Designing of MAC layer for Mission-Critical Surveillance Applications in Wireless Image Sensor Networks

Ehsan Muhammad

THÈSE

pour obtenir le grade de

Docteur de

L'UNIVERSITÉ DE PAU ET DES PAYS DE L'ADOUR

ÉCOLE DOCTORALE DES SCIENCES EXACTES ET LEURS APPLICATIONS

Domaine de recherche : INFORMATIQUE

Présentée par

Muhammad Ehsan

Couche MAC adaptative pour les applications critiques de surveillance à base d'un réseau de capteurs d'image

Soutenue le 09 Juin 2015

JURY

Rapporteurs :	M. Zoubir MAMMERI	-	Pr. Université de Toulouse
	M. Ye-Qiong SONG	-	Pr. Université de Lorraine
Examinateurs :	M. Philippe ANIORTÉ	-	Pr. Université de Pau et des Pays de l'Adour
	M. Bernard POTTIER	-	Pr. Université de Bretagne Occidentale
Directeur de thèse :	M. Congduc PHAM	-	Pr. Université de Pau et des Pays de l'Adour

Abstract

Wireless Sensor Networks (WSNs) are designed for the purpose of completing different monitoring tasks under various environmental conditions. The small electronic devices called sensors are capable of sensing, processing and communicating the environmental data through multi-hop communication and coordination. These devices have limited resources (memory, computing capabilities) and usually run on batteries. This is the reason the research on wireless sensor networks have been focussed on energy efficiency and self-organisation of the network.

We consider mission-critical surveillance applications in our research work. These applications can have different requirements than traditional WSNs. In addition, we use image sensor nodes, whose activity is defined based on criticality of the application. The criticality-based scheduling scheme defines sentry nodes with faster capture rates, to have higher probability to detect intrusions and to alert neighbour nodes.

At Medium Access Control (MAC) Layer level, duty cycled approaches are used to preserve energy and prolong the network lifetime. However, while conserving energy, mission-critical surveillance applications cannot compromise on quality of surveillance and the network should still be able to quickly propagate the alert messages. In this thesis, we propose a low latency, energy efficient adaptive MAC protocol.

We first propose an original approach to dynamically determine the duty-cycle length of sensor nodes to increase the probability of quick propagation of alerts. Simulation results confirmed that our approach succeeds in improving the system responsiveness when compared to a static duty-cycling approach. At the same time, our proposition considerably reduces the energy consumption of the network.

Then, we implemented our approach on sensor node hardware and results were found to be very close to our simulation results.

Keywords : Image sensors, Mission-critical, duty-cycled MAC, Criticality.

Résumé

Les réseaux de capteurs sans fil sont conçus dans le but de remplir différentes tâches de surveillance dans des conditions environnementales variées. Ces petits appareils électroniques sont capables de détecter, traiter et transmettre des données environnementales avec des communications multi-sauts et peuvent par conséquent aussi se coordonner. En même temps, ces dispositifs ont des ressources limitées (mémoire, capacités de calcul) et doivent fonctionner le plus souvent sur des batteries. C'est pour ces raisons que les recherches menées dans le domaine des réseaux de capteurs possèdent naturellement une forte partie qui concerne la réduction de la consommation d'énergie et une auto-organisation du réseau.

Nos recherches considèrent les applications critiques de surveillance. Ces applications peuvent avoir des exigences très différentes des réseaux de capteurs traditionnels. De plus, nous utilisons des capteurs images, dont l'activité est définie en fonction de la criticité de l'application. Un ordonnancement basé sur la criticité permet de définir des noeuds sentinelles qui possèderont une vitesse de capture plus grande, cela afin d'avoir une probabilité plus élevée de détecter des intrusions et d'alerter leurs noeuds voisins.

Au niveau de la couche de contrôle d'accès au medium (couche MAC), les approches alternant activité-sommeil (consistant à allumer et éteindre la radio de manière cyclique) sont utilisées pour préserver l'énergie et prolonger la durée de vie du réseau. Cependant, tout en conservant l'énergie, les applications critiques de surveillance ne doivent pas compromettre la qualité de la surveillance et le réseau doit être toujours en mesure de propager rapidement les messages d'alerte. Notre but est de définir un protocole MAC qui pourrait réduire la latence de propagation d'alerte ainsi que de prolonger la durée de vie du réseau.

Nous proposons tout d'abord une approche originale pour déterminer dynamiquement la durée de la période d'activité radio des noeuds pour augmenter la proba-

bilité de propager rapidement les alertes. Les résultats des simulations ont confirmé que notre approche réussit à améliorer la réactivité du système par rapport à une approche statique. En même temps, notre proposition permet de réduire considérablement la consommation d'énergie du réseau. Ensuite, nous avons implémenter notre approche sur des capteurs réels et les résultats obtenus sont très proches des résultats de simulation.

Mots-clés : Capteurs images, applications critiques de surveillance, Couche MAC, CAMP.

To my dear family who have been with me through thick and thin.

Acknowledgments

First and foremost, I would like to express my profound gratitude to my supervisor Professor Congduc PHAM. I can't find the words expressive enough to convey the deepest gratitude and appreciation I owe to my Supervisor, for his steadfast guidance, inspiration, constructive suggestions, generosity, availability, support and encouragement all along the duration of my thesis.

I will take this opportunity to thank Professor Ye-Qiong SONG and Professor Zoubir MAMMERI for honouring me by reviewing my research work. I would also like to thank the respected jury members, Professor Bernard POTTIER and Professor Philippe ANIORTÉ.

I would like to thank all the members of LIUPPA (Laboratoire Informatique de l'Université de Pau et des Pays de l'Adour), my team T2I and Informatics department of faculty of Sciences and Technology of UPPA for their warm welcome, their moral and technical support and their good humour during these years of my thesis work.

I am highly indebted to my entire family for their continuous support, care and understanding. They have been with me through all the tough times.

I thank my extraordinary friends in France and in Pakistan for their cherishing moments, their support and their valuable suggestions. A profound thanks to all the people, who directly or indirectly helped me in realising this modest work.

Finally, I would like to appreciate and acknowledge the financial support of HEC (Higher Education Commission) of Pakistan to carry out this research.

Contents

Chapter 1

Introduction

Contents

Wireless Sensor Networks have emerged as an application domain of wireless ad hoc networks. These networks consist of numerous sensor nodes spread over an area to perform a specific function. These networks have huge number of applications in daily life. The wireless node sensors have limited capacity, and are the topic of interest for a lot of researchers to improve their capacity and limited lifetime. In this chapter we give a brief introduction to Wireless Sensor Networks, their applications and the research area we have been working on.

1.1 Wireless Sensor Networks

The communication technology has evolved from the wired communications to wireless communications. Wireless sensor networks (WSN) are a part of this evolution, having many applications covering all the aspects of human life. A Wireless Sensor Network (WSN) designates a system composed of numerous sensor nodes distributed over an area in order to collect information [1][2][3]. These networks are designed for the purpose of completing different monitoring tasks under various environmental conditions at low cost.

WSNs have some constraints and challenges to deal with, which distinguish them considerably from other wireless networks [4]. At the same time, WSNs are designed for specific applications, i.e., each WSN may have different objectives and requirements to meet so they are not required to meet all the constraints at one instance.

These sensor nodes are deployed in large numbers. Once deployed, the large number of nodes make individual configuration of each and every node impractical. Similarly, in case of change in the network topology (due to loss of radio links, node destruction or battery depletion) nodes should be able to reorganise themselves independently. These sensor nodes work on a battery, so they need to be low powered. Working on a limited battery is the reason why reducing the energy consumption to prolong the network lifetime and the self-organisation of the network are the two most studied topics in WSNs. Furthermore, these sensor nodes need to be scalable and adaptable to support different network sizes under different applications and conditions [5].

1.1.1 Architecture

A typical WSN usually contains a large number of wireless sensor nodes. The wireless sensors are the spatially distributed, autonomous and embedded devices, that work together to perform common tasks. These nodes are capable of sensing, processing and communicating the environmental data through multi-hop communication and coordination. Thanks to miniaturisation of the electronic devices, these nodes are expected to be of a small size and inexpensive.

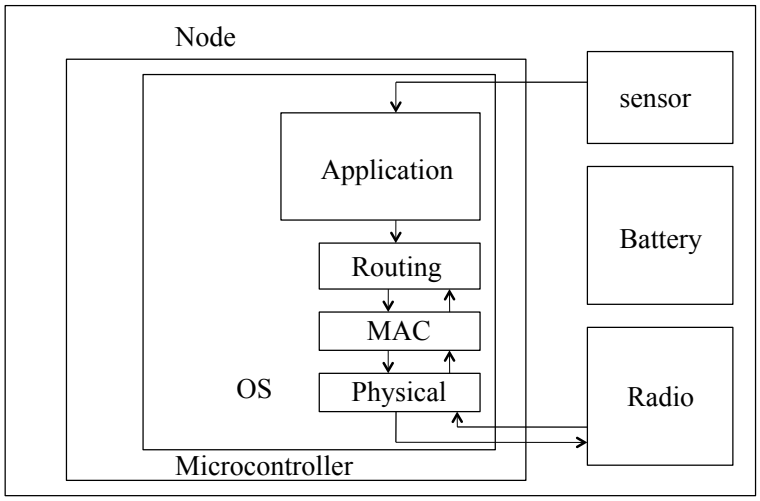

Figure 1.1: Illustration of general architecture of a sensor node in WSN

1.1.1.1 Node Architecture

A typical sensor node consists of four basic components, which are: a micro-controller unit, a sensor, a radio, and a battery. Figure 1.1 depicts the basic architecture of a sensor node. The micro-controller unit basically provides the intelligence to the node by performing tasks, processing the information/data and controlling the functionality of other components. The sensor node usually comes with self-efficient micro-controller. The node contains one or more sensors, these application specific sensors have functionalities like detecting a motion (motion based sensors), temperature or humidity sensors etc.

The radio on a sensor node enables the nodes to communicate to other nodes and the sink over the radio channel. The radio can operate in different operating modes such as transmit, receive, idle and sleep mode. The radio is a primary source of energy consumption in wireless sensor networks, hence it is of utmost importance for energy efficient operation of the sensor node. It helps in deciding several factors such as power consumption, carrier frequency, data rate, modulation, coding schemes, transmission power, error blocking and many more.

As these nodes are wireless, they are equipped with a battery to operate. This battery is often the only energy source available to the nodes. Due to its limited capacity, it is the reason for huge energy constraints of WSN.

1.1.1.2 Network Architecture

The sensor nodes work as a team in WSN. The scattered sensor nodes in the sensor field collaborate with each other to perform a common task. They need this collaboration to communicate to the sink to perform their responsibilities. To send data to the sink node, the nodes use multi-hop paths. The sensor field can have more than one sink nodes depending on the terrain, size and traffic load of the field. Depending on the network requirements and how the sensor nodes communicate to each other, the nodes can organise in two different ways. All the nodes having same sensing capabilities, each node can work as a peer. In this organisation, the nodes relay the data to the sink node through other peers using multiple path routes in a distributed fashion. This is what is known as flat architecture [5].

The second type of architecture is named as hierarchical architecture[5]. In hierarchical networks, the nodes are organised into clusters, which in turn are supervised by the cluster heads. The cluster members send their data to the cluster head, which then relays it to the sink. The cluster heads can have different capabilities than other nodes. Both type of network architectures explained above have their own pros and cons.

1.1.2 Applications

The application domain of WSNs is very diverse and broad, which includes, but is not limited to environmental, health, home automation, industrial, smart space, and security related areas [6] [7] [8]. These networks can also be deployed in large areas, unreachable and hostile areas where human presence becomes a risky affair such as earthquake or flood hit areas, battle fields or contaminated regions[3]. WSNs can also help to avoid catastrophic infrastructure disasters.

The WSNs can also be deployed to monitor the habitat of endangered species, by distributing numerous number of sensors over monitoring area. Habitat monitoring is very helpful for keeping certain animal species alive. One of the most typical

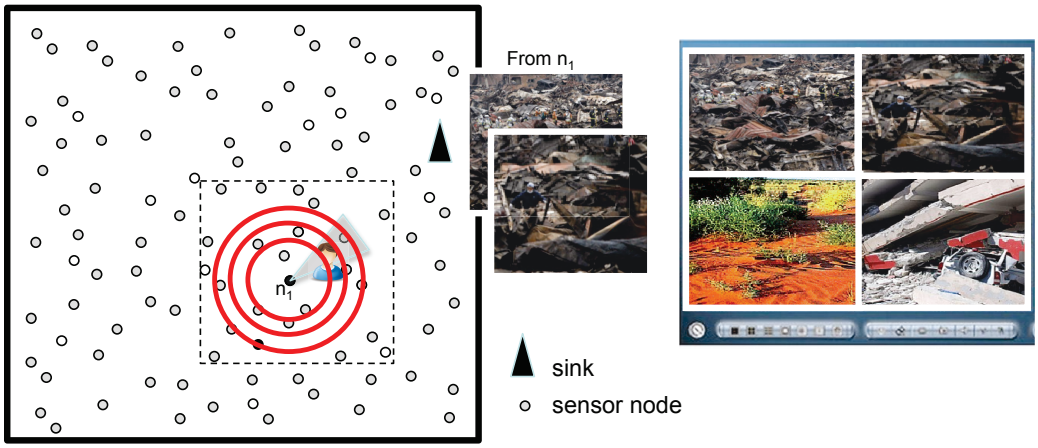

Figure 1.2: Mission critical intrusion detection system

examples is ZebraNet project of Princeton university [9] for tracking the interaction and movement of Zebras in Kenya.

WSNs can also facilitate many military systems used for monitoring purposes, such as surveillance systems and intrusion detection systems [10]. WSNs can be deployed in battlefield for situational awareness, which is important in a battlefield. The sensors are scattered in the battlefield by planes or humanly, which form a network and send information to the control centers[11].

WSNs can help to conserve precious natural resources, increase productivity, monitor bridges, enable new smart home technologies, and in many other countless applications.

In our team, we are working on critical applications such as surveillance, disaster relief operations (e.g., earthquakes, floods, etc) and intrusion detection systems, etc. Figure 1.2 shows the scenario of a random deployment of image sensor nodes which is typical of the kind of applications we want to address in this thesis. These nodes are equipped with a camera. The intrusion detection system depicted in figure 1.2 shows that once an intrusion is detected, the sensor node takes a number of images, it will send these images to the sink (or a control room) and alert it's neighbour nodes. The images are then analysed and proper action is taken.

With this brief introduction to WSNs, we will now move to MAC layer to discuss the importance and challenges related to the MAC layer designing of WSNs.

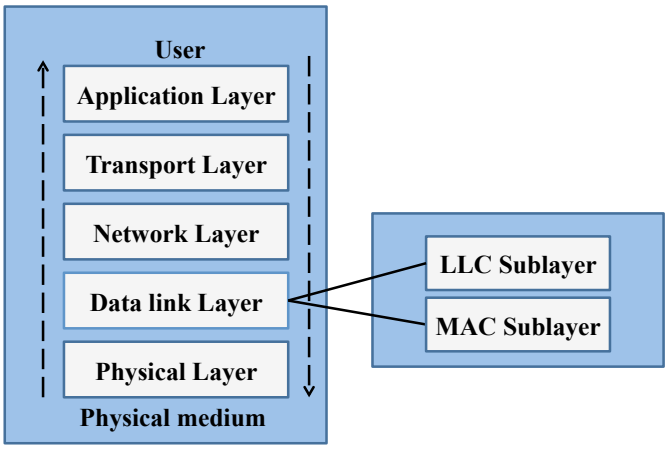

Figure 1.3: The communication protocol stack

1.2 Mac layer basics for Wireless Sensor Networks

The MAC sublayer is a part of the data link layer specified in the communication protocol stack and is shown in figure 1.3 [3]. It provides the channel access mechanism to medium sharing devices. The limited channel resources of WSNs means that the nodes within a certain radio range have to share the same medium. In a shared medium among multiple devices, a single broadcast is received by every other device in the transmission range. This can result in interference and collision of frames, when multiple devices attempt to transmit, simultaneously. Sensor nodes usually communicate via multi-hop paths, hence the communication can be quite difficult.

MAC protocols are designed to manage the communication on a shared medium and is responsible for the successful operation of the network. The MAC protocols provide the basic network infrastructure for sensor nodes to successfully communicate with each other in a collision free manner.

Moreover, as the sensors operate on a battery (which results in limited lifetime of the network), the protocols are generally designed to maximise the network lifetime. This is mostly achievable on MAC layer level. The MAC layer controls the radio,

which consumes most of the energy of the sensor. Thus MAC layer protocols for WSNs are generally designed to use the radio efficiently and consuming less energy. This mainly is the reason of MAC protocols being the focal point of research in WSNs.

In general, the fundamental task of any MAC protocol is to regulate the fair access of sensor nodes to the shared medium in order to achieve good individual throughput and better channel utilisation [12]. However, in WSNs, the sensor nodes work together to perform a common task. Hence, fairness is not an important issue in WSNs. As discussed earlier, the battery usually being the only energy source available, the lifetime of the network will mainly rely on energy efficiency. Due to special characteristics of WSNs, conventional MAC protocols may not be directly applicable to them. Conventional MAC protocols usually focus on how to provide better quality of service (QoS), achieve higher bandwidth efficiency and deal with user mobility. However in WSNs, rather than QoS, energy efficiency is considered to be the priority as network lifetime depends on it.

The nodes remains idle for longer period of times and generally the communication is unidirectional (the sensor nodes communicate to the sink node) and the end-result is obtained by processing the collective information. But at the same time, some extra problems emerge. Like the problem of energy conservation and scalability, among others. So the ideal MAC protocol have to be adaptable, scalable, efficient in communication and should utilise the energy efficiently.

The large amount of energy waste happens at radio level [13, 14, 15, 16], i.e for communication purposes. The four major causes of energy waste at radio [17] are

- *Idle listening:* When a node keeps listening to the medium while waiting for some communication to take place, the phenomenon is called as Idle listening, as the node actually is doing nothing but wait to receive a message from some neighbour as shown in figure 1.4. This is a major source of energy wastage [18, 19, 20, 21, 22, 23] , as nodes anticipate possible communication to avoid missing important messages and continue listening the radio. Radio reception drains lots of energy, even when no data is being transmitted. And in WSNs, there might be no event for long period of time, that means a minimum of communication will take place.
- *Collisions:* Packet collisions is huge source of energy wastage, especially in wireless communication as it starts some kind of chain reaction. The collision occurs when two or more packets are transmitted at the same time and are

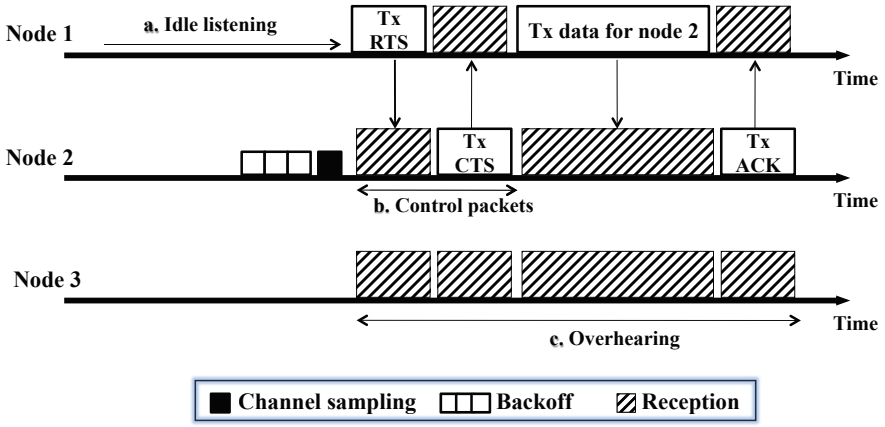

Figure 1.4: Major causes of energy waste

overlapped, these overlapped frames then must be discarded by the receiver as they are corrupted. As a result, the retransmissions of the colliding packets are required, which results in additional traffic which reduces the channel availability. This unavailability of the channel can result in even more collisions, hence further more energy is consumed.

– *Overhearing:* Overhearing occurs when a node hears a communication taking place that is not addressed to it, but is happening in its radio range as shown in figure 1.4. The wireless medium is broadcast by nature, all the nodes located in the radio range of the transmitter receive the frame, which is discarded later. Such reception uselessly drains energy. In dense networks and under heavy traffic situations, overhearing can particularly be a huge problem.
– *Control overhead:* Control Overhead is the energy consumed by exchanging control packets, shown in figure 1.4. The control packets are necessary to avoid collision and sharing of medium in WSNs. However, the control overhead is considered as a major source of energy wastage because their size is similar to the one of typical sensor data packets. The transmission and reception of these packets consumes a lot of energy. Moreover, the control packets do not directly

convey useful information related to application data, hence they also reduce the effective throughput.

In light of these characteristics of WSNs, it can be safely said that MAC layer protocols for WSNs need specific design to avoid these energy wastages. Well designed MAC protocol should keep these energy waste mechanisms to a low level, while achieving good throughput and delay performance. Duty cycling (spliting the node radio cycle in sleep mode and active mode) [20] is considered a powerful method to solve most of energy related issues in WSNs. It is implemented in most of the proposals in the literature [24, 25] and in the 802.15.4 standard [26] of IEEE. Duty cycling revolutionised the WSNs lifetime but it brings the downsides with it like latency and low throughput. Latency and low throughput can play a vital role in certain applications.

Latency becomes a very important aspect in non-delay tolerant applications. We are working on mission critical surveillance applications, and latency is of utmost importance in such applications. Critical applications can have catastrophic consequences if they are unable to communicate information in a certain time limit. In case of detection of an intrusion, a surveillance network should inform the sink about the intrusion as early as possible instead of conservation of energy. An application of critical nature such as intrusion detection system will not serve its purpose, if it cannot inform the control centre or the sink about the latest happenings. These applications are required to inform the sink about an intrusion at the earliest so that a proper action can be taken. The energy conservation remains an important constraint but it becomes the second priority in this case, as saving energy and not informing the control center about the intrusion with kill the main task of the application. Furthermore, the network should be sufficiently alert to capture any further movements of the intruder in the zone under surveillance. For this purpose, the intrusion capturing nodes needs to alert the network urgently, this is shown in figure 1.5, which is a close-up view of the dashed square area in figure 1.2 . The figure illustrates the alert process in which neighbour nodes are put in alert mode (red nodes). The sensor nodes on detecting an intrusion send an alert, and the sequence of messages is relayed to the sink (in case of figure 1.5, the nodes send the images to the sink, as we are using the image sensor nodes).

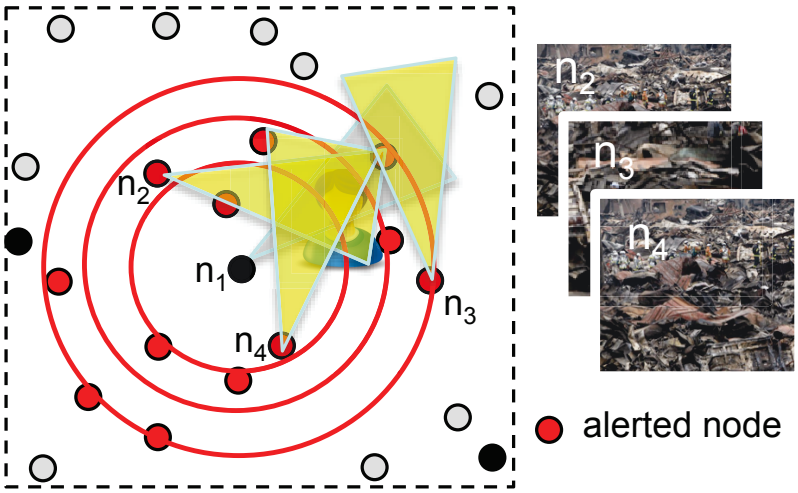

Figure 1.5: Alert propagation

MAC layer for non-delay tolerant applications will hence need to keep the latency factor into consideration and an appropriate balance between energy conservation and latency needs to be found.

1.3 Thesis contribution

In our research work, we use image capturing nodes i.e., the nodes are equipped with a unidirectional camera as a sensor as one shown in figure 3.8 (which was developed in our team's research work). Due to these unidirectional cameras, we need to take care of certain new dynamics in addition to the traditional requirements of WSNs.

We also developed multiple camera systems as the one shown in figure 1.7. Different lenses can be mounted on these cameras.

Our research considers surveillance applications where image sensor nodes are thrown in mass randomly, to start the surveillance process, e.g. intrusion/anomaly detection, situation awareness, environmental disaster monitoring. As these nodes have a unidirectional camera attached, they have a specific Angle of view (AOV), which means they cover specific part of an area in a single direction.

When large number of nodes are thrown randomly in a specific area, it becomes highly probable that there will be redundant nodes. Which means that there will be

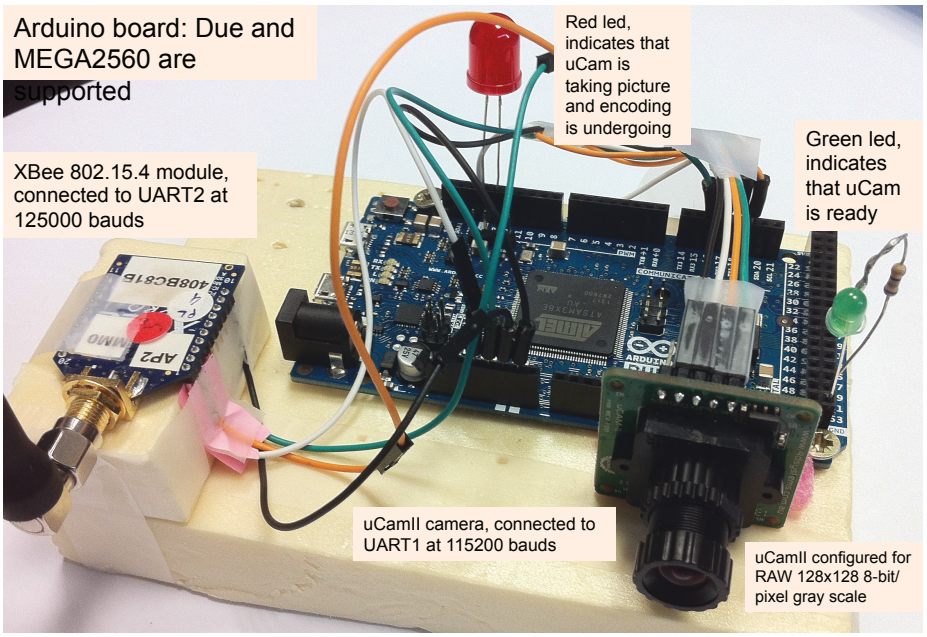

Figure 1.6: An image of a node equipped with a camera

nodes or combinations of nodes, which will be covering the same area. We can use this redundancy to our advantage, by using these redundant nodes to improve the quality of the network surveillance and to overcome the energy constraints of WSN. Once an intrusion is detected, an alert is propagated through the network and to the sink (or control room) for arrangements to be made accordingly.

