2t_{\text{ack}})}.$$

2.4.3 Comparison of Analytical Model with Prototype Implementation

Equation (2.6) shows that the data transmission rate r_{data} is variable while other terms are fixed either due to radio properties or protocol design. We have plotted the power consumption per node at different sampling periods and observed a concave-up shape behaviour showing the existence of a minimum value, which is found to be consistent with the optimal value obtained through analytical expression.

We used all the parameters for our simulations in OMNET++ v4.0 based on the CC2420 [42] radio transceiver characteristics as listed in Table 2.3. We considered a small network consisting of only three nodes in order to have a small uncongested network. The length of the preamble frame, $l_{\text{preamble\_frame}}$, was fixed to 176 bits. Each simulation with different traffic generation rates and radio duty cycle values lasted for 20 minutes and each experiment was repeated three times. Figure 2.6 shows the average power consumption of a node at different sampling periods for different data packet rates for broadcast transmissions. Using Equation (2.6), we can obtain the optimal sampling duration ($t_{\text{samp\_period}}$) for different data packet rates. For instance, it may be observed from Figure 2.6 that the minimum average power consumption obtained for 2 packets per second and 0.5 packets per second are approximately 20 ms and 50 ms, respectively. These values correspond to the results listed in Table 2.4

Table 2.3: Parameter values for the CC2420 [42] radio model used in OMNET++ v4.0 simulations.

Parameter	*Value in the CC2420 model*
Power in receive mode (P_{rx})	62.04 mW
Power in channel polling (P_{poll})	62.04 mW
Power in transmit mode (P_{tx})	57.42 mW
Power in sleep mode (P_{sleep})	69.3 nW
Time for one channel polling (t_{poll_once})	0.128×8 ms
Time to transmit/receive a bit (t_b)	$1/250$ ms

which are obtained from the analytical expression using Equation (2.6). Thus, our simulation results perfectly adhere to the analytical results.

2.4.4 Experimental Performance Analysis

We have evaluated TrawMAC both in the OMNeT++ v4.0 network simulator and using our TinyOS 2.x [63] implementation on TelosB [14] nodes. Simulation based study explores the behaviour of TrawMAC in large scale networks with different routing protocols and for a comparison with WiseMAC, which is a state-of-the-art preamble sampling protocol. A comprehensive study by Langendoen *et al.* [58] shows that WiseMAC outperforms many other preamble sampling protocols. This serves as a strong benchmark for a comparison with TrawMAC. Although the simulator allows us

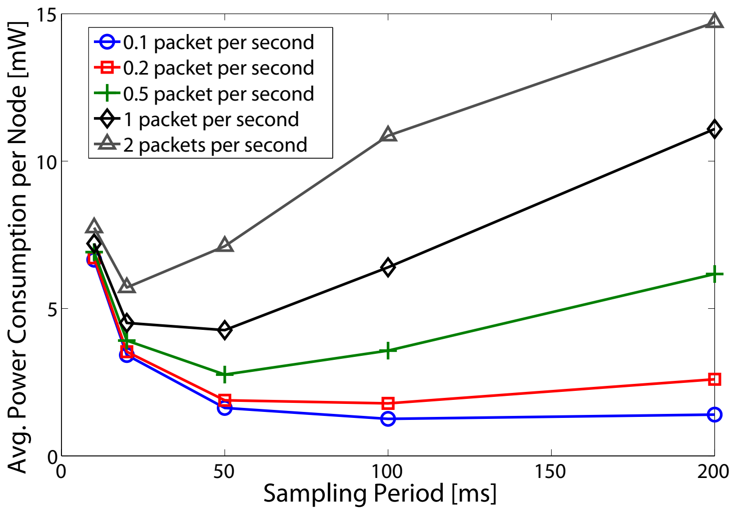


Figure 2.6: Average power consumption per node for TrawMAC at different sampling periods and different traffic data rates.

Table 2.4: Analytical values for the optimal sampling periods of TrawMAC at different data packet transmission rates.

Data packet rate [s^{-1}]	*Optimal sampling period* [ms]
0.1	105.2
0.2	74.4
0.5	47.0
1.0	33.3
2.0	23.5

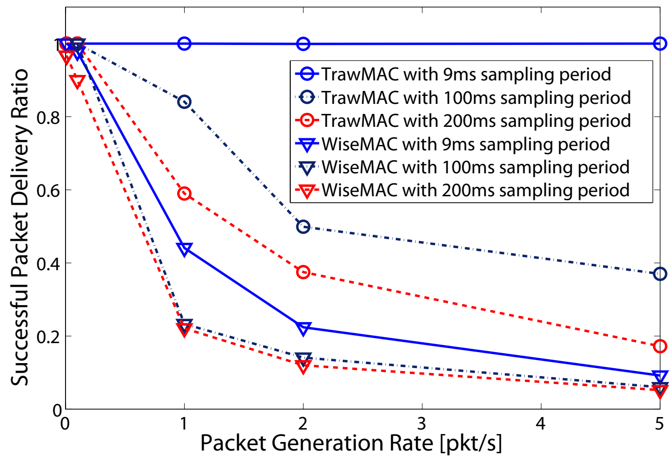
experimenting with complicated networks, the performance of simulated protocol usually differs from real node implementations since the radio models are not calibrated exactly according to the hardware, and the real wireless medium can only be modeled with approximations as we have highlighted in our empirical study. In order to validate the trend exhibited by the simulated protocol, we have carried out experiments on a a small scale testbed of TelosB nodes and compared the results to a widely used preamble sampling protocol, B-MAC+ [45].

Evaluation based on Simulation Studies
In our simulation based evaluation of TrawMAC, we have focused on three main interrelated performance metrics namely the energy consumption, the packet delivery ratio, and the end-to-end latency. The packet delivery ratio and the end-to-end latency was measured at the application level. OMNeT++ v4.0 simulator with CC2420 radio model[3]. Three scenarios were set up with different network sizes and mobility patterns to give an overview of the behaviour of TrawMAC in comparison with WiseMAC. We have used Mobility Framework [65] for generating mobility patterns in OMNET++ 4.0 simulations.

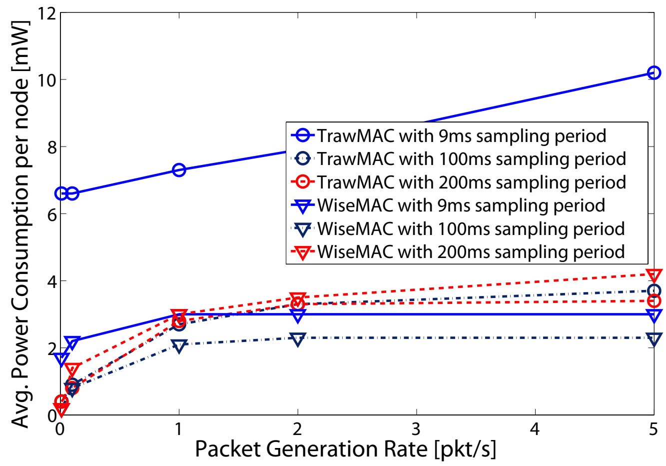
Comparison of TrawMAC with WiseMAC was carried out at the same sampling periods and application packet generation rates. Due to different CCA algorithm, TrawMAC polls the channel for a longer duration at each wake-up instance as compared to WiseMAC. This is because of the preamble strobing technique used in TrawMAC as nodes are required to wait for an acknowledgement between preamble frame transmissions and the silent intervals are needed to be covered in order to prevent false negative channel detection by other nodes. A longer channel polling duration results in a larger effective duty cycle and thus leads to relatively higher power consumption for TrawMAC as compared to WiseMAC at the same sampling period.

In one of the scenarios, 10 nodes were placed within a single hop communication distance with each other and transmit data packets to only one of the nodes (sink) using different packet generation rates. Figure 2.7 shows the performance comparison

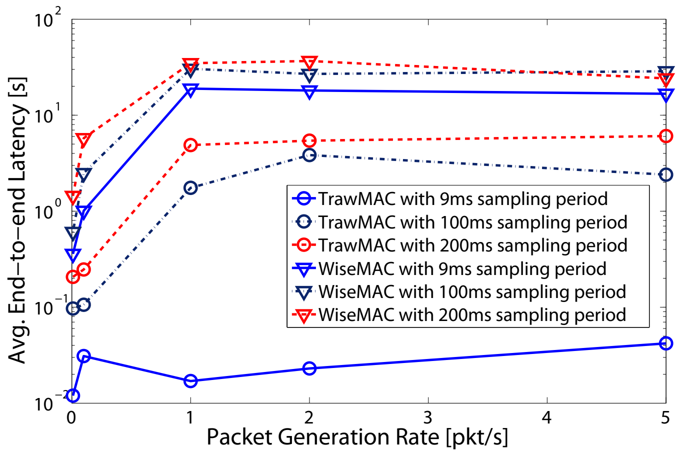
[3]The CC2420 radio model for OMNET++ 4.0 was developed by Centre Suisse d'Electronique et de Microtechnique, Neuchatel, Switzerland under the EU funded WASP [64] project grant.



(a) Packet Delivery Ratio.



(b) Average Power Consumption.



(c) Average End-to-end Latency.

Figure 2.7: Performance comparison of TrawMAC and WiseMAC in a single hop 10-node network at different average packet generation rates.

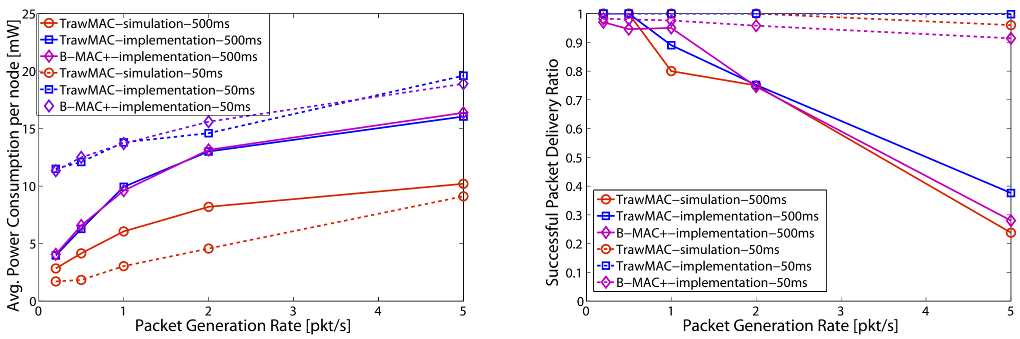
between TrawMAC and WiseMAC in this scenario. From Figure 2.7b, it can be observed that TrawMAC has higher average power consumption than WiseMAC while having the same sampling period. It is partially due to the relatively longer channel sensing duration required for TrawMAC as we have described before. Furthermore, the network traffic in this scenario is unidirectional, i.e., from nodes to the sink but not vice versa. This way the TrawMAC nodes are disadvantageous and are unable to use the sleep schedule information of the sink node. WiseMAC nodes, on the other hand, gather the sleep schedule information of the sink node from the acknowledgement packets. Although TrawMAC shows a poorer energy saving ability in this particular scenario, Figures 2.7a and 2.7c indicate that it clearly outperforms in terms of packet delivery ratio and the offered latency at different sampling periods. Increasing the sampling period values results in declination in both packet delivery ratio and the data communication latency of TrawMAC as well as for WiseMAC. However, increasing the sampling period significantly reduces the power consumption for TrawMAC. Overall, the combined effect enables TrawMAC at a sampling period of 200 ms to outperform WiseMAC in all the three metrics. It is also obvious that in a more realistic scenario when nodes exchange data over bi-directional links, TrawMAC achieves more energy savings by making use of the sleep schedules of the neighbouring nodes as we have found experimentally [17].

We have carried out the comparison of TrawMAC with WiseMAC in multihop networks with different number of nodes using different routing approaches. We refer the reader to [17] for a performance comparison in terms of the packet delivery ratio, the power consumption and the end-to-end latency. Our results indicate that overall TrawMAC has comparative edge over WiseMAC, especially in the presence of network dynamics and when higher traffic volumes are to be supported. This is because of two main reasons. When there are dynamics in a network, routing protocols trigger route discovery and new route establishments, which generate considerable amount of broadcast transmissions at the link level and the ability of TrawMAC to efficiently handle broadcast transmissions gives it an edge over WiseMAC. Secondly, aggregating multiple data packets and carrying out multi-frame transmission in TrawMAC reduces the effective preamble transmission overhead, which eventually leads to significant energy and bandwidth savings.

Evaluation based on Hardware implementation

We have implemented TrawMAC in TinyOS 2.x for TelosB nodes. In order to validate the performance trends and the behaviour of TrawMAC implementation in OMNET++ v4.0 to its implementation on TelosB nodes, we have conducted comparative studies. Furthermore, we have analyzed the performance of TrawMAC against B-MAC+ [45] in exactly the same experimental settings. B-MAC+ is the default MAC protocol for CC2420 radio based platforms in TinyOS 2.x and the code is publically available from the TinyOS 2.x repository [66]. Figure 2.8 shows the protocol behaviour in terms of power consumption and the application layer packet delivery ratio in a small network consisting of five nodes placed within the transmission range of each other and transmitting packets to the sink with different sampling periods at different packet generation rates. This simple unidirectional scenario, however, disfavors TrawMAC in a way that it would not benefit from the sleep schedules of the neighbouring nodes. TrawMAC and B-MAC+ implementations exhibit similar power consumption in this scenario, while the packet delivery ratio of TrawMAC is similar to B-MAC+.

The simulation results of TrawMAC show similar trends as its implementation on TelosB nodes. The average power consumption increases with higher packet generation rate and with lower sampling rates. Although the behaviour is similar, simulation results show a lower overall power consumption offset as compared to the real hardware implementation. It is due to different channel polling duration for the two cases. The channel polling duration in the simulations is set to be equal to the duration of taking eight samples (each consisting of four symbols) as described by Polastre *et al.* [36]. However, in the TelosB implementation, due to the SPI bus speed limitation, loading of the transmission buffer each time before any frame transmission takes a significant amount of time. This reloading time is not modeled in the CC2420 radio model in OMNeT++ v4.0. During this duration, there is no on-going radio transmission while the MAC itself is in the packet transmission process. In order to make sure that other nodes might not falsely detect channel to be inactive during this duration, the channel polling duration needs to be extended to cover the buffer loading time as well as the timeout interval for an acknowledgement packet. This results in a channel polling duration to be ten times longer in TelosB implementation than the duration set in simulations. Here we note that many network simulators, while modeling the real radio transceivers either ignore some of the factors or more importantly behaviour of platforms. For instance the communication delays between microcontroller and radio transceiver are often not modeled.



(a) Average Power Consumption. (b) Average Delivery Ratio.

Figure 2.8: Performance comparison between the implementation and simulation results of TrawMAC and B-MAC+ in a single hop network consisting of five nodes.

2.4.5 Discussion and Conclusions

We have described TrawMAC protocol in detail, which is able to combine the strengths of different state-of-the-art preamble sampling MAC schemes and offsets their weaknesses depending upon the network traffic conditions. This makes our protocol highly adaptable and suitable to a wide range of applications with diverse traffic requirements. The technique to intelligently supplant a normal broadcast transmission with multiple unicast transmissions and/or broadcast(s) with shorter preamble length leads to considerable energy savings. Another unique feature of TrawMAC is is its ability to aggregate data packets through a packet queue. This not only allows saving significant amount of energy by avoiding transmission and reception of long preambles but also helps in lessening congestion in a network thereby improving the effective channel utilization. Simulations as well as implementation on TelosB testbed show that TrawMAC outperforms B-MAC+ and WiseMAC. It is worth noting that WiseMAC has previously been shown to outperform other preamble sampling MAC solutions [58]. We also note that though commonly used simulators with real radio models are able to depict similar trends and trade-offs among different metrics as observed on real node testbed implementations, yet they often fail to accurately correspond because of simplified assumptions to keep simulations tractable and unmodeled parameters.

2.5 MULTI-RADIO MAC DESIGN FOR ENERGY EFFICIENCY

There are two major sources of energy consumption in low-power embedded networks, namely idle listening and ephemeral burst data exchange. We have designed a dual radio based MAC protocol, which combines the advantages of the two radios operating in different frequency bands in order to reduce these two sources of energy consumption. In this section, we will present the design rationale of the protocol and its extensive empirical performance evaluation in terms of power consumption and latency under various traffic loads and duty cycles. We will also describe the experimental performance comparison of the protocol with the widely used B-MAC protocol on COTS low-power nodes. The results indicate that our protocol achieves significant performance gains over B-MAC in identical application and network conditions. We have derived an analytical expression for the optimum transmit power level ratio on the two radios for minimizing energy consumption. Furthermore, we have modeled the relationship of optimal duty cycle at nodes to a given traffic load condition in a network. We have validated the model through experimental results obtained from a prototype implementation of the protocol on commercially available low-power nodes.

2.5.1 Design Philosophy

Generally, the traffic load in low-power embedded networks is very small. In an ideal case, nodes switch on their radios only for data transmission/reception and turn them off when there is no traffic in the network. While it is easy to define when data transmission is required, reception is usually unpredictable to a node. Low-power nodes therefore need to listen to the channel periodically in anticipation for potential packet transmissions. If no packet is detected, listening to the medium goes wasted. The main activities on a radio include idle listening to the medium, and data transmission and reception. The amount of the required data exchange depends upon an application, however idle listening remains dominant in majority of the low-power embedded wireless applications [24].

In order to minimize the overall energy consumption in a network, idle listening as well as the data transmission/reception has to be optimized. Since these two operations are very distinct, they impose different but unique requirements on the capabilities of a radio interface. Idle listening or channel polling operation is carried out frequently in a periodic manner while data communication is rare and ephemeral. Many single radio platform based MAC solutions remain handicapped due to the absence of specialized design to simultaneously cater these different needs. Here, we describe our Multi Radio MAC (MR-MAC) protocol, which is designed to minimize the two dominant sources of energy consumption. MR-MAC uses a low-power sniffer radio to effectively handle idle listening and a fast/bursty radio for data communication. The operating frequencies of the two radios are selected based on the characteristic features needed by the two operations. The low-power slow data rate sniffer radio, which is used for exchanging control information and coordinating the actual data communication, operates in low frequency band. Owing to typically having a larger bandwidth in higher frequency bands, data is transmitted over a fast bit rate radio operating in a higher frequency band. MR-MAC protocol design combines the advantages of the two radios very effectively and achieves a highly energy efficient performance as compared to the contemporary single radio based solutions as we will present in Section 2.5.6. MR-MAC uses preamble sampling principle since it does not impose the need for explicit synchronization among nodes in a network thereby minimizing coordination overhead [2–4]. The preamble sampling technique also effectively handles network dynamics such as node mobility, old nodes dying and new nodes appearing.

2.5.2 Protocol Details

MR-MAC is a *p*-persistent preamble sampling MAC protocol which uses dedicated high and low frequency bands for data and control, respectively. MR-MAC aims at combining the strengths of the radios in low and high frequency bands in order to achieve power efficiency. Generally, higher frequency bands have larger bandwidths which allow higher communication data rates. Oppermann *et al.* [67] show that generally radios operating in higher frequency bands consume less energy per bit as compared to radios operating in lower frequency bands. This fact inspired us to use a radio operating in high frequency band for bursty data communication. Low frequency band radios, on the other hand, consume less energy in idle listening at a given receiver sensitivity threshold. Also, a radio operating at a lower frequency band requires less transmit power level to achieve a certain communication range as compared to a radio operating at a higher frequency.

Instead of transmitting a monolithic preamble sequence without any useful information, MR-MAC transmits a number of small preamble frames containing control information. It also employs a number of preamble shortening techniques as described later in this section. Preamble sequence transmission/reception in general consumes relatively larger amount of energy for radios operating in higher frequency bands and supporting higher data rates than radios operating in lower frequency bands with lower data rates. MR-MAC performs channel polling operations only in the low frequency band, while the high frequency transceiver is turned on only during data transmission/reception. Overall, this approach leads to highly optimized utilization of the radio resources. MR-MAC has the ability to transmit multiple data frames with a single preamble reservation. This allows MR-MAC to efficiently handle large amount of traffic loads in an energy efficient manner over the bursty radio. Figure 2.9 shows a simplified state machine of MR-MAC, where the sniffer radio or Low Frequency Radio (LFR) and bursty radio or High Frequency Radio (HFR) perform different operations in a coordinated fashion.

In the following, we describe the preamble length optimization applied to MR-MAC in order to achieve energy conservation.

Preamble Framelets

MR-MAC divides a monolithic preamble sequence into small preamble frames, which are transmitted back to back to form Micro-Frame Preamble (MFP). These small pre-

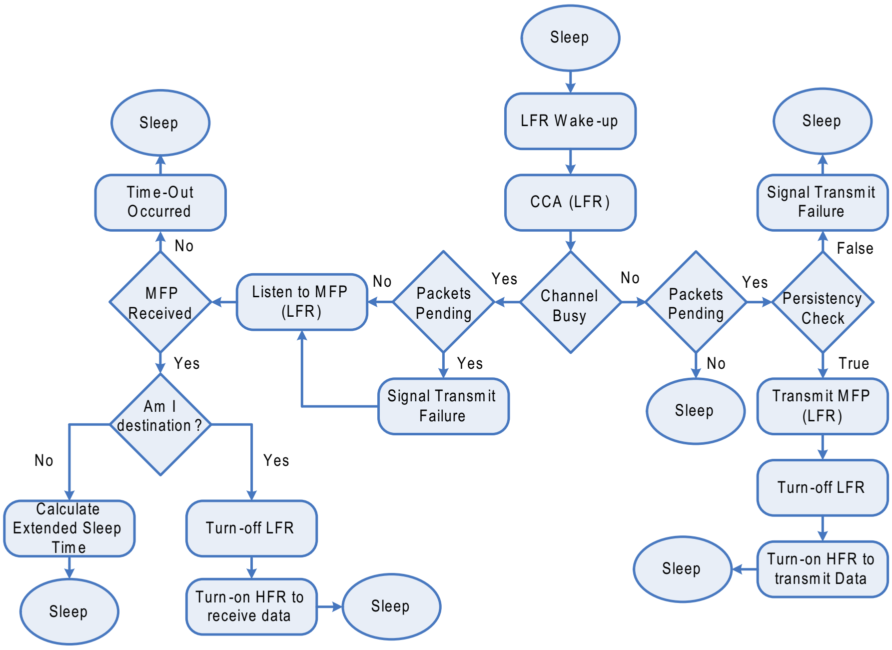


Figure 2.9: Simplified state-machine based flowchart of different operations in MR-MAC. The preamble sampling operation is carried out over the Low Frequency Radio (LFR) while bursty data is transmitted over the High Frequency Radio (HFR).

amble frames are also referred to as framelets. Each framelet contains control information such as the radio byte sequence for PLL (Phase Locked Loop) followed by synchronization bytes for a receiving node. The receiver makes bit offset adjustments based on the synchronization bytes to correctly receive the rest of the preamble frame. The address information, which is included in the framelet allows non-addressed nodes to go to sleep without listening to the rest of the preamble sequence and data packets following the preamble. The addressed node also goes to sleep in order to avoid listening to the rest of the framelets, which do not contain any useful information. A down counter value is transmitted in every framelet, which indicates the time offset for beginning of a data frame transmission. An addressed node can thereby precisely estimate when to turn on its high frequency band radio. The size of the data packet is also included in preamble framelets so that an addressed node is able to know how many data packets it needs to receive. A non-addressed node is able to estimate how long the medium is going to be busy based on these two values and accordingly extends its sleeping period to achieve additional energy savings.

Our experiments have shown that if the data size is smaller than a certain threshold (depending upon the radio transceiver characteristics), it is more energy efficient to piggyback data into preamble framelets. Piggybacking data into the preamble framelets forms Data Frame Preambles (DFPs) [68]. When DFPs are used in MR-MAC, the radio operating in high frequency band is not used. Figure 2.10 shows the operational cycle of MR-MAC for the case of MFP and DFP broadcast transmissions.

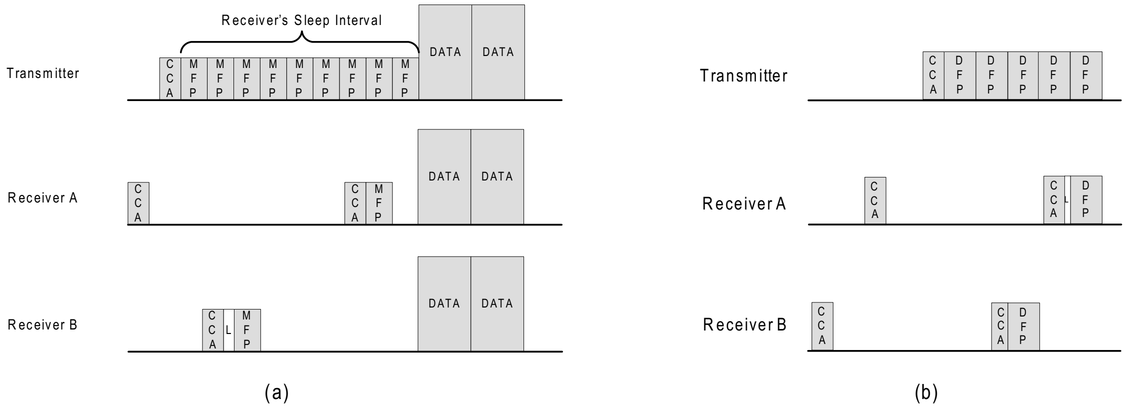


Figure 2.10: Broadcast transmission in MR-MAC, where a long preamble sequence is divided into small preamble framelets. (a) After receiving an MFP, both the receivers A and B, are implicitly synchronized in order to receive data packets over the bursty radio. (b) Data is piggy-backed to the preamble framelets. The two receivers, A and B after receiving a complete DFP can sleep during the rest of preamble transmission.

Preamble Optimization based on the Neighbourhood Sleep Schedules
In unicast transmission for preamble sampling based MAC protocols, the length of the preamble can be reduced drastically if the transmitter knows the sleep schedule of the receiver [38]. In MR-MAC, all the nodes maintain a neighbourhood sleep schedule information similar to WiseMAC. Each preamble framelet contains the information of the next wake-up offset of a node. A node receiving a framelet updates the corresponding neighbourhood sleep schedule information table based on the next wake-up offset value. A perfect timing information of a destination allows a transmitter to delay the transmission of the packet until the destination is scheduled to wake up. In this way, the transmitter needs to send only one preamble framelet. In contrast to the approach of WiseMAC protocol to announce the sleep schedules in the acknowledgment frames, MR-MAC announces the sleep schedules in the preamble frames, which also allows non-addressed nodes to update their neighbourhood timing information while overhearing a preamble framelet.

Preamble Strobing
Preamble strobing [44] is a technique used in unicast transmission where a transmitter sends a preamble framelet and waits for its acknowledgement. If an acknowledgement is not received within a certain timeout duration, subsequent preamble frames are sent. If an acknowledgement is received, preamble frame transmission is stopped and data packet is transmitted immediately afterwards. The maximum number of preamble framelets required to be sent corresponds to the periodic channel check interval. This is exercised only in the worst case, when a false estimation of the sleep schedule results in the maximum offset between the sleep schedules of the transmitter and a

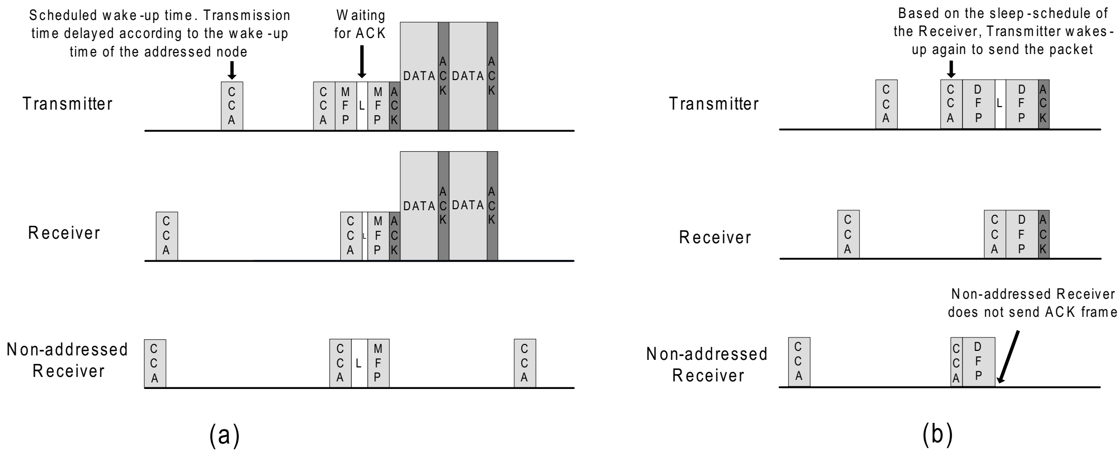


Figure 2.11: Unicast transmission in MR-MAC, where preamble strobing is combined with the sleep schedule of the neighbouring nodes. The imprecise sleep schedule estimation of a destination is compensated through the preamble strobing operation. (a) Upon receiving the acknowledgement of an MFP, data frames are immediately transmitted over the bursty radio. A non-addressed node, however, goes to sleep. (b) A receiving node acknowledges a DFP and the transmitter stops sending subsequent DFPs. A non-addressed node does not acknowledge a DFP and goes to sleep.

receiver, the receiver is out of the transmission range or the acknowledgement frame is lost. Preamble strobing has its advantages especially when neighbourhood schedule is either unavailable or is unreliable. Furthermore, offsets might be introduced in the estimated sleep schedule information of the receivers due to clock jitter accumulation over extended period of time. In MR-MAC protocol, preamble strobing technique is combined with the neighbourhood sleep schedule based preamble optimization. This way a node performs preamble strobing with additional information on the sleep schedule of the receiver. This allows robustness against clock drifts and network dynamics. Figure 2.11 shows the operational cycle of MR-MAC for unicast transmissions. Preamble optimization based on the neighbourhood sleep schedules combined with the preamble strobing technique is only used in unicast transmissions. We have evaluated the performance of MR-MAC protocol with and without unicast optimization (cf. Section 2.5.6).

2.5.3 Analytical Model

In this section, we will present the analytical model of MR-MAC protocol and derive an expression for the optimum duty cycle giving minimum energy consumption for a given traffic load in a network. Since control overhead and hence energy consumption directly depends on the type of the traffic, we have separately analyzed the cases for unicast and broadcast traffic types.

Models and Parameters

Consider a network of $n + 1$ nodes and assume that each node transmits data packets of length l_{pkt} periodically at a rate r_{data} per second. Each node consumes power in the operations: radio setup, carrier sensing, transmitting control information, transmitting data packets, receiving control information, receiving data packets, channel polling and sleep state respectively denoted by $P_{\text{radio\_setup}}$, P_{cs}, $P_{\text{tx\_1}}$, $P_{\text{tx\_2}}$, $P_{\text{rx\_1}}$, $P_{\text{rx\_2}}$, P_{poll} and P_{sleep}. Table 2.5 lists the terms used in our model.

Table 2.5: Notation for different terms used in the analytical model of MR-MAC.

Notation	*Meanings*
$t_{\text{poll\_once}}$	Single channel polling interval
$t_{\text{radio\_setup\_once}}$	Single radio setup interval
$t_{\text{samp\_period}}$	Channel sampling period
L_{mfp}	Length of a micro-frame preamble
N_{mfp}	Number of micro-frames
$t_{\text{cs\_once}}$	Channel carrier sensing interval
l_{pkt}	Length of the data packet
t_{b1}	Bit duration corresponding to low frequency radio data rate
t_{b2}	Bit duration corresponding to high frequency radio data rate
$l_{\text{ack\_1}}$	Length of the acknowledgement packets over the low frequency radio
$l_{\text{ack\_2}}$	Length of the acknowledgement packets over the high frequency radio

Broadcast

The overall energy consumption at a node is the sum of the energy spent in each operation and is given by,

$$E = E_{\text{radio\_setup}} + E_{\text{poll}} + E_{\text{rx}} + E_{\text{cs}} + E_{\text{tx}} + E_{\text{sleep}} \tag{2.7}$$

Expressing energy as the product of power and time using the variables defined gives,

$$\begin{aligned} E =& P_{\text{radio\_setup}} t_{\text{radio\_setup}} + P_{\text{poll}} t_{\text{poll}} + P_{\text{rx\_1}} t_{\text{rx\_1}} + \\ & P_{\text{rx\_2}} t_{\text{rx\_2}} + P_{\text{cs}} t_{\text{cs}} + P_{\text{tx\_1}} t_{\text{tx\_1}} + P_{\text{tx\_2}} t_{\text{tx\_2}} + P_{\text{sleep}} t_{\text{sleep}}. \end{aligned} \tag{2.8}$$

Normalizing the individual time durations in different operations by the channel sampling period, $t_{\text{samp\_period}} = L_{\text{mfp}} N_{\text{mfp}} t_{\text{b1}}$.

$$\begin{aligned}
t_{\text{radio_setup}} &= \frac{t_{\text{radio_setup_once}}}{t_{\text{samp_period}}} \\
t_{\text{poll}} &= \frac{t_{\text{poll_once}}}{t_{\text{samp_period}}} \\
t_{\text{rx_1}} &= n r_{\text{data}} (1.5 L_{\text{mfp}} t_{\text{b1}}) \\
t_{\text{rx_2}} &= n r_{\text{data}} l_{\text{pkt}} t_{\text{b2}} \\
t_{\text{cs}} &= r_{\text{data}} t_{\text{cs_once}} \\
t_{\text{tx_1}} &= r_{\text{data}} N_{\text{mfp}} L_{\text{mfp}} t_{\text{b1}} \\
t_{\text{tx_2}} &= r_{\text{data}} l_{\text{pkt}} t_{\text{b2}} \\
t_{\text{sleep}} &= 1 - t_{\text{radio_setup}} - t_{\text{poll}} - t_{\text{rx_1}} - \\
&\quad t_{\text{rx_2}} - t_{\text{cs}} - t_{\text{tx_1}} - t_{\text{tx_2}}
\end{aligned}$$

One should note that MR-MAC requires 1 to 2 micro preamble frames in order to determine the destination address and timings of the data packet in the best and worst cases, respectively. This gives on average a reception of 1.5 micro preamble frames. In this derivation, we consider the radio setup time of only the sniffer radio. The initialization of the bursty radio is carried out in parallel to preamble transmission and hence it does not inflict an extra timing overhead. The initialization of the sniffer radio accounts for the time needed to send the configuration commands over the SPI or UART interface to start the radio in a desired configuration. It also accounts for the time required by clock crystal to stabilize. The high frequency radio configuration setup is carried out in parallel to the transmission/reception of the preamble frames. Since it does not cause any extra delays, the setup time for high frequency radio is not modeled.

Our target is to find the relationship of the sampling period (directly related to duty cycle) which leads to the minimum energy consumption. Since the sampling period is directly related to the length of the preamble, we find the number of micro-frames corresponding to the minimum energy consumption. Plugging the terms in Equation (2.8) and taking the derivative w.r.t. N_{mfp} gives,

$$\begin{aligned}
\frac{dE}{dN_{\text{mfp}}} = &-\frac{P_{\text{radio_setup}} t_{\text{radio_setup_once}}}{L_{\text{mfp}} N_{\text{mfp}}^2 t_{\text{b1}}} - \frac{P_{\text{poll}} t_{\text{poll_once}}}{L_{\text{mfp}} N_{\text{mfp}}^2 t_{\text{b1}}} + \\
&P_{\text{tx_1}} r_{\text{data}} L_{\text{mfp}} t_{\text{b1}} + \frac{P_{\text{sleep}} t_{\text{radio_setup_once}}}{L_{\text{mfp}} N_{\text{mfp}}^2 t_{\text{b1}}} + \\
&\frac{P_{\text{sleep}} t_{\text{poll_once}}}{L_{\text{mfp}} N_{\text{mfp}}^2 t_{\text{b1}}} - r_{\text{data}} L_{\text{mfp}} t_{\text{b1}} P_{\text{sleep}}.
\end{aligned}$$

Putting $\frac{dE}{dN_{\text{mfp}}} = 0$ and simplifying the terms gives the optimum number of micro-frames ($\hat{N}_{\text{mfp}}$) for the minimum energy consumption in Equation (2.9).

$$\hat{N}_{\text{mfp}} = \Big(\frac{t_{\text{poll\_once}}(P_{\text{poll}} - P_{\text{sleep}})}{L^2_{\text{mfp}} t^2_{\text{b1}} r_{\text{data}}(P_{\text{tx\_1}} - P_{\text{sleep}})} + \frac{t_{\text{radio\_setup\_once}}(P_{\text{radio\_setup}} - P_{\text{sleep}})}{L^2_{\text{mfp}} t^2_{\text{b1}} r_{\text{data}}(P_{\text{tx\_1}} - P_{\text{sleep}})}\Big)^{\frac{1}{2}} \tag{2.9}$$

Since $\hat{N}_{\text{mfp}} \in$ Natural numbers, we take the ceiling value. It may be noted that the expression for the optimum number of micro-frames is independent of the number of nodes in a network because we consider a congestion free case where all the nodes are able to transmit their queued packets. The network size $n + 1$ governs a lower bound, $n \leq \frac{1}{r_{\text{data}}(L_{\text{mfp}} t_{\text{b1}} N_{\text{mfp}} + l_{\text{pkt}} t_{\text{b2}})}$. The optimal sampling time expression is,

$$S.T._{\text{opt}} = \hat{N}_{\text{mfp}} L_{\text{mfp}} t_{\text{b1}}. \tag{2.10}$$

Unicast

The total energy consumption at a node in the case of unicast transmission takes the same form as Equation (2.8). Unlike the broadcast transmission, a node happens to be the destination for k packets out of the total n packets it hears from its neighbours. MR-MAC tries to optimize the number of micro-frames to be sent for unicast transmission by using the neighbourhood sleep schedules. In the best case, only one micro-frame right at the instant when the destination is scheduled to wake-up is required to be sent. In the absence of any timing information, the number of micro-frames needed to be sent depends upon the offset between the sleep schedules of the transmitting and receiving nodes. In the worst case, the number of micro-frames needed to be sent becomes the same as in the case of broadcast transmission, given by Equation (2.9). The expressions for time durations are given by:

$$\begin{aligned}
t_{\text{radio_setup}} &= \frac{t_{\text{radio_setup_once}}}{t_{\text{samp_period}}} \\
t_{\text{poll}} &= \frac{t_{\text{poll_once}}}{t_{\text{samp_period}}} \\
t_{\text{samp_period}} &= L_{\text{mfp}} N_{\text{mfp}} t_{\text{b1}} \\
t_{\text{rx_1}} &= n r_{\text{data}}(1.5 L_{\text{mfp}} t_{\text{b1}}) + r_{\text{data}} l_{\text{ack_1}} t_{\text{b1}} \\
t_{\text{rx_2}} &= k r_{\text{data}} l_{\text{pkt}} t_{\text{b2}} + r_{\text{data}} l_{\text{ack_2}} t_{\text{b2}} \\
t_{\text{cs}} &= r_{\text{data}} t_{\text{cs_once}} \\
t_{\text{tx_1}} &= r_{\text{data}} N_{\text{mfp}} L_{\text{mfp}} t_{\text{b1}} + k r_{\text{data}} l_{\text{ack_1}} t_{\text{b1}} \\
t_{\text{tx_2}} &= r_{\text{data}} l_{\text{pkt}} t_{\text{b2}} + k r_{\text{data}} l_{\text{ack_2}} t_{\text{b2}} \\
t_{\text{sleep}} &= 1 - t_{\text{radio_setup}} - t_{\text{poll}} - t_{\text{rx_1}} - \\
&\quad t_{\text{rx_2}} - t_{\text{cs}} - t_{\text{tx_1}} - t_{\text{tx_2}}
\end{aligned}$$

2.5.4 Prototype Hardware

Our hardware prototype platform (as shown in Figure 2.12) consists of a TelosB and a CC1000 [69] radio module. TelosB node has an on-board MSP430 [70] series microcontroller and a CC2420 [42] radio transceiver from Texas Instruments. We have interfaced an external CC1000 radio module to the microcontroller. Both the configuration and signaling interfaces of the CC1000 radio chip are connected through the extension connectors on TelosB (cf. Figure A.1). The IEEE 802.15.4 compliant CC2420 radio operates in 2.4 GHz frequency band. This packetized radio acts as the bursty radio while CC1000 radio, operating in 433 MHz frequency band, acts as the low frequency sniffer radio. The CC1000 radio provides byte level interface and consumes much less power in idle mode and is therefore suitable for preamble sampling operation and control purposes. The CC1000 radio supports multiple data rates ranging from 0.6 kbps to 76.8 kbps. The CC2420 radio transceiver offers a data rate of 250 kbps and is therefore used for burst data transmission. We have implemented MR-MAC protocol using nesC programming language in TinyOS 2.x operating system and followed the recommended hardware abstraction architecture [71].

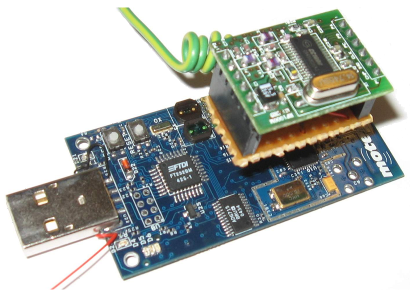


Figure 2.12: Snapshot of the prototype platform for MR-MAC implementation.

2.5.5 Optimum Transmit Power Levels

In dual radio platforms, the transmission power levels at both the radios are set high enough so that both the radios on the receiving node are in their reception range. If either of the two radios is unable to reach the receiver, the communication remains unsuccessful. If the control channel radio does not reach a destination node, data communication cannot be established at all while in the converse case, data packet remains undelivered. As a crude design principle, the transmit range of the bursty data radio should at least be the same as that of the control channel in order to allow the bursty radio to deliver data packets which have already been announced on the control channel. However, setting the transmit power level of the data channel and hence its range to be longer leads to energy wastage. Ideally, the transmit powers of the two radios should be set so as to achieve the same coverage range. Owing to the different nature of the radios on the dual radio platform, heuristically setting the transmit power is not enough and it is also necessary that the two radios give the same packet error rates. Since the packet error rate depends upon the packet size and the two radios have different packet lengths, optimally setting the transmit power levels of the two radios becomes challenging.

In the following, we will derive an analytical expression for the transmit power level ratios of the two radios. The received power P_r at a certain distance d is given by Friis transmit equation, $P_r = (P_t G_t G_r \lambda^2)/(4\pi d)^2$, where P_t is the transmit power, λ is the wavelength of the radio wave while G_r and G_t represent the gains of the receive and transmit antennas, respectively. The ratio of the transmit powers of the two radios, (1 and 2, respectively) for an equal range is given by, $(P_{t1} G_{t1} G_{r1} \lambda_1^2)/P_{r1} = (P_{t2} G_{t2} G_{r2} \lambda_2^2)/P_{r2}$. Let's assume that the receive and transmit gains of each of the antennas are the same, i.e., $G_{t1} = G_{r1} = G_1$ and $G_{t2} = G_{r2} = G_2$. Thus, the ratio of the transmit powers is related to the ratio of the received power of two radios with equation $P_{t1}/P_{t2} = (P_{r1} G_2^2 \lambda_2^2)/(P_{r2} G_1^2 \lambda_1^2)$.

The two radios are targeted for different goals (low-power sniffing and bursty communication) on the dual-radio platform and therefore, potentially use different modulation schemes and have different receiver sensitivities. This leads to different Bit Error Rates (BER) (and hence different Packet Error Rates (PER), $PER = 1-(1-BER)^L$, where L is number of bits in one packet) on the two radios at a particular received power level. Here, PER assumes uncorrelated (no bursts) error statistics for simplicity. The optimal transmit power level ratios of the two radios can be determined by obtaining the ratio of the received power levels of the two radios at the receiving node giving the desired PER. The received power can be obtained from the Signal to Noise Ratio (SNR) versus BER curves through the relation, $SNR = P_r/N_0 = E_b/(N_0 R)$. Here N_0 represents the noise power spectral density, E_b represents the energy per bit and R is the bit rate. The noise spectral density depends upon the temperature of the antenna and is given by $N_0 = k_B T_0$, where k_B is the Boltzmann constant and T_0 is the temperature at the antenna. Therefore, P_r can be expressed as $P_r = (E_b R k_B T_0)/N_0$, and the ratio of the optimal transmit powers of the two radios is given by,

$$\frac{P_{t1}}{P_{t2}} = \frac{(\frac{E_b}{N_0})_1 R_1 T_{01} G_2^2 \lambda_2^2}{(\frac{E_b}{N_0})_2 R_2 T_{02} G_1^2 \lambda_1^2}. \tag{2.11}$$

We can choose $E_b/N_0 = SNR$ using the SNR-BER curves, which leads to the same PER value on both the radios. On our prototype, the control channel radio operating in the lower frequency band uses FSK modulation while the bursty data radio operating in higher frequency band uses OQPSK modulation. The probability of bit errors for the two modulation schemes is given by $BER_{FSK} = Q(\sqrt{E_b/N_0})$ and $BER_{OQPSK} = Q(\sqrt{2E_b/N_0})$ in AWGN channels. Since the packet error rates of the two radios are required to be the same, $PER_1 = PER_2$. Substituting the equations for the two radios on our prototype gives,

$$(\frac{E_b}{N_0})_1 = (Q^{-1}(1 - (1 - Q(\sqrt{2(\frac{E_b}{N_0})_2}))^{\frac{L_2}{L_1}}))^2. \tag{2.12}$$

After selecting the SNR values of the two radios satisfying Equation (2.12), we determine the optimal transmit power level ratios the two radios on our prototype board using the Equation (2.11).

2.5.6 Performance Evaluation

We have carried out extensive performance evaluation of our prototype MR-MAC protocol implementation in terms of power consumption and latency. We have also compared and analyzed our experimental results to the widely used B-MAC protocol both on TelosB and MICA2 platforms. TelosB has an on-board CC2420 radio whereas MICA2 has a CC1000 radio. These radios are respectively the high frequency band and low frequency band radios on our prototype platform. The performance comparison studies in this case certainly give an insight on the advantages of dual radio MAC schemes on dual radio platform against those using single radio platforms. Experimental comparison for the three platforms were carried out under the same traffic loads, duty cycles, radio transmit power levels and the network size.

Validation of Empirical Results to Analytical Model

Power consumption measurements are carried out for the MR-MAC implementation on our prototype platform in order to verify the optimum duty cycle deduced from the analytical model of the protocol. From Equation (2.10), we can observe that the data transmission rate r_{data} is variable while other terms are fixed either due to radio properties or protocol design. The lowest power consumption per node should be achieved at the optimum duty cycle. We have plotted the power consumption curves at different data transmission rates in a network using the analytical expression. The parameters that we used for the analytical expression are based on our measurements and are listed in Table 2.6. The plots for power consumption per node at different duty cycles are concave in shape with a unique minimum point.

Table 2.6: Values measured on the MR-MAC prototype platform.

Parameter	*Value*
Power consumption for radio setup ($P_{\text{radio\_setup}}$)	13 mW
Power consumption in channel polling (P_{poll})	25 mW
Power consumption in transmit mode ($P_{\text{tx\_1}}$)	31 mW
Power consumption in sleep mode (P_{sleep})	1.78 mW
Time required for radio setup ($t_{\text{radio\_setup\_once}}$)	5 ms
Time required for one channel polling operation ($t_{\text{poll\_once}}$)	3.5 ms

For the hardware measurements, we considered a network of three nodes in order to have a uncongested network. The length of the preamble frame, $l_{\text{preamble\_frame}}$ was fixed to 64 bits. Figure 2.13 shows the average power consumption of a node at different duty cycles for different data packet rates in the case of broadcast transmission. It may be observed from Figure 2.13 that the minimum average power consumption obtained for 1 packet per second, 0.5 packet per second and 0.3 packet per second are at around 5 %, 3.5 % and 2 %, respectively. These values correspond closely to the

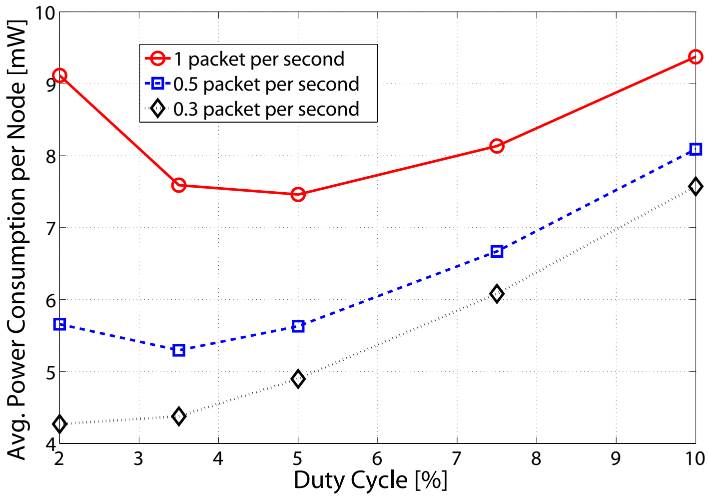


Figure 2.13: Average power consumption per node for MR-MAC at different sampling periods and under different network traffic data rates.

results listed in Table 3.2 which can be obtained from Equation (2.10). Our empirical and analytical results indicate that the optimum duty cycle values decrease as the data transmission rate decreases.

Table 2.7: Analytical values for the optimal sampling periods of MR-MAC at different data packet transmission rates.

Data packet rate [s^{-1}]	*Optimal duty cycle* [%]
1.0	5.10
0.5	3.61
0.3	2.80

Power Consumption Analysis of MR-MAC

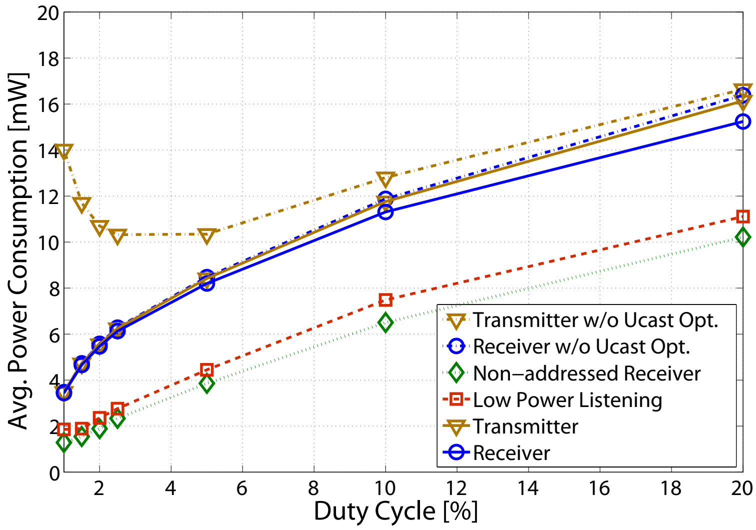
Power consumption depends on the operational duty cycle of a MAC protocol. If there is no traffic in a network, energy is drained in polling the channel periodically. Lower duty cycles obviously result in less energy consumption. When there is traffic in a network, power consumption depends on the amount of control overhead required to explicitly or implicitly synchronize nodes besides the volume of data traffic. The amount of control overhead associated with synchronization of nodes in preamble sampling protocols is directly related to the operating duty cycle values. We have measured the power consumption of MR-MAC on our prototype platform (cf. Section 2.5.4) and that of B-MAC protocol on both TelosB and MICA2 platforms. We have considered different duty cycle values and different traffic loads for a thorough evaluation.

Table 2.8: Energy and power consumption break down for different operations on MR-MAC prototype platform.

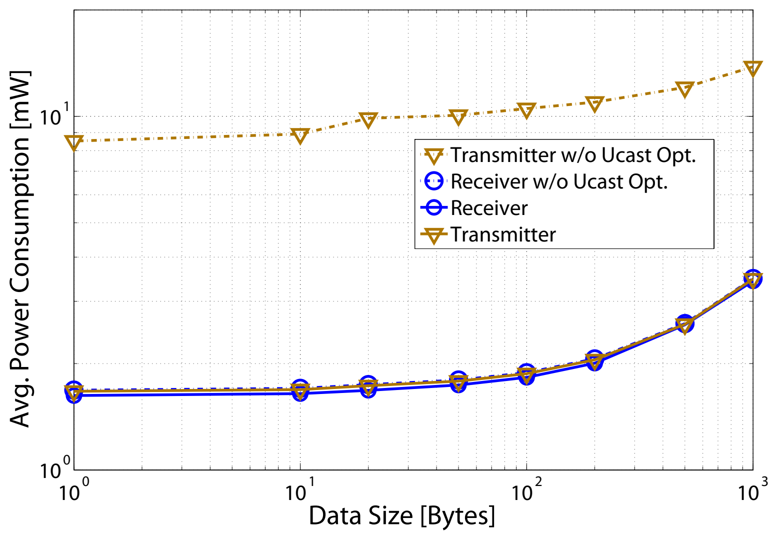
Operation break-down	*Energy* [μJ]
Turning on CC1000 to TX mode	109
Turning on CC1000 to RX mode	65.2
Turning off CC1000	6.22
Turning on CC2420 to TX mode	15.6
Turning on CC2420 to RX mode	15.7
Turning off CC2420	0.313
Operation break-down	*Power* [mW]
CC1000 in Transmission Mode	31
CC1000 in Reception Mode	25
CC2420 in Transmission Mode	52.4
CC2420 in Reception Mode	55.4

Table 2.8 lists the energy and power consumption break-down for the basic operations of both the CC1000 and the CC2420 radio chips on our platform operating at a supply voltage of 3 V. The power consumption of CC2420 chip in active mode is approximately twice as that of the CC1000 radio chip. This fact strongly supports our idea of using the CC1000 radio chip for control packets.

We have measured the power consumption of MR-MAC in unicast transmission when the transmitter can precisely estimate the sleep schedule of a receiver. Figure 2.14a shows the power consumption of MR-MAC protocol for unicast transmission with and without unicast preamble optimization. It can be observed that combining preamble strobing with neighbourhood sleep schedule based preamble optimization results in significant amount of energy savings, especially at low operating duty cycles. With the perfect knowledge of the sleep schedule of a receiver, the transmitter wakes up to send preamble framelet right at the instant when the receiver starts channel polling. A sequence of synchronization bytes (0x55 or 0xAA) enables a receiver to detect an upcoming frame. Consequently, the receiver starts searching for the radio locking sequence (0x33CC) following the synchronization bytes. The actual preamble frame containing the control information follows the radio locking sequence. In case of the perfect knowledge on the sleep schedule of a receiver, a transmitter needs to send only one framelet. The duration of the channel polling, also known as clear channel assessment duration, is long enough to cover the waiting duration for framelet acknowledgement between adjacent preamble framelets. Furthermore, both transmitter and receiver have to pay the price of receiving and transmitting the acknowledgement of a preamble frame. The slight difference in power consumption between the transmitter and the receiver is caused by the differences in power consumption of radios in different radio states, i.e., transmitting and receiving. The power consumption curve of the transmitter follows the trend of the receiver and increases with increasing duty cycle



(a)



(b)

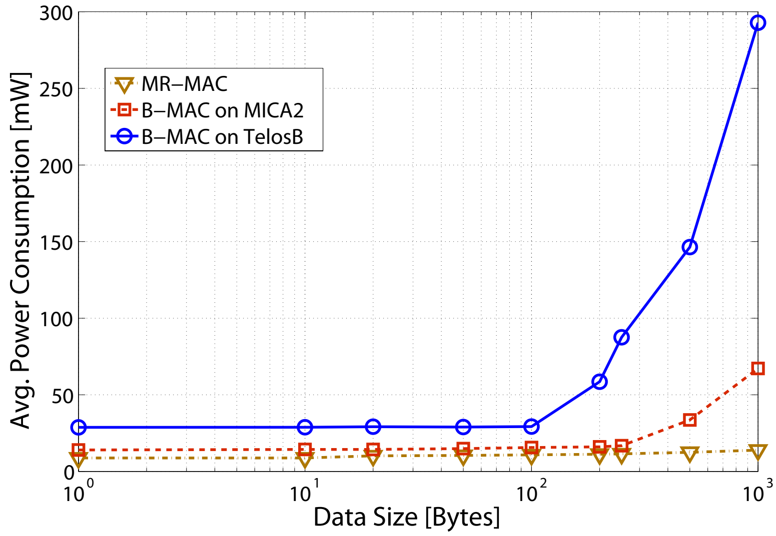
Figure 2.14: (a) Power consumption of MR-MAC running on the prototype platform at different duty cycles. The transmission rate is chosen to be 1 Hz, the considered traffic pattern is unicast and the data packet size is set to be 1000 bytes. (b) Power consumption of MR-MAC running on the prototype platform at different data sizes. The transmission rate is set to be 1 Hz, the considered traffic pattern is unicast and the duty cycle value is set to be 1 %.

due to the need for more frequent channel polling. This argument is supported by the fact that the increasing slopes of the addressed receiver, non-addressed receiver, and low power listening node (without any transmitter in the vicinity) are approximately the same. A number of factors contribute to the difference between the receiver with and without unicast optimization. Although the receiver without unicast optimization saves energy spent in acknowledgement transmission and requires a fewer radio state switching operations (from Receive-to-Transmit and Transmit-to-Receive), it needs to listen to an average of 1.5 preamble framelets instead of only one preamble frame due to the absence of implicit synchronization between nodes. Overall, the power consumption of receiver with unicast optimization is slightly lower than without the unicast optimization. The offset in energy consumption between the addressed receiver and non-addressed receiver indicates the energy consumed in data reception for the addressed node. The power consumed by the non-addressed receiver is lower than that of a low power listening node because the non-addressed receiver is able to estimate the on-going transmission duration and hence prolong its sleeping time accordingly till the end of the data transmission.

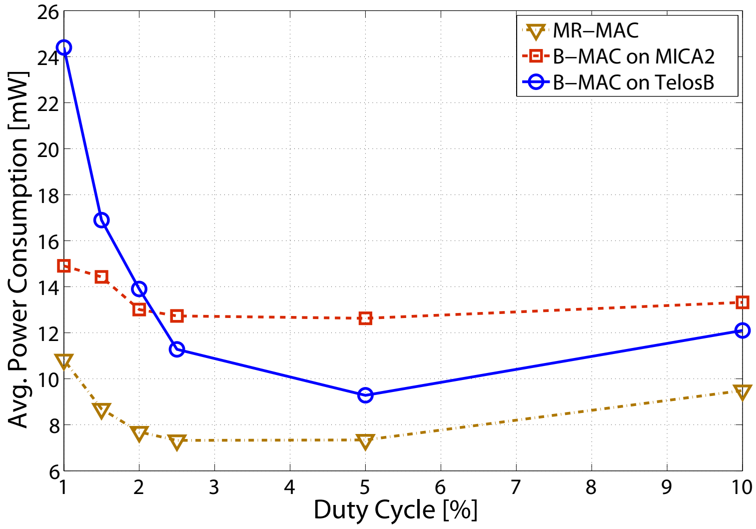
MR-MAC had also been evaluated by varying data sizes at fixed duty cycle values. From Figure 2.14b, we can observe a gradual increase in energy consumption with increasing data size. When the data size is small, data frame preamble is used and the use of high frequency radio is avoided. When the data size is large, bursty radio is used to deliver large data packets very quickly and energy efficiently. The threshold for switching data-frame preamble to micro-frame preamble is empirically found out to be 20 bytes.

