POWER SAVING AND LOAD BALANCING FOR SOLAR WLAN

Dynamic Power Saving and Load Balancing for Solar Powered WLAN Infrastructure

By
Enrique J. Vargas, B.Sc.

A Thesis
Submitted to the School of Graduate Studies
in partial fulfilment of the requirements for the degree of

Master of Applied Science
Department of Electrical and Computer Engineering
McMaster University

MASTER OF APPLIED SCIENCE (2005)
(Electrical and Computer Engineering)

McMaster University
Hamilton, Ontario

TITLE:
Dynamic Power Saving and Load Balancing for Solar Powered WLAN Infrastructure

AUTHOR: Enrique J. Vargas, B.Sc.
(Pontificia Universidad Catolica del Peru, Lima, Peru)

SUPERVISOR: Dr. Terence D. Todd

NUMBER OF PAGES: xx, 109

Abstract

The IEEE 802.11 standard has been widely adopted as a Wireless LAN (WLAN) technology. This widespread proliferation of the technology has lead to an increase in the number of users taking advantage of so-called "hot-spots" which leads to an increased demand on bandwidth provided by Access Points (APs) in the hot-spot. The logical solution is to deploy more overlapping access points in the same coverage area, thus increasing the capacity of the system by providing load balancing services. However, when a hot-spot is located in an outdoor environment, it becomes difficult to provide the AP with power which is traditionally carried over wired links thus causing the service provider to incur additional costs, not to mention the impossibility in some cases of delivering power to the AP. This problem can be overcome by using solar-panel powered APs which we will refer to as solar nodes (SNs). In this thesis we examine the load-balancing problem that arises when two or more SNs are co-located in the same coverage area. We propose and evaluate two algorithms for efficiently distributing the load among them (transferring stations (STAs) from SN to neighboring SNs) and increasing their lifetime by using power saving schemes that co-ordinate the wake/sleep patterns of the SNs based on traffic load. Finally, a Connection Admission Control (CAC) function is proposed that the SN should use in order to provide controlled access to services. We

demonstrate through simulations that our proposals can significantly reduce the hardware requirements and cost of SNs and improve the service perceived by STAs in terms of transmission delay.

Contents

List of Abbreviations and Acronyms

ACK	acknowledgment
AFIS	arbitration interframe space
AP	access point
BSS	basic service set
BSSID	basic service set identification
CAC	connection admission control
CDF	cumulative distribution function
CF	contention free
CFB	contention free burst
CFP	contention-free period
CSMA/CA	carrier sense multiple access with collision avoidance
CTS	clear to send
DCF	distributed coordination function
DIFS	distributed (coordination function) interframe space
DL	downlink
DS	distribution system
DT	disassociation toke
DTIM	delivery traffic indication message
EDCA	enhanced distributed channel access
EIFS	extended interframe space
ESS	extended service set
HC	hybrid coordinator
HCCA	hybrid controlled channel access
HCF	hybrid coordination function
IAPP	inter access point protocol
IBSS	independent basic service set

IFS	interframe space
IP	internet protocol
MAC	medium access control
NAV	Network Allocation Vector
NIC	network interface card
PCF	point coordination function
PHY	physical (layer)
PIFS	point (coordination function) interframe space
PS-STA	power save (mode) station
QAP	QoS access point
QBSS	QoS basic service set
QoS	quality of service
QSTA	QoS station
RSSI	received signal strength indication
RTC	request to send
SIFS	Short Interframe Space
SN	solar node
SSID	service set identifier
STA	station
TSPECS	traffic specifications
TXOP	transmission opportunity
UDP	user datagram protocol
UL	uplink
VoIP	voice over internet protocol
WDS	wireless distribution system
WLAN	Wireless Local Area Network

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Overview

With the evolution of wireless networking, users have come to expect a certain Quality of Service (QoS) from their wireless devices and service providers. The demand for ubiquitous always-on high speed wireless connections has long been considered one of the final frontiers for communication networks. On the other hand, wireless networks face more constraints than their wired counterparts such as limited bandwidth, frequency assignment, lower bandwidth, etc.

With the delayed introduction of 3G systems due to economical reasons and their failure to deliver the expected end-user experience, service providers have turned to the IEEE 802.11 Wireless Local Area Networks (WLAN) standard as an inexpensive alternative for providing high data rates at the expense of user mobility which is still currently limited due to the lack of city-wide deployments of the technology.

This thesis is based on the IEEE 802.11 Wireless LAN standard. However, the ideas developed here are applicable for other wireless standards in principle.

Since its standardization in 1999, the IEEE 802.11 WLAN standard has been growing in popularity. It was first adopted as a solution in enterprise markets to replace classical wired LAN connections. However, its low implementation cost made it possible for the technology to penetrate residential markets as well. With this evolution of the technology, service provider have equipped zones with expected high end-user concentration such as coffee shops and airport lounges with the technology. These zones have now become known as hot-spots. Moreover, in the past 3 years several outdoor networks have been implemented based on IEEE 802.11 networks, these networks usually rely on "meshing" different 802.11-based networks thus giving rise to WLAN Mesh Networks.

The most common IEEE networks are the Infrastructure networks which consist basically of one central device, know as the Access Point (AP), which provides network access to end users. End stations (STAs) communicate bi-directionally with APs to reach any destination and also to be contacted from other STAs. As a consequence, a Wireless Mesh network consists of a set of Access Points, which are not only interconnected establishing master/slave relationships, but also provide wireless access to end STAs.

Let us now consider the case were Wireless Mesh Networks need to be deployed in places where common energy resources are scarce, or where the implementation occurs in an ad-hoc fashion or furthermore, in situations where environmental protection policies are encouraged. In these cases, it would be

beneficial to have APs powered using renewable energy sources like solar power. One example of these networks is the SolarMESH Network which was developed here at McMaster University [17]. It is made up of a set of solar and battery powered APs. When these networks grow in size, hot-spots arise creating points of high end-user utilization. The scope of this thesis is a thorough examination of the performance of such hot-spots. The next section presents the main contribution of this thesis.

1.2 Motivation and Contribution of the Thesis

Typically, a single AP deployed cannot offer enough capacity to satisfy users' demands (i.e. traffic load). This specially occurs in places where concurrent users make use of wireless access services; as mentioned before these places are known as hot-spots. Both indoor and outdoor wireless networks face this problem. Network administrators and engineers usually resort to deploying more APs in hot-spots in order to increase the offered capacity. Thus, overlapping AP deployments arise. On the other hand, the IEEE 802.11 standard does not specify procedures to coordinate MAC activities among overlapping APs nor does it provide load balancing/distribution mechanisms. Moreover, under this scenario a new Connection Addmission Control (CAC) procedure for connection oriented services should be implemented to take advantage of these complex, overlapping topologies.

Additionally, there is another issue to consider, which is the fact that outdoor deployments can be implemented using solar/battery powered APs known

as solar nodes (SN). This fact adds some design constraints for the proposed schemes since all the SNs deployed might not be needed all the time (e.g. at nights). Our schemes propose procedures to sleep SNs when they are not required and activate them when the load increases. In this way batteries and solar panels can be dimensioned according to actual power consumption leading to a minimization of SN cost.

To summarize, the main contributions of this thesis include providing novel algorithms to redistribute the load and to control connection admission for overlapping SN deployments, thus guaranteeing good performance in terms of transmission delay under a maximum offered load. These proposed algorithms also increase the lifetime of solar/battery powered APs and reduce the cost of implementing them. Finally, detailed performance results provide verification of the validity and importance of this work.

1.3 Thesis organization

The rest of the thesis is organized as follows.

Chapter 2 introduces the IEEE 802.11 standard with emphasis on the Medium Access Control (MAC) layer. Also a subset of the standard known as the 802.11e standard that pertains to QoS is also introduced. Additionally, an extensive literature survey is presented in order to highlight previous research work related to the problem.

In the following part, Chapter 3 provides a formal description of the problem. In addition, several system variables and procedures are defined.

Chapter 4 constitutes the principal part of this thesis. Two algorithms, PLAS and LAS, are proposed to address the load balancing/distribution problem for overlapping deployments.

In Chapter 5 some analytical models are developed in order to determine the maximum utilization ($U_{BEthreshold}$) in DCF operation. A power consumption model is also developed which will be used later on to verify the simulation results. Additionally, two brief models to produce self-similar traffic and bulk traffic are presented.

Chapter 6 presents detailed performance evaluation of the proposed solutions. Performance was evaluated in this thesis for two different traffic profiles, the first representing a simple, fixed load pattern, while the second represents a more realistic traffic profile.

Finally, Chapter 7 provides a summary of the work presented, the conclusion of the thesis and some potential future work.

Chapter 2

Background

2.1 The IEEE 802.11 Standard

2.1.1 Overview

The IEEE 802.11 standard [13] defines MAC procedures for WLAN networks based on the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) protocol. Figure 2.1 illustrates the basic elements in a Wireless Local Area Network (WLAN): the Basic Service Set (BSS) is the basic building block in IEEE 802.11 networks and defines the area (volume/space) where a group of STAs can communicate among each other; the Wireless Medium (WM) is defined as the interface in which frames are transmitted and it could consist of any of the physical layers defined in the IEEE 802.11 standard; the Distribution System, usually wired, provides interconnection between BSSs and interconnection to external networks; the Access Point (AP) is a station that provides access to the DS to other stations; finally stations (STAs) are network

devices with wireless interface(s) and in most of the cases are mobile devices. In addition to these concepts, a set of BSSs connected via a DS is known as an Extended Service Set (ESS).

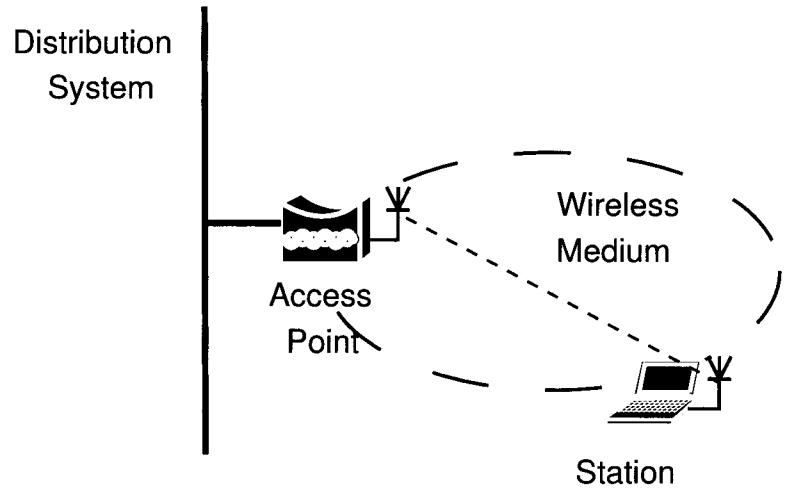
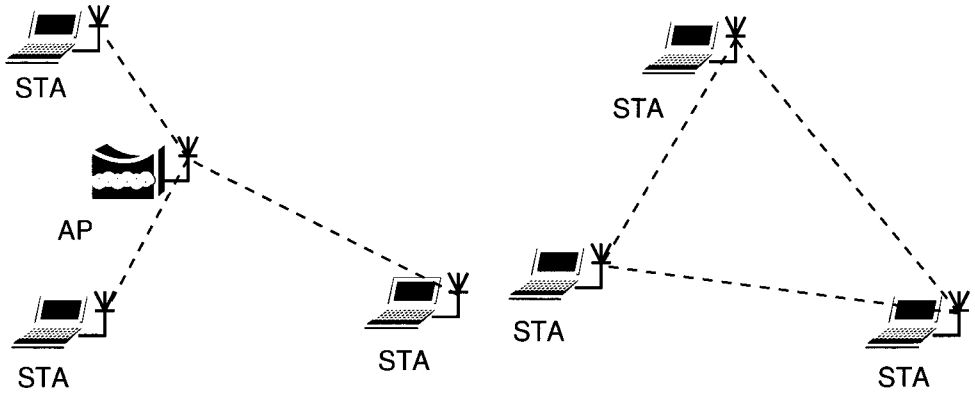


Figure 2.1: Basic Elements in IEEE 802.11 Networks

The original standard specifies two medium access mechanisms: Distributed Coordinated Function (DCF) and Point Coordinated Function (PCF).