For alert propagation to take place, the image capturing node needs to be able to communicate to its neighbours successfully and urgently, through its radio. To deal with this communication and to control the radio we go to MAC layer, which is responsible for the successful operation of the network. The work presented in this thesis is focused on the MAC layer. Our goal is to address the MAC layer for providing low energy consumption and low latency for alert propagation in critical applications.

In Figure 1.5, it is desirable that neighbour nodes can receive the alert indication as soon as possible in order to propagate the alert towards the sink. However, event detection in such wireless sensor networks can be quite sporadic and nodes can stay idle for a long period of time. Hence, as explained earlier, MAC layer is usually designed to adopt a duty-cycled behaviour in order to save the energy consumption. Instead of maintaining the radio module awake listening for incoming packets: an active or listening period alternates with an inactive or sleep period. Many protocols for

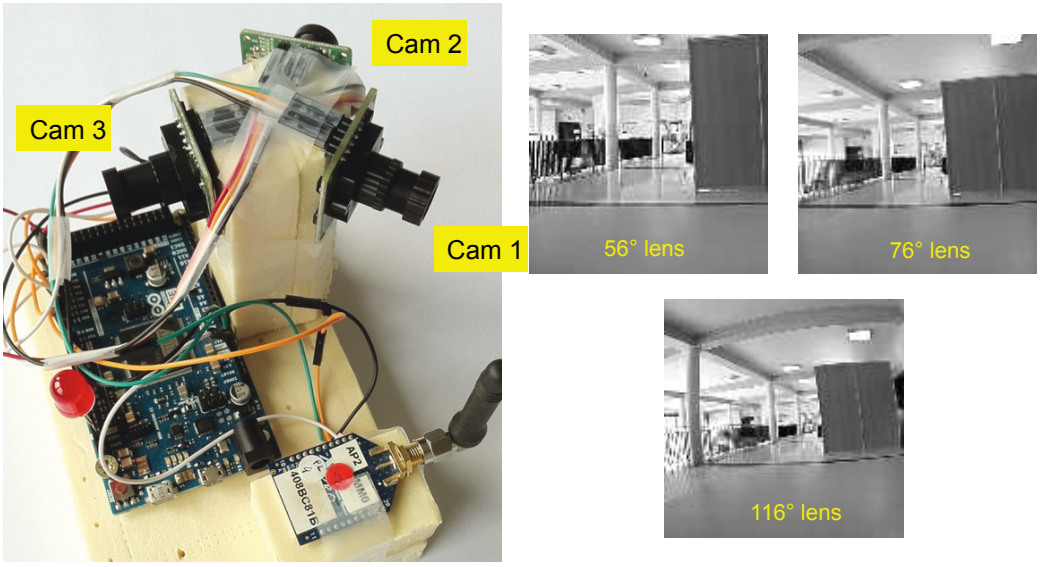

Figure 1.7: Multiple camera system developed in our team's work

MAC layer (like [22, 20, 27, 28, 29]) have been proposed in WSNs by the researchers and significant improvements have been achieved on energy efficiency and channel performance. However, these improvements have been introduced at the expense of latency.

Working on applications of critical nature, our motivation here is to minimise the latency for alert propagation. We propose to adapt the active period of radio cycle of sensor nodes to provide low-latency alert communication in the context of image sensors. At the same time it is desired, not to compromise on the node's lifetime and as a result the network's lifetime.

1.4 Thesis Organisation

The remainder of thesis is organised as follows.

- *Chapter 2:* Chapter 2 elaborates on the related work in the area of designing efficient MAC protocols for wireless sensor networks, with emphasis on low latency real time MAC protocols.
- *Chapter 3:* In chapter 3, we explained the criticality model (developed in our team [30]) for image sensors, which is base of our research in this dissertation.

- *Chapter 4:* Chapter 4 presents our approach to design a new energy efficient, low latency MAC protocol, which successfully achieves its design goals. We then present the simulation results in Chapter 4.
- *Chapter 5:* In chapter 5, we explained the implementation of our approach on sensor nodes, showing results in practical environment.
- *Chapter 6:* Finally, the conclusions and discussions about the future research perspectives are presented in chapter 6.

Chapter 2

State of the Art

Contents

WSNs have a huge constraint of Energy and research has shifted towards optimising medium access. The medium access control layer controls the radio chip, hence the energy consumption of the node and as a result, lifetime of the network. Generally, energy efficiency is a primary concern for WSNs and duty-cycling is the most energy efficient solution for WSNs. In designing a MAC layer for specific applications, care should be taken, as different applications have different requirements. Critical applications might not tolerate latency in comparison to the energy consumption of the network. This chapter presents a brief overview of the state of the art in WSN MAC layer protocols addressing energy consumption and latency.

2.1 Medium Access in WSNs

Designing Medium Access Control (MAC) protocol is a very active research area in recent years, and it attracts the interests of many researchers. MAC layer has special importance in WSN's because these networks are battery hungry. This is the reason lots of research on MAC layer have been focused on energy efficiency of the network. Certain applications in WSNs like intrusion detection systems require a quick response from the network. For these kind of applications, latency is the most important aspect in a network and energy at times becomes the 2nd priority. Huge number of MAC protocols has been proposed in literature.

We present below a brief survey about several MAC protocols proposed for WSNs. We first present the IEEE standard 802.15.4 protocol briefly. We then classified the protocols into synchronous and asynchronous protocols. As our work is focused on critical applications which require quick response, low latency protocols are presented in the section "real time protocols".

2.2 IEEE 802.15.4

An IEEE 802.15.4 [26] standard defined physical and medium access control layer for low rate wireless personal area networks to establish connectivity between low cost and low-power consuming devices. The 802.15.4 standard defines two types of network nodes.

- *Full Function Device (FFD):* Supports all characteristics from the standard. One FFD can be a network coordinator, a router, or a gateway which connects the network to other networks or it can just be a common node. FFDs are capable of communicating to any other device in the network.
- *Reduced Function Device (RFD):* RFDs are very simple devices with limited resources and it can only talk to a FFD. RFDs have low-power consumption and low complexity.

The nodes can combine to form networks with star or peer-to-peer topologies. Both topologies can use a PAN (Personal Area Network) coordinator node which is a FFD. The star topology is formed around a PAN coordinator node and is the only node allowed to form links with more than one device. In peer to peer topology, each

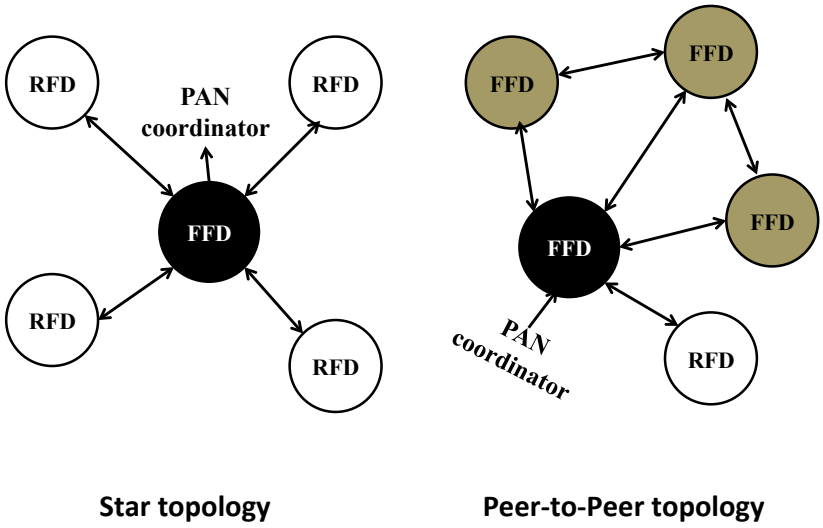

Figure 2.1: IEEE 802.15.4 Star and Peer-to-Peer topologies

device is able to form multiple direct links to other devices so that redundant paths are available. Both topologies are shown in figure 2.1.

Star topology is preferable when the coverage area is small. Communications are controlled by the PAN coordinator, which sends beacons for synchronisation. Nodes are allowed to communicate only with the coordinator and any FFD may establish its own network.

Peer-to-peer topology is preferable when large area has to be covered. The topology allows more complex networks. Their extension is only limited by the distance between each pair of nodes. They are meant to serve as the basis for ad hoc networks capable of performing self-management and organisation. They can support multi-hop communications.

The IEEE 802.15.4 uses a protocol based on a CSMA/CA algorithm, which requires listening to the channel before transmission to reduce collisions. IEEE 802.15.4 defines two different operational modes, namely beacon-enabled and non-beacon enabled, which correspond to two different channel access mechanisms. In non beacon-enabled mode, nodes use an unslotted CSMA/CA protocol to access the channel and transmit their packets. The node senses the channel before transmission, if channel

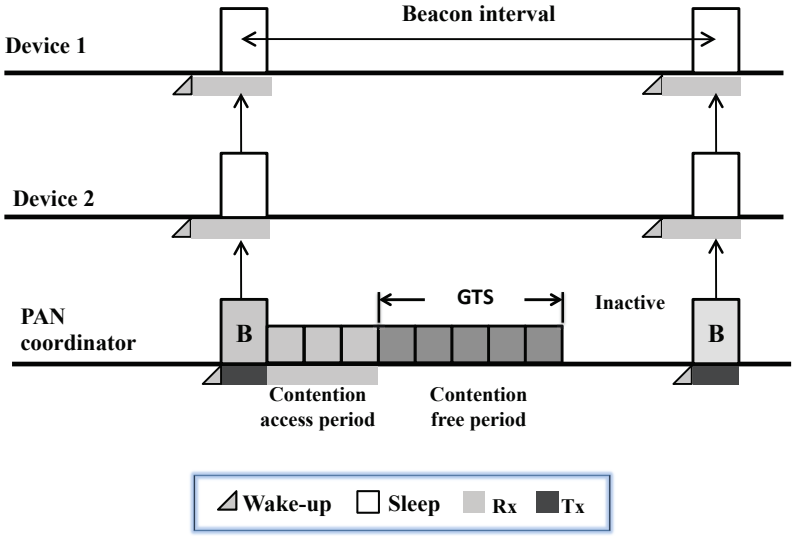

Figure 2.2: IEEE 802.15.4 beacon-enabled SuperFrame structure

is found busy, the node backs off for a random amount of time and retries again after that time. If the channel is idle, it transmits immediately. If it has to wait for a certain predefined time limit, the CSMA/CA algorithm will the terminate packet transmission with a failure.

In a beacon-enabled mode, the channel is managed through a superframe. The superframe starts with a beacon, transmitted regularly by the PAN coordinator. The nodes wake up at each beacon, perform a period of activity and go back to sleep until the next beacon. Active period is further divided into Contention Access Period (CAP) and a Contention Free Period (CFP) as illustrated in figure 2.2. At the start of an active period, the coordinator sends information about period duration in beacon frames, so that the duty cycle can vary. The beacon is followed by the CAP, which allows the nodes to send frames using slotted CSMA/CA. At the end of CAP, Contention Free Period (CFP) begins.

For bandwidth reservation in CFP, a node wanting a Guaranteed Time Slot (GTS) sends a reservation request to the coordinator during the CAP using slotted CSMA/CA. The coordinator of the network then allocates a slot (among limited slots

available) to the requesting node. The allocation information is provided in the beacons. During the inactive period, the coordinator may go to sleep.

The authors in [31] proposed a simple differentiated service scheme for slotted CSMA/CA in IEEE 802.15.4 to improve the performance of time-sensitive messages. Without any fundamental changes to the MAC protocol, they tuned certain parameters, according to the criticality of the messages. The results in [31] showed that tuning the parameters of slotted CSMA/CA might result in an improved QoS for time-critical messages. The authors argue that this practical proposal can be easily adopted in the IEEE 802.15.4b extension of the standard since it only requires minor add-ons and ensures backward compatibility with the existing standard.

2.3 Synchronous Protocols

Synchronous protocols rely on time synchronisation. Every node is supposed to be synchronised with their neighbours. The goal is to sleep and wakeup at the same time so that communication is easier. The level of synchronisation however differs according to the protocol: the slotted schemes require a tight synchronisation while the schemes relying on a common active/sleep period are less restrictive on that matter. We briefly describe some major synchronous protocols below.

2.3.1 SMAC

Sensor-MAC (SMAC) protocol [22] is a contention based MAC protocol which is derived from IEEE 802.11 protocol specially designed for Wireless Sensor Networks. The basic idea of SMAC protocol [22] consists of dividing the cycle into periodic listen - sleep intervals. It manages synchronisations locally and these synchronisations help in keeping the same sleep - listen intervals of sensor nodes. The nodes exchange the schedule by broadcasting the SYNC packet periodically. The period for each node to send a SYNC packet is called the synchronisation period. The Neighbouring nodes following the same schedule form virtual clusters having common listening time. The nodes on the border of cluster (named as Border Nodes) might come in range of different virtual clusters, which results in them following multiple schedules, i.e, they need to be listening to the clusters (whom schedule they received) because these border nodes sustain network connectivity by ensuring packet passing from one

cluster to another. This possibility of following multiple schedules, results in higher energy consumption. Some implementations suggest that border nodes adopt only some schedules to reduce the time during which the radio is on. Although this further saves energy, it may cause network fragmentation as some virtual clusters may be isolated.

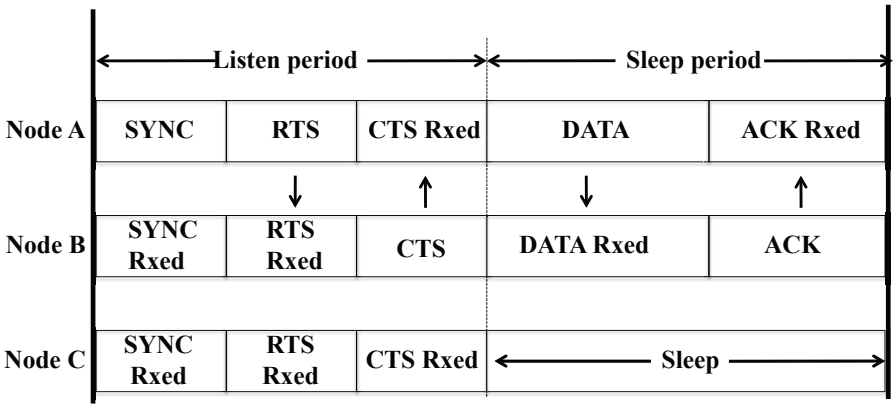

Figure 2.3: Communication among two neighbours using SMAC protocol

Nodes exchange a SYNC packet periodically. This SYNC message is broadcasted to the immediate neighbours during the synchronisation period. Figure 2.3 shows the communication taking place among two neighbours. RTS/CTS packet exchanges take place through unicast communication.

SMAC introduces message passing by dividing long messages into frames and sent in a burst. Using this technique, communication overhead is minimised, hence energy saved but at the expense of fairness in the medium. Periodic sleeping works best for up to two hops but for multi-hop routing algorithms, periodic sleeping may result in high latency as all the nodes on the path to the sink will have their own schedules. This latency is called as the sleep delay [22].

The adaptive listening technique is proposed to improve the overall latency problem. In adaptive listening, the node overhearing the neighbour's transmission wakes

up for a short interval at the end of transmission. Hence, if it is a next-hop neighbour node, it can receive the data immediately. The end of transmissions is established by the duration field of RTS/CTS packets. In SMAC, the energy waste caused by idle listening is reduced by sleep schedules. In addition to its implementation simplicity, time synchronisation overhead may be prevented by sleep schedule announcements. On the other hand, Broadcast data packets do not use RTS/CTS in SMAC, which increase the collision probability. Adaptive listening incurs overhearing or idle listening if the packet is not destined to the listening node. Sleep and listen periods are predefined and constant, which decreases the efficiency of the algorithm under variable traffic load.

2.3.2 TMAC

Static sleep-listen periods of SMAC [22] result in high latency and lower throughput as indicated earlier. Timeout-MAC (TMAC) [21] works on the same principle as SMAC but activation time of the nodes is adapted according to the traffic. TMAC improves the results of SMAC protocol under variable traffic load. In TMAC, listen period of a node ends when no activation event has occurred for a predetermined time threshold *TA* as shown in figure 2.4. *TA* must be long enough so that a node can sense the carrier and hear a potential clear to send (CTS) from a neighbour. Variable load in sensor networks are expected, since the nodes that are closer to the sink must relay more traffic. Although TMAC gives better results under these variable loads, the synchronisation of the listen periods within virtual clusters may partially break down. This is one of the reasons of the early sleeping problem.

The early sleep occurs when a node, especially third hop node, switches to sleep mode while a neighbour node still has messages for it. However, Future Request To Send (FRTS) frames is used to notify the third hop nodes either to extend their *TA* expiration, or to let them awake by the appropriate time. TMAC improves on SMACÕs energy savings and minimises collisions and redundancy, since idle nodes switch back to the sleep mode relatively earlier. However, TMAC faces problems of reduced throughput and higher network latency. TMAC also suffers from synchronisation and scaling problems.

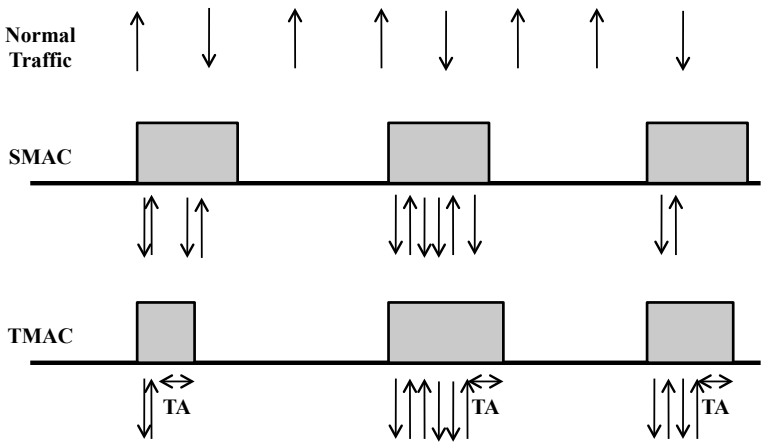

Figure 2.4: TMAC downsizes active period lengths to further save energy. The arrows in the figure indicate transmitted and received frames. In case no traffic occurs during the time *TA*, TMAC can end the active period prematurely.

2.3.3 DSMAC

Dynamic Sensor-MAC (DSMAC) [32] adds dynamic duty cycle feature to S-MAC. It aims to decrease the latency for delay-sensitive applications according to its energy level and traffic requirements. A node using DSMAC keeps track of its energy consumption level and average latency it has experienced and tries to dynamically adjust its duty cycle accordingly. Within the SYNC period, all nodes share their one-hop latency values (time between the reception of a packet into the queue and its transmission). All nodes start with the same duty cycle. Figure 2.5 depicts DSMAC duty cycle doubling.

When a receiver node notices that average one-hop latency value is high, it decides to shorten its sleep time and announces it within SYNC period. Accordingly, after a sender node receives this sleep period decrement signal, it checks its queue for packets destined to that receiver node. If there are packets to be transmitted and if its battery level is above a specified threshold, it doubles its duty cycle. The duty cycle is doubled so that the schedules of the neighbours will not be affected. DSMAC reduces the latency in comparison to S-MAC. It is also shown to have better average power

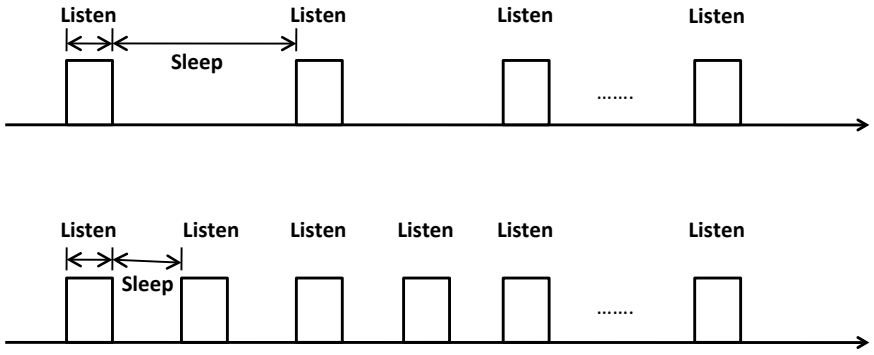

Figure 2.5: Duty cycle doubling. Neighbouring nodes having different duty cycles can still communicate with old schedule

consumption per packet. At high traffic DSMAC achieves lower throughput. With varying duty cycles, synchronisation of a virtual cluster may get affected. Complexity in adapting duty cycles, particularly within a virtual cluster and under high traffic loads, increases many times the already existing synchronisation overhead.

2.3.4 iqueue-MAC

iQueue-MAC [33] is a hybrid approach using CSMA for light traffic conditions and TDMA for heavy traffic loads. Authors propose energy-efficient throughput enhancement for heavy or bursty traffic. iQueue-MAC mainly targets data collecting networks that often adopt tree routing structure. The network uses two kind of devices: simple nodes (e.g. RFD of IEEE 802.15.4) and routers (FFD of IEEE 802.15.4, such as coordinators). Simple nodes minimise the energy consumption and only wake up to transmit data. Each simple node is associated to the router, which in turn is responsible for collecting data packets from these nodes. iQueue-MAC requires only local beacon synchronisation and does not limit the number of neighbour senders.

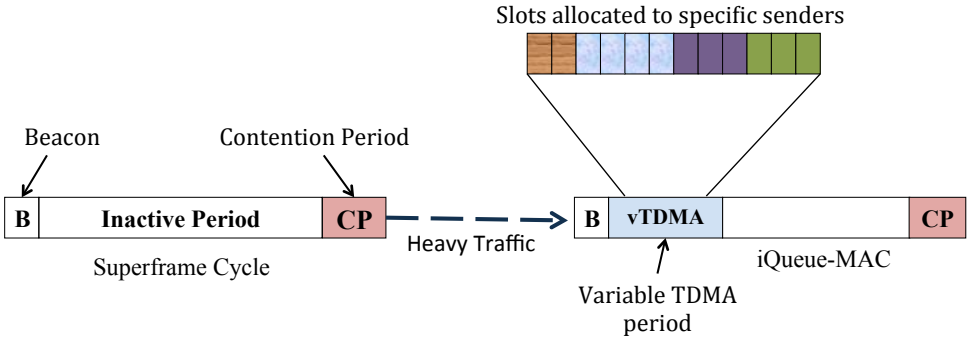

Figure 2.6: The basic idea of the iQueue-MAC. A variable TDMA period and a CSMA period are integrated to handle adaptive traffic.

iQueue-MAC uses a variable TDMA period and a CSMA period for energy efficient and traffic adaptive data transmission. The basic idea of iQueue-MAC is shown in Figure 2.6. During light traffic period, iQueue-MAC acts like other low duty-cycle MACs to conserve power.

During heavy traffic periods, iQueue-MAC senses the building up of queues and allocates extra variable TDMA period (vTDMA) slots to the demanding nodes within the inactive period for throughput enhancement. If the simple nodes have pending data packets, they apply in contention period for extra transmissions slots. Then the router allocates the requested slots to those nodes in vTDMA period.

The queue length value is piggybacked by the sender node onto every data packet. Upon receiving a data packet, the router checks the queue length information. If the value of queue length is non-zero, the router allocates the corresponding slots to the specific sender in the next cycle. As soon as iQueue-MAC senses packet queuing, it dispatches packets transmissions in the TDMA phase, which leads to short queue length and short packet delay. iQueue-MAC provides high energy-efficiency by transforming most of the communication into a slot-organised TDMA round.

2.3.5 Minimizing sleep-delay And nanoMAC

Adaptive Listening [22] suggests the use of overhearing to reduce the sleep delay. In adaptive listening, the node overhearing the transmission of neighbour, gets the information of duration of that transmission and hence may go to sleep and wake up just when the transmission ends. This idea has also been proposed in nanoMAC [34]. In nanoMAC, the node wakes up after that transmission even if it might turn out to be during its sleep time. This way the neighbour node can send data immediately, instead of waiting for node to wake up in its next scheduled wake up period.

2.3.6 FPA

FPA (Fast Path Algorithm [35]) wakes up the nodes for extra time even during their pre-programmed sleep time, to make sure relaying of frames in time. A node can calculate the estimated time when its upstream neighbour will send a frame to it, using its hop distance from the sender. At the estimated time of communication, the node wakes up and receive and potentially forward the frame to its downstream neighbour. The node sets these additional wakeup times from information piggybacked in the first data packet on that path.

2.4 Asynchronous Protocols

Asynchronous Protocols are the protocols in which each node does not need to synchronise with their neighbours and can choose their own wakeup slot, without knowing the wakeup slots of their neighbours. We will present few asynchronous protocols available in literature below.

2.4.1 WiseMAC

In [36], authors proposed WiseMAC protocol which uses non-persistent CSMA (np-CSMA) with preamble sampling to overcome idle listening. In the preamble sampling technique, a preamble is sent before each data packet for alerting the receiving node.

WiseMAC allows to reduce the size of the preamble for unicast communications. To reduce the power consumption incurred by the predetermined fixed-length preamble, WiseMAC offers a method to dynamically determine the preamble length. Every node stores the wake up times of its neighbours. Based on neighbour's sleep schedule table, WiseMAC schedules transmissions so that the destination node's sampling time corresponds to the middle of the sender's preamble.