Comparison to B-MAC

In order to study the behaviour of our dual radio MAC protocol more effectively, we have carried out the energy consumption comparison of MR-MAC with B-MAC protocol using both the radios as used by MR-MAC. Comparative results against the widely used B-MAC protocol as a reference will help the low-power wireless networking community to understand the energy conservation gains of the MR-MAC design in a better way. The experimental performance comparison was carried out from two aspects: by analyzing the power consumption at different data sizes while keeping the duty cycle constant, and vice versa. MR-MAC by default uses the maximum supported baud rate of 76.8 kbps on the CC1000 radio. All the experiments for stand-alone performance measurements of MR-MAC were carried out at this rate. For comparison with B-MAC, we have lowered down the baud rate of CC1000 chip on our prototype platform to 19.2 kbps to be consistent with that of B-MAC implementation on MICA2 for a fair comparison. Figure 2.15a and Figure 2.15b show the power consumption at transmitters with varying duty cycles and data sizes respectively, for broadcast transmission. Since the preamble length of a broadcast transmission cannot be shortened, no improvement is achieved by preamble optimization methods in MR-MAC. The receiver power consumption of a broadcast transmission is the same as the receiver without unicast optimization. Unlike MR-MAC, which is capable of transmitting a train of data frames with only a single preamble reservation, B-MAC can



(a)

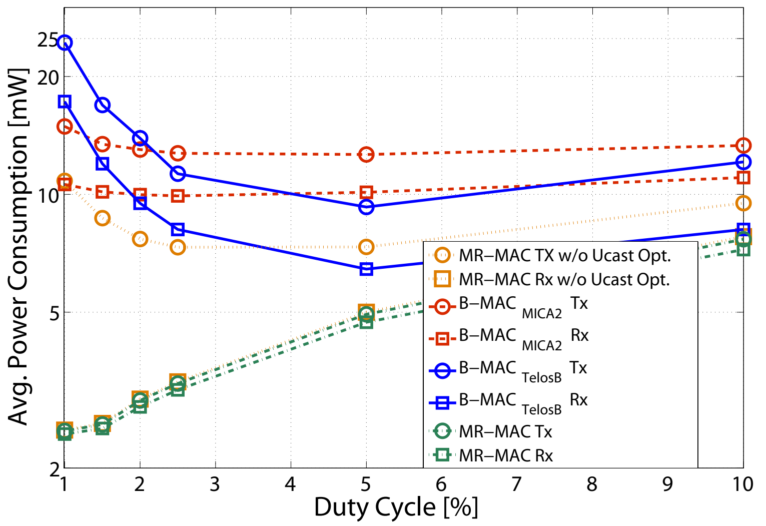


(b)

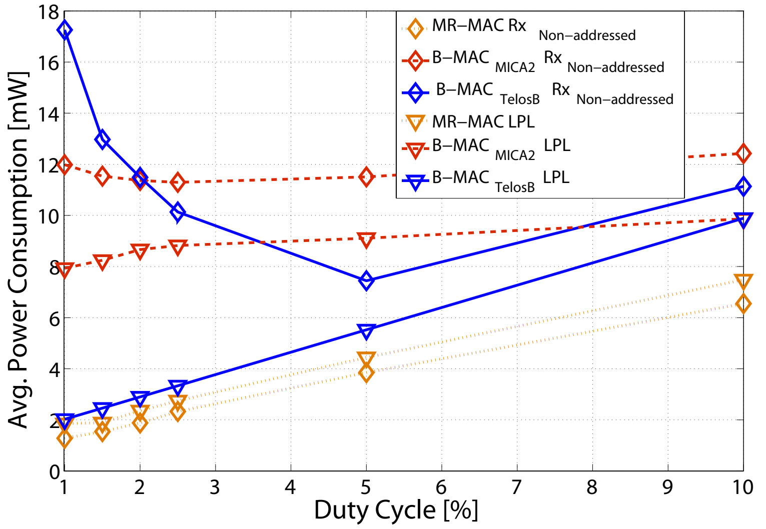
Figure 2.15: Power consumption comparison of transmitters at different data sizes using MR-MAC, B-MAC on MICA2 and B-MAC on TelosB. (a) The transmission rate is set to be 1 Hz, the considered traffic pattern is broadcast and the duty cycle value is set to be 1 %. (b) The transmission rate is set to be 1 Hz, the traffic pattern is considered to be broadcast and data size is fixed to be 100 bytes.

transmit only one packet each time it seizes the channel. This accounts for a bigger difference between B-MAC and MR-MAC when the data size increases. The maximum packet size of the B-MAC is limited to 255 bytes for MICA2 platform and 122 bytes for TelosB platform. In Figure 2.15a, we can observe that at smaller data sizes, the offset between the power consumption of B-MAC and MR-MAC remains almost constant. The power consumption of B-MAC on MICA2 shoots up when data size goes above 250 bytes while for B-MAC on TelosB a sharp rise is observed for data sizes greater than 100 bytes. Although the maximum data size used in this experiment was 1000 bytes, MR-MAC can support up to 4095 bytes with a single preamble reservation. When the data size is smaller than 100 bytes, MR-MAC still outperforms B-MAC mainly due to the energy saved by using low frequency sniffer radio for preamble transmission (as compared to B-MAC on TelosB). It uses data frame preamble only when data size is small and high data rate burst radio when data size is large. While analyzing performance at various duty cycles, the data size was kept constant to be 100 bytes as shown in Figure 2.15b. At this data size, B-MAC does not require separate packet transmissions. The general behaviour of MR-MAC is similar to B-MAC as both are preamble sampling MAC protocols. The power consumption decreases initially with increasing of duty cycle since the required preamble becomes shorter. When the duty cycle goes above 5 %, the energy spent in channel polling dominates over the energy spent in preamble transmission and the overall power consumption starts increasing for this traffic.

Figure 2.16a shows the performance of MR-MAC and B-MAC in the case of unicast transmission at different duty cycles. Comparing to Figure 2.15b, there is no observable difference in terms of power consumption for B-MAC. However, there is a significant amount of power savings for MR-MAC especially at low duty cycle due to preamble optimization techniques. We can observe that as the duty cycle increases, the power consumption difference between MR-MAC and B-MAC on TelosB gets smaller. Since the preamble length decreases as duty cycle increases, the difference between non-optimized preamble length and optimized preamble length decreases. As the preamble length approaches to one preamble frame size, the advantages brought by preamble optimization techniques become less significant at the transmitter as well as at the receiver. From Figure 2.16b, we can observe that at lower duty cycle values, the power consumption of B-MAC on MICA2 is larger than that of MR-MAC and B-MAC on TelosB in low power listening mode. Since CC1000 radio chip is used by both MICA2 and our prototype platform for channel polling operation, the power consumption by the radio chip should be the same for these two platforms. However, a small difference is observed because of different microcontrollers on the two platforms. Atmel ATmega128L [72] on MICA2 is not as energy efficient as Texas Instruments MSP430 [70] on TelosB and our dual radio platform. Although CC2420 radio offers more energy efficient switching among different radio states, it consumes higher energy while sniffing the channel. Although at low duty cycle, the power consumption difference between B-MAC on TelosB and MR-MAC is insignificant, the difference accumulates as duty cycle increases, i.e., the rate of clear channel assessment activity increases. The non-addressed nodes using B-MAC on both the platforms suffer from



(a)



(b)

Figure 2.16: (a) Power consumption of transmitters and receivers using MR-MAC, B-MAC on MICA2 and B-MAC on TelosB at different duty cycles. The transmission rate is fixed to be 1 Hz, the considered traffic pattern is unicast and the data size is fixed to be 100 bytes. (b) Power consumption of non-addressed receivers and low power listening nodes using MR-MAC, B-MAC on MICA2 and B-MAC on TelosB at different duty cycles. The data size is fixed to be 100 bytes.

receiving meaningless preambles and data packets. For longer preamble, more energy is wasted at the non-addressed receiver.

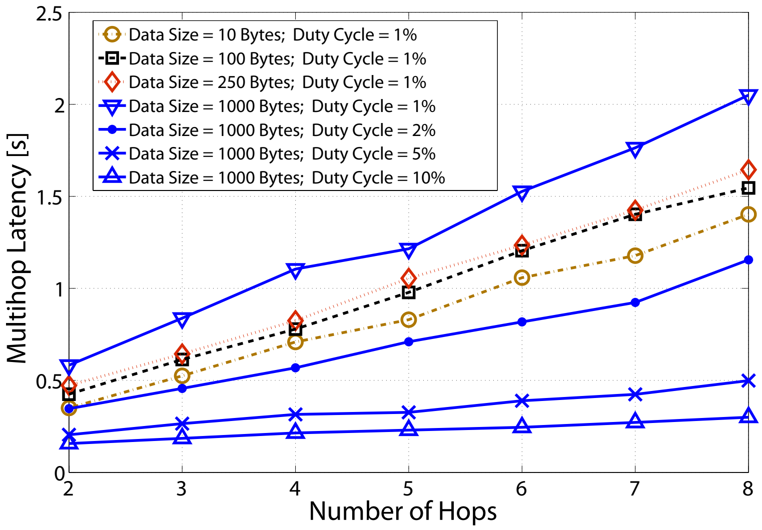
Multihop Latency

Latency can be defined as the interval between putting a data packet into the waiting queue at a transmitter and the successful reception of the packet at the receiver. It is an important metric in evaluating the performance of the low-power MAC protocols because in majority of the applications, data has its shelf life. If not delivered on time, data loses its significance. Event detection or object tracking applications pose strict deadlines on data delivery while many of the environment monitoring applications can afford loose deadlines. Low-power MAC protocols have an inherent latency because of the duty cycle operation and in multihop communication, the end-to-end latency depends upon the number of hops a packet requires while reaching from the source to the destination.

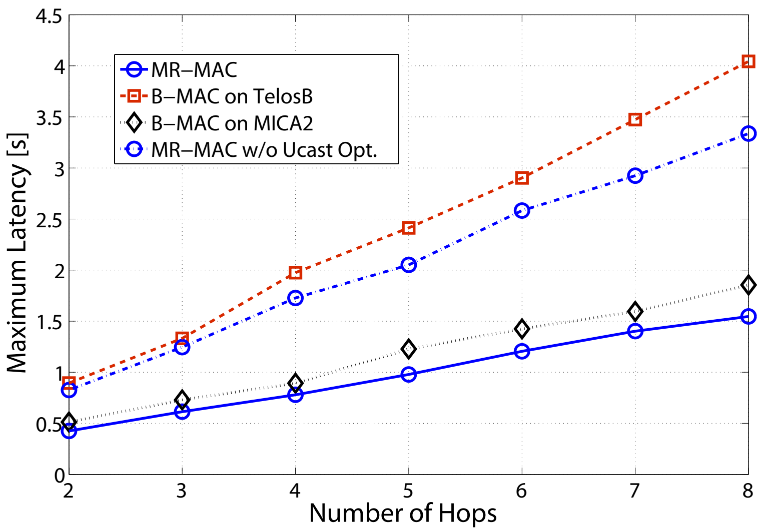
Several optimization techniques can be applied to duty cycle based MAC protocols in this regard. Data aggregation at nodes reduces the waiting time of packets in the transmission queue when all the packets in the queue can be transmitted back-to-back after a transmitter seizes the channel. With prior knowledge on the network topology, wake-up schedules of the nodes can be adjusted to optimize latency performance [73, 74]. Adaptive duty cycles can help in achieving a balance between energy consumption and latency requirement by adopting lowest possible duty cycle which satisfies the current latency requirement.

We have measured the latency of MR-MAC over multiple hops at different duty cycles and data sizes. An n-hop circular network consisting of n nodes was used. Data was sent from one particular node and routed around a circle back to this node. When a node received a packet, it immediately forwarded it to the next node. The latency was measured by running a timer at the node which initiates the traffic. The multihop latency was taken as the time interval between the starting point of transmitting a packet (including preamble) and the end of the reception of the packet. The rate of packet generation was kept so low that no packets were queued at the transmitters. Furthermore, the medium was ensured to be free of congestion and external interference.

From the measurements shown in Figure 2.17a, we can observe that latency increases with decreasing duty cycle values since the average waiting time of transmitters for their receivers to wake-up increases with decreasing duty cycle. Latency is also affected by packet sizes since bigger packets require longer transmission time. Furthermore, assuming the same offset in the wake-up schedule between nodes in all the experiments, longer transmission results in a higher possibility that a forwarding node, having to receive the packet during the wake-up period of its destination node, delays the transmission till the next wake-up period of the destination node. For example, node C wakes up in 30 ms after node B wakes up. For one transmission which requires 10 ms, node B can start transmission to node C in 20 ms after packet reception. However, if the packet transmission requires more than 30 ms, node B needs to wait for the next wake-up cycle of node C to start transmission. This delay increases



(a)



(b)

Figure 2.17: (a) Average end-to-end multihop latency of MR-MAC with various data sizes at different duty cycles. (b) Comparison of the end-to-end multihop latency between MR-MAC and B-MAC.

the latency by a period length. Therefore, the offset between lines in Figure 2.17a with the same duty cycle and different data sizes is not constant. It includes the difference in data transmission time, as well as the extra latency introduced by the difference in data transmission time. Experiments were carried out in order to compare the latency of MR-MAC over multihop links with B-MAC running on TelosB as well as on MICA2. Figure 2.17b shows the latency of MR-MAC, B-MAC on TelosB and

MICA2 at an operating duty cycle of 1 %. Since receivers' duty cycles were kept the same for all the three cases, channel polling rate determines the wake-up period and thus the offered latency. The packet size in this experiment was set to be 100 bytes so that only one data packet transmission is required. From the figure, it is shown that B-MAC on MICA2 has a lower latency than B-MAC on TelosB because it has a shorter clear-channel assessment duration. The channel polling duration of MR-MAC is longer than B-MAC implementation on MICA2 since it has to cover the gap while waiting for acknowledgement between subsequent preamble frames. Although MR-MAC uses a shorter time to transmit 100 bytes than B-MAC on MICA2, the latency at low duty cycling MAC schemes is still dominated by the channel check interval. The graph for MR-MAC without unicast preamble optimization is shown here to support the above observations. MR-MAC with preamble optimization outperforms B-MAC due to its preamble strobing feature. In B-MAC, data transmission takes place after the entire preamble is transmitted. In MR-MAC, upon receiving the acknowledgement of a preamble frame, data transmission can take place immediately afterwards on data channel. Preamble strobing technique reduces the latency to approximately 50 %.

2.5.7 Discussion and Conclusions

We have presented a new MR-MAC protocol for dual radio platforms. The protocol is able to efficiently handle the two main radio activities, namely the idle listening and the ephemeral burst data communication, using a combination of two specialized radio interfaces to result in highly energy efficient manner. The protocol is based on the preamble sampling technique, which implicitly synchronizes the asynchronously waking up nodes only when data communication is required. It does not inflict any extra overhead in order to maintain synchronization among nodes and efficiently handles the network dynamics caused by mobility and appearance/disappearance of nodes. MR-MAC protocol exercises various types of preamble shortening techniques in order to conserve energy. We have implemented the protocol on our prototype platform consisting of a commercially available external radio module interfaced with a low-power wireless node. We have analytically modeled the optimal duty cycle expression for a given amount of traffic load in a network and have shown empirically that our real node testbed implementation adheres to the optimal duty cycle expression. An expression for the optimal ratio of transmit power levels of the two radios on the prototype platform has been derived. An extensive experimental performance evaluation of the protocol in terms of power consumption and latency over different duty cycle values and under various amounts of traffic loads has also been presented. Similar comparative study against the B-MAC protocol implemented on TelosB and MICA2 platforms was carried out. These two single radio platforms have CC2420 and CC1000 radios on-board, which are the two radios used by the MR-MAC prototype hardware platform. The comparison gives an insight in quantifying the advantages of the dual radio MAC approach to a MAC using a single radio. Empirical performance evaluation indicates that MR-MAC protocol clearly outperforms B-MAC in terms of power consumption efficiency and end-to-end latency.

2.6 ENERGY EFFICIENCY USING RADIO TRIGGERED WAKE-UPS

Owing to severe resource constraints, low-power embedded networks are required to use hardware peripherals in a prudent manner. In typical low-power applications, radio resource has the highest energy consumption budget. Since the actual data traffic in some applications is very low, most of the energy is wasted in idle listening to the wireless medium, i.e., while waiting in anticipation for a data packet. Therefore, minimizing idle duration power consumption at a node is certainly very important. Radio duty cycling is a popular solution in which a radio is turned on and off according to a pre-defined scheme, specific to each MAC protocol [2, 3]. The trade-off between energy consumption and latency for data communication in duty cycling MAC protocols is seen as the biggest hurdle in achieving extremely low-power operation when a quick response time is required upon an event of interest.

We have designed a radio triggering scheme, Radio Triggered Wakeup with Addressing Capabilities (RTWAC) for low-power nodes aiming at simultaneously minimizing the idle mode energy consumption and latency in data communication. In this section, we will present the design details of a radio triggered wake-up circuit board and the protocol details. RTWAC allows a wireless node to keep its on-board radio completely switched off until data communication is required. In order to initiate data communication, an out-of-band modulated signal is transmitted as a wake-up signal. An external wake-up circuit board triggers an interrupt to a wireless node. The microcontroller of the node switches from sleep state to active mode upon the interrupt and interprets the data transmitted in the modulated wake-up signal. The data contains address information and command messages, which allows a node to quickly switch back to the sleep state if it is not the destination. On the contrary, an addressed node executes the command message transmitted inside the wake-up signal. For example, a command may require turning on the on-board radio for data communication. The wireless node uses a low-power MAC protocol running on the normal on-board radio for data communication. The MAC protocol efficiently coordinates access to the shared medium among the nodes and gives a high reliability for data communication, while radio triggering allows saving significant amount of energy wasted in idle listening. This way RTWAC advocates the idea of combining the advantages of radio triggered wake-up technique with low-power MAC procedure for reliable data communication and highly energy efficient operation.

2.6.1 Concept and Terminology

RTWAC alleviates the need for periodic channel polling by a wireless node in contrast to duty cycling MAC protocols. The radio remains completely switched-off and the microcontroller operates in the power-down mode. By using an additional hardware circuit attached to a node, the microcontroller can be woken-up from low power mode by sending a specific RF signal. This additional hardware circuit is either completely passive or active with an extremely low power consumption. A wireless node can operate for several years in the sleep mode with this low power consumption and at the same time, is able to instantaneously wake-up upon the need for data communication over its standard on-board radio. Before sending a wake-up signal, clear channel assessment is performed, which helps in avoiding potential collisions. Using a modulated RF signal to wake-up nodes, unlike the approach by Gu and Stankovic [75], allows uniquely addressing a node or a group of nodes, and sending data (e.g., command messages) in the wake-up signal.

2.6.2 Design Objectives and Opportunities

We aim at the following goals in combining radio triggered wake-ups with duty cycling MAC protocols:

- Suppressing idle listening power consumption significantly by avoiding radio duty cycling at wireless nodes. This is achieved by keeping the on-board radio completely switched off and putting the microcontroller in a low-power mode.
- Consuming a negligible power in the sleep/dormant mode at a node.
- Instantly waking-up nodes from the sleep state, i.e., soon after a node receives the wake-up signal, it should be ready to communicate over its normal on-board radio.
- Ability to address nodes uniquely, i.e., possibility to wake-up a single node or group of nodes. The wake-up address is the same as the MAC address of a node.
- Capability of sending simple command messages to a node within the wake-up signal for performing different tasks.
- Observing to frequency usage regulations in Europe.
- Keeping cost of the wake-up circuit board as low as possible.
- Causing radio triggered wake-ups with high reliability and low false triggers.
- Maximizing the operating range of the wake-up hardware.

One of the challenging engineering tasks is to achieve a long communication range over radio triggered wake ups and simultaneously keep a negligible power consumption. With a completely passive circuit design approach, communication range of more than 1 m is difficult to achieve under the frequency regulations in Europe. By introducing some active components in the wake-up hardware circuit, it is possible to increase the operating range to more than 10 m and still keep power consumption of the wake-up circuit at an extremely low level.

2.6.3 Performance Inefficiency of Existing Approaches

Experimental studies on sensor network deployments across various applications have shown that data communication occupies the highest energy consumption budget [24]. While it is easy to determine when data transmission is required, reception is usually unpredictable and uncertain to a node. Receivers therefore use either continuous listening to the wireless medium or duty cycling scheme. An evident drawback of duty cycling schemes is the increased latency compared to always on mode. On the other hand, when a radio is always on in anticipation for a data packet, a high amount of energy is wasted. One of the possible alternatives to minimize latency and energy consumption is through an additional wake-up radio hardware, which is optimized for negligible power consumption and is capable of instantly reacting upon an event of interest.

PicoRadio [76,77] from UC Berkeley is a carefully designed very low power transceiver module (as a prototype IC), which is capable of monitoring radio environment. It can be used either as a stand-alone radio module on a low-power wireless node or as an additional wake-up module in combination to a more advanced radio transceiver. The total power consumption of the module in the receive mode is 380 μW at a supply voltage of 1 V and has a receiver sensitivity level of -75 dBm. In the transmit mode, it consumes 1.6 mW with an output power level of 0 dBm. Although these power consumption ratings are much lower than many state-of-the-art radio transceivers, which typically operate at 2-3 V of supply voltage and consume a current of 10-30 mA, this solution has still significantly high power consumption in the always-on mode. Furthermore, no working prototype is evaluated for performance characteristics.

Miller and Vaidya [78] proposed a MAC protocol using two radios. The primary radio sends a wake-up signal to turn on the nodes. After waking up, nodes communicate in the secondary channel. This scheme has attractive energy saving possibilities, e.g., when compared to STEM [56], where the primary radio sends a busy tone signal and the nodes are required to listen to the primary channel all the time in anticipation for an upcoming signal. The authors devised a scheduling based method for waking-up nodes depending upon the traffic loads. Simulation studies using ns-2 have been carried out to determine the optimal value of node wake-up periods leading to minimum energy consumption for a given number of packet transmission and reception.

Gu *et al.* proposed using an external circuit for radio triggered wake-ups attached to low-power wireless nodes [75]. The idea is to use only passive components in order to collect energy from ongoing radio transmissions, similar to the principle of passive RFID technology. When the power induced at the receiving antenna is large enough, it interrupts the microcontroller, which wakes-up the "normal" wireless node radio for data communication. Due to simplicity of the wake-up hardware, it reacts to any strong electromagnetic field in the operating frequency band. Having no addressing mechanism in the design leads to undesired wake-up of nodes. In order to avoid the unwanted node wake-ups, Radio-Triggered-ID (RTID) is proposed, where different nodes are addressed by performing simultaneous transmissions at several different frequencies. This solution involves practical shortcomings such as the need for an additional wake-up hardware corresponding to each frequency and a transmitter capable of transmitting at different frequencies simultaneously. Furthermore, the approach used in RTID has obviously a very limited addressing space.

Malinowski *et al.* developed a direct amplifying RF detector, operating at 300 MHz as a part of the CargoNet project [79]. The main building blocks of this RF detector include an antenna matching network, an envelope detector and a micro-power amplifier. The receiver sensitivity and power consumption of the circuit are -65 dBm and 2.8 μW, respectively. The RF detector is able to detect an OOK signal modulated with baseband square pulses of 25 Hz.

WISP is a wireless powered platform for sensing and computation [80]. Its main building blocks are passive power harvesting hardware, demodulator circuitry, microcontroller and sensors. Although WISP is not directly related to WSNs, its hardware components and the research philosophy are close to our work. WISP is a battery-free platform for sensing and computation. A standard UHF-RFID reader is used to power it over wireless and control its sensing and data transmission capabilities. Main building blocks of WISP are passive power harvesting hardware, demodulator, microcontroller unit and sensors.

The work by van der Doorn *et al.* [81] aims at achieving energy efficiency through a low cost radio triggered wake-up circuit attached to a sensor node platform. Similar to the approach proposed earlier by Gu *et al.* [75], a sensor node's on-board radio is used to transmit wake-up signals. van der Doorn *et al.* use Texas Instruments CC1000 on SOWNet Technologies T-node platform to transmit wake-up signals for triggering the wake-up circuit attached to other nodes. Their prototype works in 868 MHz frequency band, and achieves a range of only 2 m with 3 mW of transmit power. Having no addressing mechanism, radio signals cause undesired node wake-ups, which can cause considerable energy loss especially in high density networks. The overall dormant/sleep state power consumption of the prototype varies from 171 mW to 819 mW, which is significantly higher as compared to our prototype (cf. Section 2.6.7).

2.6.4 Choice of the Operating Frequency

We have investigated the possible operating frequency bands that best suit our needs according to the application requirements and the design goals as stated earlier. In this subsection, we present the rationale behind the frequency band selection. The main factors that strongly influence the choice are:

- Communication range
- Size of the antenna and complexity of the hardware circuit
- Availability of the appropriate electronic components
- Possibility to operate license free

We have evaluated and compared the use of the following ISM frequency bands: 13.56 MHz, 433 MHz, 868 MHz, 2.4 GHz and an RFID specific frequency range of 100-135 kHz. Table 2.9 summarizes the comparison of these frequency bands based on our criteria.

Frequency range	*Communication distance*	*Antenna size*	*Hardware circuit size*	*Components availability*
100-135 kHz	-	-	+ +	+ +
13.56 MHz	-	-	+	+
433 MHz	+	-	+	+
868 MHz	+	+ -	+	+
2.4 GHz	+ -	+	+	+ -

Table 2.9: Suitability of different frequency bands for radio-triggered wake-ups.

For 100-135 kHz frequency range, Atmel U3280M [82] transponder interface for microcontroller perfectly suits our needs. It is completely powered from an external magnetic field and requires no biasing power supply. It is able to interrupt a microcontroller, when the transponder comes in the vicinity of an alternating magnetic field. Unfortunately, a large antenna size requirement and only a small resulting communication range because of the inductive coupling makes the use of this frequency unfavorable. Devices operating at 13.56 MHz also rely on inductive coupling and therefore, suffer from having a limited range of approximately 1 m.

In 433 MHz, 869 MHz and 2.4 GHz frequency bands, the principle of operation is similar. According to the Friis transmission formula, the propagation loss increases for higher frequencies, thereby making 433 MHz frequency band a preferable choice. However, since the antenna size is proportional to the wavelength, 2.4 GHz frequency band becomes more attractive option with smaller antenna size. On the other hand, the non-availability of discrete circuit components at 2.4 GHz frequency makes the choice difficult. Overall, 868 MHz frequency band turns out to be the most suitable design choice for our needs with having a reasonable antenna size and operating range, and easily available circuit components. According to the frequency regulations in Europe, an Effective Radiation Power (ERP) of 500 mW is permissible for this frequency with a maximum duty cycle of 10 %. We have designed and implemented our prototype working at 868.5 MHz to be compliant to European frequency regulations. However, the same selection criteria holds for the available 902 to 928 MHz frequency band in Americas and other parts of the world.

2.6.5 Antenna Considerations

In the reception mode, the primary task of an antenna is to transfer the maximum amount of power from air to the wake-up circuit. In order to keep the price of the prototype circuit as low as possible, we chose a half-wavelength dipole antenna made of a copper wire. The radiation pattern of an ideal half-wavelength dipole is omnidirectional in the plane perpendicular to the antenna. The input impedance of the antenna is 73Ω at the resonant frequency while the gain is 1.64. Balanis [83] states that for a given wavelength λ, the length of the dipole for the first resonance (imaginary part of the input impedance becomes zero) is approximately between 0.47λ and 0.48λ depending upon the radius of the wire. Thinner the diameter of the wire is, closer the length becomes to 0.48λ. We constructed an antenna using a 1 mm thick copper wire and used a coefficient of 0.475 for the antenna length calculations. The wavelength of an 868.5 MHz frequency wave is, $\lambda = \frac{C}{f} = \frac{3 \cdot 10^8}{868.5 \cdot 10^6} = 34.5\,\text{cm}$, which gives the antenna length,

$$l = 0.475\lambda = 0.475 \cdot 34.5\,\text{cm} \approx 16.4\,\text{cm}. \tag{2.13}$$

The power induced at the antenna is extremely low because of the path loss. Therefore, maximum possible power transfer from antenna to the rest of the wake-up circuit is required through an appropriate matching network. The matching network in our design uses two lumped reactive elements – a parallel capacitor and a series inductor.

2.6.6 Prototype Hardware

This section describes the design details of RTWAC. By using an additional hardware circuit, the microcontroller on the node can be woken-up from low-power mode by sending a specific RF signal. This additional hardware circuit is not completely passive but consumes an extremely low current (a few μA). A COTS wireless node can operate for several years in the sleep mode with this low power consumption and can wake-up instantaneously upon the need for data communication over the standard onboard radio. Before sending a wake-up signal, clear channel assessment is performed, which helps in avoiding potential collisions. Unlike [75] where no modulated signal is used for the wake-up process, a modulated RF signal is used to wake-up nodes. The "data modulation feature" in our design allows uniquely addressing different nodes or groups of nodes and sending additional data (e.g., command messages) in the wake-up signal.

Wake-up Signal Transmitter

RTWAC prototype wake-up signal transmitter consists of a TelosB node, Texas Instruments CC1000 radio transceiver and optionally a ZHL-2010 [84] radio frequency amplifier from Mini-Circuits®. The block diagram of the wake-up signal transmitter is shown in Figure 2.18. Please refer to Figure A.2 for interfacing connections between TelosB and CC1000 radio module.

The CC1000 radio generates a wake-up signal at 868.5 MHz frequency while a radio frequency amplifier can be used to increase the communication range. We used

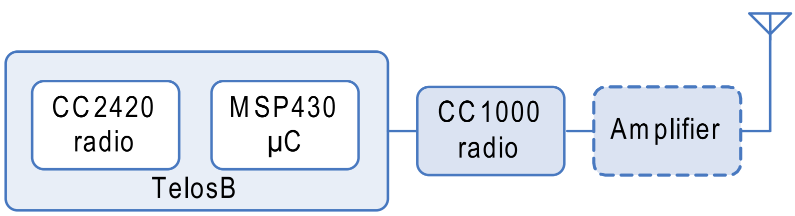


Figure 2.18: Simplified block diagram of wake-up transmitter.

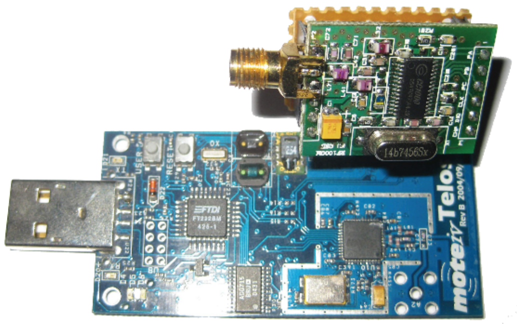


Figure 2.19: Snapshot of an RTWAC transmitter.

Texas Instruments CC1000PPK-868 [85] radio module attached externally to a TelosB node as shown in Figure 2.19. In order to generate an OOK modulated signal through the CC1000 radio, which is primarily an FSK radio, the output power of the transmit amplifier is controlled. The difference in the two frequencies (corresponding to a one and a zero) in the FSK modulator is set to zero. The output power of the amplifier is controlled by sending a command byte over the configuration bus interface of the CC1000 chip. Commands for enabling and disabling the output transmit power on the chip correspondingly lead to ones and zeros in the OOK modulation.

Wake-up Signal Receiver

A receiver consists of a wake-up circuit attached to a TelosB platform. The hardware circuit interrupts the MSP430 series microcontroller on the TelosB node upon receiving a wake-up signal. The block diagram of the receiving wake-up circuit is shown in Figure 2.20. The main building blocks include an impedance matching network, a voltage multiplier, and a digital comparator. The power induced at the antenna is extremely low due to the path loss during the radio waves propagation. Thus a matching network is used to transfer the maximum possible power from the antenna to the voltage multiplier circuit. The matching network consists of two lumped reactive elements: a parallel capacitor and a series inductor. The induced power at the output of the antenna and the matching network is potentially very small to trigger the microcontroller; moreover the voltage alternates at the radio frequency. Therefore, our design

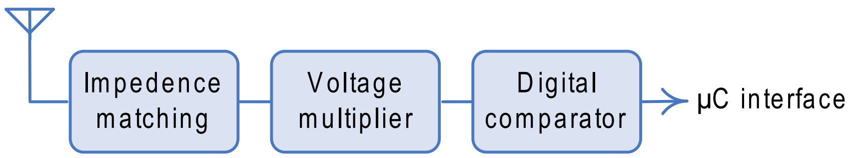


Figure 2.20: Simplified block diagram of wake-up receiver.

Frequency [MHz]	*Voltage Induced* [V]
840	3.4
850	3.9
860	4.3
868.5	4.8
870	4.6
880	3.0
890	2.1

Table 2.10: Frequency sensitivities of the prototyped RTWAC receiver and the average voltage induced at the output of the voltage-multiplier circuit at different center frequencies. The transmit power is set to be 20 dBm and the distance between the transmitter and the RTWAC receiver is fixed to 15 cm.

includes a five stage Voltage Multiplier (VM) structure (also known as charge pump) in order to increase the induced voltage to a sufficiently high level for detecting the slowly varying envelope signal from the modulated high frequency carrier. The VM consists of a combination of capacitors and diodes. The diodes are chosen carefully to be able to turn on at very low forwarding voltages and operate at high switching frequencies. Low threshold RF Schottky diodes (HSMS-2852 [86]) from Avago Technologies are used in our prototype.

In order to observe the frequency response of our prototype receiver and its tuning sensitivity levels at different nearby frequencies, we used a signal generator to generate signals with a constant transmit power of 20 dBm. The wake-up receiver circuit was placed at a fixed distance of 15 cm from the signal generator. One should note that the transmit power level and the distance used in particular was not important for studying frequency sensitivities. We measured the voltage induced at the output of the 5-stage voltage multiplier circuit at different frequencies. Table 2.10 shows the mean voltage induced at the 5-stage voltage multiplier circuit when the transmitter uses 20 dBm of output power and the RTWAC receiving node is at a distance of 15 cm. It can be observed that the circuit is very well tuned to 868.5 MHz. However, it has a wider sensitivity bandwidth, which makes it susceptible to out-of-band interference from GSM-900. Therefore, a digital comparator based threshold selector is used in our design to lessen the effect of interference from the adjacent GSM band. Digital comparator is the only active component in our wake-up circuit. It is used for digitizing the analog signal and shifting the voltage levels to “high” and “low” logical levels of the microcontroller. Digital comparator also performs an over-voltage protection for the microcontroller because it is likely that voltage multiplier in the close vicinity to the transmitter can produce voltage levels as high as 10 V. The complete schematic of the wake-up circuit is shown in Figure 2.21.

The resistor R1 turns out to be the main load for the VM in the circuit since the load at the digital comparator input pin is negligible. R1 constantly drains out current

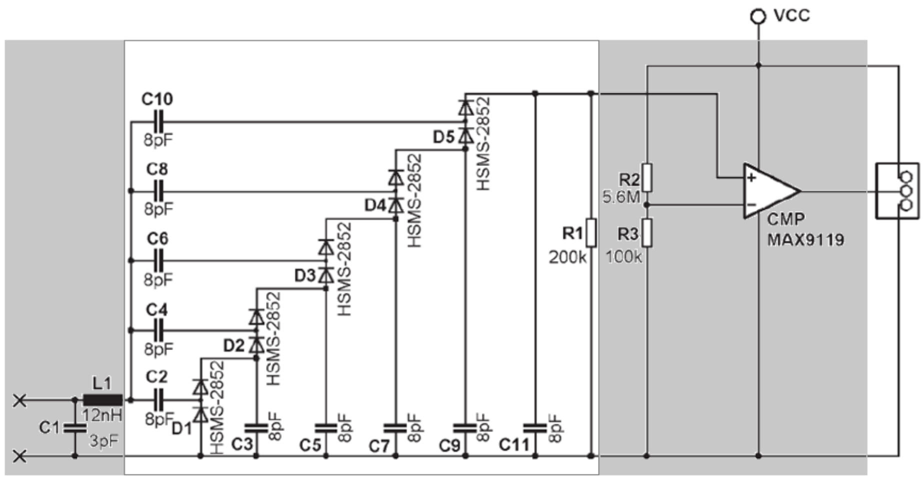


Figure 2.21: Schematic of the developed wake-up board.

from the charging capacitors C2-C11. When the power induced by the antenna is decreased, the voltage at the output of the VM is also decreased. Thus the VM and the load resistor R1 form a simple envelope detector. The obtained amplitude envelope is compared to a predefined threshold of a digital comparator in order to detect the transmission of a high or a low level. The predefined threshold level is configured by the voltage divider consisting of resistors R2 and R3. The selected threshold level is higher than the noise level in order to avoid false interrupts at the microcontroller. Decreasing the threshold level can lead to a higher operating range, however this also causes an increasing number of false positives. We empirically found out that a noise threshold level of 50-60 mV gives an operating range of more than 10 m using the permitted transmit power level and duty cycle constraint of 27 dBm and 10 % duty cycle, respectively. With this empirical threshold value, no false alarm was observed. Figure 2.22 shows a snapshot of our prototype RTWAC receiving node consisting of a radio triggered wake-up circuit attached to a TelosB node. One possible design enhancement includes using a band pass filter after the antenna matching network in order to suppress adjacent channel interference. This allows lowering the noise/interference level threshold value and hence leads to a larger operating range. For this purpose, we suggest using a SAW (Surface Acoustic Wave) filter. Golledge Electronics 868.60 MHz and EPCOS 868.30 MHz are preferable choices as these do not require any biasing voltage, have low band-pass filtering loss and have high attenuation for undesired interference from adjacent GSM band.

There are only two sources of power consumption in the wake-up circuit. The first is an extremely low-power MAX9119 digital comparator [87], which typically consumes only 350 nA of supply current at a bias voltage of 3 V. The second source of power consumption is the voltage divider that forms the threshold voltage (reference voltage for the comparator). Voltage divider is composed of two resistors, R2 and R3,

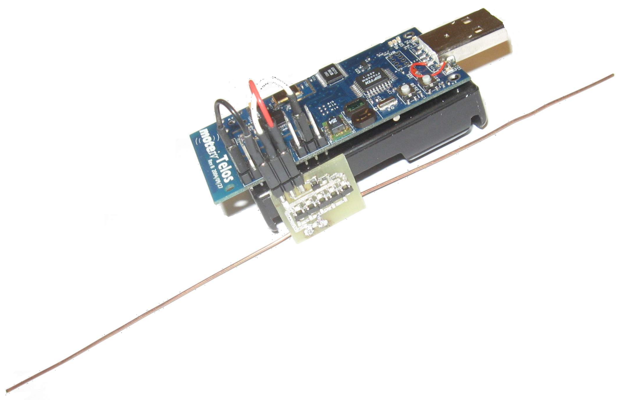


Figure 2.22: Snapshot of an RTWAC receiver node consisting of an external wake-up circuit connected to a TelosB platform.

and consumes a current of 526 nA. The total current drained by the wake-up circuit is,

$$I_{\text{wakeup}} = I_{\text{comp}} + I_{\text{div}} = 350\text{nA} + 526\text{nA} = 876\text{nA}\,. \tag{2.14}$$

Since the external wake-up circuit causes an interrupt to the microcontroller, the deepest sleep mode (LPM4) of the MSP430 series microcontroller is used, which consumes a current $I_{\text{TelosB}} = 3.3\,\mu\text{A}$ on TelosB.

Wake-up Signaling Protocol

We have designed a simple and light-weight protocol for wake-up signaling. The CC1000 radio chip is used to generate an OOK modulated signal through simply turning on and off its power amplifier. These turn-on and turn-off periods are controlled by the on-board MSP430 microcontroller on TelosB. For encoding of digital data, we chose Pulse Interval Encoding (PIE). Figure 2.23 shows the PIE scheme used in our prototype. Encoding a "0" and a "1" starts with a fixed interval T of high level transmission, followed by a low level period equal to T for a "1" and $2T$ for a "0". The average bit transmission time, therefore varies from $2T$ to $3T$.

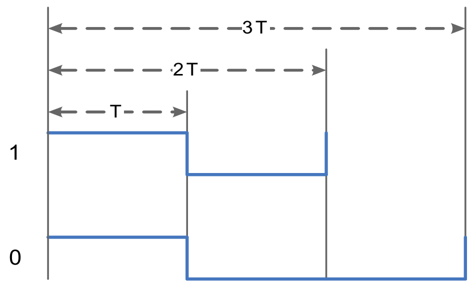


Figure 2.23: Pulse Interval Encoding (PIE).

Use of PIE instead of Manchester encoding in our implementation allows us to handle a lower number of interrupts at the microcontroller while decoding a single bit. In order to decode data sequence using Manchester encoded signal, the microcontroller has to track all the transitions from low-to-high and from high-to-low. In PIE, it is enough for the microcontroller to track only the low-to-high transitions and the time intervals between them in order to successfully decode a particular data sequence. A reduced number of interrupts saves the power consumption required in invoking and processing additional interrupt service routines. A synchronization sequence of zeros and ones is sent at the beginning of each packet. This sequence also allows dynamic calculation of the timing characteristics for the transmission of zeros and ones, i.e., determining the time duration $2T$ for a "1" and time duration $3T$ for a "0". The receiver, thus, has no hard built-in timing values, rather it can adapt to the timing characteristics of a transmitter based on the synchronization sequence. At the link layer, we have designed the following packet structure for the wake-up signal transmission. A packet starts with an 8 bits synchronization sequence, followed by a 16 bits address, a 16 bits command and an 8 bits CRC. The total packet size is therefore six bytes. It should be noted that the address space in a wake-up packet is shared with the MAC addressing scheme used by the on-board radio used in data communication. The packet structure is shown in Figure 2.24.

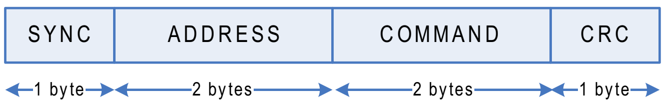


Figure 2.24: Wake-up packet structure.

In order to avoid any potential collisions before attempting to send a wake-up packet, RTWAC transmitter makes sure that the medium is free by performing clear channel assessment. This is done by polling the received signal strength indicator pin of the CC1000 radio chip. Figure 2.25 shows the basic functionality of an RT-WAC receiving node. A wireless node stays in the sleep state with the on-board radio completely switched off and MSP430 series microcontroller working in an extremely low-power mode (LPM4). When an external radio signal is detected at 868.5 MHz, the microcontroller is triggered on and tries to decode the radio message. If a synchronization sequence is not detected within a user-defined pre-selected time duration, the node times-out and switches back to the sleep state. Based on the synchronization sequence, the node estimates the decoding timings for PIE message and later on decodes the message. In a case when CRC fails or the wake-up packet is not addressed to a node (i.e., the MAC address of the node is different from the address field in the wake-up packet), it switches back to the sleep state. Otherwise, the node executes the command message. The radio chip on a wireless node is turned on only when data communication is required. Immediately after performing the indicated task, a node switches back to the sleep state. Data communication is carried out using a CSMA/CA based MAC

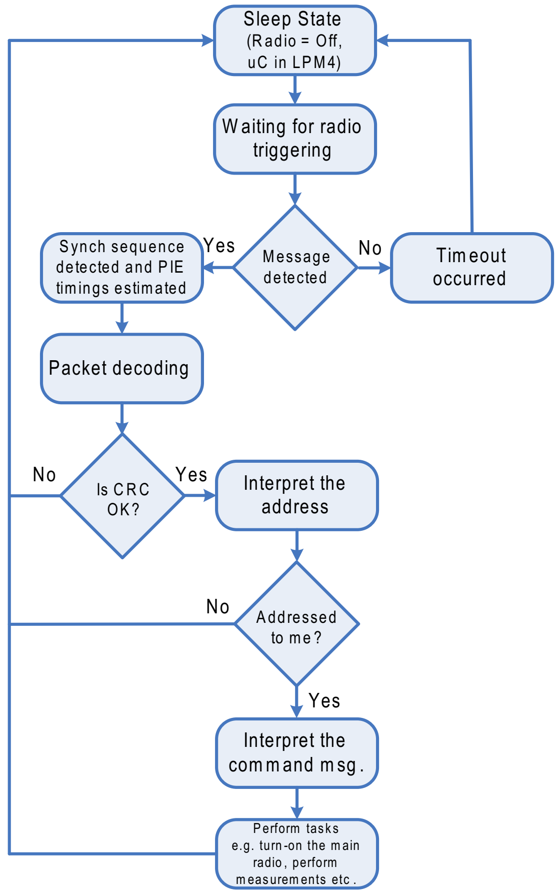


Figure 2.25: Flow chart describing RTWAC functionality.

protocol implemented on the on-board radio. CSMA/CA based MAC protocol implemented on a faster on-board radio of the wireless node has much higher efficiency and reliability for data communication than wake-up radio channel which does not use any medium sharing mechanisms.

2.6.7 Performance Evaluation

In this section, we will present our results on the range analysis of RTWAC and its empirical comparison with duty cycling MAC protocols.

Range Analysis

We can roughly estimate the received power, P_{rx} induced at the antenna using the Friis' equation [83]. The peak voltage at the antenna is calculated to be

$$V_{ant} = \sqrt{2P_{rx}R_{ant}} \quad , \tag{2.15}$$

where R_{ant} is the antenna resistance. An equivalent circuit of the wake-up hardware is used to calculate the voltage delivered from the antenna to the input of the voltage multiplier and the digital comparator. The equivalent circuit is shown in Figure 2.26. It consists of the antenna resistance R_{ant}, the matching network reactance X_{match}, the VM resistance R_{vm} and the VM reactance X_{vm}. We define impedance of the antenna and matching network as:

$$Z_{\text{ant}} = R_{\text{ant}} + jX_{\text{match}}. \tag{2.16}$$

The impedance of the VM is,

$$Z_{\text{vm}} = R_{\text{vm}} + jX_{\text{vm}}. \tag{2.17}$$

The voltage transferred from antenna and matching network to the voltage multiplier can be calculated using,

$$V_{\text{vm}} = V_{\text{ant}} \left| \frac{Z_{\text{ant}}}{Z_{\text{ant}} + Z_{\text{vm}}} \right|. \tag{2.18}$$

When the matching network is tuned for the maximum power transfer, we have,

$$\begin{aligned} Z_{\text{ant}} &= Z_{\text{vm}}*, \\ R_{\text{ant}} &= R_{\text{vm}}, \\ X_{\text{match}} &= -X_{\text{vm}}. \end{aligned} \tag{2.19}$$

For this case, the voltage level at the VM input can be expressed as,

$$V_{\text{vm}} = V_{\text{ant}} \left| \frac{Z_{\text{vm}}}{2R_{\text{vm}}} \right| = \frac{V_{\text{ant}}}{2}\sqrt{1+Q^2}, \tag{2.20}$$

where Q is quality factor and is given by,

$$Q = \frac{X_{\text{vm}}}{R_{\text{vm}}}. \tag{2.21}$$

For our voltage multiplier, Q is less than one and $Q^2 << 1$, thus we can neglect it and approximate the input voltage for VM as,

$$V_{\text{vm}} = \frac{V_{\text{ant}}}{2}. \tag{2.22}$$

Theoretically, a five-stage voltage multiplier gives ten times higher voltage level than that at the output of an antenna matching circuit. But in reality, the output of the voltage multiplier is reduced due to forward bias voltage of the diodes, parasitic capacitances and constantly draining load current. At the very low input voltages the forward bias voltage loss becomes comparable to the applied voltage, thereby decreasing the efficiency significantly. We used SPICE [88] simulation tool to estimate the multiplication factor at different input voltage levels. Based on the simulation results, a multiplication factor of five is chosen. Thus the voltage at the input of the digital comparator is, $V_{\text{comp}} = 5V_{\text{vm}}$. As we have mentioned earlier, the sufficient voltage at

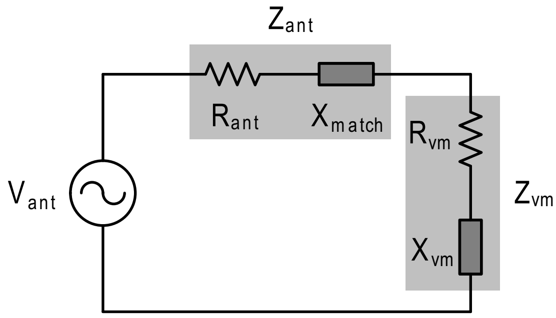


Figure 2.26: Equivalent circuit of the RTWAC receiver.

the input of the comparator is $V_{\text{comp}} = 50\,\text{mV}$, thus we can calculate the required power delivered from antenna, using equations, (2.15), (2.16) and (2.17). The minimum required receive power turns out to be, $P_{\text{rx}} = 2.74\,\text{mW} = 4.37\,\text{dBm}$. For 500 mW of transmit power, the operating range using Friis transmission formula is calculated to be 15 m. Naturally, the range is smaller in realistic conditions especially indoors.

In the above calculations, we assumed a perfect power transfer matching between the antenna and the voltage multiplier and ignore the polarization losses. If we take into account 3 dB polarization loss, when one antenna is circularly polarized while the second antenna is linearly polarized, the operating distance reduces to 10 m. For imperfect matching the operating distance is further reduced by a value depending on the mismatch loss.

We have measured the communication range of RTWAC nodes with and without the amplifier (ZHL-2010). The measurements were carried out in an indoor environment for line-of-sight propagation conditions and were repeated thrice in order to eliminate possible statistical fluctuations. Before determining the communication range, we measured the output power of the RTWAC transmitter node. For a transmitter without an amplifier, the output power was measured to be P_{out} = -1.1 dBm. The RTWAC transmitter uses $\lambda/4$ monopole antenna while the RTWAC receiver uses half-wavelength dipole antenna. The RTWAC transmitter with an amplifier (ZHL-2010) has an output power of $P_{\text{out\_amp}} = 20.56\,\text{dBm}$. In this configuration, it is possible that the signal is significantly attenuated due to cross-polarization effect, even if the distance between the transmitter and the receiver is small. In order to avoid this effect, a circularly polarized antenna can be used either at the transmitter or the receiver.

Based on our measurements, we have identified three main operating zones, where the reception of the RTWAC receiver differs significantly. The first zone is characterized by a consistent reception. Inside this zone, a node receives the wake-up signal regardless of its position and orientation in space. This area is in close proximity to the transmitter, so that polarization losses cannot block the signal reception. In the second zone, an RTWAC node gets triggered in most of the places and orientations in space, but there are some blind spots, where reception is not possible due to significant polarization loss or multipath propagation effects. The third zone is dominated by blind spots and one has to look for the correct position and orientation of an RTWAC node in order to trigger wake-ups. The borders between different zones are quite relative and may differ for different experimental environments. We obtained the following zone boundaries while using an amplifier attached to an RTWAC node:

$$\text{Zone1} = 1.6\,\text{m}, \tag{2.23a}$$
$$\text{Zone2} = 3.0\,\text{m}, \tag{2.23b}$$
$$\text{Zone3} = 7.5\,\text{m}. \tag{2.23c}$$

Here a zone denotes the radius of a circle with RTWAC transmitter in the center. For Zone3, 7.5 m is the most distant point from the transmitter, where RTWAC node is able to receive the wake-up signal. Without using the amplifier, we found the three zones for the transmitter:

$$\text{Zone1} = 5\,\text{cm}, \tag{2.24a}$$
$$\text{Zone2} = 30\,\text{cm}, \tag{2.24b}$$
$$\text{Zone3} = 65\,\text{cm}. \tag{2.24c}$$

The reader may notice that here the zone interpretation is slightly different. The reason is that for this case, cross-polarization effects significantly dominate over multipath effects, so even in a very close proximity there is a block-out in case of cross-polarized orientation of the antennas. Zone1 defines a consistent reception regardless of the orientation of antennas. In Zone2, reception is consistent only when the orientation of the antennas is identical. Zone3 shows the maximum possible distance for reception.

The obtained results are comparable to the analytical calculations as described above. Analytical calculations gives an operating distance of 8.6 m for $P_{\text{out\_amp}} =$ 20.56 dBm. This figure is pretty close to the maximum measured distance of 7.5 m. However, it should be noted that analytical calculations do not take into account antenna mismatch at the receiver and transmitter, and use free space radio waves propagation model. Our experiments indicate that the analytical results can be use as a rough estimation for the maximum operating distance.

We generated ASK modulated RTWAC packets and fed it to the ZHL-2010 amplifier using Agilent E4438C Vector Signal Generator. We set the amplitude of the waveform from the signal generator so that the ERP out of the ZHL-2010 amplifier becomes equal to the maximum allowed power of 27 dBm at the 868.5 MHz frequency in Europe. We obtained the following ranges for the three zones:

$$\text{Zone1} = 2.3\,\text{m}, \tag{2.25a}$$
$$\text{Zone2} = 3.9\,\text{m}, \tag{2.25b}$$
$$\text{Zone3} = 10.1\,\text{m}. \tag{2.25c}$$

We have also counted the number of packets received with errors. An 8-bit CRC is included in the wake-up packet, thus we can calculate the percentage of packets received with CRC failures. A maximum CRC failure of 10 % was observed for all the received packets regardless of the distance and zone. Even in Zone3, the mean packet receive error ratio did not exceed over 10 %. CRC failures and triggering of false alarms result in energy wastage. False alarms are caused by external interferers. The CRC failures occur when an RTWAC receiver fails to correctly receive a data packet after detecting the synchronization pattern. Energy is also wasted when an RTWAC receiver receives a packet, which is not addressed to it. However, since the packet transmission lasts only for a short time and correspondingly involves switching of the microcontroller from LPM4 mode to the active mode and back, the amount of energy spent is very low.

Comparison with Duty Cycling MAC Protocols

The traffic volume is typically very small in majority of the low-power applications. Implicit synchronization of nodes in preamble sampling MAC protocols has the advantage of not wasting energy in maintaining common sleep schedules of the nodes unlike other medium access approaches [2, 3]. Therefore, preamble sampling protocols become a logical choice for very low data rate applications. We have selected B-MAC [36] as a representative preamble sampling MAC protocol for a comparison with RTWAC. We have analyzed the power consumption and offered latency of the RTWAC wake-up circuit board attached to a TelosB in comparison to B-MAC [36] operating at different duty cycles on a TelosB platform.

The power consumption of RTWAC receiver was calculated in the sleep mode whereas that of B-MAC is calculated while performing only low power listening (LPL) or channel polling operation without packet receptions. This is justified because it represents the most common (idle) state of a node. In the sleep mode, the current drawn by an RTWAC node is the sum of the currents drawn by the wake-up circuit board and the current consumed by TelosB node (3.1μA for the biasing voltage of the FTDI chip handling the UART-USB communication + 0.2μA for MSP430 microcontroller in LPM4) and is given by

$$I_{\text{RTWAC}} = I_{\text{TelosB}} + I_{\text{wakeup}} = 3.3\mu A + 876nA = 4.176\mu A. \tag{2.26}$$

Hence the power consumption of an RTWAC node is P_{RTWAC} = 12.528 μW. Unlike constant power consumption on an RTWAC node, the power consumption of a MAC protocol is dependent upon its duty cycle value. We have calculated the power consumption for the reference implementation of B-MAC in TinyOS 2.0 at various duty cycles with the default channel polling duration of 1 ms when the hardware acknowledgements on a CC2420 radio are disabled. The power consumption is given by,

$$P_{\text{LPL\_MAC}} = P_{\text{TelosB\_sleep}} + \text{DutyCycle} \cdot P_{\text{poll}}. \tag{2.27}$$

Figure 2.27 shows the power consumption comparison of RTWAC with B-MAC protocol on TelosB platform in idle mode of operation. One may notice that an RTWAC node shows a remarkably low power consumption. Only B-MAC duty cycle values of lower than 0.001 % have a comparable power consumption but at a high cost of latency. For latency of RTWAC, the time required to transmit a complete modulated signal containing the data was measured. Since RTWAC uses PIE at the PHY layer, the latency depends upon the number of zeros and ones contained in a wake-up packet.

The transmission duration for a "1" and a "0" are $2T$ and $3T$, respectively. In our reference implementation, T=530 μs. So the average transmission duration for a 6 bytes long packet is 63.6 ms. The average offered latency for data communication on the reference B-MAC implementation is considered for a comparison. The maximum offered latency of a packet is given by

$$\text{B-MAC\_latency}_{\text{max}} = T_{\text{duty-cycle}} + t_{\text{pkt}}. \tag{2.28}$$

The minimum latency is equal to the time required for transmitting a packet,

$$\text{B-MAC\_latency}_{\text{min}} = t_{\text{pkt}}. \tag{2.29}$$

The average latency of B-MAC is:

$$\text{B-MAC\_latency}_{\text{avg}} = \frac{\text{B-MAC\_latency}_{\text{max}} + \text{B-MAC\_latency}_{\text{min}}}{2}. \tag{2.30}$$

In order to calculate t_{pkt} of B-MAC, raw packet size with default payload length of 28 bytes is considered. The MAC layer overhead is 12 bytes, and the synchronization header of the IEEE 802.15.4 PHY layer is 5 bytes. For the CC2420 raw data rate of 250 kbps, we have:

$$t_{\text{pkt}} = \frac{45\,\text{bytes}}{250\,\text{kbps}} = 1.44\,\text{ms}. \tag{2.31}$$

The average latency comparison of an RTWAC node with B-MAC implemented on TelosB platform is shown in Figure 2.28. It can be observed that the latency of RTWAC is comparable to B-MAC protocol at 1 % duty cycle. However, at this duty cycle

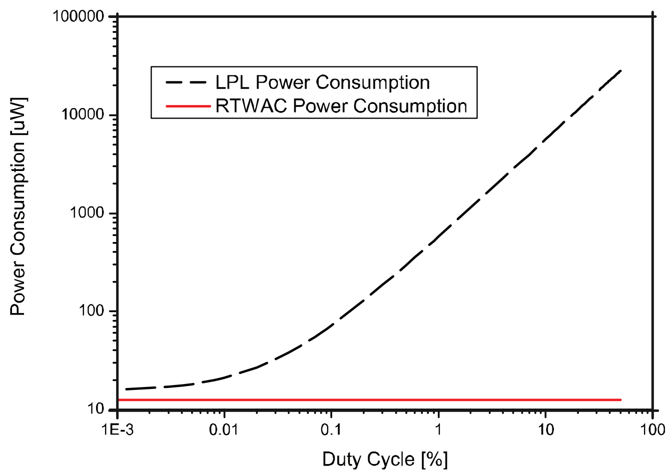


Figure 2.27: The average power consumption comparison of RTWAC with an LPL MAC (B-MAC) protocol at different duty cycles, implemented on the CC2420 radio. The channel polling duration for the MAC protocol is set to be 1 ms.

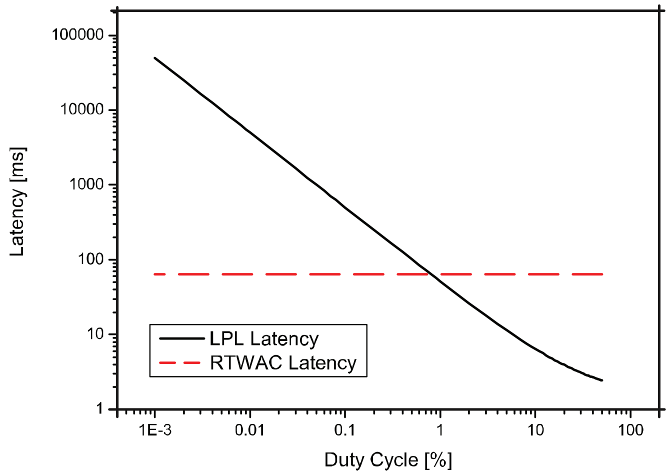


Figure 2.28: The average latency comparison of RTWAC with an LPL MAC (B-MAC) protocol implemented on the CC2420 radio at different duty cycles.

value, B-MAC implemented on TelosB platform consumes significantly higher power as compared to RTWAC.

The latency can easily be lowered for RTWAC by sending data packet at a faster rate. In our reference implementation, it is limited by the software implementation of the SPI protocol through bit-banging on the MSP430 microcontroller.

The amount of energy spent in receiving an addressed and a non-addressed packet is the same in RTWAC. The average time required to transmit a packet in our prototype implementation is 63.6 ms. The time required by the MSP430 microcontroller to switch from LPM4 mode to the active mode is negligible. In the active mode, the microcontroller consumes 340 μA at 3 V). Therefore, the energy spent in decoding one RTWAC message turns out to be approximately 65 μJ.

2.6.8 Discussion and Conclusions

We have developed a radio triggered wake-up solution with addressing capabilities. RTWAC minimizes idle listening power consumption and response time through a radio triggered wake-up circuit which uses on-board radio transceiver for data communication. In duty cycling MAC protocols, a high synchronization overhead has to be paid in order to align the active periods of nodes for data communication. Unlike the trade-off between energy consumption and latency in duty cycling MAC protocols, RTWAC allows keeping the power consumption at a node to a negligible level when communication is not required and instantly waking-up when required.

We have developed a prototype RTWAC wake-up circuit working in the 868.5 MHz ISM frequency band. It complies to the permitted power level and duty cycle constraints imposed by the frequency regulators in Europe. We have carried out the performance evaluation of our prototype RTWAC platform and conducted comparative empirical studies with B-MAC protocol implemented on TelosB. The results indicate that RTWAC consumes much lower energy and provides a significantly lower latency as compared to B-MAC protocol. Our approach of combining RTWAC with low-power MAC protocols running on the standard on-board radio of the low-power wireless nodes efficiently addresses application specific requirements. One of the test case scenarios for RTWAC has been its use for asset tracking logistic value chain.

2.7 SUMMARY

We have presented the design trends of MAC development for low-power embedded wireless networks, especially in the context of WSNs. We have shown that MAC protocols based on the preamble sampling principle suit better to network dynamics and low data rate requirements. A number of protocols have been proposed based on the preamble sampling principle and use different techniques for preamble length optimization. Depending upon the traffic and network conditions, different protocols perform better than others. We have developed TrawMAC, which has the ability to optimize its preamble length according to the traffic conditions. Therefore, it can adapt itself to a wide set of application requirements. Experimental performance evaluation indicates that TrawMAC outperforms other state-of-the-art solutions.

We have analyzed the dominant sources of energy consumption in low-power wireless applications and prototyped a dual radio platform, which minimizes the overall energy consumption at a node by dedicating specific operations to the two radios. We have developed MR-MAC protocol for the dual radio platform. It uses a sniffer radio operating in a lower frequency band for preamble sampling operation while a bursty radio operating in a higher frequency band for efficient data exchange. Experimental performance evaluation and comparison with B-MAC protocol show that MR-MAC utilizes radio resource efficiently and leads to much higher energy conservation and lower latency. These results hold against B-MAC implementation on single radio platforms using the two individual radios as are on our dual radio prototype platform. We have demonstrated MR-MAC prototype publically [89].

One of the dilemmas of duty cycling protocols is their inherent trade-off between energy consumption and the offered latency for data communication. Consequently, extremely low-power consumption at a node can only be achieved at a cost of high latency. We have developed RTWAC, which simultaneously allows highly energy efficient operation and low data communication latency response. RTWAC achieves radio triggered wake-ups with addressing capability and hence avoids energy wastage for non-addressed nodes. Furthermore, RTWAC combines radio wake-ups with duty cycling MAC protocols for reliable data communication over the on-board radio of a wireless node. Empirical performance comparison with B-MAC shows that RTWAC is highly suitable for applications requiring short range communication, fast response and extremely long lifetimes. We have demonstrated RTWAC publically [90].