The IEEE 802.11 standard also defines two possible network architectures: the Infrastructure Basic Service Set (Infrastructure BSS) and Independent Basic Service Set (IBSS). In the former case, communications can only occur bidirectionally between the AP and STAs. The infrastructure BSS implementation is more commonly deployed in WLANs, because centralized communication with the AP provides better network stability and management. The coverage area of the AP defines the BSS set. In the latter case, communications are allowed directly among the STAs (the AP does not exist in this case). IBSS are a subset of what is referred to in the literature as Ad-Hoc networks and

they are usually are implemented in situations where short-lived networks are needed. The BSS set is composed of the links created among stations, in this case we cannot specify a coverage area. Figure 2.2(a) shows the infrastructure BSS architecture while Figure 2.2(b) shows the IBSS architecture. The rest of this thesis will only focus on infrastructure BSS networks.



(a) Infrastructure BSS Network **(b)** Independent BSS Network

Figure 2.2: Two Network Topologies Specified in IEEE 802.11

2.1.2 Distributed Coordination Function (DCF)

Even though IEEE 802.11 defines two access mechanisms, DCF and PCF, only the former is supported in most of the commercial APs and STAs. In this section we will describe in detail the DCF access mechanism, which is based on the CSMA/CA protocol.

DCF was designed to be used in networks where the interaction between stations is not centrally coordinated. Thus, it may be used for either Independent or Infrastructure BSSs. A set of rules exist that defines the DCF access

mechanism. Before each STA attempts to transmit, it has to check whether the channel is idle. If the channel is not idle, the STA defers the transmission using an exponential backoff algorithm to avoid collisions. The CSMA/CA is similar to the CSMA/CD protocol used in Ethernet networks, but the former tries to avoid collisions that significantly degrade the performance of WLAN networks.

Carrier Sensing Function and Network Allocation Vector (NAV)

Each STA attempting to transmit a frame must sense the channel to determine if it is available. There are two carrier sensing functions defined in the IEEE 802.11 standard: the physical carrier-sensing function and the virtual carrier-sensing function. Both should indicate that the medium is available in order to transmit a frame.

The physical carrier-sensing function depends on the modulation and medium used (physical layer). Medium activity is sensed using the receiver; if a transmission is occurring, then the receiver should sense it and notify the MAC layer that the medium is busy.

Sometimes physical carrier-sensing functions are not enough to determine if the channel is busy. Problems associated with hidden STAs cannot be solved by the physical carrier-sensing function. To overcome this problem, IEEE 802.11 defines a virtual carrier-sensing function, which is provided by the Network Allocation Vector (NAV). In this case, the medium is reserved by using timers carried in a duration field present in some control frames. The NAV itself is a timer that indicates the amount of time the medium will be reserved. STAs

that gained access to the medium, set the NAV for the time they expect to use the medium (this can vary depending on transmission rate supported by the STAs). Other STAs that receive/listen a frame setting the NAV, have to count down to zero when they want to transmit; if the NAV is greater than zero, this means that the medium is busy, otherwise the medium is considered idle.

Interframe Space (IFS)

The time interval between frame transmissions is denoted the Interframe Space (IFS). There are four different IFSs defined in IEEE 802.11 and they are defined in units of time that are independent of the transmission rate of each STA. The Short Interframe Space (SIFS) is the shortest IFS defined and is used for the highest-priority transmissions. For instance, in the RTS/CTS exchange (to be explained later) or to acknowledge a frame. The PCF Interframe Space (PIFS) is used by PCF during contention-free operation and provides higher priority than traffic transmitted using DCF. The third is the DCF Interframe Space (DIFS), which is the minimum idle time between consecutive transmissions (including atomic operations like RTS/CTS) for contention-based services. The last one is the Extended Interframe Space (EIFS), which unlike the others, does not have a fixed value and is used after erroneous frame transmissions. Figure 2.3 illustrates the IFSs and their respective priority.

Random Backoff Time

After a frame has been successfully transmitted during the contention-period, STAs that have buffered frames may attempt to transmit them after a DIFS

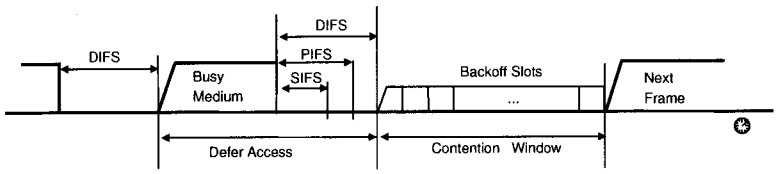


Figure 2.3: Interframe Space and Deferred Access

period. STAs attempting to transmit should sense the medium thus invoking the carrier-sensing functions previously explained and should defer access until the medium is sensed idle without interruption for a DIFS period of time. After this DIFS period, STAs attempting to transmit must also defer access for a random backoff period of time. The expression to calculate the backoff time is shown in Equation 2.1:

$$\text{Backoff Time} = \text{Random}() \times \text{aSlotTime} \tag{2.1}$$

Where aSlotTime is a fixed value that depends on the physical layer being used. Random() is a pseudorandom integer drawn from a uniform distribution over the interval [0,CW], where CW can takes values between CW_{min} and CW_{max}, both being dependent on the physical layer parameters. CW takes an initial value of CW_{min} and after every failed attempt to transmit (which increases the associated retry counters) CW augments to the next value, this procedure might go on until CW is equal to CW_{max}. After a successful transmission, CW is reset to CW_{min}. The main idea behind using the Random Backoff Timer, is to reduce the probability of collisions. The STA that has the lowest Random Backoff Timer gains access to the medium and transmits before the other contending STAs.

Basic Access Procedure and RTS/CTS Procedure

DCF includes two transmission procedures, the Basic Access procedure, which is intended to be used for transmitting small data frames, and the Request to Send / Clear to Send (RTS/CTS) procedure, which should be used to protect large data frame transmissions and in this way overcome problems caused by hidden STAs. The IEEE 802.11 standard suggests the use of a configurable threshold to select frames that should be transmitted either with the Basic Access procedure or the RTS/CTS procedure. When using the Basic Access procedure, the STAs should sense that the medium is idle for a DIFS period (or EIFS if an error occurred in the previous transmission attempt), then defer using the Random Backoff Time procedure. The STA that gained access to the medium may transmit a data frame to the intended receiver. If the transmission is successful, the receiver must acknowledge this frame by sending an ACK frame. The IEEE 802.11 standard requires all transmissions to be acknowledged, providing in this way a reliable MAC. Unacknowledged transmissions are considered failures and senders will have to retransmit these frames (if the retry limits have not been exceeded). Figure 2.4 illustrates the Basic Access procedure.

As mentioned before, RTS/CTS protects transmission against hidden STAs and improves the performance when transmitting large data frames. However, if the data frames are small the RTS/CTS procedure reduces the overall data throughput because of the associated overhead. Hidden STA problems might arise when not all the STAs in the BSS can listen each other, therefore while one of them is transmitting other STAs might sense the medium idle trying

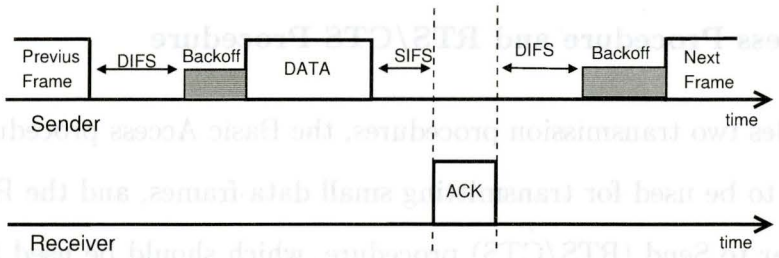


Figure 2.4: Basic Access Procedure

to transmit simultaneously. When a STA is contending for channel access, it has to wait for a DIFS period after the last transmission using carrier-sensing procedures, then it must defer for a Random Backoff Time. If the STA wins, it transmits a small frame named RTS, which has the Duration/ID set to the time required for the data frame transmission ($3SIFS + Time_{CTS} + Time_{DATA} + Time_{ACK}$). All the STAs that hear this RTS frame should set their NAV if it is lower than the Duration/ID field in the RTS frame. The intended receiver of the data frame should reply to the original transmitter with a small CTS frame after a SIFS period has elapsed, likewise, this frame sets the Duration/ID field to the necessary time to complete the transmission ($2SIFS + Time_{DATA} + Time_{ACK}$). All the STAs that hear the CTS frame update their NAV if lower than the Duration/ID value in the CTS frame. This RTS/CTS exchange ensures that the hidden STAs set their NAV, thus avoiding collisions. After receiving the CTS frame the sender (original STA) can transmit the data frame after waiting for a SIFS period. Finally, if the transmission was successful, the receiver waits for a SIFS period and then it acknowledges the sender by sending an ACK frame. Figure 2.5 shows the RTS/CTS procedure.

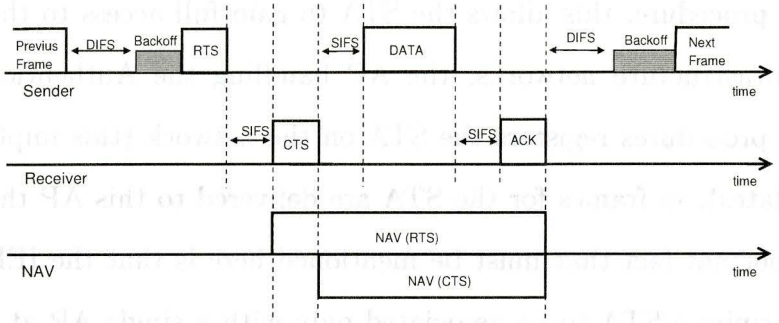


Figure 2.5: RTC/CTS Procedure and NAV Operation

Station Association Procedure

When a STA is just powered on and wants to initialize its network interface, the first step it has to do is to start scanning all the possible channels (this number depends on the physical layer being used). The IEEE 802.11 standard defines two scanning procedures: passive scanning, which involves listening on a given channel for a certain period of time for a special management frame known as the Beacon frame; the other procedure is active scanning and the difference is that instead of waiting for a Beacon frame, the STA transmits a Probe-Request frame, which should be replied to by a Probe-Response frame by the AP using the scanned channel. Regardless of the procedure used, the STA should obtain the following parameters: BSSID, SSID, BSSType, Beacon Interval, DTIM Period, BSSBasicRateSet, etc. After gathering these parameters, the next step the STA has to perform is the Authentication procedure which aims to identify the user trying to gain access to the network and vice versa. The IEEE 802.11 standard provides two schemes to accomplish this: Open System and Shared Key. Once authenticated, the final step the STA has to execute is the

Association procedure, this allows the STA to gain full access to the network (DS). In Infrastructure networks, the AP handling the Authentication and Association procedures registers the STA on the network (this implies L2/L3 routing updates), so frames for the STA are delivered to this AP through the DS. An important fact that must be mentioned here is that the IEEE 802.11 standard permits a STA to be associated only with a single AP at any given time.