Figure 2.7 presents the WiseMAC concept.

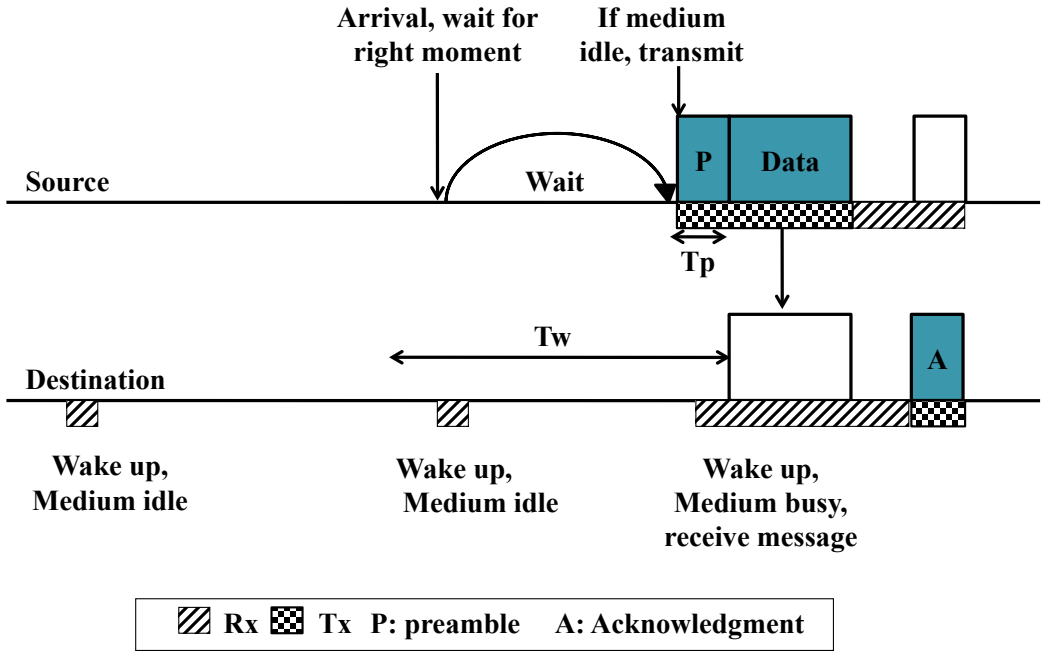

Figure 2.7: Conceptual depiction of WiseMAC [36]

The preamble length depends on the quality of the clocks of the nodes. If the clocks differ slightly, the preamble can be very short. It only has to be large enough to absorb the maximum drift between the clocks. This technique reduces the listening time of preamble, but if two nodes have similar wakeup times, they'll stay awake for both packets and thus partly for traffic that is not addressed to them.

WiseMAC is based on non-persistent CSMA which results in collisions when one node starts to transmit the preamble to a node that is already receiving another node's transmission where the preamble sender is not within the range.

2.4.2 BMAC

The Berkeley Media Access Control (BMAC) [20] is a contention based asynchronous MAC protocol for WSNs, which uses CSMA/CA. B-MAC uses clear channel assessment (CCA) and packet backoffs for channel arbitration, link layer acknowledgments for reliability and low power listening (LPL) [37] for low power communication. BMAC tries not to keep the nodes awake, if no preamble is transmitted. The sender node in BMAC uses preamble signalling to initiate its potential receiver. All nodes periodically wake up at the beginning of their duty cycle and start checking preamble signals. If a preamble is found, they stay awake and keep their radios on. Otherwise, they turn on the radios after a data packet arrives or after a time-out value. The sender needs to send a long preamble in order to notify the receiver about the next transmission of data packet. The preamble length should be at most about the length of two operational cycles.

2.4.3 XMAC

X-MAC [23] revisited the concept of preambles by introducing packetised preambles. The long preamble is divided into alternating packets and periods of contention. The short preamble packet contains the destination address, which serves two purposes. It helps irrelevant nodes to go to sleep and it allows the intended receiver to immediately involve in the data transmission. The contention period allows to send an acknowledgement to preamble packets. When a node needs to send a packet, it sends preamble packets containing the destination address. When the destination node wakes up, it responds with an acknowledgment to the preamble packet, it stops the transmission of the preamble and the payload data packet is sent.

After receiving a data packet, a receiver in X-MAC stays awake for duration equal to the maximum backoff window size to allow queued packets to be continuously transmitted. By doing so, X-MAC saves a huge amount energy by avoiding overhearing while reducing latency almost by half on average.

2.4.4 ContikiMAC

ContikiMAC [38] borrows mechanisms of B-MAC [20], WiseMAC [36] and X-MAC [23]. ContikiMAC uses the preamble, divided into packets. However, instead

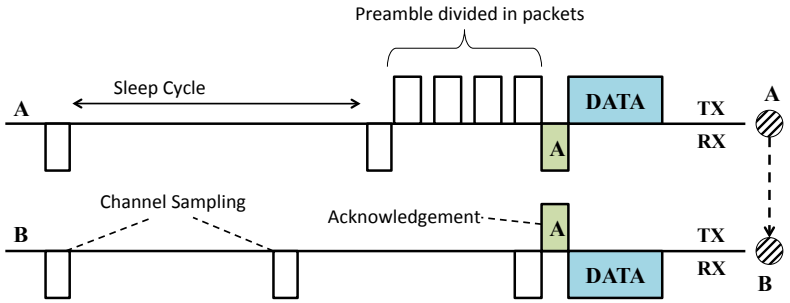

Figure 2.8: An example of XMAC: Source node A transmitting to Destination node B

of sending short preamble packets as X-MAC [23], the transmitter sends the data directly in the preamble. When a node wakes up, it has the same behaviour as in X-MAC, except that it directly accepts the payload packet. A mechanism for storing Wakeup times of the neighbours similar to WiseMAC is implemented. A mechanism for rapid re-sleep is used to limit the impact of false-alarm Clear Channel Assessment. Nodes go back to sleep if they detect a signal for a period, that does not match any pattern of communication. The detected signal is considered as noise and the node goes back to sleep. All these mechanisms can reduce the energy consumption. However, with these type of protocols, broadcast becomes expensive in terms of energy because it is necessary to synchronise all neighbours of the transmitter before transmission or the nodes have to transmit several times.

2.4.5 Aloha with Preamble Sampling

Authors in [39] proposed a protocol with a combination of ALOHA protocol [40] with preamble sampling technique, hence the name Aloha with preamble sampling. They proposed low power listening technique which switches the radio on and off periodically (duty cycling the radio). This approach works at the physical layer based on the PHY Header going to sensor's radio. The header starts with a preamble

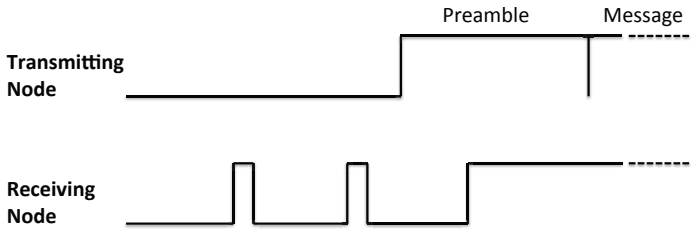

Figure 2.9: Low power listening and Preamble sampling

which informs the receiver about the upcoming messages. The receiver node turns on the radio periodically to check any incoming messages. If preamble is detected, it continues listening for the normal message reception. If no preamble is detected, it turns off radio till the next sample. The process is shown in figure 2.9. Aloha with preamble sampling is suitable for low traffic applications.

2.4.6 DMAC

In sensor networks, the communication mostly occurs from sources to the sink. This converge-cast communication can be represented as data gathering trees. DMAC [19] claims to be low latency energy efficient MAC protocol.

DMAC can be summarised as an improvement of Slotted Aloha algorithm, where slots are assigned to the sets of nodes based on a data gathering tree as shown in Figure 2.10.

During the reception period of a node, all of its child nodes have transmission periods and they contend for the medium to transmit to the node. Low latency is achieved by assigning subsequent slots to the nodes that are successive in the data transmission path. DMAC achieves good latency and can be a very good candidate in scenarios which are non-delay tolerant.

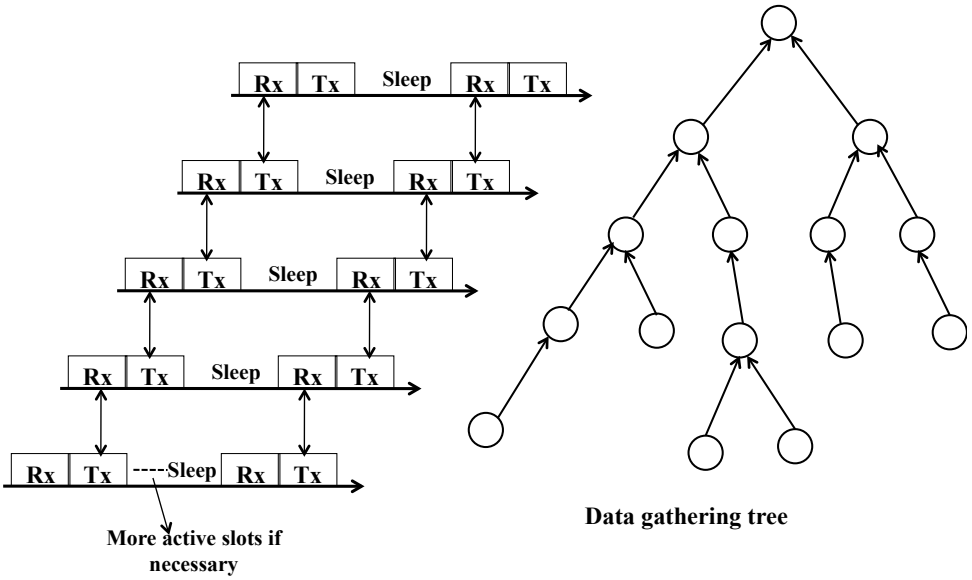

Figure 2.10: A data gathering tree and its DMAC implementation [19]

The problem occurs when a number of nodes, having the same schedule (same level in the tree) try to communicate to the same node, collisions will occur and no collision avoidance methods are utilised in DMAC. This is a possible scenario in event-triggered sensor networks.

2.4.7 ZMAC

Z-MAC [41] is a hybrid MAC protocol combining the advantages of TDMA and CSMA to achieve better channel utilisation and lower latency under different traffic levels. ZMAC behaves like CSMA under low contention and like TDMA under high contention. It has a setup phase of neighbour discovery, slot assignment, local frame exchange and global time synchronisation.

ZMAC uses DRAND, which is an efficient scalable channels scheduling algorithm [42]. DRAND can assign unique slots to the neighbour nodes and avoid the hidden terminal problem. A node can transmit at any time slot, after sensing the channel and can transmit once the channel is clear. But the owner node (To whom the slot

has been assigned) always has the priority in accessing the channel. Owner nodes are given the chance to transmit at first, but when a slot is not in use by its owners, the other nodes can steal the slot.

In ZMAC, a node can switch between two modes of operation: low contention level (LCL) or high contention level (HCL). During LCL, non-owner nodes can compete to transmit in any slow but with low priority. When a node start experiencing more data contention, its switches to HCL mode. In HCL mode, only the owners of the current slot and their one-hop neighbours are allowed to contend for the channel, the owner of the node having the highest priority.

ZMAC uses a simple local clock synchronisation scheme among sender in two-hop neighbourhoods, where sending node adjusts the synchronisation frequency based on its current data rate and its resources (receivers do not send synchronisation messages).

ZMAC has the advantages of CSMA under low contention(i.e., high channel utilisation and low delays) and benefits of TDMA under high traffic levels (i.e., high channel utilisation, low contention and fairness).

2.4.8 Qos-CoSenS

CoSenS (Collect then Send burst Scheme) [43] is proposed to overcome CSMA/CA's drawbacks by enhancing its performances for medium and high traffic loads and to facilitate the implementation of QoS mechanisms. It is implemented on top of CSMA/CA. In CoSenS, the routers have the priority over simple nodes for medium access. The idea is that during the period named Waiting Period (WP), the router should not transmit packets one by one on their arrival, instead it should wait and collet data from its children (its associated simple nodes) or other neighbour routers. After the end of the WP, the router starts transmitting all packets queued in its buffer in a single burst during the time period named Transmission Period (TP). At the end of TP, the router start another cycle and goes again to the WP for receiving packets. The duration of the WP depends on the received traffic. The WP and TP are shown in figure 2.11.

The performance analysis shows that CoSenS greatly enhances throughput, end to end delay and reliability.

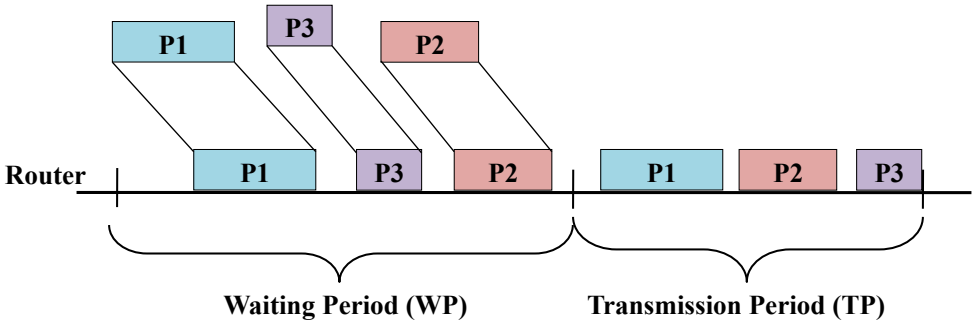

Figure 2.11: An example of how CoSenS works [43]

2.5 Real-time MAC Protocols

Applications with timeliness constraints are called realtime applications. Real time applications demand that the network should transmit a message within a known and bounded delay. Taking more time can have a potentially (very) negative impact. Real-time applications are classified in two categories. Some applications have strict real time requirements, for example a radioactive leakage detection in a nuclear power plant and other such life saving applications. Other applications might tolerate that fraction of messages do not meet their deadline (e.g., multimedia applications).

Some real-time MAC protocols for WSNs are inspired by industrial wired networks. Because constraints of WSNs and wired LAN settings are very different, these protocols sometimes put unrealistic constraints which severely limit their applicability.

In the next section, we give a brief introduction to some of the real time MAC protocols proposed in the literature in our knowledge.

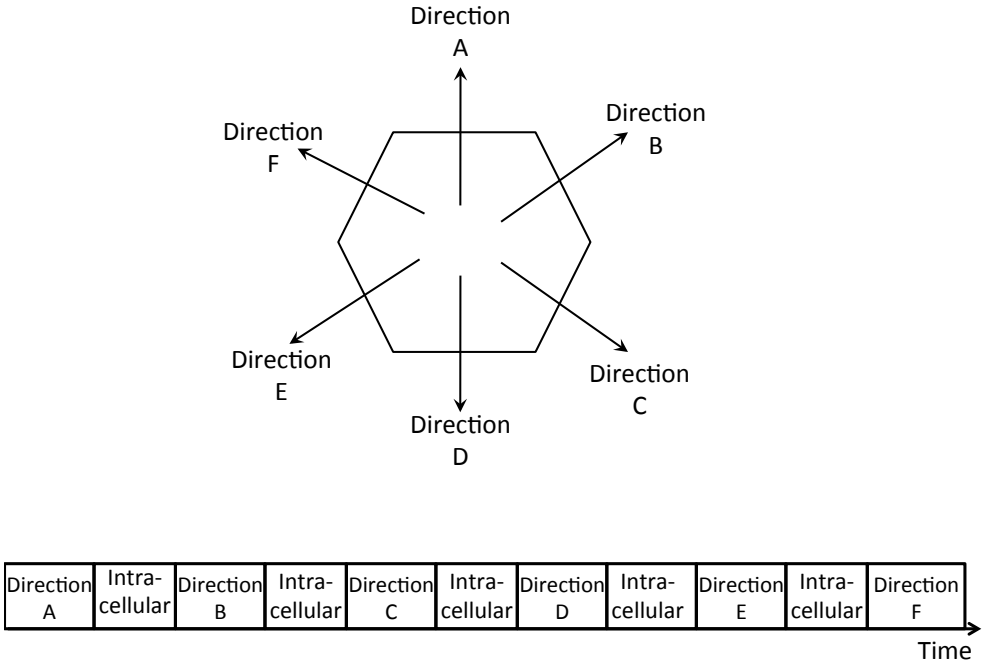

Figure 2.12: General behaviour of IEDF protocol

2.5.1 IEDF

The Implicit Earliest Deadline First (I-EDF) protocol [44] assumes the network is organised in regular hexagonal cells, with a router node in the middle of each cell. The nodes of each cell form a cluster operating on different carrier frequency than the neighbouring cells. Time is divided between the intra-cellular and inter-cellular communication.

FDMA is used for intra-cellular communication, the nodes inside each cell are assumed to be fully connected, each node in the cell knows the frequency and size of messages from all nodes in the cluster, which means a node can transmit messages to any other node in a single hop. Between two intra-cellular communications, time slots are reserved for inter-cellular communication.

During this period, a router node which is in the centre of the cell transmits the data retrieved during the intra-cell communications. A router node can transmit in six directions corresponding to six sides of the hexagonal cell. To transmit in one direction, it uses the frequency corresponding to the neighbour it wants to reach. This approach guarantees bounded end-to-end delay. However, energy consumption

is not taken into account, there is no sleep cycle. In addition, the hexagonal cells configuration is extremely constraining. Finally, the nodes must be capable of communicating on seven different carrier frequencies, this increases the cost of network elements which can be a problem for large-scale networks.

2.5.2 TRAMA

Traffic-Adaptive MAC Protocol (TRAMA) [28] is a TDMA-based algorithm and proposed to increase the utilisation of classical TDMA in a collision-free and an energy efficient manner. The nodes are supposed to be synchronised globally.

Time is divided into slots having two parts: random-access and scheduled-access (transmission) periods. Random-access period is used for two-hop topology information gathering, where channel access is contention-based. Nodes then exchange information on the traffic they have to transmit during a predefined period. During this period, based on information from neighbours and traffic, the node calculates the number of slots for which it will have the highest priority among two-hop neighbours. The node wanting to transmit and having the highest priority for the slot can then use that slot. It also indicates the intended receivers for these slots through a schedule packet. The node also announces the slots which it will not use despite having highest priority. The potential senders are evaluated for re-use of announced vacant slots. The priorities of the nodes are known thanks to the discovery of 2-hop neighbourhood, they are exchanged with a pseudo-random generator using the technique described in [45], it helps ensure fairness in channel access.

The main disadvantages of TRAMA are cumbersome control exchanges to indicate traffic and topology, and the fact that node's information must be consistent enough to ensure the absence of collisions in slots in fixed access. There is a risk that two nodes of the same 2-hop neighbourhood might have the highest priority for the same slot. Also, if the network is very dense, 2-hop neighbourhood of a node can become too large to be stored in its memory. With TRAMA, higher percentage of sleep time and less collision probability is achieved compared to CSMA based protocols.

2.5.3 FMAC

FMAC [46] (Framelet MAC) is a contention-based MAC protocol guaranteeing bounded bandwidth and delay without requiring synchronisation. It is a localised protocol that allows deterministic access to the medium. FMAC protocol uses a framelet approach. A node which wants to transmit a packet sends multiple instances of the same packet (named framelet) within a specific retransmission period (i.e. the amount of time between two successive retransmissions). The authors showed that by applying mathematical rules for period selection, it can be ensured that at least one framelet from each node is transmitted without collision, even when all its neighbours are transmitting. Moreover, nodes can be grouped into clusters, each cluster having its own framelet period.

The FMAC functions in a fully asynchronous manner and there is no preamble sampling mechanism. Nodes that have nothing to transmit continuously listen to the channel which leads to high energy consumption. The authors argue that it is possible to use a preamble sampling mechanism with FMAC. The biggest drawback of this approach is its poor bandwidth utilisation as the same information is sent in times. Furthermore, the worst case delay increases exponentially with the number of nodes within the same collision domain. This protocol is not suitable for large scale and dense sensor networks.

2.5.4 PEDAMACS

Power Efficient and Delay Aware Medium Access Control protocol for Sensor networks (PEDAMACS [47]) is a TDMA scheme that extends the common one-hop TDMA to a multi-hop sensor network. Assuming the sink gathers topology (connectivity) information, for example, through an initialisation phase. It is then able to build a network-wide TDMA schedule using this information. PEDAMACS assumes being equipped with high-powered sink to synchronise the nodes and to schedule their transmissions and receptions. The sink node can then send the global TDMA schedule after gathering topology information. The sensor nodes communicate with the sink through a multi-hop path.

2.5.5 RT-LINK

RT-LINK [48] is a TDMA-based link protocol and is applicable to networks which require predictability in throughput, latency and energy consumption. RT-LINK uses hardware-based global time synchronisation. RT-LINK includes two phases which are topology-gathering and scheduling. A cycle is defined as the duration between two synchronisation pulses. Each cycle consists of a large number of frames divided into scheduled and contention slots. A node which intends to transmit data, randomly chooses a slot within the contention slots and sends hello messages. A hello message is transmitted to the sink node in a multi-hop manner. The sink is responsible for network wide slot assignment. The node is active in its assigned slots.

2.5.6 PR-MAC

PR-MAC [49] (Path-oriented Real-time MAC protocol) is designed to ensure a bounded delay of data transmission in sleep/wakeup sensor networks. It targets monitoring applications where data is sent periodically. It aims to obtain a bounded and minimal end-to-end delay of data transmission and fast adaptation of dynamic real time requirements. Two causes of data transmission delay are addressed in PR-MAC. Sleep latency through a schedule algorithm called Bidirectional Pipelining Schedule (BPS) and a multi-channel communication mechanism is used to reduce data latency.

A sensor node sends a message to the sink using a (non real-time) contention-based MAC protocol. This message contains a description of the sensed value, and the path taken by the message. Using the reverse path, the sink sends a series of control messages to the relaying nodes, indicating the periodicity of the subsequent messages and act as resource reservation messages. Once all relaying nodes are contacted, the path is set up and the sink node now can expect data messages to reach it in a real-time fashion.

2.5.7 Dual-Mode MAC

Dual-Mode MAC [50] is a hard real-time MAC layer protocol for linear WSNs. This protocol is only applicable to the networks having a line topology (monitoring of highways, pipelines, etc.).

As the name suggests, Dual-Mode MAC protocol has two modes of operation. Unprotected mode which is a contention based mode and protected mode which is a contention free mode. Unprotected mode allows optimal time for medium access when there are no collisions. Each node relays a message when a back-off time proportional to its distance to the sink elapses (and no other message is heard). When a collision occurs, the protocol enters into the protected mode: a path is then reserved to ensure minimum delay and avoid collisions. The unprotected node allows faster transmission, hence when the number of messages fall below a certain threshold, the nodes return to the unprotected mode. This protocol does not require synchronisation and constructs a schedule in a fully distributed manner. But the protocol only works on networks with a line topology which reduces the number of applications considerably. Moreover, the energy consumptions of the nodes are ignored.

2.5.8 AS-MAC

An energy- efficient Asynchronous Scheduled MAC protocol (AS-MAC) is presented in [51]. AS-MAC uses duty cycling to avoid idle listening and employs Low-Power-Listening (LPL) to minimise the periodic wakeup time. The nodes store the wakeup schedules of their neighbours. As the nodes know the wake up time of their neighbours, hence does not require long preambles at the start of transmission. AS-MAC asynchronously coordinates the wakeup times of neighbouring nodes to reduce the problems of overhearing, contention and delay, which are unavoidable in synchronous scheduled MAC protocols such as SMAC [22], TMAC [21], and SCP-MAC[52].

A multi-hop energy consumption model for the proposed protocol is presented and its performance is compared with SCP-MAC.

To validate the design and the energy model, the authors implemented their protocol AS-MAC [51] in two major TinyOS platforms. The experimental results showed that AS-MAC, considerably reduces energy consumption while providing good delay and packet loss in comparison with existing WSN MAC protocols.

2.5.9 ER-MAC

ER-MAC [53] is a hybrid MAC protocol for emergency response WSNs having flexibility to adapt well to traffic and topology changes. ER-MAC allows contention in TDMA slots to cope with large volumes of traffic. This scheme trades energy efficiency for higher delivery ratio and lower latency.

ER-MAC schedules collision-free slots. During the normal monitoring conditions, nodes sleep to save energy and only wake up for their scheduled slots. During an emergency, the nodes participating in the emergency monitoring allow contention in each slot to achieve high delivery ratio and low latency, but they have to sacrifice energy efficiency due to this change in MAC behaviour. ER-MAC supports fairness so that the sink can receive complete information from all the nodes. The protocol maintains two priority queues to separate high priority packets from low priority packets. It also offers a synchronised and loose slot structure, where nodes can modify their schedules locally. This allows nodes to join or leave the network easily. The simulation results in [53] demonstrated the scalability of ER-MAC and showed that ER-MAC achieves higher delivery ratio and lower latency with low energy consumption compared to Z-MAC [41].

2.5.10 DW-MAC

Demand Wakeup MAC (DW-MAC) [54] introduces a low-overhead scheduling algorithm that allows nodes to wake up on demand during the Sleep period of an operational cycle and ensures that data transmissions do not collide at their intended receivers. This demand wakeup adaptively increases effective channel capacity during an operational cycle as traffic load increases, allowing DW-MAC to achieve low delivery latency under a wide range of traffic loads including both unicast and broadcast traffic.

DW-MAC outperforms SMAC [22] and RMAC [55] protocols, with increasing benefits as traffic load increases. For example, under high unicast traffic load, DW-MAC reduces delivery latency by 70% compared to S-MAC and RMAC, and uses only 50% of the energy consumed with S-MAC with adaptive listening. Under broadcast traffic, DW- MAC reduces latency by more than 50% on average while maintaining higher energy efficiency.