3

SPECTRUM AGILITY AND COEXISTENCE

The increasing trend in the number of wireless devices and networks integrated into our daily life is leading to the crowding of existing wireless spectrum. Congestion in spectrum leads to mutual interference among devices sharing the same medium. Different protocols are designed and evaluated in isolated channel conditions. Without having appropriate considerations for avoiding potential interference and mechanisms for symbiotic coexistence, these protocols fail to achieve the desired performance characteristics in realistic interfering environments. The performance degradation is more significant for low-power embedded networks as they remain handicapped when competing with less resource constrained networks. In low-power embedded networks, traffic load is generally low and wireless channels usually remain under-utilized. Dedicating a wireless channel to each network is not only a waste of resource, but has also become less practical due to the increasing scarcity of the spectrum. A growing number of different wireless devices in shared wireless spectrum requires spectrally efficient operation that goes beyond simplistic MAC layers. Efficient use of spectral resources and the need for symbiotic coexistence features in the medium access procedures has therefore become necessary to avoid collisions and reduce packet retransmissions.

Spectrum agility and coexistence aspects are new to low-power embedded applications. We have identified the key design goals for MAC protocols aiming at spectrally efficient operation in low-power embedded applications in Section 3.1. Section 3.2 provides an analysis and comparison of different approaches aiming at dynamic spectrum access and mitigating wireless interference in resource constrained embedded networks. We will also discuss the implications and practical constraints in adapting MAC protocols designed for Cognitive Radios (CRs) [91] for low-power applications. Section 3.3 describes our novel approach on multi-channel polling operation combined with heuristics for channel quality assessment in order to achieve spectrum agile medium access and its prototype implementation. We will present the empirical performance evaluation of our prototype implementation on a TelosB testbed in the presence of different types of wireless interferers. Empirical results indicate that our scheme allows high degree of reliability for data communication through spectrally efficient operation. We have also developed a decentralized MAC protocol for opportunistic spectrum access for CR networks, which uses the heuristics based dynamic channel selection strategy. Section 3.4 presents the design details of the protocol and the performance evaluation on a testbed consisting of WARP SDR platform [15]. Classification and detection of potential wireless interferers certainly improves the reliability for data communication in spectrum agile and cognitive MAC protocols. Since most of the currently deployed low-power embedded applications use the 2.4 GHz ISM frequency band, we have prototyped a scheme which allows fast and reliable detection of the commonly interfering IEEE 802.11b/g networks. Section 3.5 presents our scheme and its performance evaluation. Finally, Section 3.6 summarizes the discussion in this chapter. This chapter is mainly based on our articles [10, 19, 34, 92–94], which were published during the dissertation work.

3.1 GOALS FOR LOW-POWER DYNAMIC CHANNEL ACCESS

MAC protocols are expected to provide reliable data communication with the acceptable levels of latency and throughput through energy efficient medium access. Though power consumption aspect influences every facet of network design, here we look into the algorithmic and protocol aspects of spectrum agile medium access. The opportunistic selection of an interference free channel is of high importance for a MAC design since failure to do so can either lead to high packet losses or subvert the ability of a MAC protocol to communicate. We have described the key characteristics of resource constrained networks in Section 2.1.1, which demand that spectrum agile channel selection is computationally simple, scalable, and able to handle network dynamics.

3.1.1 Lightweight Spectrum Sensing

Sensing spectral characteristics is fundamental to any cognitive and spectrum agile MAC protocol. Typically, nodes in a low-power network are equipped with simple radios, which can only provide Received Signal Strength Indicator (RSSI) or a threshold based CCA metric. The access to the I- and Q- modulation components for feature detection on the radio chip remains impractical with the present low-power radio technology. Texas Instruments CC2420, being the most popular radio used in many low-power embedded applications, provides a metric for Link Quality Indicator (LQI) over the exercised link, which can later be used for assessing the goodness of a channel [95]. Since spectrum sensing is highly energy consuming process, performing exhaustive spectrum profiling as is typically the case in the classical cognitive and spectrum agile MAC protocols [5] is unaffordable in low-power applications. This makes spectrum sensing more challenging for resource constrained embedded wireless networks.

3.1.2 Spectrum Occupancies

In many urban (home and office) deployments, nodes are expected to first detect potentially frequent and permanent interferes to be able to avoid them – for instance, Wi-Fi and ZigBee networks in the 2.4 GHz frequency band. In these situations, empirical models [96] can be established and fed to the network. However with the crowding of the spectrum especially in ISM frequency bands where resource constrained networks usually operate, availability of a free channel can be highly intermittent and volatile. Since spectrum occupancies vary temporally as well as spatially, agreeing on a single globally available network wide channel for data communication is unpractical. Owing to the diversified and varying nature of spectrum occupancy levels, nodes should be able to detect local spectrum holes at any given time and utilize them opportunistically.

3.1.3 Use of Machine Learning Algorithms

Some proposed cognitive MAC protocols rely on machine learning techniques in order to predict spectrum occupancies and tune MAC parameters appropriately [5]. Many researchers have a common belief that spectrum occupancies can be well predicted only if a fairly large set of samples is considered. We argue that intelligent guesses about spectrum occupancy characteristics can be made based on heuristics and learning methods even with partial or occluded information. Use of learning algorithms common to classical cognitive wireless networks are typically unfeasible for resource constrained networks because of the insufficient number of spectral samples and lack of computational power available at nodes. However, light-weight learning techniques have a high potential for predicting spatial and temporal occupancy characteristics of interferers and Primary Users (PUs).

3.1.4 Use of Control Channel(s)

Wireless networks generally use a single half duplex radio transceiver[1] and therefore the same radio is used for sending control information as well as data. Typically the same wireless channel is used to send control information as data. However, a few designs are available, where a secondary radio is used for transmitting a busy tone signal in an entirely different frequency band [56] or a separate secondary channel is used in the same frequency band to transmit control information [99]. MR-MAC [19] achieves spectrum agility through a dedicated control channel in less congested frequency band to establish an agreement on the use of a bursty data channel in crowded frequency band. Typically, resource constrained networks such as WSNs consist of a large number of nodes and using a dual-radio platform increases the economic cost. Furthermore, owing to the increasing number of non-cooperative devices and a wide scale penetration of different wireless technologies, guaranteeing a dedicated interference free Common Control Channel (CCC) is becoming unpractical.

3.1.5 Lack of Infrastructure

Only a little has been done so far towards standardization of ad hoc sensor networks. Therefore, different networks neither coordinate among themselves nor with other networks for dynamic channel access. IEEE 802.22 [100] working group focuses on infrastructure based centralized coordination of the available channels for coexistence of cognitive users in the TV frequency spectrum. In low-power embedded networks, absence of coordination and centralized controlling infrastructure restrict dynamic spec-

[1]Although a working prototypes of a full duplex transceivers have recently been presented [97, 98], no transceivers are yet commercially available.

trum access for potentially exploiting spectrum opportunities as in IEEE 802.22. Due to lack of coordinating infrastructure, explicit synchronization among mobile ad hoc nodes and sharing of network-wide spectrum occupancy information becomes difficult. Therefore, protocols relying on distributed algorithms for dynamic channel selection are more practical.

3.1.6 Support for Network Dynamics

The network dynamics in low-power applications are because of the node mobility as well as due to new nodes appearing and old nodes dying out from a network. The network dynamics demand frequent updates of the neighbourhood and channel occupancy information, which disfavor the use of graph coloring methods or fixed channel assignments such as [101]. Furthermore, owing to low data rate requirements, topology based scheme, like [102], for assigning different channels to different portions in the network becomes expensive in terms of the control information overhead. The update rate for neighbourhood channel assignment information depends upon the degree of network dynamics. High network dynamics demand more frequent need for updating the neighbourhood channel information. MAC protocols supporting mobility such as [103, 104] are devoid of any spectrum agile channel selection methods. Xu *et al.* devised a scheme for coordinated channel switching to avoid channel jamming by exploiting the link quality metrics from the routing layers [105]. Their scheme works well when the jammer is not intermittent and the network has low mobility. Low overhead spectrum agile MAC protocols are needed that allow nodes to asynchronously join and leave a network and are still be able to converge to the channels used by their neighbours.

3.2 EXISTING APPROACHES AND THEIR SHORTCOMINGS

Spectrum sharing and symbiotic coexistence are becoming inevitable due to a rapidly growing number of wireless networks and technologies. CR technologies aim at enabling higher spectrum utilization by opportunistically using spectrum opportunities through Dynamic Spectrum Access (DSA) schemes and cross-layer design methodologies. Medium access techniques in CR networks allow nodes to efficiently communicate with each other without affecting the performance characteristics of a PU[2]. Secondly, they allow mitigating interference through spectrum agility thereby facilitating coexistence of different networks. Channel sensing and spectrum access are tightly coupled to medium access—no matter if it is through random access, time slotted principle or hybrid scheme. In recent years, a number of spectrum agile and cognitive MAC protocols have been proposed [5, 106]. These solutions have different requirements for network infrastructure and hardware capabilities of the Secondary Users (SUs) or CR nodes.

[2]Since most of the present low-power applications use ISM frequency bands so as such a PU does not exist. Instead, the goal becomes to avoid interfering transmissions.

There are both centralized and distributed MAC designs for CR networks [5]. MAC protocols in infrastructure based networks require a central controller, e.g., for management of network activities, synchronization among nodes, gathering and distribution of data. IEEE 802.22 [100] standard for Wireless Regional Area Network (WRAN) is a good example in this category. As compared to the distributed protocols, centralized MAC approaches typically demand simpler hardware and software capabilities for CR nodes. MAC protocols relying on a centralized infrastructure, at the same time, impose much stringent requirements in terms of sensing and coordination on the infrastructure. These include various mechanisms for spatio-temporal spectrum sensing and active coordination among nodes. Numerous centralized protocols, such as [107, 108] have been designed. Lien *et al.* [107] use the CSMA principle for underlay systems [109], where a SU activity is permitted with simultaneous PU activity as long as the interference caused to the PU stays within a radio technology dependent threshold. The protocol enables coexistence by adjusting the transmit power and data rates of the CR nodes. The intelligence for adapting the protocol parameters lies at the controlling infrastructure. Most of the MAC designs in CR networks rely on high degree of cooperation among the SUs and a centralized entity coordinating the DSA. If the spectrum dynamics are high, much higher granularity and accurate sensing are required for MAC protocols [110]. In situations, where accurate spectrum occupancies are hard to predict and cooperation from the PU or the interfering network is less likely, decentralized MAC protocols are desirable, which is typically the case for low-power embedded applications operating in ISM frequency bands.

Cognitive users gather spectrum information either locally or through cooperation from neighbouring nodes. These protocols have different assumptions on the hardware platform and the network capabilities. Some of these schemes assume that cognitive users are equipped with multiple transceivers, and nodes have the capability to access multiple channels simultaneously and select the best channel [111–113]. Some of the decentralized MAC protocols assume that all the entities in a network have an always available CCC for exchanging control information and establishing an agreement on the selection of data transmission channel between the transmitter and receiver pairs. We refer the reader to [114] for a detailed taxonomy of CR MAC protocols. The use of a CCC has its limitations. An in-band CCC is exposed to PU activities and thus its reliability and availability are not guaranteed while an out-of-band CCC requires either an extra radio interface or active switching between different frequency bands. Furthermore, a CCC needs to be allocated by a regulator and agreed through standardization. Simulation based studies have been conducted to show the theoretical effectiveness of the above mentioned protocols. However, these lack performance measurements in real environments and network conditions. C-MAC [115] is a decentralized cognitive MAC with a prototype implementation. Instead of fixing a CCC, C-MAC uses a rendezvous channel, which is selected based on the reliability of the available channels. It uses the TDMA principle, where each CR node has a distinct slot for periodic beaconing duration. This work is definitely a way forward and shows the feasibility of implementing a cognitive MAC but the evaluation of the prototype implementation is not provided. Though a real implementation of C-MAC has been carried out, yet it is impractical for low-power applications mainly due to its fairly high synchronization requirements and energy hungry beaconing mechanism.

DSA and opportunistic channel selection strategies are relatively new to low-power embedded networks such as WSNs. Although many previous research studies have indicated performance degradation for resource constrained networks in the presence of wireless interference [7, 8, 11]. Owing to the communication and computational constraints, many of the existing cognitive and spectrum agile MAC protocols [5, 116] cannot be directly migrated to low-power applications. While many commercially available low-power radios support multi-channel operation, most of the MAC protocols use a fixed single channel [3]. A few multi-channel MAC schemes [117, 118] have been proposed for resource constrained networks but with an objective of achieving higher throughput by utilizing additional bandwidth at nodes. Wu *et al.* suggest of assigning different channels to nodes in order to efficiently balance the traffic loads [119]. While these schemes are useful when higher traffic volumes are to be supported, inability to select interference free channels hampers their operation in the presence of environmental interference. Graph coloring schemes such as [120, 121] have been designed for mitigating channel interference and simulation results show performance improvements. Borms *et al.* proposed a technique for dynamic channel selection [122], which is based on TDMA and uses frequency hopping principle as described in McMAC [123]. The authors argue that drifts of $\pm$ 40ppm in clock crystals require a synch period of 52 min. Though the required synch period and clock drifts are not very strict, the protocol has deficiencies when it comes to network scalability and coping with mobile environments.

Schurgers *et al.* have proposed a MAC protocol, STEM, using two radios operating in separate frequency bands to completely isolate data transfer from control and radio wake-up [56]. In the tone based approach of STEM, a long wake-up tone is sent to make sure that the destination node has awaken. The protocol does not consider wireless interference while selecting the channels. In contrast to the approach in STEM, MR-MAC uses a low data rate CCC in less interfering frequency band to agree on the selection of a bursty radio channel in crowded high frequency band [19]. Having two radios on a platform however increases the economic cost of a network. Furthermore, crowding of the spectrum is making the availability of a dedicated non-interfering CCC impractical in many deployment scenarios. A scheme for dynamically selecting an interference free channel using a quad-radio platform is described [124]. The use of multi-radio platforms have a downside of increased cost per node. Many low-power applications, especially in WSNs use CC2420 [42] radio operating in the 2.4 GHz frequency band. It is well known that commonly present IEEE 802.11b/g networks cause heavy interference to IEEE 802.15.4 based networks [11, 125]. Yuan *et al.* establish a model for the coexistence of IEEE 802.15.4 and IEEE 802.11b/g networks depending on the power and timing aspects for certain operating ranges [126]. Though their simulation results promise coexistence, it requires modification of the timing aspects of the two protocols and does not completely solve the problem of receiving mutual interference. The reference [127] claims that IEEE 802.15.4 compliant CC2420 radio has good properties for alternate and adjacent channel rejection thereby allowing it to coexist with WLAN, Bluetooth and other IEEE 802.15.4 based networks. However, these measurements do not consider the MAC behaviours and therefore the performance of IEEE 802.15.4 networks suffer heavily due to interference in real deployments [126].

3.3 SPECTRUM AGILE MAC PROTOCOL FOR WSNS

In order to allow dynamic spectrum access and minimize wireless interference in low-power applications, we have developed Spectrum Agile MAC (SA-MAC) protocol. According to the best of our knowledge, our protocol is the first practical realization of a spectrum agile MAC scheme for low-power embedded networks that uses the CR paradigm. The performance evaluation indicates that SA-MAC is very promising in efficiently solving dynamic spectrum management issues in resource constrained embedded network environments. In the following, we describe the design details of SA-MAC protocol.

3.3.1 Protocol Details

SA-MAC is based on the preamble sampling principle. Using a light-weight channel selection strategy, a transmitting node dynamically selects an interference minimal channel based on the observed spectral characteristics and sends a preamble sequence followed by data. Unlike single channel MAC protocols, a receiving node sequentially scans all the frequency channels available in its pool and is able to detect activity in the channel being used by the transmitter. At the same time, the receiving node is also able to ascertain the presence of external interferers and their strengths. Since our protocol is based on the preamble sampling scheme and uses a distributed channel selection algorithm, its characteristics suits well to the dynamic nature and light traffic requirements in low-power applications. Figure 3.1 shows the LPL operation of the MAC protocol, where nodes A, B and C poll the channels asynchronously from one another. Unlike the classical definition of duty cycle for single channel MAC protocols, which is defined as the ratio of on-time of the radio to the sum of the on-time

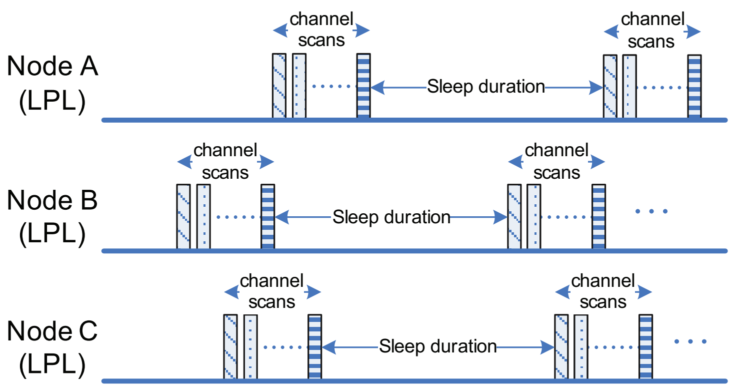


Figure 3.1: Multi-channel preamble sampling operation in SA-MAC protocol.

and the sleep duration, the duty cycle of SA-MAC however is defined to be the ratio of radio on-time in one of the channels to the total period. This is so because if the radio is on in another channel, the transmission is invisible to it and therefore a longer preamble transmission is required to compensate the sensing duration of other channels at a node. SA-MAC shortens the length of the preamble sequence to achieve energy conservation by combining different techniques from single channel protocols such as MFP-MAC [43], WiseMAC [38], and X-MAC [44].

Preamble Optimization

In preamble sampling MAC protocols, a preamble sequence sent before the data and implicitly synchronizes the active periods of a transmitter and receiver(s). Since the sampling periods of the nodes are typically long, the required preamble length to be transmitted/received also correspondingly becomes long causing energy wastage both at the transmitter and the receiver(s). In the following, we describe the preamble shortening techniques used by SA-MAC protocol.

Microframes

Instead of using a long monolithic preamble, it is divided into a series of back-to-back Micro Frame Preamble (MFP). Each MFP contains the information of the destination node, the exact time of the data frame transmission and the sleep schedule of the transmitting node. Upon receiving an MFP, a non-addressed node immediately switches back to the sleep state and is therefore able to avoid receiving the unnecessary preamble sequence and the data packet following it. An addressed node also goes to sleep in order to avoid receiving the rest of the preamble frames. However, it awakens again at the right instant to receive data frame(s). Figure 3.2 illustrates the operational cycle of the MAC, where the transmitter (TX) first scans all the channels in the pool in order to make sure that there is no on-going transmission. Then it chooses the most suit-

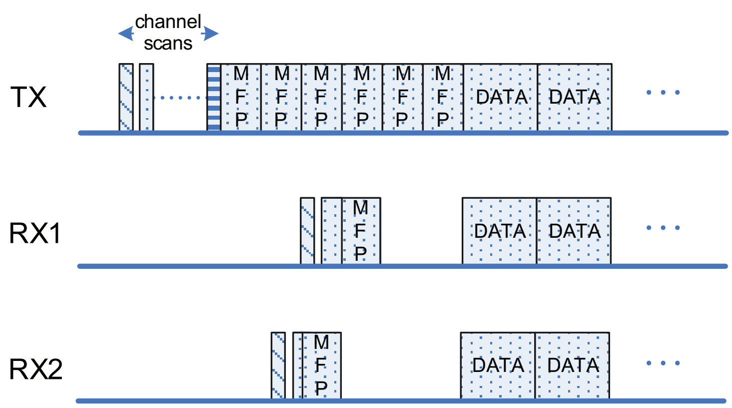


Figure 3.2: Broadcast transmission in SA-MAC. Back-to-back MFPs followed by the data frames are sent for a duration equal to the periodic channel check interval.

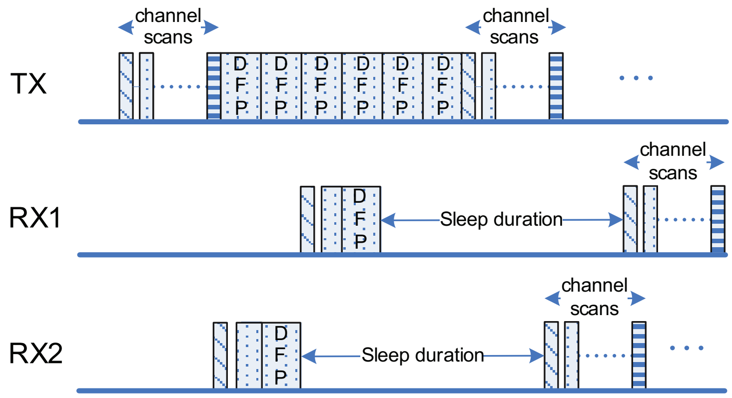


Figure 3.3: Broadcast transmission in SA-MAC. Back-to-back DFPs are sent for a duration equal to the periodic channel check interval. The data bytes are piggy-backed onto the preamble frames.

able channel (based on a channel selection algorithm described later in this section) and transmits the preamble framelets followed by the data frames. The asynchronously awakening nodes (RX1 and RX2) poll the channel and detect the transmission, receive a complete MFP, go back to sleep and wake up again to detect an ongoing transmission. Experiments have shown that if the data size is small, waking-up again in order to receive the data frame is less efficient than piggy-backing the data to the preamble framelets. Based on this premise, if the data size is smaller than a certain empirically found threshold, the data is piggybacked onto the preamble framelets to form a Data-Frame-Preamble (DFP). A series of back-to-back DFPs serve the purpose of both preamble and data. Figure 3.3 shows the operational cycle of the MAC for the case of DFP. Note that the asynchronously waking-up nodes RX1 and RX2 need to receive just one complete preamble frame (DFP). SA-MAC exercises the above described preamble shortening techniques only for broadcast transmission. For unicast transmission, additional information is exploited to further reduce the length of the transmission/reception as described in the later subsections.

Neighbourhood Sleep Schedules

SA-MAC protocol maintains the sleep schedules of the neighbouring nodes in order to apply the preamble shortening technique for unicast transmission in a manner similar to WiseMAC [38]. However, unlike WiseMAC which announces the sleep schedules in the acknowledgement frames, nodes in SA-MAC announce their sleep schedule within the preamble framelets. This also allows non-addressed nodes to update their sleep schedule tables. A transmitting node shortens the length of the preamble by delaying the packet transmission until the destination node is scheduled to wake-up. In the ideal case, a transmitter only needs to send one preamble frame just when the receiver is scheduled to wake-up. Unicast transmission implies that the transmitting node wake-ups according to the wake-up schedules of receiver(s) but also forces itself not to miss waking-up according to its own sleep schedule so that any other potential transmitter would still be able to address it at its scheduled wake up instant. In unicast transmissions, the amount of energy saved by the transmitting node is proportional to the delay in packet transmission.

Preamble Strobing

SA-MAC combines the neighbourhood sleep schedule based optimization with preamble strobing technique [44] for unicast transmission. After transmitting a preamble frame, the transmitter waits for an acknowledgement of the frame from the receiver. Subsequent preamble frames are transmitted after timing out for the acknowledgement of the previously transmitted preamble frame. After receiving the acknowledgement for a DFP, a transmitter stops sending further DFPs as the data (within the DFP) has already been delivered to the destination node. In the case of an MFP, after receiving the acknowledgement, a transmitter immediately sends the data frames. In this way a transmitter only needs to send a reduced preamble sequence. In SA-MAC, a transmitter first tries to delay the transmission until the scheduled wake-up time of the receiver and exercises preamble strobing. This is beneficial because as time passes, clock drifts are established between the nodes and the sleep schedule estimation of the neighbours become imprecise. SA-MAC has the ability to compensate the clock jitters established in the neighbourhood sleep schedule estimation over time due to genetic inaccuracies of crystal oscillators through the preamble strobing principle. In the best case, with perfect timing estimation of a neighbour's sleep schedule, only one micro-frame is

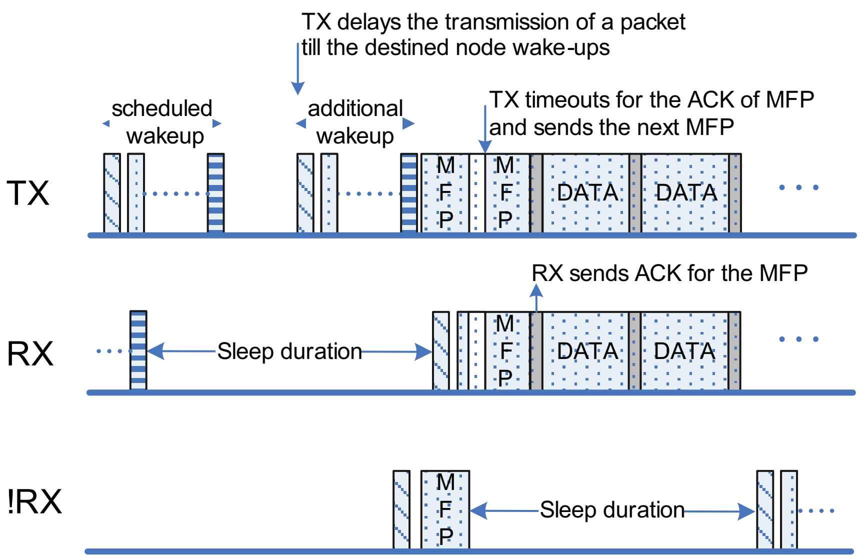


Figure 3.4: Unicast transmission in SA-MAC, where neighbourhood sleep schedule information is combined with preamble strobing technique. After an MFP is acknowledged, data frames are immediately sent.

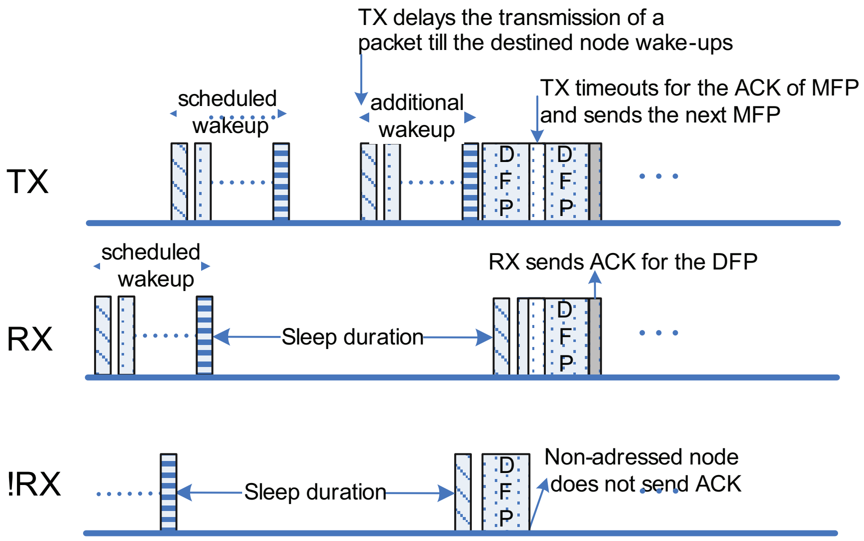


Figure 3.5: Unicast transmission in SA-MAC. After a DFP is acknowledged, the transmitter stops sending further DFPs.

needed to be transmitted and in the worst case the length of the preamble transmission becomes equal to the periodic channel check interval (similar to the broadcast case). Figure 3.4 shows the unicast transmission for the case when the transmitter (TX) does not have the perfect knowledge of the sleep schedule of a receiver (RX). After timing out for the acknowledgement of the first MFP, the transmitter sends the next MFP frame which is acknowledged by the receiver and the data frames follow immediately. A non-addressed node (!RX) on the contrary goes to sleep after receiving an MFP for the entire duration of the transmission. Figure 3.5 shows the unicast transmission for the DFP case where after receiving a complete DFP frame, the destination node sends an acknowledgement, which marks the completion of the transmission and the transmitter stops sending further DFPs.

Channel Selection Algorithm

SA-MAC uses a heuristics algorithm for selecting the communication channel. In the following, we describe the various parts of the algorithm.

Channel Weights

Efficient spectrum sensing is an important aspect of the protocol. A more realistic and practical metric obtained from low-power radios is the spectral energy detection. Scanning all the channels is not only an energy expensive operation but also imparts high latency. On the other hand, having a small subset of channels in the pool reduces the diversity for opportunistically finding an interference free channel. In SA-MAC, a small subset of channels are chosen in the pool but at the same time the channels are refreshed as the quality deteriorates. This way, a lower latency and a lower energy consumption in channel scanning operation is achieved and high quality channels can be maintained in the pool. The size of the channel pool can be pre-selected based on the type of spectrum interference. For permanent interferers, a small channel pool suffices while more dynamic interferers require a larger pool size. Further details on dynamic pool sizing is described later in this section.

In order to categorize the quality of channels, a weight is associated with each channel. The weights are readjusted each time channel polling operation is performed. If a particular channel is sensed idle, its weight is increased by one. When data communication is initiated in a channel (whether the radio is in the transmit mode or in the receive mode), the channel weight is increased by two. Even if a node is not the destination, the associated weight is increased by two because from the transmitter's perspective the exercised channel is the best. If during channel scanning, an interferer is detected, the associated channel weight is decreased by two. If a certain channel is found to be frequently occupied by an external interferer over time, its weight is reduced so much that it is dropped out from the pool of the available channels. If the channel pool size becomes small or the channel quality is observed to be deteriorated, the nodes expand the number of channels in the pool. Maintaining history of the channels allows avoiding re-inclusion of the channels that have previously been discarded from the channel pool.

Channel History

The channel selection scheme maintains a record of the past channel activities of all the channels. Maintaining history of the channels, which are not in the pool helps in keeping track of the interfering channels while replenishing the pool. This also allows considering the updated information about the channels which are not in the local pool of a node and receive an update from the neighbouring nodes in the form of channel maps. A local time-stamp on the updates indicates the validity of the channel history. Using a local time-stamp at a node alleviates the need for global clock synchronization. The number of events required to be stored in the channel history depends upon the sampling intervals and the spectrum dynamics. If the sampling interval is small, relatively sparse samples can be stored. Besides storing raw events, statistical metrics on the channel weights such as the variance and the median value is also stored, which gives an indication of the environmental spectrum dynamics. The channels are sorted in the descending order of their weights in the pool every time the medium is sampled. If the channel weights are found to be equal, the channel with higher score for the last activity (IDLE, RX, TX, INTERFERER) is stored first in the sorted pool. During the polling operation, channels are polled in the descending order of their quality or weights. Since the transmitter selects the channel with best quality (i.e., with the highest weight), scanning channels in descending order of the weights helps receiving nodes in detecting the transmission earlier. Hence, the receiving nodes can save energy by avoiding the need to scan channels further.

Threshold based Channel Pool Expansion and Contraction

During the polling operation, all the channels in the pool are scanned. A larger pool-size leads to higher energy consumption for channel polling activity. Keeping a smaller pool size is energy efficient but at the same time, a smaller pool size reduces the diversity in channel selection. In SA-MAC, we aim at achieving the best of the two by keeping a small pool size and updating the channels in the pool dynamically as the quality of channels deteriorates. The channel pool expansion and contraction is distributed in nature and a node relies on the locally sensed information. The channel expansion and contraction algorithm uses a combination of dual (hysteresis) threshold values for channel weights, τ_1 and τ_2 with $\tau_1 < \tau_2$. Initially, a big channel pool is created with all the weights initialized to τ_2. The initial pool size depends on the interference conditions where the network is deployed. Over multiple sensing intervals, if the channel is found interfering and consequently its weight falls eventually below τ_1, it is deleted from the pool thereby contracting the pool. On the contrary, if the channel quality in the contracted channel pool goes inferior over time, the pool size is expanded and new channels are popped-in with their weights initialized to τ_2. The channel pool quality is indicated through the median and the maximum channel weight values. All the channels with weights equal to or below the median weight are replaced when the channel quality is detected to be degraded. Keeping the history of the channels, a lower priority is given to the channels, which have previously been deleted from the pool when adding a channel to the pool. Furthermore, the channels adjacent to the previously deleted channels have lower preference for inclusion into the channel pool in order to avoid potentially wider band interferers. In our prototype implementation, the threshold values τ_1 and τ_2 are selected to be 3 and 7, respectively. The maximum channel weight is fixed to 20. The channel pool is considered to be deteriorated and is replenished when the median weight of the channels becomes equal to or smaller than 4 and the highest weight becomes equal to or smaller than 6. The above mentioned values were found experimentally through the prototype implementation on Texas Instruments CC2420 radio.

3.3.2 Analysis for the Optimal Energy Consumption

We have analytically modeled SA-MAC protocol and derived an expression for the optimum duty cycle giving minimal energy consumption for a given network. Since the energy consumption directly depends upon the traffic loads and the type of the traffic, we consider the cases of unicast and broadcast communication patterns have been considered separately in our formulation. In the following, we will derive an expression for the case when MFP is used by the MAC protocol. Similarly, optimal duty cycle expression for DFP can also be derived.

Models and Parameters

Consider a network neighbourhood consisting of $n + 1$ nodes and assume that each node transmits data packets of length l_{pkt} periodically at the rate r_{data} per second.

Each node spends power in the operations: radio initialization, carrier sensing, packet transmission, packet reception, channel polling and sleep state, which are denoted by $P_{\text{radio\_setup}}$, P_{cs}, P_{tx}, P_{rx}, P_{poll} and P_{sleep}, respectively. In the following, we list the terms used in our model:

Notation	*Meaning*
$t_{\text{poll\_once}}$	Single channel polling interval
n_{ch}	Number of channels to be polled
$t_{\text{radio\_setup\_once}}$	Single initialization duration of the radio
$t_{\text{poll\_period}}$	Channel sampling period
L_{mfp}	Length of a microframe preamble
N_{mfp}	Number of microframes
$t_{\text{cs\_once}}$	Channel carrier sensing interval
l_{pkt}	Length of the data packet
t_{b}	Bit duration
l_{ack}	Length of the acknowledgement frame

Broadcast

The overall energy expenditure of a node is the sum of the energy spent in each operation. Normalizing the energy consumption over a channel polling interval is expressed as,

$$E = E_{\text{radio\_setup}} + E_{\text{poll}} + E_{\text{rx}} + E_{\text{cs}} + E_{\text{tx}} + E_{\text{sleep}} \tag{3.1}$$

$$E = P_{\text{radio\_setup}} t_{\text{radio\_setup}} + P_{\text{poll}} t_{\text{poll}} + P_{\text{rx}} t_{\text{rx}} + P_{\text{cs}} t_{\text{cs}} + P_{\text{tx}} t_{\text{tx}} + P_{\text{sleep}} t_{\text{sleep}} \tag{3.2}$$

Normalizing the individual time durations with a factor of the channel polling period, ($t_{\text{poll\_period}} = L_{\text{mfp}} N_{\text{mfp}} t_{\text{b}}$), gives

$$\begin{aligned}
t_{\text{poll}} &= \frac{n_{\text{ch}} t_{\text{poll_once}}}{t_{\text{poll_period}}} \\
t_{\text{radio_setup}} &= \frac{n_{\text{ch}} t_{\text{radio_setup_once}}}{t_{\text{poll_period}}} \\
t_{\text{rx}} &= n r_{\text{data}} (1.5 L_{\text{mfp}} + l_{\text{pkt}}) t_{\text{b}} \\
t_{\text{cs}} &= n_{\text{ch}} r_{\text{data}} (t_{\text{poll_once}} + t_{\text{radio_setup}}) \\
t_{\text{tx}} &= r_{\text{data}} (N_{\text{mfp}} L_{\text{mfp}} + l_{\text{pkt}}) t_{\text{b}} \\
t_{\text{sleep}} &= 1 - t_{\text{radio_setup}} - t_{\text{poll}} - t_{\text{rx}} - t_{\text{cs}} - t_{\text{tx}}
\end{aligned}$$

An SA-MAC node requires listening for one to two preamble frames (an average of 1.5) in order to determine the timing information of the data frames and to determine the destination address. A transmitter, however, needs to send preamble frames for a duration equal to the sampling period (the periodic check interval of the channel being used). Unlike single channel preamble sampling MAC protocols, SA-MAC pays the price of scanning multiple channels in the polling operation, which reduces the effective sleep duration for a given sampling period. We have experimented with our prototype implementation and observed that the time needed to setup the radio in a particular channel is non-negligible and accounts for the time needed to send commands over SPI or UART interface in order to start the radio in the desired configuration. Additionally, it includes the time required for the crystal oscillator to stabilize.

Our aim is to find the relationship of the sampling period (directly related to duty cycle) which leads to the minimum energy consumption. Since the sampling period is directly related to the length of the preamble, the number of required MFPs corresponding to the minimum energy consumption is determined. Plugging in the terms in equation (3.2) and taking the derivative w.r.t. N_{mfp} gives,

$$\begin{aligned}\frac{dE}{dN_{\text{mfp}}} &= -\frac{n_{\text{ch}}(P_{\text{poll}}t_{\text{poll\_once}} + P_{\text{radio\_setup}}t_{\text{radio\_setup\_once}})}{L_{\text{mfp}}N_{\text{mfp}}^2 t_{\text{b}}} \\ &+P_{\text{tx}}r_{\text{data}}L_{\text{mfp}}t_{\text{b}} + n_{\text{ch}}\frac{P_{\text{sleep}}(t_{\text{poll\_once}} - t_{\text{radio\_setup\_once}})}{L_{\text{mfp}}N_{\text{mfp}}^2 t_{\text{b}}} \\ &-r_{\text{data}}L_{\text{mfp}}t_{\text{b}}P_{\text{sleep}}.\end{aligned} \tag{3.3}$$

Requiring $\frac{dE}{dN_{\text{mfp}}} = 0$ and simplifying the terms gives the optimum number of MFPs.

$$\hat{N}_{\text{mfp}} = \sqrt{\frac{n_{\text{ch}}(t_{\text{poll\_once}}(P_{\text{poll}} - P_{\text{sleep}}) + t_{\text{radio\_setup\_once}}(P_{\text{radio\_setup}} - P_{\text{sleep}}))}{r_{\text{data}}(L_{\text{mfp}}t_{\text{b}})^2(P_{\text{tx}} - P_{\text{sleep}})}}. \tag{3.4}$$

The reader should note that since $\hat{N}_{\text{mfp}} \in N$, we take the ceiling value. The expression for the optimum number of microframes is independent of the number of nodes in the network because here we have considered a congestion free case where all the nodes are able to transmit their queued packets. The network of size $n + 1$ governs a lower bound, $n \leq (r_{\text{data}}t_{\text{b}}(L_{\text{mfp}}N_{\text{mfp}} + l_{\text{pkt}}))^{-1}$. The optimal sampling time expression is, $S.T._{\text{opt}} = \hat{N}_{\text{mfp}}L_{\text{mfp}}t_{\text{b}}$.

Unicast

The total energy consumption of a node for unicast transmission is also expressed by the equation (3.2). However in the case of unicast transmission, a node happens to be the destination for k packets out of the total n packets it hears from its neighbours. SA-MAC optimizes the preamble length, i.e, the number of MFPs to be sent for unicast transmission by using the techniques described in Section 3.3.1. In the best case with the perfect knowledge of the sleep schedule of a neighbour, only one MFP is needed to be sent. In the absence of any timing information, the number of MFPs needed to be sent depends upon the offset between the sleep schedules of the transmitting and receiving nodes. In the worst case, the required number of MFPs needed to be sent correspond to the periodic channel check interval as in broadcast transmission and is given by equation (3.4). The timing expressions are given as below:

$$
\begin{aligned}
t_{\text{poll\_period}} &= L_{\text{mfp}} N_{\text{mfp}} t_{\text{b}} \\
t_{\text{poll}} &= \frac{n_{\text{ch}} t_{\text{poll\_once}}}{t_{\text{poll\_period}}} \\
t_{\text{radio\_setup}} &= \frac{n_{\text{ch}} t_{\text{radio\_setup\_once}}}{t_{\text{poll\_period}}} \\
t_{\text{rx}} &= n r_{\text{data}} (1.5 L_{\text{mfp}} t_{\text{b}}) + k r_{\text{data}} l_{\text{pkt}} t_{\text{b}} + 2 r_{\text{data}} l_{\text{ack}} t_{\text{b}} \\
t_{\text{cs}} &= n_{\text{ch}} r_{\text{data}} (t_{\text{poll\_once}} + t_{\text{radio\_setup}}) \\
t_{\text{tx}} &= r_{\text{data}} (N_{\text{mfp}} L_{\text{mfp}} + l_{\text{pkt}}) t_{\text{b}} + 2 k r_{\text{data}} l_{\text{ack}} t_{\text{b}} \\
t_{\text{sleep}} &= 1 - t_{\text{radio\_setup}} - t_{\text{poll}} - t_{\text{rx}} - t_{\text{cs}} - t_{\text{tx}}
\end{aligned}
$$

A factor of two appears in the expression for acknowledgement frames because a node needs to send/receive acknowledgement(s) for both MFPs and data frames. In this formulation, only one data frame transmission is considered however, SA-MAC also allows sending multiple data frames with a single preamble reservation. The individual acknowledgements of the data frames need to be included in the model correspondingly. SA-MAC intrinsically optimizes the number of preamble frames needed to be sent/received. However, in the absence of any neighbourhood sleep schedule information, network is designed based on the worst case. Similar to the case of broadcast transmission, in unicast, the network follows a lower bound, $n \leq (r_{\text{data}} t_{\text{b}} (L_{\text{mfp}} N_{\text{mfp}} + l_{\text{pkt}} + 2 l_{\text{ack}})^{-1}$.

3.3.3 Performance Evaluation

We have implemented SA-MAC protocol in TinyOS 2.x for the TelosB platform. We have conducted extensive performance evaluation of SA-MAC protocol with different parameter settings and under different types of interference. The performance metrics include power consumption, throughput and successful packet delivery ratio under different conditions of traffic loads, number of channels in the pool, sampling periods and radio transmit power levels. The interference effects have been studied spatially as well as temporally with varying transmit powers and bandwidth occupation. Furthermore, we have carried out a comparative study against a single channel protocol, B-MAC+ [45]. Each experiment was repeated five times to obtain better statistics.

Optimal Power Consumption: Comparison of Analytical and Empirical Results

In order to validate the analytical model for the optimal power consumption in a given network with a certain traffic load, we have conducted power consumption studies. A small uncongested network consisting of only three nodes is considered. All the nodes are programmed to use a channel pool size of four, generate broadcast data packets of 100 bytes payload at a rate of 1/16 (0.0625) packets per second and 1/2 (0.5) packets per second at different sampling intervals. Figure 3.6 shows the average power consumption for this experiment. It is easy to observe from the figure that at small

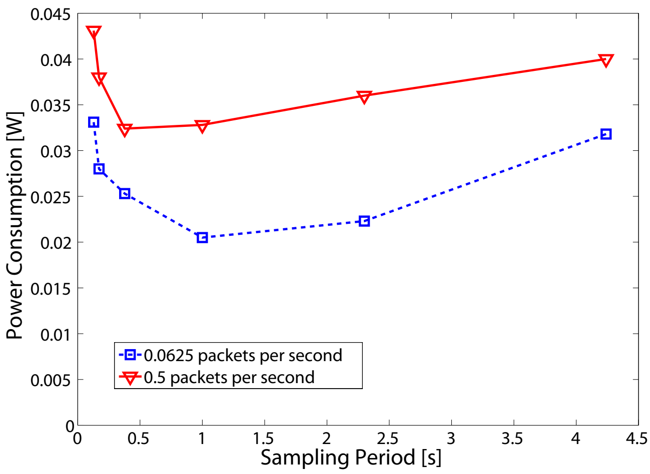


Figure 3.6: Power consumption of SA-MAC at a TelosB node with different sampling periods and on different traffic data rates. In these measurements, an uncongested network of three nodes is considered. All the nodes use a channel pool size of four and are in the perfect wireless range of each other with 0 dBm of transmit power.

sampling periods, energy consumption show an increasing trend because of more frequent channel polling operations. On the other hand, the increasing trend in power consumption with higher sampling periods is because of the need to transmit longer preamble sequence. From the figure, it is evident that the optimum sampling period for $1/16$ and $1/2$ packets per second were obtained at 1 s and 377 ms, respectively. We have measured the power consumption and timings for various operations on the TelosB platform as listed in the Table 3.1. In order to verify the analytical model of the MAC protocol as described in Section 3.3.2, we put the measured values in optimal sampling time expression. Table 3.2 shows the analytical values for the optimum sampling periods of 1.1519 s and 407.2 ms for data packet rates of $1/16$ and $1/2$, respectively. It can easily be seen that our analytical model very closely adheres to the empirical values.

Table 3.1: Parameter values on a TelosB node using SA-MAC.

Parameter	*Measured Value*
Power in receive mode (P_{rx})	58.9 mW
Power in channel polling (P_{poll})	58.9 mW
Power in radio setup (P_{radio_setup})	10.7 mW
Power in transmit mode (P_{tx})	46.5 mW
Power in sleep mode (P_{sleep})	3.6 mW
Time for one channel polling (t_{poll_once})	15.8 ms
Time for once radio setup ($t_{radio\_setup\_once}$)	2.4 ms
Time to transmit/receive a bit (t_b)	$1/250000$ s

Table 3.2: Optimal sampling periods at different data packet rates.

Data packet rate [s^{-1}]	*Optimal sampling period* [s]
0.0625	1.1519
0.5	0.4072

Relationship of Power Consumption with Channel Pool Size and Sampling Period
In many low-power embedded network applications, the average traffic load is relatively low so that the idle listening power consumption starts to dominate. Idle listening power consumption is the power expended in the channel polling operation when no data packets are exchanged. Since SA-MAC is based on the preamble sampling principle, it does not impose explicit synchronization of the sleep schedule of the node and the only energy consumption cost comes from the channel polling operation when no data is needed to be transmitted/received. Figure 3.7 shows the relationship of power consumption during the channel polling (LPL) with respect to the sampling period and the number of channels in the pool. The plot is obtained based on the formulation described in Section 3.3.2 using the experimentally measured values as listed in Table 3.1. We have also verified a few plotted values for the LPL operation to the empirically obtained values. One can observe that the power consumption increases linearly with the number of channels, while the power consumption increases exponentially with the sampling period.

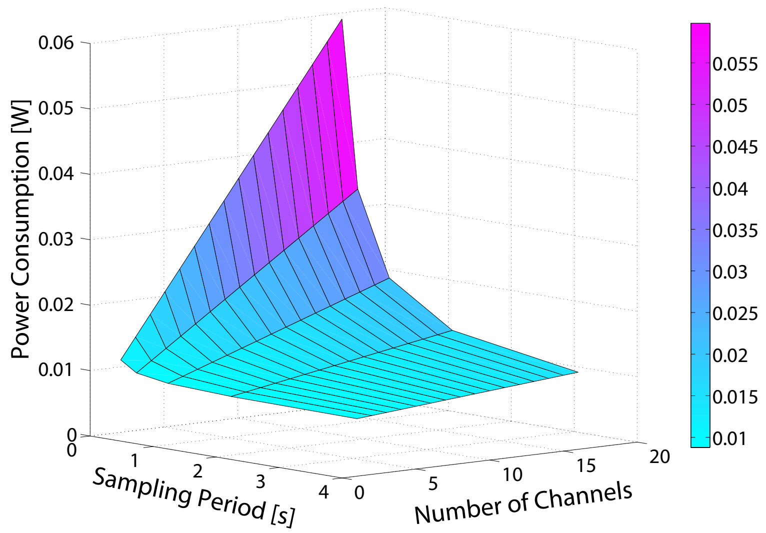


Figure 3.7: Relationship of the power consumption of SA-MAC with the number of channels in the pool and the sampling period.

Interference Avoidance

We generated cyclic interference of 22 MHz bandwidth using AirHorn Dual-Band signal generator. The center frequency of the periodic sweeping interferer was adjusted with a gap of 5 MHz thereby covering the whole spectrum used by TelosB nodes. A node using SA-MAC was programmed to generate traffic bursts periodically. Channel selection was carried out dynamically every time before packet transmissions. In this experiment, the channel pool was initialized with IEEE 802.15.4 channels 13,17,19 and 23 on a node. It can be seen from the spectrogram in Figure 3.8 that SA-MAC node is able to hop away to interference free channels when the cyclic interference comes into the shared frequency channel.

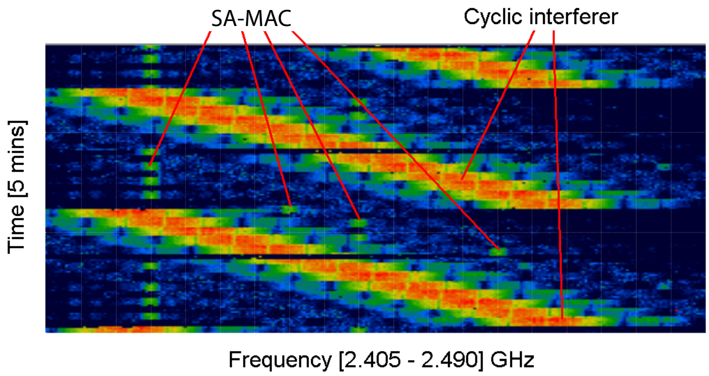


Figure 3.8: Spectrogram showing that SA-MAC allows a TelosB node to dynamically hop away to a free channel and avoid interference.

Channel Convergence

We connected the nodes to a PC over the USB interface to obtain the channel weights at each sensing cycle. At each update interval, the nodes pushed their channel weights over the USB interface. Figure 3.9 shows the convergence of channels and the changing channel weights over time for the two nodes running SA-MAC. The channel pool was initialized with eight IEEE 802.15.4 channels ranging from 11 to 18. The targeted converged pool size was set to two in this experiment. At each sampling instant, nodes update their channel weights in a distributed way. The channel weights are shown on the y-axis while the sampling instants are shown on the x-axis. The pixel intensity represents the channel weight. A darker pixel shows a lower weight and vice versa. It can be observed from the figure that as time passes both channel weights follow an increase in the value on the two nodes represented through lighter pixel intensities. Depending upon the subjected interference and the communication pattern, the two nodes tend to remove the channels with lower weights. Using the distributed algorithm, both the nodes find channels 15 and 16 to be good and converge to these channels in just a few iterations. The convergence and stability in channel selection can be observed from consecutive light grey pixels. On the contrary, as the weight of a channel degrades,

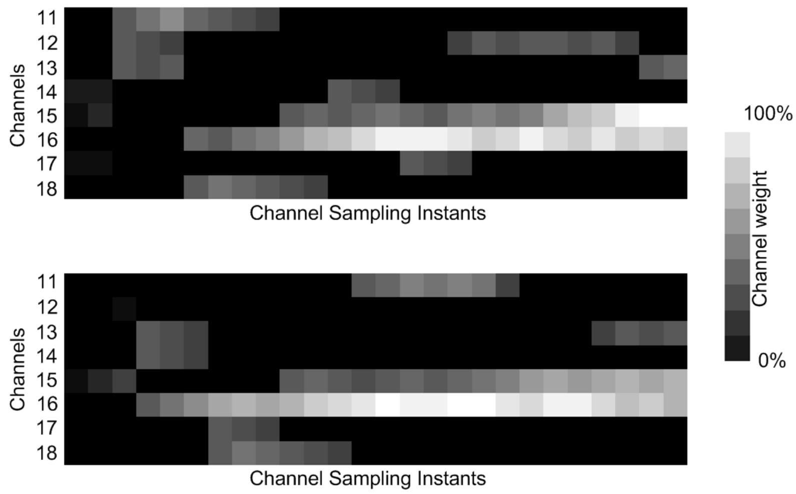


Figure 3.9: Channel weight adjustments on two different TelosB nodes running SA-MAC. The two nodes are shown to converge to channels 15 and 16.

it is removed from the pool. This is represented through a trend of pixels going from lighter intensity to darker and eventually black when the channel has been removed from the pool. New channels are included in order to maintain high quality channels in the pool. The differences in the grey-scale channel weight representation of the two nodes are because the nodes sense the channels asynchronous to each other and therefore measure different spatio-temporal interference characteristics have different channel histories and view different data communication patterns over time. However, it is worth noting that the two nodes converge to the same channel pool.

Throughput and Successful Packet Delivery Ratio Comparisons

A wireless interferer affects the successful packet delivery ratio and the achieved throughput of a MAC protocol. An interferer leads to packet collisions, lower SNR, and the need for retransmissions. In CSMA/CA based protocols, the channel unavailability causes congestion backoffs. Thus, besides degradation in successful delivery ratio and throughput, interference also causes increased latency and energy wastage. An interferer is characterized both temporally as well as spatially. In the following, we will describe our experiments for successful packet delivery ratio and throughput of SA-MAC in the presence of interferers of different strengths. We have varied the transmit power levels at the nodes and the position of the interferers in order to empirically study the effects of symmetrical as well as asymmetrical interference. We have also conducted similar studies on B-MAC+ protocol running on the same TelosB platform to observe the coexistence behaviour of a single channel protocol. In order to ensure fairness in our experiments, we used the same sampling period, data packet size, radio transmit power and clear channel assessment threshold value on the two protocols. Figure 3.10 shows the setup for these experiments. We used Agilent E4438C

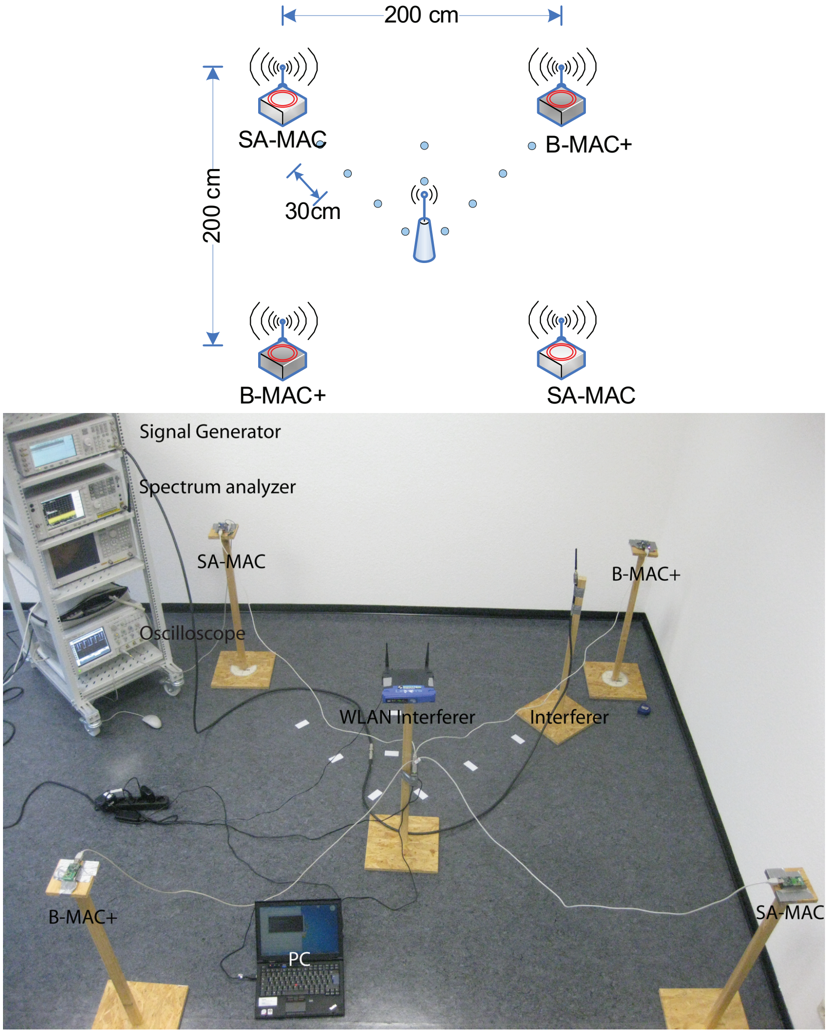
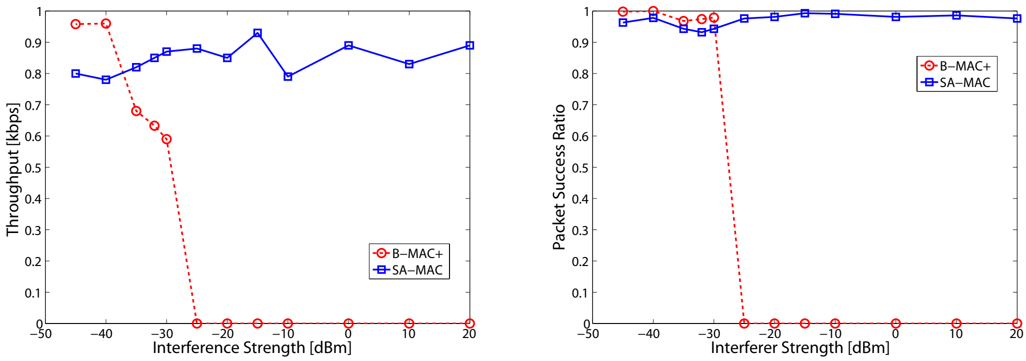


Figure 3.10: Experimental setup for studying the coexistence behaviour and performance of B-MAC+ and SA-MAC in the presence of interference. The parameters varied during the experiments include the transmit power on the nodes, the spatial position of the interferer and its strength.

signal generator and a WLAN access point as controlled interferer sources. We altered the transmit power levels and the location of the interferers to study the behaviour of symmetrical as well as asymmetrical interference on the performance characteristics. We placed TelosB nodes on raised wooden foundations to avoid the grounding effects of the antenna radiation pattern. The nodes were connected to a PC over the USB interface, which was used for setting parameters and collecting experimental data.

In one of the experiments, we placed an interferer of variable strength in the middle of a square grid as shown in Figure 3.10 and measured the throughput and packet successful ratio of SA-MAC and B-MAC+. Both nodes of each of the two MAC protocols transmits a data of 100 bytes every 500 ms with a radio transmit power of



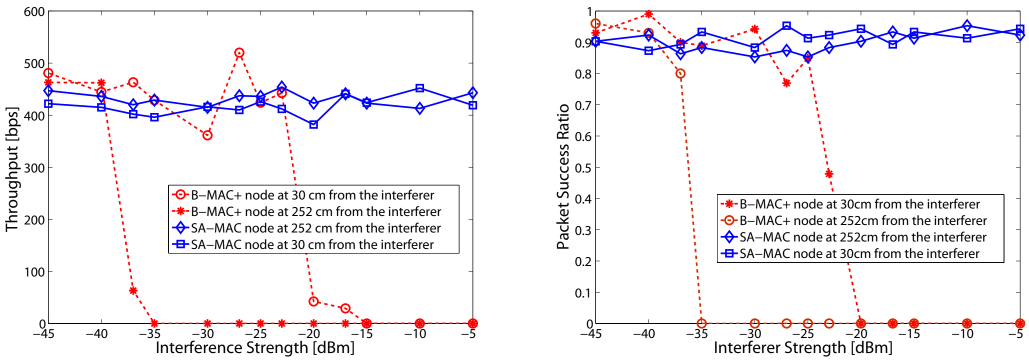
(a) Average throughput. (b) Average successful packet delivery ratio.

Figure 3.11: (a) Aggregate throughput and (b) successful packet delivery ratio achieved for B-MAC+ and SA-MAC in the presence of an interferer at equidistant from the nodes. All the nodes transmit packets of size 100 bytes at the rate of 0.5 packets/second and operate at a sampling period of 1 s with radio transmit power of 0 dBm. The two B-MAC+ and SA-MAC nodes are placed at a distance of 282 cm from each other.

0 dBm. B-MAC+ channel overlaps with the wide-band interferer while one of the four channels used by SA-MAC is non-overlapping with the interferer.

It can be seen from Figure 3.11 that the aggregate throughput of SA-MAC remains relatively stable over the whole range of interferer strengths. B-MAC+ on the contrary, shows a gradual degradation until -30 dBm followed by a sharp drop in the throughput. The gradual drop is because of the less reliable carrier sensing of the channel which caused the node to go to sleep without transmitting a packet. At higher interference strengths, B-MAC+ nodes find the channel to be busy and do not transmit at all. The packet success ratio follows only a sharp drop at -30 dBm for B-MAC+. The reason is that when a B-MAC+ node transmits a packet at 0 dBm, it is received by the other node with high reliability. Owing to the CSMA/CA nature of the protocol, at higher interference strengths when the channel is completely jammed, no transmission takes place and hence no energy is wasted in packet collisions. In contrast to B-MAC+, SA-MAC shows a high successful packet delivery ratio throughout because of its ability to dynamically avoid interfering channels. The protocol is based on the preamble sampling principle and the channel weight for each individual channel in the pool is updated each time when the LPL operation is performed. The channel which is found to be currently free and has the highest weight is selected for data transmission. Since the channel selection is based on per data transmission basis, SA-MAC is able to effectively deal with random interference.

In order to study the effect of asymmetrical and spatial interference, we placed an interferer at a distance of 30 cm from B-MAC+ and SA-MAC nodes. The distance of the other nodes from the interferer was kept 252 cm in this case. We repeated the ex-



(a) Average throughput. (b) Average successful packet delivery ratio.