2.1.3 Point Coordination Function (PCF)

As part of the original IEEE 802.11 standard, PCF was designed to provide contention-free frame transfer for infrastructure networks. The Point Coordinator (PC) functionality is implemented in the AP and has complete control over the medium during the Contention Free (CF) period. PCF implementation is not mandatory for STAs; therefore they can be categorized as CF-Pollable or Non-CF-Pollable STAs. The AP/PC keeps track of the CF-Pollable STAs using a polling list and it must at least poll once each of the STAs on the list every predefined interval (Superframe).

The CF period begins immediately after the AP/PC transmits a Beacon frame. All the STAs should set their NAV to the length of the CF period (CFMaxDuration), thus blocking Non-CF-Pollable STAs. Each STA is individually polled by the AP/PC and after this action they can transmit a single frame. Similar to DCF, MAC reliability is also present in PCF, hence frames must be acknowledged. The time gap between consecutive frame transmissions is given by the PIFS. PCF makes use of a more complex type of frames than

DCF, which allow piggybacking of acknowledgments and polling requests thus reducing the MAC overhead.

Several drawbacks, such as the lack of different traffic priorities and its failure to provide throughput/delay/jitter guarantees, impeded the take off of PCF. This is the main reason why many commercial APs and STAs do not support it.

2.1.4 Power Saving in 802.11 Infrastructure Networks

Since IEEE 802.11 networks allow STAs to be in any location within the coverage area, many STAs are mostly mobile devices powered by batteries. However, battery power is a scarce resource, hence the use of techniques that make more efficient use of power are needed. In this context IEEE 802.11 provides the Power-Saving (PS) functionality, which allows the STAs to periodically switch off their radios. A STA is considered to be in Doze/PS/Sleep mode when its radio is off and is considered Awake/Active/On when its radio is on. While a STAs is in PS mode, frames destined to it should be buffered to be delivered at later time when it is Active again. This functionality is implemented in the APs, which are responsible to not only buffer frames for doze STAs, but also notify periodically through Traffic Indication Map (TIM) messages about the existence of buffered frames. When associating with an AP, STAs accept the Listen Interval specified by the AP, which defines the maximum period of time the STAs can sleep. A PS-Poll frame is used by the STAs that have just woken up (i.e. Active STAs) to start retrieving buffered frames after being notified via TIM messages.

2.1.5 IEEE 802.11f Inter-Access Point Protocol (IAPP)

The IAPP [16] is a sub-standard of the IEEE 802.11 standard, its main objective is to define the interaction among APs which are members of the same ESS through the DS and to provide roaming capabilities to STAs. IAPP allows STAs to move from AP to AP while maintaining a network layer address (e.g. IP address) referred to as a STA context. When a STA that is associated with an AP moves to another AP, it should re-authenticate and re-associate to regain access to the network (DS). The old AP should transfer all the context of the roaming STA to the new AP and forward buffered frames that were received after the STA disassociated. Finally, the new AP must update the location of the roaming STA on the DS.

2.1.6 QoS in WLAN: the IEEE 802.11e standard

New enhancements to the original IEEE 802.11 standard are currently under development. One of them is the recently completed IEEE 802.11e standard [14]. This standard tries to overcome some problems related to the original standard with focus on the QoS aspects of the technology. In the new standard, APs supporting 802.11e are denoted QAPs and in the same way STAs are known as QSTAs. IEEE 802.11e introduces the Hybrid Coordination Function (HCF), which combines and improves characteristics of DCF and PCF. The concept of Transmission Opportunity (TXOP) is introduced, which is defined by a start time and maximum duration that a STA is allowed to transmit, and it can be used for sending single or multiple frames. HCF has two modes of operation:

the Enhanced Distributed Channel Access (EDCA) and the HCF Controlled Channel Access (HCCA). Both are improvements of the PCF and DCF access mechanisms respectively.

Basically, EDCA provides an improved contention-based access mechanism which is similar to the original DCF, but specifies up to 8 traffic priority levels, which are mapped to one of 4 access categories (transmit queues). Each queue may have different parameters such as wait-time before contending for the channel (TXOP), Arbitration Inter-Frame Spacing AFIS, maximum and minimum contention window size. The highest priority queue contending for the channel will gain control of it and might transmit multiple frames of the same category within the time defined by the TXOP. In other words, EDCA provides QoS by giving higher priority traffic a better opportunity to capture the wireless medium. However there are no guarantees because it relies essentially on CSMA/CA.

On the other hand, HCCA provides statistical guarantees for throughput, delay and jitter requirements. HCCA uses a polling mechanism similar to PCF but it schedules the transmission of frames based on traffic flow requirements of each STA. Hence, HCCA consists of contention free periods where the channel access is controlled by the Hybrid Coordinator (HC), which is no other than the AP. Unlike PCF, these periods can be started at any time. According to the Traffic Specifications (TSPECS) negotiated at the Connection Admission (CA) between the HC and the STAs, the HC schedules HCCA periods in such a way that the individual TSPECS are met.

Additional new features are introduced in the IEEE 802.11e standard such

as: The direct-link protocol (DLP), which allows direct communication between STAs without intervention of the AP for Infrastructure networks; Contention-Free Burst (CFB), which defines transmission of multiple frames within the time limits of the TXOPs and can make use of block acknowledgments. These new features are not explained further because they are outside of the scope of this thesis. It is important to point out that IEEE 802.11e ensures backward compatibility with legacy IEEE 802.11 STAs.

2.1.7 ESS Mesh Networks

The IEEE 802.11 Task Group S [12] is currently attempting to standardize 802.11 Extended Service Set (ESS) Mesh Networking. As its name suggests, the amendment will aim to define the interaction of APs interconnected using their IEEE 802.11 interfaces to form a mesh, thus creating a Wireless Distribution System (WDS). The IEEE 802.11s amendment will define an architecture and protocol for providing IEEE 802.11 Meshes which make use of the IEEE 802.11 MAC to support broadcast/multicast and unicast delivery at the MAC layer using radio-aware metrics over self-configuring multihop-topologies. The driving force behind this amendment is the increasing need for creating IEEE 802.11 networks that may be extended wirelessly simply by introducing more APs. ESS Meshes will also support a new class of IEEE 802.11 applications that require untethered/unlicensed infrastructure.

Although Task Group S has not released a draft to date, it is till possible to create IEEE Mesh Networks using proprietary technologies or by simply relying on the use of WDSs. With the expected widescale deployment of such mesh

networks especially in outdoor scenarios, the problem of providing the APs with power will grow due to the potentially high costs involved with achieving that. Thus, one possible solution would be to deploy APs that rely on renewable energy sources in order to provide sustained operation.

One example is the Solar Mesh Network [17] implemented at the McMaster University Campus, the network provides a contiguous coverage area for mobile STAs. It is comprised of solar-powered APs that are completely autonomous. More details about the architecture of these solar nodes will be provided in Chapter 3.

2.2 Related Work

Our literature survey revealed some previous work proposing load balancing schemes for overlapping APs. We found the algorithm presented in [10] by Velayos, Aleo and Karlsson to be the most appropriate to balance traffic, since it does not depend on a centralized server architecture and load balancing is done in a distributed manner without modifying the IEEE 802.11 standard. Although we found their proposal to be the most suitable for our problem, their experimental results relied on using constant rate UDP traffic from two sources with an algorithm execution cycle time of 100 ms for load balancing. In cases when bursty data traffic is used we found via simulation that their algorithm presents some problems such as increased handover rate.

The work done in [2] by Balachandran, Bahl and Voelker presents a centralized scheme where a central server collects the traffic load of each AP and

decides to which AP each STA associates. They combine this with user location schemes, such as RADAR [22], which might not be accurate enough in our opinion; moreover their scheme requires the installation of additional software on the mobile STAs.

A load balancing algorithm proposed by Sheu and Wu can be found in [27], but the authors' goal is to balance and maximize the average RSSI[1] between the mobile STAs and APs. Another load balancing proposal based on 4-sector antennas in the APs is introduced in [25] by Wang, Cuthbert and Bigham. This last scheme changes the propagation patterns of the overlapping APs in such a way that the STAs and load are distributed among them. Our approach differs from both schemes [27] and [25], thus, they are not applicable in our case.

There are some proprietary industrial implementations proposed by Cisco Systems and Proxim Wireless Networks [5], [24], but both require modifications to be made to the mobile STAs. They suggest adding load information in the beacon frames such as the number of users in the cell. The problems with both solutions are that they are proprietary and do not guarantee a uniform load distribution. Also, Nortel Networks [21] provides user load balancing via a centralized WLAN Security Switch, which continually monitors user load and automatically redirects new users to alternative APs to deliver the best possible user experience, however the mechanisms and criteria used to achieve this are not described.

---

1 RSSI is the received signal strength indication

Additionally, the IEEE 802.11e [14] standard allows the AP to transmit the QBSS Load element, which contains: the number of STAs associated with the AP, channel utilization and available admission capacity. The last variable is helpful for roaming QSTAs to select an AP that is likely to accept future connection admission requests (for contention free services), but it does guarantee that the QAP will admit these requests. Legacy STAs will not be able to take advantage of this information.

In [32], a connection admission control scheme with load balancing is proposed by Zou and Zhao for ESS Mesh networks, but it only focuses on contention free service provided via HCCA and no schemes are defined to manage contention-based traffic.

It is important to point out that our target is different to that of previous authors since we are not only trying to distribute the traffic load to maintain a low transmission delay, but also to *increase* the lifetime of the SNs and *decrease* their hardware requirements (solar panel size and battery capacity). This essential requirement leads to additional constraints while formulating our problem and developing the algorithms to solve it.

Chapter 3

System Description

3.1 Problem Formulation

The IEEE 802.11b/g standard [13] specifies the use of up to 3 non-overlapping frequency channels when implementing Direct Sequence (DS) in the physical layer, on the other hand, the IEEE 802.11a standard allows for 12 non-overlapping channels, 8 dedicated to indoor and 4 to point to point.

Although the IEEE 802.11a seems to be more suitable for overlapping deployments, it has not been widely adopted in the consumer sector like IEEE 802.11b/g, consequently, if backward compatibility is desired, the implementation would be constrained to 3 non-overlapping channels. The technology selected would dictate the maximum number of SNs that can be deployed. However, the actual number of SNs needed will depend on the traffic demand on the hot-spot. It is possible to increase the number of channels used by deploying channels that have partial overlaps, however this will require some

advanced frequency assignment schemes to be implemented in order to minimize the interference.