2.5.11 PDCA

The work in [56] presents a predictive framework to provide nodes with an ability to anticipate the arrival of objects in the field-of-view of their cameras. The approach is called Predictive Duty Cycle Adaptation (PDCA). This predictive framework increases the duty cycle of the nodes which are one step away from the immediate neighbourhood nodes where the objects are currently visible. This approach eliminates the need for the busy nodes to inform their non-busy neighbours to get ready for the arrival of the object, ending up with a more robust strategy for updating the duty cycle at the nodes where the objects are highly likely to appear soon. The proposed scheme works by using an existing MAC header bit that is already in the 802.15.4 protocol and, in that sense, the anticipatory approach for notifying the nodes about the current state of the object location uses no additional expenditure of energy.

The adaptation at a node takes place on the basis of the probability of object arrival at the node. This probability is estimated using a Kalman filter that takes into account both the direct measurements of the object position, when the object can be seen by the camera at the node, and on the basis of the indirect measurements of the object's position as conveyed by the Explicit Event Notification (EEN) bit in the MAC header. In case of strict latency requirement, this approach has proved to be effective.

2.5.12 VTS

In [57] the authors present the virtual TDMA for sensors (VTS) MAC protocol, a protocol for WSNs with bounded latency. VTS provides TDMA access scheme, dynamically creating a superframe of time-slots and adapting its length to the number of actual nodes in a cell for optimum performance. After a transient adjustment phase, this mechanism results in a scalable and collision-free MAC protocol, consuming considerably less energy in comparison to contention-based protocols and has a bound packet latency (providing support for soft real-time services). VTS addresses network setup and synchronisation issues as well. But at the price of slightly worse average latency than contention protocols under low/medium loads.

VTS protocol borrows the synchronisation mechanism from SMAC [22] to establish listen/sleep schedule but unlike SMAC, only one node can transmit in every listen/sleep cycle. Thus, every cycle becomes a time-slot. This simple procedure

means the nodes in a cluster will transmit in different time-slots. In this way, a frame of time-slots is built in a distributed way as each node is transmitting in a different time-slot. VTS allows the reduction or increase in the number of time-slots of the frame, which improves throughput compared to a TDMA frame with fixed number of time-slots. With this TDMA-like access there is no contention for data transmission and latency is guaranteed. Finally, in order to meet realtime deadlines, VTS nodes adapt the duration of the sleep interval (dynamic duty cycle) to keep latency below a given threshold when nodes dynamically leave and join a cell. This way, when new nodes join a TDMA frame, the sleep interval is reduced in order to keep constant maximum latency. Simulations results showed that VTS save considerable amount of energy. Under low loads VTS compromises latency and energy consumption, while it guarantees low latency at high loads.

2.5.13 ADV-MAC

ADV-MAC[58] is a contention-based protocol that has two contention periods: Advertisement (ADV) period and Data period. Nodes that have data to exchange reserve the medium using short ADV packets during ADV period and Data period is used for the data exchange. The data exchange in the Data period is contention based and follows a sequence of RTS/CTS/Data/ACK exchanges. ADV-MAC improves in terms of energy savings, packet delivery ratio (PDR) and latency in comparison to well known contention based protocols such as SensorMAC (S-MAC) [22] and Timeout-MAC (T-MAC) [21]. However, the efficiency of ADV-MAC can further be improved by eliminating one of the two contention periods, which is the basis of ATMA [29], which is described next.

2.5.14 ATMA

In [29] authors proposed Advertisement-based Time-division Multiple Access (ATMA) , a distributed TDMA-based MAC protocol for wireless sensor networks that utilises the bursty or periodic nature of the traffic to prevent energy waste through advertisements and reservations for data slots. ATMA efficiently combines contention-based and TDMA-based approaches, where the reservation management is done using contention-based medium access and data transmissions are done using TDMA-based medium access.

ATMA defines an Advertisement (ADV) period, nodes exchange ADV packets to reserve slots for data exchange in the data period. ADV packet includes the intended receiver information and the corresponding ACK packets. This assures successful reservations and reduces the hidden terminal problem. Since the nodes that are not part of the data exchange sleep in the data period, this approach minimises the idle listening and overhearing which are the most important sources of energy waste at MAC level. ATMA reserves data slots for a specific duration, which enables adaptation to varying traffic.

Simulations showed that the ATMA protocol adapts nicely to dynamic bursty traffic, providing reductions in energy consumption as high as 80% compared to other WSN MAC protocols such as Sensor-MAC (S-MAC) [22], Timeout-MAC (T-MAC) [21], Advertisement MAC (ADV-MAC) [58] and TRAMA [28]. ATMA also achieves very good packet delivery ratio and latency performance in comparison to the evaluated protocols.

2.5.15 ARQ-CRI PROTOCOL

A novel low-latency MAC protocol namely automatic repeat request-cooperative receiver initiated MAC protocol (ARQ-CRI) for low-power cooperative wireless sensor networks WSNs is proposed in [59], while preserving (i.e. in high traffic mode) or even increasing (i.e. in low traffic mode) energy-efficiency. The ARQ-CRI protocol [59] is designed based on the receiver-initiated protocols. In addition to that, a new relay selection technique is also proposed which is based on the wakeup period of relay nodes to reduce latency. ARQ-CRI does not need a perfect synchronisation between nodes for a common rendezvous point. ARQ-CRI does not need a perfect synchronisation and can be a really effective cooperative MAC protocol for real time applications as it gains on energy efficiency and latency.

Simulations results showed that ARQ-CRI performs better in terms of latency while preserving energy and in low traffic conditions.

2.6 Conclusions

Applications can be of various kinds, each having its own requirements. Having a specifically designed protocol is the need of the hour.

A simple approach for duty-cycling such as the one proposed by the 802.15.4 standard can be improved with synchronisation features to have common active periods (SMAC [22], TMAC [21] to name a few), with low-power listening (LPL) capabilities and preamble transmissions (B-MAC [20], X-MAC [23], TP-MAC [27] to name a few) or adapted to bursty traffic [29]. While synchronous approaches are not scalable for large networks, LPL and preamble-based approaches still suffer from high latencies when node's sleeping period is large.

TDMA approaches like [28] are highly responsive but as the events occur sporadically and there might be no event for a long period of time, the reserved radio duty cycle time slots for nodes are left empty and precious energy is lost. Energy preservation is therefore one main reason to avoid TDMA approaches despite their high level of responsiveness. An hybrid approach, such as [53], still suffer from high energy consumption in case of rare events detection.

The authors in [56] propose a predictive duty cycle adaptation mechanism for tracking in wireless image networks. This work is probably the closest to our work as it has also been targeted for image sensors. However, while [56] focuses on tracking by other sensor nodes once detection has been made, we focus on improving the responsiveness of our system in terms of alert relaying.

Moreover, although many of protocols mentioned in this chapter are delay-sensitive, i.e., have low-latency, none of these protocols takes criticality of the network into account. Also, when using sensor nodes equipped with camera (as we do in our research work), one have to consider capture rate of each node. As this capture rate will determine the quality of network surveillance. The aim is to find a delicate balance of energy consumptions, surveillance quality and low latency. Lots of research is still required in this area.

The applications can have variable criticality level. In addition to energy constraints, highly critical applications like intrusion detection applications demand strict surveillance quality and low latency in comparison to low criticality applications (e.g., environmental monitoring, home automation) which can be delay-tolerant. Also the requirements of the network can change with changing dynamics of the network topology.

Hence, it is preferable to design a MAC layer keeping in mind the critical requirements of the network. Our research work in this thesis is targeted for mission critical surveillance applications with image sensors. We proposed a low latency alert prop-

agation and low energy consumption MAC protocol for mission critical surveillance applications. Our goal is to link the duty cycle of nodes with the image capture rate. The contribution in this thesis is based on an original criticality model developed in our team [30] for image sensors. This criticality model is explained in the next chapter.

Chapter 3

Criticality Model

Contents

In this chapter we explain the criticality model, which is the basis of our research on MAC layer. We suppose random deployment of wireless image sensor nodes. The random deployment results in different redundancy level of every node. Some nodes might be more redundant than others. Hence, the frame capture speed of all the nodes cannot be (and should not be) the same. Also, the criticality level of the application is termed as an important parameter which can be used to efficiently utilise the network resources. The criticality model explained here, takes into consideration the redundancy of the node and application's criticality to calculate the frame capture rate of the node's camera. This frame capture rate can be a quality parameter of the network surveillance.

3.1 Criticality-based node scheduling

When considering the wireless sensor networks for critical applications, the application's criticality must be taken into account. Previous research on Critical applications like intrusion detection [60, 61, 62, 63, 64, 65] mostly focused on coverage of the surveillance field and energy optimisations of the network without explicitly using the application's criticality, which can play a very important role in the network. Different applications can have different criticality levels. Less critical applications like home automation can have fewer constraints but applications of high criticality can have strict requirements. For example, an application of lesser criticality can tolerate the latency but latency can have catastrophic consequences in case of highly critical applications like surveillance(depending on the sensitivity of the area) or intrusion detection systems. So to design a wireless network, taking into account the applications criticality can considerably improve the network performance.

Another important aspect to consider is that all the nodes cannot work in the same manner and hence shall not be treated the same way. Typically, all the nodes in WSNs work together to perform a common task. Hence, fairness of utilisation of the node's resources is not an issue. The nodes we use in our work are equipped with a camera. All of these nodes cannot and should not take images at the same frequency, because this process energy consuming. But to design a successful working model, we have to take the frame capture rate into account. The higher frame capture rate of a camera results in better detection and identification of the events in the network. However, even for extreme critical conditions, it is not realistic to consider all the nodes capturing at their maximum frame capture rate when active.

When huge number of nodes are randomly deployed in an area, there can be nodes which will be monitoring the same region. These overlapping redundant nodes can be used extensively as they are monitoring the same area as some of other nodes.

Therefore, a common approach is to define a group of nodes jointly covering the area to be active while other nodes can sleep to conserve energy. The node's activity periods are scheduled in such a way that the monitoring of the region and the network connectivity are guaranteed. Authors in (([66] [67]) to name a few) have proposed interesting energy-efficient approaches that aim at providing the highest detection quality. However, it is also desirable to have differentiated surveillance services for

various target areas with different degrees of security requirements [68] or to be able to probabilistically support flexible QoS [69] without over-provisioning resources.

In our team's research work, a criticality model is proposed [70] that takes into account the applications criticality and dynamically defines multiple levels of activity corresponding to how many images are captured per second. The capturing and transmitting images are much more energy-consuming in Wireless Image Sensor Networks (WISN), so scheduling efficacy and ability to provide multiple levels of activity is even more important than traditional sensor networks. The authors in [70] present a framework for adaptively scheduling video sensor node's activity while keeping in consideration the specific objectives of the application and the energy constraints without compromising on the network coverage.

The criticality model takes into account the application's criticality and the redundancy of nodes to define the number of frames it should capture per second. The redundancy of nodes is denoted by the number of cover sets, which is defined in the next section.

3.1.1 Notion of cover sets

A simple multiple cover-sets approach is proposed in [70] to manage field of view (FoV) redundancies of randomly deployed sensor nodes. The approach is based on a distributed algorithm that helps each node in organising its neighbours, which cover its FoV (fully or partially), into groups of nodes which cover together a large part of the same FoV. These groups are termed as *cover sets*. Then, depending on the activity of these neighbours, the node can then decide to go to sleep mode or stay active, without any compromise on its FoV coverage.

In figure 3.1, the cone of vision (CoV) of an image sensor is represented by a triangle. The CoV of sensor v is the (*pbc*) triangle and considering nodes v_1, v_2 and v_3, it can be seen that their coverage area jointly covers a large part of CoV of node v. The nodes $\{v_1, v_2, v_3\}$ form together one cover set of node v. If nodes v_4, v_5 and v_6 are added, node v can have more cover sets as multiple combinations are possible as depicted in figure 3.1. Note that v itself is counted as one cover set. Cover sets of node v are represented by $Co(V)$.

One obvious way through which energy can be saved is by putting the nodes, whose sensing area is covered by other nodes, to sleep mode. However, in mission-

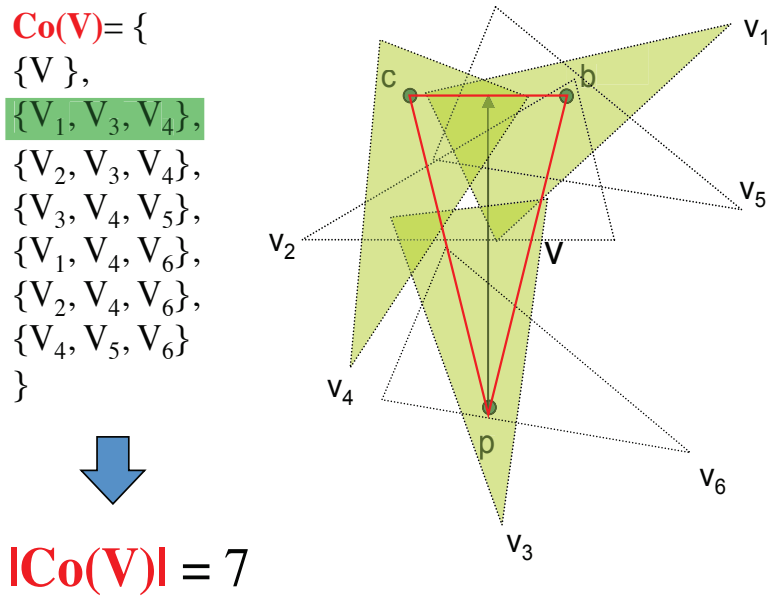

Figure 3.1: Node and it's Coversets

critical applications where it is desirable to increase responsiveness, nodes that possess a high redundancy level (their sensing area are covered many times by other nodes so that they have many cover sets) could rather be more active than other nodes with less redundancy level. In other words, the nodes having more cover sets can be used extensively for surveillance purposes, in comparison to the nodes with less or no cover sets.

In our team's research work [71], the idea was developed that when a node has several covers, it can increase its frame capture rate because if it runs out of energy, it can be replaced by one of its cover sets hence not losing on coverage. Then, depending on the application's criticality, the frame capture rate of nodes (depending on their respective number of cover sets) can vary accordingly: a low criticality level indicates that the application does not require a high frame capture rate while a high criticality level means the application demand high frame capture rate.

3.2 Dynamic Frame Capture Rate

Fixing the frame capture rate of all the video nodes would be naïve. Static frame capture rate is not an ideal solution for WSNs, even in high risk applications. A naïve way of regulating the capture speed proportionally to the dynamic risk level r^0 is shown in 3.2. For example, a high criticality level results in video nodes capturing at near their maximum frame capture rate capability. However, this simple approach have some drawbacks.

- (i): Setting all the nodes to work at lower frame capture rate provides better network lifetime but quality of network surveillance decreases considerably.
- (ii): Setting all the nodes to work at maximum frame capture rate provides excellent surveillance but reduces the network lifetime heavily.
- (iii): A moderate frame capture rate could might have been the solution but in this case, the capabilities of sensors cannot be fully exploited when required.

To fully exploit the video node capabilities, it was proposed that the frame capture rate of a node be defined by the size of its cover-set, based on the multiple cover sets construction described above. The idea behind is that a node with multiple cover sets can work on full capacity and can enhance its capture speed because in case of it being dead due to extensive use, it can be replaced by one of its covers and still have entire network covered.

Additionally, applications can be classified into two categories according to the maximum criticality threshold R^0 (see figure 3.3 that illustrates the case where $R^0 = max$, i.e. 1 and $R^0 = min$, i.e 0):

Class 1 "low criticality", $0 \leq R^0 < 0.5$, does not need a high level surveillance and hence can work at a low frame capture rate. A concave curve can be used to present these kind of applications, where most of values of x are close to zero. Large number of nodes will have a small capture rate. However, the nodes with large number of cover sets can still continue capturing at a high frame capture rate, and can act as a sentry node in the network.

Class 2 "high criticality", $0.5 \leq R^0 \leq 1$, needs nodes to work extensively at a high frame capture rate, as the surveillance level for the critical applications needs to be high. These kind of applications can be represented by a convex curve. Most of the projections of x values are close to max frame capture rate. To have better detection results, most of the nodes will work on high frame capture rate. At the same time,

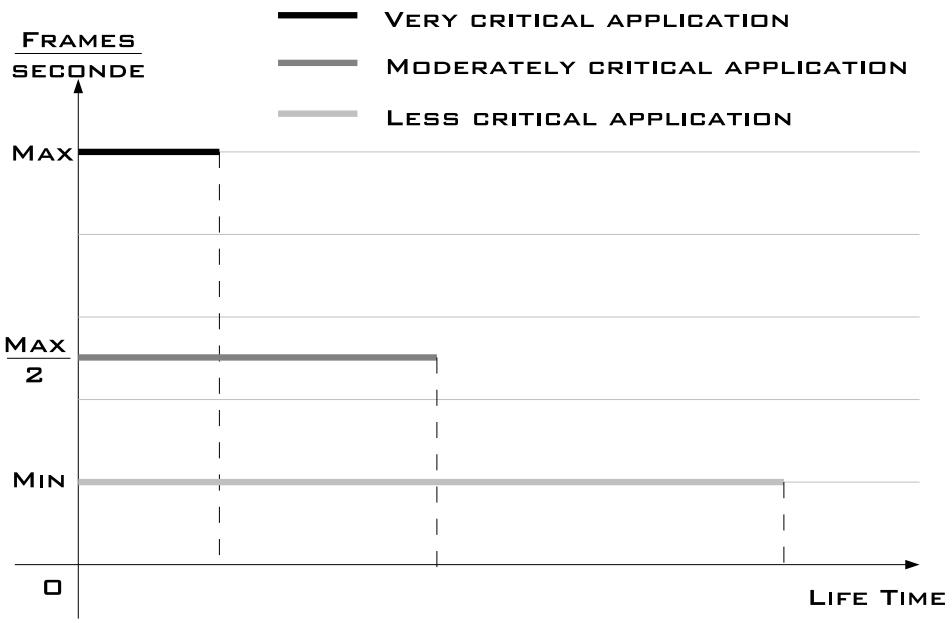

Figure 3.2: Naïve approach.

the nodes with no or very few cover sets will capture at a relatively low capture rate as they don't have backups and hence their energy needs to be conserved.

3.2.1 Behaviour function

Given the desired behaviour described above, a mathematical function was defined in [71], which allows each node i to link its frame capture speed to the cover set cardinality $|Co_i|$ and the criticality level r^0. This function is called BV (BehaViour) function which is an increasing function regardless of the value of r^0: nodes with larger number of cover sets capture faster because if they finish their battery, they can be replaced with ease. But a node with small number of cover sets should preserve its energy because if its battery dies, there is hardly a replacement available which will result in lost coverage of its FoV. At the same time, it is desirable that if the application increases the risk level, the capture speed of the nodes should increase, even for the nodes with small number of cover sets as critical conditions demand quick action and energy saving in those conditions will not serve the purpose.

It was hence proposed to use BV functions expressed by a Cartesian form of the quadratic Bezier curve. Such a curve is defined as follows, see figure 3.4:

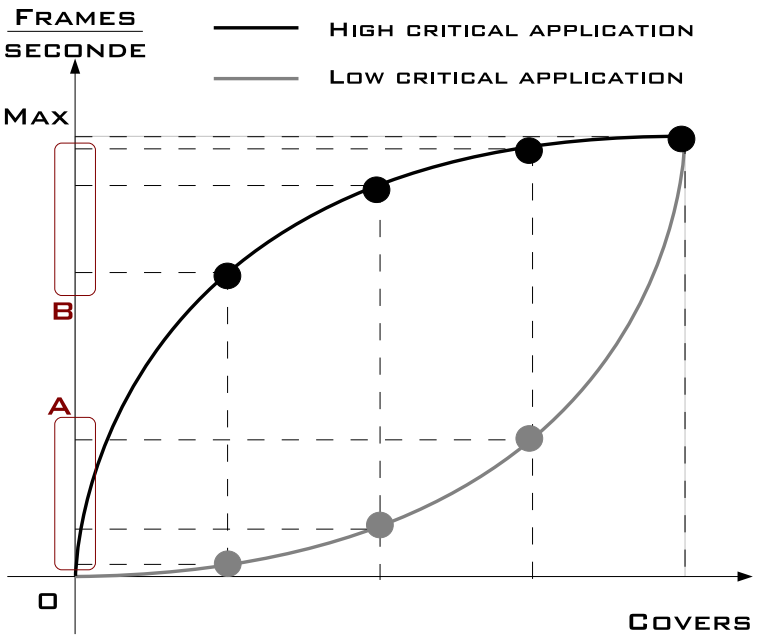

Figure 3.3: Dynamic approach.

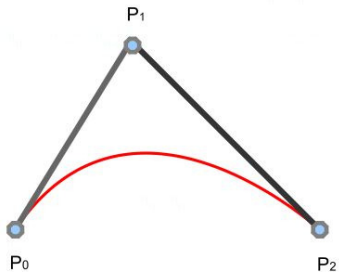

Figure 3.4: The Bezier curve

A quadratic Bezier curve is the path traced by the function B(t), given points P_0, P_1, and P_2.

$$B(t) = (1-t)^2 * P_0 + 2t(1-t) * P_1 + t^2 * P_2. \tag{3.1}$$

The curve passes through three points: the origin $P_0(0,0)$, the behaviour point $P_1(b_x, b_y)$ and the threshold point $P_2(h_x, h_y)$ where h_x is the highest cover cardinality and h_y the maximum frame capture rate determined by the sensor node hardware capabilities (for simplicity we assume that $R^0 = 1$. As illustrated in Figure 3.5, by moving the behavior point P_1 diagonally through the rectangle defined by P_0 and P_2, we are able to adjust the curvature of the Bezier curve, therefore adjusting the

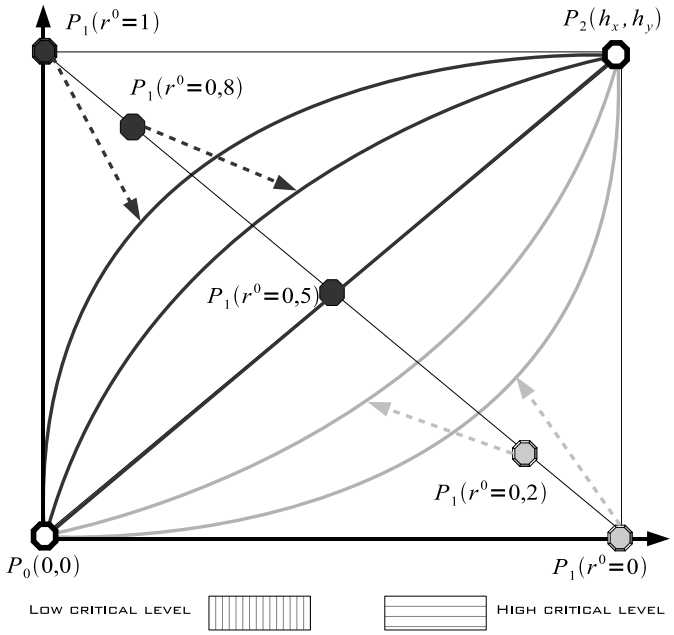

Figure 3.5: The Behavior curve functions

criticality level: according to the position of point P_1 the Bezier curve can move from a convex to a concave form. P_1 therefore defines a criticality level r^0 which is between 0 and 1 (1 being the highest criticality level which requires fast frame capture rate). The advantage of using Bezier curves is that with only three points we can easily define a ready-to-use convex (high criticality) or concave (low criticality) curve.

The BV function describes the application criticality taking $|Co|$ as input on the x axis and returning the corresponding "frame capture rate" on the y axis. To apply the BV function with the Bezier curve, we modify this latter to obtain y as a function of x, instead of taking a temporal variable t as input to compute x and y. We obtain

the following function:

$$\begin{array}{rcl} BV : [0, h_x] & \longrightarrow & [0, h_y] \\ X & \longrightarrow & Y \end{array}$$

$$BV_{P_1,P_2}(X) = \begin{cases} \frac{(h_y-2b_y)}{4b_x^2}X^2 + \frac{b_y}{b_x}X & if\ (h_x - 2b_x = 0) \\ (h_y - 2b_y)(\propto (X))^2 + 2b_y \propto (X), & if\ (h_x - 2b_x \neq 0) \end{cases} \tag{3.2}$$

$$Where\ \propto (X) = \frac{-b_x + \sqrt{{b_x}^2 - 2b_x * X + h_x * X}}{h_x - 2b_x} \wedge \begin{cases} 0 \leq b_x \leq h_x \\ 0 \leq X \leq h_x \\ h_x > 0 \end{cases}$$

3.2.2 Moving the behavior point P_1

As discussed above, the criticality level r^0 of an application is given into the interval $[0, R^0]$. According to this level, we define a criticality function called Cr which operates on the behavior point P_1 to control the BV function curvature.

According to the position of point P_1 the Bezier curve will morph between parabolic and hyperbolic form. As illustrated in figure 3.5 the first and the last points delimit the curve frame. This frame is a rectangle and is defined by the source point $P_0(0, 0)$ and the threshold point $P_2(h_x, h_y)$. The middle point (behavior point) $P_1(b_x, b_y)$ controls the application criticality. We assume that this point can move through the second diagonal of the defined rectangle $b_x = \frac{-h_y}{h_x} * b_y + h_y$.

We define the Cr function as follows, such that varying r^0 between 0 and R^0 gives updated positions for P_1:

$$\begin{array}{rcl} Cr : [0, R^0] & \longrightarrow & [0, h_x] * [0, h_y] \\ r^0 & \longrightarrow & (b_x, b_y) \end{array}$$
$$Cr(r^0) = \begin{cases} b_x = -h_x \times r^0 + h_x \\ b_y = h_y \times r^0 \end{cases} \tag{3.3}$$

Level r^0 is represented by the position of point P_1. If $r^0 = 0$ P_1 will have the coordinate $(h_x, 0)$. If $r^0 = R^0$ P_1 will have the coordinate $(0, h_y)$ where h_x is the highest cover cardinality and h_y the maximum frame capture speed determined by

the value of R^0. Therefore, according to P_1 (i.e. r^0) the BV function can oscillate from concave to convex shape as illustrated in figure 3.5.