Figure 3.12: Throughput and successful packet delivery ratio achieved for B-MAC+ and SA-MAC in the presence of an interferer at asymmetrical distances from the nodes. All the nodes use a rate of 0.5 packet/second and operated at the sampling period of 1 s with a radio transmit power of -15 dBm. The two B-MAC+ and SA-MAC nodes are placed at a distance of 282 cm from each other.

periments as described above with a 100 bytes long packet transmitted every 500 ms. Figure 3.12a shows the throughput of SA-MAC and B-MAC+. It can be observed that SA-MAC has a stable throughput even in the presence of the asymmetric interference. B-MAC+ nodes placed at distances of 30 cm and 252 cm, on the contrary, show a sharp drop in the throughput at -37 dBm and -20 dBm, respectively. This is because the node closer to the interferer detects the channel busy at much lower interference strength than the node which is further away and hence the node tends to skip packet transmissions. We may observe a very low throughput between -20 dBm and -15 dBm. This is because the transmitter further away from the interferer detects the channel to be free during this range and transmits packets which get garbled due to the interference and are not correctly recovered at the receiver. In SA-MAC, the channel weighting algorithm automatically assigns lower weights to the interference prone channels which helps it to achieve a high throughput and packet success ratio. Figure 3.12b shows that SA-MAC gives a stable packet delivery ratio. Similar to the throughput trends, B-MAC+ experiences a sharp degradation in the packet success ratio. One should note that between -25 dBm and -15 dBm, a B-MAC+ node placed at 252 cm from the interferer transmits more packets than those successfully delivered.

Successful Packet Delivery Ratio in Challenging Multihop Scenarios

In order to study the effects of multihop communication in a network with non-uniform and irregular interference from different sources, we have designed the experimental setup as shown in Figure 3.13. We put five SA-MAC nodes in a line and adjusted the transmit power level at each node to be -15 dBm so that only the adjacent node(s) were in the wireless range of a particular node. Two interferers of strength -25 dBm were

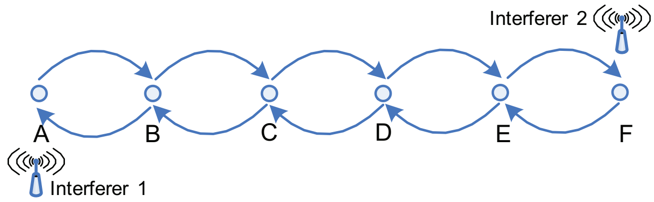


Figure 3.13: Experimental setup for measuring the packet success ratio of SA-MAC over multiple hops when using different number of channels in the pool and when the network is subjected to different channel interference across the two ends.

placed at the opposite ends of the network. The frequency and bandwidth of interferers were chosen so that they overlap with IEEE 802.15.4 channels, 11 to 13 and 16 to 18, respectively. The SA-MAC channel pool was initialized with consecutive channels from 11 to 18 and the MAC was allowed to choose the contracted channel pool size of 6, 4, 2 and 1. We developed an application so that a packet of size 50 bytes generated from node A makes multiple hops (order of 2) and comes back to A. In the 2-hop case, the packet generated from A goes to B and then returns back to A. In the 4-hop case, the packet from A goes to B and then to C. The packet comes back to node A from C via node B. In this way multihop scenarios of 2, 4, 6, 8, and 10 hops are obtained in the network.

It can be observed from Figure 3.14 that as the number of hops increases, the successful packet delivery ratio goes down and as the number of channels in the pool decreases, there is a decreasing trend in the packet delivery ratio. It is clear that in this challenging scenario for channel pool size of 1 and 2, the network is partitioned and the far end nodes converge to channels which are not common with the other nodes. This causes the success ratio to drastically fall in the cases of channel pool size of 1 and 2 beyond 4 hops. In order to deal with this potential problem, a bit encoded channel-map information (consisting of $1-2$ bytes) which is embedded inside the preamble frames so that during the convergence phase, the nodes may also make use of the two hop neighbourhood information. We will describe this scheme and the corresponding performance evaluation in the following section.

3.3.4 Cooperative Channel Sensing

When the channel selection algorithm is completely distributed and nodes independently use only the local spectrum sensing for making decisions on the channel pool selection, geographically asymmetric interferers may split a multihop network. Therefore, we have devised a scheme so that nodes exchange their local channel information for sharing spectrum occupancy characteristics beyond local sensing. The channel sensing characteristics are compressed in the form of a bit vector, where a bit corresponding to each channel indicates whether a particular channel has a weight higher than τ_2. The use of a bit vector keeps the overhead to a minimal. If a particular channel

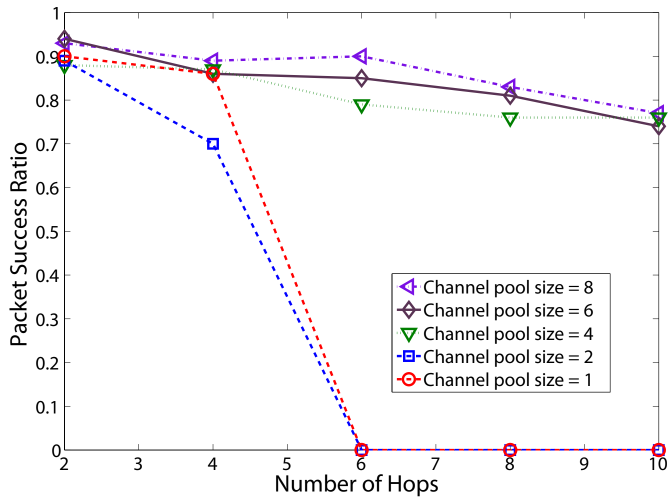


Figure 3.14: Packet success ratio of SA-MAC over multiple hops in the presence of non-uniform interference at the two end of the networks. The transmit power of the nodes is set to be -15 dBm and packet size used is 50 bytes.

on a neighbouring node is indicated to have a weight higher than τ_2, the node increases its own weight corresponding to the channel by one. Otherwise, it decreases its weight by one. By exchanging the neighbourhood channel map information, the nodes can have a view of the network wide spectral characteristics.

The channel selection algorithm is illustrated in Figure 3.15 where two channels on a node are shown to readjust their weights at every sampling instant when the carrier is sensed. As the weight falls to the lower threshold τ_1, it is deleted from the pool. Every new channel is put into the pool with a weight initialized to τ_2.

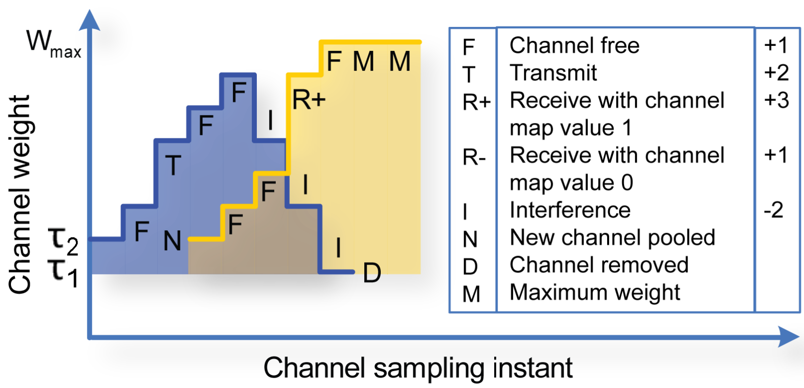


Figure 3.15: Illustration of the weight assignment scheme in SA-MAC for two different channels on a particular node.

Experimental Performance Evaluation

We have carried out the performance evaluation of the channel selection scheme in the presence of different types of interference patterns. We have studied the effects of a 1) permanent interferer, 2) a cyclic interferer, which sequentially switches channels, and 3) an interferer, which hops to different channels randomly. We used Rice University's Wireless Open Access Research Platform (WARP) board [15] for generating different types of interference patterns. We have also conducted comparative analysis of the channel selection in SA-MAC to a random channel selection on the same hardware platform (TelosB) with identical radio and MAC parameters. In order avoid the effects of any astray wireless interference, we monitored all of our experiments using Agilent E4440A spectrum analyzer.

Performance in a Multihop Network

In a multihop asynchronous network, nodes sense the spatio-temporal characteristics of interference differently. In order to study the behaviour of the channel convergence, the achieved throughput and the packet success ratio, we used the experimental setup as illustrated in Figure 3.16. A snapshot of the setup is shown in Figure 3.17.

We placed five nodes (labeled as A to E) in a line topology. The radio transmit power levels at the nodes was adjusted to form a four hop network. An interferer (in our case a WARP board) was placed at one end of the network. The nodes were connected to a PC for gathering statistics. The following packet forwarding scheme was devised. When Node A sends a data packet, it is received by Node B. Node B then forwards this packet to node C. The intermediate nodes carry on forwarding the packet till it reaches Node E. Further on, the packet hops back from Node E back to Node A on the reverse path. An interferer was placed in the vicinity of Node E whose transmit power was adjusted so that it only affected the nodes E, D and C while nodes B and A could still communicate in the interfered channels.

Only Node A generated data traffic with a packet size of 50 bytes. Each packet transmitted from Node A contained a unique identifier. All the nodes recorded their local packet transmit and receive counters. Figure 3.18 shows the packet success ratio obtained when the channel pool size is contracted to 2 and 4. In this experiment, the channel pool is initialized to 8 and only a permanent interferer was considered. It can

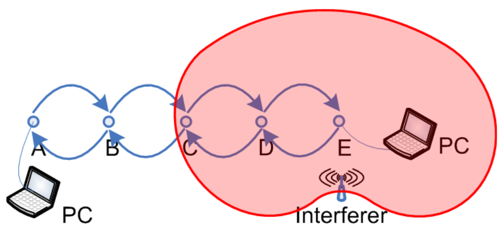


Figure 3.16: Experimental setup for the performance evaluation of SA-MAC in asymmetrically interfering multihop network.

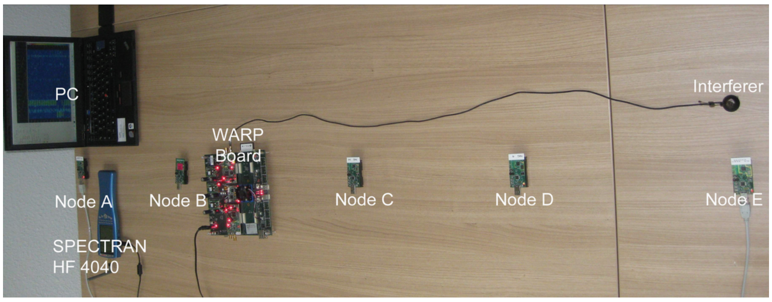


Figure 3.17: Snapshot of the experimental setup for studying the channel selection and successful packet delivery ratio of SA-MAC in a multihop network.

be seen from the figure, that the packet success ratio goes down as the number of hops increases. However, it remains above 75 % over 8 hops. When enough free channels are available, the channel pool size shows little effect on the overall packet success ratio.

In order to observe the exercised channel for data communication, we have gathered the information on the exercised channel at Node A. This experiment was performed to observe the effects of packet exchange and sharing channel map information on the channel weights. Figure 3.19 shows the channels used by Node A for a transmission of 50 data packets when the interferer is out of the direct sensing range of Node A. The permanent interferer occupies IEEE 802.15.4 channels 11, 12 and 13. It can be observed from the figure that node A converges to channel 14 and 15 for the channel contraction limit of 2 and 4, respectively. Although, the channel used by node A is quite stable, yet it is worth noting that the interfering channel 13 used for the 3^{rd}

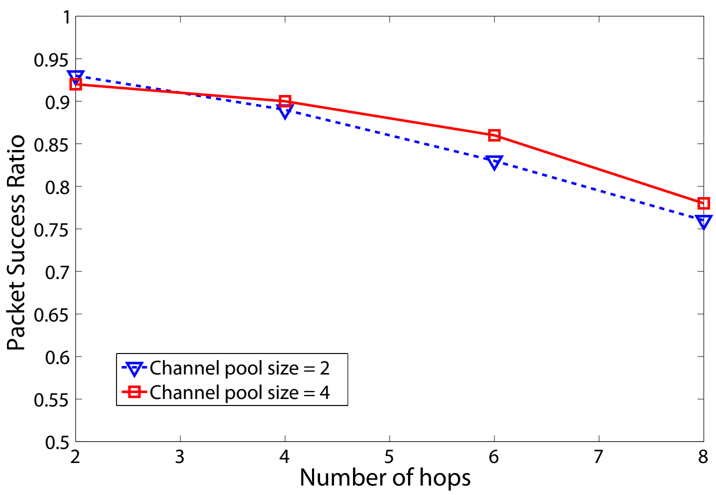


Figure 3.18: End-to-end packet success ratio of SA-MAC in a multihop network.

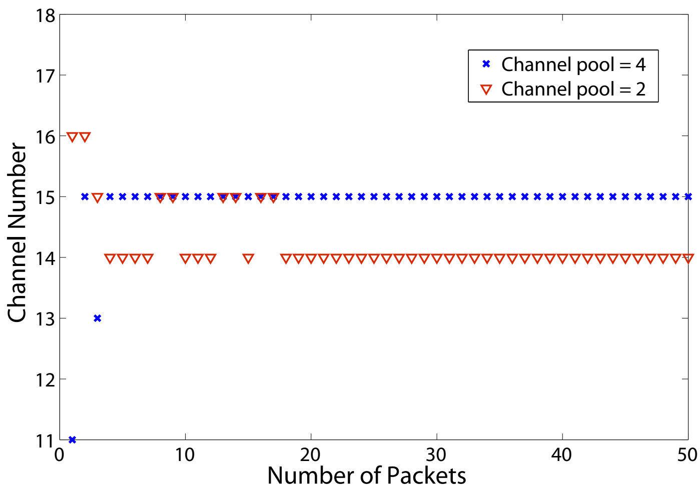


Figure 3.19: Channels used by Node A in the presence of a permanent interferer in the network which is out of its sensing range. Channel pool size of 2 and 4 for SA-MAC are considered.

packet transmission in the case of channel pool size of 4, is not further used by Node A because of the channel state information that it gathers from its neighbouring nodes. This experiment also suggests that for statically interfering environments, higher channel diversity (larger number of channels in the pool) is not necessary rather identifying the interfering channels and blacklisting them is enough.

The effects of the neighbourhood information for the selection of channels is more obvious if the hidden interferer is made to switch different channels. Figure 3.20 shows the channels used by node A for packet transmission when the channel contraction size is set to 4. It can be observed from the figure that Node A hops to different channels in the case of cyclic interferer. The channel hopping at node A indicates that data exchange and channel map information has a direct impact on the channel selection strategy at Node A. The exact behaviour for the channel selection is not straightforward to determine since our algorithm is heuristic. The channel weight adjustments depend on the channel interference, data exchange rates and the sampling intervals. If the rate of data exchange among nodes is low, the channel selection mainly relies on the local sensing. If the network has multiple hidden interferers, the channel weights fluctuates but the least interfering channels tend to accumulate higher weights.

Comparison to Random Channel Selection

We have conducted experiments to compare the achieved throughput and the successful packet delivery ratio of our heuristics based channel selection scheme to a random channel selection scheme in exactly the same spectral conditions. We setup a pair of nodes using the channel selection strategy of SA-MAC and a pair of nodes with random

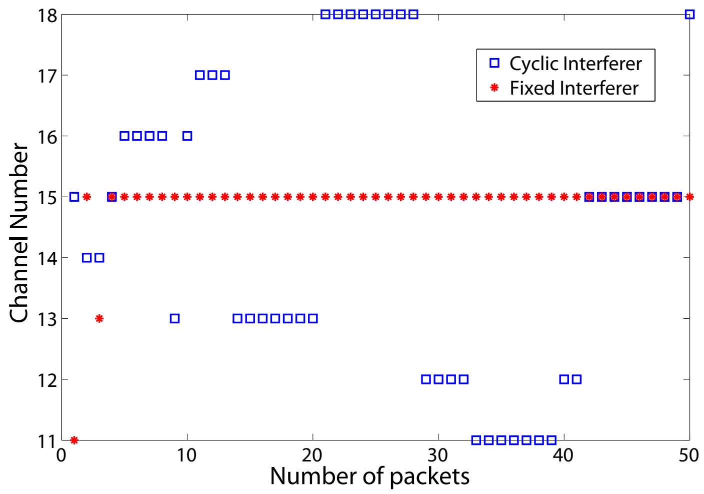


Figure 3.20: Channels used by Node A in the presence of a permanent and cyclic interferer in a multihop network. The interferer is out of the direct sensing range of Node A. The figure indicates that channel map information in SA-MAC allows a node to benefit from the spectral characteristics beyond its direct sensing range.

channel selection. We placed the two pairs of nodes equidistantly on a 2 m×2 m grid from an interferer of strength 0 dBm. The two pairs of nodes performed periodic sensing operation at 1 s and generated a periodic traffic of data size 100 bytes every 500 ms using a transmit power of 0 dBm. All the radio and MAC parameters were identically set on the two pairs of nodes. We used an initial pool size of 8 channels and forced the nodes not to contract the channel pool. In the case of random selection, the nodes chose a channel randomly with a uniform distribution out of this initial pool. Different types of interference patterns were generated through a WARP board. A permanent interferer was created by fixing the channel on the WARP board. A cyclic interference was generated by allowing the WARP board to sequentially switch channels after a regular interval of 30 s and finally a random interference was generated by hopping to channels randomly after an interval of 30 s. We measured the average success packet delivery ratio and the achieved throughput over a duration of 30 min while subjected to different interference conditions for the two channel selection schemes. Table 3.3 summarizes the results. It is clear that the heuristics algorithm for channel selection in SA-MAC outperformed random channel selection method in all the three different interference conditions. The results show that SA-MAC efficiently utilizes the available spectrum opportunities. A random channel selection has a lower likelihood of finding a free channel. Owing to the CSMA nature of the MAC protocol, when a channel was found busy (interfering), packet transmission was not attempted. This resulted in a significantly lower overall throughput. However, the successful packet delivery ratio did not suffer substantial degradation.

Table 3.3: Packet success ratio and throughput comparison of SA-MAC with random channel selection scheme.

Interferer	Throughput [kbps]		Success Ratio [%]	
	SA-MAC	Random Ch. Selection	SA-MAC	Random Ch. Selection
Cyclic	0.82	0.67	94	83
Random	0.76	0.58	91	87
Permanent	0.82	0.64	95	84

3.3.5 Discussion and Conclusions

We have described in detail the design, implementation and performance evaluation of SA-MAC protocol. Spectrum agility is an important feature for low power embedded wireless networks in order to coexist in the unavoidable crowded spectrum. SA-MAC adheres quite well to the design considerations described in Section 3.1. We advocate the idea of combining spectrum sensing operation with periodic duty cycling in low-power MAC protocols. Since the sensing operation is local to a node, it bears high scalability properties and does not require a centralized coordinating infrastructure. Furthermore, a node can learn about the spatial spectrum occupancies beyond the locally sensed radio characteristics using the channel map information, which is piggybacked in the transmitted packets. The channel selection is performed dynamically before using the medium and a node updates the channel weights in every polling operation. Therefore, the scheme is able to effectively cope with node network dynamics. SA-MAC protocol is able to work reliably with high degree of packet success ratio even in the presence of heavy interference as confirmed by our experimental results. On the contrary, contemporary solutions running on the same platform without any spectrum agility features fail to achieve a stable throughput in interfering channel conditions. SA-MAC combines various preamble optimization techniques from different single channel preamble sampling protocols in order to achieve energy efficiency. We have also presented an analytical model of the protocol and derived an expression for the optimum energy consumption in a particular network at a given amount of traffic volume. The empirical results for the minimum energy consumption obtained through our prototype implementation adheres to the analytical expression. We have conducted comparative studies in terms of throughput and packet success ratio with a random channel selection method in the presence of permanent, cyclic and random interferers. Empirical results show that SA-MAC achieves high performance gains over random channel selection method. To the best of our knowledge, our effort is the first attempt to design, implement and evaluate a spectrum agile MAC protocol for low power embedded networks, which allows nodes to dynamically find interference free channels in unregulated crowded frequency bands and achieves reliable communication.

3.4 SPECTRUM AGILE MAC FOR COGNITIVE RADIO NETWORKS

The use of spectrum opportunities, so called white spaces, can often require stochastic approaches due to difficulty in predicting their appearance. Infrastructure based coordinated access techniques are not a viable option for all the applications and spectrum bands. We have developed a decentralized Cognitive MAC (CogMAC) protocol based on multi-channel preamble reservation scheme. The protocol dynamically selects an available communication channel using a distributed channel selection scheme and allows nodes to be completely asynchronous to each other. The main goal of CogMAC is to provide a simple learning based distributed MAC protocol that is able to choose a new channel in a weighted manner if they have to vacate an existing channel for a PU or to avoid interference from another network. CogMAC shares the distributed channel selection scheme with SA-MAC. However, CogMAC is targeted for relatively less resource constrained ad hoc networks requiring DSA capabilities. CogMAC requires only a half-duplex radio interface without needing a CCC. CogMAC protocol is suitable for both licensed and ISM frequency bands, where the protocol adapts its channel selection to the stochastic spectrum occupancy behaviour of the PUs and mitigates the effects of the random interference, respectively.

Studies on the impact of frequency-agility on dynamic spectrum sharing show that radios which are capable of using non-contiguous frequencies give better performance over radios using single frequency for transmission [128]. CogMAC uses a distributed channel selection scheme, which adaptively expands and contracts the number of frequency channels to be used depending upon the interference conditions. The protocol has been implemented on WARP [15] boards and has been evaluated in terms of throughput and packet delivery ratio with respect to different carrier sensing durations and the number of used channels. A vector signal generator was used to model the behaviour of a PU. We believe that our implementation and evaluation provides significant insights into the practical aspects of cognitive radio MAC protocols. This has a particular value as majority of the current designs remain at a theoretical and simulation stage.

3.4.1 Protocol Design

CogMAC design is targeted for infrastructureless environments and uses a distributed channel selection strategy, which allows it to handle network mobility in an effective manner. The MAC design aims at solving two issues at once. First, how to sense possible channels without introducing undue latency and sense busy channels with active PUs. Second, avoiding overloading the same channel with SUs. The protocol uses multi-channel carrier sensing principle where a node scans all the channels in the pool sequentially. The transmitting node ensures that the transmission in the selected channel lasts for long enough duration that the asynchronous receiving nodes detect it when scanning that particular channel. Upon detecting a packet transmission activity, the receiving nodes do not scan subsequent channels and keep on listening to the channel until a data packet is received. In order to engage the channel, the transmitter repeats

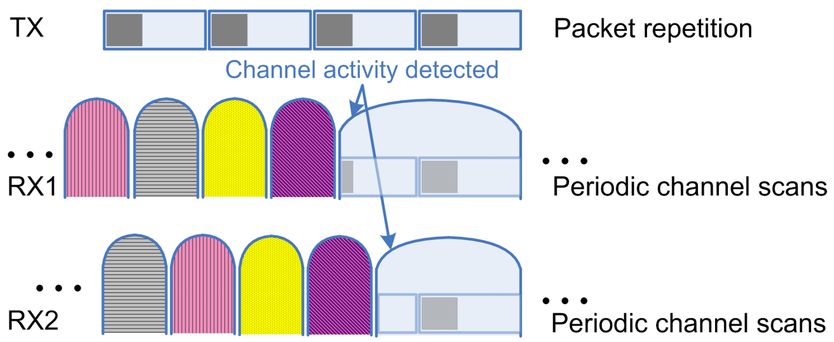


Figure 3.21: Multi-channel sensing and packet transmission in CogMAC. The transmitter repeats a packet back-to-back for the maximum required duration for multi-channel polling. As the two (unsynchronized) receivers detect the transmission activity, they stay in the channel until a complete packet is received.

the data packet back to back as shown in Figure 3.21. The total number of packet repetitions, N_{pkt} required to be sent governs an upper bound, $N_{\text{pkt}} \geq (t_{\text{cs}} + t_{\text{switch}})n_{\text{ch}}/t_{\text{pkt}}$, where t_{cs} is the carrier sensing duration, t_{switch} is the channel switching duration, n_{ch} is the number of channels and t_{pkt} is the time required to send a packet.

A transmitter first scans all the channels in the pool to ensure that there is no other on going packet transmission before attempting to send a packet. This also justifies the receiver(s) for not sensing subsequent channels upon detecting a packet activity. Scanning channels prior to transmission helps in avoiding cases of multiple simultaneous transmissions to the same receiver(s) though there exists possibilities to utilize additional available bandwidth. Since an activity in the medium can also be because of an interferer or a PU, the protocol uses a timeout scheme in order to characterize an interferer. Please note that here, in terms of terminology, we do not distinguish between a PU and an interferer[3]. If channel activity is detected and no valid packet is received within an interval of two maximum sized packet transmissions, the channel is identified as interfering. A timeout duration of two packet transmission interval allows receiving a back to back transmitted packet with any boundary offset. The protocol uses a heuristics based method for channel selection similar to the one used by SA-MAC. In each sensing cycle, all the nodes scan the available channels sequentially for potential spectrum activity. Weights are associated with the channels, which are updated in each sensing cycle based on the type of the activity. If a particular channel is found free and data communication is established, the weight associated with the channel is increased. On the contrary, if a channel is found interfering, its weight is decreased. Channel weight history is also maintained, which helps in identifying interfering channels and blacklisting them. Before a packet transmission, the sender carries out dynamic channel selection in order to opportunistically utilize an available

[3]In this work, we did not take signal classification and feature detection into account. However, without losing generality, such a capability can be straightforwardly included into CogMAC.

spectrum hole. Nodes are able to communicate with each other even without having any prior knowledge about the sensed spectrum characteristics of other nodes in the network. However, nodes do exchange the sensed channel characteristics (compressed in a binary format, called as "Channel maps") inside the packet header. A channel map is essentially a bit encoded information indicating whether a particular channel has a weight higher than a predefined threshold, τ_2 as shown in the pseudo-code below.

Algorithm 3.4.1: CHANNEL SELECTION ALGORITHM()

```
ch.wt ← τ2
for i ← init to max_channel
  do { CARRIERSENSING(ch[i])
       if ch[i] = Free
         then ch[i].wt ← ch[i].wt + 1
       if Interference
         then ch[i].wt ← ch[i].wt − 7
       if Tx or Rx
         then ch[i].wt ← ch[i].wt + 5
       if Rx
         then { for j ← init to max_channel
                  do if channel_map = 1
                    then ch[j].wt ← ch[j].wt + 3
       if ch[i].wt < τ1 and ch_contraction_enabled
         then DELETE(ch[i])
cond ← True
for each ch
  do if ch.wt < τ2
  then cond ← False
if cond = True and CHANNELWTMEDIAN(ch) < 1.5τ1
  then { REFRESH(ch)
         TIMESTAMPCHANNELACTIVITY(ch)
SORT(ch)
comment: Sort all channels in descending order of weights.
```

Channel maps of neighbours allow a node to avoid spatially local interferers by benefiting from their radio environment. Scanning a larger number of channels adds latency to the communication and reduces the throughput as confirmed by our experiments in Section 3.4.3. Therefore, our channel selection scheme tries to maintain a smaller number of channels with higher weights in the pool and delete the blacklisted interfering channels. Also if the quality of channels (expressed through the median channel weight) deteriorates, the channel pool is replenished by replacing the low-weighted channels from the pool. New channels with their weights initialized to the threshold τ_2, are included in the pool. Keeping a channel history prevents adding previously deleted channels in the pool. Precedence is given to channels which have not been blacklisted before and to channels with the oldest blacklist time-stamp. The dynamic

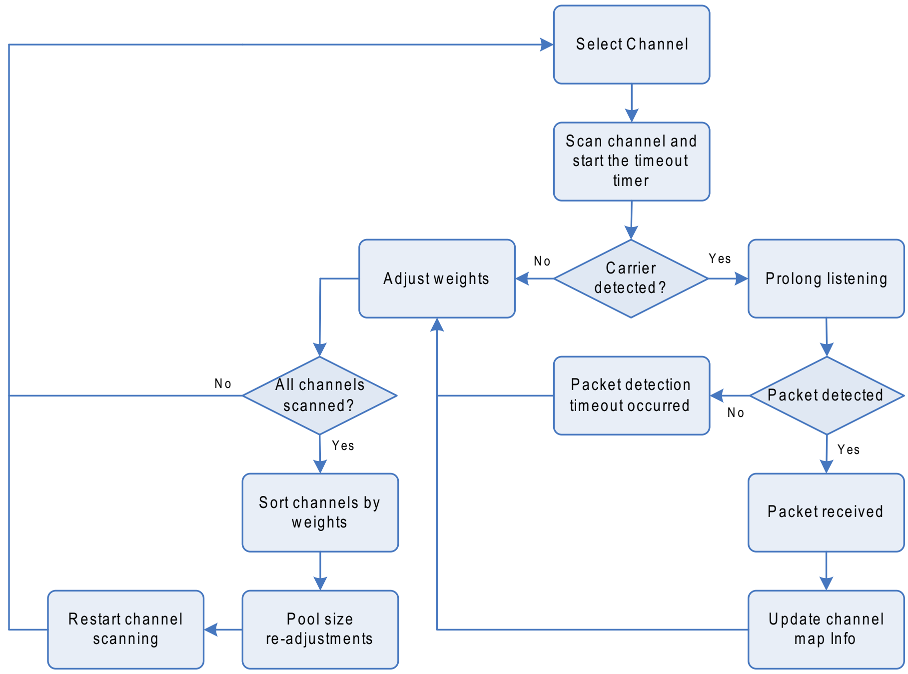


Figure 3.22: Simplified flowchart of the channel selection algorithm in CogMAC.

expansion and contraction of channel pool allows keeping diversity in the channel pool and simultaneously lowers the latency in data communication (cf. Figure 3.30). Owing to the distributed nature of the channel selection algorithm, there is a danger that neighbouring nodes may converge to non-overlapping channel pools. However, our experimental results indicate that this is unlikely to happen in practice due to the exchange of channel maps. Figure 3.22 shows the distributed channel selection algorithm. The values of the hysteresis thresholds, τ_1 and τ_2, are empirically chosen to be 15 and 40, when the ceiling channel weight value is set to be 100. Since the channels are sorted in the descending order of their weights, this scheme inherently allows to scan and use the least interfering channels first.

A transmitting node sends a repeated sequence of packets in the selected channel. The header is encoded with the base rate modulation and contains control information such as the destination address, source address, the modulation scheme for data payload and the channel map. A non-addressed node is also able to gather the spatial spectral characteristics of the transmitting node and other relevant meta-data by overhearing a packet header. In order to efficiently support higher data traffic loads at a node, queued data packets are transmitted back to back without repetition after a repetitive transmission of only the first packet. The first packet repetition serves to implicitly synchronize the receiver(s) and reserve the medium for further packet transmissions by a node. Since our protocol is designed for platforms with a single radio transceiver, while the transmission is underway at a node, it cannot detect if the channel is re-engaged by a PU. In order to prevent causing interference to the PU in

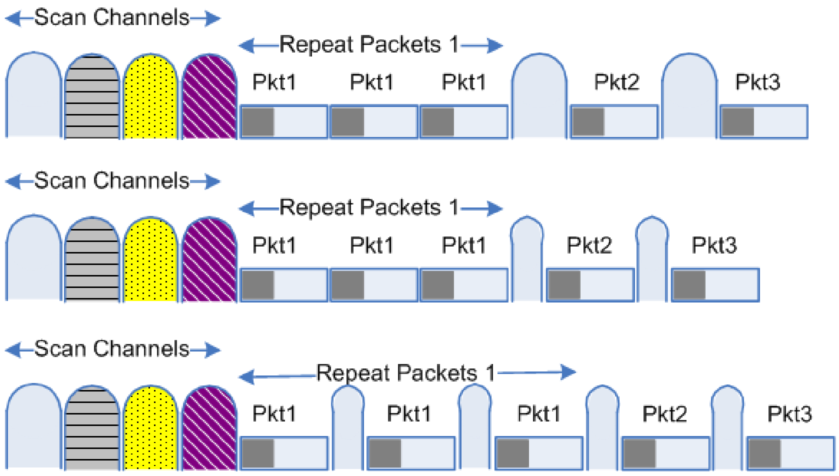


Figure 3.23: Multi-packet transmission schemes for supporting higher traffic volumes in CogMAC. The transmitter checks the availability of the exercised channel between successive packet transmissions.

the case when higher data volumes are to be supported, the transmitter quickly senses the channel between packet transmissions to ensure its availability. The interval after which channel sensing is required during the transmission process depends upon the stringency of the spectrum. This scheme is illustrated in Figure 3.23.

3.4.2 Implementation Details

We have implemented CogMAC protocol on WARP boards [15]. Our MAC implementation uses WARP OFDM Reference Design v.14 and is based on the component based MAC development framework [129]. Table 3.4 lists the parameter values used for our prototype implementation. We have exposed different MAC parameters to the application, like the initial channel pool, contracted channel pool size, the weighting metric thresholds, CCA durations, backoff window sizes, persistency values, etc. which can be tuned according to the application requirements. These parameters have their implications on the MAC performance characteristics as we have shown in a demonstration [130].

Table 3.4: PHY/MAC parameter values on WARP board.

Parameter	*Value*
Max. packet size used ($L_{\text{max\_pkt}}$)	1000 bytes
Max. packet transmission time ($t_{\text{max\_pkt}}$)	1.48 ms
Channel switching interval (t_{switch})	35 μs
Interferer timeout interval ($t_{\text{int\_timeout}}$)	2.96 ms

3.4.3 Performance Evaluation

We have carried out the evaluation of the MAC protocol on a 6-node WARP testbed as shown in Figure 3.24. The WARP boards were connected to Gigabit Ethernet and RS232 interfaces to a PC in order to control the parameter settings and gather results during our experiments. We configured Agilent E4438C Vector Signal Generator to generate different types of interference patterns in order to emulate the behaviour of PUs. We monitored the spectrum occupancy characteristics using Agilent E4440A spectrum analyzer and WiSpy DBx spectrum scanner. In order to measure the timings of different radio parameters, we used Agilent Infiniium DSO8104A Oscilloscope.

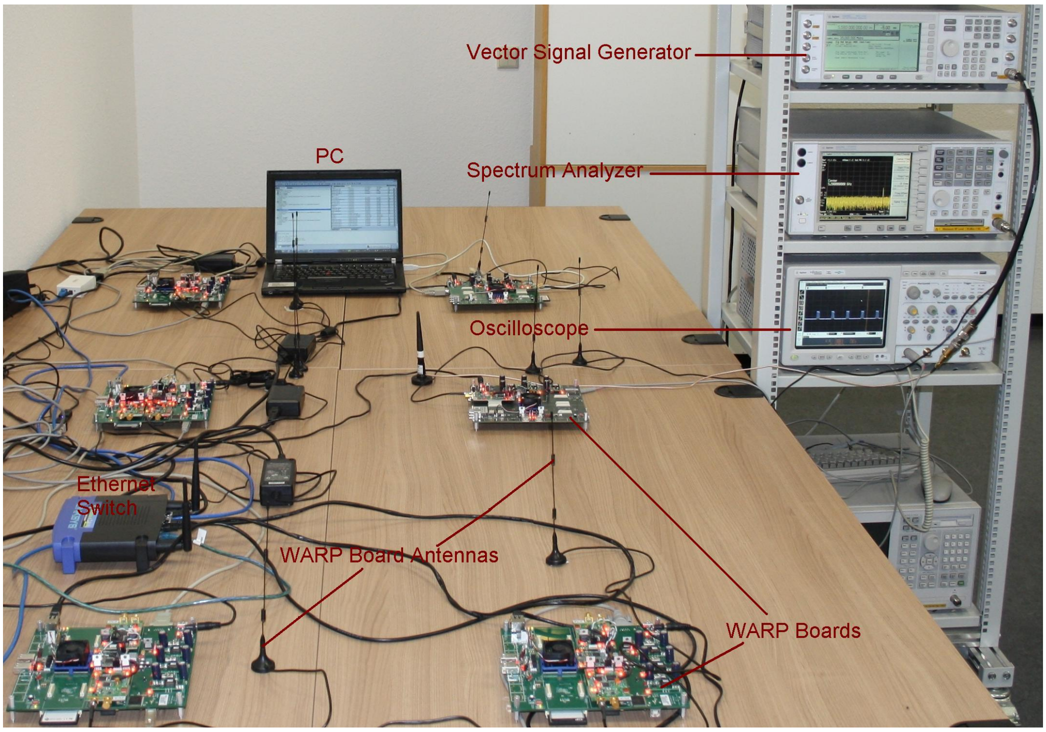


Figure 3.24: Snapshot of the WARP testbed used for the evaluation of CogMAC.

Interference Avoidance

We configured Agilent E4438C signal generator to transmit random interfering signal in different channels for a certain duration using different transmit power levels. WARP boards using CogMAC with 4 channels in the pool were placed in the interference range. As can be seen from the spectrogram in Figure 3.25 that our MAC protocol is always able to detect the interferer and quickly hops away to an available free channel for data communication. The behaviour is very similar to our previous observation (cf. Figure 3.8) in SA-MAC protocol.

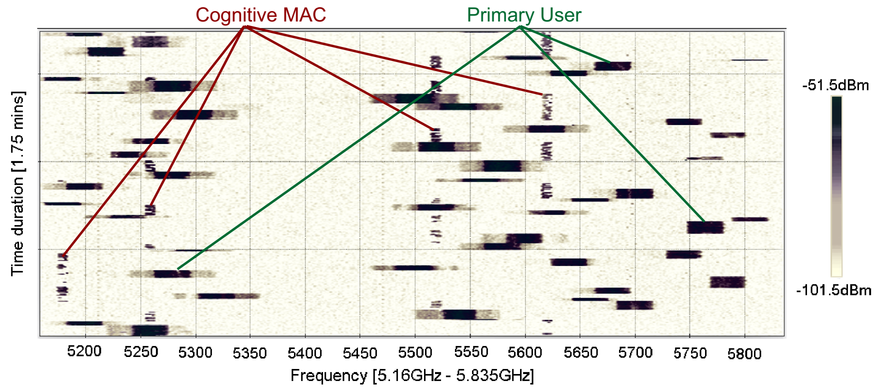


Figure 3.25: Spectrogram showing that the CogMAC protocol is able to dynamically select interference free channels when subjected to random interference patterns emulating a primary user.

Effects of Channel Pool Size and Carrier Sensing Duration

In order to study the effects of the channel pool size on the goodput and the packet success ratios, we used a transmitter and a receiver pair without the presence of any interferer. The transmitting node tries to send packets with payload size of 1000 bytes as fast as possible without using the multipacket transmission scheme. Figure 3.26 shows the achieved normalized goodput with a normalizing factor of 2.393 Mbps. The results show a trade-off between transmission and sensing. It is evident from the figure that the goodput goes down exponentially as the number of channels increases in the pool

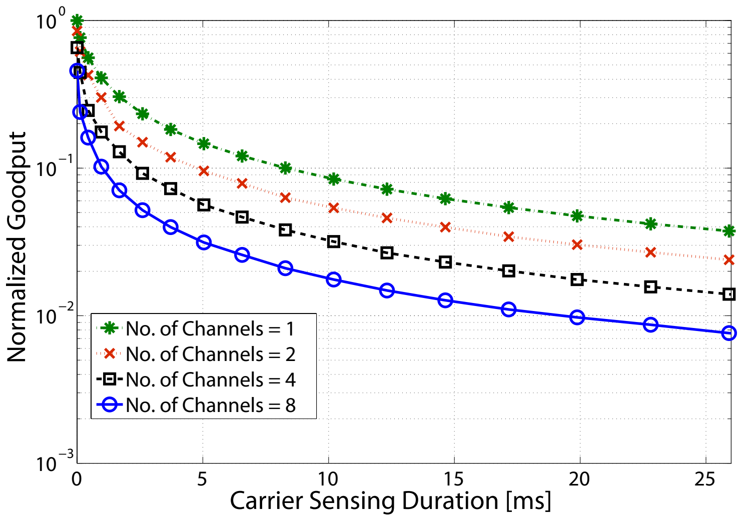


Figure 3.26: Normalized goodput behaviour of CogMAC protocol with respect to the number of channels in the pool and the channel sensing duration.

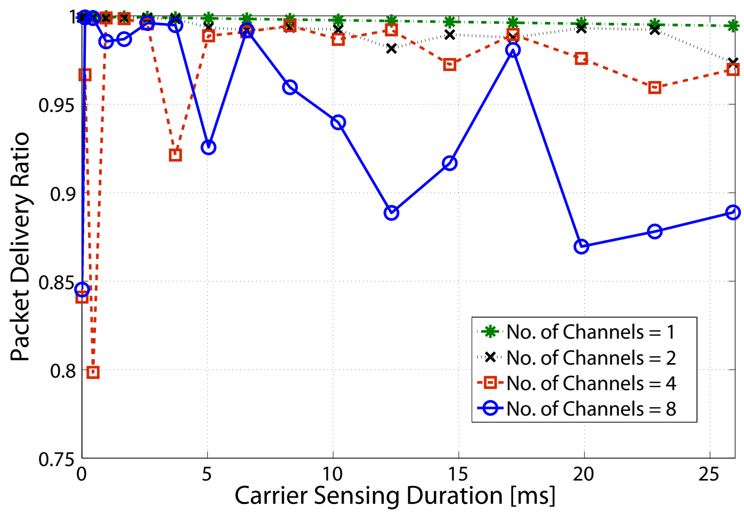


Figure 3.27: Effects of the channel pool size and the channel sensing duration on the successful packet delivery ratio in CogMAC.

and as the CCA duration increases. This is because of a larger overall sensing duration; sensing more channels and a longer per channel scanning interval. The results indicate that if a node can maintain a smaller pool of channels, the total capacity is of course larger. Figure 3.27 shows that the corresponding packet delivery ratio does not suffer any significant change with the increase in the number of channels in the pool and with the CCA duration. These two graphs also serve as the benchmark performance characteristics while studying the goodput and packet delivery ratio in the presence of different interference patterns.

Goodput and Packet Success Ratios

In order to study the empirical goodput characteristics with respect to the number of channels in the pool and the channel assessment duration, we used a WARP transmitter-receiver pair placed in the vicinity of an interferer. The signal generator as an interferer was configured to generate a frequency sweeping signal with 100 ms in the channels used by the WARP board. Figure 3.28 shows the normalized goodput achieved when the transmitter tries to send as many packets as it can with a payload size of 1000 bytes. The normalization factor used is 2.393 Mbps. It can be seen that the goodput has gone down as compared to the interference free case as shown in Figure 3.26. The slight degradation in the goodput is because of the timeouts for interference characterization. Compared to the case with no interference in the medium, the goodput values for the channel pool size of one suffers the most. The channel pool size of two performed better than the rest, which shows that channel diversity is good but keeping a high channel diversity is not necessary in all the cases since it adds to the control overhead. This result corresponds to the earlier theoretical findings [128].

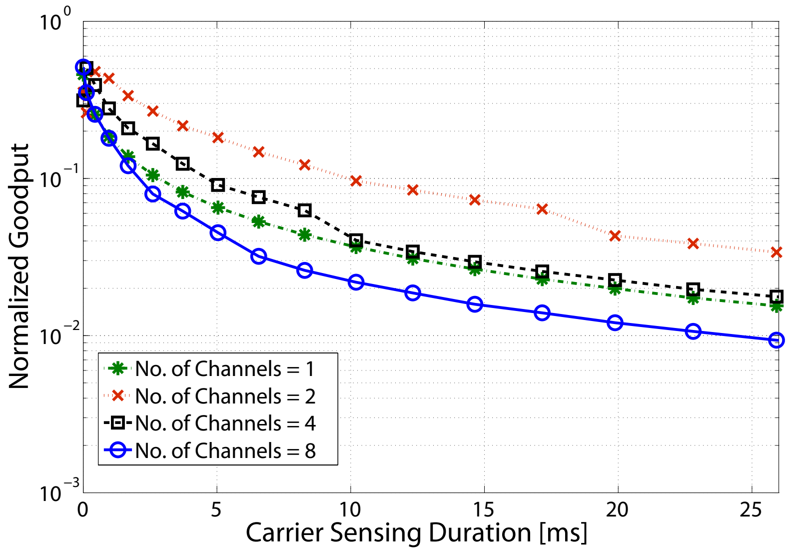


Figure 3.28: Effects of the channel pool size and the channel sensing duration in CogMAC on the goodput in the presence of a sweeping frequency interferer.

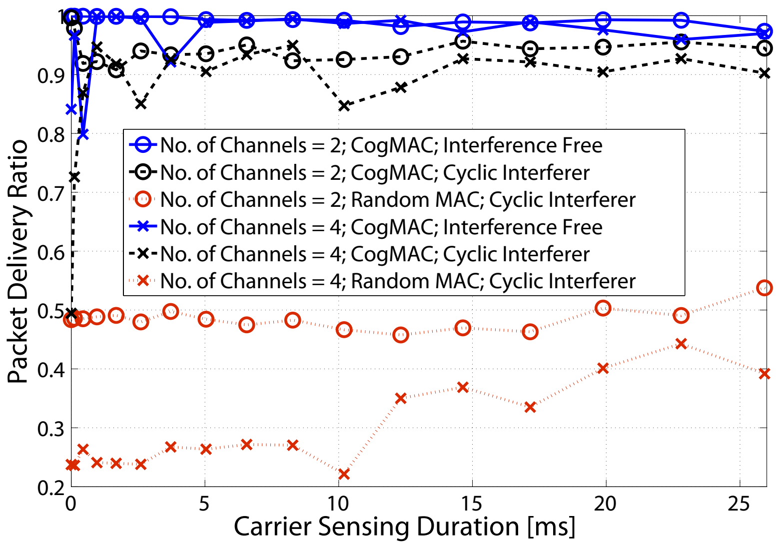


Figure 3.29: Packet delivery ratio comparison of a random channel selection scheme with CogMAC in the presence of a sweeping interferer.

Figure 3.29 shows the packet delivery ratios of a random channel selection scheme and our cognitive MAC protocol in the presence of an interferer that cyclically sweeps frequency in the channels used by the WARP boards with a channel dwell time of 100 ms. It can be observed from the figure that our protocol is able to achieve remarkably high packet delivery ratio as compared to the random channel selection based scheme in the presence of the interferer. At larger CCA durations, the random channel selection scheme improves the delivery ratio since longer channel assessment durations makes sensing more reliable.

Multihop Latency

The multihop latency of the MAC protocol was measured using a linear topology with no interferer in the network and a CCA duration of 0.961 ms. Figure 3.30 shows that the multihop latency follows a linearly increasing behaviour with the number of hops and roughly a linearly increasing trend with the number of channels in the pool.

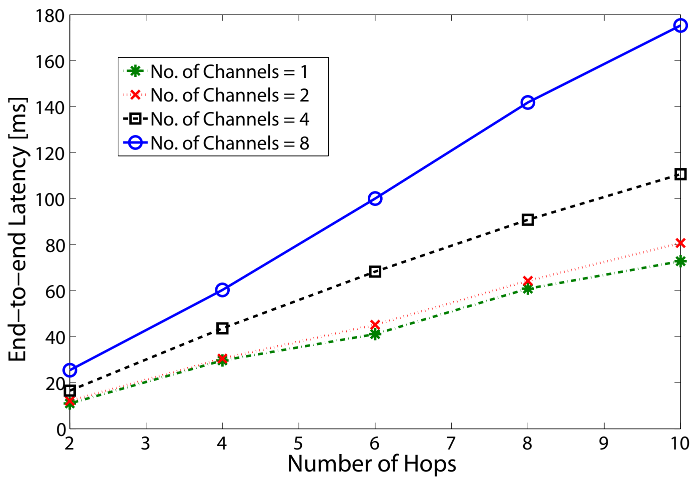


Figure 3.30: End-to-end multihop latency of CogMAC on WARP testbed.

Coexistence of Multiple CogNets

In order to study the coexistence behaviour of two cognitive networks using our MAC protocol in the same interfering environment, we used a transmitter-receiver pair of WARP boards with BPSK encoded preamble and data modulation while the other pair with QPSK encoded preamble and data modulation. In this way, the two pairs interfered each other without being able to communicate. We forced the nodes to use the same set of channels with a pool size of two. We placed the four nodes at a square grid and interferer in the center, jumping randomly in the two channels. We have observed that the two pairs of WARP boards were dynamically able to hop onto the available free channel. Figure 3.31 shows the packet delivery ratios of the two networks. One should note that CogMAC allows symbiotic coexistence of both the networks and is able to provide reliable communication.

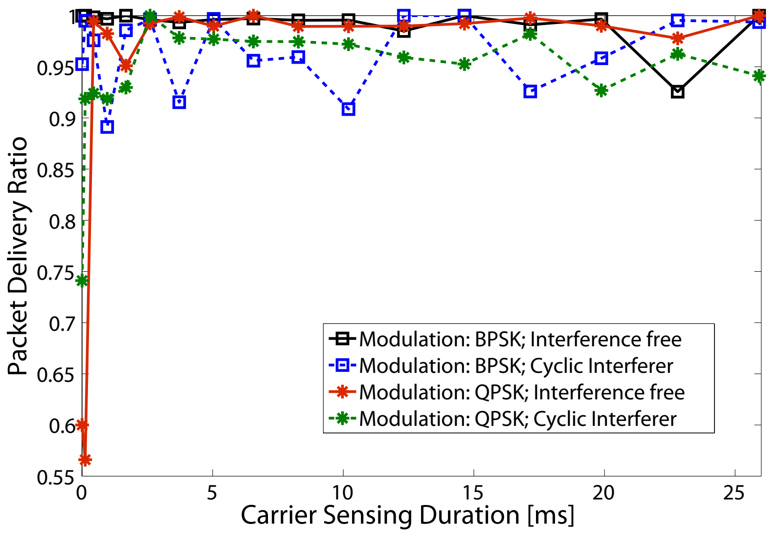


Figure 3.31: Coexistence of different cognitive networks using CogMAC in a shared spectrum.

3.4.4 Discussion and Conclusions

Available spectrum holes can be hard to predict and tight cooperation among different networks sharing the crowded spectrum is often infeasible. In order to address these issues, we have designed and implemented a decentralized cognitive MAC protocol, which allows nodes to communicate reliably even in highly interfering environments. We have described the design rationale and implementation details of CogMAC protocol on WARP boards. CogMAC protocol dynamically selects an interference minimal channel using a distributed channel selection strategy. An available wireless channel can further be utilized in an opportunistic fashion using the multipacket scheme (cf. Figure 3.23 with only a little overhead for potentially subsequent transmissions by keeping the nodes listening for a short duration to the available channel. Experimental performance evaluation conducted on a WARP testbed has shown that CogMAC protocol allows spectral coexistence and is able to deliver packets with high reliability and throughput in interfering environments.

3.5 INTERFERENCE DETECTION

In recent years, numerous resource constrained networks and devices using the popular IEEE 802.15.4 compliant radios at 2.4 GHz have emerged for home automation, building monitoring and health care applications. Since IEEE 802.11b/g based WLANs or commonly known as Wi-Fi[4] networks already exist in the same frequency

[4]Here, we refer to IEEE 802.11b and IEEE 802.11g networks in an informal way as Wi-Fi networks or simply Wi-Fi.

band, it leads to mutual interference. Wi-Fi networks typically use approximately 100 times higher transmit power than IEEE 802.15.4 networks and therefore inflict heavy packet losses on IEEE 802.15.4 communication as has been reported by other research studies [7,8,11]. It has been observed that in a close proximity, IEEE 802.15.4 networks also cause interference to IEEE 802.11b/g networks, which results in significant performance degradation [125]. Both IEEE 802.11b/g and IEEE 802.15.4 networks have been designed without provision for symbiotic coexistence with each other [10,122,131], yet they have actively been deployed in the same environments. A possibility of pre-assigning dedicated channels to each network is not only tedious and expensive in terms of human effort, it is also a waste of spectral resource and is becoming unpractical due to increasing spectrum scarcity [132]. Furthermore, a centralized infrastructure for spectrum management and cooperative spectrum access is impractical for heterogeneous networks and devices with varying and unpredictable spectral needs operating in ad hoc environments in the ISM frequency bands.

Reliable detection of potential interferers is a fundamental requirement for applying an interference mitigation scheme and for enabling spectral coexistence. The faster a system is able to identify interfering channels, the more energy efficient it is. In this context, we have designed a low cost platform named WiSpot [94], which is able to identify Wi-Fi channels in a reliable manner.

Before discussing the design details of WiSpot, related research on the interference signal signature detection in the ISM frequency band is provided in Section 3.5.1. Section 3.5.2 describes the design rationale of WiSpot, its prototype implementation and the algorithmic details. Section 3.5.3 presents the empirical performance evaluation of WiSpot and its comparison to existing approaches.

3.5.1 Existing Approaches on Interference Detection and their Shortcomings

Spectrum sharing and symbiotic coexistence among different networks and devices are becoming inevitable due to the ever increasing number of wireless network applications in our daily life. The 2.4 GHz ISM frequency band is a popular choice for many consumer networks because of the available bandwidth, current radio technology, transmission range, energy consumption aspects and to a greater extent for license free operation. Uncoordinated spectrum access and lack of spectrum sharing features in this frequency band cause mutual interference among different networks which eventually leads to performance degradation [8]. Compared to Wi-Fi, low-power IEEE 802.15.4 networks suffer significantly higher performance losses [7,11]. Infrastructure based coordinated spectrum access such as in IEEE 802.22 networks [100] is not a viable option for heterogeneous networks with often user-deployed consumer devices operating in ISM frequency bands.

In order to minimize wireless interference, many approaches enabling dynamic spectrum access have been proposed by the research community during the past few years. Simulation results show that graph coloring schemes such as [120, 121, 133] help in mitigating interference. Interference aware medium access procedures such as [10, 122, 125, 131] have been designed to ensure communication reliability by selecting interference minimal channels. Huang *et al.* have devised a Pareto model to characterize white spaces in Wi-Fi networks based on the statistical analysis of white space empirical data [134]. They proposed a new ZigBee control protocol to exploit the expected frame collision probability and the channel utilization ratio. While most of the IEEE 802.15.4 and IEEE 802.11 network deployments are carried out indoors in unstructured and ad hoc topologies, due to fast fading and multi-path effects, the resulting channel propagation conditions are complicated and hard to predict. These lead to complex time-varying distributions for the expected channel occupancies in contrast to rather simple channel occupancy distributions [134]. In [126], the authors proposed a model to allow co-existence between IEEE 802.15.4 and IEEE 802.11b/g networks in different operating ranges. However, this scheme requires timing modification of the two protocols and does not completely solve the problem of mutual interference. It is claimed that IEEE 802.15.4 compliant CC2420 [42] radio has good properties for alternate and adjacent channel rejection thereby allowing it to coexist with WLAN, Bluetooth and other ZigBee interfering networks [127]. However, these measurements do not take into account out-of-band leakages and MAC behaviours and therefore, real deployments show a performance loss [7, 11, 126].

Liang *et al.* [125] have analyzed interference patterns between IEEE 802.15.4 and Wi-Fi networks at a bit-level granularity. They have noticed that a significantly high packet loss ratio in IEEE 802.15.4 transmission is due to the corruption of header bytes while the rest of the packet remains uncorrupted. Based on this observation, they have devised a scheme of repeating headers back to back in the frame and using Reed Solomon encoding based FEC scheme to mitigate Wi-Fi interference. While an FEC based approach can certainly reduce the packet losses in wireless interference, dynamic selection of less interfering channels can lead to higher performance gains without imparting extra channel coding overhead [5, 92]. Accurate and reliable detection of interfering channels is rudimentary to any Wi-Fi interference mitigation strategy. Yücek and Arslan survey the various techniques used for spectrum sensing and classification of signals [135]. With the current low-power radio technology, energy detection is the most practical and viable option. Other methods such as spectrum cyclostationarity detection and matched filter based detection [135] are too complex to be implemented on resource constrained nodes. Chowdhury and Akyildiz apply spectral mask fitting of the signal strength measurements to WLAN transmissions [136]. This scheme has limitations in the case of out of band signal leakages and in the case of overlapped transmissions.

Zhou *et al.* have developed ZiFi [137], which utilizes an IEEE 802.15.4 radio to ascertain the existence of IEEE 802.11b/g Access Points (APs) generating periodic beacons. It uses a Common Multiple Folding (CMF) algorithm for detecting the RSSI of the periodic interfering signal. The basic idea of CMF is to search for a periodic signal (with period P) within a series of RSSI samples. The series is divided into smaller sequences of length P at different starting points (phases). If the phase of the folding happens to align with that of the periodic signal, the magnitude of the sum is amplified at a period of P while the aggregate value of the noisy samples in the series remains low. In order to determine the correct phase, CMF requires systematically calculating the result for all the different phases. ZiFi has been implemented on an Asus Linux notebook connected to a TelosB sensor node and on a Nokia N73 smartphone connected to a ZigBee module over the miniSD interface. ZiFi claims to have an average accuracy of 95 % reliable detection with a spectrum sensing overhead of 786.4 ms.

3.5.2 WiSpot - Fast and Reliable Wi-Fi Detector

A high surge of different low-power applications based on the IEEE 802.15.4 radios in the shared 2.4 GHz frequency band causes interference mainly from the existing Wi-Fi networks. Detecting Wi-Fi signals is therefore highly important for any interference mitigation scheme. Most of the existing schemes lack real implementation and a few recently realized schemes do not provide the desired characteristics in real deployments.

We have developed WiSpot which allows fast and reliable detection of Wi-Fi transmitters. It is composed of COTS hardware components, which makes it ready for technology transfer in real applications. WiSpot uses a low-complexity algorithm and is therefore suitable for resource constrained devices. It is robust to out of band signal leakages as are present in commercially available Wi-Fi NICs. It has shown high performance characteristics in its extensive performance evaluation (cf. Section 3.5.3). In the following we describe hardware and software design details of WiSpot.

Prototype Hardware Design

WiSpot consists of two IEEE 802.15.4 compliant radios interfaced to a low-power microcontroller as shown in a simplified block diagram in Figure 3.32. We selected CC2420 radio transceivers interfaced with an MSP430 series microcontroller for our design. The SPI bus is used for reading/writing the configuration registers as well as the RX/TX FIFOs on the two radios. The RSSI value from the CC2420 radio is mapped to an internal register and is also read over the SPI bus. The CCA (Clear Channel Assessment) pin interrupts the microcontroller with the status of the clear

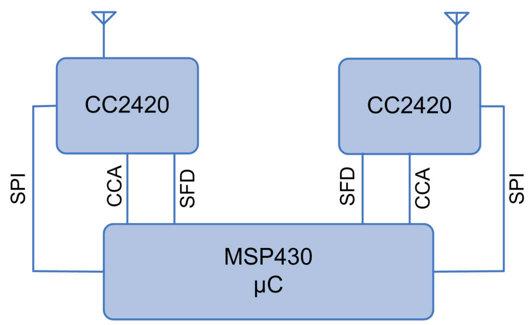
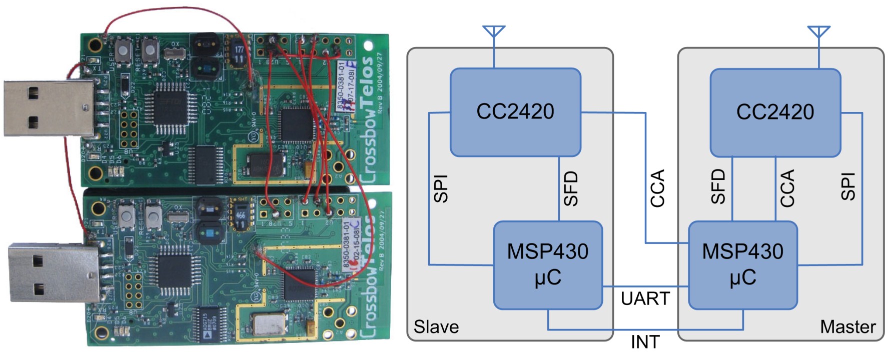


Figure 3.32: A simplified block diagram of WiSpot consisting of an MSP430 [70] microcontroller and two CC2420 [42] radios.



(a) Snapshot. (b) Simplified schematics.

Figure 3.33: WiSpot prototype platform.

channel assessment while the SFD (Start-of-Frame-Delimiter) pin triggers an interrupt when an IEEE 802.15.4 header is detected.

In our prototype implementation (cf. Figure 3.33a), we interfaced two TelosB nodes in a Master-Slave configuration as shown in Figure 3.33b. A TelosB platform consists of a CC2420 radio transceiver and an MSP430 series microcontroller operating at 4 MHz. The microcontroller supports commonly used serial bus interfaces and provides multiple General Purpose Input/Output (GPIO) lines. The Universal Asynchronous Receiver/Transmitter (UART) interface is used for communicating with the slave TelosB. In particular, RSSI values and the status for different operations is read over the UART. Furthermore, it is also used for communicating the CCA detection threshold value, the channel to be scanned and the channel scanning duration to the slave node. The CCA interrupt from both the radios is processed at the master microcontroller. Besides the radio interfacing, a GPIO line (shown as INT in Figure 3.33b) is used for coordinating the synchronized sensing operation on the two radios.