We can assume that through proper network design the number of SNs needed in the hot-spot has been determined beforehand using as a design criterion the maximum traffic load that should be supported. When more than one SN is providing access to STAs, it is desirable to distribute as uniformly as possible their load capacity and decrease the transmission delay in the hot-spot. To make this possible we need to avoid having the SNs operating below but close to saturation state. Therefore, the need for load balancing/distribution schemes is evident. On the other hand, we have to consider that the capacity provided by these SNs is only used to satisfy the demand during heavy load hours. Hence, at some intervals during a given period, not all the SNs will be needed (i.e. probably during the night) to provide access to STAs. In this case, when a SN is not required, it should sleep (change to doze mode), thus saving in this way energy. We will present in Chapter 4 our two proposed algorithms that address this problem.

It is possible to formulate our problem as an optimization problem. Our objective in this case would be to minimize the average power consumption of the SNs, denoted by P_{SN}, subject to constraints on system performance such as mean packet delay and handover rate. More formaly, we can express this as:

$$min\{P_{SN}\} \tag{3.1}$$

subject to

$$\bar{d} \leq D_{max} \tag{3.2}$$

$$\bar{h} \leq H_{max} \tag{3.3}$$

Maintaining the mean packet delay denoted by $\bar{d}$ below a maximum average delay denoted by D_{max} and additionally trying to reduce the average handover rate denoted by $\bar{h}$ (needed to achieve this objective) below a maximum handover rate denoted by H_{max}, are the main constraints of this problem. In Equation 3.2, $\bar{d}$ depends mainly on the packet arrival process from the SN point of view, which is a random variable denoted by λ. This problem is very difficult to solve since the arrival process is mostly random for each STA and very hard to model accurately (using well known distributions). There are also other factors that increase the complexity of this problem such as the presence of hidden STAs, existence of power saving functionality in the STAs (unpredictable awakening/sleeping patterns), channel interference, random number of STAs in the hot-spot and the fact that each STA can only be associated to one SN at the same time. In addition to these factors we have to mention that from the perspective of the SNs it is not possible to measure the actual access delay as perceived by their associated STAs (waiting time of the STA to gain access of the medium and successfully transmit a packet), thus an alternative variable must be used instead of $\bar{d}$. In our work, we consider the average channel utilization, denoted by $\bar{U}$, as this alternative variable. We provide an analysis later in Chapter 5 to estimate average channel utilization thresholds to avoid working in saturation state, which produces an undesirable high transmission delay.

If we try to simplify this problem into two sub-problems, the first would be to distribute/balance the load to satisfy mean delay requirements and the

second would be to determine the minimum number of SNs such that the SNs can support a maximum fixed capacity.

In the first case, when we have more that one SN providing service one way to reduce the transmission delay is to try to avoid overloading any of the SNs through load balancing/redistribution schemes. Also, this *efficient* load redistribution permits us to make a better use of the capacity of each SN, thus not increasing the number of SNs needed unnecessarily. Indeed, the problem of balancing the load equally among the SNs can be modeled as an instance of the PARTITION problem described in [7]. The PARTITION problem description consists of a finite set A and a size $s(a) \in \mathbf{Z}^+$ for each $a \in A$. The problem to address is finding a subset $A' \subseteq A$ such that $\sum_{a \in A'} s(a) = \sum_{a \in A-A'} s(a)$. In our case, the fraction of time that each STA is utilizing the medium denoted $t(a)$ is equivalent to the size $s(a)$ and the objective is obtaining an equal (or almost close) utilization/load distribution among the SNs (the model is intended for two equal partitions of the subset A, but it could be easily extended to a greater number of partitions). The PARTITION problem is NP-complete, thus the use of heuristic solutions to solve it is justifiable.

Given that the system is operating under certain conditions, we would like to know the minimum number of SNs required. By obtaining this we are also reducing the power consumption of the SNs because some of these SNs would not be required, thus being eligible to use power saving mechanisms. Let us assume that the average time that each STA is utilizing the channel on each SN is fixed and denoted $t(a)$. Determining the minimum number of SNs needed to provide service can be formulated as another optimization problem. We

want to service all the STAs in the hot-spot using the minimum number of SNs given that each of them can only support a finite traffic load that guarantees our delay constraint. In this case we are dealing with another NP-complete problem referred to in the literature [7] as the BIN PACKING problem. In our case, the problem consists of a finite STA population A, the time each STA utilized the channel (in the last measurement) $t(a) \in R^+$, a SN capacity C (that satisfy our delay constraint), and a number of SNs denoted by K. The objective is to find a way to partition A into disjoint sets $A_1, A_2, ..., A_K$ such that the sum of the utilization $t(a)$ of each STA in each set A_i is C or less and using the minimum possible K (i.e. minimum number of SNs). This problem is also NP-complete [7]. Therefore, approximation algorithms are also best suited to solve it. One algorithm available to solve this problem is the *Best Fit Decreasing* algorithm. One of our proposed schemes is based on a modification of this well known algorithm. The *Best Fit Decreasing* solution is guaranteed never to be more than 22% worse than the optimal solution, more information about it can be found in [11].

Although these two sub-problems are NP-complete it is important to remember that they are just simplifications of our main optimization problem. Some approximation algorithms can be found in the literature to solve these sub-problems. However, they can be solved based on the *last measured utilization* of the system. In fact this utilization depends on the traffic characteristics (packet arrival process λ) of each STA, which is indeed random and difficult to estimate as mentioned before. Therefore, depending on the characteristics of these individual traffic flows the solution of this problem (number of SNs) will

vary in time i.e., for each different *snapshot* of the system at any time a different number of SNs can be required. In Chapter 6 we study the performance of our solutions under different traffic models just to provide an idea that differences in results can be obtained depending on the traffic model selected. We also need to avoid having thrashing effects (i.e. awakening and sleeping the same SN repeatedly), thus our solution should take this into consideration through providing some sort of hysteresis mechanisms. In addition, the capacity of each SN is not constant in time due to the presence of CO traffic. Even though this CO traffic is serviced in a quasi-deterministic manner the connection-request arrival process is completely unknown.

An additional constraint for both of the above-mentioned sub-problems is to reduce the number of STA handovers caused by these processes in order to improve the end-user experience as mentioned for our main optimization problem. Since these two sub-problems are simplifications of our main problem and both are NP-hard ([7]), this confirms that our main problem is very difficult to solve.

Clearly, the problem of determining the minimum number of SNs to satisfy performance constraints has a high complexity. Moreover, our literature survey presented in Section 2.2 revealed none of the previous solutions that dealt with the load balancing problem in IEEE 802.11 networks addressed the underlined optimization problem. Another issue to consider is the computational overhead caused by trying to solve this problem optimally. Since our objective is to reduce the power consumption of the SNs, any proposed solution should take into consideration the running time of the algorithm and its power con-

sumption. The schemes to be proposed later in Chapter 4 for this scenario are indeed heuristic or approximate solutions for this problem.

3.2 Solar/Battery Powered Access Points: Solar Nodes

One special implementation of APs are solar and battery powered APs powered known as solar nodes (SN). These SNs lack of wired connections to power sources and the distribution system, thus being complete tetherless devices. This property permits the deployment of SNs in a wide range of scenarios which are very difficult for conventional APs.

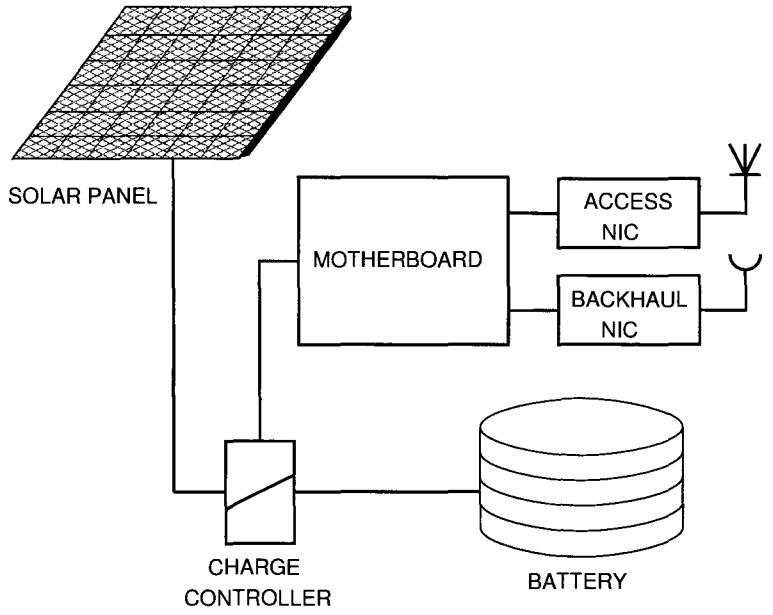


Figure 3.1: Solar Node Components

Figure 3.1 shows the basic components of a SN. During daylight the solar panel should be large enough to feed the motherboard and network interface cards (NICs), and also recharge the battery. When solar radiation levels are low or zero (i.e. at night), the battery provides energy to all the SN circuits. The power consumption of SNs depends on the the traffic load they support. Therefore a conscious SN design should take into consideration the solar radiation in the place where the SNs would be deployed and also the expected traffic profile. Dimensioning correctly both components (battery and panel) is really a challenging work, even more if we consider that is very difficult to predict traffic profiles accurately. Nevertheless, a thorough methodology to implement SNs based on these variables can be found in [6]; moreover, a control scheme to control excessive load and/or unexpected lack of solar energy is also proposed there. We discuss in Chapter 6 this methodology and we also make use of it to design SNs under certain traffic conditions.

3.3 System Architecture and Definitions

The two schemes to be proposed have some similar features such as:

- The coverage area of all the SNs is almost the same, we define a service area named the **Joint Coverage Area** as illustrated in Figure 3.2. This represents the zone where STAs are supposed to be served by the overlapping SNs. Since we are not considering using user-location based schemes, some STAs out of the Joint Coverage Area might get service, but this does not represent a problem unless an attempt is made to trans-

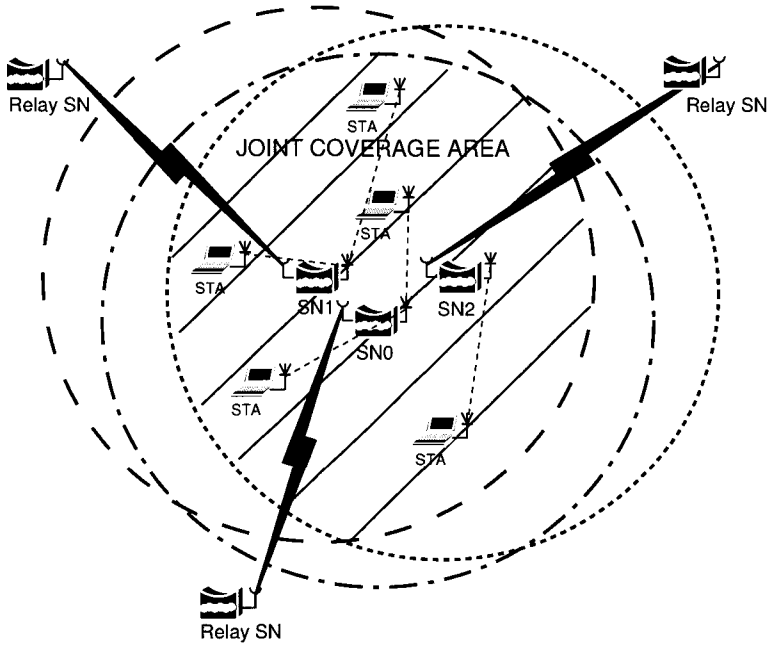


Figure 3.2: System Architecture

fer them to another SN. If this occurs, the STA will lost service, which is acceptable since the STA is out of the Joint Coverage Area.