3.2.3 Frame capture rate calculation

In previous contributions of our research team using simulations [72, 73], we set the maximum capture rate to 3fps and defined the maximum number of cover-sets to be 12 (nodes with higher number of cover sets will only consider 12 cover sets), i.e. $P_2(h_x, h_y) = (12, 3)$. Therefore, using the criticality model presented above, we can plot for a given criticality level the curve that defines the image sensor capture rate according to its number of cover-sets. Figure 3.6 illustrates such curve for a high criticality level of 0.8. In the graph, the right-most axis shows the time between 2 snapshots in seconds.

The high criticality curve shown in figure 3.6, is convex in nature. The criticality level is fixed at 0.8. The criticality curve shows the calculation of frame capture rate for each node depending on its cover sets and criticality level of the network. As detailed earlier, to avoid loss of coverage area, the nodes with small number of cover sets should preserve energy as it is difficult to replace them. At the same time, the nodes with high number of cover sets should work extensively to enhance the surveillance quality of the network. Figure 3.6 also shows the time a node will take to capture two successive snapshots while the network is operating at criticality level of 0.8. The capture rate of 3 frames per second will result in 1/3 sec delay in second frame capture. This higher frame capture rate results in high energy consumption but better quality of surveillance. It should be kept in mind that high risk application demands excellent surveillance quality standards and saving energy might not be a good idea in highly secure intrusion detection environments, where detection of intrusion and reporting it (propagating alert) has higher priority.

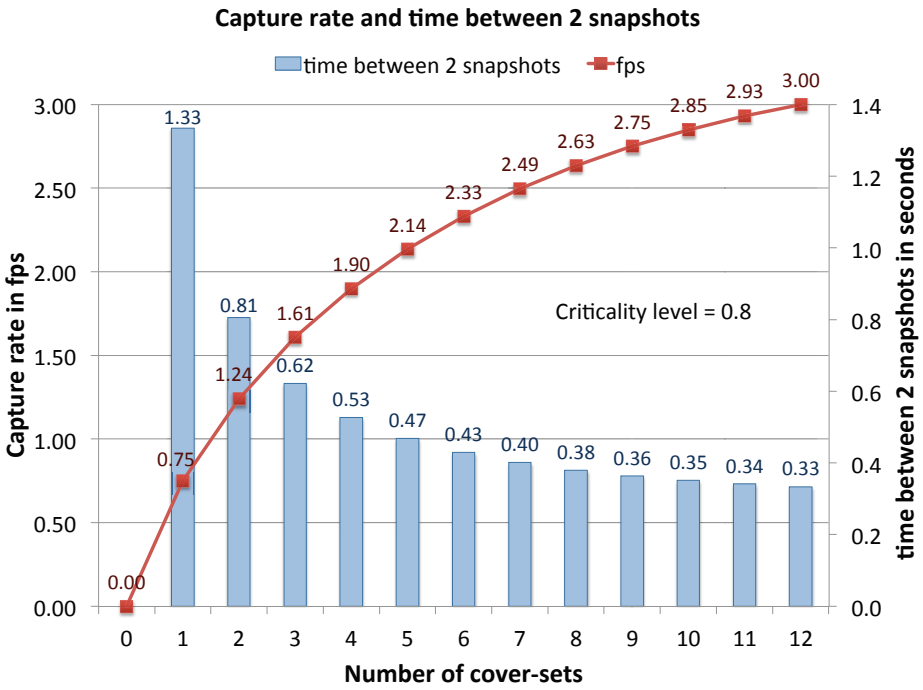

Figure 3.6: Bezier curve for high-criticality level

At the same time, figure 3.6 also shows the nodes having small number of cover sets will have low frame capture rate. A node having a single cover set will have a low capture rate of 0.75 means the node will wait approximately 4 times longer (in comparison to the node having a maximum frame capture rate) to capture another image. The smaller frame capture rate (i.e., the longer time between two snapshots) results in smaller amount of energy consumed but the surveillance quality is compromised. High criticality demands high quality surveillance but it is also important to have the full network coverage, the nodes need to find a balance between surveillance and conservation of energy if they don't have sufficient number of back-ups.

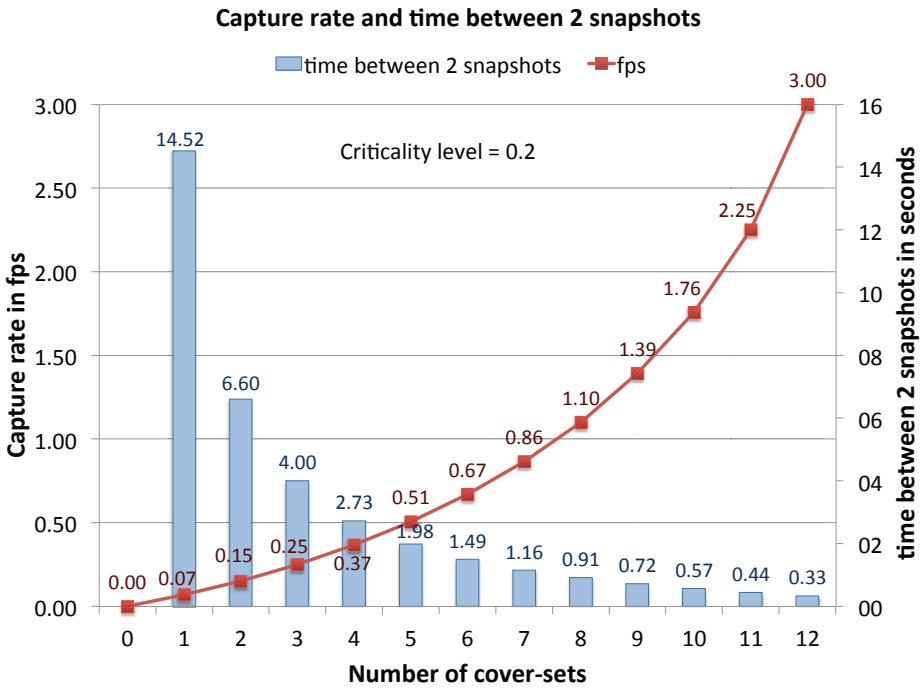

Figure 3.7: Bezier curve for low-criticality level

Similarly, in low risk (low criticality) environment, shown in figure 3.7, the criticality level is represented by a concave curve. The nodes with smaller number of cover sets will capture at extremely low frame capture rates, but the nodes having large number of back-ups will capture fast. Time taken between two snapshots, shown in the right-most axis of 3.7, increases exponentially as the low critical application's can get away with lower quality of surveillance. It can be seen in the figure 3.7 that the node having large number of cover sets still has high capture rate but as the number of cover sets decreases the time between two snapshots increases exponentially. This happens because the lower criticality level of the application allows to conserve energy, to maximise network lifetime.

3.3 Adapting the model to real image sensor hardware

In our team's recent development, we built an image sensor from off-the-shelves components, Arduino boards (both Due and MEGA2560 are supported) with a

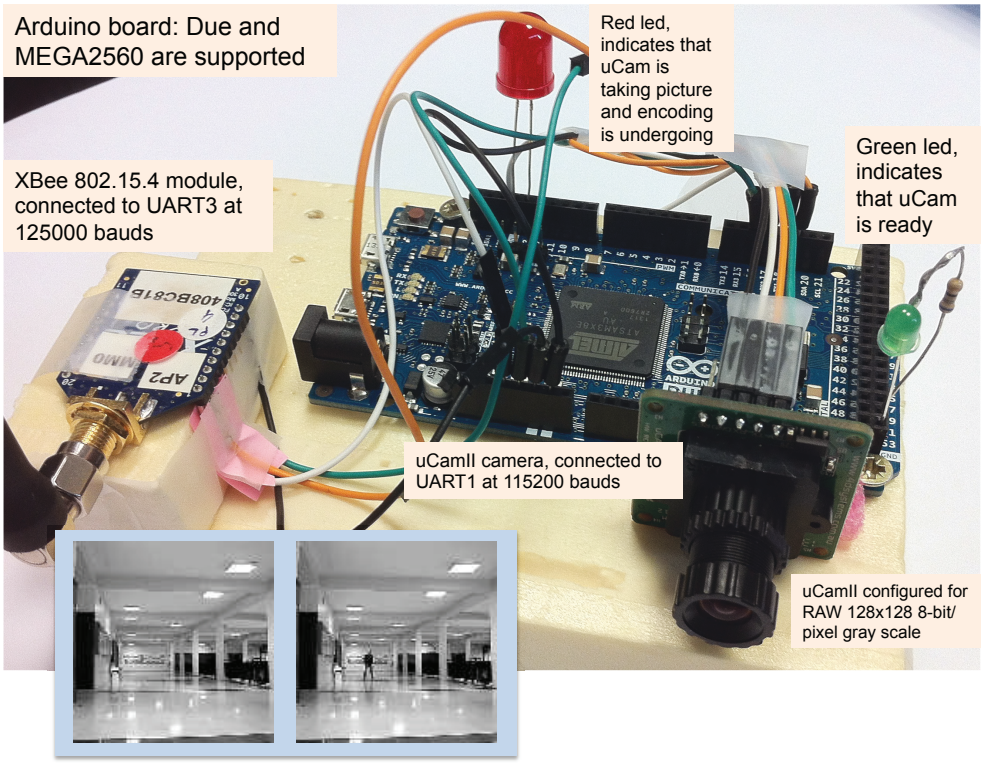

Figure 3.8: Image sensor built with Arduino (Due or MEGA) and uCAM camera

uCamII camera from 4D Systems, capable of taking 128x128 images (8-bit/pixel), performing "simple-differencing" between a reference image and the newly captured one to implement an intrusion detection mechanism [74, 75]. Fig. 3.8 shows the image sensor node and the images produced by the intrusion detection mechanism.

On intrusion detection, an alert message will be broadcasted at 1-hop to alert 1-hop neighbour nodes and the captured image will be encoded with a robust (with regards to packet loss rates) and efficient (for instance, from 128x128=16384 bytes down to 2265 bytes with high visual quality) encoding scheme for transmission towards the sink. With the developed image sensor, the measured time to read the data from the uCam camera and to make the "simple-differencing" procedure is 1.5s. The time to communicate with the uCAM in order to initiate the snapshot and be ready to get the data from the uCAM is about 200ms. Therefore, the time between each snapshot is about 1.7s which gives a maximum capture rate of about 0.58fps. Figures. 3.9 and 3.10 show the adapted criticality model that takes into account the real hardware constraints of the developed image sensor with point at $P_2 = (12, 0.58)$.

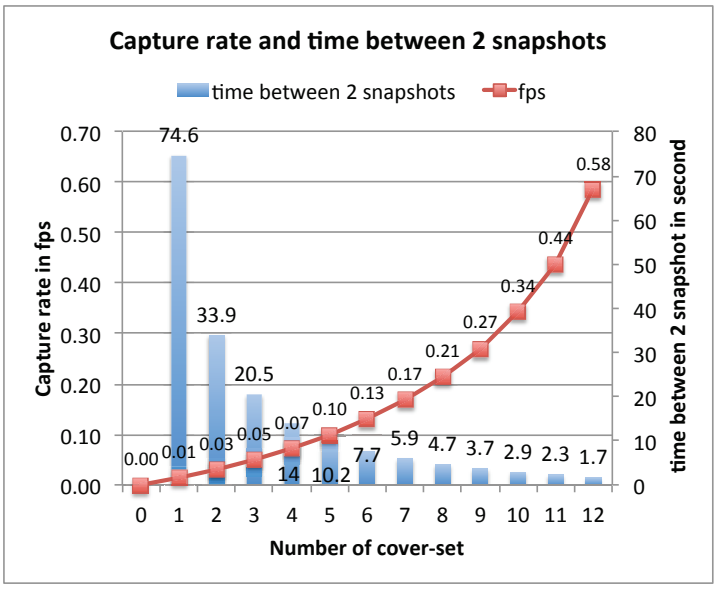

Figure 3.9: Criticality model having criticality level of 0.8 adapted to the image sensor hardware

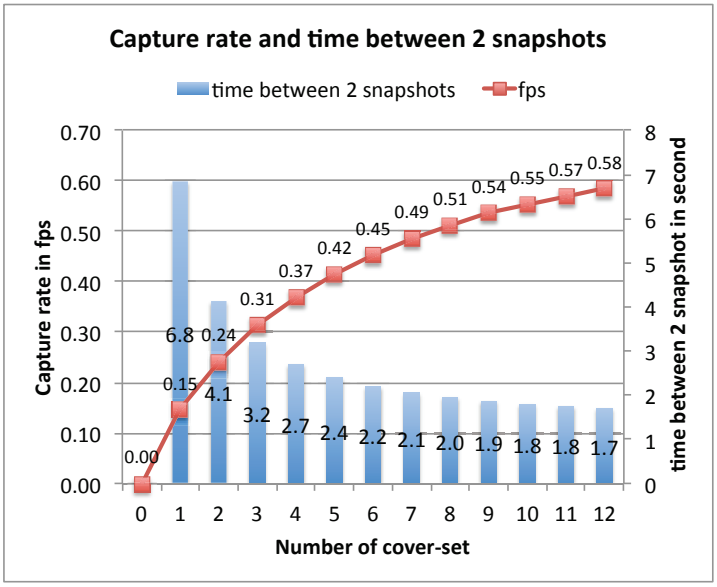

Figure 3.10: Criticality model having criticality level of 0.2 adapted to the image sensor hardware

Figure 3.9 shows the 0.8 case(high criticality level), while figure 3.10 plots for a criticality level of 0.2 (low criticality level). In both images, the right-most axis represents the time between 2 snapshots in seconds.

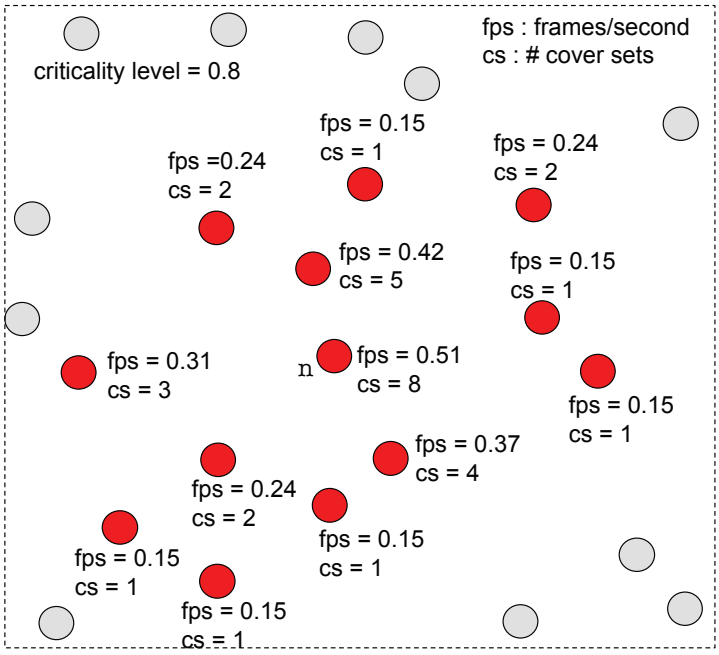

Figure 3.11: Node's frame capture rate under the criticality scheduling with criticality level 0.8.

3.3.1 Image node scheduling and sentry nodes

At startup, every sensor will broadcast its position P and the line of sight $\overrightarrow{V}$ of its camera [1] to its 1-hop neighbours [70], see Figure 3.1. We assume GPS capability, or other means to get the sensor's position. This step can be integrated into a routing initialisation phase with no additional cost. With these local informations, each node v will determine its cover-set ($Co(v)$) and then its frame capture rate using the criticality curves model. The cover-sets are sorted by increasing cardinality order. If needed, sensors could also estimate the percentage of coverage of each cover set. Once the frame capture rate of each image node has been determined, the node's scheduling procedure can start. Figures 3.12 and 3.11 show some nodes (those in red) with their number of cover-sets, cs, and their associated frame capture rate, fps. In the figure 3.12, the criticality level is 0.2. The deployed network can operate with a criticality level of 0.8 if necessary, see figure 3.11 . For instance, upon intrusions alert messages can set the criticality level to a higher value, e.g. 0.8, for a given period of time.

1. We also extended our image sensor platform to support multiple cameras. With 3 cameras equipped with a 116^o lens, we can provide quasi omnidirectional sensing capability, removing the need to determine the camera's line of sight, only the node's position is then required.

With this criticality-based scheduling approach, nodes with high number of cover-set will implicitly behave as sentry nodes by having a higher frame capture rate. Note that there is not a dedicated sentry selection procedure because each node will capture at a rate that only depends on the criticality level, which could be dynamically changed by alert messages for instance, and the size of their own cover-sets. The term of sentry is used only because nodes that will have a higher frame capture rate will be able to detect intrusions with a higher probability, as they are capable of taking more number of images per second. In figure 3.11, we can see that the central node n having 8 cover-sets captures in both cases at a much faster rate than its neighbour nodes. n will be called a sentry node because it will be able to detect intrusions with a higher probability.

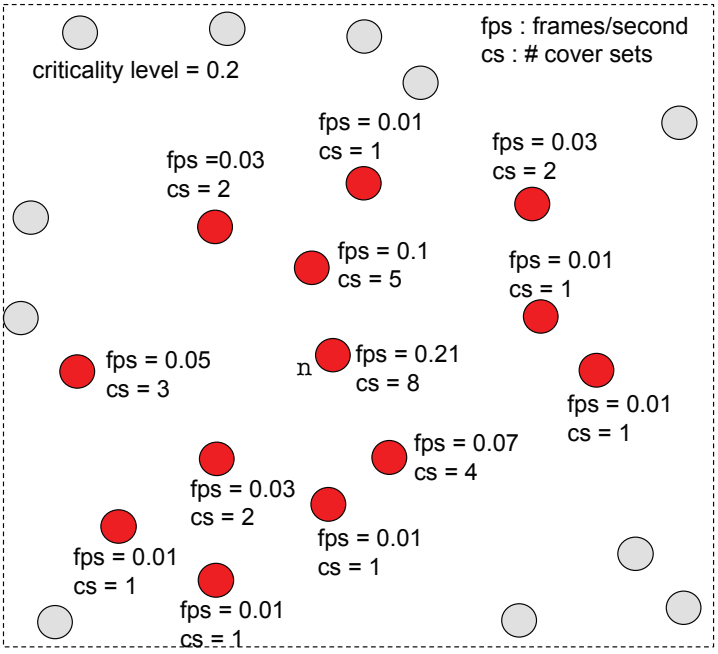

Figure 3.12: Node's frame capture rate under the criticality scheduling with criticality level of 0.2.

3.4 Conclusions

We explained a criticality model, which was developed in the earlier research work of our team. This criticality model uses number of cover sets of the nodes and the applications criticality level to calculate the frame capture rate of the node's camera. This frame capturing speed can be used to select the sentry nodes, which have highest

frame capture rate in its 1-hop neighbourhood. Highest frame capture rate results in highest probability of detection in the vicinity. This detected event must be notified to the neighbours of the nodes and eventually to the sink.

The motivation of the work, we present in this dissertation, is to allow fast reception of alert messages from the node which detected the intrusion. For that purpose, we propose an adaptive listening period, making each neighbour node of a sentry node (a node with a high capture rate) ready to receive the alert message so that alert can be propagated and can be forwarded to the sink. We will describe our contribution in the next chapters.

Chapter 4

Criticality Adaptive MAC protocol

Contents

We present a Criticality-based Adaptive MAC protocol in this chapter, which is designed with critical surveillance applications in mind. The criticality model presented in the chapter 3 is the basis of research work presented in this chapter. Our approach takes the application's criticality and the node's redundancy to calculate the activity period of the nodes. The frame capture rates are compared and the nodes with highest frame capture rates in 1-hop neighbourhood are chosen as sentry nodes. These sentry nodes have high probability of detection, so their neighbour node's activity period should be designed in a way that it is active and ready to communicate with the sentry. At the end of this chapter, we present the simulation results of the proposed low latency, energy efficient MAC protocol.

4.1 Criticality-based Adaptive MAC Protocol (CAMP)

The criticality based adaptive MAC protocol assumes random deployment of large number of image sensor nodes. These camera equipped sensor nodes are capable of covering their unidirectional field of view (FoV). Their redundancy (as explained in chapter 3) helps in improvement of the network surveillance. Once an intrusion detection takes place, the nodes should send an alert to the sink. In critical applications, the propagation of this alert plays an extremely important role. The network needs to relay this important information about intrusion to sink with minimum delay. Our goal here is to minimise latency for this alert propagation and maximising the network lifetime in critical applications.

4.1.1 CAMP's levels of synchronisation

With our proposed approach, all the nodes are capable of detecting an intrusion but according to the previously described criticality-based scheduling method, some nodes having higher redundancy will capture images at high capture rates, which in turn means they have a higher probability of detecting an intrusion or any changes in the environment (under the assumption that such events occur uniformly in the covered area) and can therefore act as sentry nodes in the network. On detecting an intrusion, a node will generate an alert message which has to be propagated in the network with minimum latency. As illustrated in Figure 4.1, reducing the alert message latency upon intrusion detection serves 2 purposes:

1. To alert 1-hop neighbour nodes so that local actions can be performed: The neighbour nodes can then increase the criticality level and send additional pictures for disambiguation, for instance.
2. To propagate the alert to the remote sink in a multi-hop manner.

We can therefore distinguish 2 distinct synchronisation areas.

- The first area, referred to as 1-hop area, involves the node which detected the intrusion (most likely a sentry node) and its 1-hop neighbours.
- The second area, referred to as k-hop area, contains all the intermediate nodes from a 1-hop neighbour to the remote sink.

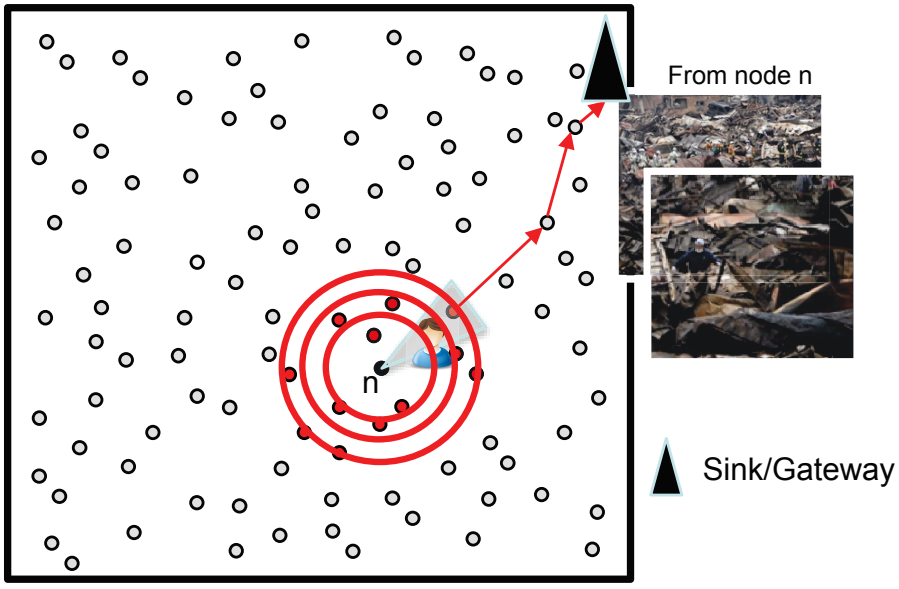

Figure 4.1: Mission-critical intrusion detection system

Our CAMP proposition mainly addresses the 1-hop area synchronisation issue, i.e. from the node which detected the intrusion and its 1-hop neighbour nodes. The main objective of this synchronisation level is to make neighbour nodes receive the alert message as soon as possible. We assume that all nodes have a radio duty-cycled behaviour where the radio module is put to sleep for some time, and then woken up to listen for other nodes wanting to communicate with it, e.g. transmission of an alert message for instance. Figure 4.2 shows how, at the application layer level, the frame capture rate of a node can be determined based on an application criticality level and the node's number of cover-sets using the criticality scheduling approach reviewed in chapter 3. The node's activity at the application level can actually be independent from the radio activity which is the focus of this work. While a node which has detected an intrusion can simply switch on its radio when needed as shown in Figure 4.2, it is more difficult to set the listening period of neighbour nodes that will have to relay the alert message. CAMP's objective is to maximise the probability for a neighbour node to be in the listening mode when the alert is sent. So that the alert gets propagated in the network and further action can be taken in quick time.

For instance, Figure 4.2 shows an undesired case of neighbour node that is out of sync from the node detecting the intrusion, thus delaying the alert message propagation. In order to avoid difficult and costly synchronisation mechanisms for the 1-hop area, CAMP uses a probabilistic approach where a neighbour node will set its

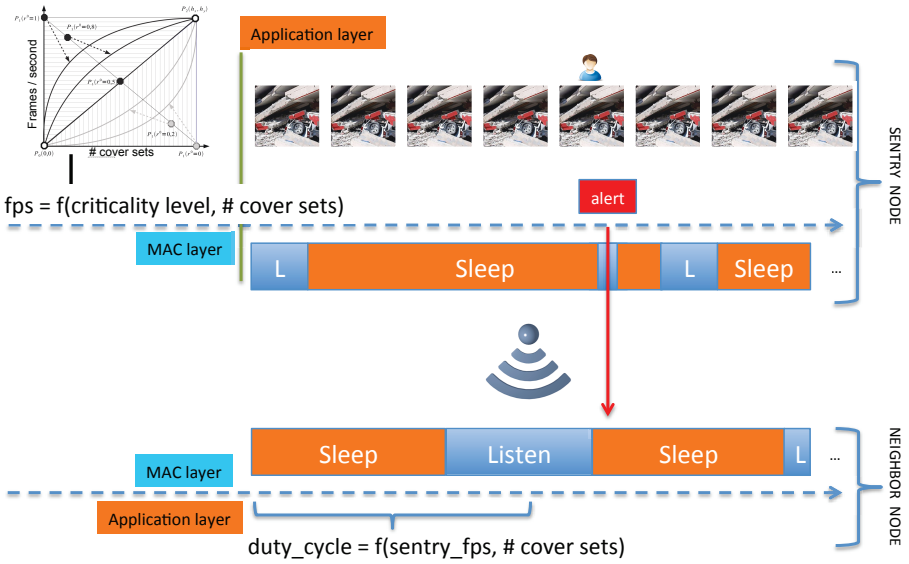

Figure 4.2: Active and Sleep periods of the MAC layer

listening period according to a sentry's frame capture rate and its own redundancy level (i.e. number of cover-sets).