Embedded Software Implementation

We have implemented the drivers and the algorithm in TinyOS 2.x. MSP430 assembly language and low-level TinyOS 2.x abstractions have been used to minimize the signaling overhead in TinyOS 2.x while handling interrupts and GPIO callback functions. New MSP430 clock drivers have been developed to allow higher SPI and UART bus speeds. In the CC2420 driver implementation, we have separated the commands necessary for switching a frequency channel from the sequence of commands required only at boot-up. Furthermore, while switching a frequency channel, the voltage regulator is kept running instead of completely switching off the radio. This implementation has improved the frequency channel switching speeds to approximately 60 % faster as compared to our previous implementation [10]. Table 3.5 lists the average time required for different operations in our prototype.

The algorithm is computationally light-weight and is implemented on the host microcontroller. The executable code for TinyOS 2.x is in the form of a single binary file. The memory footprint in terms of the RAM and ROM usage on the master microcon-

Table 3.5: Average time durations measured on the WiSpot prototype.

Operation	Required time
Reading an RSSI sample	70 μs
Checking the CCA status	22 μs
Radio boot-up duration (only once)	2.41 ms
Frequency channel switching	740 μs
Timeout for SFD interrupt after detecting channel busy	160 μs
Maximum channel sensing duration	100 ms

troller is 17308B and 9184B, respectively. The RAM and ROM usage of the executable binary file on the slave microcontroller are 12156B and 4470B, respectively. The code size accounts for approximately 35 % of the available flash memory.

WiSpot Algorithm

WiSpot algorithm uses a pair-wise sensing of the IEEE 802.15.4 channels. The pair-wise channel sensing is performed to cover the whole 2.4 GHz frequency spectrum. During the spectrum scanning process, samples for signal strength levels are gathered, which are later analyzed at the master microcontroller for Wi-Fi transmitter detection. The algorithm consists of multiple parts and will be explained in detail in this section.

A unique characteristic of IEEE 802.11 networks is the periodic transmission of a beacon frame. A beacon frame is transmitted every 100 ms in most of the deployed networks [94]. The minimum in-air duration for a beacon frame and the smallest packet in IEEE 802.11b/g networks are 224 μs and 194 μs, respectively [94]. This implies that WiSpot prototype implementation (cf. Table 3.5) allows obtaining at least two RSSI samples. If a measured RSSI value is found to be higher than the estimated noise floor in the channel, it may indicate the presence of a Wi-Fi transmitter. However, sensing a 2 MHz wide channel cannot guarantee the presence of a Wi-Fi transmitter and its center frequency unless the adjacent channels are also sensed. Considering typical transmitters in the 2.4 GHz spectrum, a high energy level sensed in one of the IEEE 802.15.4 channels can possibly also be due to another IEEE 802.15.4 transmitter or a Bluetooth device. The channel switching duration of 740 μs (cf. Table 3.5) is so long that the presence of an on-going Wi-Fi signal cannot be guaranteed. Therefore, unless a signal is periodic and/or the Wi-Fi transmission intervals are continuous and long, reliable detection of a Wi-Fi transmitter is not possible using only a single IEEE 802.15.4 radio.

The overhead for detecting a Wi-Fi transmitter can be divided into three distinct categories: communication overhead, sensing overhead and processing overhead. While the sensing overhead is attributed to the characteristics of the IEEE 802.11b/g signals (including the frequency channel bandwidth and the minimum in-air frame duration), the communication and processing overhead has been optimized on the WiSpot prototype as we will describe later with the help of Figure 3.34. The sensing overhead is fixed to the maximum sensing duration of 100 ms, which corresponds to the default

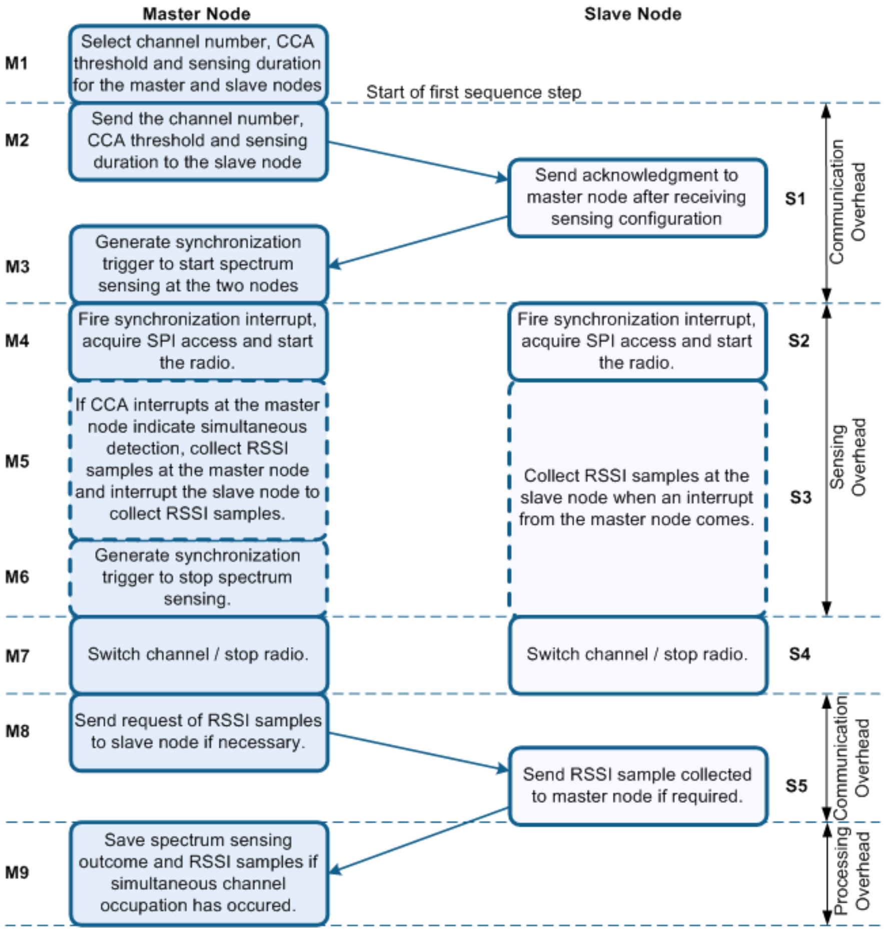


Figure 3.34: Sequence of steps performed on the Master-Slave nodes at the WiSpot prototype for IEEE 802.11b/g channel detection using the CCA mode.

beaconing rate of a Wi-Fi AP. The communication overhead on the WiSpot accounts for the master-slave communication and the overhead for sending configuration command messages to the radios. Furthermore, reading the sampled RSSI values and handling the interrupts are also attributed to the communication overhead. Finally, the processing overhead includes the analysis of RSSI samples and the decision logic.

WiSpot has two modes of operation. In the RSSI mode, the RSSI samples are gathered from the two radios in adjacent channels for a duration of 100 ms and post-processed for a simultaneous channel occupancy detection. In the RSSI+CCA mode, besides the RSSI samples, the CCA interrupt signals are also analyzed. Please note that a CC2420 radio transceiver has a built in clear channel assessment algorithm [42]. A CCA threshold is selected through the software (we used a default value of -77 dBm). If the sensed energy in the medium is detected to be higher than this threshold, the CCA interrupts from the two radios are invoked at the master microcontroller.

Figure 3.34 shows the different steps performed on the master and slave nodes while the simultaneous channel sensing operation is performed in the RSSI+CCA mode. The operations on the master node are indicated through M1-M9 while those performed at the slave node are indicated through S1-S5. Furthermore, these operations are divided into the communication, sensing and processing overhead categories. Compared to the RSSI+CCA mode, the operations in M5 and M6 on the master node and the corresponding operation S3 (as shown in Figure 3.34) differs in the RSSI mode, where both the master and slave nodes always collect RSSI samples for a fixed duration of 100 ms. Since gathering and simultaneously processing the RSSI samples on-the-fly significantly slows down the process, in the RSSI mode the gathered RSSI samples are post-processed for simultaneous channel occupancy. In the RSSI+CCA mode, only in the case when the CCA interrupts on both the radios are not triggered, channel sensing steps M5 and S3 are performed for the maximum duration of 100 ms. In RSSI+CCA mode, as soon as the CCA interrupts on both the radios are triggered in M5, RSSI samples at the master and the slave nodes are collected and the channel sensing operation is stopped. This scheme results in much shorter overall detection duration as compared to the RSSI mode as we will describe in Section 3.5.3. Having a shorter detection duration, RSSI+CCA mode is also more energy efficient. The algorithm for the scanning sequence of different channels to cover the complete spectrum is described later in this section. The set of operations shown in Figure 3.34 are repeated sequentially for each channel step.

In order to uniquely detect a particular Wi-Fi channel, three scanning sequences

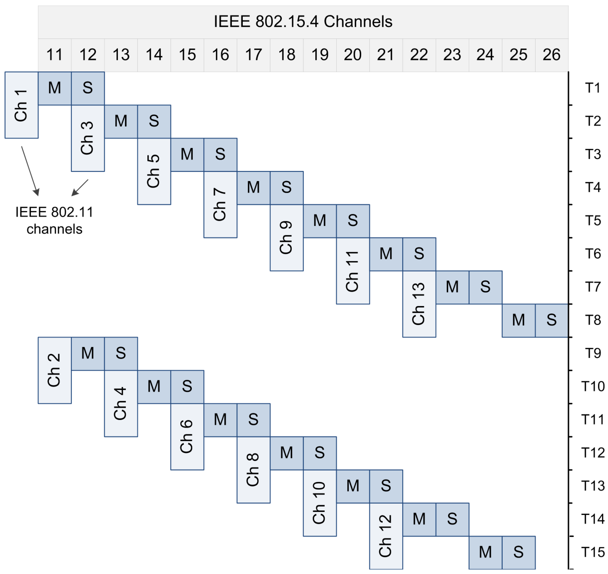


Figure 3.35: Synchronized Master (M) and Slave (S) spectrum sensing sequence on WiSpot for detection of IEEE 802.11b/g channels. A total of 15 Time steps (T1-T15) are required to cover the whole spectrum and uniquely detect an IEEE 802.11 channel using standard IEEE 802.15.4 channels.

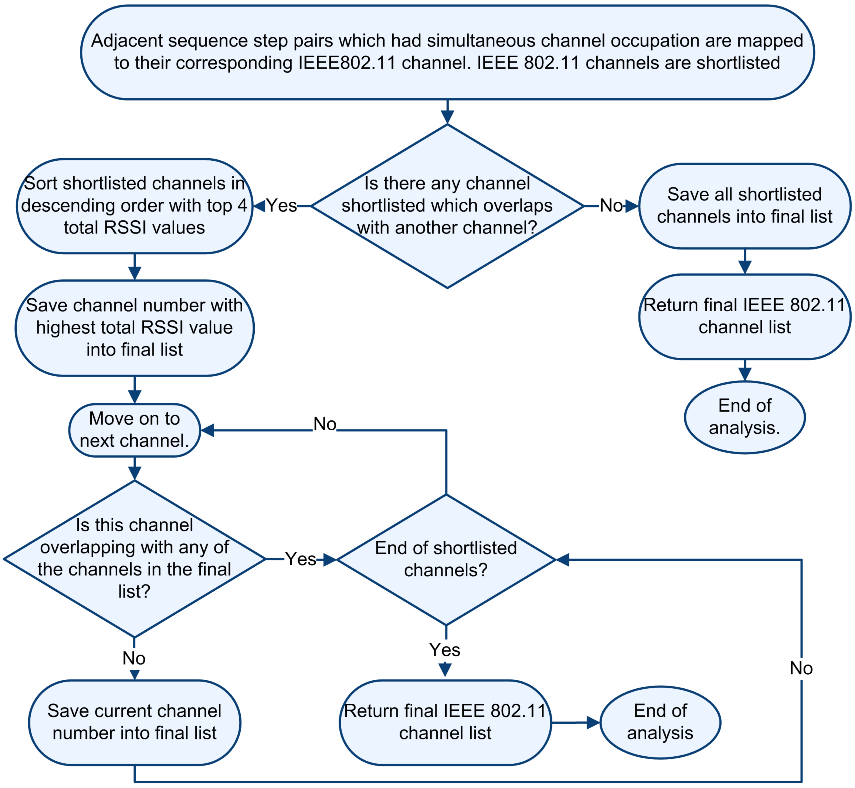


Figure 3.36: Final analysis and channel conflict resolution algorithm.

of adjacent IEEE 802.15.4 channels are required. Using the scheme described in Figure 3.35, WiSpot requires only 15 sensing steps to cover the complete 2.4 GHz spectrum. For instance, in the first step (T1), the master node chooses IEEE 802.15.4 channel 11 while the slave is assigned IEEE 802.15.4 channel 12. During a particular sequence step, the set of operations described in Figure 3.34 are performed at the master and slave nodes. If an IEEE 802.15.4 channel pair is detected to be simultaneously occupied, the mean RSSI values for the channel pair is stored before proceeding to the next channel pair. The IEEE 802.11b/g channels corresponding to the IEEE 802.15.4 channel pair (as shown in Figure 3.35) with simultaneous channel occupation are marked as *shortlisted*. At the end of the complete spectrum scanning operation (after step T15), identification of IEEE 802.11b/g channels is carried out at the master node using the algorithm described in Figure 3.36. The shortlisted IEEE 802.11b/g channels are checked for any overlaps. If no overlap is found, the shortlisted channel becomes the final list for IEEE 802.11b/g detected channels. A shortlisted channel list without a conflict indicates that the collocated Wi-Fi channels are non-overlapping, i.e., the central frequencies are more than 25 MHz separated apart.

In certain cases, a conflict situation may also arise due to the out-of-band channel leakages and overlapping Wi-Fi channels. If an overlap is found, an iterative *conflict* *resolution* strategy is applied. The conflict is resolved by sorting all the shortlisted channels in a descending order based on their four channel RSSI total. The four channel RSSI total is the sum of the mean RSSI values (higher than the CCA threshold) of the four IEEE 802.15.4 channels and is a representation of the total energy measured over a bandwidth of 15 MHz. Based on the fact that the total energy is highest around the central frequency of an IEEE 802.11 transmission channel, finding the correct channel from a list of overlapping channels is done by picking the shortlisted channel with the highest RSSI total. It is due to the fact that shortlisted channels have been sorted, the first entry is stored in the final list and all other subsequent channels which overlap with this entry are ignored. Thereafter, when the next non-overlapping channel with the highest RSSI total is found, it is automatically added to the final list and once again all other subsequent channels which overlap with this channel are ignored. Once the search has reached the end of the sorted list, it becomes the final list. The conflict resolution scheme is also robust to the significant amount of Wi-Fi signal leakages as observed within a range of ca. 8 m on most of the commercially available NICs. Besides using the RSSI samples to detect the presence of an IEEE 802.11 transmitter, the RSSI samples over two sensing durations are also analyzed to characterize the channel utilization ratio as well as the interference levels. This information can be later used for the selection of transmission channel in an IEEE 802.15.4 network [92]. Furthermore, after a separation of 11 MHz and 22 MHz from the center frequency, the radiated energy from an IEEE 802.11b transmitter is 30 dB and 50 dB below the maximum level, respectively. In contrast, IEEE 802.11g transmitter's radiated energy level after a separation of 11 MHz and 22 MHz from the central frequency is only 20 dB and 30 dB below the maximum, respectively. This characteristic is exploited in WiSpot RSSI readings over a distance of ca. 10 m to identify the Wi-Fi signal mask.

3.5.3 Experimental Performance Evaluation

We have carried out the performance evaluation of WiSpot both in the RSSI mode and the RSSI+CCA mode. We used the IEEE 802.11b/g AirHORN dual band signal generator, Linksys WRT54GL broadband router and Atheros based mini-PCI-E card on Asus Eee netbooks as signal sources providing 20 dBm of transmit power level. The beacon rate for all the APs is fixed to the default value of 100 ms. All the experi-

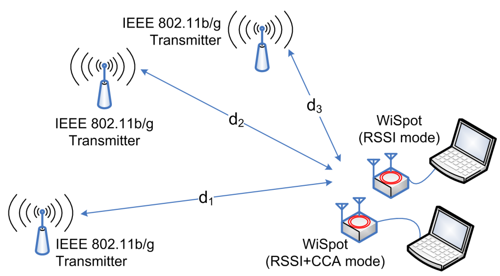
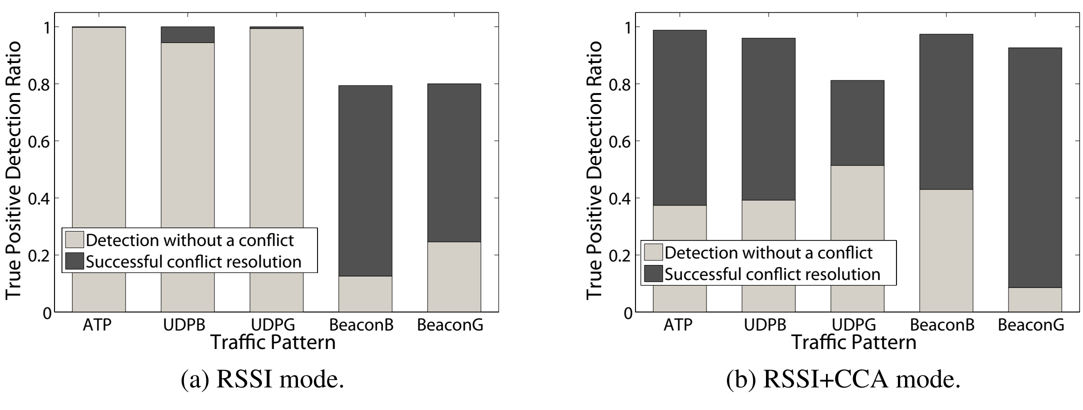


Figure 3.37: Experimental setup for WiSpot measurements.

ments were conducted in an indoor office environment with line-of-sight between the WiSpot and the transmitter(s). We obtained 500 samples for each measurement point in all the experiments. Figure 3.37 shows the experimental setup for a single transmitter and multiple transmitters, where the distances of the transmitters were varied from a WiSpot in the RSSI mode and another WiSpot in the RSSI+CCA mode. PCs connected to the WiSpot platforms record the detection characteristics for each experiment. During the experiments, Agilent E4440A spectrum analyzer was used to monitor that no astray signals influence our measurements.

True Positive and False Negative Detections

We used an AirHORN IEEE 802.11b signal generator transmitting with the maximum possible rate, giving a radio duty cycle of approximately 72 % (ATP), an IEEE 802.11b/g beaconing Linksys AP (BeaconB, BeaconG) and a netbook sending a UDP stream using the *iperf* tool in both the IEEE 802.11b and IEEE 802.11g modes (UDPB and UDPG). The UDP stream used a datagram size of 1470 B and the observed traffic statistics per second were reported as an average of 7.3 Mbps and 24 Mbps for UDPB and UDPG, respectively. In the rest of the experiments, the traffic settings were kept the same and the same notations as above are followed. Figure 3.38 shows the true positive ratio for different traffic patterns in a single transmitter scenario when the distance of WiSpot to the transmitter is fixed to 3 m (to allow a comparison with ZiFi [137]). It can be observed that the RSSI mode shows a detection ratio between 95 % to 99.8 % when the channel utilization is high for the case of ATP, UDPB and UDPG, while it drops to 80 % for the case of beacon transmissions. On the contrary, the RSSI+CCA mode shows a detection ratio of 96 % except for the case of UDPG where it falls to 82 %. The figure also shows the breakdown analysis of the true positive ratio in terms of the detection without the conflict situations for overlapping Wi-Fi channel detections and the detections after successfully resolving the conflict situations. The RSSI+CCA mode shows much higher conflict cases, which are due to a high sensitivity threshold and leakages at this short distance. However, our algorithm (cf. Figure 3.36)



(a) RSSI mode. (b) RSSI+CCA mode.

Figure 3.38: True positive detection ratio on WiSpot at a transmitter distance of 3m.

is able to successfully resolve the conflicts and is able to provide an overall detection accuracy between 95 % to 99.8 %. In both the operating modes, false detections are not encountered. This is because compared to a single channel sensing scheme, the chance of getting a simultaneous noisy spike in both independently sensed adjacent channels is negligible.

Analysis of the Detection Time

Fast and reliable detection leads to energy efficiency but is also desirable in dynamic environments to sense the interfering channels and opportunistically utilize this information. A fast detection time also leads to energy savings. The RSSI mode shows almost the same detection time for different traffic patterns and the average detection duration is 1.99 ± 0.2s. The absence of a signal source has almost no effect on the detection duration. On the contrary, as can be observed from Figure 3.39, the required time for Wi-Fi detection on WiSpot operating in RSSI+CCA mode is much lower. Based on the average detection time of 1.24 s, the longest per IEEE 802.11 channel detection and the overall sensing durations turn out to be 310 ms and 2.33 s, respectively. The average per channel communication and processing overhead is approximately 55 ms, which accounts for 17.74 % of the overall timing overhead. The figure also indicates that as the channel utilization ratio increases (from beacon to UDP traffic and ATP), the required detection time decreases. However, it is worth noting that WiSpot is able to detect Wi-Fi transmissions for any channel utilization ratio.

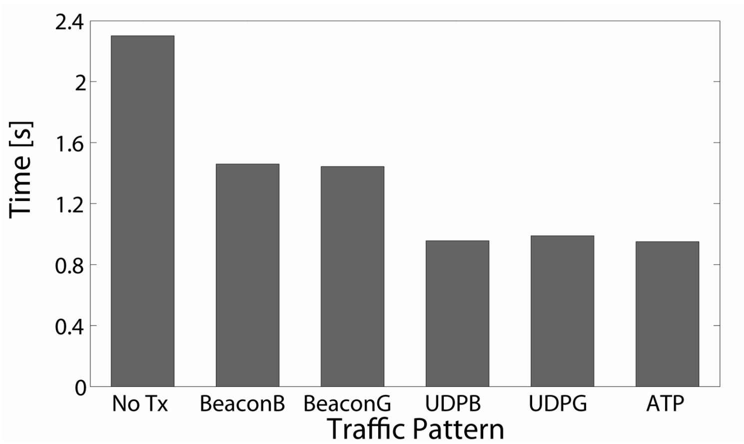
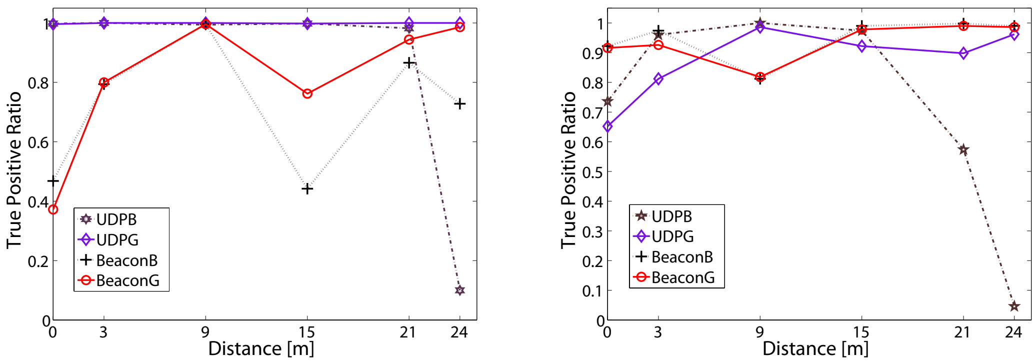


Figure 3.39: Average detection duration on WiSpot in the RSSI+CCA mode for scanning all the Wi-Fi channels. The average is obtained over 500 samples.

Effect of Different Distances

In order to study the effect of physical distance from the transmitter on the performance of WiSpot platform, we have conducted the above mentioned experiments at different distances. Figure 3.40 shows that at larger distances from the transmitter, the performance of the RSSI mode for BeaconB suffers. WiSpot operating in the RSSI+CCA mode shows higher stability compared to the RSSI mode over a longer range of distances.



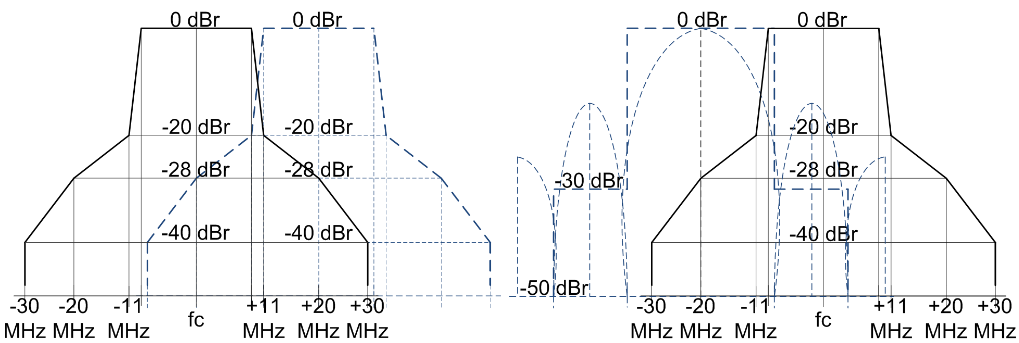
(a) RSSI mode. (b) RSSI+CCA mode.

Figure 3.40: Average true positives on WiSpot w.r.t. distance from the transmitter.

The measured received power at a distance of 21 m is observed to be approximately -78.1 dBm which is very close to the detection threshold of -77 dBm and hence beyond this range WiSpot does not detect the signal. Of course, a lower detection threshold can be selected to achieve a longer range but it may lead to higher number of false detections.

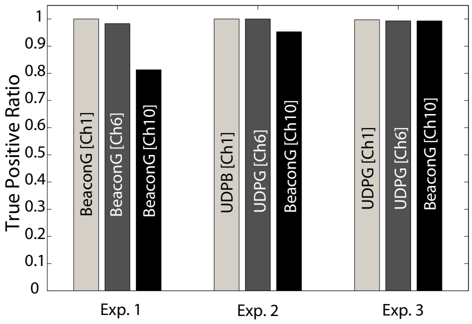
Detection of Multiple Collocated Transmitters

The existence of multiple collocated Wi-Fi networks is very common in office and home environments. WiSpot is able to detect multiple collocated Wi-Fi networks. In most of the Wi-Fi deployment setups, non-overlapping channels 1, 6 and 13 are used. In order to study the reliability and accuracy of WiSpot, we have conducted a series of experiments with 3 collocated Wi-Fi networks in channels 1, 6 and 10 to represent a separation of 5 and 4 channels, respectively. Furthermore, we have used heterogeneous traffic patterns as described above and use different distances of the transmitter

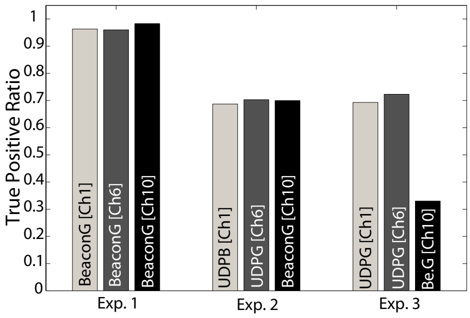


(a) IEEE802.11b and IEEE802.11g (b) Two IEEE 802.11g signals

Figure 3.41: Overlap of two IEEE 802.11 signals with 4 channels separation.



(a) RSSI mode.



(b) RSSI+CCA mode.

Figure 3.42: Avg. true positives on WiSpot for 3 transmitters in channels 1, 6 and 10.

from the WiSpot platforms operating in both the modes. Figure 3.41 shows the mask overlap of two Wi-Fi networks with a separation of 4 channels. In our measurements, the scenario was more challenging as we placed the transmitters at unequal distances which led to unequal scaling of the overlapping masks.

Figure 3.42 shows the true positives for the case when the transmitters in channel 1 and channel 6 are placed at a distance of 1 m, while the transmitter in channel 10 is at a distance of 15 m. It is worth noting that the RSSI mode performs better than RSSI+CCA mode and is able to give an average successful detection to be over 95 %, while for the case of RSSI+CCA mode, in the non-overlapping channels 1 and 6, a true positive ratio of 70 % is observed. This shows that the ability of WiSpot in the RSSI+CCA mode to detect IEEE 802.11 transmitters in the correct channel is higher when the transmission channels do not overlap. RSSI mode has the advantage of first gathering all the samples and then perform the analysis while the RSSI+CCA mode, upon getting a CCA interrupt may lose the later samples and this may lead to a false detection of an aggregate signal from the side-bands of two overlapping channels. This has resulted in an overall detection ratio of 75 %. One solution is to gradually decrease the CCA threshold in steps, which will avoid the trigger of simultaneous detections when the threshold is small but will require more scanning rounds.

Comparison with ZiFi Scheme

Unlike ZiFi, WiSpot requires two IEEE 802.15.4 radios. However, WiSpot is able to detect multiple collocated Wi-Fi transmitters and receive IEEE 802.15.4 frames. It is light-weight and is implemented on a sensor node microcontroller. We have evaluated WiSpot under identical experimental settings [137]. Table 3.6 compares different performance metrics on the two platforms. It can be observed that WiSpot outperforms ZiFi in terms of the detection ratio and the timing characteristics. Additionally, WiSpot does not pose any restriction on the channel utilization ratio and is able to detect multiple collocated Wi-Fi transmitters.

Table 3.6: Comparison between ZiFi and WiSpot.

Performance metric	ZiFi	WiSpot	
		RSSI mode	RSSI+CCA mode
True positive ratio	96%	91%	96%
False positive ratio	5%	0%	0%
Min. detection time per channel	786.4 ms	500 ms	266 ms
Max. detection time per channel	786.4 ms	500 ms	310 ms
Whole spectrum scanning duration	–	3.75 s	2.33 s
Required channel utilization for detection	0-30%	0-100%	0-100%

3.5.4 Discussion and Conclusions

We have described the design and implementation details of WiSpot – a platform for fast and reliable detection of IEEE 802.11b/g signals. WiSpot is based on two IEEE 802.15.4 radios and uses a synchronized channel sensing scheme for detecting Wi-Fi transmitters. Using a conflict resolution strategy and the spectral masks of the IEEE 802.11b/g signals, our algorithm is able to accurately identify Wi-Fi transmitters and is robust to out-of-band signal leakages in commercially available Wi-Fi NICs. Additionally, WiSpot is able to recognize IEEE 802.15.4 frames. The developed algorithm works without modification well for IEEE 802.11n networks as long as they do not use channel bonding, i.e., use a bandwidth of 22 MHz. However, in the case of channel bonding (40 MHz frequency bandwidth) the algorithm requires minor modifications in the look-up table. The algorithm is light-weight and is implemented on the host microcontroller. This will certainly allow it to be implemented as a part of a MAC protocol such as [10, 122, 125, 131] for improved spectrum sensing. This will ultimately result in higher communication reliability by avoiding external interference and allow coexistence with Wi-Fi networks.

3.6 SUMMARY

In this chapter, we have highlighted the importance of spectrum agility and coexistence features in resource constrained embedded networks. Spectrum agility is necessary for low-power applications in order to avoid interference and achieve reliable data communication. Although many experimental studies have shown that resource constrained networks suffer high packet losses from other networks operating in the same frequency band, only a little has been towards designing interference mitigation schemes. We think that distributed solutions which do not require centralized coordination and control infrastructure favor low-power applications. Though we have shown spectrum agile operation through our dual-radio MR-MAC proposal [19], we think that with the crowding of the wireless spectrum, availability of a dedicated CCC will not be guaranteed in the future. Therefore, we have introduced the concept of multi-channel polling operation combined with distributed channel selection in SA-MAC protocol [10]. SA-MAC uses a heuristics based fully decentralized dynamic channel selection strategy. Prototype implementation of SA-MAC on commercially available low-power nodes has shown high performance characteristics in terms of packet delivery ratio and throughput even in the presence of challenging wireless interference. We have also discussed the constraints why classical CR MAC protocols cannot simply be adapted for low-power applications. We have designed a decentralized MAC protocol, CogMAC [93], for CR networks, which uses a similar distributed channel selection strategy as SA-MAC. CogMAC is targeted for relatively less resource constrained ad hoc networks requiring DSA capabilities. It is suitable to both licensed and ISM frequency bands, where the protocol adapts its channel selection to the stochastic spectrum occupancy behaviour of the PUs and mitigates the effects of the random interference, respectively. We have implemented CogMAC on WARP SDR boards, which provide significant insights into the practical aspects of cognitive radio MAC protocols, especially when majority of the state-of-the-art designs remain at a theoretical and simulation stage. The performance evaluation indicates CogMAC allows efficient utilization of spectral opportunities. We also advocate that MAC parameters such as backoffs, persistency values and the CCA algorithms affect the coexistence behaviour. In a separate work [34], we have empirically studied the behaviour of these MAC parameters for nanoMAC and B-MAC. Reliable detection of interfering networks is highly important for dynamic channel selection schemes. Existing studies show that low-power applications using the IEEE 802.15.4 based radios suffer severe performance degradation due to the interference from IEEE 802.11b/g networks [7, 11, 136]. Therefore, we have designed WiSpot [94], which is based on a dual IEEE 802.15.4 radio platform and uses a synchronized channel sensing algorithm for detecting IEEE 802.11b/g transmitters. Empirical performance evaluation shows that it allows fast and reliable detection of IEEE 802.11b/g transmitters and outperforms other state-of-the-art approaches. WiSpot has certainly a high potential to be a part of a MAC solution aiming at interference mitigation.

4

RECONFIGURATION AND ADAPTATION

A wide range of applications and deployment conditions put varying degree of requirements on MAC solutions for embedded wireless networks. There is no universal MAC protocol that fits to all the low-power applications. Therefore, it becomes increasingly important to be able to efficiently prototype customized MAC solutions that may satisfy different application requirements. Cognitive radios are becoming the technological foundation for efficiently managing the scarcity of wireless spectrum, fulfilling various Quality of Service (QoS) demands and allowing different networks to coexist. Cognition and spectrum agility in MAC protocols require adaptability and close PHY-MAC interaction. Traditionally, MAC protocols have been implemented in hardware, which gives a limited possibility for reconfiguration and customization. Recently, software based MAC implementations have emerged although a close hardware-software co-design is typically required to keep the time critical operations in silicon.

In this chapter, we investigate 1) rapid prototyping of MAC protocols and 2) enabling runtime reconfiguration in medium access schemes. We have addressed these two objectives by developing a component-oriented design methodology and introducing a MAC development framework for enabling fast composition of MAC protocols, which are best fitted to the application requirements, communication capabilities of the radio and current spectrum regulations and policies. We have analyzed a wide range of MAC protocols and identified the basic common functionalities so that these may be used as building blocks to compose different protocols. Composing MAC protocols based on reusable components allow fast prototyping and a wider room for experimentation. We have developed a set of light-weight tools that allow on-the-fly realization of an envisioned MAC solution through wiring of the components.

We have listed the requirements and design goals for MAC development framework for fast prototyping of MAC solutions and enabling reconfiguration in Section 4.1. Section 4.2 describes the main contributions of our work on the flexible and reconfigurable MAC design. Section 4.4 describes the design rationale, various components of the designed framework, and the performance evaluation of the framework on WARP [15] boards. Section 4.5 describes the developed interactive GUI based environment and the related tools as part of our framework which facilitate fast prototyping of MAC solutions. The MAC solutions generated through our framework are compared with their hand-coded monolithic counterparts to validate that while substantially easing the development efforts, the performance characteristics remain similar. This chapter is mainly based on our articles [129,138–141], which were published during the dissertation work.

4.1 DESIGN GOALS

In the following, the major design goals for a MAC development framework are listed.

Fast Prototyping
The rapid development of standards and technologies, and the wide range of different applications with varying requirements jointly lead to the need for fast prototyping of different MAC solutions. The development of different MAC layers for varying wireless technologies has become very important for industrial and academic research community. The major hurdles in prototyping new solutions include high implementation efforts and the lack of experimental support in the MAC development frameworks.

Portability and Code Reuse
Most of the existing MAC implementations in embedded networks are carried out in a monolithic fashion with tight coupling to the underlying radio and microcontroller. This not only restricts portability across different platforms and the possibility of code reuse, but also code modifications. Therefore, it is important for a development framework to provide MAC abstractions with well-defined interfaces, which allow code reuse across different MAC protocol implementations and facilitate portability.

Granular Parameters
Intelligent management of spectral resources, cooperation among nodes and advanced sensing in medium access procedures require adaptability and close interaction between PHY and MAC layers. Adaptability and reconfiguration based on network statistics and channel conditions for meeting the application QoS requirements are needed during the execution of a cognitive MAC. Flexible MAC solutions and cross-layer designs, especially in the case of spectrum agile and cognitive MAC protocols require fine-grained access control to different PHY and MAC layer parameters [5]. These "hooks" are needed to be exposed through a rich set of APIs for versatile protocol implementations.

Runtime Reconfiguration
One of the key characteristics of future MAC protocols will be the ability to adapt to the changing environment. Generally, adaptability is important for MAC layers in the context of the future software defined radios and cognitive radios. MAC designs should therefore support mechanisms for fast on-the-fly reconfiguration.

Hardware-software Co-design
MAC protocols implemented in hardware (e.g., IEEE 802.11 NICs, Bluetooth devices, etc.) leave limited room for reconfiguration and customization. These static and rigid hardware based MAC implementations are inefficient for fulfilling the diverse and changing application demands and thus fail in flexible spectrum management domains [142]. Pure software implementations often remain incapable to meet the time-critical deadlines, while hardware implementations tend to be rigid, lack room for

experimentation and are unable to provide the needed flexibility and reconfigurability by cognitive MAC protocols. Therefore, HW/SW co-design approaches and hybrid processor FPGA designs such as Garp [143] becomes desirable.

4.2 MAIN CONTRIBUTIONS

In the following, we summarize the main contributions of our work towards reconfigurable and adaptable MAC design.

- We have analyzed a wide range of MAC protocols based on the CSMA, TDMA and hybrid principles to identify the basic components as building blocks. Since protocols can be implemented heavily using these components, our approach allows implementing MAC solutions with high degree of code reusability.

- We have designed the necessary APIs of the components so that these are generic enough to be reusable across different MAC protocol realizations and hence provide code portability.

- We have developed a tool, which we refer to as Wiring Engine that binds the components together, manages the component resources and executes the state-machine of a MAC protocol. It allows composition and reconfiguration of MAC solutions at the runtime. Wiring Engine is light-weight and thus can easily be implemented on resource constrained nodes.

- For speed gains, we advocate the idea of partitioning the PHY/MAC components across software and hardware depending upon the computational complexity and associated communication overhead. This is highly important for cognitive and spectrum agile MAC solutions where granular control over the PHY/MAC parameters, fine-grained timing deadlines and cross-layer design are required [5]. We have defined a set of MAC components that efficiently support a hardware-software co-design.

- We have also developed a user friendly Integrated Development Environment (IDE) where MAC components are exposed to a developer through an interactive GUI. In order to realize a MAC protocol, a user simply needs to ‘drag n drop’ components to a drawing grid, connect them and set the parameters of each component through dialog boxes. The code for a supported target platform is auto-generated and downloaded. This certainly speeds up the prototyping of MAC protocols. We also allow the possibility of reusing and modifying existing designs and adding more components into the provided MAC component library for future usage. Experimental results indicate that the auto-generated code shows similar performance characteristics as their hand-coded monolithic counterparts.

- Pure hardware PHY/MAC implementations tend to be rigid and do not provide the required flexibility and adaptability features for cognitive MAC designs and experimental prototyping [138]. On the contrary, pure software implementations, though flexible, have limitations in satisfying strict timeliness requirements. With our component oriented design approach, key functionalities requiring hardware acceleration can be implemented in dedicated silicon or on multicore computational fabrics. We have developed a toolchain, which exploits these architectures. Furthermore, we have introduced a MAC meta-compiler as part of the toolchain to allow autonomous realization of a MAC protocol expressed in a few lines of code [140, 141].

4.3 RELATED WORK

MAC protocols are typically implemented in a monolithic fashion with tight coupling to the underlying hardware. A hardware specific implementation restricts reconfigurability and flexibility. Furthermore, it limits the portability and design extension possibilities. Efforts have however been made to achieve a unified protocol structure for easing implementation burden and for maximizing code reuse. Polastre *et al.* proposed a unified link layer abstraction, which allows implementation of a broad range of networking and data link technologies without significant loss in efficiency [144]. ULLA [145] focuses on the design of flexible link layer APIs and offers a common interface to fetch link layer information independently from the underlying radio technology. In Unified Power Management Architecture (UPMA) [146], link-layer interfaces are defined to separate core radio functionalities from power management features so that power managers can be plugged into different hardware specific MAC implementations. In a continuation work on UPMA, focus has been laid on the optimization of all the radio states [147] for achieving energy efficiency. Bianchi *et al.* discuss the advantages of an adaptive and programmable MAC framework, which satisfies time varying QoS demands [148]. MultiMAC [57] allows switching among a few pre-defined standalone MAC solutions according to the network and application requirements. This approach has a limited scope because it can only find an approximate "best-fit" solution from a limited number of available MAC protocols. Many of the resource constrained nodes are limited in terms of the available memory to afford such a selection.

Messerschmidt has discussed in detail the modularity and component oriented design approach [149]. Following the component based approach, MAC Layer Architecture (MLA) [150, 151] extends the UPMA idea by identifying and modularizing platform-independent components. The implementation shows significant code reuse across different protocols with a low memory overhead and without significant loss in terms of performance metrics. However, MLA decomposes only the platform independent part of MAC protocols. The hardware and protocol specific code still contributes considerably to the MAC implementation effort, which we have addressed in our approach [138, 139]. Braden *et al.* have proposed role based approach based on the component oriented and non-classical view for protocol realization to allow reusability, flexibility and portability [152]. However, the protocol heap approach was never actually tested with implementations and by closer analysis.

In order to provide reconfigurability features to a MAC developer through flexible interfaces, Sharma *et al.* propose FreeMAC [153], which is a multi-channel MAC development framework on top of the standard IEEE 802.11 hardware. FreeMAC aims at supporting frequent channel switching and efficiently controlling the timings for packet transmissions. A similar work has been demonstrated [154]. Sharma *et al.* control the timings of the frames through the MadWifi [155] driver for Atheros based NICs in order to improve the overall throughput [156]. Djukic and Mohapatra implement a TDMA MAC protocol in software (Linux user-space) using the MadWifi driver for Atheros AR5212 chipset [157]. Since IEEE 802.11 hardware is restricted in providing accessibility to the radio and PHY parameters as are needed by spectrum agile and cognitive MAC protocols, the configurability aspects of these protocols are limited. Furthermore, these designs are not implemented in a modular fashion to allow portability – only a few MAC parameters can be tuned.

In recent years, software based MAC implementations using GNU radio [158] have emerged to offer higher radio and physical layer reconfigurability. However, pure software implementation approaches have shortcomings in meeting time critical deadlines. SDR platforms have demonstrated possible advantages in MAC designs based on flexible radios. Some of the low-cost SDR platforms do not offer good support for flexible MAC implementation as has been learnt by several research groups for USRP [159] and GNU Radio platforms [142, 160]. WARP [15] provides experimental opportunities for customizing hardware and prototyping MAC schemes. Hunter *et al.* have implemented a fully distributed cooperative PHY/MAC system on WARP boards with accurate timing characteristics [161]. However, despite the fine-grained access control knobs offered by the platform, current MAC development frameworks on SDR platforms lack the support for flexible MAC design and fast on-the-fly reconfiguration.

In order to realize efficient MAC implementations, Nychis *et al.* have split the MAC functionalities based on the timing characteristics and their latency tolerances [142]. Following similar design methodology, we have designed Decomposable MAC Framework [138], which partitions the PHY/MAC component implementation in software and hardware in a way that time critical components reside in hardware for computing and communication speed gains while flexible binding of the PHY/MAC components and the MAC state-machine logic is implemented in software. The concept of component interconnections has long been established in software engineering since a software waveform can be expressed as a network of boxes communicating with each other via connecting buses. Compositional adaptation techniques [162] have been thoroughly investigated for runtime reconfigurations, such as in robotic software systems. Networking researchers have actively used Click [163], which provides a generic approach to binding components. However, in order to allow fast speed and suit to limited resources, a customized tool is more suitable as we show later in our evaluation. Unlike the approach of Nychis *et al.*, in Decomposable MAC Framework [138], we define the PHY/MAC components not just based on the timeliness requirements but also communication demands and the degree of reuse. Compared to [57], our toolchain does not require large memory for storing the many different perceived protocols in order to switch among them as the application requirements evolve. Realizing components with high communication and computational demands on multi-core platform opens a wider experimental room and a new perspective for dynamic protocol realization. In order to benefit from this approach and exploit the platform architecture, we have also designed a supporting toolchain known as TRUMP: Toolchain for RUntiMe Protocol realization [140, 141]. A compiler assisted approach for developing MAC protocols is proposed by Lichte *et al.* [164], which provides a systematic way of automating MAC implementation, analysis and code generation. However, this approach does not address the timeliness demands of MAC implementations.

4.4 DECOMPOSABLE MAC FRAMEWORK

We have analyzed a wide range of MAC protocols in order to distil the common features into a set of reusable basic components. Our framework enables realization of a particular MAC solution through interconnection of these components, and coordination of the control and data flow in an appropriate manner. In the following, we will describe the design details of our framework.

4.4.1 Component-based MAC Design

Different MAC protocols share many different functional commonalities. We have analyzed a number of MAC protocols based on the CSMA, TDMA and hybrid principles in order to identify the basic common functional components. Using these basic components as building blocks, different MAC protocols can be realized. This also reduces the design and implementation complexity. The idea is similar to the LEGO philosophy, where complex structures can be constructed using basic building blocks. Figure 4.1 illustrates the concept where two different MAC protocols are realized using the same set of basic components.

MAC functionalities such as timer, backoff counter, carrier sensing, frame formation, sending a frame, receiving a frame, etc., are not only common to different protocols but are often also repeated within a particular protocol realization. Additionally, access to radio control parameters like switching the state of the radio (transmit, receive, sleep), setting the transmit power level, tuning the receiver sensitivity level,

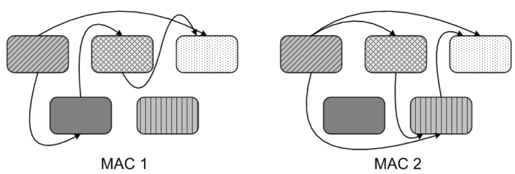


Figure 4.1: Illustration of two different MAC realizations using a set of common MAC components as building blocks.

Table 4.1: Timer Component Interface.

Command	*Meaning*
`start()`	starts a timer Inputs: Type {`ONE_SHOT, PERIODIC`} Precision {`Millisecond, Microseconds`} Output: signals when timer is fired
`stop()`	stops a timer if the timer is not expired yet.
`suspend()`	suspends the running of a timer until it is resumed.
`resume()`	resumes the running of a timer after the suspension.
`getStart()`	returns start time of the timer
`getDuration()`	returns timer duration
`getNow()`	returns current time
`getStatus()`	returns the status {`RUNNING, SUSPENDED, STOPPED`}

selecting the operating frequency, etc., are also typically needed by different MAC implementations. As an example, Table 4.1 lists the commands in the *Timer* component.

The pattern in which various primary components are connected can be identical across different MAC implementations. Therefore, we define secondary level MAC building blocks, which are composed of the basic components. In the following, we will describe in detail with an example of *Random backoff*, how a secondary level component is realized in our framework. The random backoff functionality is very common in CSMA/CA based protocols and consists of the primitive components *Timer*, *Carrier Sensing* and *Random Number Generator* (RNG). Figure 4.2 illustrates the realization

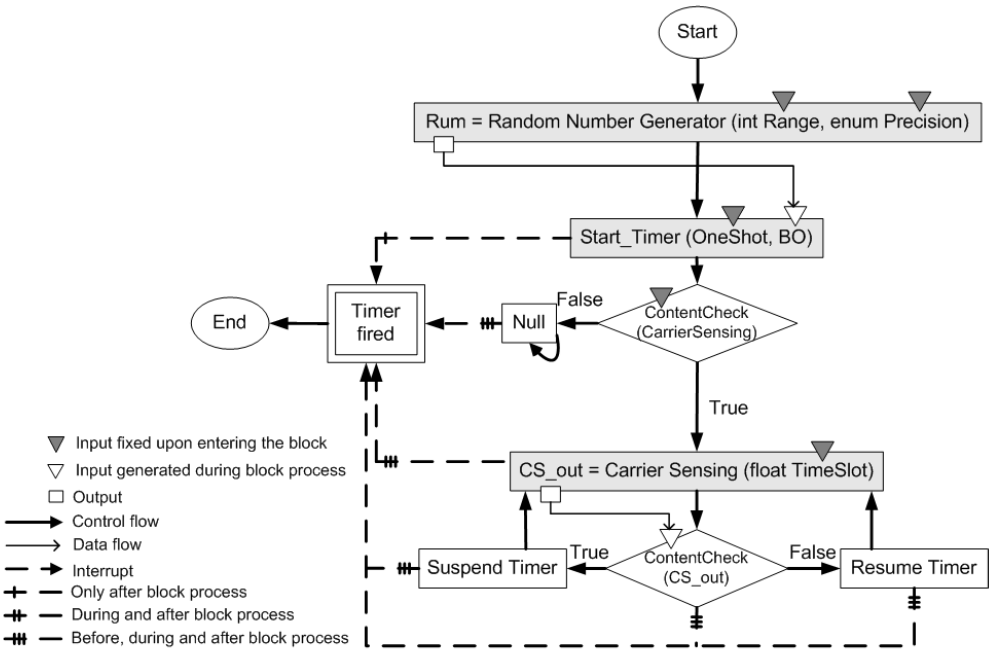


Figure 4.2: Realizing Random backoff using the basic MAC components.

of *Random backoff* component with inputs, outputs and the execution flow.

The *Random Number Generator* component takes *Range* and *Precision* as inputs for the random number generation, which is used to start a `ONE_SHOT` timer. If *Carrier Sensing* (CS) is disabled, a *Null* waiting state is assumed till the timer expires and the component execution is terminated. Otherwise, CS is performed for a specified duration. If CS returns true, the timer is suspended and CS is performed again. However, if the output is false, the timer is resumed and CS is repeated. However, if the timer expires during the operation, the block exits. The figure shows a detailed control flow (sequence of execution and preemptions) in the *Random Number Generator* component. We have realized different secondary MAC components in our framework based on the same principle. Table 4.2 lists the most commonly used components. The secondary level components can also be used to form other complex components.

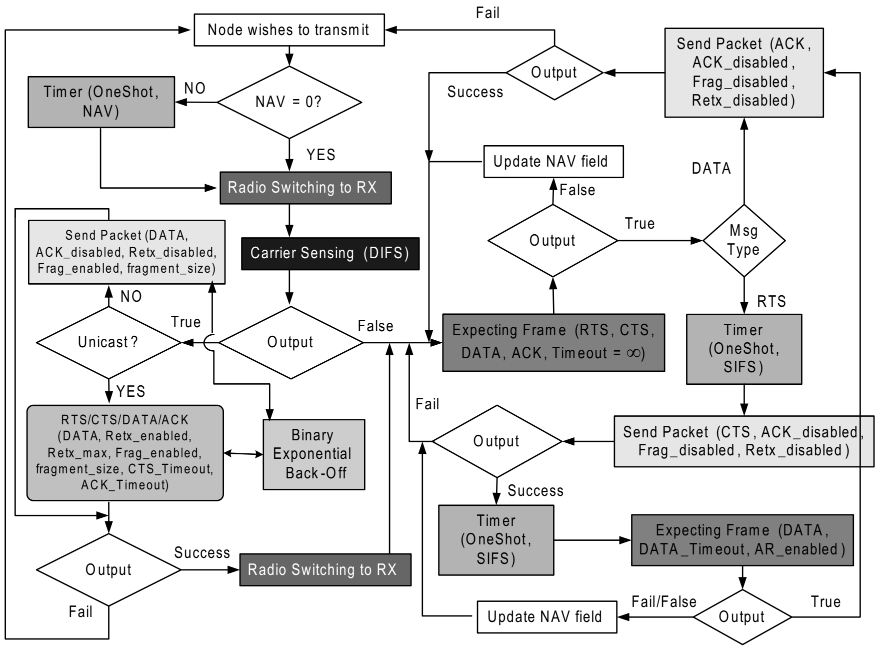
Table 4.2: Commonly used secondary components.

Component	Usage and the *composition*
Random Backoff	Random backoff mechanism *Timer, Random Number Generator, Carrier Sensing*
Expecting Frame	Used when the node is waiting in anticipation of a packet *ReceiveFrame, Timer, Radio Switching, SendFrame*
Send Packet	Called after seizing a channel free *SendFrame, Expecting Frame, Radio Switching, Random backoff*
RTS/CTS/DATA/ACK	Four-way handshake mechanism *Send Packet, Expecting Frame*
LplCoordinator	Controlling the low-power listening operation *Radio Control, Carrier Sensing, Timer*

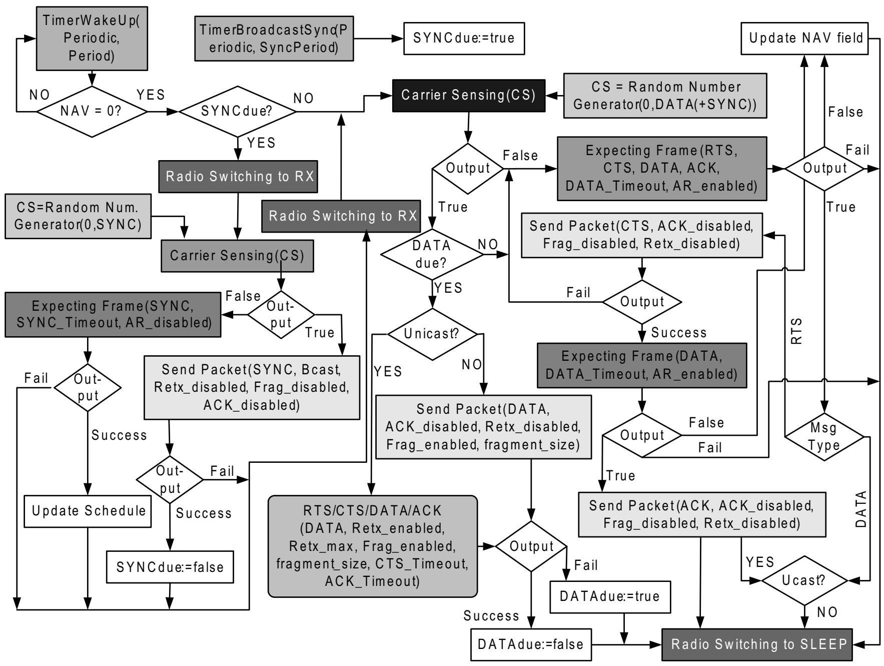
Figure 4.3 illustrates the realization of IEEE 802.11 DCF and S-MAC using the fundamental and secondary components. The details on data and control flow are omitted to keep the figure simple. The binding logic, decisions and variable assignments, which determine the state-machine of the MAC protocols are indicated through unfilled blocks. The state-machine logic is controlled by Wiring Engine. While the two protocols are realized using the same set of components, it is worth noting that certain components, for instance *Send Packet*, *Timer* and *Radio Switching*, are also repeated within the MAC realizations.

4.4.2 Wiring Engine

In order to bind different components together for realizing different MAC protocols, we have developed a tool called as *Wiring Engine*. It coordinates the flow of data and control among different components and schedules their execution sequence. It uses dependency tables (as we describe later in this section) for different components in order to direct the execution flow and avoid any potential race condition. The



(a) IEEE 802.11 DCF



(b) S-MAC

Figure 4.3: Realization of IEEE 802.11 DCF and S-MAC based on the fundamental and secondary components.

approach of binding components through Wiring Engine widens the possibilities for different MAC realizations. It also allows the possibility of composing and modifying MAC behaviours at runtime. Wiring Engine is a light-weight tool that introduces very low overhead since it is based on the use of function pointers corresponding to the components that form a MAC protocol. The construction and execution path of a state-machine is dynamically redirected through modification of function pointer assignments. This approach enables both runtime reconfiguration of MAC protocols according to pre-defined rules within the framework, and on-the-fly realization of user configured protocols.

Wiring Engine allows runtime user specified modifications. A user can specify to add/remove elements from currently running protocol and also start/stop/jump in the configuration interactively (for instance, over the serial or Ethernet link). For example, when a user inputs `add_CarrierSense`, an entry for the linked list is created with element `CarrierSense` and inserted to the list according to pre-defined component dependencies, i.e., `CarrierSense` should be inserted before the `SendPacket` component. The inter-dependency among components is kept in a table. There are three possible relationships between any two components: 'insert before', 'append after' and 'independent'. Our approach of injecting code blocks in an already executing code and constructing protocols on the fly from a set of building blocks is not limited to a pre-selected set of MAC protocols and is hence more flexible than the Multi-MAC [57] scheme. Besides user triggered reconfiguration, Wiring Engine also facilitates runtime auto-adaptation of the MAC protocol based on the same principle as described above. A set of pre-defined rules can be used to modify the MAC protocol at runtime as we show through a reconfiguration example in Section 4.4.5.

4.4.3 Meta-Compiler Toolchain and Support for Multicore Platforms

In order to simplify construction of MAC protocols through binding a limited set of components we have developed a user-friendly MAC specific language, where a user can construct a protocol in a few lines of code. We extend the idea of composing MAC protocols at runtime using Decomposable MAC Framework with a rich set of tools including a meta-compiler and support for multi-threading on many-core platforms. We refer our toolchain as TRUMP [140, 141]. An illustration of the MAC realization process is shown in Figure 4.4. A user developed code in domain specific MAC language or auto-generated code through a GUI based interface [141] is downloaded onto the target platform using a serial communication interface. The host compiler parses the code and maps it to the corresponding MAC components. It also checks the implementation logic and reports errors like the violation of inter-component dependencies, etc. Wiring Engine controls the state-machine of a MAC protocol (i.e., coordinates the control and data flow) and executes the MAC components. A resource manager in Wiring Engine efficiently links different MAC components and a logic controller governs the logical execution of components by respecting their dependencies. The resource manager also supports parallelization and prioritization of processes on multi-core architectures.

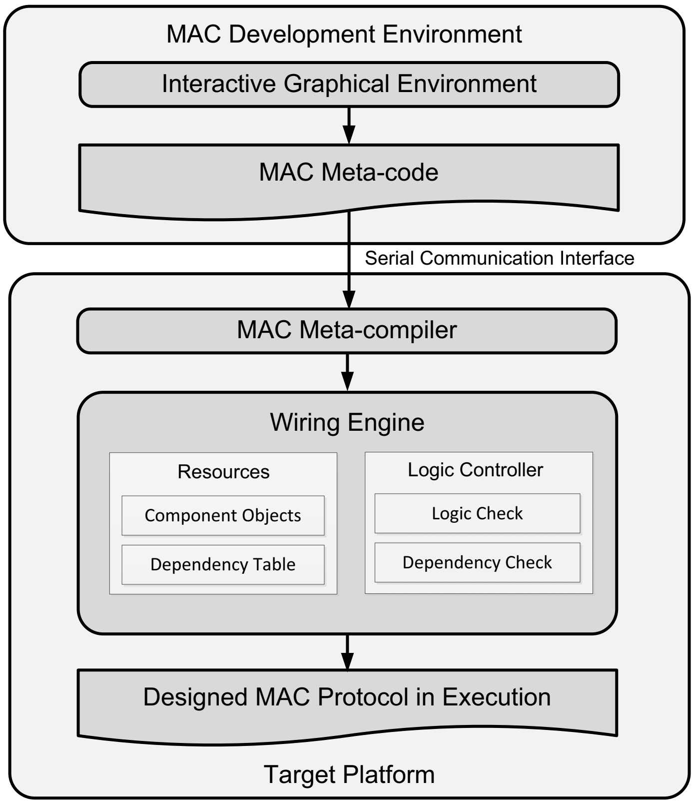


Figure 4.4: A compiler toolchain assisted approach to MAC realization in Decomposable MAC Framework.

4.4.4 Prototype Implementation on WARP Board

We have implemented our framework on WARP boards [15] using the OFDM Reference Design v.14 [165]. All the fundamental MAC components in the framework are implemented in the FPGA, except RNG, which is implemented in software on the PowerPC core. The virtualization of the fundamental components through flexible wrapper APIs and Wiring Engine implementation is carried out in software running on the PowerPC processor. In our framework, standard components such as timers, memory resources, serial interfaces and GPIOs are combined with the state-of-art radio control. We have also exposed an extended radio functionality to users such as setting the transmit power levels, channel selection, receiver sensitivity thresholds, setting modulation schemes, etc. These additional interfaces facilitate cross-layer interactions. For instance, our implementation of CogMAC [93] using the framework lavishes from such a close PHY-MAC interaction. In order to avoid polling delays, we have modified the basic hardware reference design to include an interrupt controller for all the components available to the framework. Events such as the reception of a packet or CRC failure, are combined with user-defined Interrupt Service Routine (ISR) functions. The interface for each object is standardized to ease the debugging process.

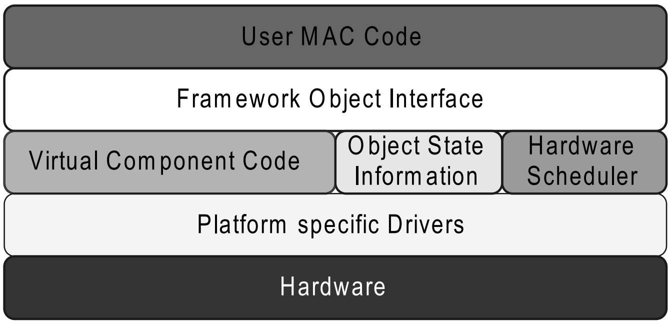


Figure 4.5: Hardware abstraction layering of the framework.