- We define the concept of a **neighbor**, all the SNs form neighboring relationships with the other SNs in the hot-spot. Also, the concept of **neighborhood** represents all the SNs in the hot-spot that have neighboring relationships.

- The SNs in the hot-spot are operating on different non-overlapping frequency channels. Therefore, each SN can make use of all the channel capacity in terms of load.

- We assume that the SNs have two radio interfaces, one to provide access to

STAs and the other to connect to the Distribution System (DS) via Relay nodes. The interface between the SNs and the relay SNs could be IEEE 802.11a using directional antennas, IEEE 802.16, or even proprietary.

- Our work focuses only on single-hop networks (the access link between SNs and end STAs). The end-to-end delay added by the ESS mesh network as well the transmission delay of the relay links are not considered part of the scope of this thesis.

- We define 2 possible states for the SN operation: in **ACTIVE mode**, the SN is providing access to STAs and all its interfaces are operating; in **IDLE mode**, the SN has its radios and dispensable circuits switched off, thus minimizing its power consumption (also know as **SLEEP mode**).

- When a SN is in IDLE mode it associates with another of the ACTIVE SNs just like a STA. IDLE SNs should use the power saving mechanisms previously described in Section 2.1.4, in this way it can save considerable energy and communicate with its neighbors using its serving SN as a relay node. Figure 3.4 shows this scheme.

- A communication protocol must be defined to exchange information among SNs such as: traffic load, battery-capacity level, estimated SN lifetime, disassociated STAs, etc. We propose software agents running on the SNs to transmit, receive and process such information.

- We consider that the system could be using IEEE 802.11e, where the timeline is divided into a Connection Oriented (CO) subframe serviced

via HCCA, and the Best Effort (BE) subframe serviced via EDCA. In fact as explained in Section 2.1.6, IEEE 802.11e specifies that both the HCCA and the EDCA period can start at any time depending on the HC (SN) decision, therefore the duration of the CO subframe and the BE subframe are the summation of HCCA periods and EDCA periods over an inter-beacon interval (superframe) respectively. Figure 3.3 illustrates the CO and BE subframes.

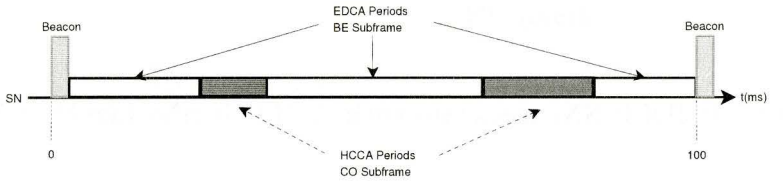


Figure 3.3: The HCCA and EDCA periods define the length of the BE subframe and CO subframe respectively

3.3.1 Parameter Definitions

In this section, we define the parameters that we will be using throughout the rest of this thesis:

- T_W: Measuring interval used to sleep and awaken SNs. It defines the period of time over which the decision is made to change the number of ACTIVE SNs. The larger this parameter is, the less reactive the system is to traffic load fluctuations.

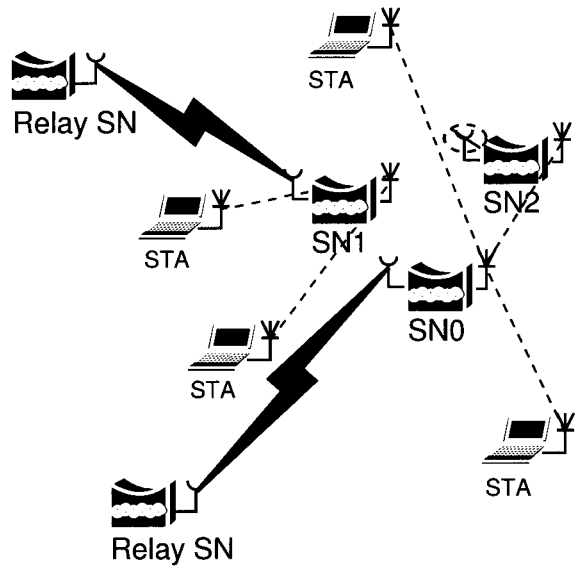


Figure 3.4: IDLE SNs associate with ACTIVE SNs like PS-STAs

- $T_{interbeacon}$: Interbeacon period or superframe. It represents the period of time between two consecutive transmissions of Beacon frames by the same SN.

- T_{CO}: Summation of the HCCA periods in a superframe. It indicates how much time in a superframe was allocated for CO service.

- T_{BE}: Summation of the EDCA periods in a superframe. It is the complement of T_{CO}, representing the time used for BE service.

- t_{BE}^{k}: Period of time on the BE Subframe in which the STA k has been utilizing the medium.

- t_{CO}^{k}: Period of time on the CO Subframe in which the STA k has been utilizing the medium.

- U_{BE}: Channel utilization during the EDCA periods normalized to T_{BE}. This utilization includes the collisions and effect of hidden STAs.

- U_{BEmax}: Maximum U_{BE} threshold, at which the BE subframe of the SNs should operate.

- U_{CO}: Utilization of the HCCA periods normalized to T_{CO}.

- U_{COmax}: Maximum U_{CO} threshold, at which the BE subframe of the SNs should operate. This parameter could be quite close to *one* since the HC has complete control of the CO subframe resources and therefore, as explained in Chapter 2, effects of contention and collision are not present.

- H_{factor}: Hysteresis factor used when deciding to change SN(s) to IDLE mode.

- $E_{threshold}$: Maximum difference in battery-capacity level between the ACTIVE SN with lowest battery-capacity level and the IDLE SN with the best battery-capacity level that the system can tolerate. Also, if one ACTIVE SN is draining energy faster than other ACTIVE SNs, $E_{threshold}$ can be used as a criterion to redistribute traffic load. A detailed explanation of how this parameter is used will be provided in Chapter 4.

- $U_{BEthreshold}$: Maximum capacity (under saturation) of the BE subframe. When the BE subframe is operating above this parameter, the transmission delay increases considerably. As a consequence, the performance degrades also. A complete analysis to determine this parameter is described in Chapter 5.

3.3.2 Station Transfer Procedure

We need to distribute the load by transferring the STAs from SN to SN. The most straightforward way to do this is by disassociating STAs by sending the standard Disassociation Notification messages. This will force the STAs to re-associate with another SN using the procedures explained in Section 2.1.2. We can assume that in a normal ESS handover, the new SN should communicate to a central AAA server or to the old serving SN to re-authenticate and reauthorize the roaming STA. However, we have the advantage of knowing in advance what SN(s) will serve the disassociated STA since neighbors are aware of each other's loading and energy status. Once a SN disassociates a STA, it should notify its peers that such STA is roaming to another SN, so the new SN will know beforehand the identity of the roaming STA. This could also facilitate context transfer between SNs. If we compare this scheme to the standard IAPP protocol introduced in Section 2.1.5, at least one step (MOVE-NOTIFY message) is saved.

An additional option would be to provide the roaming STAs with the list of SNs that could support their traffic demands. Since the SN that is issuing the disassociation message knows beforehand the SNs with available capacity, it might transmit this information to the mobile STA in a proprietary disassociation message. However, this requires a modification of the IEEE 802.11 standard and an upgrade to the STAs, therefore this option might not be feasible at this time because our main priority is to maintain backward compatibility.

Even though the IEEE 802.11 family of standards allows STAs to roam from BSS to BSS within the same ESS, there are no QoS guarantees for appli-

cations that are sensitive to packet loss and delay such as voice and video traffic. This means that when a STA roams from SN to SN, the time required by the process may be too long, hence real-time sessions might be torn down. The International Telecommunication Union (ITU) [15] has placed recommendations for maximum acceptable one-way delay limits for voice communications, which are shown in Table 3.1. On the other hand, data applications are usually resilient to long delays that may be caused by handovers. Normally, data applications rely on TCP/IP mechanisms, which provides reliable transmissions. From Table 3.1 we can infer that the end to end delay can increase to a maximum of 400 ms during handovers. Therefore, the available delay budget we have is approximately 250 ms in the worst case scenario, if we consider that the one-way delay between the STA and the other party of the voice call being handed over should be less than 150 ms.

	Voice Communication
Acceptable Delay (ms)	0 - 150
Tolerable Delay (ms)	150 - 400
Unacceptable Delay (ms)	Above 400

Table 3.1: ITU G.114: One-way Delay Thresholds

Some measurements of the delay time in handovers for different NICs were done in [30]. These results are shown in Table 3.2. The detection stage represents the period of time it takes a STA to realize that there is a need to roam to another SN. The search stage includes the time spent in searching for another SN using active or passive scanning processes. The third part is the execution

phase, which represents the time needed by the STA to re-associate with the new SN. Altough the total values exceed on average our delay budget, we have to consider that the detection phase is not present in our scheme, because the STAs are disassociated regardless of the quality of their links. Therefore, only the search and execution phases should be considered. This yields an average handover time of 186 ms if we assume that the re-association request is accepted at the first attempt. Otherwise, each additional attempt adds in average 2ms the handover delay. Clearly, this falls within our original delay budget of 250 ms. Consequently, we can affirm that voice communications can support the station transfer procedure.

NIC	D-Link 520	Spectrum 24	ZoomAire	Orinoco
Detection	1630ms	1292ms	902ms	1016ms
Search	288ms	98ms	263ms	87ms
Execution	2ms	3ms	2ms	1ms
Total	1920ms	1393ms	1167 ms	1104ms

Table 3.2: Handover delay for different NICs

The idea of sending disassociation messages was also used in the experiments described in [10]. However, it is worth mentioning that in those experiments the handover time was 3 seconds. The authors attributed this to a bug in the firmware of the NICs of the mobile STAs (after receiving a disassociation message, the mobile STAs disconnected from the network instead of re-associating with another AP).

Finally, it is worth mentioning that the IEEE 802.11 standard does not spec-

ify the behavior of disassociated STAs. Therefore, the case of STAs trying to re-associate with the old SN is mainly dependent on practical implementations.

3.3.3 Communication Among the Solar Nodes

In our schemes, information exchange among the SNs is essential. Every SN communicates periodically the following information via broadcast messages to all the neighboring SNs: battery-capacity level(Ah), expected SN lifetime (sec), service mode (ACTIVE or IDLE), load state (overloaded, balanced, underloaded), T_{CO}, T_{BE}, throughput, U_{CO}, U_{BE}, number of STAs connected. Additionally, the SNs broadcast which STAs have been disassociated during the last algorithm execution. This information, received by all the neighbors, is used to distribute the load and modify the number of ACTIVE SNs if necessary.