Our contribution works in 2 phases.

1. The first phase is to determine for each image sensor node its associated sentry node, i.e. the image node in its neighbourhood with the highest frame capture rate. A node with an associated sentry node will be called a *follower node*.
2. Then, in a second phase, we adapt the follower node's listening period to increase its responsiveness in case of issued alerts, so that it is ready to receive and quickly relay data to the sink.

CAMP does not address explicitly the k-hop area issues to synchronise active period for multi-hop transmissions. However, as neighbours of a sentry node have well defined listen/sleep schedules, it is possible for CAMP to use existing propositions such as DW-MAC [54] or AS-MAC [51] to synchronise and share these wakeup schedules in the k-hop area.

4.1.2 Sentry selection phase

In this first phase, after having determined its cover-set and frame capture rate, every node broadcasts this information at 1-hop. The nodes keep this information

in a table. Once all the nodes have finished broadcasting, each node can identify the node with the highest capture rate in its 1-hop vicinity. That node is termed as *sentry node* or *master node* in its 1-hop neighbourhood. Remind that the capture rate of a node is calculated using the criticality model described in chapter 3. Figure 4.3 depicts the end of phase 1 where a sentry node (the black node with the highest frame capture rate) has been identified and associated to follower nodes in a given neighbourhood.

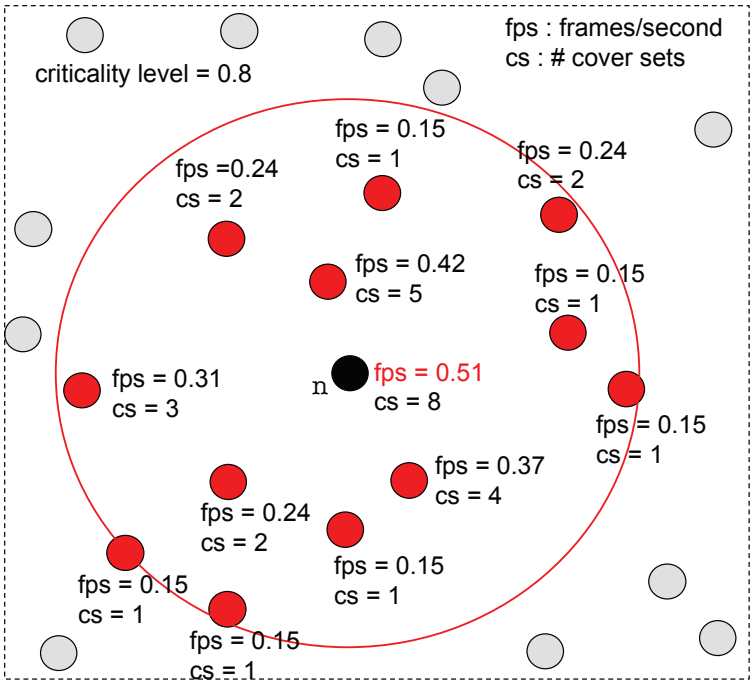

Figure 4.3: Sentry node selection at the end of phase 1

Once the sentry node has been identified and its frame capture rate known, the second phase is to set the follower node's radio duty-cycling pattern. We propose that the listening period of the follower nodes be calculated in relation to the frame capture rate of their sentry node. However, another important factor to consider is the follower node's redundancy level. When a node has several cover sets, if it runs out of energy it can easily be replaced by one of its cover sets without losing the network coverage. Therefore this follower node can afford to have a duty cycling pattern with longer listening time. This is the purpose of phase 2 described in the next paragraphs in more details.

4.1.3 Determining duty-cycling pattern

If a follower node has a small number of cover sets then it is preferable that it preserve its energy because there is hardly any replacement available, in case its energy gets depleted. Hence, depending on the size of cover sets, each follower node of a given sentry node may have different listening time period.

We propose that the duty-cycling pattern of a follower node should follow the convex/concave criticality model described in chapter 3, in order to maintain the properties of criticality-based scheduling. However, the y axis will now give the corresponding duty-cycle value (between 0 and 1, corresponding to the listening period ratio) based on the cardinality of the cover sets of the follower node itself, expressed on the x axis, and the sentry node's frame capture rate, see bottom part of Figure 4.4.

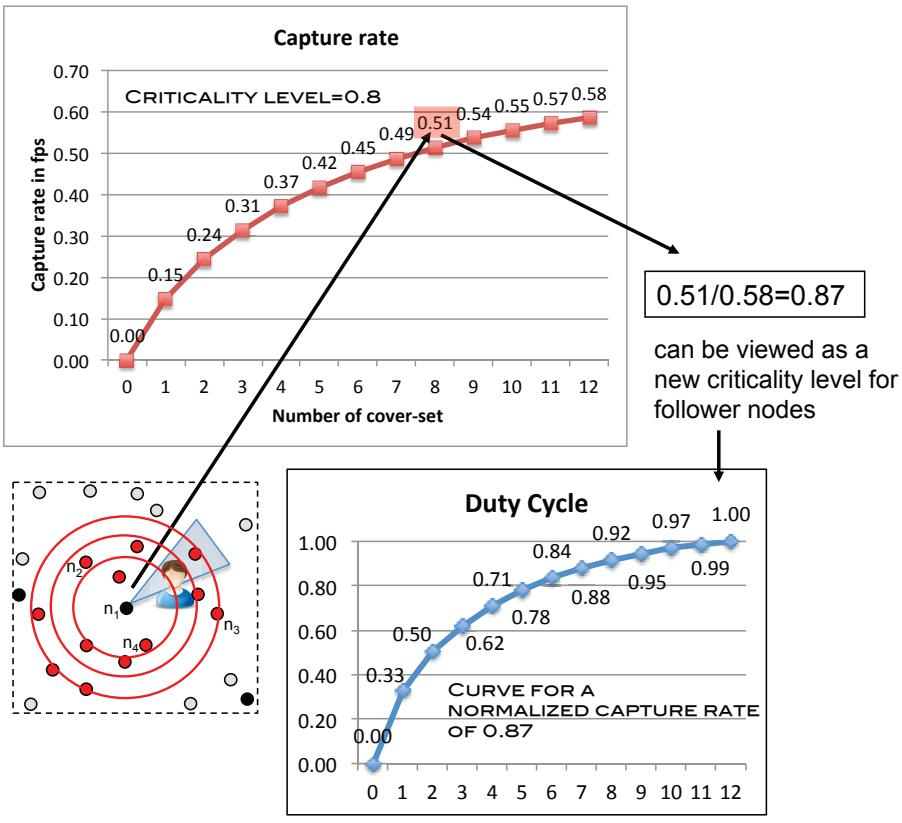

Figure 4.4: Criticality curve example

Actually, the sentry node's capture rate value is normalised against the maximum frame capture rate and is used as a new criticality level for the node, whose duty-

cycle value is being calculated. In this duty-cycle model, we therefore now have $P_2(h_x, h_y) = (12, 1)$: maximum considered number of cover sets is 12 and duty-cycle ratio is between 0 and 1. The concave curves will represent the smallest capture rate (normalised) where most duty-cycle values will give smaller listening periods, i.e. values are near to zero, unless if a node has high number of cover sets in which case it will have a larger listening period. Similarly, the convex curves represent the highest capture rate (normalised). In this case the duty-cycle values calculated for the follower nodes will be longer. Follower nodes with larger number of cover sets will have duty-cycle values close to maximum duty cycle value.

Figure 4.4 illustrates the entire duty-cycling computation process at follower nodes. In this example, with a criticality level of 0.8, a node having 8 cover sets will capture at a frame rate of 0.51 fps. Assuming that this node is selected as a sentry node, then its neighbours will use its capture rate to compute their own duty cycle value.

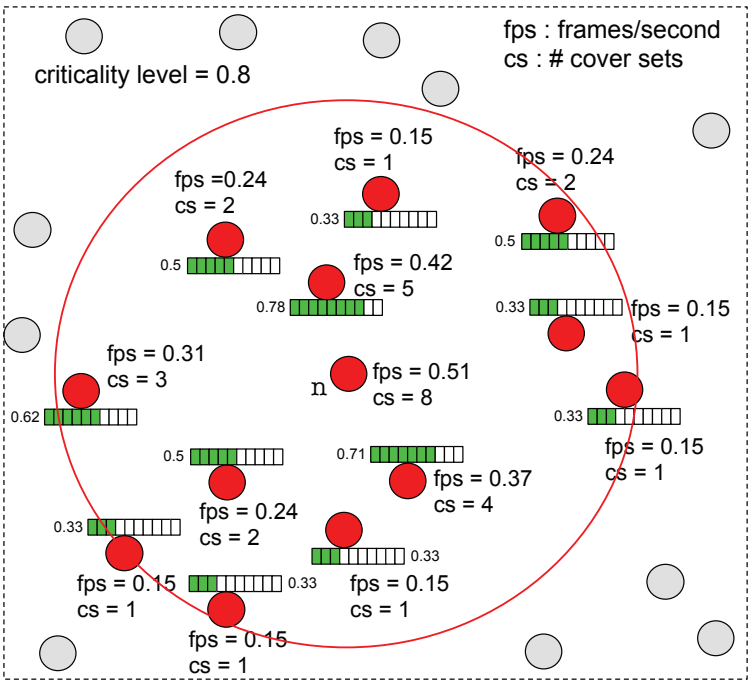

Figure 4.5: Duty cycle of follower nodes

We can see in Figure 4.4 how the capture rate is normalised (against the maximum capture rate defined by hardware constraints, 0.58 fps in the example) and can be used as a new criticality level (here we have 0.87) for computing the duty cycle value

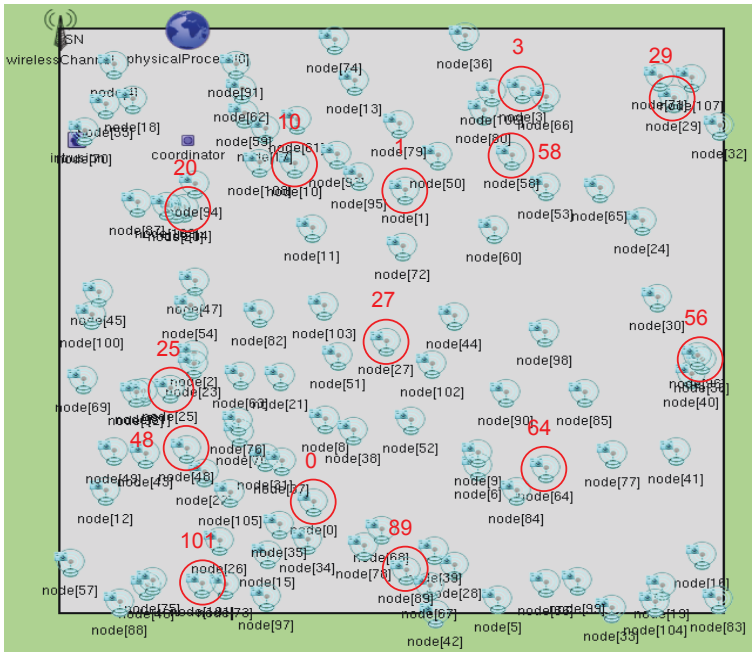

Figure 4.6: Snapshot of the Omnet++ Simulator

at a follower node, taking it's number of cover sets into account. Here, this new criticality level gives a curve which is more convex, i.e. most values on the y axis will be in the upper half of the curve even with smaller number of cover sets, which means longer duty-cycle values for follower nodes.

Figure 4.5 illustrates the duty-cycle pattern of follower nodes where the green bars represent the duty-cycle value between 0 and 1.

4.2 Simulation Results

4.2.1 Simulation settings

To evaluate our approach we conducted a series of simulations using the OMNET++/ Castalia simulator. For these set of experiments, we randomly deployed 110 sensor nodes in a 400mx400m area as illustrated by the OMNET++'s screenshot in Figure 4.6.

Each sensor node captures with a given number of frames per second (between 0 fps and 0.58 fps). We set the maximum number of cover sets for a node to be 12:

nodes with higher number of cover sets will only consider 12 cover sets. Minimum duty cycle is fixed at 0.1, the cycle duration is 3000ms and the criticality level is set at 0.8. Random intrusions are introduced in the simulation model and nodes can detect an intrusion if the intruder is covered by their field of view at the time of the image capture. Upon intrusion detection, a node will broadcast an alert message. All simulation parameters are summarised in Table 4.1. The simulation time is 500s.

Table 4.1: Simulation model parameters

all cases	
# of nodes	110
field size	400 m x 400 m
maximum # of cover-sets	12
maximum frame capture rate	0.58 fps
transmission rate	250 $Kbps$
cycle duration	3000 ms
baseline sensor power	6 mW
Tx power level	0 dBm
Tx power	57.42 mW
Rx power	62 mW
AdaptiveMac only	
criticality level	0.8
minimum duty cycle	0.1

As explained previously, at startup each node determines its cover-set and frame capture rate. Then every node broadcasts this information. Once all the nodes have finished broadcasting, each node can identify the node with the highest capture rate in its neighbourhood. That node will be its associated sentry node. Out of 110 nodes, 51 have been selected as sentry/master nodes. However, a large number of them do capture at a low rate because they have few cover sets. Therefore they will not impact much on the intrusion detection nor on the duty-cycle value of their followers. The total number of alert messages sent in the network upon intrusion is 1425. 1203 have been sent by sentry nodes and only 222 have been sent by non-sentry nodes. In Figure 4.6 we show with red circles 14 sentry nodes (out of the 51 sentries) with high capture rates that sent in total more than 87% (1070) of the total number of alerts. We will focus on these nodes to show the detailed results of the simulations.

After the initialisation phase, all nodes will start their MAC duty-cycling behaviour. With our MAC proposition a follower node's duty cycle values varies de-

pending on the capture rate of its sentry node and on its number of cover sets. In Figure 4.7, the duty cycle values of all nodes in our simulation scenario are shown after calculation from the criticality model in descending order. The x-axis is not the node index or ID, just a node count. A sentry node does not actually need to keep its radio active for a long period of time, hence its duty cycle can be kept at minimum, i.e. 0.1 (the duty-cycle value of the previous 14 sentries are shown in red in Figure 4.7).

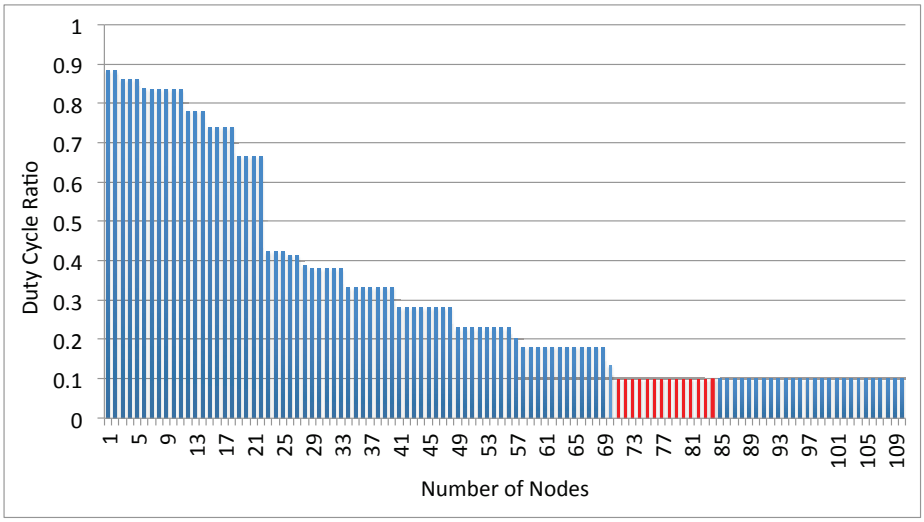

Figure 4.7: Duty cycle lengths of all the nodes

4.2.2 Sentry node statistics

For the purpose of determining the performance of our approach, we designed in the simulation model all the follower nodes to respond with an acknowledgment message on reception of an alert, confirming the reception of the alert. The responses received from the followers confirm that the alert was successfully propagated. On the other hand, no response from the followers goes on to show that all the follower nodes were on the sleep mode and they did not received the alert message. Now, this means that the alert was sent but none of the follower nodes was available to hear that communication, alert was not propagated and it was not relayed to the sink, which can have severe consequences for applications of critical nature. The total number of alerts sent by all the 14 sentry nodes identified in Figure 4.6 is 1070 and Figure 4.8 summarises the important statistics of the 14 main sentry nodes.

Sentry node ID	# neighbors	# followers	# cover sets	# alerts sent	# alerts confirmed	% confirmed alerts	# alerts missed	% missed alerts	Total responses received	% 1 follower	% 2 followers	% 3 followers	% 4+ followers
0	5	3	7	75	60	80.00%	15	20.00%	88	44.00%	37.33%	0.00%	0.00%
1	3	2	8	74	65	87.84%	9	12.16%	124	36.49%	29.73%	20.27%	2.70%
3	5	4	10	80	77	96.25%	3	3.75%	202	12.50%	36.25%	35.00%	15.00%
10	5	5	8	78	75	96.15%	3	3.85%	105	20.51%	28.21%	20.51%	26.92%
20	4	4	9	74	67	90.54%	7	9.46%	177	8.11%	39.19%	25.68%	18.92%
25	8	3	7	80	73	91.25%	7	8.75%	111	51.25%	28.75%	11.25%	0.00%
27	3	3	10	75	52	69.33%	23	30.67%	50	69.33%	0.00%	0.00%	0.00%
29	3	3	9	73	59	80.82%	14	19.18%	74	60.27%	20.55%	0.00%	0.00%
48	5	3	8	81	78	96.30%	3	3.70%	135	39.51%	46.91%	11.11%	0.00%
56	3	3	7	73	62	84.93%	11	15.07%	95	43.84%	36.99%	4.11%	0.00%
58	4	1	8	83	73	87.95%	10	12.05%	88	72.29%	16.87%	0.00%	0.00%
64	4	2	10	76	73	96.05%	3	3.95%	105	56.58%	40.79%	0.00%	0.00%
89	6	4	8	76	57	75.00%	19	25.00%	122	13.16%	38.16%	23.68%	0.00%
101	4	4	7	72	66	91.67%	6	8.33%	81	75.00%	12.50%	4.17%	0.00%
				1070	937		133		1557				

Figure 4.8: Statistics for the 14 main sentry nodes

Except for sentry nodes 27 and 89, all sentries have a percentage of confirmed alerts above 80%. 8 sentries have a percentage of confirmed alerts close to 90% or above. As the total number of acknowledgment is greater than the number of sent alerts, in most cases, more than 1 follower acknowledges an alert. The last 4 right-side columns in the table shown in figure 4.8 shows the percentage of alerts that was acknowledged by 1, 2, 3, or 4 followers. For instance, if we look at sentry node 10 with all 5 neighbours (nodes 11, 17 and 108 with 1 cover set, node 62 with 5 cover sets and node 93 with 6 cover sets) being its followers, more than 20% of the alerts sent have been acknowledged by only 1 follower. More than 48% of the alerts sent have been acknowledged by 1 follower or 2 followers. Figure 4.9 shows for sentry node 10 the number of acknowledgements received for each alert message (which is sent upon an intrusion detection). The total number of alerts sent is 78 as shown previously by the table in figure 4.8.

Figure 4.10 shows the number of sentries (maximum is 14) whose alert messages have been acknowledged by 1, 2, 3 or 4 followers for a given minimum percentage of acknowledged alert messages. For instance, we can see that the case where only 1 follower has responded to an alert message is the most frequent.

4.2.3 Comparison with a static duty cycle approach

We then compared our approach with a traditional static duty-cycled MAC protocol with varied duty cycle values: 0.5, 0.6, 0.7 and 0.8. For instance, a cycle duration

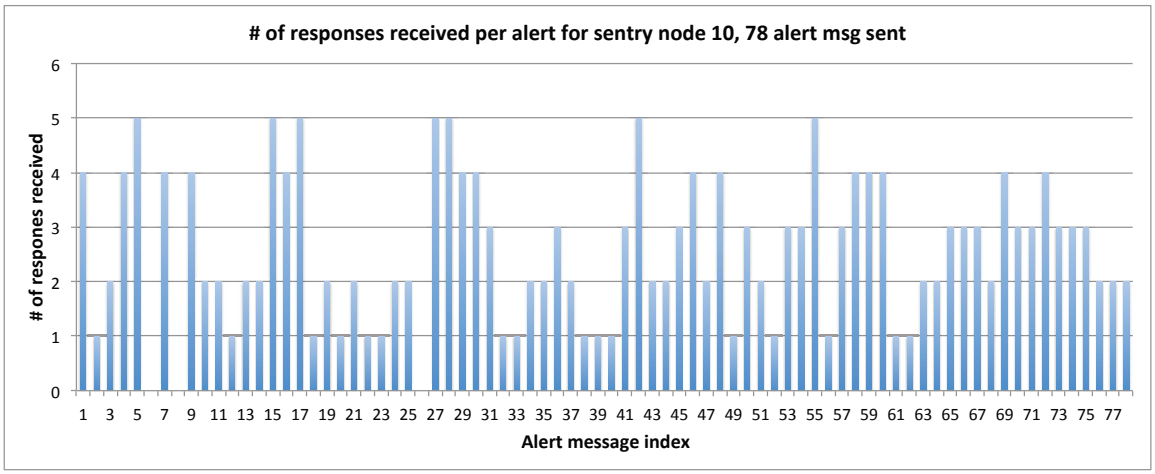

Figure 4.9: Number of acknowledgements received per intrusion for sentry node 10

	% 1 follower	% 2 followers	% 3 followers	% 4 followers
5%+	14	13	7	2
10%+	13	13	6	2
20%+	11	11	4	0
30%+	11	8	1	0
40%+	8	2	0	0
50%+	6	0	0	0
60%+	4	0	0	0
70%+	2	0	0	0

Figure 4.10: Number of sentries whose alerts are acknowledged by 1, 2, 3 or 4 followers

of 3s with a duty-cycle value of 0.8 will give 2.4s of radio activity (e.g. can receive) followed by a 0.6s period of inactivity (e.g. can not receive). Figure 4.11 shows the total number of alerts sent by sentry nodes to which no responses were received (missed alert message), i.e. the number of alerts which were not propagated through the network.

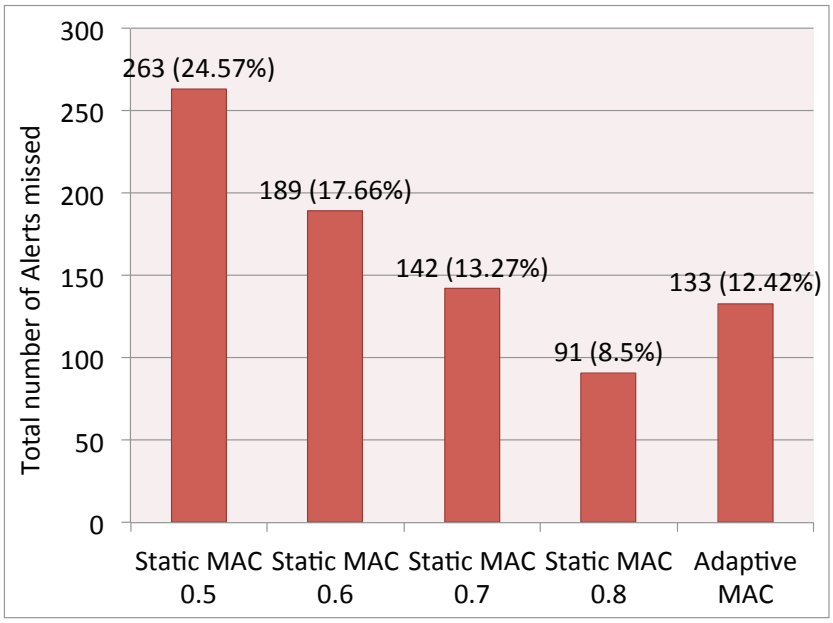

Figure 4.11: Number of missed alert messages

Fig. 4.12 shows the percentage of received and successfully propagated alert messages in the network. We see in Fig. 4.11 and 4.12 that the results of the MAC protocol proposed in this paper, are second only to a static MAC with 80% duty cycle.

Criticality adaptive MAC proposed in this paper shows better results in comparison to duty cycled static MAC with duty cycle of 0.7. Actually, as the number of nodes working at 0.7 duty-cycle or above is small, the results shown in Figures 4.11 and 4.12 clearly illustrate the benefit of our criticality adaptive MAC approach: fewer nodes working on high duty cycle values but better responsiveness of the network.

Figure 4.13 shows the comparison of total energy consumption of all the nodes in the network in Joules, when using the energy model of Castalia (see Table 4.1).

In the figure 4.13, it can be seen that our criticality adaptive MAC protocol consumed 48% less energy in comparison to a static MAC with duty cycle of 0.8. Which means the network lifetime is almost doubled in comparison to static MAC 0.8. For a static MAC with duty cycle of 0.7, the energy saved was around 44%, and the corresponding values were 38% and 32% respectively for static MAC 0.6 and static MAC 0.5. Taking the global energy consumption of the network, Figure 4.14 shows the energy consumed per successfully propagated alert message for the static and adaptive MAC protocols.