Particular hardware requirements include, for instance, granularity of timers or type of modulation schemes, etc., which can be modified at runtime.

Code portability and ease of MAC designing are achieved through our high-level hardware abstraction architecture as shown in Figure 4.5. More complex MAC designs can be envisioned on a platform, where the number of required resources of a particular type exceeds the number of available hardware components. A user in this case is exempted from the resource management tasks as the framework keeps state information of all the resources. Timers are a typical example of such scheduled objects. In the User MAC Code, timer objects may be generated by specifying object properties - a timer may either be periodic or fires only once and the granularity of the timer is in the scale of milli- or microseconds. The Framework Object Interface manages these objects and destroys them upon expiration of their lifetime. Instead of assigning a particular hardware timer on the WARP, the Virtual Component Code only updates the Object State Information. For this reason, an update to the duration of the timer object causes an update in the object state information. Only if inherently hardware-dependent operations, such as starting the timer, are requested by the user, the hardware scheduler allocates real resources. Interaction with those resources is carried out through platform-specific drivers that interact with the available hardware. Other resources in our implementation are RNG and CRC units. In traditional designs for adaptive MAC frameworks, the entire hardware platform is reset upon changes in the code. On the other hand, component-wise resetting allows a quicker adaption of the new code and lowers black-out times of the hardware in our framework. The object-persistency is supported in the framework through a split-phase resource initialization.

4.4.5 Performance Evaluation on WARP Boards

This section describes the code reuse across different MAC implementations on our framework, the reconfiguration response time and the memory footprint of our prototype implementation. Different GCC compiler optimizations were also considered for the generated code efficiency.

Component Reuse

Table 4.3 lists the MAC component reuse in realizing various protocols on the WARP boards. The components listed in italics are the primary components. It can be observed that certain components are used multiple times within a particular protocol, which advocates the idea of realizing the key atomic functionalities in hardware for higher computing and communication efficiency. From the perspective of a designer, a MAC scheme becomes simpler to implement if the framework provides higher support for secondary level components. We have realized CogMAC [93], a decentralized spectrum agile cognitive MAC protocol with fairly complex multi-channel operation in a fast and easy way using the framework.

Table 4.3: Component reuse for MAC realizations on WARP.

Component	Aloha	Pure_CSMA	B-MAC	IEEE 802.11	S-MAC	CogMAC
Send Packet	0	0	1	3	4	5
Expect Frame	0	0	1	2	3	4
RTS/CTS/DATA/ACK	0	0	0	1	1	0
Random No. Gen.	0	0	0	0	1	2
Timer	0	0	2	3	2	5
Carrier Sensing	0	1	1	1	1	4
Channel Switching	0	0	0	0	0	7
Radio States	0	0	2	2	3	4
SendFrame	1	1	0	0	0	1
ReceiveFrame	1	1	0	0	0	0

Code Size and Memory Footprint

Wiring Engine is implemented in software on the PowerPC core on WARP board, therefore we have analyzed the ELF (Executable and Linkable Format) binary targeted for PowerPC. Since compiler optimizations directly influence the memory use [166], we have analyzed ELF files generated with different optimization options. Table 4.4 lists the memory footprint break-down for a pure CSMA MAC protocol developed using our framework without Wiring Engine in a monolithic way (left) and using Wiring

Table 4.4: The memory footprint (in bytes) of a pure CSMA MAC protocol implementation without using Wiring Engine (left) and with Wiring Engine (right).

Level	.text	.data	.bss	**Total**	.text	.data	.bss	**Total**
Opt 0	109750	1704	17812	**129266**	114642	1704	17888	**134234**
Opt 1	97246	1704	17796	**116746**	100946	1704	17864	**120514**
Opt 2	98226	1704	17796	**117726**	101886	1704	17864	**121454**
Opt 3	102914	1704	17796	**122414**	106738	1704	17864	**126306**

Engine (right). Please note that the memory usage of Wiring Engine depends upon the number and type of components used in a particular MAC and its state-machine logic. Therefore, for our analysis, a pure CSMA MAC realization (with only three components) with and without using Wiring Engine was considered. It can be observed from the table that Wiring Engine imparts only a slight memory overhead in the uninitialized data contributed to the program memory (.bss) and the code size (.text). Low memory footprint makes the framework also relevant to component-based network implementations and cross-layer design in a broader range of embedded network applications.

Software versus Hardware Implementation

We noted that implementing a MAC protocol purely in software fails to achieve timing requirements and therefore results in poor performance characteristics. On the contrary, a hardware implemented MAC protocol, e.g., IEEE 802.11 on a commercial NIC, though optimized for the design, does not permit flexibility and block most of the upgrade possibilities that could be made through software updates. We have approached this problem by implementing selective basic functionalities in hardware while providing flexible APIs and virtualization in the software (cf. Section 4.4.4). This enables achieving hardware acceleration without compromising on flexibility. A one-to-one comparison in terms of the response time and execution speed (which directly affects the latency and throughput [153]) of a MAC implementation on a WARP board with an implementation on a standard IEEE 802.11 NIC or with a GNU radio [158] based software implementation would be uninformative. Therefore, in order to study the benefits of implementing the key atomic functionalities in hardware, as examples, we have carried out the implementation of CRC and RNG components both in a custom hardware as well as in the software on the ARM926 core. A performance gain of ca. 20 and 8 times has been observed in the speeds of computing a CRC and generating a random number through the hardware implementations, respectively.

Response Time Overhead

A fast response time is achieved through a function pointer based implementation approach followed in Wiring Engine. Figure 4.6 shows the average time required for a MAC to reconfigure itself. Please note that since the execution time for different components is different, here we studied the case when a configuration is updated (component(s) removed or inserted) instantly without waiting for the completion of the currently executing component. In other words, the executing component has an 'independent' entry in its dependency table. We considered the elapsed time for Wiring Engine to update the state-machine of the MAC upon receiving a new configuration. The graph shows the average results for 10 000 samples at different optimization levels. The bar graph also shows the percentages of the individual reconfiguration delays. Our results indicate a fast reconfiguration response, which makes it suitable for designs with time-critical requirements.

In addition to the auto-reconfiguration response time, we have also measured the latency associated with user-driven configuration updates of the framework implementation on WARP boards. Figure 4.7 shows the average response time for different

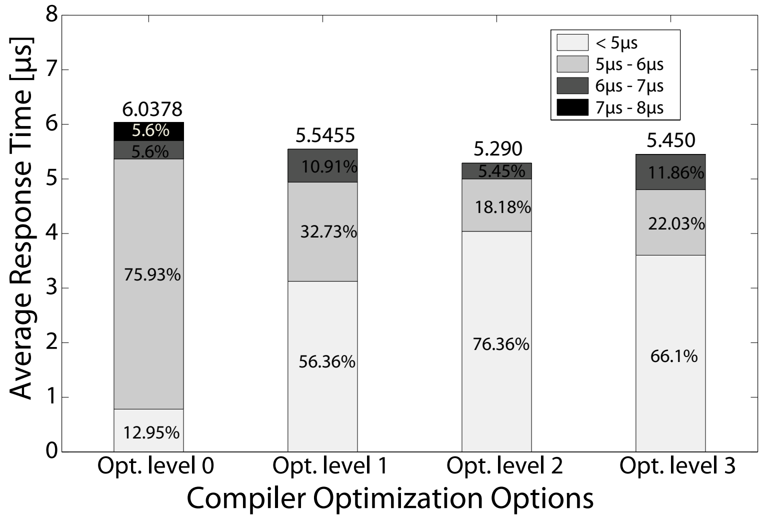


Figure 4.6: Response time for MAC auto-configuration on WARP boards.

number of user triggered reconfiguration operations (insert, remove, change the order). The user updates were sent from the PC to the WARP board over the UART interface. The delay in UART communication was later on deducted to obtain the reconfiguration timings. It can be observed that for all the optimization levels, the required reconfiguration time shows a linearly increasing behaviour with increasing number of reconfiguration operations.

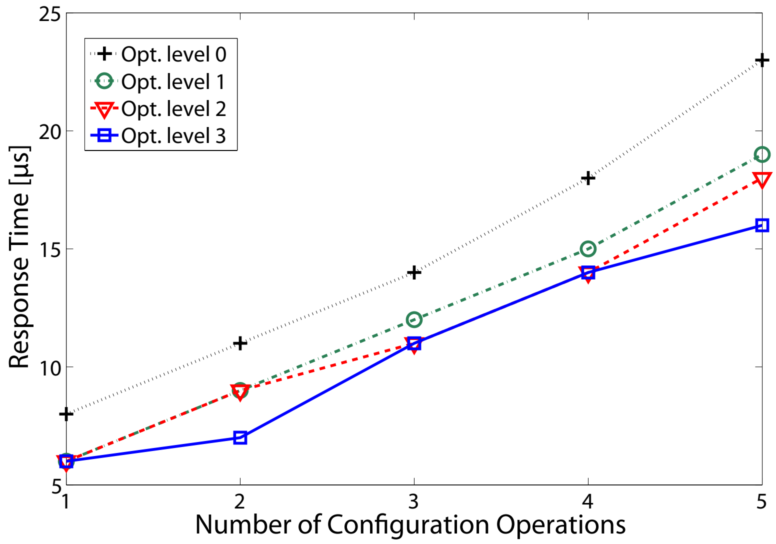


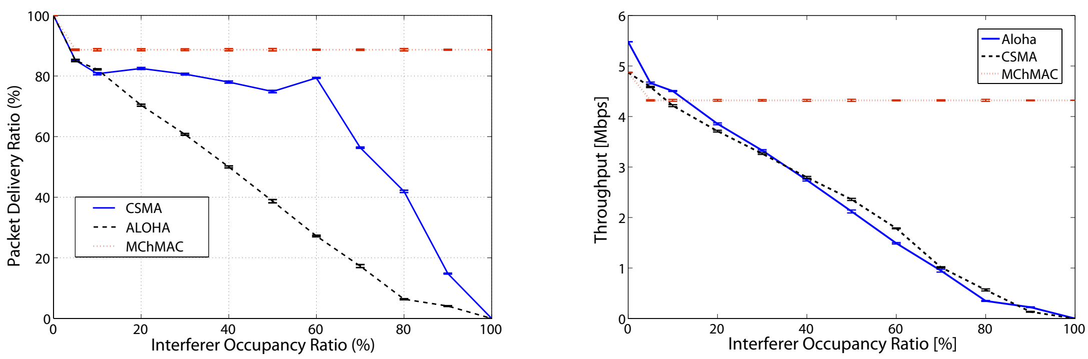
Figure 4.7: Average response time for user based configuration of MAC.

Runtime MAC Composition and Reconfiguration

One may note from the above results that the reconfiguration time is in the order of a few microseconds. Therefore, Wiring Engine based MAC realization approach enables a new dimension MAC optimization through protocol composition at runtime. This is achieved by simply changing the protocol composition by adding, deleting or modifying the combination of components forming a protocol.

Figure 4.8 shows the performance of ALOHA and CSMA MAC protocols achieved on WARP board, measured with a single transmitter-receiver pair. CSMA MAC protocol can be transformed from ALOHA by simply adding a `CarrierSense()` element with an `if/endif` conditional statement. Our measurements show, as expected from theory and experience that with a single channel single transmission flow, when the interference occupancy ratio is below 10 %, ALOHA is the most efficient choice. However, when the interferer occupancy ratio is between 10 % and 35 %, a choice between throughput and packet delivery ratio has to be made. When the performance of ALOHA and CSMA becomes substantially degraded, a `ChannelSwitching()` component is introduced where other channels supported by a transceiver can also be utilized. We refer to this as a Multi-channel MAC (MChMAC). In our implementation, the overhead for switching a channel and establishing a link between the transmitter-receiver pair result in 5 % drop in the throughput.

Using the above mentioned characteristics (cf. Figure 4.8) as a prior knowledge, we have carried out an experiment with an optimization goal of maximizing a weighted combination of throughput and packet delivery ratio. The interferer occupancy ratio in a particular channel was varied from 0 % to 80 % as shown in Figure 4.9 while the rest of the spectrum was kept free. The protocol was initially composed as ALOHA with the assumption of low interference ratio as it gives the highest performance. Unlike pure ALOHA, a `ReadRssi()` component was used to monitor the channel condition in an online fashion and feed it to the optimizer. The achieved throughput and packet



(a) Packet delivery ratio. (b) Throughput.

Figure 4.8: Packet delivery ratio and throughput of Aloha, CSMA and MChMAC protocols running on WARP board with respect to interference occupancy ratio.

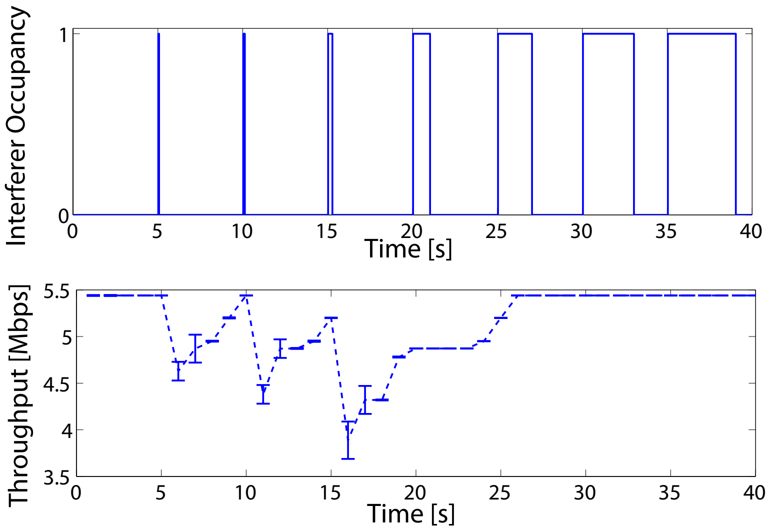


Figure 4.9: The optimized throughput through runtime MAC composition.

delivery ratio experienced an immediate drop when an interference was encountered and stayed high when the channel was free. When the interference ratio went to 10 %, a `CarrierSense()` component was added to form a CSMA protocol and when the interference ratio had risen to 15 %, a `ChannelSwitching()` element was added in the protocol configuration to form MChMAC. However, since a good channel condition was experienced afterwards, the components `ChannelSwitching()` and `CarrierSense()`, were later on removed from the MAC linked list. Consequently, a high throughput performance was regained on the new interference free channel as the MAC behaviour gradually switched from MChMAC to CSMA and then eventually to ALOHA. This transformation can be observed from the figure: Between 15 s and 20 s, MChMAC behaviour is adapted which is transformed to CSMA-like between 20 s and 25 s and after 27 s, ALOHA like behaviour is continued. We have observed a performance gain of up to four times when the interferer occupancy ratio goes over 80 %. This experiment proves that MAC optimization can be achieved through protocol composition at runtime on commercially available SDR platforms. We have demonstrated the runtime composition of protocols and reconfiguration [167]. Our demonstration showed that runtime composition of MAC protocols is realizable on commercially available SDR platforms and this certainly opens up new possibilities for MAC optimization.

4.5 RAPID AND EASY PROTOTYPING THROUGH MAC-PD

We have developed an interactive graphical user interface to support design of MAC protocols. We refer this tool as MAC Protocol Designer (MAC-PD). MAC-PD allows designers to express MAC protocols through flowcharts composed of various MAC subcomponents. It is able to directly generate the executable target code for different platforms without requiring code development efforts. It takes off the burden of learning programming language and platform details from the designers and eases the MAC designing process through an interactive drag-and-drop based user friendly environment.

MAC-PD generates the MAC meta-code targeted for WARP boards from the designed flowcharts. The code can be compiled and executed using our MAC meta-compiler assisted toolchain, which we refer to as TRUMP [140, 141]. In order to demonstrate the universality of our approach we have also provided a port and targeted compiler that supports resource limited sensor nodes, i.e., our approach is not limited to powerful WARP SDR-board as a target platform. MAC-PD is able to generate TinyOS 2.x source code (nesC) [63] and executable scripts, which are directly compiled and downloaded onto the COTS sensor node platforms. MAC-PD supports commonly used sensor node platforms such as TelosA [168], TelosB [14], MICAz [169], BSN-node [170] and Intelmote2 [171]. The XML model files generated from the flowcharts are used for storing the re-loadable design developed by a user. Thus users can simply reuse, replicate and modify previously saved design sessions. A particular design can be added to the MAC-PD component library as a new reusable component. This helps in minimizing future efforts for protocol design.

4.5.1 Implementation Architecture

Figure 4.10 shows the layered architecture of the MAC-PD, where each layer performs its tasks in a top-down work flow. The layers are decoupled through well-defined interfaces. In the following, we describe the key parts of MAC-PD in detail.

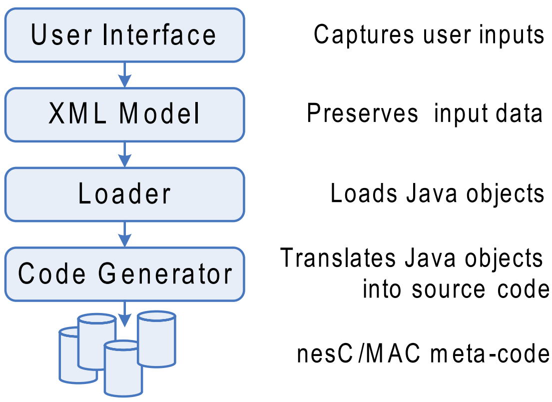


Figure 4.10: Layered architecture of MAC-PD.

User Interaction

The top most layer provides a flexible and user-friendly interface for MAC design. It is based on the Java-Swing GUI (cf. Figure 4.11) and allows interactive 'drag-and-drop' feature for MAC development. All the basic and secondary MAC components are made available to a user in the GUI. Furthermore, state-machine logic for a MAC design including arithmetic, Boolean and logical expressions are also exposed to the user in the GUI. A designer can simply select components, drag them to the provided drawing grid and connect them to form a particular protocol. The parameter settings of a component and variable assignments can be carried out from the dialog screens. The User Interface layer gathers information of the state-machine and logic flow of the MAC protocol from the user. Graph data structures are maintained to store the TinyOS 2.x syntax and MAC language specifics corresponding to the flow-charts. While parsing, the User Interaction layer processes the information of the graph and reports the missing or invalid data. It also validates the basic design and reports illogical connections.

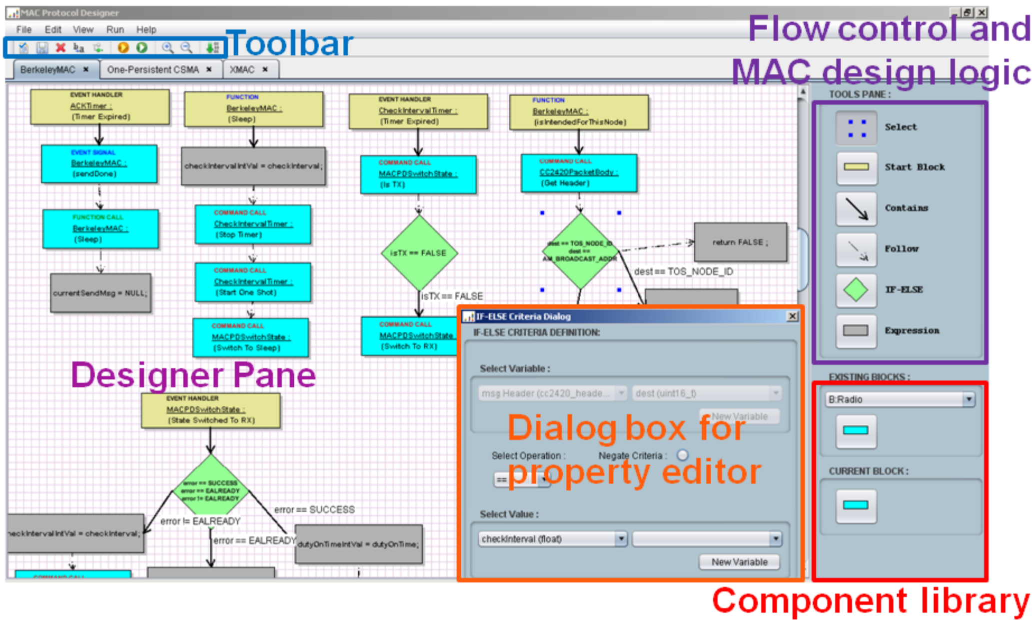


Figure 4.11: Snapshot of the MAC designing tool.

XML Modeling

This layer maintains an XML model of the user design. XML tags are generated for each node, which preserves the state of a node and its relationship with other nodes in the graph. An XML model is used for storing the MAC meta-language syntax or TinyOS 2.x constructs in the form of a tree structure. All the user designs are preserved in XML so that these can be utilized and updated in possibly multiple sessions without having the need to redesign everything from scratch.

Code Loader

The Loader Layer interacts directly with the XML model layer to load an XML design in RAM in the form of Java objects. These objects are populated with the state information of the nodes from the XML model and are used in the auto code generation process.

Code Generation

The Code Generator Layer parses and translates the Java objects into either TinyOS 2.x source code, which is compiled and directly deployed onto the targeted platform or the MAC meta-code, which is compiled and executed using TRUMP on WARP boards. The auto-generated code is human readable so that advanced users may easily make modifications and build custom applications. We have implemented a number of MAC protocols using the framework including S-MAC [29], B-MAC [36], MFP-MAC [43] and X-MAC [44]. The implementations provide a well-defined MAC API to allow interaction of the MAC to the application and to control the MAC parameters. For instance, the interface of the MFP-MAC below contains the common functions to send/receive packets and to control the LPL operation of the MAC protocol.

Since the auto-generated MAC meta-language code through MAC-PD for WARP boards corresponds very closely to their hand-coded implementation, we do not evaluate the code comparison. Instead we focus only on the evaluation of the auto-generated MAC code for sensor node platforms.

4.5.2 Performance Evaluation on Sensor Node Platforms

There is a common belief that auto-generated embedded code results in large memory footprint and lacks performance characteristics. However, our work shows that careful component design approach based on functional commonalities among protocols gives similar performance metrics on COTS sensor nodes as the monolithic MAC implementations and also allows fast prototyping.

We have conducted the analysis of component reuse, proportion of the total number of lines of code to the reusable lines of code, memory footprint and the achieved throughput for well-known MAC protocols implemented using MAC-PD. We have also carried out comparative empirical studies on these metrics for different MAC protocols to their monolithic counterparts and those developed through MLA [172].

Component Re-usage

MAC-PD allows implementation of MAC protocols by simply connecting the functional components together. The idea is to keep the protocol specific code to a minimal by providing flexible and reusable MAC components. Table 4.5 lists the components which are used in the implemented protocol realizations. The components listed in italics are the basic components. One should note that the radio control (functionalities: turning on/off the radio, switch state TX/RX), sending and receiving frames are fundamental to all the protocols. Timer instances and carrier sensing functionalities are also common to all the contention based protocols. Having a rich set of reusable

Table 4.5: Component reuse for MAC realizations through MAC-PD.

Component	B-MAC	X-MAC	MFP-MAC	S-MAC	Aloha	CSMA
Send Preamble	1	0	1	0	0	0
Expect Frame	0	0	0	1	0	0
Send Packet	0	0	0	1	0	0
Rand. No. Gen.	0	0	0	0	0	1
BEB	0	0	0	1	0	0
LplCoordinator	1	1	1	1	0	0
SendFrame	1	1	1	0	1	1
ReceiveFrame	1	1	1	0	1	1
RadioControl	1	1	1	1	1	1
Timer/Alarm	3	4	3	6	2	2
Carrier Sensing	1	1	1	1	0	1

components reduces the implementation efforts. This is indicated through a high proportion of the reusable lines of code as described later in this section.

Code Line Reuse

Figure 4.12 shows the total number of lines of nesC code and the number of lines of reusable code for different MAC protocols implemented using the framework. Please note that the nesC code is completely auto-generated through MAC-PD and here the unused portion of the code constitutes the binding logic for different components and

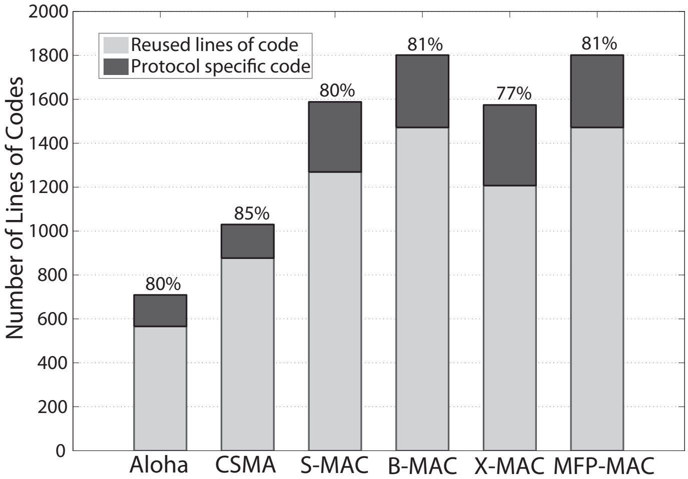


Figure 4.12: Overall number of lines of nesC code and the percentage of the reusable code for different protocols implemented using MAC-PD.

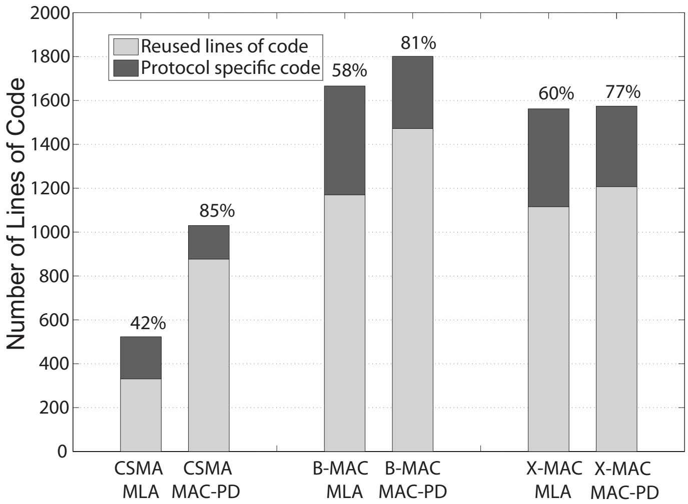


Figure 4.13: Comparison of the overall number of lines of code and the percentage of the reusable code between MLA and MAC-PD.

the protocol specifics. It is remarkable that the overall proportion of the code reusability across different MAC implementations is approximately 80 %. This indicates that our framework gives a high degree of code reusability and provides a fast means for prototyping different MAC solutions.

Figure 4.13 compares the proportion of the nesC code reuse for our framework to MLA [150] approach. It is evident that our approach allows a higher proportion of code reuse as compared to MLA. For the three protocols considered, the code reuse for MAC-PD is above 80 % while it is approximately 55 % for MLA. The total number of lines of code for the MAC implementations in MLA is approximately 75 % of the auto-generated code through our framework. MAC-PD gives a bigger total number of lines of code due to the higher component modularization offered.

Memory Footprint

MAC-PD is able to generate the nesC code for TinyOS 2.x targeted for a sensor node platform. The MAC code is embedded with the rest of the components and with the TinyOS 2.x scheduler as a single executable binary. Table 4.6 compares the memory footprint on TelosA, MICAz and Intelmote2. It is worth noting that TinyOS 2.x binary along with the MAC protocols occupies less than approximately 5 % of the available RAM and ROM on these platforms. Table 4.7 shows the comparison of the memory footprint for different MAC implementation among the monolithic implementation, MLA based implementation and the implementation using our framework. In order to ensure fairness in our analysis, we used the same driver stack (TinyOS 2.0.1) for CC2420 radio and the same test application for the three design approaches. The table shows that the MAC implementations using MAC-PD are clearly more memory

Table 4.6: Memory footprint (bytes) on different sensor node platforms.

Protocol	TelosA		TelosB		MICAz		Intelmote2	
	RAM	ROM	RAM	ROM	RAM	ROM	RAM	ROM
B-MAC	620	14346	620	14500	614	16228	708	27238
X-MAC	758	15472	758	15626	733	15152	852	27688
CSMA	692	11712	692	11866	671	13212	780	22786
S-MAC	778	12978	778	13132	751	14752	880	24524
MFP-MAC	620	14346	620	14500	614	16228	708	27238
Aloha	616	11400	616	11554	601	12578	706	22140

efficient than MLA and despite auto-generated code, their memory footprint closely resemble to those of the monolithic implementations.

Table 4.7: Comparison of memory footprint on TelosB.

Protocol	Monolithic		MLA		MAC-PD	
	RAM	ROM	RAM	ROM	RAM	ROM
B-MAC	922	17586	921	18338	758	18074
X-MAC	866	17408	864	18000	690	17660
CSMA	330	11120	334	11928	632	14026

Goodput Analysis

In order to observe the performance characteristics of MAC protocols developed through MAC-PD, we replicated the experimental scenario for B-MAC [36]. A receiver was placed in the center while the number of equidistant transmitters (at a radius of 2 m) around the receiver was gradually increased. We counted the number of packets unicast from each transmitter and correspondingly received to measure the throughput. A fixed sampling period of 100 ms and a packet size of 32 bytes were used, giving a theoretical maximum throughput of 2.415 kbps. Each experiment lasted 120 s and was repeated thrice. The mean aggregate goodput in Figure 4.14 shows identical trend as have been observed by the authors of B-MAC [36]. With a sampling period of 100 ms and at a Packet Generation Rate (PGR) of 1 ms, the network experiences saturation. Therefore, the aggregate goodput remains similar in the figure despite increasing the number of transmitting nodes. However, for traffic generation rates of 250 ms and 1000 ms, the network is not in saturation and thus with an increasing number of transmitters, an increasing trend in the goodput can be seen from the figure. These results indicate that MAC implementation using MAC-PD gives the expected behaviour [36]. Furthermore, we have compared the goodput of B-MAC implemented using MAC-PD to the implementation using MLA with identical parameter settings, which show that both MLA and MAC-PD exhibits comparable performance characteristics and hence to the monolithic implementation [150].

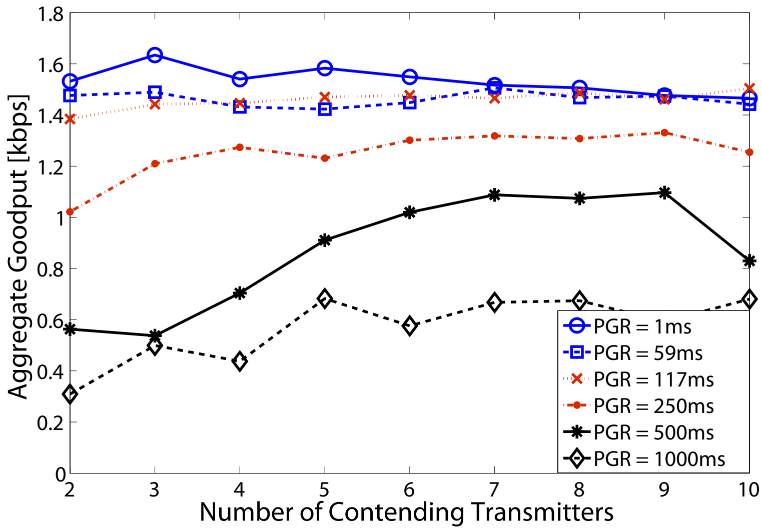


Figure 4.14: Aggregate goodput of B-MAC developed using MAC-PD at different Packet Generation Rates (PGR).

4.6 SUMMARY

Bachir *et al.* have shown that none of the popularly used MAC protocols give a universally optimal behaviour [1]. This advocates designing suitable protocols depending upon the application requirements. Designing and implementing MAC protocols for embedded devices and networks is a tedious task and typically requires domain knowledge of the underlying hardware platform and the operating system. MAC protocols are generally implemented in a monolithic fashion with tight coupling to the hardware platform which limits the room for experimentation, reusability and modification of code. We have developed Decomposable MAC Framework [138], which allows rapid prototyping of MAC protocols from a set of basic functional components and allows high degree of code reusability. We have identified the fundamental MAC components as building blocks and developed Wiring Engine for binding the components to form MAC protocols. Our approach allows software-hardware co-design to enable flexibility as well as hardware acceleration. We have introduced the concept of runtime composition of MAC protocols through Wiring Engine. The developed Wiring Engine coordinates the data and the control flow among components and controls their execution. Function pointer based execution of functional component list enables Wiring Engine to meet timeliness requirements of the spectrum agile and cognitive MAC protocols as we have shown through the prototyping of CogMAC [93]. Extending Decomposable MAC Framework, we have developed an interactive user friendly tool MAC-PD, which allows to "drag-drop-and-connect" components in order to design a particular MAC protocol [139]. Prior knowledge of the platform and the language syntax is no longer required by MAC developers. MAC-PD generates downloadable code for TinyOS 2.x supported COTS sensor node platforms. Furthermore, it also generates MAC meta-language code, which can be compiled and executed on a host WARP board using a meta-compiler toolchain referred to as TRUMP [140, 141]. Experimental performance evaluation for various commonly used MAC protocols indicates that the memory footprint and throughput analysis of the MAC implementations generated through MAC-PD closely resemble their monolithic counterparts. We strongly believe that Decomposable MAC Framework and its design philosophy will allow a wider experimentation room for prototyping new solutions. Furthermore, the ability to compose MAC solutions using TRUMP will give a new dimension to runtime MAC reconfiguration. We have demonstrated an earlier working prototype of Decomposable MAC Framework at [129], the ease of MAC designing through MAC-PD at [173] and TRUMP at [167].

5

CROSS LAYER DESIGN

MAC layer is regarded as a critically important protocol layer in a protocol stack. It resides on top of the physical layer and underneath the routing layer. A MAC layer therefore, plays a key role in determining the overall performance of networked systems. Examining a MAC protocol behaviour in isolation, without considering its interaction with the PHY- and routing layers, is therefore not only incomplete but can lead to wrong conclusions and suboptimal design decisions.

Traditionally, protocols at different layers are decoupled from each other and are designed independently. Although this approach lowers the design complexity, it adds redundant functionalities at different protocol layers. This causes extra processing and signaling across different layers, and increases the network communication overhead. Moreover strict layering may mean that some of the protocol layers do not have access to crucial information or interfaces that are available in other protocol layers of protocol stack. Many research studies in the past five years have shown that there exists a significant potential for improving the performance characteristics of the overall network through cross-layer design techniques where PHY/MAC/routing protocols benefit by sharing mutual information, exploit commonalities and minimize redundant functionalities [174]. Layered network architecture does not provide the desired level of interaction among protocols and restricts information exchange across different protocols in order to collaboratively adapt their behaviour towards a common goal. Adapting to dynamic changes in network, traffic and spectral conditions requires cross-layer design and efficient management of the network resources. Due to resource constrained nature of embedded wireless networks, improving the performance metrics and optimizing the use of resources through cross-layer design evidently becomes highly important.

Cross-layer design involves altering the behaviour of PHY/MAC/routing protocols in order to maximize the performance characteristics of a network. Enhancing network efficiency and supporting application QoS demands typically include increasing the throughput, maximizing the packet delivery ratio, minimizing the latency in data communication, prolonging the lifetime of a network or a combination of the above. Often application QoS demands are expressed through system constraints and/or minimum level of requirements. An optimizer tries to adapt the behaviour of protocols through mutual sharing of information towards achieving the global objective of increasing network efficiency while satisfying the system constraints. Different optimization techniques such as mathematical programming, convex optimization, machine learning and heuristics-based methods have been actively used by network researchers [174, 175]. Each of these techniques has its strengths and weaknesses [176] depending upon its usage in a particular application scenario. In this chapter, our focus is not on the optimization methods but instead on different approaches for cross-layer design. Enabling cross-layer design requires changes to MAC protocols as well as to the physical and routing layers. We will describe three main cross-layer design approaches for embedded wireless networks. These approaches are validated empirically through MAC/routing and PHY/MAC optimization on real node testbeds. We have carried out a comprehensive study of cross-layer interaction among protocols and observed that our developed methods can lead to considerable improvement in network efficiency. The results obtained have a practical significance because only a few cross-layer optimization studies have so far been conducted on real node testbeds.

- **Optimizing the performance of a network by sharing information across different layers through parameters tuning**. This is achieved by appropriately setting the parameter values of protocols exposed to other layers through well-defined interfaces. Section 5.1.1 describes this approach with the help of MAC/routing and PHY/MAC optimization for our case studies on Indriya [13] TelosB testbed and a WARP [15] board testbed.

- **Exploiting common functionalities among protocols in a network stack and removing redundancies.** Eliminating redundant functionalities across PHY/ MAC/ routing protocols does not only promise efficient memory usage but also reduces the communication and control overhead. It leads to energy savings, low latency response and bandwidth conservation. Section 5.2 presents our approach with a detailed case study of MAC-routing cross-layer design and optimization on Indriya TelosB testbed.

- **Exploring alternate interaction paradigms instead of the traditional client-server model.** The resulting energy consumption and latency associated with data communication depends upon the way nodes interact with each other in a network. In wired and static networks, traditional client-server model performs efficiently. However, in applications involving mobility of nodes, network dynamics and distributed data sources (e.g., wireless sensor networks, data centric networks, etc.), client-server model becomes inefficient and leads to high communication cost and increased latency. Section 5.3 investigates Publish/Subscribe (P/S) interaction paradigm for application scenarios in order to achieve scalable, robust, and energy efficient communication.

This chapter is mainly based on our articles [17, 140, 177, 178], which were published during this work.

5.1 OPTIMAL PARAMETER SETTINGS

Decoupling of PHY/MAC/routing protocols through layering restricts sharing of information across protocol layers. Consequently, it limits the potential benefits of information sharing in order to improve the network performance characteristics. Various studies have shown that using the cross-layer methods and jointly optimizing different protocols in a protocol stack considerably increase the overall network efficiency [179]. In order to allow sharing of information, the idea of stackless network was introduced [152]. However, completely dismantling a PHY/MAC/routing stack and merging all the layers is unpractical [180]. In order to share information across different layers and exploit dependencies and interactions among PHY/MAC/routing protocols, tunable parameters are exposed at each protocol layer so that appropriate parameters setting can be carried out. This way a protocol stack consists of pseudo-layers while the information can be shared across protocols and parameters can be tuned through well-defined interfaces. Setting parameters appropriately across protocol layers is one of the commonly used methods for achieving cross-layer optimization. In order to apply parameter settings, their relationship to the system model and the behaviour on network performance should be known.

5.1.1 Selection of MAC-routing Protocols and their Parameters at the Pre-deployment Phase

The application requirements and deployment conditions vary across different applications. There is no well-established knowledge and understanding which sets of protocols and their parameters suit best to a particular application scenario. This is challenging mainly because of two reasons: First, not all the combination of protocols are realizable and therefore, dependencies and interactions among protocols have to be considered while selecting a protocol stack. Second, the application demands and deployment conditions directly influence the protocol parameters and therefore these should be taken into account. During the past years, a large number of protocols have been proposed and ironically this has complicated the selection of an optimal protocol stack for a particular application. In this context, we have developed a new framework known as Component based Optimization for Networks with deployment Feedback (CONFab). It aims at facilitating application driven selection of protocols and their parameters at the pre-deployment phase.

CONFab maps user requirements to the application and deployment conditions in order to recommend the most appropriate protocol stack and the optimal protocol parameters based on the previous user experience and performance feedback. The optimization task is essentially the best possible selection of protocols (or components) and their parameters. CONFab learns and improves its knowledge-base using feedback on performance metrics of a deployed system. The performance feedback is stored in ontology which is queried using a reasoning engine.

CONFab Design

CONFab allows a user to specify an application scenario, system constraints and the desired level of performance characteristics. The selection of an appropriate networking stack is carried out by filtering the system constraints in order to achieve the best performance metrics based on the past user experience. CONFab extracts the perform-

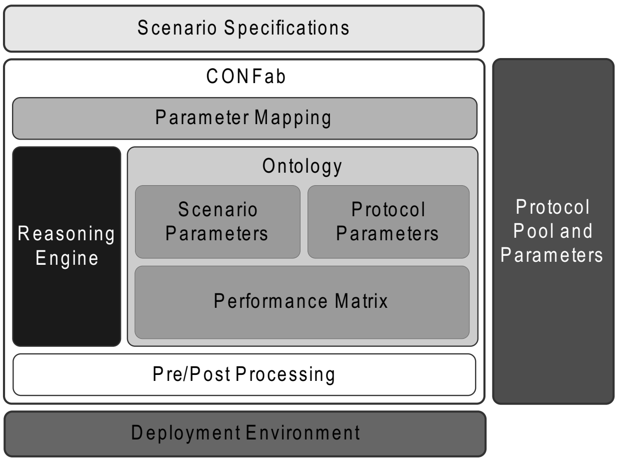


Figure 5.1: Block diagram of CONFab.

ance measurements from an already deployed system to improve its knowledge-base.

Figure 5.1 shows the block diagram of CONFab. The *Protocol Pool and Parameters* block provides a set of MAC and routing protocols. The *Scenario Specifications* block provides a list of parameters to a user for defining a particular application scenario. The *Parameter Mapping* block specifies the mapping of scenario parameters to the performance matrix, which stores the processed data gathered from a deployed protocol stack. Furthermore, this block also binds protocol parameters to the performance matrix. The *Ontology* module stores scenario parameters, protocol parameters and performance matrix. The ontology provides a semantic representation of knowledge as a set of concepts within a specific domain and the relationship between these concepts. The *Reasoning Engine* applies reasoning techniques on the ontology for the selection of an appropriate protocol stack. The *Pre/Post Processing* block is responsible for analyzing the feedback on performance metrics from a deployed network to be reasoned upon in the future.

Figure 5.2 shows the CONFab process flow: A pool of protocols is added to the system. Each protocol is specified through a set of parameters (which are saved using ontology) and is bonded with a performance matrix class. Each instance of the matrix class represents a single scenario deployment and consists of a routing protocol, a MAC protocol and protocol specific parameters used in the network deployment. The scenario is specified after adding a pool of protocols. CONFab allows expressing a scenario using a set of parameters, which are saved in the ontology. The scenario constraints are then applied to the performance matrix. The reasoning engine extracts the test cases, which fulfil the scenario constraints and recommends them as appropriate MAC-routing stacks with their protocol parameters. The user makes the final selection from the recommended stacks and deploys it to a network. The gathered data on performance metrics from the deployed MAC-routing stack is processed by CONFab and merged with the performance matrix. In the following, we describe the key blocks of CONFab in detail.

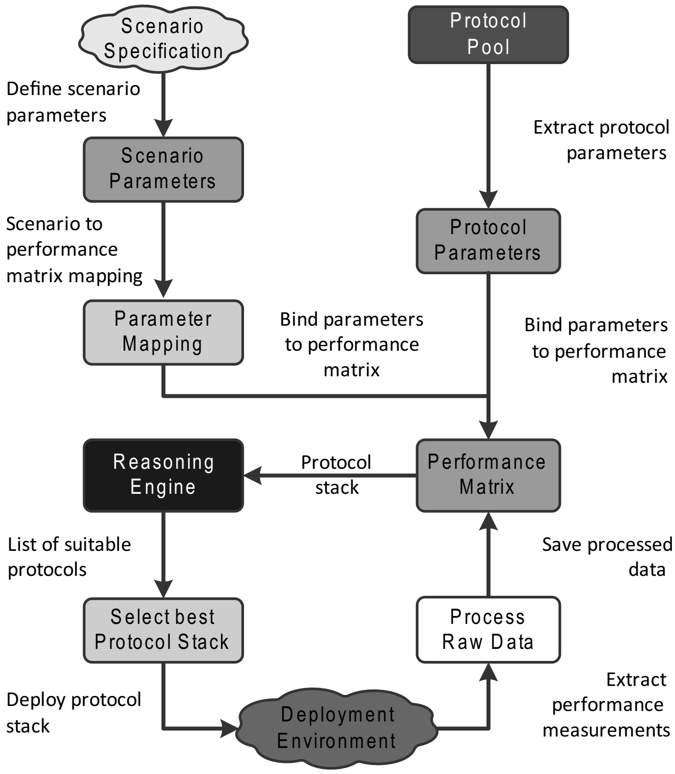


Figure 5.2: Process diagram of CONFab.

Scenario Parameters
The diversity of embedded network applications makes it challenging to define all the parameters and their settings which can fully capture an application scenario. In addition to the life-time of a network, application data rate, fault resilience and network dynamics are the most prominent parameters. Therefore, CONFab aims at supporting the most common application parameters as listed in Table 5.1. Currently supported set of values are listed in the table. Although, additional parameters and/or values can easily be added.

Table 5.1: CONFab scenario parameters and values.

Parameter	**Values**
Number of nodes	< 30, $30 - 70$, > 70
Power source	AAA, AA, C-battery, D-battery
Number of power sources	1, 2, 3, 4
Desired life-time	1 week, 1 month, 4 months, 8, months, 1 year, 2 years
Topology dynamics	High, Medium, Low, None
Fault tolerance	High, Medium, Low

Protocol Parameters
CONFab describes a set of parameters to characterize routing and MAC protocols. A user can filter a smaller subset of protocols based on these parameters. The filtered out protocols are then used for reasoning to obtain the recommended stacks. Parameter values of both routing and MAC protocols are extracted from experts' inputs. Routing protocol parameters are mainly acquired from the work [181]. MAC protocol parameters are extracted from [2, 3].

Performance Matrix Parameters
The performance matrix keeps track of the processed data obtained from a deployed protocol stack. Table 5.2 shows the possible values each performance matrix parameter takes.

Table 5.2: Values of each performance matrix parameter.

Parameter	**Value**
Number of nodes	20, 40, 80
Application rate	High, Low
Periodic sleep interval [ms]	100, 250, 500, 1000, 2000
Channel sensing duration [ms]	11
Packet success ratio [%]	$0 - 100$
Network energy consumption [J]	$50 - 5000$
Topology setup duration [s]	$1 - 50$
Routing protocols	CTP, S4
MAC protocols	B-MAC+, X-MAC, TrawMAC(w & w/o HW ACK)

Parameter Mappings
Depending upon a given scenario parameters, CONFab filters out a few combinations of MAC-routing protocols that suit the application from the complete set of protocols in the pool. The filtered out protocols and scenario parameters are mapped to the performance matrix. The reasoning engine performs reasoning on the performance matrix and infers the suitable protocol stack from the filtered out protocols. For instance, with a scenario parameter, number of nodes < 30 (cf. Table 5.1), the number of nodes in performance matrix parameter can get a value of 20 (cf. Table 5.2). Similarly, if the scenario allows only a low fault tolerance (cf. Table 5.1), the packet success ratio in performance matrix parameter (cf. Table 5.2) is set to be between 90 % and 100 %.

Performance Evaluation and Validation
Our prototype implementation of CONFab uses Web Ontology Language (OWL) and Protégé[1] framework to design ontology. CONFab uses Fact++ [183] reasoner. The

[1]Protégé is an open-source platform that provides a suite of tools to construct domain models and knowledge-based applications with Ontologies [182].

Jena API in Protégé framework uses the standard XML based Description logic Implementation Group (DIG) interface to connect to the Fact++ reasoning engine.

We have carried out an experimental campaign for $700+$ hours on Indriya testbed [13] to establish the knowledge-base for CONFab. In particular, we have varied the parameters of the MAC protocols (B-MAC+ [45], X-MAC [44], TrawMAC [17] with HW ACK, TrawMAC without HW ACK) and routing protocols (S4 [184], CTP [185]), generate different amount of application traffic and vary the network size. Each of the different combinations of tests lasted for 20 min and were repeated three times in order to obtain better statistical average. In order to systematically extract the performance metrics from the network, software adapters for monitoring energy consumption and packet transmission/reception statistics at the MAC and routing protocols are plugged-in at each node. The data logging adapters were first verified on a small scale in-house network against actual measurements. The performance data logged for each experiment on the testbed was fed to the CONFab knowledge-base.

In order to cross-validate the CONFab suggested protocol stack for a particular application against the actual testbed results, we have conducted a case study of the habitat monitoring system [12]. The scenario specifications are extracted to be 50 fixed nodes each powered by four C-batteries. The required application data transmission rate and the overall desired lifetime were considered to be one packet every 50 s and 4 months, respectively. Table 5.3 shows the user specified scenario parameters to CONFab.

Table 5.3: CONFab scenario parameter values for habitat monitoring case study.

Parameter	**Value**
Number of nodes	$30 - 70$
Power source	C-battery
Number of power sources	1
Desired life-time	4 months
Topology dynamics	Low
Fault tolerance	High

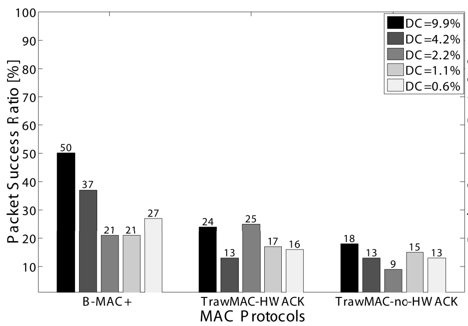
Figure 5.3 shows the screen-shot of the resulting CONFab protocol recommendations with parameter settings and the performance attributes for a network size of 50 nodes. We noted that the CONFab knowledge-base did not contain explicit tests for the suggested 50 nodes wide network. CONFab combines as a weighted average the results of the closely resembling scenarios, i.e., in this case mostly the results of 40 and 80 nodes network with a low application traffic rate. The scenario goal is set to provide a high fault tolerance.

In order to validate the results from CONFab, we have conducted experiments for the suggested 50 nodes network on the testbed at different duty cycles and correspondingly observed the energy consumption and packet success ratio.

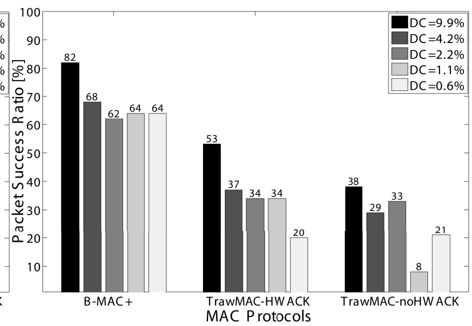
PM	Routing	MAC	Nodes	Sleep Interval [ms]	Success Ratio [%]	Energy Consumed [J] ▲	Topology Time [ms]	App Send Rate
PM_20	CTP	B-MACP	50	2000	78	100	5	Low
PM_18	CTP	B-MACP	50	1000	74	100	8	Low
PM_12	CTP	B-MACP	50	100	91	150	5	Low
PM_14	CTP	B-MACP	50	250	83	150	11	Low
PM_16	CTP	B-MACP	50	500	77	150	5	Low
PM_42	CTP	TrawMAC_HW_Ack	50	100	54	150	4	Low

Figure 5.3: Protocol stacks selected by the CONFab reasoning engine for habitat monitoring case study.

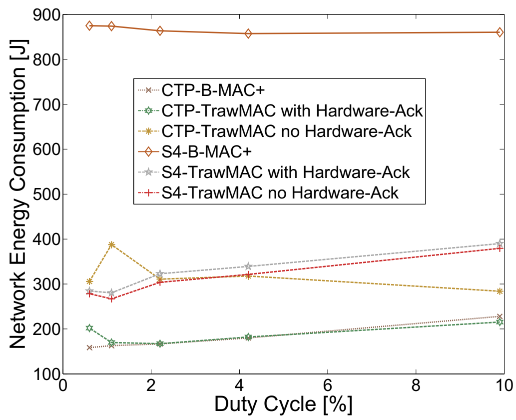
In particular, Figure 5.3 shows that CONFab suggests using CTP with B-MAC+ with a sampling period of 100 ms, i.e., duty cycle value of 0.5 %. Figure 5.4 shows the packet delivery ratio and the energy consumption obtained on the actual testbed for different combinations of MAC-routing stack at different duty cycle values for a duration of 5 min. These results (cf. Figure 5.4b/c) indicate that CONFab is able to suggest the best energy consumption and packet delivery ratio trade-offs.



(a) Average packet success ratio using S4.



(b) Average packet success ratio using CTP.



(c) Network energy consumption.

Figure 5.4: Performance characteristics of MAC/routing protocols at different duty cycle values in a 50 nodes network at an application rate of 1 packet per 50 s.

5.1.2 PHY/MAC Parameter Tuning at Runtime

In order to select the optimum values of different parameters at runtime according to the specified objective function, an optimizer requires performance feedback from the network. The relationship of the tunable parameters with the performance metrics is either known on prior basis or is established using the performance feedback. In the following, we will describe our case study for a PHY/MAC cross-layer design on WARP boards in order to maximize the packet delivery ratio while consuming the least amount of power. This is achieved by adapting the type of the modulation scheme at runtime when subjected to different interference levels. The optimizer monitors the wireless interference levels and selects the appropriate modulation scheme before transmitting a packet.

A comprehensive prior knowledge of the execution durations and the power consumption associated with different PHY/MAC processes is required for the optimizer to evaluate the energy consumption cost before transmitting a packet. The execution time and the current consumption for different PHY and MAC processes on a WARP board, measured using Agilent Infiniium DSO8104A oscilloscope at a sampling rate of 10 MS/s, are shown in Table 5.4.

The current consumption of a WARP board with one daughter board attached at different operating stages was measured using Agilent N2783A current probe along the main power supply cable. The readings are listed in Table 5.5. The base current in supporting WARP board without any PHY/MAC radio activity was observed to be 526.5 mA.

Camp and Knightly have observed that for each modulation scheme, there is a range of SNR values at which highest throughput is achieved [186]. Here, we have measured the successful packet delivery ratio in a single transmitter-receiver pair configuration when subjected to different interference levels. The effects of different payload modulation rates have been studied in order to derive a scheme for automatic rate adaptation, which gives the lowest energy consumption and highest throughput. The interference was generated using Agilent E4438C signal generator as an unmod-

Table 5.4: PHY/MAC process execution duration on WARP.

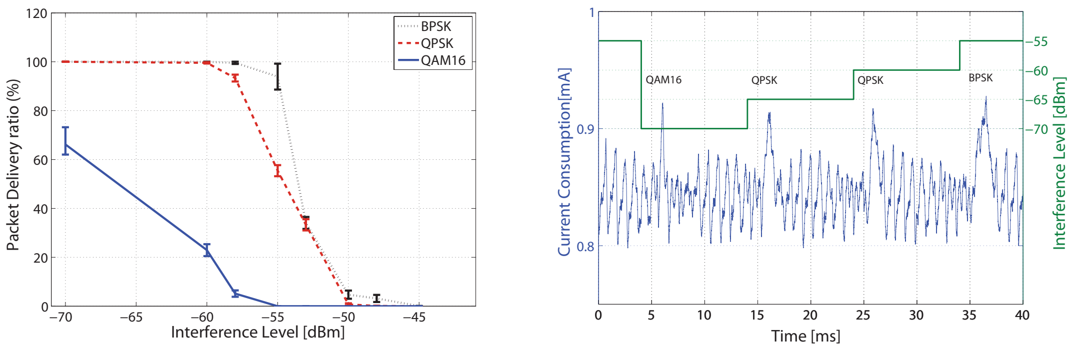
Function	*Execution Time*
`Radio_to_Tx()`	$4\,\mu$s
`Radio_to_Rx()`	$4\,\mu$s
`ReadRssi()`	$1.4\,\mu$s
`CarrierSensing (1 sample,1 $\mu$s/sample)`	$3.7\,\mu$s
`WriteToTxBuffer (1000 Bytes)`	$19.5\,\mu$s
`ReadFromRxBuffer (1000 Bytes)`	$33.5\,\mu$s
`TxPacket (1000 Bytes, BPSK)`	$1435\,\mu$s
`TxPacket (1000 Bytes, QPSK)`	$758\,\mu$s
`TxPacket (1000 Bytes, QAM16)`	$502\,\mu$s

Table 5.5: Average current consumption for different operations on a WARP board.

Radio Status	*Current consumption*
Off	838.5 mA
Rx mode of the radio	861.8 mA
Tx packet with zero payload at 100 packet/s	865.6 mA
Tx packet with 1000 byte payload (BPSK) at 100 packet/s	891.7 mA
Tx packet with 1000 byte payload (QPSK) at 100 packet/s	887.7 mA
Tx packet with 1000 byte payload (16QAM) at 100 packet/s	879.8 mA

ulated signal in the frequency band (5580 MHz) where the transmission reception path between WARP boards are established with amplitude varying from -20 dBm to 5 dBm. We have measured that the attenuation between the signal generated and the signal detected by the WARP radio front-end is 50 dBm, i.e., an external signal with 0 dBm amplitude creates an interference of -50 dBm. The antenna co-linear distances between the transmitter-receiver, the transmitter-interferer, the receiver-interferer were fixed at 10 cm, 5 cm and 5 cm, respectively.

Figure 5.5a shows that BPSK and QPSK has the same delivery ratio when the interference level stays below -60 dBm. Between -57 dBm and -53 dBm, BPSK offers approximately 20 % to 30 % performance gain. However, as the interference level goes above -52 dBm, the advantage of BPSK over QPSK diminishes to less than 5 %. QAM16, while being very energy efficient and offering high throughput in ideal environment, performs poorly in normal office environment in an uncongested 5 GHz



(a) (b)

Figure 5.5: (a) Packet delivery ratio on WARP boards using different packet payload modulation schemes at different interference level. Packets are generated at 100 packet/s. The size of packet header and payload are 24 bytes and 1000 bytes respectively. Packet header is modulated with BPSK. (b) Optimized current consumption at a WARP board under different interference levels.

ISM frequency channel on WARP boards. The experimental readings serve as prior knowledge for the optimizer. We have defined the optimization criteria to be satisfying a packet delivery ratio of at least 60 % with the lowest energy consumption. Figure 5.5b shows the behaviour of a simple MAC protocol that periodically transmits a packet every 10 ms and tunes the appropriate modulation scheme. Before transmitting a packet, the optimizer reads an RSSI value to determine the interference strength and is able to select the most efficient modulation scheme, which satisfies the minimal packet delivery ratio criteria and gives the least energy consumption.

5.2 EXPLOITING COMMON FUNCTIONALITIES

A completely independent design of PHY/MAC/routing protocols leads to sub-optimal performance [9]. Section 5.1 describes an approach of sharing information across different protocol layers through tunable parameters in order to jointly optimize a network stack and to achieve a higher network efficiency. While appropriately tuning parameters leads to performance enhancements, even higher gains can be achieved if protocols are structured in such a way that common functionalities are combined and redundant functionalities are eliminated. This approach minimizes the signaling and processing overhead and also leads to lower communication and maintenance overhead in a network. This would in turn result in energy savings, bandwidth conservation and higher throughput. Leveraging from our Decomposable MAC Framework proposal (see Section 4.4), we extend the component oriented design methodology to routing protocols so that common functionalities among MAC and routing components can be combined and redundancies can be removed. Component oriented design approach allows a broader perspective for sharing information among protocol layers and jointly optimizing protocols in a network stack. Unlike the idea of protocol heap [152], we advocate the idea of maintaining quasi-network layers and define common components and data structures, which are shared by MAC and routing layers.

5.2.1 MAC and Routing Functional Components

We have identified the common components among different MAC protocols as described in Section 4.4. These can be used as building blocks for realizing MAC solutions. Following the same philosophy, we have identified common routing components as building blocks for realizing routing schemes. These include : 1) *Beaconing component* broadcasts protocol specific knowledge set. 2) *Forwarding Engine* calculates efficient routes based on the indicated routing metric. 3) *Routing Daemon* stores the routing information for different types of routing protocols. We have also analyzed MAC and routing components in order to identify shared functionalities and data structures. Sharing information in common data structures allow a MAC protocol to piggyback information related to neighbourhood discovery, sleep-schedule of nodes, time-stamps of packet reception/transmission, etc. This would minimize explicit control overhead in a network. Common packet queues allow a MAC-routing scheme to efficiently schedule, aggregate and prioritize packet transmissions.

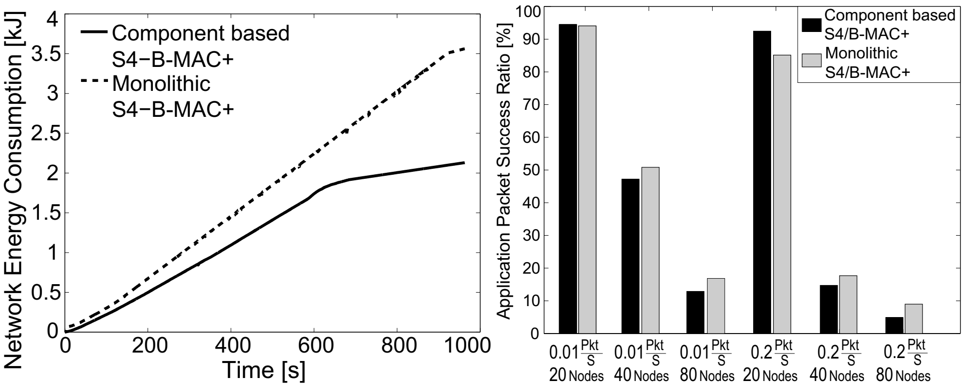
5.2.2 Design Validation of Component Based Protocol Implementation

In order to validate our approach, we have selected well-known and widely used MAC and routing protocols. The chosen MAC protocols, B-MAC+ [45] and TrawMAC [17] are based on the preamble sampling principle while the routing protocols, CTP [185] and S4 [184] use the Expected Transmission count (ETX) for link estimation. Though the behaviour of the selected MAC and routing protocols vary significantly depending upon the network and traffic conditions, yet both the sets of MAC and routing protocols share common functionalities.