3.3.4 Connection Admission Control (CAC)

The current IEEE 802.11e standard considers a basic Call Admission Control procedure, which does not take into consideration overlapping deployments and load balancing. We consider the case where the SNs differentiate between CO service (using HCCA) and BE service (using EDCA).

Every SN knows the $U_{CO}, U_{BE}, T_{CO}, T_{BE}$ of its neighboring SNs gathered via broadcast messages. When a STA tries to establish a new connection, it sends a service request including information such as: bandwidth, delay and jitter. Since the details of the IEEE 802.11e scheduler are outside of the scope of this thesis, we will assume the requested bandwidth will be the only parameter

considered by the SN for admission control purposes. The bandwidth requested should be translated by the SN into a corresponding time allocation, $T_{request}$, in the CO subframe. The SN handling the request evaluates the effects of accepting the new connection:

$$T'_{CO} = T_{CO} + T_{request},$$

$$T'_{BE} = T_{BE} - T_{request},$$

$$U'_{BE} = \frac{U_{BE}T_{BE}}{T'_{BE}} \tag{3.4}$$

If $U'_{BE} < U_{BEmax}$, then the SN accepts the connection. Otherwise, the connection request is rejected and the STA must try to associate with another SN. If the SN has rejected the request and it knows (via the received broadcast message) that at least one neighboring SN is able to accept this request, then the SN handling the request might attempt to transfer the STA to this other SN; if no other ACTIVE SN can accept the request at this time, then the SN handling the request could command an IDLE SN to ACTIVE mode (assuming there is one available that is capable of changing its status). Then it could transfer the STA to this new SN. In the worst case, if neither the SN processing the request nor any neighbor are able to handle the request, it would be rejected and the STA would not be able to establish the connection at this time.

3.3.5 IDLE Solar Node Scheme and Hot-Spot Outage Control Procedure

We also study the behavior of the hot-spot if one of the SNs fails (i.e. outage). We know that when a SN is in IDLE mode it must be associated just

like another STA to an ACTIVE SN. This IDLE SN should use an aggressive power saving scheme as defined in [13] and only wakeup to exchange broadcast messages every T_W period of time with its serving SN. If the IDLE SN wakes up and does not receive any beacon from its serving SN for 3 consecutive times, then it has to change its state to ACTIVE mode and start providing service. This scheme provides redundancy mechanism under an outage in the hot-spot.

In addition, ACTIVE SNs should relay information received from other ACTIVE SNs to IDLE SNs; also in the opposite direction, ACTIVE SNs with associated IDLE SNs should share information, such as battery-capacity level, with all other ACTIVE SNs. In this way, we ensure that all SNs know the battery-capacity level of their ACTIVE or IDLE neighbors.

Chapter 4

Proposed Solutions

In this chapter we introduce both of our proposed schemes. The first one is based on the load balancing scheme proposed in [10] and the second uses the the *"Best Fit Decreasing"* algorithm describe in [7].

4.1 Solution 1: Persistent Load Balancing & Awakening/Sleeping (PLAS) Algorithm

This scheme is based on the work from [10]. In particular, we use their load-balancing algorithm while modifying it to allow for the sleep/wakeup process of the SNs. This scheme considers only BE service using EDCA, thus $T_{CO} = 0$, for all the SNs. The ACTIVE SNs should always trying to equalize the load among them.The authors in [10] introduce a parameter β which indicates how well the load is balanced in the neighborhood. The expression to calculate β is

given by:

$$\beta = \frac{(\sum B_i)^2}{N_{ACTIVE} \sum B_i^2} \tag{4.1}$$

Where N is the number of ACTIVE SNs and B_i is the throughput of the SN-i. β can take values from 0 to 1 and when it is equal to 1, it represents a perfect load balancing among the SNs (assuming all the SNs have the same throughput). The objective of this algorithm is to maximize the the value of β all time.

Although β indicates how well the load is balanced, the authors used another parameter known as the average load indicator L defined by:

$$L = \frac{\sum B_i}{N_{ACTIVE}} \tag{4.2}$$

This parameter L represents the exact share of the load that corresponds to each ACTIVE SN. According to the value of L the SNs determine their load state. The SNs with throughput $B < L$ are considered underloaded and are eligible for receiving more STAs; the SNs with load above L are categorized as balanced or overloaded. A balanced SN is one with $L \leq B \leq 1.1L$, while an overloaded SN is defined as having $1.1L < B$. SNs that are overloaded should reject one STA every time they run this algorithm until their load state changes to balanced or underloaded. The window $[L, 1.1L]$, which indicates a balanced state, gives stability to the algorithm, because in practice a perfect load distribution is difficult to achieve because STAs can only associate to one SN.

To determine L, every ACTIVE SN measures the throughput B over a $T_{interbeacon}$ period of time and then broadcasts it to its neighbors. In this way, all

ACTIVE SNs know the throughput B of each other and are able to determine L, and thus their load state. If a SN is overloaded, it should disassociate only one STA, which is the one with throughput closest to the difference $(L - B)$. If the SN is underloaded, then it will accept roaming STAs from other SNs. Finally, if the SN is balanced then it should only accept new STAs joining the hot-spot. After running this algorithm, the SNs broadcast the SN information as specified in Section 3.3.3.

Now, we introduce some modifications to the original algorithm to handle ACTIVE and IDLE SN states. First, all SNs, regardless of their service state, calculate every T_W period of time the average BE utilization denoted $\overline{U_{BE}}$ and given by:

$$\overline{U_{BE}} = \frac{\sum U_{BE}^{i}}{N_{ACTIVE}} \tag{4.3}$$

If $\overline{U_{BE}} > U_{BEmax}$, then a new SN is needed to lower the utilization and this SN is the IDLE SN with the highest battery-capacity level ($E_{highest\_IDLE}$). It must change to ACTIVE mode by itself. In case there is no other IDLE SN that can wake up, the hot-spot has to operate above U_{BEmax}. On the other hand, if $\overline{U_{BE}} < U_{BEmax}$ it may be possible to sleep one of the ACTIVE SNs. To evaluate this, the ACTIVE SN with lowest battery-capacity level (E_{lowest_ACTIVE}) has to first estimate $\overline{U_{BE}}'$ using equation 4.3, but over $(N_{ACTIVE} - 1)$ SNs instead of N_{ACTIVE}. If $\overline{U_{BE}}' \leq U_{BEmax} H_{factor}$ then this SN disassociates all its STAs and switches to IDLE mode. The H_{factor} provides a hysteresis margin to the wakeup/sleep criteria, thus we avoid sleeping a SN and waking it up it for short periods of time which leads to many unnecessary handovers.

Additionally, when the hot-spot has been operating under an almost con-

stant load for a long time, there is a chance that one of the ACTIVE SNs will have the lowest battery-capacity level (E_{lowest_ACTIVE}). Therefore, it would be better to switch this SN with one of the IDLE SNs with better battery-capacity level ($E_{highest\_IDLE}$). To evaluate the candidate SN for awakening, we use the parameter $E_{threshold}$ defined in Section 3.3.1. When the difference between the lowest ACTIVE SN and the highest IDLE SN is greater that $E_{threshold}$, then both SNs have to switch their service modes. The ACTIVE SN disassociates all its STAs and changes to IDLE as soon as it detects that the IDLE SN is awake. This procedure is executed periodically every T_{switch} time units and aims to maintain similar battery-capacity levels on all the SNs.

A pseudocode description of the PLAS algorithm can be seen in Figures 4.1 and 4.2. Performance evaluation of PLAS algorithm is presented in Chapter 6.

4.2 Solution 2: Load Distribution & Awakening/Sleeping (LAS) Algorithm

According to our literature survey, this scheme has never been previously proposed for load balancing/distribution. The algorithm is based on the *"Best Fit Decreasing"* algorithm introduced in Section 3.1. Figure 4.3 describes the *Bin Packing Based Algorithm* used for the LAS algorithm. This algorithm is indeed a modification of the *"Best Fit Decreasing Algorithm"* to handle different "bin sizes" related to the remaining capacity ($(U_{BEmax} - U_{BE})T_{BE}$) of each ACTIVE SN.

This solution is also capable of supporting both CO and BE services. The

```
1  if T_interbeacon timer expired
2  begin
3    if SN-i in ACTIVE mode
4    begin
5      Calculate Load: B_i
6      Compute: L
7      if B_i > 1.05L %Overloaded
8      begin
9        Reject new roaming stations
10       Find best candidate: STA closest to (B_i − L)
11       Disassociate candidate
12     end
13     else if B_i < L %Underloaded
14       Accept new and roaming STAs
15     else %Balanced
16       Accept new STAs
17   end
18 end
19
20 if T_W timer expired
21 begin
22   Calculate U_BE (mean)
23   if SN-i is E_lowest_ACTIVE
24   begin
25     if U_BE (mean)' < H_factor U_BEmax
26     begin
27       Disassociate all STAs
28       Change to IDLE mode
29     end
30   end
31   else if SN-i is E_highest_IDLE
```

Figure 4.1: PLAS Algorithm

```
32    begin
33      if $\overline{U_{BE}} > U_{BEmax}$
34        Change to ACTIVE mode
35    end
36  end
37
38  if $T_{switch}$ timer expired and SN-i is $E_{lowest\_ACTIVE}$
39  begin
40    if $[E_{highest\_IDLE} - E_{lowest\_ACTIVE}] > E_{threshold}$
41      Wait for highest IDLE SN to awake, then change to IDLE mode
42  end
43
44  Transmit SN-i information to neighbors
```

Figure 4.2: PLAS Algorithm (cont.)

main idea behind this algorithm is to transfer STAs only when is really needed. The question to answer is "when is it really necessary to transfer STAs?". First of all, since the Hybrid Coordinator has complete control over the CO subframe, the problem mainly resides in the BE subframe which works with contention based access mechanisms. It was shown in [28] that while $U_{BE} \leq U_{BEthreshold}$ the transmission delay is low and tolerable for delay-sensitive applications, moreover it does not increase significantly when the load increases, but as soon as U_{BE} gets close to $U_{BEthreshold}$ the transmission delay rises very rapidly reaching unacceptable values. $U_{BEthreshold}$ is the saturation threshold for the BE subframe introduced in Section 3.3.1, which is analyzed and calculated later in Chapter 5.

Unlike the PLAS algorithm, this algorithm is executed every T_W seconds

by each SN. A DISASSOCIATION_TOKEN (DT), which is passed from SN to SN in a round-robin fashion using broadcast messages, is used to grant the right of disassociating STAs to SNs. Every algorithm execution cycle (T_W), the SN that has possession of the DT can transfer STAs to other SNs if it wants to. The idea behind the DT, is to ensure that only one SN is transferring STAs at any time, and therefore avoid having two or more SNs redistributing load (moving STAs) simultaneously to any other SN, or even worse between them. Without the DT, SNs might get unexpectedly overloaded. All SNs are aware of which SN has the DT. Thus, if the next SN expecting to receive the DT did not receive it within a certain time, then it should request the DT directly from the SN that has it. If there is no answer, the next SN considers this an outage and takes possession of the DT (announcing this to other SNs). In this way we address the case where the DT might be lost.