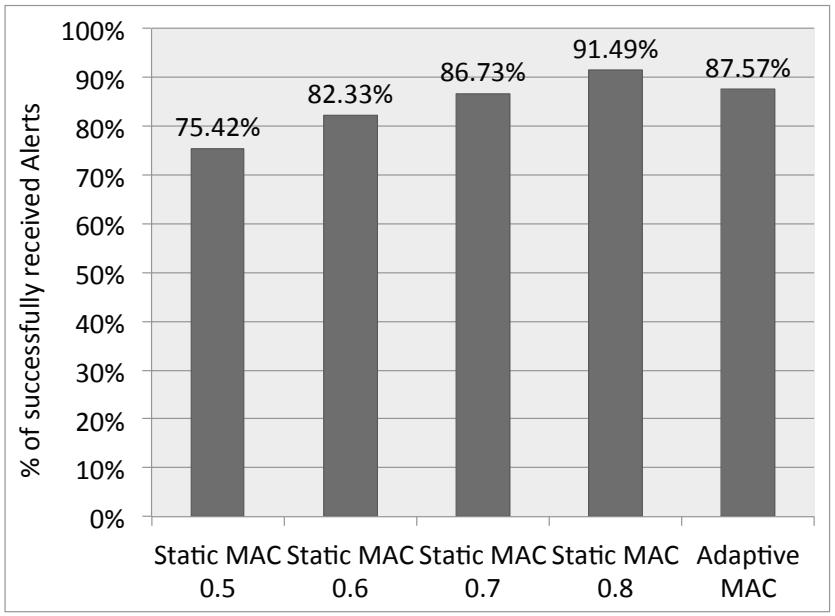

Figure 4.12: Received and successfully propagated alert messages

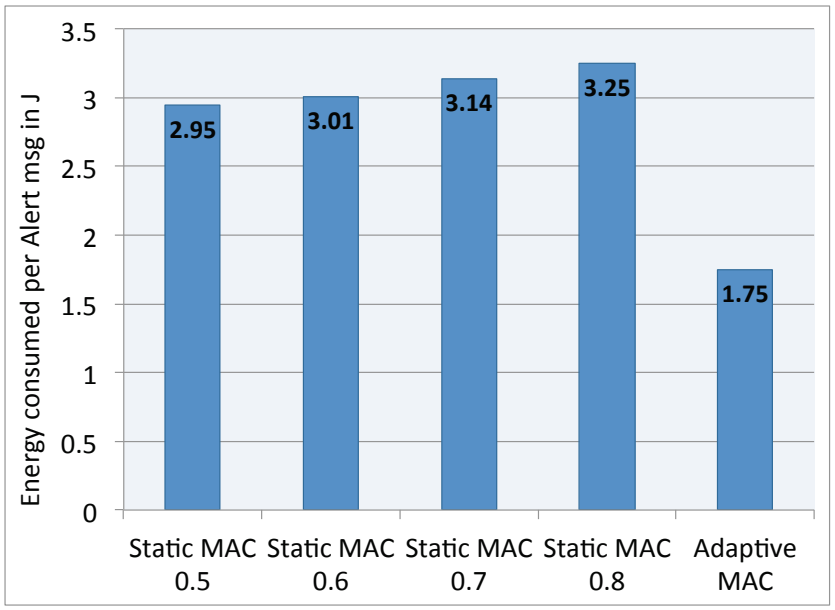

Figure 4.14: Comparison of energy consumed per alert message

We see in Figure 4.14 that the adaptive MAC approach gives significantly better results in comparison to static MAC with different duty cycle ratios. The energy is efficiently utilised to increase the network lifetime.

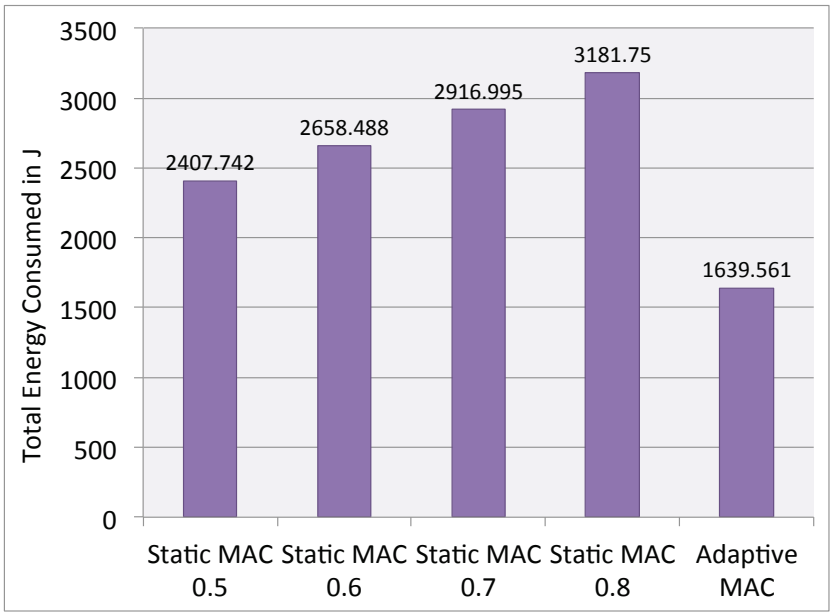

Figure 4.13: Comparison of total consumed energy

4.2.4 Varying the cycle duration

The cycle duration was set so far to 3000ms. We varied the cycle duration to have values of 200ms, 500ms, 1s, 3s, 5s and 10s. Once again, we focus on sentry node 10 which has all its 5 neighbours (nodes 11, 17 and 108 with 1 cover set, node 62 with 5 cover sets and node 93 with 6 cover sets) being its followers. We extended the simulation time to have at least 500 alert messages sent by sentry node 10. Figure 4.15 shows for sentry node 10 the distribution of correctly received alert messages by its follower nodes 10 under our criticality adaptive MAC protocol when the cycle duration is varied. We can see that the cycle duration has little impact on the detection quality. We can however observe that a small cycle duration, i.e. less that 1s for instance, decreases the probability of having a large number of followers active at the same time, i.e. above 3 followers for instance.

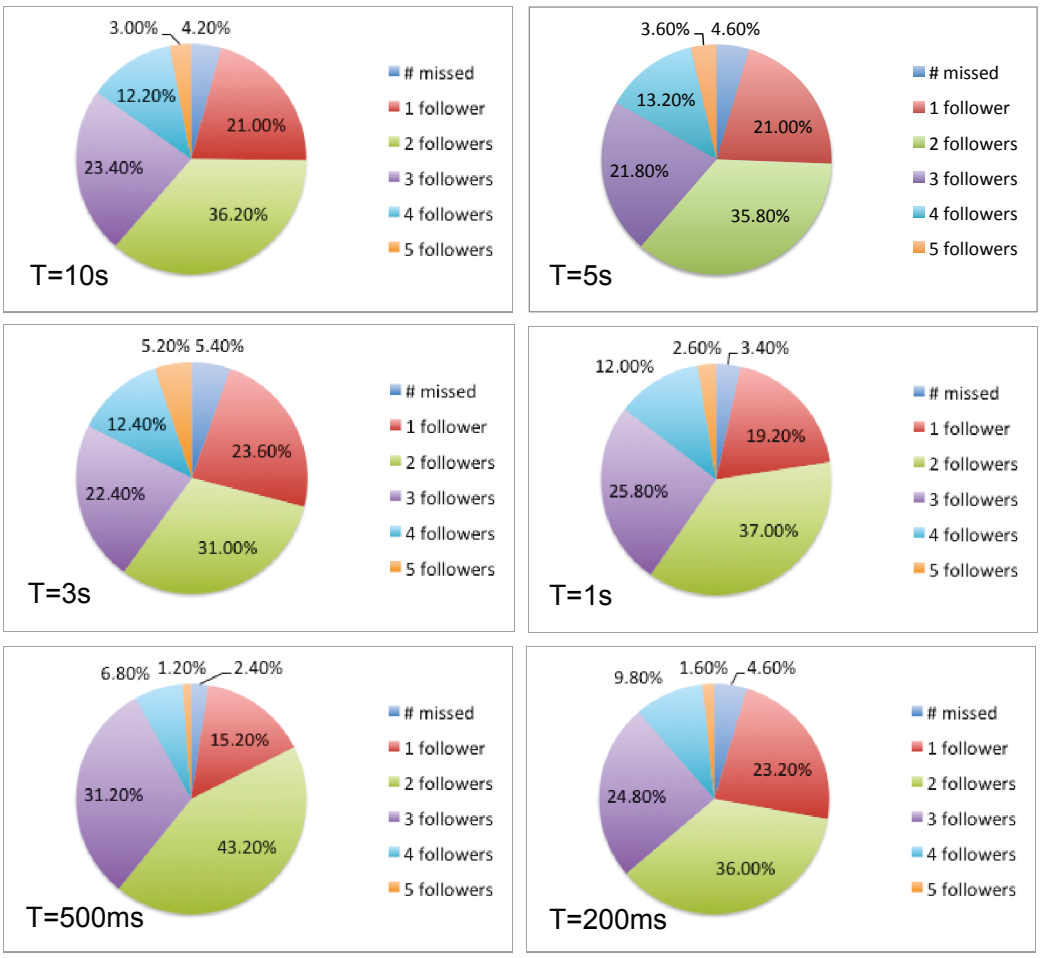

Figure 4.15: Impact of cycle duration

4.3 Discussions

4.3.1 1 follower vs multi-follower nodes

Our proposed approach defines an active period for all follower nodes of a given sentry to increase the probability of being able to receive and propagate alert messages from the sentry. In the best case, only 1 follower node needs to be active when the sentry sends the alert. One approach could be to determine, among all the sentry's neighbours, a dedicated node for this purpose, possibly the node that is the next-hop to the sink. Similar to a cluster head in cluster-based routing approaches, this solution however adds overheads of determining such a dedicated node and, most

importantly, is less robust in case of node failure. Coordinated wakeup approaches such as AS-MAC [51] are also less robust in case of node failure. This is the reason why we did not consider these methods in the context of our application.

4.3.2 Adapting LPL parameters

As mentioned previously, some MAC protocols use Low-Power Listening (LPL) features to decrease the cost of maintaining the radio in an active listening state (B-MAC [20] and X-MAC [23] to name a few). Under LPL operation, a node periodically sleeps and wakes up only to sample the channel (so-called LPL phase). At the sender side, a preamble is sent prior to data transmission. Such preamble must be sufficiently long to be detected by receiver nodes when they wake up and sample the channel. Many improvements can reduce the preamble duration with early acknowledgments from receivers, or reduce the overhearing issue with the identity of the target receiver in the preamble, or adapting the inter-listening interval [76]. Even though LPL needs dedicated hardware support to be energy efficient, under LPL features our approach can be an effective method for determining the time between periodic channel sampling at the receiver nodes (e.g. the follower nodes): the higher the frame capture rate of the associated sentry, the smaller the time between 2 channel sampling. Under the assumption that early ack can be used, a smaller channel sampling interval means that the preamble duration at the sender will practically be greatly reduced.

4.3.3 Duty-cycling patterns

There are also several ways to define duty-cycling behaviour. Figure 4.16(a) shows the basic behaviour where a duty-cycle ratio determines the active and inactive period given the total cycle duration: a total duration of 3s and a duty-cycle ratio of 0.6 gives an active period of 1.8s. In Figure 4.16(b), the cycle duration has been reduced to 200ms, therefore, keeping the duty-cycle ratio of 0.6 gives an active period of 120ms.

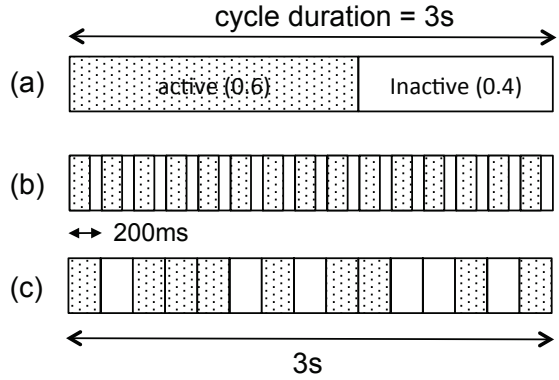

Figure 4.16: Various methods for duty-cycling

Case (a) and (b) have been evaluated by simulations in this thesis. Only case (a) has also been experimented because a cycle duration of 200ms is too short in practice for our prototype based on an XBee radio module that is switched ON and OFF. However, the simulation results have shown very little changes between the 3s-cycle case and the 200ms-cycle case, see Figure 4.15. In Figure 4.16(c), we illustrate the case of a cycle of duration 3s that is further sliced into 15 time slots of duration 200ms. In this case, the duty-cycle ratio can also be used to determine how many time slots in a cycle should be dedicated to radio activity. For instance, a duty-cycle ratio of 0.6 would give 9 active time slots out of the 15 time slots. These active time slots could be assigned in a random/probabilistic manner within the cycle. We haven't simulated this case yet and future works will integrate this alternative behaviour.

4.4 Conclusions

In this chapter, we proposed an adaptive low-latency, energy-efficient MAC protocol for mission critical surveillance applications. Critical applications need quick alert propagation. We design the duty cycle of follower nodes in relation to the fast frame capturing sentry node. The node's duty cycle now depends on its number of cover sets and frame capture rate of its sentry. This means that all the nodes in the network can have different duty cycles. Higher frame capture rate of the sentry node leads to high duty cycle of its one-hop neighbours. Hence better chances of alert propagation in case of detection by the sentry node.

Simulation results presented in this chapter show the high percentage of successfully propagated alerts, in comparison to a static MAC approach. At the same time, our approach was successful in considerable prolongation of the network lifetime.

We then implemented our approach on real sensors, which is detailed in the next chapter.

Chapter 5

CAMP Implementation

Contents

In this chapter, we present detailed implementation of Criticality-based Adaptive MAC protocol, presented in the chapter 4. We implemented CAMP protocol on Libelium WaspMote sensor board. We then compared the implementation results with our simulation results shown at the end of chapter 4.

5.1 Implementation of Follower nodes

We implemented the follower nodes with Libelium WaspMote sensor board. The WaspMote (www.libelium.com) is built around an Atmel ATmega1281 micro-controller running at 8MHz. There are 2 UARTs in the WaspMote that serve various purposes, one being to connect the micro-controller to the radio modules. The radio module is an XBee 802.15.4 radio from Digi (www.digi.com) that offers the basic 802.15.4 PHY and MAC layer service set in non-beacon mode. One advantage of the WaspMote is to allow through a single pin programming a complete power off of the radio module from the control software to implement the duty-cycle behaviour. The control program waits for initialisation messages to define the number of cover-sets ("C" message), the sentry capture rate ("R" message), the cycle length ("T" message) and a static schedule ("D" message) for the static duty-cycling case. When the number of cover-sets and the sentry's capture rate have been received, the criticality-based duty-cycling behaviour will start and the follower nodes will wait for alert messages ("A" messages) in order to respond by an "ACK" message. A led connected to the WaspMote will show when the radio is ON (active) or OFF (sleep), see Figure 5.1.

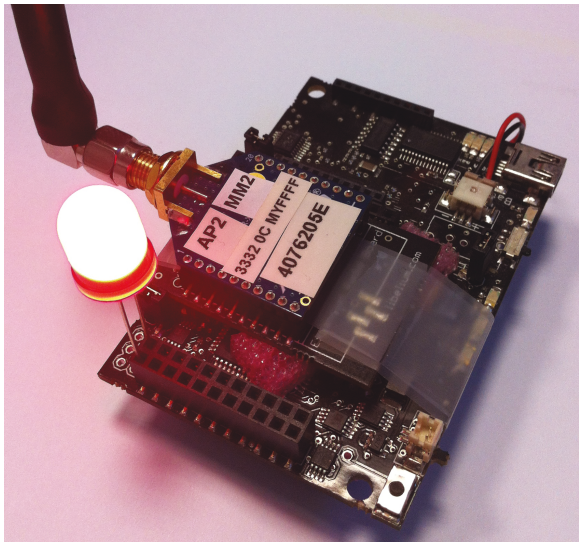

Figure 5.1: A Waspmote follower node

5.2 Implementation of a Sentry node

The sentry node behaviour is taken from the simulation model run on a Linux computer: we take the simulation time trace (converted into integer) of alert sending from a specific sentry node in order to have the time sequence as depicted below for node 10.

```
10 SN.node[10].Application Sending [alert]
18 SN.node[10].Application Sending [alert]
23 SN.node[10].Application Sending [alert]
29 SN.node[10].Application Sending [alert]
35 SN.node[10].Application Sending [alert]
40 SN.node[10].Application Sending [alert]
47 SN.node[10].Application Sending [alert]
54 SN.node[10].Application Sending [alert]
62 SN.node[10].Application Sending [alert]
69 SN.node[10].Application Sending [alert]
79 SN.node[10].Application Sending [alert]
86 SN.node[10].Application Sending [alert]
93 SN.node[10].Application Sending [alert]
101 SN.node[10].Application Sending [alert]
107 SN.node[10].Application Sending [alert]
113 SN.node[10].Application Sending [alert]
...
```

A python script reads the time sequence and broadcasts through an IEEE 802.15.4 XBee gateway the corresponding "A" message at the appropriate wall-clock time. For the moment, the simulation is run first, then the time trace is extracted for a given node. The python script output for the alert trace of node 10 is shown below.

```
$ python TraceSend.py
Sentry node: start intrusion detection
Read from trace file
Start time is
Mon Apr 21 15:01:07 2014
10
sleep for 10
Mon Apr 21 15:01:17 2014 : time 10 Intrusion 1 : sending alert
18
sleep for 8
Mon Apr 21 15:01:25 2014 : time 18 Intrusion 2 : sending alert
```

```
23
sleep for 5
Mon Apr 21 15:01:30 2014 : time 23 Intrusion 3 : sending alert
29
sleep for 6
Mon Apr 21 15:01:36 2014 : time 29 Intrusion 4 : sending alert
35
sleep for 6
Mon Apr 21 15:01:42 2014 : time 35 Intrusion 5 : sending alert
40
sleep for 5
Mon Apr 21 15:01:47 2014 : time 40 Intrusion 6 : sending alert
47
sleep for 7
Mon Apr 21 15:01:54 2014 : time 47 Intrusion 7 : sending alert
...
```

5.3 Test-bed

Figure 5.2 shows our test-bed with 1 sentry node (the Linux machine with the XBee gateway) and 5 follower nodes. This configuration is therefore similar to the one of sentry node 10 in our simulation study. A shell script will automatically configure each follower node with an appropriate cover-set size by sending a "C" message and will also broadcast the sentry's capture rate with an "R" message. On Figure 5.2, each follower node is identified by its MAC address (prefixed by 0x0013A200) and has an associated cover-set size (the same number of cover sets than for node 10's follower nodes in the simulation, i.e. 1, 1, 1, 5 and 6). When all follower nodes are configured (they have their number of cover sets and the frame capture rate of their sentry node) they start the duty-cycling behaviour. We set the cycle length to 3000ms and the frame capture rate of the sentry node to 0.51fps (i.e. the sentry node has 8 cover sets – 7 real cover sets and itself – and run at a criticality level of 0.8). The duty cycle value of follower nodes according to their number of cover sets was previously shown in Figure 4.4 and this value is reported for each follower node in Figure 5.2.

All commands are summarised below, using the `XBeeSendCmd` tool to send radio packet with an XBee gateway. For the sentry capture rate, instead of using broadcast communication, we deliberately used unicast in order to avoid follower nodes'

Figure 5.2: Test-bed with Waspmote follower nodes

duty-cycle synchronisation.

```
XBeeSendCmd -addr 0013A2004086D835 "C1#"
XBeeSendCmd -addr 0013A200408BC823 "C1#"
XBeeSendCmd -addr 0013A2004076205E "C1#"
XBeeSendCmd -addr 0013A2004086D828 "C6#"
XBeeSendCmd -addr 0013A20040762053 "C5#"
XBeeSendCmd -b "T3000#"
XBeeSendCmd -addr 0013A2004086D835 "R51#"
XBeeSendCmd -addr 0013A200408BC823 "R51#"
XBeeSendCmd -addr 0013A2004076205E "R51#"
XBeeSendCmd -addr 0013A2004086D828 "R51#"
XBeeSendCmd -addr 0013A20040762053 "R51#"
```

Figure 5.2 also illustrates a snapshot where each follower node has computed its duty-cycle ratio and where 2 follower nodes, 408BC823 and 4086D828, have their radio ON (their led is ON) identified by the green "active" banner, while the 3 others are in sleep mode.

5.4 Number of missed alert messages

Figure 5.2 also shows a radio promiscuous sniffer plugged into the `wireshark` packet analysis tool to record all the exchanged messages. We made the sentry node script stop after 500 alerts sent (we therefore have 500 "A" messages) and we counted the number of "ACK" messages from followers nodes. In the best case, when all follower nodes are active at the time of the alert message, we would have a total of 2500 ACK messages. A given alert message failed to be propagated if there are no ACK messages for this alert message. The `wireshark` trace allows us to easily track down such cases and an `awk` script will count for each alert messages how many responses were captured. The first lines of the `wireshark` trace is shown below. The sentry node has address `0xb3e8`. We can see that the time between 2 alerts corresponds to the time trace previously shown, e.g. 8s, 5s, 6s,... In these first lines, we can see that alert #1 has been acknowledged by 2 followers, alert #2 by 4 followers, alert #3 by 2 followers and alert #4 by 3 followers. We also show the case at time 9799 (alert #18) and time 9812 (alert #19) where no followers were active to receive the 18th alert.

```
9694.566880 0xb3e8                  Dst: Broadcast, Src: 0xb3e8
9694.736000 00:13:a2:00:40:76:20:53 Dst: Broadcast, Src: Maxstrea_00:40:76:20:53
9694.862784 00:13:a2:00:40:86:d8:35 Dst: Broadcast, Src: Maxstrea_00:40:86:d8:35
9702.221312 0xb3e8                  Dst: Broadcast, Src: 0xb3e8
9702.387296 00:13:a2:00:40:86:d8:35 Dst: Broadcast, Src: Maxstrea_00:40:86:d8:35
9702.388820 00:13:a2:00:40:86:d8:28 Dst: Broadcast, Src: Maxstrea_00:40:86:d8:28
9702.390560 00:13:a2:00:40:76:20:5e Dst: Broadcast, Src: Maxstrea_00:40:76:20:5e
9702.393216 00:13:a2:00:40:8b:c8:23 Dst: Broadcast, Src: Maxstrea_00:40:8b:c8:23
9707.064864 0xb3e8                  Dst: Broadcast, Src: 0xb3e8
9707.230816 00:13:a2:00:40:8b:c8:23 Dst: Broadcast, Src: Maxstrea_00:40:8b:c8:23
9707.630624 00:13:a2:00:40:86:d8:28 Dst: Broadcast, Src: Maxstrea_00:40:86:d8:28
9713.010560 0xb3e8                  Dst: Broadcast, Src: 0xb3e8
9713.097024 00:13:a2:00:40:76:20:53 Dst: Broadcast, Src: Maxstrea_00:40:76:20:53
9713.099616 00:13:a2:00:40:8b:c8:23 Dst: Broadcast, Src: Maxstrea_00:40:8b:c8:23
9713.176720 00:13:a2:00:40:86:d8:28 Dst: Broadcast, Src: Maxstrea_00:40:86:d8:28
...
9799.369728 0xb3e8                  Dst: Broadcast, Src: 0xb3e8, Bad FCS
9812.351552 0xb3e8                  Dst: Broadcast, Src: 0xb3e8, Bad FCS
...
```

We then compared the results obtained from the simulation model when the cycle duration was 3s to the results of the experimental settings. Figure 5.3 shows on the left part the alert response distribution for 500 alerts obtained from the simulation results, already shown in Figure 4.15, and on the right part the results from the experimentations.

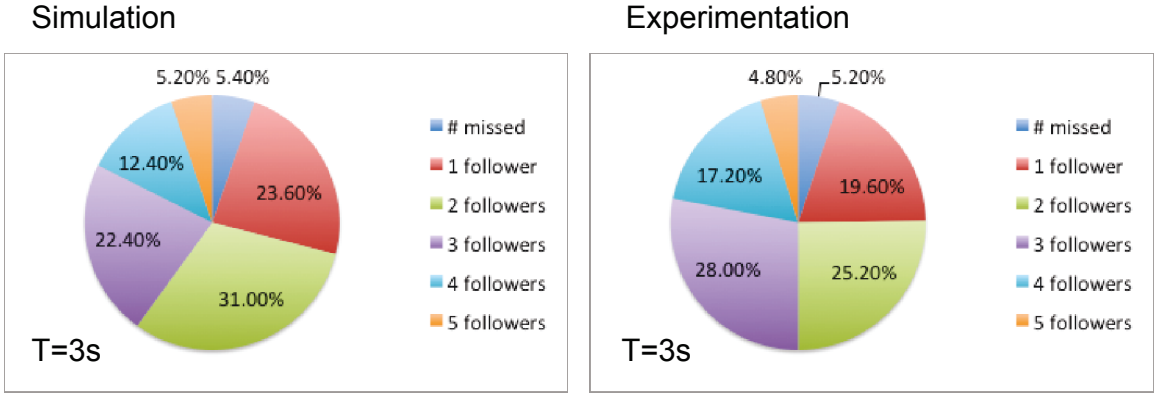

Figure 5.3: Alert response distribution for 500 alerts. Simulation (left), experimentation (right)

5.5 Energy consumption

We also measured the energy consumption of the WaspMote node when the radio is powered-off and when the radio is always powered-on. With the radio OFF and minimum processing tasks, the WaspMote consumes about 0.036J/s (36mW). With the radio ON and ready to receive and propagate alert messages it consumes about 0.236J/s (236mW). With a static duty-cycle strategy the energy gain can be directly obtained from the duty-cycle ratio as all nodes have the same ratio. With our criticality-based duty-cycle approach we can still know the overall energy gain with the duty-cycle value of each follower node. For instance, with a cycle duration of 3000 ms, the total amount of power consumed per cycle for all the 5 follower nodes are summarised in Table 5.1. For our Adaptive MAC, the duty-cycle value for each follower node is the one computed by the criticality model and shown previously in Figure 5.2, i.e. 0.33, 0.33, 0.33, 0.84 and 0.78. We can derive a "mean duty-cycle value" of 0.522 for this 5-node follower set.

When compared with the simulation results, the experimental results with this specific 5-node scenario show a smaller energy gain for Adaptive MAC when compared

Table 5.1: Total consumed power in Joules for all 5 follower nodes per cycle (3000 ms)

MAC	consumed power in J	mean duty-cycle	normalised ratio
Adaptive MAC	2.14 J	0.522	1
Static MAC 0.5	2.04 J	0.5	0.96
Static MAC 0.6	2.34 J	0.6	1.11
Static MAC 0.7	2.64 J	0.7	1.25
Static MAC 0.8	2.94 J	0.8	1.39

to static MAC cases. One reason is that the simulations are performed on a much larger topology and that the energy consumption was analysed in a global manner, where follower node sets can have very varying "mean duty-cycle value".