We have used Indriya TelosB testbed [13] for our experimental evaluation. We have conducted comprehensive performance comparison studies of monolithic implementation combinations (TrawMAC/S4, TrawMAC/CTP, B-MAC+/S4 and B-MAC+/CTP) of the selected MAC and routing protocols to their corresponding Component Based (CB) implementations. The experiments were performed on 20, 40 and 80 nodes at a low application transmission rate of one packet every 100 s (0.01 pkt/s) and a high application transmission rate of one packet every 5 s (0.2 pkt/s). The duty cycle was kept to be 2.2 % while studying the six cases. Each experiment was repeated three times to obtain statistically significant results. We have observed that the performance characteristics of the component based implementations of the protocols closely resemble to their monolithic counterparts in terms of the energy consumption and the successful packet delivery ratio in all the cases.

5.2.3 Combining Data-structures and Piggybacking Information

Common data structures for a MAC-routing stack allow saving memory usage while combining maintenance traffic (synchronization, beacon messages, neighbourhood and route discovery, etc.) leads to significant reduction in network traffic. A lower network traffic results in conservation of bandwidth as well as energy. Figure 5.6a and Figure 5.6b show the comparison of energy consumption and application level packet delivery ratio for a protocol stack consisting of CB-S4 and CB-B-MAC+ to its monolithic counterpart. In the monolithic implementation of S4, each node periodically broadcasts reverse link qualities of its neighbours to establish bidirectional link qualities. CB-S4 implementation uses the Link Estimator component to piggyback reverse link qualities for each broadcast transmission. This reduces considerable amount of link specific messages in a network as can be observed from Figure 5.6a. More savings are achieved for a larger and more densely deployed network. Figure 5.6b shows that while allowing significant amount of energy conservation, the average application packet delivery ratio of the CB implementation remains almost the same as in the monolithic implementation for different network sizes. It can however be seen that as the network size grows, the average packet delivery ratio decreases. This is because of the absence of packet queues in S4, which causes packet drops as was also previously observed by the designers [184].



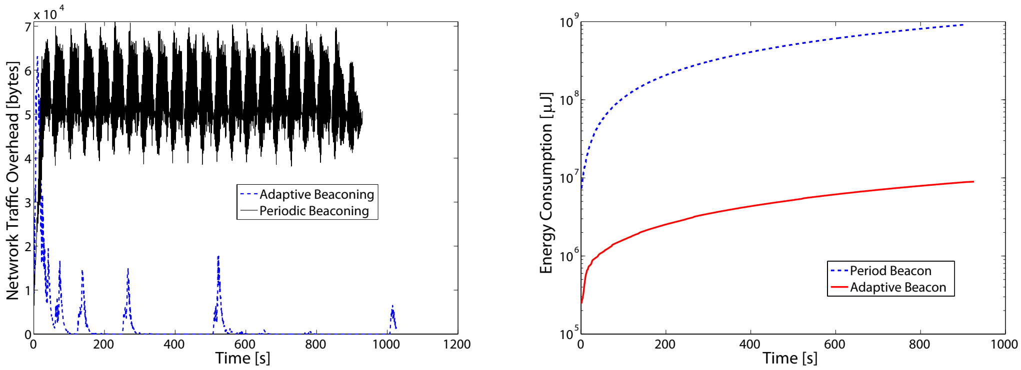
(a) (b)

Figure 5.6: (a) Energy consumption comparison between CB-S4/CB-B-MAC+ with monolithic-S4/B-MAC+ at an application packet generation rate of 1 pkt/s in a network size of 80 nodes. (b) Application level packet delivery ratio comparison between CB-S4/CB-B-MAC+ with monolithic-S4/B-MAC+ in different cases for application and network conditions. The MAC duty cycles are fixed to 2.2 % in all the cases.

5.2.4 Combining Components for Improved Behaviour

Unlike the layered approach of a protocol stack implementation, component based implementation of protocols allow combining different functionalities among protocols to achieve improved performance characteristics. However, neither all the combinations are possible nor they guarantee network efficiency. An experimental basis, simulation study or analytical modeling is required to establish the efficient combination of components.

Typically, beacon messages are broadcast at fixed periodic intervals. The beaconing interval is generally selected based on the dynamics in a network and the required allowance time for updating information across a network. A short interval usually results in higher network traffic overhead but is able to propagate information (time synchronization, new routes, new nodes, removal of outdated links, unreachable nodes, etc.) in a shorter time. Adaptive beaconing efficiently addresses this trade-off and simultaneously achieves a faster information update while keeping the network traffic overhead at a lower rate. We used the Trickle algorithm [187] for adaptive beaconing component, which is part of our CB-CTP implementation. Component oriented design allows plugging the adaptive beaconing component to S4, which originally uses periodic beaconing instead. Figure 5.7a shows the comparison of network traffic overhead between periodic beaconing scheme and adaptive beaconing scheme for S4/B-MAC+ stack in a network consisting of 80 nodes. B-MAC+ operates at a duty cycle of 2.2 % with a periodic channel sensing duration of 11 ms. The periodic beaconing interval is set to the default value of 17.5 s. For the case of adaptive beaconing, we can observe a sharp peak in the beginning while all the nodes compete for the channel at the highest beacon transmission rate of 128 ms. This rate gradually slows down after every suc-



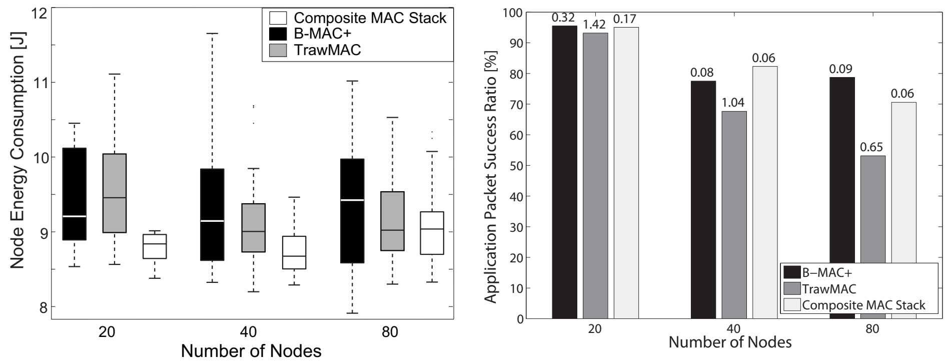
(a) (b)

Figure 5.7: Comparison of periodic beaconing with adaptive beaconing in S4 routing protocol in a network consisting of 80 nodes. S4 routing protocol is used with B-MAC+. B-MAC+ operates at a duty cycle of 2.2 % with a channel sensing duration of 11 ms. (a) Network traffic overhead. (b) Energy consumption.

cessful transmission and ultimately settles down to a value of 8.3 min. The settling duration, however, depends on the number of nodes in a network. In the presented results, the network topology is established within an interval of 200 s and therefore, the network traffic decreases drastically afterwards. This gives rise to high energy savings in S4 as shown in Figure 5.7b. Applying adaptive beaconing to S4, as a representative example, validates the idea of composing efficient protocol behaviour by simply combining different components.

5.2.5 Composite MAC/Routing Stack and Runtime Optimization

We have studied the performance characteristics of different combination of protocols under different application transmission rates, network sizes and MAC duty cycles in order to study the performance trends. These experiments enabled building hypothesis for static optimization decisions. For instance, we observed that although TrawMAC is more energy efficient, it has a lower application packet success ratio as compared to B-MAC+ regardless of the network size. Furthermore, we observed that B-MAC+ on average requires less retransmissions for unicast packets as compared to TrawMAC. Based on these experimental observations, we deduced the following: In order to achieve high packet delivery ratio with least energy consumption, broadcast transmissions should be attempted with TrawMAC and if the first unicast packet transmission fails with TrawMAC, it should later be sent through B-MAC+. Component based implementation of protocols allow composing a composite MAC stack which may benefit from such a runtime optimization decision. In order to allow TrawMAC and B-MAC+ behaviour to simultaneously coexist, we have defined common data structures for storing neighbourhood sleep schedule information and a field to identify the type of the MAC scheme. Figure 5.8a and Figure 5.8b show the energy consumption and

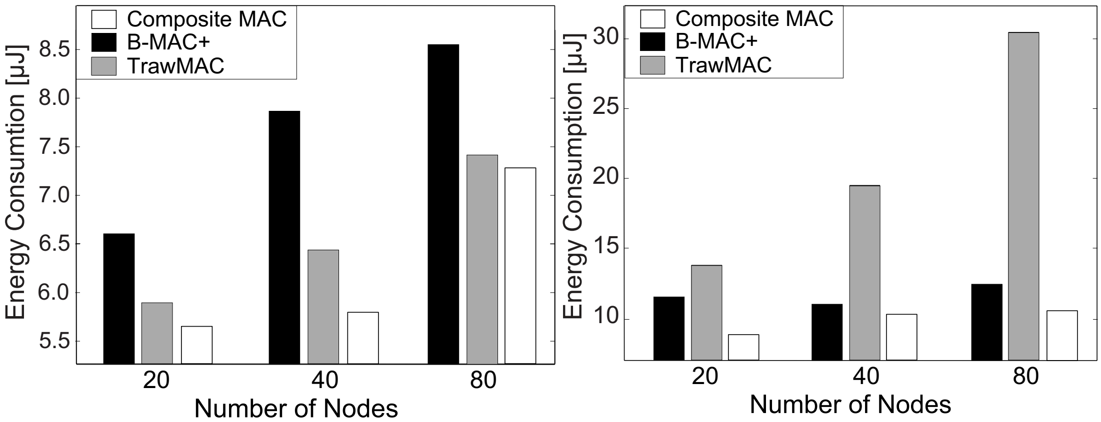


(a) (b)

Figure 5.8: Comparison of B-MAC+, TrawMAC and Composite MAC stacks in different network sizes for an application rate of 0.2 pkt/s. The MAC protocols operate at a duty cycle of 2.2 % with a channel sensing duration of 11 ms. CTP is used as the routing protocol. (a) Energy consumption per node. (b) Application level average packet delivery ratio. The average retries are listed on top of the bar graphs.

the application packet delivery ratio of B-MAC+, TrawMAC and a Composite MAC Stack which enables the above optimization decision at runtime. The duty cycle and the channel sensing duration for all the MAC protocols are set to be 2.2 % and 11 ms, respectively. CTP is used as the underlying routing protocol. It can be observed from the two figures that the composite stack gives the most energy efficient operation while achieving the maximum successful packet delivery ratio.

We have also studied variable application traffic data rates. We have carried out an experiment where the application traffic rate is changed during the experiment. After the topology is established, the network starts with a low application traffic rate of 0.01 pkt/s for a duration of 300 s. Later on, the application traffic rate is changed to 0.2 pkt/s for 200 s, which falls back to 0.01 pkt/s for a duration of 300 s. Figure 5.9a shows the energy consumption per node for a successfully delivered packet during the low traffic conditions. We can observe that Composite-MAC/CTP adapts its behaviour and is thereby able to achieve 5 %, 10 % and 3 % performance gains over monolithic MAC stacks for 20, 40 and 80 nodes network, respectively. For low application traffic rates, the PRR stays similar for all the network sizes so we see satisfying performance of TrawMAC. Figure 5.9b shows the energy consumption per node for a successfully delivered packet for low as well as high traffic rate conditions. The Composite MAC stack when estimates high traffic conditions behaves like B-MAC+, while it adapts to TrawMAC-like behaviour during the low traffic conditions. For high traffic rates and larger network sizes, the PRR of TrawMAC drops by 50 % due to deafness problem and high amount of retransmissions in the network. Composite MAC adapts B-MAC+ like behaviour when the packet retransmissions increase. Composite MAC stack achieves performance gains of 25 %, 16 % and 6 % for networks consisting of 20, 40 and 80 nodes respectively over the monolithic protocol stacks.



(a) Low-traffic rate.

(b) Low+high traffic rate.

Figure 5.9: Average energy consumption per node per successful packet for a network with variable traffic rate.

Extending the idea of composite MAC stack, we have designed a composite routing stack which simultaneously support the behaviour of S4 and CTP. CTP relies on a single root node[2] while S4 allows many root nodes and is suitable for multiple application flows or point-to-point communication. Experiments were conducted to deduce runtime optimization decisions for the composite routing stack in a similar way. It was observed that CTP achieves more energy efficiency and higher packet delivery ratio for a single sink static network regardless of its size. Due to network dynamics and mobility if a node lose its parent, CTP would trigger adaptive beaconing pulls thereby causing high surge of energy consumption. Also when a node has lost its parent and has reset its adaptive transmission interval to a lower value, packets would remain undelivered and decrease the resulting throughput. On the other hand, we could use each beacon node as a sink in S4 to allow point to point communication. However, absence of per flow based packet queues leads to packet drops which eventually lowers the success packet delivery ratio for high traffic conditions (higher application rate and/or larger network size). This reason has also been reported in the primary S4 implementation [184]. Due to memory constraints, per flow based packet queues cannot be supported in the composite MAC/routing stack on TelosB with our CB implementation.

We have devised an experiment with multiple application flows for evaluating the performance of composite MAC/routing stack. We applied all the above mentioned optimization decisions including 1) the composite MAC decisions, 2) S4/CTP composite stack initialization with adaptive beaconing and 3) link quality estimation while exploiting the reverse links. Furthermore, we have applied a decision that if the destination is a CTP root node or its next hop neighbour, it should be sent using CTP otherwise it is sent through S4. The destination is randomly selected with equal probabilit-

[2]Only the CTP implementation on Linux/Click [188] supports point-to-point communication.

ies to be a CTP root or any other node. The empirical results show that our composite stack allows such a multiple application flow combination and gives more than 75 % successful packet delivery ratio for both the application rates of 1pkt/100s and 1pkt/5s in a network consisting of 20 nodes. One should note that the composite MAC/routing stack requires almost twice the time for routing topology setup while the CTP and S4 behaviours coexist simultaneously. Furthermore, longer sized packet transmissions are required for routing control information in the composite stack. Since the TinyOS 2.x CTP implementation does not support multiple links, experimental comparison was not conducted. Monolithic S4 implementation gives 44.3 % higher energy consumption and lower packet delivery ratio (65 % for 1pkt/100s and 59 % for 1pkt/5s) than the composite MAC/routing stack.

5.3 USING ALTERNATE COMMUNICATION PARADIGMS

We have investigated the use of alternate communication paradigms for achieving cross-layer optimization in contrast to the classical communication interaction style among nodes [189]. A number of different communication models have been proposed during the past two decades, however only a few communication interaction paradigms better suit to low-power applications. In the message transfer model, a sender invokes the network for conveying a message to a particular destination. The sender is not notified about the reception of message at destination. However, the receiver is notified about the arrival of a message, which has to be accepted. The mailbox model [190] extends the message transfer model by decoupling the address of the destination from its location. It does not require the destination to be available at all the times. In client-server model, a client requests a server for information. The server responds back with the requested information and the client issues a confirmation. The client remains in blocking state during sending of the request and its confirmation. Remote Procedure Calls (RPCs) are similar in essence to the client-server model. The client-multiserver [191] approach allows multiple servers to collaborate in order to respond to the request of a client. This model is often used in peer-to-peer (P2P) networks. The client-server, message passing and mailbox models exercise point-to-point connections in which network protocols just have to carry the information (message) from a source to a destination with strict QoS requirements. If a single piece of information is needed by more than one node at the same time (which is common in data oriented applications), the network will have no choice but to send the same information as many times as the number of requests.

Publish/Subscribe (P/S) paradigm provides time, space and synchronization decoupling to nodes. Time decoupling is achieved by avoiding the need for the active participants in a message exchange to be present at the same time. Space decoupling is accomplished as the participants do not need to know about their location while synchronization decoupling is obtained via non-blocking interactions. P/S is suitable for asynchronous communication and is able to meet demands of dynamic distributed applications by providing highly versatile information delivery, e.g., personalized information dissemination, distributed system monitoring, alerting and notification. The P/S model exhibits many interesting properties that suit to the application areas of sensor networks and other low-power embedded networks [192]. It is often argued that P/S model has high implementation complexity on the network side, however, we hypothesize that the model offers more freedom for network optimization and thereby leads to lower power consumption. A network governing P/S model is notified that different nodes are subscribed to the same information and hence it may optimize the distribution of information from a particular publisher to different subscribers. There is no need to propagate information in case there is no subscriber. In the absence of a publisher, no traffic is generated and hence significant amount of energy and bandwidth is saved. The P/S model also allows reducing the overall network traffic by concatenating broadcast and address-less messages and exploits network topologies. For instance, in tree like topologies, upwards traffic can be collected by the root node and distributed to all the nodes via a single efficient broadcast. P/S model allows adapting operations by dynamically trade between a push and a pull model. The P/S model provides a high degree of decoupling on the application side and this freedom can be exploited in a network for optimizing its operations.

In particular, we have explored the cross-layer design interaction of TrawMAC [17] and Content and Context Based Routing (CCBR) in P/S paradigm. TrawMAC is a preamble sampling based protocol designed with the primary goal of optimizing energy consumption through exploitation of shared information across protocol layers regarding traffic types, traffic loads and timing information. CCBR [62] is a context and content based routing protocol for WSNs, which is designed to operate efficiently in the presence of mobility in networks. CCBR offers three main primitives to an application: the ability of setting the context (properties) of a sensor node, the ability of sending messages and the ability of expressing an interest in receiver specific messages from specific sensor nodes. Furthermore, additional data can be specified by a sink and transported to relevant sensor nodes in order to steer the sensing and communication behaviour at a node. The "additional data" is blindly transported while the format of filters, properties and messages only affect the matching process. In CCBR, each sink has an associated unique sink number and in order to forward data from sources to sinks, each node maintains a distance table and a content table. Different cross-layer design techniques applied to the TrawMAC/CCBR stack are described below:

5.3.1 Support for Broadcast Traffic

CCBR generates only broadcast traffic at the routing layer. TrawMAC collects the neighbourhood sleep-schedule information and shortens the preamble length for broadcast packets. In a static network neighbourhood, nodes can have complete information about their neighbours after a few message exchanges. Based on the gathered wake-up schedules of all one-hop neighbours, the nodes can shorten the length of a preamble for broadcast packets by substituting broadcast transmissions with multiple unicasts, a shortened broadcast, or a combination of these, depending on the sleep schedules of the neighbouring nodes (cf. Section 2.4.1).

5.3.2 Timing Offset

For small sized packets, TrawMAC embeds data into preamble framelets and transmits the framelets repetitively back-to-back for a maximum duration equal to the sleep interval of the duty cycling nodes. Since nodes wake up asynchronously in a network, the time-stamps for reception of framelets (containing a particular data) can be different at different nodes. CCBR calculates the forwarding of a packet based on the time-stamp of a packet reception and hence requires that all the nodes receiving the same packet have the same notion of packet generation time. TrawMAC, therefore, includes the timing offset between the start of the preamble and the transmission of a particular preamble frame into the header of each frame. Using this offset, all the neighbouring nodes are able to estimate the actual packet generation time. This however is true only for smaller data sizes in which DFP scheme is used (cf. Section 2.4.1) For bigger packet sizes, TrawMAC uses MFP scheme, i.e., it first transmits the preamble frames (without piggybacked data) followed by the data frames. In this case, all the nodes are implicitly synchronized by the preamble frames and receive data at the same instant.

5.3.3 Message Queue and Data Aggregation

The packets to be transmitted can be scheduled at any point of time. Often the packets cannot be processed immediately due to the unavailability of a free communication channel or the transceiver being in the busy or inactive state. Consequently, more than one data packets are often scheduled before a transmitter is able to initiate transmission. Especially in CCBR, where the forwarded packets can be delayed, queued and retransmitted, it is ideal for a MAC to be able to accept more packets than what can be transmitted at a particular time instant. A MAC queue is therefore implemented in TrawMAC, which allows multiple data packets to be aggregated and transmitted with a lower control overhead. In CCBR where all the packets are broadcast, the queued packets at the MAC are transmitted using only one medium reservation. This not only conserves energy but also lowers the latency and improves the channel bandwidth utilization.

5.3.4 Packet Cancellation

In dense multihop networks, there is a high probability that nodes receive duplicated requests for packet forwarding. Forwarding duplicated packets increases the network traffic tremendously especially with increasing number of hops from source to destination(s). In order to minimize this duplication, TrawMAC uses a packet cancellation scheme taking advantage of its MAC queue. When a packet is given to TrawMAC, it is able to check if the same packet is already in the queue for transmission. Identical packets are removed from the MAC queue since the message has already been forwarded by other neighbouring nodes.

5.3.5 Lifetime of Packets and Prioritization

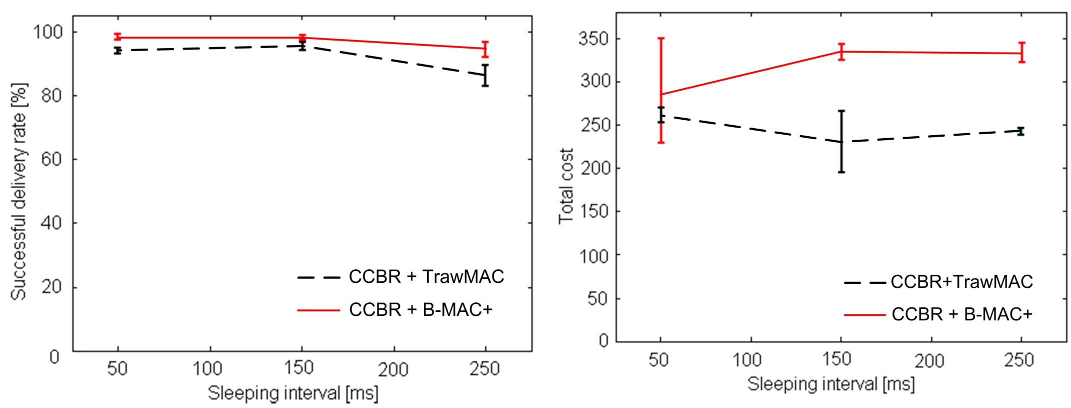
The latency associated with packet delivery from a source to a sink is hard to predict for a WSN protocol stack. CCBR imposes lifetime limit to each packet when it is passed to the MAC layer. The lifetime is decided at the routing layer depending on how delay-tolerant a packet is. The packet is kept in the MAC queue for transmission until the expiration of its lifetime and is dropped from the queue afterwards. The application is consequently signaled with a failed transmission. As an extension of this functionality, queue management techniques based on priority, lifetime, etc., can be implemented to further enhance the performance of a protocol stack. Packets with low delay tolerance level will be pushed to the front of the queue and the back-off window at the node can be reduced to increase the chance of seizing a medium.

5.3.6 Adjustment of CCBR Timeouts based on the TrawMAC Duty Cycle

Like any preamble sampling protocol, the latency of TrawMAC directly depends upon the sampling period. A long sampling period (small duty cycle) has high latency and vice versa. CCBR has different timeouts in packet forwarding which are expressed in terms of the inherent delay from MAC duty cycle.

5.3.7 Experimental Performance Evalutation

The cross-layered TrawMAC and CCBR stack was implemented in TinyOS 2.x. It has been evaluated for performance characteristics on a multihop 9-node TelosB test-bed and was compared to a reference layered protocol stack consisting of CCBR and B-MAC+ [45]. Figure 5.10(a) shows the successful packet delivery ratio, which is pretty high and is very similar for the two stacks. Figure 5.10(b) quantifies the overall traffic reductions obtained. It can be observed that CCBR generates much less overall



(a) Average successful packet delivery ratio. (b) Total cost.

Figure 5.10: Packet delivery ratio and total cost comparison of CCBR/TrawMAC stack to CCBR/B-MAC+

traffic when running on top of TrawMAC with all our cross layer optimization in place than when running on top of the non-optimized stack with B-MAC+. Looking at the two figures together, we can observe that for instance at a sleep interval of 150 ms, CCBR+TrawMAC lose less than 3 % of delivery w.r.t. CCBR+B-MAC+, however the optimized stack generates 32 % less traffic to obtain this result. Since transmitting packets has a high cost in terms of energy consumption, our optimized stack definitely lead to a longer network lifetime. Also, reduction of network traffic results in a much better scalability and bandwidth utilization.

5.4 SUMMARY

Although MAC protocols play a pivotal role in the overall network performance and therefore designing efficient MAC solutions is highly important, analyzing MAC protocols in isolation is incomplete from a practical point of view. Classical layered protocol stack design restricts information sharing and cooperation among different protocols, which makes it inefficient. Cross-layer network design and cooperatively optimizing PHY/MAC/routing protocol layers has therefore received considerable attention in the past years. However, only a few design proposals have been deployed on real nodes, while most of the investigations are limited to analytical and simulation based studies. Efficient use of network capabilities and performance optimization becomes even more important for resource constrained devices and networks. In this chapter, we have investigated three cross-layer design approaches for optimizing the performance of embedded wireless network stack. We have validated these design approaches through comprehensive performance evaluation on testbeds consisting of TelosB nodes and WARP SDR platform.

With a growing number of networking protocols and application areas of embedded wireless networks, the appropriate selection of a protocol stack and protocol parameter settings is becoming highly challenging. To the best of our knowledge, there exists no tool with comprehensive empirical basis to provide optimal selection of a protocol stack for embedded wireless networks. We have designed CONFab in order to facilitate the selection of an appropriate protocol stack at the pre-deployment phase for a particular application scenario. Cross validation results from habitat monitoring application [12] show that CONFab is able to recommend the best selection of the protocol stack and parameters. CONFab has the ability to improve its knowledge-base based on the performance feedback extracted from an already deployed network.

Most of the MAC and routing protocols are designed independently from each other and exhibit common functionalities. There is a high potential for improving network performance by eliminating redundant functionalities and combining commonalities among PHY/MAC/routing protocols. Following a component based design approach, we have identified common MAC and routing components which serve as the building blocks for composing protocol stacks. Extensive performance measurements of 700+ hour has been conducted on Indriya TelosB testbed to comprehensively study the characteristics of the well-known MAC and routing protocols and to validate their performance against their component based implementation. Cross-layer design approach of combining commonalties and optimizing the network behaviour using composite MAC/routing protocol stack has shown an increased lifetime of more than 40 % while providing the same packet delivery ratio over classical layered protocol stack implementation.

Finally, we have investigated the cross-layer MAC-routing design in P/S communication paradigm. P/S allows decoupling of the source and destination. This leads to overall lower network traffic, which in turn conserves energy consumption and communication bandwidth.

6

CONCLUSIONS

The ever increasing demand for of emerging applications, new challenging deployment conditions, and the need for resource efficient operation are widening the gap between the performance expectations and the actual capabilities of the existing embedded wireless networking solutions. MAC protocols play a critically important role in determining the overall efficiency of embedded wireless communication systems. Though many new MAC protocols have emerged in recent years, most of them lack insights into their performance behaviour in realistic deployment conditions. Consequently, when these protocols are deployed in real-world applications, they often fail to provide the promised performance characteristics as have been indicated through analytical and simulation studies. In order to exploit embedded wireless networks in different daily life applications and explore further areas and services, it is necessary to design new solutions with high relevance to real world deployments and application demands.

This dissertation has introduced many new concepts and protocol proposals in order to bridge the gap between theoretical understanding MAC designs and their behaviour in practice. A special focus has therefore been given to prototyping of the proposed MAC solutions and their evaluation in realistic network, traffic and spectral conditions. Extensive performance measurements have been conducted to study the trade-offs among different performance metrics. The results achieved during this work provides meaningful understanding of the trade-offs among different performance metrics. A summary of the main results acquired during this dissertation are provided in Section 6.1. Section 6.2 discusses the trade-offs among different solutions. Finally, Section 6.3 outlines some future work directions.

6.1 SUMMARY OF THE KEY RESULTS

Energy efficiency is considered to be of high importance in the design of MAC protocols for low-power applications since it is directly related to the lifetime of a network. Instead of just considering the power consumption aspect, we have looked into the trade-offs among energy consumption, offered latency and data reliability. Preamble sampling is a popular approach used in duty cycling MAC protocols as it does not impose the need for explicit synchronization among nodes and inflicts control overhead only when data communication is required. Different schemes are designed to shorten the length of the preamble transmission and reception for energy conservation. Different techniques for shortening the preamble length suit better to different applications and traffic conditions. We have analyzed these techniques and designed *TrawMAC*, which combines the strengths of different preamble optimization techniques and offsets their weaknesses depending upon the traffic patterns and volume. Furthermore, TrawMAC uses a novel scheme of substituting broadcast preambles with a combination of shortened broadcast preambles and/or unicast preambles. Experimental results on a testbed consisting of TelosB nodes have indicated considerable energy savings through TrawMAC compared to other state-of-the-art protocols. The ability of TrawMAC to adapt its behaviour according to the patterns and volume of the subjected network traffic makes it suitable to a wide range of applications. TrawMAC has been exploited for the application areas investigated in the EU funded WASP project [64]. We have analyzed the dominant sources of energy consumption in embedded wireless communication and prototyped a dual radio platform operating in different frequency bands. The two radios are dedicated to specific operations so as to minimize the overall energy expenditure at a node. We have proposed a novel protocol, *MR-MAC*, which uses a simpler sniffer radio operating in a lower frequency band for preamble sampling operation and a faster burst radio operating in a higher frequency band for data exchange. We have prototyped a hardware platform using commercially available components for carrying out MR-MAC implementation. We have evaluated the performance of MR-MAC against B-MAC in identical experimental conditions. The results indicate that MR-MAC achieves significantly high energy savings and imparts lower latency in data communication as compared to B-MAC in realistic traffic conditions. These results hold against B-MAC implementation on single radio platforms using the two individual radios as are on our dual radio prototype platform. Duty cycling MAC protocols govern a trade-off between the energy consumption and the offered latency in data communication. Consequently, extremely low-power operation can only be achieved at a high cost of the offered latency. We have proposed an innovative idea of combining radio triggered wake-ups with duty cycling MAC protocols. Our solution, *RTWAC*, achieves extremely high energy efficiency through a wake-up circuit (cf. Figure 2.21 for schematics) attached to an embedded wireless node. The circuit allows the on-board radio of a node to be completely switched off and the microcontroller to be in a low power operational mode. Instead of transmitting a tonal signal, RTWAC uses OOK modulated data in the wake-up signal. The data includes the MAC address of a node and hence triggers wake up only at the destination. Having addressing capability helps in preventing considerable amount of energy wastage in undesired node wake-ups. Upon wake-up, a node uses a low-power duty cycling MAC protocol running on the on-board radio for data communication to achieve a higher reliability and throughput for data communication compared to the wake-up radio. RTWAC simultaneously allows achieving a fast response and extremely high energy efficiency.

A growing number of applications relying on embedded wireless networks is increasing the congestion on the available wireless spectrum. Medium access without spectrum efficiency and coexistence features lead to mutual interference, which consequently results in performance degradation. Spectrum agility is therefore necessary to avoid interference and achieve reliable data communication through dynamic channel access. Although many experimental studies have shown that low-power networks suffer high packet losses from other networks operating in the same frequency band [7, 11], not much has so far been done towards prototyping of interference mitigation schemes. We have designed a low-power MAC protocol, *SA-MAC*, which allows spectrally efficient operation and enables reliable data communication. We have introduced the concept of multi-channel polling operation combined with distributed channel selection in SA-MAC protocol. SA-MAC uses a heuristics based fully decentralized dynamic channel selection strategy. Prototype implementation of SA-MAC on a testbed of TelosB nodes has shown high performance characteristics in terms of packet delivery ratio and throughput even in the presence of challenging wireless interference conditions. We have also developed a decentralized MAC protocol, *CogMAC*, for CR networks. CogMAC is designed for infrastructureless networks and uses a similar distributed channel selection strategy as SA-MAC. CogMAC is targeted for relatively less resource constrained ad hoc networks requiring dynamic channel access capabilities. It is suitable for both licensed and ISM frequency bands, where the protocol adapts its channel selection to the stochastic spectrum occupancy behaviour of the PUs and mitigates the effects of the random interference. We have implemented CogMAC on a testbed consisting of WARP SDR platforms. Our implementation provides significant insights into the practical aspects of CR MAC protocols, especially while majority of the state-of-the-art designs remain at a theoretical and simulation stage. The performance evaluation indicates that CogMAC allows efficient utilization of spectrum opportunities.

With emerging new application areas, customized MAC solutions are required. Designing and implementing MAC protocols for embedded networks is considered to be a tedious task and typically require substantial domain knowledge of the underlying hardware platform and operating system. The need for a deep programming knowledge, lack of flexibility and the supporting tools have been serious roadblocks for rapid research and development in the field, especially within the academic community. MAC protocols are generally implemented in a monolithic fashion with tight coupling to the hardware platform which limits the room for experimentation, reusability and modification of code. We have introduced *Decomposable MAC Framework*, which allows composition of MAC protocols from a library of basic functional components. This approach enables rapid prototyping of MAC protocols and allows high degree of code reusability. We have identified the basic MAC components and implemented them in a software-hardware co-design fashion to allow flexibility as well as hardware acceleration for time critical operations. Furthermore, we have introduced the concept of MAC reconfiguration through composition of an appropriate MAC solution at runtime. The MAC components are connected, using a tool called as Wiring Engine, to form a particular MAC composition. Wiring Engine coordinates the data and the control flow among different MAC components and controls their execution sequence. Function pointer based fast execution of the component list enables Wiring Engine to meet timeliness requirements of spectrum agile and cognitive MAC protocols as we have shown through prototyping of CogMAC. We have developed a MAC meta-language, which allows expressing a MAC protocol in a few lines of code. A meta-compiler running on the host node allows autonomous compilation of the MAC meta-code, which facilitates self-configuration. We have demonstrated this concept when a MAC adapts its behaviour to Aloha, CSMA/CA or spectrum agile depending upon the wireless interference conditions [167]. Extending Decomposable MAC Framework, we have designed an interactive user friendly graphical tool, *MAC-PD*, which allows to drag-and-drop-and-connect components in the form of flow-charts for designing a particular MAC protocol. This way prior knowledge of the platform and the language syntax is no longer required from a designer. The framework generates downloadable MAC code for the selected COTS sensor node platforms and the WARP board. Experimental performance comparison for various commonly used MAC protocols indicates that the memory footprint and the achieved throughput of the MAC implementations closely resemble their monolithic counterparts. We strongly believe that Decomposable MAC Framework will provide a wider experimentation room for prototyping new solutions. Furthermore, it will give a new dimension to runtime reconfiguration of MAC protocols.

Classical layered protocol stack design restricts information sharing and cooperation among different protocols, which makes it inefficient. Efficient use of network capabilities and performance optimization becomes even more important for resource constrained embedded wireless networks. MAC protocols play a pivotal role in the overall network performance and therefore designing efficient MAC solutions is highly important, looking at MAC protocols in isolation is incomplete from a practical point of view. A growing number of protocols and different application requirements make the optimal selection of the composition of a protocol stack and correspondingly protocol parameter settings very challenging. We have designed *CONFab* for facilitating the selection of a suitable protocol stack for a particular application scenario at the pre-deployment phase. To the best of our knowledge, there exists no other tool, which uses comprehensive empirical basis and performance feedback to provide optimal selection of a protocol stack for embedded wireless networks. Cross validation results from habitat monitoring application [12] show that CONFab is able to provide a suitable recommendation for the selection of a protocol stack and parameters. CONFab has the ability to improve its knowledge-base based on the performance feedback extracted from an already deployed network and user experience. There is a high potential for improving network performance by eliminating redundant functionalities and combining commonalities among protocols in a stack. Following a component based design approach, we have identified common MAC and routing components which serve as the building blocks for composing protocol stacks. Extensive performance evaluation for over 700 hours have been conducted on Indriya TelosB testbed [13] to comprehensively study the characteristics of the well-known MAC and routing protocols and to validate their performance against their component based implementation. Finding an appropriate combination of components as protocol stack and switching their combination at runtime allows cross-layer optimization. This approach of using composite MAC/routing stack has shown an increased lifetime of over 40 % while providing similar packet delivery ratio as compared to classical layered protocol stack implementation. We have also investigated the cross-layer MAC-routing design in the P/S communication paradigm. The P/S interaction paradigm allows decoupling of the source and the destination nodes. This leads to an overall lower network traffic, which in turn conserves energy and communication bandwidth as confirmed by our results on a testbed consisting of TelosB nodes.

6.2 TRADE-OFFS AMONG DIFFERENT SOLUTIONS

In this section, we briefly discuss the scope and trade-offs among different solutions that have been proposed in this dissertation.

- TrawMAC is a highly scalable protocol, which augments different preamble optimization techniques. It uses an algorithm for the substitution of broadcast preambles and maintains queues to allow data aggregation and packet prioritization. While it suits to a much wider range of application traffic requirements, its algorithm is relatively complex and requires a larger memory footprint as compared to the protocols with only a single preamble optimization strategy. Despite a higher complexity, TrawMAC has easily been realized on commercially available TelosB and BSN [170] nodes.

- MR-MAC is based on a dual radio platform. Our prototype hardware platform is composed of commercially available components: TelosB node interfaced to a CC1000PP radio module. Such a platform is commercially unavailable and requires hardware interfacing of COTS components as shown in Figure A.1.

- WiSpot platform consists of two IEEE 802.15.4 radios. In our prototype implementation, we have interfaced two TelosB nodes (cf. Figure A.3) in a master-slave configuration. The algorithm can also be realized on a commercially available quad-radio platform [193].

- RTWAC is designed to achieve extremely long operational lifetimes with a fast response time whenever required. The prototype platform has an operating range of approximately 10 m with the transmit power regulations in Europe. Therefore, RTWAC is suitable to applications requiring relatively short communication range for instance in industrial automation, logistic toolchain, BANs, etc. As an example, other researchers have applied our solution to a BAN application [194].

- While the concept of full-duplex communication has been demonstrated [97,98] for a certain range of bandwidth and transmit power, CogMAC assumes that the sensing and transmission operations cannot be simultaneously conducted. Half-duplex radio interface based design, on the contrary, makes CogMAC applicable to a much wider class of existing networks requiring dynamic channel access. Inability to sense and transmit at the same time potentially decrease the achieved throughput as compared to a full-duplex radio interface. Furthermore, the time required to vacate a particular channel upon detecting a PU activity is limited by a frame transmission duration, which is typically very short.

- The meta-compiler approach for MAC composition at runtime is prototyped on WARP SDR platform. This approach is challenging to realize on currently available COTS sensor node platforms owing to the memory and computational constraints. Furthermore, the realization of MAC components on the WARP SDR platform is carried out using a software-hardware co-design approach. Absence of an FPGA on sensor node platforms restricts such a custom implementation of MAC components.

6.3 FUTURE WORK

In this dissertation, we have investigated fundamental practical issues of MAC protocols in embedded wireless networks through theoretical and empirical studies. We have carried out prototype implementation of our main proposals and evaluated them with different performance metrics under realistic traffic, network and spectrum use conditions. Though this dissertation contributes new MAC designs and understanding their behaviour in practice, the topics investigated warrant further research and investigating new design methodologies. In this section, we describe a few possible extensions for our work.

RTWAC with its short ranged communication, ultra-low power consumption and fast response time has already attracted some researchers to use it in Body Area Networks (BANs) [194]. We think that the design philosophy of RTWAC suits well to industrial automation and asset tracking applications. Instead of discrete components, an integrated chip solution would certainly lead to further energy savings and size optimization.

The design principle of CogMAC has a high potential for ad hoc networking applications requiring dynamic channel selection. The dynamic channel selection in CogMAC is carried out using a heuristics algorithm. Channel weight assignments and selection can certainly be improved using channel occupancy prediction algorithms. As an extension, we have applied our `ON-OFF` activity pattern prediction scheme [195] to CogMAC, which is part of an integrated framework using cognitive radio principles in home networking scenarios [196]. In the future, more sophisticated channel weight assignment schemes exploiting the temporal and spatial occupancy models of PUs can be applied to CogMAC.

Decomposable MAC Framework provides a platform for designing and realizing reconfigurable MAC protocols. There are a few research directions that we have already been following. Realization of MAC components on multi-core computing fabrics is underway in the EU funded 2PARMA [197] project. Reconfigurable PHY-MAC integration based on the component oriented design methodology is part of an on-going research activity in UMIC Research Center's [198] Nucleus flagship project funded by German research foundation (Deutsche Forschungsgemeinschaft). The framework can be extended with a meta-compiler support that predicts energy consumption and execution duration of the perceived solutions at a pre-deployment phase based on the profiles of individual MAC components.

A

Schematics

A.1 Multi Radio Platform

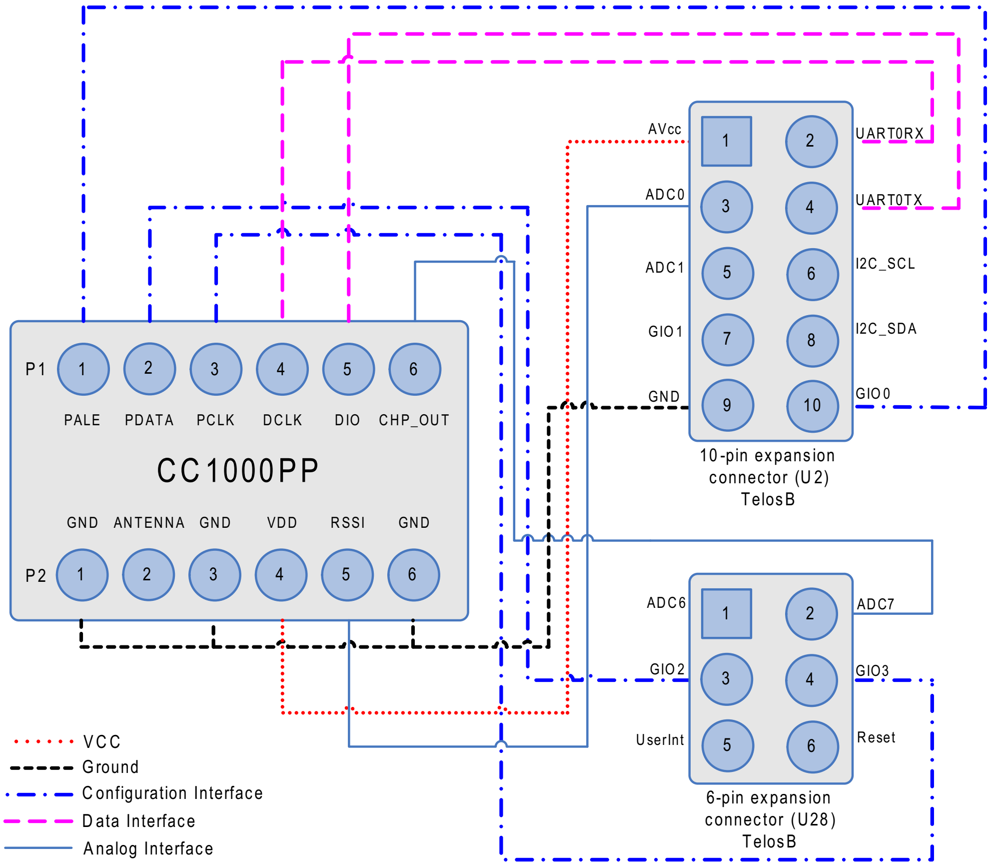


Figure A.1: Pin connections of the CC1000PP radio module to a TelosB node. The data, configuration and power interfaces of the module are connected to the 10-pin expansion connector (U2) and the 6-pin expansion connector (U28) on TelosB node.

A.2 RTWAC TRANSMITTER

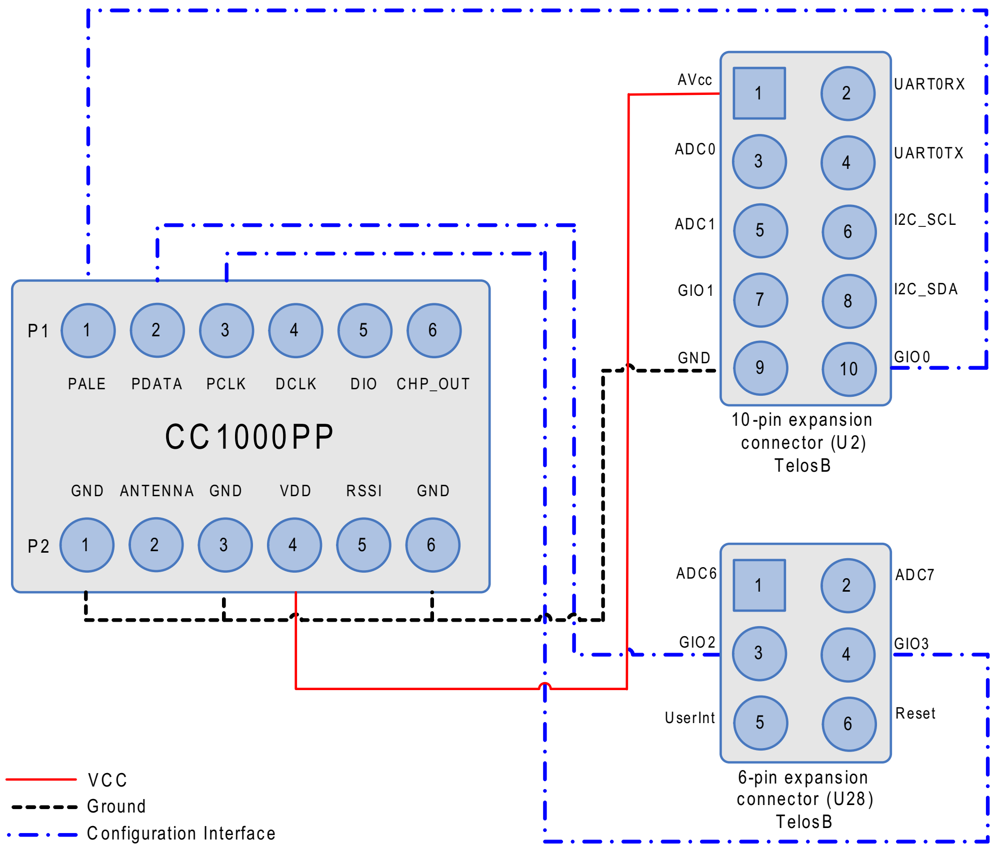


Figure A.2: Pin connections of the CC1000PP radio module to a TelosB node. The configuration and power interfaces of the module are connected to the 10-pin expansion connector (U2) and the 6-pin expansion connector (U28) on TelosB node.

A.3 WISPOT PLATFORM

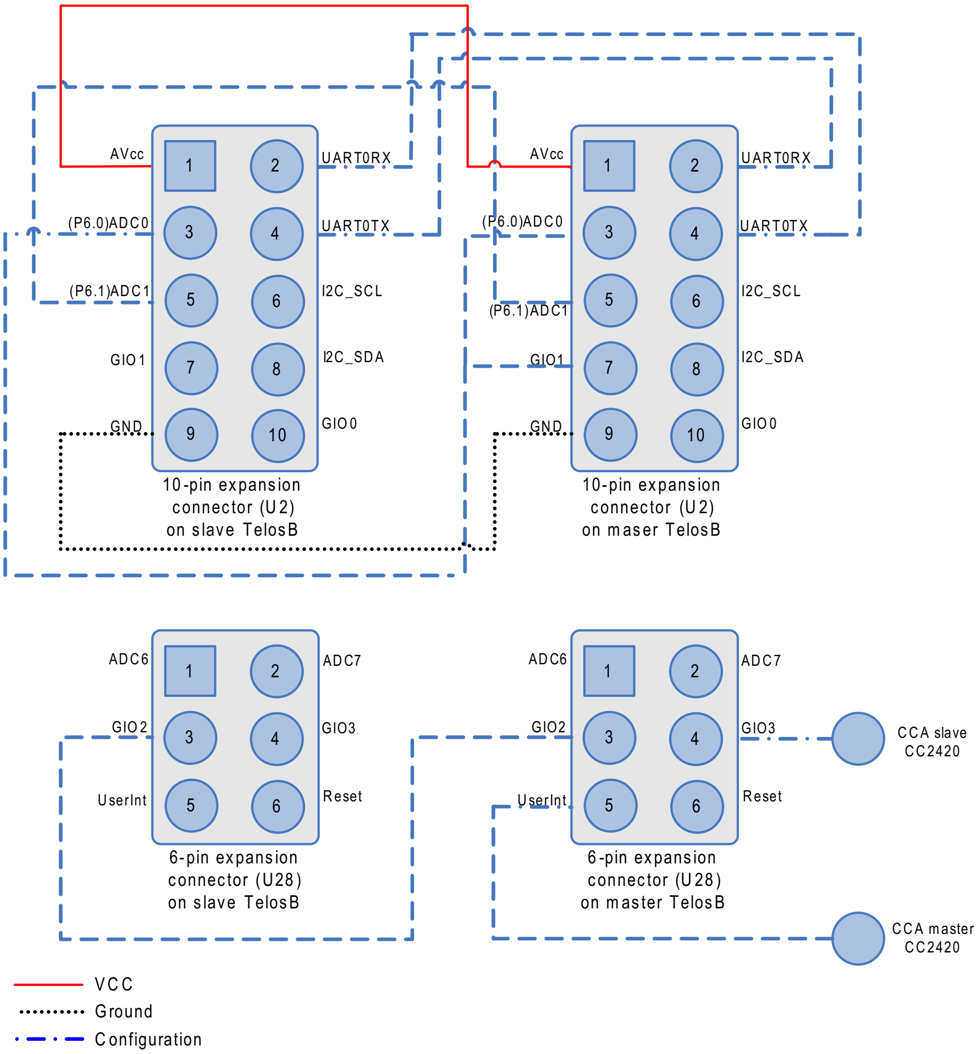


Figure A.3: Pin connections on the WiSpot prototype platform consisting of two TelosB nodes in a Master-Slave configuration. The 10-pin expansion connector (U2) and the 6-pin expansion connector (U28) on TelosB nodes are connected. Additionally, the CCA pins from the master and the slave CC2420 radio chips are also connected to the Master microcontroller via U28.

B

NOTATIONS

In the following we list the notations in an alphabetical order and their meanings.

Table B.1: Notations in capital Latin letters used throughout this dissertation.

Symbol	Meaning
E	Overall energy consumption at a node
E_{b}	Energy per bit
E_{bcast}	Energy consumption in a broadcast transmission
E_{cs}	Energy consumption in performing carrier sensing operation
E_{frame}	Energy consumption in transmitting a frame
E_{poll}	Energy consumption in channel polling operation
$E_{\mathrm{poll\_once}}$	Energy consumption in performing one channel polling operation
$E_{\mathrm{radio\_setup}}$	Energy consumption in initializing a radio
E_{rx}	Energy consumption in receive mode
E_{rxsetup}	Energy consumption in setting up the radio in receive mode
E_{rxtx}	Energy consumption in switching from receive mode to transmit mode
E_{sleep}	Energy consumption in sleep mode
E_{tx}	Energy consumption in transmit mode
E_{txrx}	Energy consumption in switching from transmit mode to receive mode
E_{ucast}	Energy consumption in a unicast transmission
G_i	Antenna gain of radio interface i
G_{r}	Receive antenna gain
$G_{\mathrm{r}i}$	Receive antenna gain at radio interface i
G_{t}	Transmit antenna gain
$G_{\mathrm{t}i}$	Transmit antenna gain at radio interface i
I_{comp}	Current consumption by a comparator
I_{div}	Current consumption by a voltage divider
I_{RTWAC}	Current consumption by an RTWAC platform
I_{TelosB}	Current consumption at a TelosB node
I_{wakeup}	Current consumption of a wake-up circuit board
L	Number of bits in a packet
L_i	Number of bits in a packet transmitted via radio interface i
$L_{\mathrm{max\_pkt}}$	Maximum number of bits in a packet
L_{mfp}	Length of a micro preamble frame

Table B.2: Notations in capital Latin letters used throughout this dissertation.

Symbol	Meaning
N_0	Noise power spectral density
N_{mfp}	Number of micro preamble frames
$\hat{N}_{\text{mfp}}$	Optimum number of micro preamble frames
N_{pkt}	Total number of packet repetitions
P_{cs}	Power consumption in performing carrier sensing operation
$P_{\text{LPL\_MAC}}$	Average power consumption by a low-power listening MAC
P_{out}	Power at the output of an antenna
P_{poll}	Power consumption in channel polling operation
P_{r}	Received power
$P_{\text{radio\_setup}}$	Power consumption in configuring a radio
$P_{\text{r}}(d)$	Power received at a distance d
$P_{\text{r}i}$	Received power at radio interface i
P_{rx}	Power consumption in reception mode
$P_{\text{rx\_}i}$	Power consumption in reception mode at radio interface i
P_{sleep}	Power consumption in sleep mode
P_{t}	Transmit power
$P_{\text{TelosB\_sleep}}$	Power consumption at TelosB in sleep mode
$P_{\text{t}i}$	Transmit power at radio interface i
P_{tx}	Power consumption in transmission mode
$P_{\text{tx\_}i}$	Power consumption in transmission mode at radio interface i
Q	Quality factor
R	Bit rate
R_{ant}	Antenna resistance
R_i	Bit rate of radio interface i
R_{vm}	Resistance of a matching network circuit
$S.T._{\text{opt}}$	Optimal sampling duration
T	Time interval
T_0	Thermal noise at an antenna
T_{0i}	Thermal noise at the antenna of radio interface i
$T_{\text{duty\_cycle}}$	Average latency associated with duty cycling operation
T_i	Time step i
T_{poll}	Time period for channel polling or preamble sampling operation
V_{ant}	Voltage induced at an antenna
V_{vm}	Output voltage at a voltage multiplier circuit
X_{match}	Reactance of a matching network circuit
X_{vm}	Reactance of a voltage multiplier circuit
Z_{ant}	Antenna impedance

Table B.3: Notations in small Latin letters used throughout this dissertation.

Symbol	Meaning
d	Distance between a transmitter and a receiver
k	A subset of the total number of nodes in a network
k_{B}	The Boltzmann constant
l_{ack}	Number of bits in an acknowledgement frame
$l_{\mathrm{ack\_}i}$	Acknowledgement frame length sent via radio interface i
l_i	Packet length transmitted via radio interface i
$l_{\mathrm{preamble\_frame}}$	Number of bits in a preamble frame
l_{pkt}	Number of bits in a packet
n	Number of nodes
n_{ch}	Number of channels
r_{b}	Bit rate of a radio
r_{data}	Data generation rate
t_{ack}	Time required for sending an acknowledgement frame
t_{b}	Time required for sending a single bit
$t_{\mathrm{b}i}$	Time required for sending a single bit at radio interface i
t_{cs}	Time required for performing carrier sensing operation
$t_{\mathrm{int\_timeout}}$	Interferer timeout interval
$t_{\mathrm{max\_pkt}}$	Maximum sized packet transmission duration
t_{pkt}	Packet transmission duration
t_{poll}	Time required for overall channel polling operation
$t_{\mathrm{poll\_once}}$	Time required for a single channel polling operation
t_{data}	Time required for sending data
$t_{\mathrm{radio\_setup}}$	Time required for setting up radio configuration
$t_{\mathrm{radio\_setup\_once}}$	Time required for setting up radio configuration only once
$t_{\mathrm{samp\_period}}$	Channel sampling period
t_{rx}	Time spent in receive mode
$t_{\mathrm{rx\_}i}$	Time spent in receive mode by radio interface i
t_{sleep}	Time spent in sleep mode
t_{switch}	Channel switching duration
t_{tx}	Time spent in transmit mode
$t_{\mathrm{tx\_}i}$	Time spent in transmit mode by radio interface i

Table B.4: Notations in small Greek letters used throughout this dissertation.

Symbol	Meaning
λ	Wavelength
λ_i	Transmission wavelength used by radio interface i
τ_i	Threshold i for the number of channels in a pool

C

CODE DEVELOPMENTS

- TrawMAC code in TinyOS 2.x for TelosB nodes.
- MR-MAC code in TinyOS 2.x.
- RTWAC transmitter code in TinyOS 2.x.
- RTWAC receiver code in TinyOS 2.x.
- SA-MAC code in TinyOS 2.x for TelosB nodes.
- WiSpot code in TinyOS 2.1.
- CogMAC code for Decomposable MAC Framework with reference design v14.0 for WARP v1.2.
- CogMAC code for Decomposable MAC Framework with reference design v16.1 for WARP v1.2.
- MAC component API implementation in Decomposable MAC Framework for WARP v1.2.
- Spectrum scanning application using TelosB nodes in TinyOS 2.x.
- Spectrum scanning application using WARP v1.2 boards.
- Enhanced CC2420 driver implementation in TinyOS 2.x for TelosB platforms
- CC1000 driver implementation in TinyOS 2.x for MSP430
- Adapters for measuring online energy consumption on TelosB nodes in TinyOS 2.x and data logging application.
- Continuous signal generators for 2.4 GHz based on TelosB nodes. This also supports user specified `ON-OFF` patterns with one of the selected random distributions.
- MAC-PD executable and component library in TinyOS 2.x.

D

ABBREVIATIONS

ACK Acknowledgement

ACM Association for Computing Machinery

AP Access Point

ASIC Application Specific Integrated Circuit

ASK Amplitude Shift Keying

AWGN Additive White Gaussian Noise

BAN Body Area Network

BCAST Broadcast

BPSK Binary Phase Shift Keying

BS Base Station

BT Bluetooth

CA Collision Avoidance

CB Component Based

CCA Clear Channel Assessment

CCBR Content and Context Based Routing

CCC Common Control Channel

CDF Cumulative Distribution Function

CMF Common Multiple Folding

CONFab Component based Optimization for Networks

COTS Commercial off-the-shelf

CR Cognitive Radio

CRC Cyclic Redundancy Check

CS Carrier Sensing

CSMA Carrier Sense Multiple Access

CTS Clear To Send

DC Duty Cycle

DCS Dynamic Channel Selection

DECT Digital Enhanced Cordless Telecommunications

DFG Deutsche Forschungsgemeinschaft

DFP Data Frame Preamble

DIFS DCF Interframe Space

DSA Dynamic Spectrum Access

EEPROM Electrically Erasable Programmable Read-Only Memory

ELF Executable and Linkable Format

ERP Effective Radiated Power

EU European Union

FEC Forward error correction

FIFO First In First Out

FM Frequency Modulation

FPGA Field-programmable Gate Array

FSK Frequency Shift Keying

GSM Global System for Mobile communications

GUI Graphical User Interface

GPIO General Purpose Input Output

HFR High Frequency Radio

HW Hardware

IC Integrated Circuit

ID Identity

IDE Integrated Development Environment

IEEE Institute of Electrical and Electronics Engineers

IP Internet Protocol

ISM Industrial, Scientific, and Medical

ISR Interrupt Service Routine

LFR Low Frequency Radio

LLC Logical Link Control

LPL Low Power Listening

LPM Low Power Mode

LOS Line Of Sight

LQI Link Quality Indicator

MAC Medium Access Control

MAC-PD MAC Protocol Designer

MChMAC Multi Channel MAC

MFP Micro Frame Preamble

MLA MAC Layer Architecture

MLE Maximum Likelihood Estimation

MR-MAC Multi-Radio Medium Access Control

NIC Network Interface Card

OFDM Orthogonal Frequency Division Multiplexing

OOK On-Off-Keying

OQPSK Offset Quadrature Phase Shift Keying

OS Operating System

P2P Peer to peer

PC Personal Computer

PGR Packet Generation Rate

PHY Physical layer

PIE Pulse Interval Encoding

PLL Phase Locked Loop

ppm parts per million

PS Preamble Sampling

PSD Power Spectral Density

PU Primary User

QoS Quality of Service

QPSK Quadrature Phase Shift Keying

RAM Random Access Memory

REM Radio Environment Map

RF Radio Frequency

RFID Radio Frequency Identification

RIP Receive-in-progress

RNG Random Number Generator

RPC Remote Procedure Call

ROM Read Only Memory

RSSI Received Signal Strength Indicator

RTID Radio Triggered Identification

RTS Request To Send

RTWAC Radio Triggered Wake-up with Addressing Capabilities

RX Receiver

SA Spectrum Analyzer

SA-MAC Spectrum Agile Medium Access Control

SAW Surface Acoustic Wave

SDR Software Defined Radio

SFD Start of Frame Delimiter

SIFS Short Interframe Space

SNR Signal to Noise Ratio

SPI Serial Peripheral Interface

SPICE Simulation Program with Integrated Circuit Emphasis

SU Secondary User

SW Software

SYNC Synchronization

TCP Transport Control Protocol

TDMA Time Division Multiple Access

TrawMAC Traffic Aware Medium Access Control

TRUMP Toolchain for RUntiMe Protocol realization

TX Transmitter

UART Universal Asynchronous Receiver/Transmitter

UCAST Unicast

UDP User Datagram Protocol

UDPB UDP traffic generation using IEEE 802.11b

UDPG UDP traffic generation using IEEE 802.11g

UHF Ultra High Frequency

UMIC Ultra Highspeed Mobile Information and Communication

UMTS Universal Mobile Telecommunications System

UPMA Unified Power Management Architecture

USB Universal Serial Bus

USRP Universal Software Radio Peripheral

VM Voltage Multiplier

WARP Wireless Open-Access Research Platform

WG Working Group

WiMAX Worldwide Interoperability for Microwave Access

WLAN Wireless Local Area Network

WRAN Wireless Regional Area Networks

w.r.t. with respect to

WSN Wireless Sensor Network

XML Extensible Markup Language

BIBLIOGRAPHY

[1] A. Bachir, M. Heusse, A. Duda, and K. Leung, "Preamble sampling MAC protocols with persistent receivers in wireless sensor networks," *Transactions on Wireless Communications*, vol. 8, no. 3, pp. 1091–1095, 2009.