For ease of notation, the SN that has the DT will be denoted SN-i. All ACTIVE SNs run this algorithm (IDLE SNs only broadcast their battery-capacity level). They measure and communicate U_{BE} every T_W period using broadcast messages as defined in Section 3.3. In addition, the SN-i depending on the value of U_{BE} can take the following actions:

1. The main objective is always operate below U_{BEmax}. Therefore, if $U_{BE} > U_{BEmax}$ the next inequality expresses the target at this stage:

$$U_{BE}^{i'} \leq U_{BEmax} \tag{4.4}$$

where $U_{BE}^{i'}$ represents the future value of U_{BE}^{i}. There are two options to bring down $U_{BE}{}^{i}$ below U_{BEmax}:

```
1  %Input Parameters:
2  %Bins Vector → D_j
3  %Boxes Vector → X_k
4  %Objective → Target
5  Create list of X_k → List
6  while Target > 0 and Sizeof(List) > 0
7  begin
8    Find the biggest element on List: X_k
9    Find the emptiest bin: D_j
10   if X_k ≤ D_j
11   begin
12     D_j = D_j − X_k
13     Target = Target − X_k
14     Add (k, j) to Tuples(k, j)
15     Remove k from List
16   end
17   else
18     Remove k from List
19 end
20 if Target > 0
21   Tuples(k, j)= Null
22
23 Return Tuples(k, j)
```

Figure 4.3: Bin Packing Based Algorithm

(a) **TEST 1: Evaluate transferring BE-only Stations** Transferring STAs that are only using BE services is more suitable because CO sessions are likely related to real-time applications, which as explained in Section 3.3.2 are sensitive to packet loss and delay. The required amount of load to be transferred is denoted α and is given

by:

$$\alpha = (U^i_{BE} - U_{BEmax})T^i_{BE} \tag{4.5}$$

Also the SN-i has to determine how much load can be transferred to neighboring ACTIVE SNs (denoted j). We denote this quantity as γ, which is given by:

$$\gamma = \sum_{j=1}^{N_{ACTIVE}} (U_{BEmax} - U^j_{BE})T^j_{BE}, \qquad \forall j \neq i \tag{4.6}$$

where if $U_{BEmax} < U^j_{BE}$ for a SN-j then the contribution of this SN to the previous summation is zero.

If $\gamma > \alpha$, then SN-i is able to evaluate redistributing part of its load. In order to do that it calculates for each ACTIVE SN the maximum load that it can receive. This expression is given by:

$$C(j) = (U_{BEmax} - U^j_{BE})T^j_{BE}, \qquad \forall j \neq i$$
$$\text{if } C(j) < 0 \Rightarrow C(j) = 0 \tag{4.7}$$

then SN-i looks for the BE-only STA with greatest t^k_{BE} that can be handled by the SN with greatest $C(j)$. If there is no SN that can handle this STA, then the STA is considered non-transferable and is removed from the list of STAs that can be transferred. Otherwise, if one SN can accommodate the STA, the STA is removed from the list and the ***tuple*** STA-SN is recorded (for future disassociation) and $C(j)$ is reduced by t^k_{BE}. This process continues, without overloading any SN (making $C(j) < 0$), until:

$$\sum t^k_{BE} > \alpha, \qquad \text{of all selected STAs} \tag{4.8}$$

or, the SN-i evaluated moving all the STAs without satisfying Inequality 4.8. If this test fails, SN-i continues with the next test. Otherwise, it skips it.

(b) **TEST 2: Transfer any station**

Now SN-i evaluates transferring STAs regardless of the fact that they may have CO sessions (all STAs are treated equally). In this case, each time SN-i transferring a STA can reduce its U_{BE} in two ways: First, increasing T_{BE} by reducing T_{CO} by t_{CO}^{k}; second, reducing the load over the BE subframe, by moving a STA that is consuming t_{BE}^{k} time units. Where k represents the transferred STA. The following equation describes this:

$$U'_{BE} = \frac{U_{BE}T_{BE} - \pi}{T_{BE} + \epsilon} \tag{4.9}$$

where π is the total time usage t_{BE}^{k} on the BE subframe of the transferred STAs and ϵ is the total time allocation t_{CO}^{k} on the CO of them. Also in this case the objective is to make $U_{BE}^{i'} < U_{BEmax}$. Hence using Equation 4.9 we can formulate the following inequality:

$$\pi + \epsilon U_{BEmax} \geq (U_{BE}^{i} - U_{BEmax})T_{BE}^{i} \tag{4.10}$$

Using this inequality, SN-i calculates for each of its associated STAs:

$$v(k) = t_{BE}^{k} + t_{CO}^{k}U_{BEmax} \tag{4.11}$$

Now with $v(k)$ SN-i can select which STAs disassociates. Similar to **Test 1**, SN-i evaluates moving STAs to the neighboring SNs.

It begins looking for the STA with the highest $v(k)$ and searches through all of the ACTIVE SNs starting from the one with the highest $C(j)$ (capacity available) in descending order until it can find one that can support the STA demands (t^k_{CO} and t^k_{BE}) without being overloaded (exceeding U_{BEmax}). If a SN capable of handling this STA is found, a STA-SN ***tuple*** is created, $C(j)$ is updated for this SN-j and the STA is removed from the list. Otherwise, the STA cannot be transferred and is removed from the list anyway. SN-i goes on picking the next remaining STA with highest $v(k)$ and evaluating transferring it to the SN with highest $C(j)$ that can accept it just like in the previous step. The test continues until Inequality 4.10 is satisfied or SN-i is unable to move more STAs because it is not possible to redistribute the load of SN-i. If the test is successful, then SN-i will disassociate STAs at the end of the algorithm.

2. In case both **Test 1** and **Test 2** failed and U^i_{BE} is still above U_{BEmax} then SN-i commands the IDLE SN with best battery-capacity level ($E_{highest\_IDLE}$) to change to ACTIVE mode, denoted SN-h. Once SN-h is ACTIVE, SN-i shares its BE load with it. To do this the SN-i runs **Test 1** as described before but with a slight difference in the objective: Instead of just reducing U^i_{BE} below U_{BEmax}, now it looks for:

$$U^{i'}_{BE} \leq \frac{U^i_{BE}}{2} \tag{4.12}$$

Which means splitting the BE load between SN-i and the new SN.

3. If $U_{BE}^{i} > U_{BEmax}$ and there is no other IDLE SN that could be woken up, then the SN-i should try to equalize U_{BE}^{i}. Therefore, SN calculates the average BE utilization in the hot-spot denoted $\overline{U_{BE}}$. Since a perfect load balancing is not feasible SN-i has to transfer STA only if $U_{BE}^{i} > 1.05\overline{U_{BE}}$ (we assume a 5 % degree of tolerance because perfect load balancing is not feasible as explained before). If so, SN-i makes use of both **Test 1** and **Test 2** similarly to described previously (in that order), but the new objective in this case is to obtain:

$$U_{BE}^{i'} \leq \overline{U_{BE}} \tag{4.13}$$

4. If $U_{BE}^{i} < U_{BEmax}$ and SN-i has decided not to transfer any STAs until now, (this means that SN-i was underloaded before running the algorithm) then SN-i can evaluate changing to IDLE mode. This step assesses the effect of transferring *ALL* of its CO and BE load to neighboring SNs. To do this it has to use **Test 2**, but the objective in this case is to transfer all the STAs. However, there are differences in the capacity that neighboring SNs can support. Instead of pushing the neighbors to U_{BEmax}, it is better to provide some hysteresis window, therefore in this case $C(k)' = C(k)H_{factor}$, where H_{factor} is the hysteresis factor defined in Section 3.3.1. The idea is to provide a gap below the U_{BE} of the neighboring SNs thus preventing the SN-i from changing to IDLE mode and shortly after that waking up another SN needlessly.

5. Regardless of the value of U_{BE}^{i} (above or below threshold) every time when the SN-i has the DT, it compares its battery-capacity level with

the battery-capacity level of all the SNs. If it has the lowest battery-capacity level (E_{lowest_ACTIVE}) among the ACTIVE SNs, a level difference with the IDLE SN with the highest battery-capacity level ($E_{highest\_IDLE}$) and its battery-capacity level is greater than $E_{threshold}$ (defined in Section 3.3.1) then these SNs have to switch modes. Consequently, SN-i commands this IDLE SN (denoted SN-c) to change to ACTIVE mode, then it has to transfer all of its STAs to this new SN (STA-SN ***tuples*** are created). Additionally, when more than one SN has been in ACTIVE mode for a long period of time and there is no IDLE SN to wake up, the situation could arise where a certain SN-i is handling more load than other ACTIVE SN and therefore depleting its battery faster. In this case if SN-i has the shortest expected lifetime (draining battery energy) then it should try to reduce its load by forcing the neighboring SNs to work close to U_{BEmax}. SN-i should use the procedures mentioned before to decrease its power consumption. This aims to extend the lifetime of the SNs by having not so different battery-capacity levels.

6. At this point of the algorithm, if some STA-SN ***tuples*** have been created, then the SN-i disassociates STAs, as described in Section 3.3.2, according to the ***tuples*** STA-SN previously created.

7. Finally, SN-i broadcast the information as defined in 3.3 and additionally sends the STA-SN ***tuples*** (unlike the PLAS algorithm). The objective is that if a roaming SN tries to re-associate with the incorrect SN, then its request should be rejected. The STA would keep trying to associate

until it finds the correct SN (the ***tuple*** STA-SN is matched).

The pseudocode for the LAS algorithm is shown in Figures 4.4 and 4.5. The clear advantage of this algorithm is that aims to balance the U_{BE} equally for all the SNs only when it cannot be reduced below U_{BEmax}. Moreover, STAs are not transferred if U_{BE} is lower than U_{BEmax}, thus reducing the number of handovers. Performance of this algorithm is shown in Chapter 6.

```
1  SN-i executes this algorithm
2  Create list of BE STAs: t^k_BE → s(k)
3  Create list of all STAs: t^k_BE + t^k_CO U_BEmax → v(k)
4  if T_W timer expired & DT = i
5  begin
6    Calculate utilization: U_BE, U_CO
7    if U^i_BE > U_BEmax
8    begin
9      Compute: α = [U^i_BE − U_BEmax]T^i_BE
10     Create: [U_BEmax − U^j_BE]T^j_BE → C(j), ∀j_ACTIVE ≠ i
11     Compute: γ = ∑C(j)
12     if α < γ, %TEST 1
13     begin
14       BinPacking(C(j),s(k),α) → Tuples(k,j)
15       if Tuples(k,j) ≠ Null
16         α = 0
17     end
18     if α > 0 & α < γ, %TEST 2
19     begin
20       BinPacking(C(j),v(k),α) → Tuples(k,j)
21       if Tuples(k,j) ≠ Null
22         α = 0
23     end
24     if α > 0 & there is an IDLE SN
25     begin%Splitting the BE load
26       Command SN-h with E_highest_IDLE to awake
27       C(h) = C(i) = T^i_BE
28       α = 0
29       BinPacking(C(i) : C(h),v(k),U^i_BE T^i_BE)
30     end
31     if α > 0
```