5.6 Conclusions

In this chapter we explained the implementation of criticality based adaptive model for mission critical surveillance applications. The implementation results are quite similar in many ways to the simulation results for alert propagation. The energy comparison shows lesser amount of energy gain during experimentation in comparison to the simulations. This is understandable because during simulations the energy consumptions are taken globally, while implementation process had only one set of 5 neighbour nodes. In a larger network, on average less number of nodes work on high duty cycle values as shown in figure 4.7.

Chapter 6

Conclusions and Perspectives

6.1 Conclusions

In this thesis, we proposed a duty-cycled MAC protocol for low latency alert propagation and low energy consumption targeted for mission-critical applications with image sensor nodes. Critical applications have to tackle with requirements like latency which might not be of utmost importance to traditional MAC protocols. Also, It is important to consider the criticality level of the network, as it can represent quality of service requirements of the network. Another important aspect to consider is that all the nodes in the network cannot be treated in a same way. In a WSN all the nodes work for the same purpose so if a node has high redundancy, it can be used more extensively for detection purposes. At the same time, a node with no back-up should maximise its lifetime, as the coverage of whole area under surveillance is important and once its battery is dead, there is no replacement available.

Our objective is to design a low latency MAC protocol for critical environments like intrusion detection systems, which can take into account the criticality level of the applications and the node's cover sets (the number of back-ups available for its replacement, in case it's battery is dead).

The key point in our approach is to link the duty cycle of nodes with the frame capture rate of fast image capturing nodes in their 1-hop neighbourhood. The nodes having the best capturing rate in their 1-hop vicinity are referred to as sentry nodes and their neighbouring nodes with low capturing speeds (low frame capture rate in comparison to the sentry node) are termed as follower nodes. Having fast capture

rate gives a node better probability of detecting an intrusion. As a result these fast capturing nodes will probably need to communicate more often (e.g., propagation of alerts, transmission of images to the sink) than the rest of its 1-hop neighbours. Sentry node's neighbours are therefore expected to receive data more often and hence need to be in a position to be able to communicate with the sentry node.

We proposed that the duty cycle of the follower nodes should depend on the frame capturing rate of the sentry nodes. The higher is the capture rate of the sentry node, the longer should be the duty cycle of its 1-hop neighbours as they have higher probability to receive data.

We used a criticality model to relate duty cycle of follower nodes to the sentry node's frame capture rate. The criticality model takes into account the number of cover sets of the node and the criticality level to calculate the duty cycle of the nodes.

6.1.1 Responsiveness of the Network

Simulation results presented in chapter 4, show the efficacy of the approach used. Our approach was successful in maximising the network lifetime and responding to the requirements of critical environments. Once an intrusion is captured, an alert should propagate in the network, to alert the neighbours of the intrusion capturing node and to relay the information to the sink. The number of successfully propagated alerts in the network hence can be a good quality parameter of responsiveness of the network. The number of missed alerts (or alerts which were not propagated) can determine the effectiveness of the approach used. We compared the number of successfully propagated alerts in the network using CAMP approach with various static duty cycle length values. The results have shown that our approach was responsive to high number of alerts in comparison to various static duty cycle lengths. We showed in chapter 4 that our proposed CAMP gives better results in terms of successful propagation of alerts in the network in comparison to a static duty cycled MAC protocol in which the nodes stay active 70% of the time. The results are marginally second to those of static MAC approach with 0.8 duty cycle (nodes staying active 80% of the time and conserving energy for only 20% of the time).

We implemented our approach on Libelium WaspMote and the experimental measures (done in chapter 5) have confirmed our simulation results. The experimental results were found be similar to the simulation results shown earlier in chapter 4.

6.1.2 Energy Consumption of the Network

In addition to the successful alert propagation, the results have shown that the energy consumed as a whole in the network can be reduced by 44% by using the CAMP approach as compared to a static duty-cycle approach with comparably same or lesser level of responsiveness (i.e, the static MAC with 0.7 duty cycle). In comparison to 50% and 60% duty cycled static MAC, our approach succeeds in reducing the energy consumption of the network up to 32% and 38% respectively, showing far more better results for successful alert propagation at the same time.

The experimental results show less amount of energy gains in comparison to the simulation results. One reason for this can be that during simulations, the energy consumption was analysed in a global manner, where follower nodes can have very varying mean duty cycle values.

6.2 Perspectives

In future we want to extend our proposition to two-hops for sentry selection, so that two-hop nodes can relate their duty cycles to the sentry nodes. Latency is to be minimised for alert propagation to two-hops.

Also, we want to make the duty cycle of nodes on the path to the sink to be somehow synchronised. However, instead of tight synchronisation that is hard to achieve in a large scale, we can probabilistically increase the contact time between two adjacent nodes (by relating their duty cycle with sentry) and then to the sink. So that once an image is captured, the latency is further minimised for alert propagation. By using the routing information, the probability of quick relaying of the information from each sentry to the sink can be vastly improved.

Although our approach has been designed for image sensors, the proposition can also work with traditional scalar sensors with disk coverage. Our contribution can also be used with low power listening and preamble approaches to determine the receiver periodic channel sampling interval, thus reducing the cost of preambles.

Publications

- Muhammad Ehsan, Congduc Pham, "Adaptive duty-cycled MAC for low-latency mission-critical surveillance applications" In ADHOC-NOW, Spain, 2014.

- Muhammad Ehsan, Congduc Pham, "Designing and implementing a criticality-based duty-cycled MAC for low-latency mission-critical surveillance applications" In the IEEE Symposium on Computers and Communications (ISCC), Portugal , 2014.

Bibliography

[1] G. J. Pottie and W. J. Kaiser, "Wireless integrated network sensors," *Commun. ACM*, vol. 43, no. 5, pp. 51–58, May 2000. [Online]. Available: http://doi.acm.org/10.1145/332833.332838 2

[2] J. Rabaey, M. Ammer, J. da Silva, J.L., D. Patel, and S. Roundy, "Picoradio supports ad hoc ultra-low power wireless networking," *Computer*, vol. 33, no. 7, pp. 42–48, Jul 2000. 2

[3] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *Communications Magazine, IEEE*, vol. 40, no. 8, pp. 102–114, Aug 2002. 2, 4, 6

[4] J. Stankovic, T. Abdelzaher, C. Lu, L. Sha, and J. Hou, "Real-time communication and coordination in embedded sensor networks," *Proceedings of the IEEE*, vol. 91, no. 7, pp. 1002–1022, July 2003. 2

[5] J.Zheng and A. Jamalipour, *Wireless Sensor Networks: A Networking Perspective.* Wiley-Interscience, 2009. 2, 4

[6] M. C. V. Ian F. Akyildiz, *Wireless Sensor Networks.* Wiley-Interscience, 2010. 4

[7] C. P. Waltenegus Dargie, *Fundamentals of Wireless Sensor Networks: Theory and Practice.* Wiley-Interscience, july 2010. 4

[8] K. Sohraby, D. Minoli, and T. Znati, *Wireless Sensor Networks: Technology, Protocols, and Applications.* Wiley-Interscience, 2007. 4

[9] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. shiuan Peh, and D. Rubenstein, "Energy-efficient computing for wildlife tracking: Design tradeoffs and early experiences with zebranet," 2002. 5

[10] A. Arora, P. Dutta, S. Bapat, V. Kulathumani, H. Zhang, V. Naik, V. Mittal, H. Cao, M. Demirbas, M. Gouda, Y. Choi, T. Herman, S. Kulkarni,

U. Arumugam, M. Nesterenko, A. Vora, and M. Miyashita, "A line in the sand: A wireless sensor network for target detection, classification, and tracking," *Comput. Netw.*, vol. 46, no. 5, pp. 605–634, Dec. 2004. [Online]. Available: http://dx.doi.org/10.1016/j.comnet.2004.06.007 5

[11] S. H. Lee, S. Lee, H. Song, and H. S. Lee, "Wireless sensor network design for tactical military applications : Remote large-scale environments," in *Military Communications Conference, 2009. MILCOM 2009. IEEE*, Oct 2009, pp. 1–7. 5

[12] M. Labrador and P. Wightman, "Topology control," in *Topology Control in Wireless Sensor Networks*. Springer Netherlands, 2009, pp. 61–70. [Online]. Available: http://dx.doi.org/10.1007/978-1-4020-9585-6_6 7

[13] L. Feeney and M. Nilsson, "Investigating the energy consumption of a wireless network interface in an ad hoc networking environment," in *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 3, 2001, pp. 1548–1557 vol.3. 7

[14] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris, "Span: An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks," *Wirel. Netw.*, vol. 8, no. 5, pp. 481–494, Sep. 2002. [Online]. Available: http://dx.doi.org/10.1023/A:1016542229220 7

[15] M. Sichitiu, "Cross-layer scheduling for power efficiency in wireless sensor networks," in *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, vol. 3, March 2004, pp. 1740–1750 vol.3. 7

[16] S. Singh and C. S. Raghavendra, "Pamas—power aware multi-access protocol with signalling for ad hoc networks," *SIGCOMM Comput. Commun. Rev.*, vol. 28, no. 3, pp. 5–26, Jul. 1998. [Online]. Available: http://doi.acm.org/10.1145/293927.293928 7

[17] I. Demirkol, C. Ersoy, and F. Alagoz, "Mac protocols for wireless sensor networks: a survey," *Communications Magazine, IEEE*, vol. 44, no. 4, pp. 115–121, April 2006. 7

[18] P. Dutta, S. Dawson-Haggerty, Y. Chen, C.-J. M. Liang, and A. Terzis, "Design and evaluation of a versatile and efficient receiver-initiated link layer for low-power wireless," in *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, ser. SenSys '10. New York, NY, USA: ACM, 2010, pp. 1–14. [Online]. Available: http://doi.acm.org/10.1145/1869983.1869985 7

[19] G. Lu, B. Krishnamachari, and C. S. Raghavendra, "An adaptive energy-efficient and low-latency mac for tree-based data gathering in sensor networks: Research articles," *Wirel. Commun. Mob. Comput.*, vol. 7, no. 7, pp. 863–875, Sep. 2007. [Online]. Available: http://dx.doi.org/10.1002/wcm.v7:7 7, 29, 30, 109

[20] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," in *Proceedings of the 2Nd International Conference on Embedded Networked Sensor Systems*, ser. SenSys '04. New York, NY, USA: ACM, 2004, pp. 95–107. [Online]. Available: http://doi.acm.org/10.1145/1031495.1031508 7, 9, 12, 27, 42, 79

[21] T. van Dam and K. Langendoen, "An adaptive energy-efficient mac protocol for wireless sensor networks," in *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems*, ser. SenSys '03. New York, NY, USA: ACM, 2003, pp. 171–180. [Online]. Available: http://doi.acm.org/10.1145/958491.958512 7, 21, 37, 40, 41, 42

[22] W. Ye, J. Heidemann, and D. Estrin, "Medium access control with coordinated adaptive sleeping for wireless sensor networks," *IEEE/ACM Trans. Netw.*, vol. 12, no. 3, pp. 493–506, Jun. 2004. [Online]. Available: http://dx.doi.org/10.1109/TNET.2004.828953 7, 12, 19, 20, 21, 25, 37, 38, 39, 40, 41, 42

[23] 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*, ser. SenSys '06. New York, NY, USA: ACM, 2006, pp. 307–320. [Online]. Available: http://doi.acm.org/10.1145/1182807.1182838 7, 27, 28, 42, 79

[24] A. Bachir, M. Dohler, T. Watteyne, and K. K. Leung, "Mac essentials for wireless sensor networks," *Commun. Surveys Tuts.*, vol. 12, no. 2, pp. 222–248, Apr. 2010. [Online]. Available: http://dx.doi.org/10.1109/SURV.2010.020510.00058 9

[25] M. Khanafer, M. Guennoun, and H. Mouftah, "A survey of beacon-enabled ieee 802.15.4 mac protocols in wireless sensor networks," *Communications Surveys Tutorials, IEEE*, vol. 16, no. 2, pp. 856–876, Second 2014. 9

[26] "Ieee standard for local and metropolitan area networks–part 15.4: Low-rate wireless personal area networks (lr-wpans) amendment 1: Mac sublayer," *IEEE Std 802.15.4e-2012 (Amendment to IEEE Std 802.15.4-2011)*, pp. 1–225, April 2012. 9, 16

[27] A. Grilo, M. Macedo, and M. Nunes, "An energy-efficient low-latency multi-sink mac protocol for alarm-driven wireless sensor networks," in *Wireless Systems and Mobility in Next Generation Internet*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2007, vol. 4396, pp. 87–101. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-70969-5_7 12, 42

[28] 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. [Online]. Available: http://dx.doi.org/10.1007/s11276-006-6151-z 12, 34, 41, 42

[29] S. Ray, I. Demirkol, and W. Heinzelman, "Supporting bursty traffic in wireless sensor networks through a distributed advertisement-based tdma protocol (atma)," *Ad Hoc Networks*, May 2013. 12, 40, 42

[30] A. Makhoul, R. Saadi, and C. Pham, "Risk management in intrusion detection applications with wireless video sensor networks," in *in IEEE WCNC*, 2010. 12, 43

[31] B. N. Y.-q. S. Anis Koubaa, Mrio Alves, "Improving the ieee 802.15.4 slotted csma/ca mac for timecritical events in wireless sensor networks," in *In Proc. of the 2nd Workshop on Real Time Networks (RTN*, 2006, pp. 574–579. 19

[32] C. Q. Peng Lin and X. Wang, "Medium access control with a dynamic duty cycle for sensor networks," in *Wireless Communications and Networking Conference, 2004. WCNC. 2004 IEEE*, vol. 3, March 2004, pp. 1534–1539 Vol.3. 22

[33] S. Zhuo, Z. Wang, Y.-Q. Song, Z. Wang, and L. Almeida, "iqueue-mac: A traffic adaptive duty-cycled mac protocol with dynamic slot allocation," in *Sensor, Mesh and Ad Hoc Communications and Networks (SECON), 2013 10th Annual IEEE Communications Society Conference on*, June 2013, pp. 95–103. 23

[34] J. Ansari, J. Riihijarvi, P. Mahonen, and J. Haapola, "Implementation and performance evaluation of nanomac; a low power mac solution for high density wireless sensor networks," *Int. J. Sen. Netw.*, vol. 2, no. 5/6, pp. 341–349, Jul. 2007. [Online]. Available: http://dx.doi.org/10.1504/IJSNET.2007.014361 25

[35] Y. W. Li, Y. and J. Heidemann, "Energy and latency control in low duty cycle mac protocols," 2005. 25

[36] C. Enz, A. El-Hoiydi, J.-D. Decotignie, and V. Peiris, "Wisenet: an ultralow-power wireless sensor network solution," *Computer*, vol. 37, no. 8, pp. 62–70, Aug 2004. 25, 26, 27, 108

[37] A. El-Hoiydi, "Spatial tdma and csma with preamble sampling for low power ad hoc wireless sensor networks," in *Proceedings of the Seventh International Symposium on Computers and Communications (ISCC'02)*, ser. ISCC '02. Washington, DC, USA: IEEE Computer Society, 2002, pp. 685–. [Online]. Available: http://dl.acm.org/citation.cfm?id=839293.843207 27

[38] A. Dunkels, "The contikimac radio duty cycling protocol," Tech. Rep., 2011. 27

[39] A. El-Hoiydi, "Aloha with preamble sampling for sporadic traffic in ad hoc wireless sensor networks," in *Communications, 2002. ICC 2002. IEEE International Conference on*, vol. 5, 2002, pp. 3418–3423 vol.5. 28

[40] N. Abramson, "The aloha system: Another alternative for computer communications," in *Proceedings of the November 17-19, 1970, Fall Joint Computer Conference*, ser. AFIPS '70 (Fall). New York, NY, USA: ACM, 1970, pp. 281–285. [Online]. Available: http://doi.acm.org/10.1145/1478462.1478502 28

[41] I. Rhee, A. Warrier, M. Aia, J. Min, and M. Sichitiu, "Z-mac: A hybrid mac for wireless sensor networks," *Networking, IEEE/ACM Transactions on*, vol. 16, no. 3, pp. 511–524, June 2008. 30, 38

[42] I. Rhee, A. C. Warrier, and L. Xu, "Randomized dining philosophers to tdma scheduling in wireless sensor networks," Tech. Rep., 2004. 30

[43] B. Nefzi and Y.-Q. Song, "Qos for wireless sensor networks: Enabling service differentiation at the mac sub-layer using cosens," *Ad Hoc Networks*, vol. 10, no. 4, pp. 680 – 695, 2012, advances in Ad Hoc Networks (II). [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1570870511001338 31, 32, 109

[44] M. Caccamo, L. Zhang, L. Sha, and G. Buttazzo, "An implicit prioritized access protocol for wireless sensor networks," in *Real-Time Systems Symposium, 2002. RTSS 2002. 23rd IEEE*, 2002, pp. 39–48. 33

[45] L. Bao and J. J. Garcia-Luna-Aceves, "A new approach to channel access scheduling for ad hoc networks," in *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '01. New York, NY, USA: ACM, 2001, pp. 210–221. [Online]. Available: http://doi.acm.org/10.1145/381677.381698 34

[46] in *Wireless Sensor Networks*, ser. Lecture Notes in Computer Science, 2006, vol. 3868. 35

[47] S. Ergen and P. Varaiya, "Pedamacs: power efficient and delay aware medium access protocol for sensor networks," *Mobile Computing, IEEE Transactions on*, vol. 5, no. 7, pp. 920–930, July 2006. 35

[48] R. Mangharam, A. Rowe, and R. Rajkumar, "Firefly: A cross-layer platform for real-time embedded wireless networks," *Real-Time Syst.*, vol. 37, no. 3, pp. 183–231, Dec. 2007. [Online]. Available: http://dx.doi.org/10.1007/s11241-007-9028-z 36

[49] J. Chen, P. Zhu, and Z. Qi, "Pr-mac: Path-oriented real-time mac protocol for wireless sensor network," in *Embedded Software and Systems*, ser. Lecture Notes in Computer Science, Y.-H. Lee, H.-N. Kim, J. Kim, Y. Park, L. Yang, and S. Kim, Eds. Springer Berlin Heidelberg, 2007, vol. 4523, pp. 530–539. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-72685-2_49 36

[50] T. Watteyne, I. Augé-Blum, and S. Ubéda, "Dual-mode real-time mac protocol for wireless sensor networks: A validation/simulation approach," in *Proceedings of the First International Conference on Integrated Internet Ad Hoc and Sensor Networks*, ser. InterSense '06. New York, NY, USA: ACM, 2006. [Online]. Available: http://doi.acm.org/10.1145/1142680.1142683 36

[51] B. Jang, J. B. Lim, and M. L. Sichitiu, "An asynchronous scheduled mac protocol for wireless sensor networks," *Computer Networks*, vol. 57, no. 1, pp. 85 – 98, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128612003246 37, 66, 79

[52] W. Ye, F. Silva, and J. Heidemann, "Ultra-low duty cycle mac with scheduled channel polling," in *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems*, ser. SenSys '06. New York, NY, USA: ACM, 2006, pp. 321–334. [Online]. Available: http://doi.acm.org/10.1145/1182807.1182839 37

[53] L. Sitanayah, C. J. Sreenan, and K. N. Brown, "A hybrid mac protocol for emergency response wireless sensor networks," *Ad Hoc Networks*, vol. 20, no. 0, pp. 77 – 95, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1570870514000638 38, 42

[54] Y. Sun, S. Du, O. Gurewitz, and D. B. Johnson, "Dw-mac: A low latency, energy efficient demand-wakeup mac protocol for wireless sensor networks," in *Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, ser. MobiHoc '08. New York, NY, USA: ACM, 2008,

pp. 53–62. [Online]. Available: http://doi.acm.org/10.1145/1374618.1374627 38, 66

[55] S. Du, A. Saha, and D. Johnson, "Rmac: A routing-enhanced duty-cycle mac protocol for wireless sensor networks," in *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE*, May 2007, pp. 1478–1486. 38

[56] P. J. Shin, J. Park, and A. C. Kak, "A predictive duty cycle adaptation framework using augmented sensing for wireless camera networks," *ACM Trans. Sen. Netw.*, vol. 10, no. 2, pp. 22:1–22:31, Jan. 2014. [Online]. Available: http://doi.acm.org/10.1145/2530280 39, 42

[57] E. Egea-Lopez, J. Vales-Alonso, A. S. Martinez-Sala, J. Garcia-Haro, P. Pavon-Marino, and M. V. Bueno Delgado, "A wireless sensor networks mac protocol for real-time applications," *Personal Ubiquitous Comput.*, vol. 12, no. 2, pp. 111–122, Jan. 2008. [Online]. Available: http://dx.doi.org/10.1007/s00779-006-0111-6 39

[58] S. Ray, I. Demirkol, and W. Heinzelman, "Adv-mac: Analysis and optimization of energy efficiency through data advertisements for wireless sensor networks," *Ad Hoc Netw.*, vol. 9, no. 5, pp. 876–892, Jul. 2011. [Online]. Available: http://dx.doi.org/10.1016/j.adhoc.2010.10.005 40, 41

[59] D.-L. Nguyen, L. Q. V. Tran, O. Berder, and O. Sentieys, "A low-latency and energy-efficient mac protocol for cooperative wireless sensor networks," in *Global Communications Conference (GLOBECOM), 2013 IEEE*, Dec 2013, pp. 3826–3831. 41

[60] O. Dousse, C. Tavoularis, and P. Thiran, "Delay of intrusion detection in wireless sensor networks," in *Proceedings of the 7th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, ser. MobiHoc '06. New York, NY, USA: ACM, 2006, pp. 155–165. [Online]. Available: http://doi.acm.org/10.1145/1132905.1132923 46

[61] A. Czarlinska and D. Kundur, "Wireless image sensor networks: event acquisition in attack-prone and uncertain environments," *Multidimensional Systems and Signal Processing*, vol. 20, no. 2, pp. 135–164, 2009. [Online]. Available: http://dx.doi.org/10.1007/s11045-008-0071-2 46

[62] E. Pignaton de Freitas, T. Heimfarth, C. Pereira, A. Ferreira, F. Wagner, and T. Larsson, "Evaluation of coordination strategies for heterogeneous sensor networks aiming at surveillance applications," in *Sensors, 2009 IEEE*, Oct 2009, pp. 591–596. 46

[63] M. Alaei and J. Barcelo-Ordinas, "Priority-based node selection and scheduling for wireless multimedia sensor networks," in *Wireless and Mobile Computing, Networking and Communications (WiMob), 2010 IEEE 6th International Conference on*, Oct 2010, pp. 151–158. 46

[64] S. Paniga, L. Borsani, A. Redondi, M. Tagliasacchi, and M. Cesana, "Experimental evaluation of a video streaming system for wireless multimedia sensor networks," in *Ad Hoc Networking Workshop (Med-Hoc-Net), 2011 The 10th IFIP Annual Mediterranean*, June 2011, pp. 165–170. 46

[65] I. Butun, S. Morgera, and R. Sankar, "A survey of intrusion detection systems in wireless sensor networks," *Communications Surveys Tutorials, IEEE*, vol. 16, no. 1, pp. 266–282, First 2014. 46

[66] H. Gupta, Z. Zhou, S. Das, and Q. Gu, "Connected sensor cover: self-organization of sensor networks for efficient query execution," *Networking, IEEE/ACM Transactions on*, vol. 14, no. 1, pp. 55–67, Feb 2006. 46

[67] S. Shakkottai, R. Srikant, and N. Shroff, "Unreliable sensor grids: coverage, connectivity and diameter," in *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, vol. 2, March 2003, pp. 1073–1083 vol.2. 46

[68] T. Yan, T. He, and J. A. Stankovic, "Differentiated surveillance for sensor networks," in *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems*, ser. SenSys '03. New York, NY, USA: ACM, 2003, pp. 51–62. [Online]. Available: http://doi.acm.org/10.1145/958491.958498 47

[69] Y. Zhu and L. Ni, "Probabilistic approach to provisioning guaranteed qos for distributed event detection," in *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, April 2008, pp. –. 47

[70] C. Pham, A. Makhoul, and R. Saadi, "Risk-based adaptive scheduling in randomly deployed video sensor networks for critical surveillance applications," *Journal of Network and Computer Applications*, vol. 34, no. 2, pp. 783 – 795, 2011, efficient and Robust Security and Services of Wireless Mesh Networks. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1084804510001712 47, 59

[71] C. Pham and A. Makhoul, "Performance study of multiple cover-set strategies for mission-critical video surveillance with wireless video sensors," in *Wireless and Mobile Computing, Networking and Communications (WiMob), 2010 IEEE 6th International Conference on*, Oct 2010, pp. 208–216. 48, 50

[72] C. Pham, "Scheduling randomly-deployed heterogeneous video sensor nodes for reduced intrusion detection time," in *Distributed Computing and Networking*, ser. Lecture Notes in Computer Science, M. Aguilera, H. Yu, N. Vaidya, V. Srinivasan, and R. Choudhury, Eds. Springer Berlin Heidelberg, 2011, vol. 6522, pp. 303–314. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-17679-1_27 54

[73] C. Pham, "Network lifetime and stealth time of wireless video sensor intrusion detection systems under risk-based scheduling," in *Wireless and Pervasive Computing (ISWPC), 2011 6th International Symposium on*, Feb 2011, pp. 1–6. 54

[74] C. Pham, "Low cost wireless image sensor networks for visual surveillance and intrusion detection applications," *12th IEEE International Conference on Networking, Sensing and Control (ICNSC)*, Apr 2015. 57

[75] C. Pham, "An image sensor board based on arduino due and ucamii camera. http://www.univ-pau.fr/ cpham/wsn-model/tool-html/imagesensor.html," accessed 12/2/2015. 57

[76] C. Merlin and W. Heinzelman, "Schedule adaptation of low-power-listening protocols for wireless sensor networks," *Mobile Computing, IEEE Transactions on*, vol. 9, no. 5, pp. 672–685, May 2010. 79

List of Figures

List of Tables