[2] B. Yahya and J. Ben-Othman, "Towards a classification of energy aware MAC protocols for wireless sensor networks," *Wireless Communications and Mobile Computing*, vol. 9, no. 12, pp. 1572–1607, 2009.

[3] I. Demirkol, C. Ersoy, and F. Alagoz, "MAC protocols for wireless sensor networks: a survey," *IEEE Communications Magazine*, vol. 44, no. 4, pp. 115–121, 2006.

[4] J. Ansari, X. Zhang and P. Mähönen, "Practical Experiences and Design Considerations on Medium Access Control Protocols for Wireless Sensor Networks in Handbook of Research on Developments and Trends in Wireless Sensor Networks: From Principle to Practice by H. Jin and W. Jiang (editors)," *IGI Global*, 2010.

[5] C. Cormio and K. Chowdhury, "A survey on MAC protocols for cognitive radio networks," *Ad Hoc Networks*, vol. 7, no. 7, pp. 1315–29, 2009.

[6] M. Ali, U. Saif, A. Dunkels, T. Voigt, K. Römer, K. Langendoen, J. Polastre, and Z. A. Uzmi, "Medium Access Control Issues in Sensor Networks," *SIGCOMM Computer Communication Review*, vol. 36, no. 2, pp. 33–36, 2006.

[7] J.-H. Hauer, V. Handziski, and A. Wolisz, "Experimental Study of the Impact of WLAN Interference on IEEE 802.15.4 Body Area Networks," in *Proceedings of the European Conference on Wireless Sensor Networks*, Cork, Ireland, 2009.

[8] S. Y. Shin, H. S. Park, and W. H. Kwon, "Mutual interference analysis of IEEE 802.15.4 and IEEE 802.11b," *Computer Networks*, vol. 51, no. 12, pp. 3338–3353, 2007.

[9] R. Jurdak, *Wireless Ad Hoc and Sensor Networks: A Cross-Layer Design Perspecitve*. Springer-Verlag, 2007.

[10] J. Ansari, T. Ang and P. Mähönen, "Spectrum Agile Medium Access Control Protocol for Wireless Sensor Networks," in *Proceedings of the IEEE Sensor, Mesh and Ad Hoc Communications and Networks*, Boston, Massachusetts, USA, 2010.

[11] M. Petrova, J. Riihijärvi, P. Mähönen, and S. Labella, "Performance study of IEEE 802.15.4 using measurements and simulations," in *Proceedings of the IEEE Wireless Communications and Networking Conference*, Las Vegas, Nevada, USA, 2006.

[12] R. Szewczyk, A. Mainwaring, J. Polastre, J. Anderson, and D. Culler, "An analysis of a large scale habitat monitoring application," in *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems*, Baltimore, Maryland, USA, 2004.

[13] "Indriya TelosB testbed at National University of Singapore," http://indriya.comp.nus.edu.sg/motelab/html/index.php [Last visited: 02.08.2011].

[14] *TelosB Wireless Sensor Node*, MEMSIC Inc., http://memsic.com/products/wireless-sensor-networks/wireless-modules.html, Last visited: 11.10.2011.

[15] A. Khattab, J. Camp, C. Hunter, P. Murphy, A. Sabharwal and E. W. Knightly, "WARP: a flexible platform for clean-slate wireless medium access protocol design," *SIGMOBILE Mobile Computing Communication Review*, vol. 12, no. 1, pp. 56–58, 2008.

[16] G. Halkes and K. Langendoen, "Practical Considerations for Wireless Sensor Network Algorithms," *Wireless Sensor Network*, vol. 2, no. 6, pp. 441–446, 2010.

[17] X. Zhang, J. Ansari and P. Mähönen, "Traffic Aware Medium Access Control Protocol for Wireless Sensor Networks," in *Proceedings of the 7th ACM International Symposium on Mobility Management and Wireless Access*, Tenerife, Canary Islands, Spain, 2009.

[18] J. Ansari, X. Zhang and P. Mähönen, "Poster Abstract: Traffic Aware Medium Access Control Protocol for Wireless Sensor Networks," in *European Professional Society on Computer Systems*, Glasgow, Scotland, U.K., 2008.

[19] ——, "Multi-radio Medium Access Control Protocol for Wireless Sensor Networks," *International Journal of Sensor Networks*, vol. 8, no. 1, pp. 47–61, 2010.

[20] J. Ansari, D. Pankin and P. Mähönen, "Radio triggered wake-ups with addressing capabilities for extermely low power sensor network applications," *International Journal of Wireless Information Networks*, vol. 16, no. 3, pp. 118–130, 2009.

[21] F. Oldewurtel, J. Ansari, and P. Mähönen, "Cross-layer design for distributed source coding in wireless sensor networks," in *Proceedings of the International Conference on Sensor Technologies and Applications*, Cap Esterel, France, 2008.

[22] J. G. Wedster (Ed.), *The Measurement, Instrumentation and Sensors Handbook*. CRC Press, 1998.

[23] S. Soloman, *Sensors Handbook*. McGraw-Hill Professional, 2009.

[24] O. Landsiedel, K. Wehrle, and S. Götz, "Accurate prediction of power consumption in sensor networks," in *Proceedings of the 2nd IEEE Workshop on Embedded Networked Sensors*, Sydney, Australia, 2005.

[25] N. Alsharabi, L. R. Fa, F. Zing, and M. Ghurab, "Wireless sensor networks of battlefields hotspot: Challenges and solutions," in *Proceedings of the International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks*, Berlin, Germany, 2008.

[26] S. Kim, S. Pakzad, D. Culler, J. Demmel, G. Fenves, S. Glaser, and M. Turon, "Health monitoring of civil infrastructures using wireless sensor networks," in *Proceedings of the International Conference on Information Processing in Sensor Networks*, Cambridge, Massachusetts, USA, 2007.

[27] M. C. Mozer., *Lessons from an adaptive house. In Smart environments: Technologies, protocols, and applications*. J. Wiley & Sons., 2004.

[28] L. Chen, Z. Chen, and S. Tu, "A realtime dynamic traffic control system based on wireless sensor network," in *Proceedings of the International Conference on Parallel Processing*, Oslo, Norway, 2005.

[29] W. Ye, J. Heidemann and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks," in *Proceedings of the IEEE Conference on Computer Communications*, New York, NY, USA, 2002.

[30] K. Langendoen, *Medium Access Control in Wireless Networks, Volume II: Practice and Standards*, 1st ed. Nova Science Publishers, 2007.

[31] V. Rajendran, K. Obraczka, and J. Garcia-Luna-Aceves, "Energy-efficient, collision-free medium access control for wireless sensor networks," *Wireless Networks*, vol. 12, no. 1, pp. 63–78, 2006.

[32] L. v. Hoesel and P. Havinga, "A lightweight medium access protocol (LMAC) for wireless sensor networks," in *Proceedings of the International Conference on Networked Sensing Systems*, Tokyo, Japan, 2004.

[33] M. Ringwald and K. Romer, "BitMAC: a deterministic, collision-free, and robust MAC protocol for sensor networks," in *Proceeedings of the Second European Wireless Sensor Networks Conference*, Istanbul, Turkey, 2005.

[34] J. Ansari, J. Riihijärvi, P. Mähönen and J. Haapola, "Implementation and Performance Evaluation of nanoMAC: A Low-Power MAC Solution for High Density Wireless Sensor Networks," *International Journal of Sensor Networks*, vol. 2, no. 5, pp. 341–9, 2007.

[35] T. van Dam and K. Langendoen, "An adaptive energy-efficient MAC protocol for wireless sensor networks," in *Proceedings of the International Conference on Embedded Networked Sensor Systems*, Los Angeles, California, USA, 2003.

[36] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," in *Proceedings of the International Conference on Embedded Networked Sensor Systems*, Baltimore, Maryland, USA, 2004.

[37] J. Polastre, "Sensor network media access design," University of California, Berkeley, Tech. Rep., 2003.

[38] A. El-Hoiydi and J.-D. Decotignie, "WiseMAC: an ultra low power MAC protocol for the downlink of infrastructure wireless sensor networks," in *Proceedings of the International Symposium on Computers and Communications*, Alexandria, Egypt, 2004.

[39] I. Rhee, A. Warrier, M. Aia, J. Min, and M. Sichitiu, "Z-MAC: a hybrid MAC for wireless sensor networks," in *Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems*, San Diego, California, USA, 2005.

[40] A. Ruzzelli, G. O'Hare, R. Jurdak, and R. Tynan, "Advantages of dual channel MAC for wireless sensor networks," in *Proceedings of the International Conference on Communication System Software and Middleware*, New Delhi, India, 2006.

[41] Y. Sun, O. Gurewitz, and D. B. Johnson, "RI-MAC: a receiver-initiated asynchronous duty cycle MAC protocol for dynamic traffic loads in wireless sensor networks," in *Proceedings of the 6th ACM conference on Embedded network sensor systems*, Raleigh, North Carolina, USA, 2008.

[42] Texas Instruments Inc. CC2420 Radio Transceiver Datasheet, "http://focus.ti.com/lit/ds/symlink/cc2420.pdf," Last visited: 27.12.2010.

[43] A. Bachir, D. Barthel, M. Heusse, and A. Duda, "Micro-frame preamble MAC for multihop wireless sensor networks," in *Proceedings of the IEEE International Conference on Communications*, Istanbul, Turkey, 2006.

[44] M. Buettner, G. V. Yee, E. Anderson, and R. Han, "X-MAC: a short preamble MAC protocol for duty-cycled wireless sensor networks," in *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems*, Boulder, Colarado, USA, 2006.

[45] M. Avvenuti, P. Corsini, P. Masci and A. Vecchio, "Increasing the efficiency of preamble sampling protocols for wireless sensor networks," in *Proceedings of the First International Conference on Mobile Computing and Wireless Communication*, Amman, Jordan, 2006.

[46] A. El-Hoiydi, J.-D. Decotignie, C. Enz, and E. L. Roux, "Poster abstract: WiseMAC, an ultra low power MAC protocol for the WiseNET wireless sensor network," in *Proceedings of the International Conference on Embedded Networked Sensor Systems*, Los Angeles, California, USA, 2003.

[47] W. Ye, F. S, and J. Heidemann, "Ultra-low duty cycle MAC with scheduled channel polling," in *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems*, Boulder, Colarado, USA, 2006.

[48] G.-S. Ahn, S. G. Hong, E. Miluzzo, A. T. Campbell, and F. Cuomo, "Funneling-MAC: a localized, sink-oriented MAC for boosting fidelity in sensor networks," in *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems*, Boulder, Colarado, USA, 2006.

[49] J. Li and G. Y. Lazarou, "A bit-map-assisted energy-efficient MAC scheme for wireless sensor networks," in *Proceedings of the International Conference on Information Processing in Sensor Networks*, Berkeley, California, USA, 2004.

[50] M. Ringwald and K. Romer, "BurstMAC – An efficient MAC protocol for correlated traffic bursts," in *Proceedings of the Sixth International Conference on Networked Sensing Systems*, Pittsburgh, Pennsylvania, USA, 2009.

[51] P. Hurni and T. Braun, "MaxMAC: A Maximally Traffic-Adaptive MAC Protocol for Wireless Sensor Networks," in *Proceedings of the 7th European Conference on Wireless Sensor Networks Conference*, Coimbra, Portugal, 2010.

[52] S. H. Lee, J. H. Park, and L. Choi, "AMAC: Traffic-Adaptive Sensor Network MAC Protocol through Variable Duty-Cycle Operations," in *Proceedings of the IEEE International Conference on Communications*, Glasgow, Scotland, U.K., 2007.

[53] Z. Fu, Y. Yang, T. Lee and M-S. Park, "Adapted Listening in Wireless Sensor Network MAC Protocol," *Ubiquitous Intelligence and Computing*, vol. 4159, no. 6, pp. 546–555, 2006.

[54] S. Bac, D. Kwak, and C. Kim, "Traffic-Aware MAC Protocol Using Adaptive Duty Cycle for Wireless Sensor Networks," in *Proceedings of the International Conference on Information Networking*, Estoril, Portugal, 2007.

[55] F. Österlind, N. Wirström, N. Tsiftes, N. Finne, T. Voigt and A. Dunkels, "StrawMAN: Making Sudden Traffic Surges Graceful in Low-Power Wireless Networks," in *Proceedings of the Workshop on Hot Topics in Embedded Networked Sensors*, Monterey, California, USA, 2010.

[56] C. Schurgers, V. Tsiatsis, S. Ganeriwal and M. Srivastava, "Optimizing sensor networks in the energy-latency-density design space," *IEEE Transactions on Mobile Computing*, vol. 1, no. 1, pp. 70–80, 2002.

[57] C. Doerr, M. Neufeld, J. Fifield, T. Weingart, D. Sicker, and D. Grunwald, "Multimac - an adaptive mac framework for dynamic radio networking," in *Proceedings of the IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks*, Baltimore, Maryland, USA, 2005.

[58] K. Langendoen and A. Meier, "Analyzing mac protocols for low data-rate applications," *ACM Transactions on Sensor Networks*, vol. 7, no. 2, pp. 1–40, 2010.

[59] Y. Panthachai and P. Keeratiwintakorn, "An Energy Model for Transmission in Telos-based Wireless Sensor Networks," in *Proceedings of the 4th International Joint Conference on Computer Science and Software Engineering*, Nakorn, Thailand, 2007.

[60] "OMNeT++, Discrete Event Simulation System," http://www.omnetpp.org Last visited: 15.06.2011.

[61] B. Krishnamachari, D. Estrin, and S. Wicker, "The impact of data aggregation in wireless sensor networks," in *Proceedings of the 22nd International Conference on Distributed Computing Systems*, Vienna, Austria, 2002.

[62] G. Cugola and M. Migliavacca, "A context and content-based routing protocol for mobile sensor networks," in *Proceedings of the European Conference on Wireless Sensor Networks Conference*, Cork, Ireland, 2009.

[63] P. Levis and D. Gay, *TinyOS Programming*. Cambridge University Press, 2009.

[64] European Union funded Wirelessly Accessible Sensor Population (WASP) Project, "http://www.wasp-project.org/ [Last visited: 02.08.2011]."

[65] "Mobility Framework," http://mobility-fw.sourceforge.net/ [Last visted: 02.08.2011].

[66] *TinyOS: an open-source operating system designed for wireless embedded sensor networks*, http://www.tinyos.net/, Last visited: 21.10.2011.

[67] I. Oppermann, L. Stoica, A. Rabbachin, Z. Shelby, and J. Haapola, "UWB wireless sensor networks: UWEN-a practical example," *IEEE Communications Magazine*, vol. 42, no. 12, pp. 527–532, 2004.

[68] S. Mahlknecht and M. Boeck, "CSMA-MPS: A minimum preamble sampling MAC protocol for low power wireless sensor networks," in *Proceedings of the IEEE International Workshop on Factory Communication Systems*, Vienna, Austria, 2004.

[69] Texas Instrument Inc. CC1000 radio transceiver Datasheet, "http://focus.ti.com/lit/ds/symlink/cc1000.pdf," Last visited: 15.12.2010.

[70] Texas Instruments Inc. MSP430x1611 - Mixed Signal Microcontroller, "http://focus.ti.com/docs/prod/folders/print/msp430f1611.html," Last visited: 11.12.2010.

[71] TinyOS 2.0 TEP 2, "http://www.tinyos.net," Last visited: 15.10.2011.

[72] Atmel Corporation – ATmega 128L microcontroller, "http://www.atmel.com/dyn/resources/prod_documents/2467s.pdf," Last visited: 11.12.2010.

[73] S. Basagni, A. Carosi, and C. Petrioli, "Sensor-DMAC: dynamic topology control for wireless sensor networks," in *Proceedings of the 60th IEEE Vehicular Technology Conference*, Los Angeles, California, USA, 2004.

[74] M. Miladi, T. Ezzedine, and R. Bouallegue, "Latency of Energy Efficient MAC Protocols for Wireless Sensor Networks," in *Proceedings of International Conference on Digital Telecommunications*, Cap Esterel, France, 2006.

[75] L. Gu and J. Stankovic, "Radio-Triggered Wake-Up for Wireless Sensor Networks," *Real-Time Systems*, vol. 29, no. 2-3, pp. 157–182, 2005.

[76] B. Otis, "Ultra-low power wireless technologies for sensor networks," Ph.D. dissertation, University of California, Berkeley, 2005.

[77] J. Rabaey, J. Ammer, T. Karalar, S. Li, B. Otis, M. Sheets and T. Tuan, "PicoRadios for wireless sensor networks: the next challenge in ultra-low-power design," in *Proceedings of the IEEE International Solid-State Circuits Conference*, San Francisco, California, USA, 2002.

[78] M. J. Miller and N. H. Vaidya, "A MAC protocol to reduce sensor network energy consumption using a wakeup radio," *IEEE Transactions on Mobile Computing*, vol. 4, no. 3, pp. 228–242, 2005.

[79] M. Malinowski, M. Moskwa, M. Feldmeier, M. Laibowitz and J. Paradiso, "CargoNet: A low-cost micropower sensor node exploiting quasi-passive wakeup for adaptive asynchronous monitoring of exceptional events," in *Proceedings of the 5th ACM Conference on Embedded Networked Sensor Systems*, Sydney, Australia, 2007.

[80] A. Sample, D. Yeager, P. Powledge, and J. Smith, "Design of a Passively-Powered, Programmable Sensing Platform for UHF RFID Systems," in *Proceedings of the IEEE International Conference on RFID*, Grapevine, Texas, USA, 2007.

[81] B. van der Doorn, W. Kavelaars and K. Langendoen, "A Prototype Low-Cost Wakeup Radio for the 868MHz Band," *International Journal of Sensor Networks*, vol. 5, no. 1, pp. 22–32, 2009.

[82] Atmel Corporation, "U3280M transponder interface for microcontroller, datasheet," http://atmel.com/dyn/resources/prod_documents/doc4688.pdf. Last visited: 21.10.2011.

[83] C. A. Balanis, *Antenna theory: analysis and design*. John Wiley & Sons, Ltd, 1996.

[84] Mini-Circuits, "ZHL-2010, datasheet," http://www.minicircuits.com/pdfs/ZHL-2010.pdf. Last visited: 21.10.2011.

[85] Texas Instruments' CC1000PP User Manual, "http://focus.ti.com/lit/ug/swru060/swru060.pdf," Last visited: 15.12.2010.

[86] Avago Technologies, "Surface mount zero bias schottky detector diodes, HSMS-285x series, datasheet," http://www.avagotech.com/assets/downloadDocument.do?id=2475. Last visited: 21.10.2011.

[87] Maxim Integrated Products, "Nanopower, beyond-the-rails comparators MAX9117 - MAX9120, datasheet," http://datasheets.maxim-ic.com/en/ds/MAX9117-MAX9120.pdf. Last visited: 21.10.2011.

[88] Simulation Program with Integrated Circuit Emphasis for PCs from Cadence Design Systems, Inc., "http://www.cadence.com/products/orcad/pages/downloads.aspx," Last visited: 11.10.2011.

[89] J. Ansari, X. Zhang and P. Mähönen, "Demo abstract: Multi-radio medium access control protocol for wireless sensor networks," in *Proceedings of the 5th International Conference on Embedded Networked Sensor Systems*, Sydney, Australia, 2007.

[90] J. Ansari, D. Pankin and P. Mähönen, "Demo abstract: Radio-triggered wake-ups with addressing capabilities for extremely low power sensor network applications," in *5th European Conference on Wireless Sensor Networking (Demonstration)*, Bologna, Italy, 2008.

[91] J. Mitola, "An integrated agent architecture for software defined radio," Ph.D. dissertation, 2000.

[92] J. Ansari and P. Mähönen, "Channel Selection in Spectrum Agile and Cognitive MAC Protocols for Wireless Sensor Networks," in *Proceedings of the ACM International Symposium on Mobility Management and Wireless Access*, Bodrum, Turkey, 2010.

[93] J. Ansari, X. Zhang and P. Mähönen, "A Decentralized MAC for Opportunistic Spectrum Access in Cognitive Wireless Networks," in *Proceedings of the ACM SIGMOBILE Workshop on Cognitive Wireless Networking*, Chicago, Illinois, USA, 2010.

[94] J. Ansari, T. Ang and P. Mähönen, "WiSpot: Fast and Reliable Detection of Wi-Fi Networks Using IEEE 802.15.4 Radios," in *Proceedings of the ACM International Symposium on Mobility Management and Wireless Access*, Miami Beach, Florida, USA, 2011.

[95] J. Ko and A. Terzis, "Power control for mobile sensor networks: An experimental approach," in *Proceedings of the IEEE Sensor, Mesh and Ad Hoc Communications and Networks*, Boston, Massachusetts, USA, 2010.

[96] M. Wellens, J. Riihijärvi, and P. Mähönen, "Empirical time and frequency domain models of spectrum use," *Physical Communication*, vol. 2, no. 1-2, pp. 10 – 32, 2009.

[97] J. I. Choi, M. Jain, K. Srinivasan, P. Levis, and S. Katti, "Achieving single channel, full duplex wireless communication," in *Proceedings of the International Conference on Mobile Computing and Networking*, Chicago, Illinois, USA, 2010.

[98] M. Duarte and A. Sabharwal, "Full-duplex wireless communications using off-the-shelf radios: Feasibility and first results," in *Proceedings of the Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, California, USA, 2010.

[99] H. Zhai, J. Wang, Y. Fang, and D. Wu, "A dual-channel MAC protocol for mobile ad hoc networks," in *Proceedings of the IEEE Global Telecommunications Conference Workshop*, Dallas, Texas, USA, 2004.

[100] C. Cordeiro, K. Challapali, D. Birru and S. Shankar, "IEEE 802.22: The first world-wide wireless standard based on cognitive radios," in *Proceedings of the IEEE International Dynamic Spectrum Access Networks Symposium*, Baltimore, Maryland, USA, 2005.

[101] M. Salajegheh, H. Soroush and A. Kalis, "HyMAC: Hybrid TDMA/FDMA Medium Access Control Protocol for Wireless Sensor Networks," in *Proceedings of the IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, Athens, Greece, 2007.

[102] W. Yafeng, J. Stankovic, H. Tian and L. Shan, "Realistic and Efficient Multi-Channel Communications in Wireless Sensor Networks," in *Proceedings of the IEEE International Conference on Computer Communications*, Phoenix, Arizona, USA, 2008.

[103] H. Pham and S. Jha, "An adaptive mobility-aware MAC protocol for sensor networks (MS-MAC)," in *Proceedings of the IEEE International Conference on Mobile Ad-hoc and Sensor Systems*, Fort Lauderdale, Florida, USA, 2004.

[104] M. Ali and Z. A. Uzmi, "Medium access control with mobility adaptive mechanisms for wireless sensor networks," *International Journal of Sensor Networks*, vol. 1, no. 3/4, pp. 134–142, 2006.

[105] W. Xu, W. Trappe, and Y. Zhang, "Defending wireless sensor networks from radio interference through channel adaptation," *ACM Transactaions on Sensor Networks*, vol. 4, no. 4, pp. 1–34, 2008.

[106] I. Akyildiz, W.-Y. Lee, M. Vuran, and S. Mohanty, "A survey on spectrum management in cognitive radio networks," *IEEE Communications Magazine*, vol. 46, no. 4, pp. 40–48, 2008.

[107] A.-Y. Lien, C.-C. Tseng and K.-C. Chen, "Carrier sensing based multiple access protocols for cognitive radio networks," in *Proceedings of the International Conference on Communications*, Beijing, China, 2008.

[108] C. Zhou and C. Chigan, "A game theoretic DSA-driven MAC framework for cognitive radio networks," in *Proceedings of the International Conference on Communications*, Beijing, China, 2008.

[109] J. Xiang, Y. Zhang and T. Skeie, "Medium access control protocols in cognitive radio networks," *Wireless Communication and Mobile Computing*, vol. 10, no. 1, pp. 31–49, 2010.

[110] M. Wellens, J. Riihijärvi and P. Mähönen, "Evaluation of Adaptive MAC-Layer Sensing in Realistic Spectrum Occupancy Scenarios," in *Proceedings of the IEEE Symposium on New Frontiers in Dynamic Spectrum*, Singapore, 2010.

[111] A. Nasipuri, J. Zhuang and Das, S., "A multichannel CSMA MAC protocol for multihop wireless networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference*, New Orleans, Louisiana, USA, 1999.

[112] N. Jain and S. Das, "A multichannel CSMA MAC protocol with receiver-based channel selection for multihop wireless networks," in *Proceedings of the International Conference on Computer Communications and Networks*, Las Vegas, Nevada, USA, 2001.

[113] S.-L. Wu, C.-Y. Lin, Y.-C. Tseng and J.-P. Sheu, "A new Multi-Channel MAC Protocol with On-Demand Channel Assignment for Multi-hop Mobile Ad Hoc Networks," in *Proceedings of the International Symposium on Parallel Architecture, Algorithms and Networks*, Dallas/Richardson, Texas, USA, 2000.

[114] I.F. Akyildiz, W.-Y. Lee, and K.R. Chowdhury, "CRAHNs: Cognitive radio ad hoc networks," *Ad Hoc Networks*, vol. 7, no. 5, pp. 810–836, 2009.

[115] C. Cordeiro and K. Challapali, "C-MAC: A Cognitive MAC Protocol for Multi-Channel Wireless Networks," in *Proceedings of the IEEE International Dynamic Spectrum Access Networks Symposium*, Dublin, Ireland, 2007.

[116] B. Hamdaoui and K. G. Shin, "OS-MAC: An Efficient MAC Protocol for Spectrum-Agile Wireless Networks," *IEEE Transactions on Mobile Computing*, vol. 7, no. 8, pp. 915–930, 2008.

[117] G. Zhou, C. Huang, T. Yan, T. He, J. A. Stankovic and T. F. Abdelzaher, "MMSN: Multi-frequency media access control for wireless sensor networks," in *Proceedings of the IEEE Conference on Computer Communications*, Barcelona, Spain, 2006.

[118] Y. Kim, H. Shin, and H. Cha, "Y-MAC: An Energy-Efficient Multi-channel MAC Protocol for Dense Wireless Sensor Networks," in *Proceedings of the International Conference on Information Processing in Sensor Networks*, St. Louis, Missouri, USA, 2008.

[119] Y. Wu, M. Keally, G. Zhou and W. Mao, "Traffic-aware channel assignment in wireless sensor networks," in *Proceedings of the 4th International Conference on Wireless Algorithms, Systems, and Applications*, Boston, Massachusetts, USA, 2009.

[120] C. Guo, L. C. Zhong, and J. Rabaey, "Low power distributed MAC for ad hoc sensor radio networks," in *Proceedings of the IEEE Global Communications Conference*, San Antonio, Texas, USA, 2001.

[121] K. R. Chowdhury, N. Nandiraju, P. Chanda, D. P. Agrawal and Q. Zeng, "Channel allocation and medium access control for wireless sensor networks," *Ad Hoc Networks*, vol. 7, no. 2, pp. 307 – 321, 2009.

[122] J. Borms, K. Steenhaut and B. Lemmens, "Low-overhead Dynamic Multi-Channel MAC for Wireless Sensor Networks," in *Proceedings of the European Conference on Wireless Sensor Networks*, Coimbra, Portugal, 2010.

[123] H. Sheung, W. So, J. Walrand and J. Mo, "McMAC: A Parallel Rendezvous Multi-Channel MAC Protocol," in *Proceedings of the IEEE Wireless Communications and Networking Conference*, Hong Kong, China, 2007.

[124] Z. Liu and W. Wu, "A Dynamic Multi-radio Multi-channel MAC Protocol for Wireless Sensor Networks," in *Proceedings of the International Conference on Communication Software and Networks*, Singapore, 2010.

[125] C. Liang, N. Priyantha, J. Liu, and A. Terzis, "Surviving Wi-Fi Interference in Low Power ZigBee Networks," in *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, Zurich, Switzerland, 2010.

[126] W. Yuan, X. Wang, and J.-P. Linnartz, "A Coexistence Model of IEEE 802.15.4 and IEEE 802.11b/g," in *14th IEEE Symposium on Communications and Vehicular Technology in the Benelux*, Delft, The Netherlands, 2007.

[127] "Chipcon CC2421 Application Note 41," http://www.chipcon.com [Last visted: 02.08.2011].

[128] L. Cao, L. Yang, and H. Zheng, "The Impact of Frequency-Agility on Dynamic Spectrum Sharing," in *Proceedings of the IEEE Symposium on New Frontiers in Dynamic Spectrum*, Singapore, 2010.

[129] J. Ansari, X. Zhang, A. Achtzehn, M. Petrova, and P. Mähönen, "Decomposable MAC Framework for Highly Flexible and Adaptable MAC Realizations," in *Proceedings of the IEEE Symposium on New Frontiers in Dynamic Spectrum*, Singapore, 2010.

[130] J. Ansari, X. Zhang, and P. Mähönen, "Demo Abstract: A Decentralized MAC Protocol for Opportunistic Spectrum Access in Cognitive Wireless Networks." in *IEEE International Conference on Computer Communications*, San Diego, California, USA, 2010.

[131] C. A. Boano, T. Voigt, N. Tsiftes, L. Mottola, K. Roemer, and M. A. Zuniga, "Making Sensornet MAC Protocols Robust Against Interference," in *Proceedings of the European Conference on Wireless Sensor Networks*, Coimbra, Portugal, 2010.

[132] G. Zhou, J. A. Stankovic, and S. H. Son, "Crowded spectrum in wireless sensor networks," in *Proceedings of 3rd Workshop on Embedded Networked Sensors*, 2006.

[133] J. Riihijärvi, M. Petrova and P. Mähönen, "Frequency Allocation for WLANs Using Graph Colouring Techniques," *Ad Hoc & Sensor Wireless Networks*, vol. 3, no. 2-3, pp. 121–139, 2007.

[134] J. Huang, G. Xing, G. Zhou and R. Zhou, "Beyond Co-existence: Exploiting WiFi White Space for ZigBee Performance Assurance," in *Proceedings of the IEEE International Conference on Network Protocols*, Kyoto, Japan, 2010.

[135] T. Yücek and H. Arslan, "A Survey of Spectrum Sensing Algorithms for Cognitive Radio Applications," in *IEEE Communications Surveys & Tutorials*, 2009, pp. 116 – 130.

[136] K. Chowdhury and I. Akyildiz, "Interferer classification, channel selection and transmission adaptation for wireless sensor networks," in *IEEE International Conference on Communications*, Dresden, Germany, 2009.

[137] R. Zhou, Y. Xiong, G. Xing, L. Sun, and J. Ma, "ZiFi: Wireless LAN Discovery via ZigBee Interference Signatures," in *Proceedings of the 16th Annual International Conference on Mobile Computing and Networking*, Chicago, Illinois, USA, 2010.

[138] J. Ansari, X. Zhang, A. Achtzehn, M. Petrova and P. Mähönen, “A Flexible MAC Development Framework for Cognitive Radio Systems,” in *Proceedings of the IEEE Wireless Communications and Networking Conference*, Cancun, Mexico, 2011.

[139] J. Ansari, X. Zhang, O. Salikeen and P. Mähönen, “Enabling Flexible MAC Protocol Design for Wireless Sensor Networks,” in *Proceedings of the IEEE International Conference on Wireless On-demand Network Systems and Services*, Bardonecchia, Italy, 2011.

[140] X. Zhang, J. Ansari, G. Yang and P. Mähönen, “TRUMP: Supporting Efficient Realization of Protocols for Cognitive Radio Networks,” in *Proceedings of the IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks*, Aachen, Germany, 2011.

[141] J. Ansari, X. Zhang and P. Mähönen, “A Compiler Assisted Approach for Component based Reconfigurable MAC Design,” in *Proceedings of the 10th IEEE IFIP Annual Mediterranean Ad Hoc Networking Workshop*, Favignana Island, Sicily, Italy, 2011.

[142] G. Nychis, S. Seshan and P. Steenkiste, “Enabling MAC Protocol Implementations on Software-Defined Radios,” in *Proceedings of the Symposium on Networked Systems Design and Implementation*, Boston, Massachusetts, USA, 2009.

[143] J. R. Hauser and J. Wawrzynek, “Garp: a MIPS processor with a reconfigurable coprocessor,” in *Proceedings of the 5th IEEE Symposium on FPGA-Based Custom Computing Machines*, Napa Valley, California, USA, 1997.

[144] J. R. Polastre, “A unifying link abstraction for wireless sensor networks,” Ph.D. dissertation, Berkeley, CA, USA, 2005.

[145] K. Rerkrai, J. Riihijärvi, M. Wellens and P. Mähönen, “Unified Link-Layer API Enabling Portable Protocols and Applications for Wireless Sensor Networks,” in *Proceedings of the IEEE International Conference on Communications*, Glasgow, Scotland, U.K., 2007.

[146] K. Klues, G. Xing and C. Lu, “Towards a Unified Radio Power Management Architecture for Wireless Sensor Networks,” in *Proceedings of the International Workshop on Sensor Network Architecture*, Cambridge, Massachusetts, USA, 2007.

[147] G. Xing, M. Sha, G. Hackmann, K. Klues, O. Chipara, and C. Lu, “Towards unified radio power management for wireless sensor networks,” *Wireless Communication and Mobile Compututing*, vol. 9, no. 3, pp. 313–323, 2009.

[148] G. Bianchi and A. Campbell, "A programmable MAC framework for utility-based adaptive quality of service support," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 2, pp. 244–255, 2000.

[149] D. G. Messerschmitt, "Rethinking Components: From Hardware and Software to Systems," *Proceedings of the IEEE*, vol. 95, no. 7, pp. 1473–1496, 2007.

[150] K. Klues, G. Hackmann, O. Chipara, and C. Lu, "A Component Based Architecture for Power-Efficient Media Access Control in Wireless Sensor Networks," in *Proceedings of the ACM Conference on Sensor Systems*, Sydney, Australia, 2007.

[151] K. Klues, G. Xing, and C. Lu, "Link layer driver architecture for unified radio power management in wireless sensor networks," *ACM Transactions on Embedded Computer Systems*, vol. 9, no. 4, pp. 1–28, 2010.

[152] R. Braden, T. Faber, and M. Handley, "From protocol stack to protocol heap: role-based architecture," *SIGCOMM Computer Communication Review*, vol. 33, no. 1, pp. 17–22, 2003.

[153] A. Sharma and E. M. Belding, "Freemac: Framework for multi-channel mac development on 802.11 hardware," in *Proceedings of the ACM workshop on Programmable routers for extensible services of tomorrow*, Seattle, Washington, USA, 2008.

[154] M.-H. Lu, P. Steenkiste, and T. Chen, "FlexMAC: a wireless protocol development and evaluation platform based on commodity hardware," in *Proceedings of the third ACM International Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization*, San Francisco, California, USA, 2008.

[155] MadWiFi driver for Atheros chipsets, "http://madwifi.org/," Last visited: 11.10.2011.

[156] A. Sharma, M. Tiwari and H. Zheng, "MadMAC: Building a Reconfigurable Radio Testbed Using Commodity 802.11 Hardware," in *Proceedings of the IEEE Workshop on Networking Technologies for Software Defined Radio Networks*, Reston, Virginia, USA, 2006.

[157] P. Djukic and P. Mohapatra, "Soft-TDMAC: A Software TDMA-Based MAC over Commodity 802.11 Hardware," in *Procedings of the IEEE International Conference on Computer Communications*, Rio de Janeiro, Brazil, 2009.

[158] E. Blossom, "GNU radio: tools for exploring the radio frequency spectrum," *Linux Journal*, no. 122, p. 4, 2004.

[159] "The USRP Board," https://radioware.nd.edu/documentation/hardware/the-usrp-board [Last visited: 21.10.2011].

[160] M. Neufeld, J. Fifield, C. Doerr, and A. Sheth, "SoftMAC: A Flexible Wireless Research Platform," in *Proceedings of the 4th ACM workshop on Hot Topics in Networking*, College Park, Maryland, USA, 2005.

[161] C. Hunter, P. Murphy and A. Sabharwal, "Real-time testbed implementation of a distributed cooperative MAC and PHY," in *Proceedings of the IEEE Conference on Information Sciences and Systems*, Chengdu, China, 2010.

[162] P. K. Mckinley, S. M. Sadjadi, E. P. Kasten and B. H. C. Cheng, "A Taxonomy of Compositional Adaption," Michigan State University, MSU-CSE-04-17, Tech. Rep., 2004.

[163] E. Kohler, R. Morris, B. Chen, J. Jannotti, and M. F. Kaashoek, "The click modular router," *ACM Transactions Computer Systems*, vol. 18, no. 3, pp. 263–297, 2000.

[164] H. S. Lichte, S. Valentin, and H. Karl, "Automated development of cooperative MAC protocols: A compiler-assisted approach," *Mobile Networks and Applications*, vol. 15, no. 6, pp. 769–785, 2010.

[165] "WARP Board OFDM reference design," http://warp.rice.edu/trac/wiki/OFDMReferenceDesign [Last visted: 02.08.2011].

[166] "ARM Compiler Tips and Code Size Optimization Using DSP/BIOS Link," http://focus.ti.com.cn/cn/lit/an/spraac3/spraac3.pdf [Last visited: 21.10.2011].

[167] X. Zhang, J. Ansari, G. Yang and P. Mähönen, "TRUMP: Supporting Efficient Realization of Protocols for Cognitive Radio Networks," in *IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks (Demonstration)*, Aachen, Germany, 2011.

[168] *TelosA Wireless Sensor Node*, MEMSIC Inc., http://memsic.com/products/wireless-sensor-networks/wireless-modules.html, Last visited: 11.10.2011.

[169] *MICAz Wireless Sensor Node*, MEMSIC Inc., http://memsic.com/products/wireless-sensor-networks/wireless-modules.html, Last visited: 11.10.2011.

[170] *Body Sensor Node*, Imperial College London, the U.K., http://www.doc.ic.ac.uk/vip/ubimon/bsn_node/index.html, Last visited: 11.10.2011.

[171] R. Adler, M. Flanigan, J. Huang, R. Kling, N. Kushalnagar, L. Nachman, C.-Y. Wan, and M. Yarvis, "Intel mote 2: an advanced platform for demanding sensor network applications," in *Proceedings of the International Conference on Embedded Networked Sensor Systems*, San Diego, California, USA, 2005.

[172] MAC Layer Abstraction code repository, "http://tinyos.cvs.sourceforge.net/tinyos/tinyos-2.x-contrib/wustl/upma/," Last visited: 14.08.2010.

[173] O. Salikeen, J. Ansari, X. Zhang and P. Mähönen, "Demo Abstract: Enabling Flexible Medium Access Design for Wireless Sensor Networks," in *Proceedings of the ACM Conference on Sensor Systems*, Zurich, Switzerland, 2010.

[174] X. Lin, N. B. Shroff, and R. Srikant, "A tutorial on cross-layer optimization in wireless networks," *IEEE Journal on Selected Areas in Communication*, vol. 24, pp. 1452–1463, 2006.

[175] M. Chiang, S. H. Low, R. A. Calderbank, and J. C. Doyle, "Layering as optimization decomposition," *Proceedings of the IEEE*, vol. 95, pp. 255–312, 2006.

[176] K. P. Bennett, P. Bennett, and E. Parrado-Hernandez, "The interplay of optimization and machine learning research," *Journal of Machine Learning Research*, vol. 7, pp. 1265–1281, 2006.

[177] J. Ansari, E. Meshkova, W. Masood, A. Muslim, J. Riihijärvi and P. Mähönen, "CONFab - Component-based Optimization of WSN Protocol Stacks using Deployment Feedback," in *Proceedings of the ACM International Symposium on Mobility Management and Wireless Access*, Paphos, Cyprus, 2012.

[178] J. Ansari, X. Zhang, C. Cugola, M. Migliavacca, M. Bennebroek and P. Mähönen, "Poster Abstract: Wirelessly Accessible Sensor Populations (WASP) – Cross-layer Design of Low Power Medium Access and Content based Routing," in *Proceedings of the IEEE Sensor, Mesh and Ad Hoc Communications and Networks*, Boston, Massachusetts, USA, 2010.

[179] T. Melodia, M. C. Vuran and D. Pompili, "The state of the art in cross-layer design for wireless sensor networks," in *Proceedings of EuroNGI Workshops on Wireless and Mobility*, Villa Vigoni, Italy, 2005.

[180] V. Kawadia and P. R. Kumar, "A cautionary perspective on cross layer design," *IEEE Wireless Communications*, vol. 12, no. 1, pp. 3–11, 2005.

[181] J. N. Al-Karaki and A. E. Kamal, "Routing techniques in wireless sensor networks: a survey," *IEEE Wireless Communications*, vol. 11, no. 6, pp. 6 – 28, 2004.

[182] *Protégé*, Stanford University, USA, http://protege.stanford.edu/overview/index.html, Last visited: 11.10.2011.

[183] "Fact++," http://owl.man.ac.uk/factplusplus/ [Last visited: 02.08.2011].

[184] Y. Mao, F. Wang, L. Qiu, S. S. Lam, and J. M. Smith, "S4: small state and small stretch routing protocol for large wireless sensor networks," in *Proceedings of the 4th USENIX Symposium on Networked Systems Design and Implementation*, Cambridge, Massachusetts, USA, 2007.

[185] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss, and P. Levis, "Collection Tree Protocol," in *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems*, Berkeley, California, USA, 2009.

[186] J. Camp and E. Knightly, "Modulation Rate Adaptation in Urban and Vehicular Environments: Cross-layer Implementation and Experimental Evaluation," in *Proceedings of the International Conference on Mobile Computing and Networking*, San Francisco, California, USA, 2008.

[187] P. Levis, N. Patel, D. Culler, and S. Shenker, "Trickle: a self-regulating algorithm for code propagation and maintenance in wireless sensor networks," in *Proceedings of the 1st Conference on Symposium on Networked Systems Design and Implementation*, San Francisco, California, USA, 2004.

[188] *Collection Tree Protocol Linux/Click Implementation*, Stanford University, USA, http://sing.stanford.edu/gnawali/ctp/code/ctpmesh_release.tar.gz, Last visited: 11.10.2011.

[189] G. Cugola, M. S. Vennarucci, A. Corongiu, J.-D. Decotignie, J. Rousselot, L. Thiem, P. Todorova, A. Sorniotti, P. Stanley-Marbell, R. Serna, J. Ansari and X. Zhang, "WASP Project Deliverable Document - D4.4: Initial Definitions of New Protocols, First Version of Trade-off Formalization," European Commission FP-6 Funded WASP Project, Tech. Rep., 2008.

[190] J. Cao, X. Feng, J. Lu, and S. K. Das, "Mailbox-Based Scheme for Designing Mobile Agent Communication Protocols," *Computer*, vol. 35, no. 9, pp. 54–60, 2002.

[191] J.P. Thomesse, "Time and industrial local area networks," in *Proceedings of Computers in Design, Manufacturing, and Production*, Paris-Evry, France, 1993.

[192] P. T. Eugster, P. A. Felber, R. Guerraoui, and A.-M. Kermarrec, "The many faces of publish/subscribe," *ACM Computer Survey*, vol. 35, no. 2, pp. 114–131, 2003.

[193] M. Kohvakka, T. Arpinen, M. Hännikäinen, and T. D. Hämäläinen, "High-performance multi-radio wsn platform," in *Proceedings of the 2nd International Workshop on Multi-hop Ad Hoc Networks: from Theory to Reality*, Florence, Italy, 2006.

[194] N. Ullah, P. Khan and K. S. Kwak, "A Very Low Power MAC (VLPM) Protocol for Wireless Body Area Networks," *IEEE Sensors*, vol. 11, no. 4, pp. 3717–3737, 2011.

[195] E. Meshkova, J. Ansari, J. Riihijärvi, J. Nasreddine, and P. Mähönen, "Estimating transmitter activity patterns: an empirical study in the indoor environment," in *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, Toronto, Canada, 2011.

[196] Z. Wang, J. Ansari, V. Atanasovski, D. Denkovski, T. Farnham, L. Gavrilovska, A. Gefflaut, R. Manfrin, E. Meshkova, J. Nasreddine, K. Rerkrai, M. Sooriyabandara, A. Zanella, "Self-organizing Home Networking based on Cognitive Radio Technologies," in *Proceedings of the IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks*, Aachen, Germany, 2011.

[197] European Union funded PARallel PAradigms and Run-time MAnagement techniques for Many-core Architectures (2PARMA) Project, "http://www.2parma.eu/ [Last visited: 02.08.2011]."

[198] Ultra Highspeed Mobile Information and Communication (UMIC) Research Center, "http://www.umic.rwth-aachen.de/ [Last visited: 02.08.2011]."

Curriculum Vitae

Personal Information

Name	Junaid Ansari
Date of birth	5^{th} April 1980
Place of birth	Karlsruhe, Germany
Family status	Married

Education

Jul 2006–Nov 2011	Doctoral studies RWTH Aachen University	Germany
Oct 2003–Mar 2006	Master of Science in Communications Engineering RWTH Aachen University	Germany
Dec 1999–Mar 2002	Bachelor of Electrical Engineering National University of Sciences and Technology	Pakistan

Professional Work Experience

Jul 2006–Nov 2011	Research Assistant Institute for Networked Systems RWTH Aachen University	Germany
Apr 2006–Jun 2006	Research Engineer Department of Wireless Networks RWTH Aachen University	Germany
Jan 2003–Jul 2003	Design Engineer in Machine Vision Group Center for Advanced Research in Engineering	Pakistan
May 2002–Jan 2003	Design Engineer in Systems Group Communications Enabling Technologies	Pakistan

LIST OF PUBLICATIONS

Book chapters relevant to the dissertation topic

J. Ansari, X. Zhang and P. Mähönen, "Practical Experiences and Design Considerations on Medium Access Control Protocols for Wireless Sensor Networks", in *Handbook of Research on Developments and Trends in Wireless Sensor Networks: From Principle to Practice*, by H. Jin and W. Jiang (editors), IGI Global, 2010.

Journal articles relevant to the dissertation topic

J. Ansari, X. Zhang, P. Mähönen, "Multi-radio Medium Access Control Protocol for Wireless Sensor Networks" in *International Journal of Wireless Sensor Networks*, vol. 8, no. 1, pp. 47-61, 2010.

J. Ansari, D. Pankin, P. Mähönen, "Radio-Triggered Wake-ups with Addressing Capabilities for Extremely Low Power Sensor Network Applications" in *International Journal of Wireless Information Networks*, Vol. 16, Issue 3, pp. 118-130, 2009.

J. Ansari, J. Riihijärvi, P. Mähönen and J. Haapola, "Implementation and Performance Evaluation of nanoMAC: A Low-Power MAC Solution for High Density Wireless Sensor Networks" in *International Journal of Sensor Networks*, vol. 2, No. 5, 2007.

Conference papers relevant to the dissertation topic

J. Ansari, E. Meshkova, W. Masood, A. Muslim, J. Riihijärvi and P. Mähönen, "CONFab - Component-based Optimization of WSN Protocol Stacks using Deployment Feedback", in *Proceedings of the ACM International Symposium on Mobility Management and Wireless Access*, Paphos, Cyprus, October 2012.

J. Ansari, T. Ang and P. Mähönen, "WiSpot - Detecting Wi-Fi Networks Using IEEE 802.15.4 Radios" in *Proceedings of the ACM International Symposium on Mobility Management and Wireless Access*, Miami Beach, Florida, USA, November 2011.

J. Ansari, X. Zhang and P. Mähönen, "A Compiler Assisted Approach for Component Based MAC Design" in *Proceedings of the* 10^{th} *IEEE IFIP Annual Mediterranean Ad Hoc Networking Workshop*, Favignana Island, Sicily, Italy, June 2011.

X. Zhang, **J. Ansari** and P. Mähönen, "TRUMP: Supporting Efficient Realization of Protocols for Cognitive Radio Networks" in *Proceedings of the IEEE Symposium on New Frontiers in Dynamic Spectrum*, Aachen, Germany, May 2011.

J. Ansari, X. Zhang, A. Achtzehn, M. Petrova and P. Mähönen, "A Flexible MAC Development Framework for Cognitive Radio Systems", in *Proceedings of the IEEE Wireless Communications & Networking Conference*, Cancun, Mexico, March 2011.

J. Ansari, T. Ang and P. Mähönen, "WiSpot - Detecting Wi-Fi Networks Using IEEE 802.15.4 Radios", Demonstration paper in *Proceedings of the European Conference on Wireless Sensor Networks*, Bonn, Germany, February 2011.

J. Ansari, X. Zhang, O. Salikeen and P. Mähönen, "Enabling Flexible MAC Protocol Design for Wireless Sensor Networks", in *Proceedings of the IEEE International Conference on Wireless On-demand Network Systems and Services*, Bardoneccia, Italy, January 2011.

J. Ansari and P. Mähönen, "Channel Selection in Spectrum Agile and Cognitive MAC Protocols for Wireless Sensor Networks", in *Proceedings of the ACM International Symposium on Mobility Management and Wireless Access*, Bodrum, Turkey, October 2010.

J. Ansari, X. Zhang and P. Mähönen, "A Decentralized MAC for Opportunistic Spectrum Access in Cognitive Wireless Networks", in *Proceedings of the Second ACM Workshop on Cognitive Wireless Networking*, Illinois, Chicago, USA, September 2010.

J. Ansari, T. Ang and P. Mähönen, "Spectrum Agile Medium Access Control Protocol for Wireless Sensor Networks", in *Proceedings of the IEEE Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, Boston, MA, USA, June 2010.

J. Ansari, X. Zhang, C. Cugola, M. Migliavacca, M. Bennebroek and P. Mähönen, "Poster Abstract: Wirelessly Accessible Sensor Populations (WASP) – Cross-layer Design of Low Power Medium Access and Content based Routing", in *Proceedings of the IEEE Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, Boston, MA, USA, June 2010.

J. Ansari, X. Zhang, A. Achtzehn, M. Petrova and P. Mähönen, "Decomposable MAC Framework for Highly Flexible and Adaptable MAC Realizations", Demonstration paper in *Proceedings of the IEEE Symposium on New Frontiers in Dynamic Spectrum*, Singapore, April 2010.

J. Ansari, X. Zhang and P. Mähönen, "Demo Abstract: A Decentralized MAC Protocol for Opportunistic Spectrum Access in Cognitive Wireless Networks", Demonstrated at the *IEEE Conf. on Computer Comm.*, San Diego, CA, USA, March 2010.

X. Zhang, **J. Ansari** and P. Mähönen, "Traffic Aware Medium Access Control Protocol for Wireless Sensor Networks", in *Proceedings of the ACM International Symposium on Mobility Management and Wireless Access*, Canary Islands, Spain, October 2009.

J. Ansari, D. Pankin and P. Mähönen, "Radio-Triggered Wake-ups with Addressing Capabilities for Extremely Low Power Sensor Network Applications", in *Proceedings of the IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, Cannes, France, September 2008.

J. Ansari, X. Zhang and P. Mähönen, "Multi-radio Medium Access Control Protocol for Wireless Sensor Networks", in *Proceedings of the Workshop on Energy in Wireless Sensor Networks*, Santorini, Greece, June 2008.

J. Ansari, X. Zhang and P. Mähönen, "Demo Abstract: Multi-radio Medium Access Control Protocol for Wireless Sensor Networks", in *Proceedings of the ACM Conference on Sensor and Systems*, Sydney, Australia, November 2007.

J. Ansari, J. Riihijärvi, P. Mähönen and J. Haapola, "Implementation and Performance Evaluation of nanoMAC: A Low-Power MAC Solution for High Density Wireless Sensor Networks", in *Proceedings of the IEEE International Conference on Communications*, Istanbul, Turkey, June 2006.

Book chapters in addition to the ones mentioned in this dissertation

C. Silvano, W. Fornaciari, S. C. Reghizzi, G. Agosta, G. Palermo, V. Zaccaria, P. Bellasi, F. Castro, S. Corbetta, A. D. Biagio, E. Speziale, M. Tartara, D. Siorpaes, H. Hubert, B. Stabernack, J. Brandenburg, M. Palkovic, P. Raghavan, C. Ykman-Couvreur, A. Bartzas, S. Xydis, D. Soudris, T. Kempf, G. Ascheid, R. Leupers, H. Meyr, **J. Ansari**, P. Mähönen, and B. Vanthournout, "2PARMA: Parallel Paradigms and Run-time Management Techniques for Many-Core Architectures", in *Designing Very Large Scale Integration Systems: Emerging Trends & Challenges*, by N. Voros, A. Mukherjee, N. Sklavos, K. Masselos, M. Huebner (editors), Springer, 2011.

J. Ansari, J. Riihijärvi and P. Mähönen, "Experiences in data processing and Bayesian filtering applied to localization and tracking in Wireless Sensor Networks", in *Localization Algorithms and Strategies for Wireless Sensor Networks*, by G. Mao and B. Fidan (editors), IGI Global, 2009.

Journal articles in addition to the ones mentioned in this dissertation

S. Corroy, J. Beiten, **J. Ansari**, H. Baldus and P. Mähönen, "Selection of Computing Elements for Energy Efficiency in Wireless Sensor Networks using a Statistical Estimation Method" in *Int. J. on Advances in Networks and Services*, Vol. 2. No. 3, 2009.

Conference papers in addition to the ones mentioned in this dissertation

E. Meshkova, **J. Ansari**, J. Riihijärvi, J. Nasreddine, and P. Mähönen, "Estimating transmitter activity patterns: an empirical study in the indoor environment", in *Proceedings of the IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, Toronto, Canada, September 2011.

C. Silvano, W. Fornaciari, S. C. Reghizzi, G. Agosta, G. Palermo, V. Zaccaria, P. Bellasi, F. Castro, S. Corbetta, A. D. Biagio, E. Speziale, M. Tartara, D. Siorpaes, H. Hubert, B. Stabernack, J. Brandenburg, M. Palkovic, P. Raghavan, C. Ykman-Couvreur, A. Bartzas, S. Xydis, D. Soudris, T. Kempf, G. Ascheid, R. Leupers, H. Meyr, **J. Ansari**, P. Mähönen, and B. Vanthournout, "Parallel Paradigms and Run-time Management Techniques for Many-core Architectures: The 2PARMA Approach", in *Proceedings of the IEEE International Conference on Industrial Informatics*, Caparica, Lisbon, Portugal, July 2011.

C. Silvano, W. Fornaciari, S. C. Reghizzi, G. Agosta, G. Palermo, V. Zaccaria, P. Bellasi, F. Castro, S. Corbetta, A. D. Biagio, E. Speziale, M. Tartara, D. Siorpaes, H. Hubert, B. Stabernack, J. Brandenburg, M. Palkovic, P. Raghavan, C. Ykman-Couvreur, A. Bartzas, S. Xydis, D. Soudris, T. Kempf, G. Ascheid, R. Leupers, H. Meyr, **J. Ansari**, P. Mähönen, and B. Vanthournout, "Parallel programming and Run-time Resource Management Framework for Many-core Platforms: The 2PARMA Approach", in *Proceedings of the Int. Workshop on Reconfigurable Communication-centric Systems-on-Chip*, Montpellier, France, June 2011.

Z. Wang, **J. Ansari**, V. Atanasovski, D. Denkovski, T. Farnham, L. Gavrilovska, A. Gefflaut, R. Manfrin, E. Meshkova, J. Nasreddine, K. Rerkrai, M. Sooriyabandara and A. Zanella, "Self-organizing Home Networking based on Cognitive Radio Technologies", Demonstration paper in *Proceedings of the IEEE Symposium on New Frontiers in Dynamic Spectrum*, Aachen, Germany, May 2011.

E. Meshkova, **J. Ansari**, D. Denkovski, J. Riihijärvi, J. Nasreddine, M. Pavloski, L. Gavrilovska, and P. Mähönen, "Experimental Spectrum Sensor Testbed for Constructing Indoor Radio Environmental Maps", Poster paper in *Proceedings of the IEEE Symposium on New Frontiers in Dynamic Spectrum*, Aachen, Germany, May 2011.

A. Kovacevic, **J. Ansari** and P. Mähönen, "NanoSD: A Flexible Service Discovery Protocol for Wireless Sensor Networks", in *Proceedings of the IEEE Mobile Ad-hoc and Sensor Networks conference*, Hangzhou, China, December 2010.

O. Salikeen, **J. Ansari**, X. Zhang and P. Mähönen, "Demo Abstract: Enabling Flexible MAC Protocol Design for Wireless Sensor Networks", in *Proceedings of the ACM Conference on Sensor and Systems*, Zurich, Switzerland, November 2010.

C. Silvano, W. Fornaciari, S. C. Reghizzi, G. Agosta, G. Palermo, V. Zaccaria, P. Bellasi, F. Castro, S. Corbetta, A. D. Biagio, E. Speziale, M. Tartara, D. Siorpaes, H. Hubert, B. Stabernack, J. Brandenburg, M. Palkovic, P. Raghavan, C. Ykman-Couvreur, A. Bartzas, S. Xydis, D. Soudris, T. Kempf, G. Ascheid, R. Leupers, H. Meyr, **J. Ansari**, P. Mähönen, and B. Vanthournout, "2PARMA: Parallel Paradigms and Run-time Management Techniques for Many-Core Architectures", in *Proceedings of the International Conference on VLSI Design*, Banglore, India, January 2010.

M. Bennebroek, **J. Ansari**, X. Zhang, C. Cugola, M. Migliavacca and P. Mähönen, "Poster Abstract: Wirelessly Accessible Sensor Populations (WASP) – Deployment and enterprise integration of energy efficient wireless sensor networks", in *Proceedings of the IEEE Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, Rome, Italy, June 2009.

A. Achtzehn, E. Meshkova, **J. Ansari** and P. Mähönen, "MoteMaster:A scalable sensor network testbed for rapid protocol performance evaluation", in *Proceedings of the IEEE Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, Rome, Italy, June 2009.

A. Kovacevic, **J. Ansari** and P. Mähönen, "Demo Abstract: Discovering Services in Mobile, Flexible and Heterogeneous WSN", in *Proceedings of the Conference on Information Processing in Sensor Networks*, San Francisco, CA, USA, April 2009.

E. Meshkova, J. Riihijärvi, **J. Ansari**, K. Rerkrai and P. Mähönen, "An Extendible Meta-Data Specification for Component-Oriented Networks with Applications to WSN Configuration and Optimization", in *Proc. of the IEEE Int. Symposium on Personal, Indoor and Mobile Radio Communications*, Cannes, France, Sept. 2008.

F. Oldewurtel, **J. Ansari** and P. Mähönen, "Cross-Layer Design for Distributed Source Coding in Wireless Sensor Networks", in *Proceedings of the International Conference on Sensor Technologies and Applications*, Cap Esterel, France, August 2008.

S. Corroy, J. Beiten, **J. Ansari**, H. Baldus and P. Mähönen, "Energy Efficient Selection of Computing Elements in Wireless Sensor Networks", in *Proceedings of the International Conference on Sensor Technologies and Applications*, Cap Esterel, France, August 2008.

M. Rondinone, **J. Ansari**, J. Riihijärvi and P. Mähönen, "Designing a Reliable and Stable Link Quality Metric for Wireless Sensor Networks", in *Proceedings of the Workshop on Real-World Wireless Sensor Networks*, Glasgow, U.K., April 2008.

M. Popa, **J. Ansari**, J. Riihijärvi and P. Mähönen, "Combining Cricket System and Inertial Navigation for Indoor Human Tracking", in *Proceedings of the IEEE Wireless Communications & Networking Conference*, Las Vegas, NV, USA, March-April 2008.

F. Siegemund, M. Haroon, **J. Ansari** and P. Mähönen, “Senslets - Applets for the Sensor Internet”, in *Proceedings of the IEEE Wireless Communications & Networking Conference*, Las Vegas, NV, USA, March-April 2008.

P. Kuckertz, **J. Ansari**, J. Riihijärvi and P. Mähönen, “Sniper Fire Localization using Wireless Sensor Networks and Genetic Algorithm based Data Fusion”, in *Proceedings of the IEEE International Conference for Military Communications*, Orlando, Florida, USA, October 2007.

J. Ansari, J. Riihijärvi and P. Mähönen, “Combining Particle Filtering with Cricket System for Indoor Localization and Tracking Services”, in *Proceedings of the IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, Athens, Greece, September 2007.

J. Ansari, J. Sanchez, M. Petrova, J. Riihijärvi, O. Raivio, K. Rerkrai, C. Jardak, F. Oldewurtel, M. Wellens, L. Wu and P. Mähönen, “Flexible HardwareSoftware Platform for Tracking Applications”, in *Proceedings of the International Vehicular Technology Conference*. Dublin, Ireland, April 2007.

J. Ansari, J. Sanchez, M. Petrova, J. Riihijärvi, O. Raivio, K. Rerkrai, C. Jardak, F. Oldewurtel, M. Wellens, L. Wu and P. Mähönen, “Demo Abstract: Flexible HardwareSoftware Platform for Tracking Applications”, in *Proceedings of the ACM Conference on Sensor and Systems*, Boulder, CO, USA, November 2006.