Figure 4.4: LAS Algorithm

32 **begin**%*over* U_{BEmax} *operation*
33 Calculate $\overline{U_{BE}}$
34 **if** $U^i_{BE} > 1.05\overline{U_{BE}}$ %*SN overloaded*
35 **begin**
36 $\alpha = \overline{U_{BE}} - U^i_{BE}$
37 Create: $[\overline{U_{BE}} - U^j_{BE}]T^j_{BE} \rightarrow \boldsymbol{C(j)}$, $\forall j_{ACTIVE} \neq i$
38 BinPacking($\boldsymbol{C(j)}$,$\boldsymbol{s(k)}$,α) $\rightarrow$ $\boldsymbol{Tuples(k,j)}$
39 **if** $\boldsymbol{Tuples(k,j)}$= *Null*
40 BinPacking($\boldsymbol{C(j)}$,$\boldsymbol{v(k)}$,α) $\rightarrow$ $\boldsymbol{Tuples(k,j)}$
41 **end**
42 **end**
43 **end**
44 **else if** SN-i is E_{lowest_ACTIVE}
45 **begin**%*Sleeping Case*
46 $\alpha = U^i_{BE}T^i_{BE} + U^i_{CO}T^i_{CO}U_{BEmax}$
47 Create: $[U_{BEmax}H_{factor} - U^j_{BE}]T^j_{BE} \rightarrow \boldsymbol{C(j)}$, $\forall j_{ACTIVE} \neq i$
48 BinPacking($\boldsymbol{C(j)}$,$\boldsymbol{v(k)}$,α) $\rightarrow$ $\boldsymbol{Tuples(k,j)}$
49 **if** $\boldsymbol{Tuples(k,j)} \neq$ *Null*
50 When full STA transfer complete SN-i changes to IDLE mode
51 **end**
52 **end**
53 **if** T_{switch} timer expired & SN is $E_{highest\_IDLE}$ & $DT = i$
54 **begin**
55 **if** $[E_{highest\_IDLE} - E_{lowest\_ACTIVE}] > E_{threshold}$
56 Command SN-c with $E_{highest\_IDLE}$ to awake
57 Transfer all STAs to SN-c $\rightarrow$ $\boldsymbol{Tuples(k,j)}$
58 When full STA transfer complete SN-i changes to IDLE mode
59 **end**
60 **if** $\boldsymbol{Tuples(k,j)} \neq$ *Null*
61 Disassociate STA-k | $\in$ $\boldsymbol{Tuples(k,j)}$
62 Transmit SN-i information to neighbors including $\boldsymbol{Tuples(k,j)}$

Figure 4.5: LAS Algorithm (cont.)

Chapter 5

System Analysis

This Chapter presents some analytical models that help to understand the system behavior.

5.1 Maximum BE Utilization: $U_{BEthreshold}$

The correct selection of U_{BEmax} is essential for the proper operation of the PLAS and LAS algorithms. When the SNs operate below U_{BEmax}, the transmission delay is so low that it does not have major effects on real-time applications as explained in Section 4.2 and [28]. Nevertheless, when the channel utilization exceeds a certain value, denoted $U_{BEthreshold}$, the delay rises to unacceptable values for real-time applications. Moreover, part of the decision of activating and putting SNs to sleep relies also on the U_{BEmax} parameter. We now present some analysis to find an upper bound for $U_{BEthreshold}$ under different conditions. The author in [3] presented a detailed analysis of the satu-

ration throughput for DCF mode. In this section, we extend his work to obtain the channel utilization ($U_{BEthreshold}$) where the saturation throughput occurs. From [3], the probability that at least one STA transmits in a time slot is given by:

$$P_{tr} = 1 - (1 - \tau)^n \tag{5.1}$$

Where τ is the probability that a STA transmits in a randomly chosen slot time and n is the number of STAs contending for the channel. However, we are interested in the probability of a successful transmission P_s, which is conditioned by the fact that at least one STA transmits during the same time slot. This yields [3]:

$$P_s = \frac{n\tau(1-\tau)^{n-1}}{P_{tr}} = \frac{n\tau(1-\tau)^{n-1}}{1-(1-\tau)^n} \tag{5.2}$$

now we can express the throughput S as:

$$S = \frac{E[Payload\,information\,transmitted\,in\,a\,slot\,time]}{E[length\,of\,a\,slot\,time]} \tag{5.3}$$

The numerator of 5.3 is given by $P_{tr}P_sE[M]$, where *E[M]* is the mean time required to transmit a packet payload. The denominator is obtained considering that: there are no transmissions with probability $1 - Ptr$; a successful transmission occurs with probability $P_{tr}P_s$; and a collision occurs with probability $P_{tr}(1 - P_s)$. Replacing these terms in the equation 5.3 is obtained [3]:

$$S = \frac{P_sP_{tr}E[M]}{\sigma(1-P_{tr}) + P_{tr}P_sT_s + P_{tr}(1-P_s)T_c} \tag{5.4}$$

where T_s is the mean transmission time of a successful packet including the related overhead (MAC and PHY) and T_c is the channel occupancy time due to a collision. Both variables take different values depending on the channel

access procedure (Basic Access or RTS/CTS) used. Similarly we define the channel utilization U as:

$$U = \frac{E[time\ the\ channel\ has\ been\ sensed\ busy\ during\ a\ slot\ time]}{E[length\ of\ a\ slot\ time]} \tag{5.5}$$

In fact, the expression for U is very similar to the one for S and the only difference is that U takes into account the usage of the channel when packets collide (in reality it also includes effects of interference and hidden STAs) and the overhead due to transmission of management messages, MAC layer and Physical layer overheads.

Also, we define U_s and U_c as the time the channel has been busy due to a successful and collided transmission of a packet respectively. Similarly, both variables depend on the channel access procedure. Hence, we can express:

$$U = \frac{P_{tr}P_sU_s + P_{tr}(1-P_s)U_c}{\sigma(1-P_{tr}) + P_{tr}P_sT_s + P_{tr}(1-P_s)T_c} \tag{5.6}$$

Using the information from Section 2.1.2 we obtain for the Basic Access procedure:

$$\begin{aligned} T_{BAS\_s} &= T_{PHY} + T_{MAC} + E[M] + SIFS + 2\delta + T_{ACK} + DIFS, \\ T_{BAS\_c} &= T_{PHY} + T_{MAC} + E[M] + DIFS + \delta, \\ U_{BAS\_s} &= E[M] + T_{MAC} + T_{PHY} + T_{ACK}, \\ U_{BAS\_c} &= E[M] + T_{MAC} + T_{PHY} \end{aligned} \tag{5.7}$$

likewise for the RTS/CTS procedure we have:

$$\begin{aligned} T_{RTS\_s} &= T_{PHY} + T_{MAC} + E[M] + T_{RTS} + T_{CTS} + T_{ACK} + \\ &\quad 4\delta + 3SIFS + DIFS, \\ T_{RTS\_c} &= T_{RTS} + DIFS + \delta, \\ U_{RTS\_s} &= E[M] + T_{MAC} + T_{PHY} + T_{RTS} + T_{CTS} + T_{ACK}, \\ U_{RTS\_c} &= T_{RTS} \end{aligned} \tag{5.8}$$

Where T_{PHY} is the overhead time due to physical layer when transmitting a data frame; T_{MAC} is the overhead time due to the MAC header; T_{ACK} is the transmission time required to acknowledge the transmission including the physical layer overhead; T_{RTS} and T_{CTS} are the times needed to transmit an RTS and CTS frame respectively including the physical overhead; δ is the propagation delay between the SN and the STA; finally, SIFS and DIFS are the interframe spaces.

In [3], the author also obtained an approximate expression for the theoretical saturation throughput S_{max}, which was found to be practically independent of the number of STAs in the channel. It was also found that after reaching the saturation point, a further increase in the load only decreases the throughput. Consequently, U_{max} is obtained when $S = S_{max}$ because at this point the best performance is achieved. Equations 5.9 provide the expressions to calculate

both S_{max} and U_{max}:

$$\begin{aligned} F &= \sqrt{\frac{T_c}{2\sigma}}, \\ S_{max} &= \frac{E[M]}{T_s + \sigma F + T_c(F(e^{1/F} - 1) - 1)}, \\ U_{max} &= \frac{U_s + U_c F((e^{1/F} - 1) - 1)}{T_s + \sigma F + T_c(F(e^{1/F} - 1) - 1)} \end{aligned} \tag{5.9}$$

where σ is the slot time duration which dependens on the physical layer used.

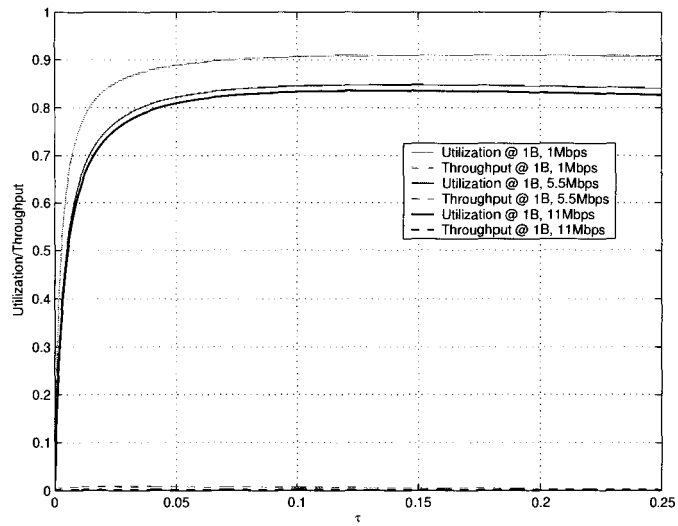


Figure 5.1: Saturation/Utilization Analysis for an average payload of 1 Byte using the basic access procedure for different mean Tx rates

Results for Equations 5.4 and 5.6 are shown in Figures 5.1, 5.2 and 5.3 when varying τ. The maximum values obtained from Equations 5.9 are summarized in Table 5.1. Clearly, the worst case performance is obtained when STAs transmit on average small packets of almost negligible payload at high rates. (11 Mbps & 1 Byte) where U_{max} is approximately 0.79. However, voice packets, while being small are much larger than 1 byte. Moreover, these small payloads

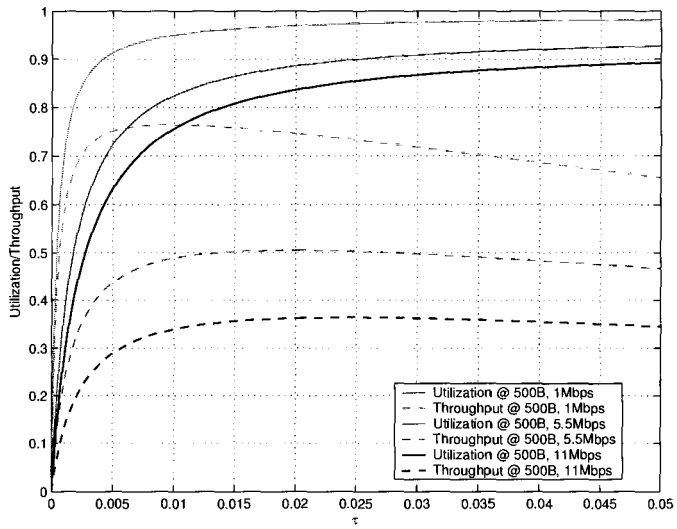


Figure 5.2: Saturation/Utilization Analysis for an average payload of 500 Bytes using the basic access procedure for different mean Tx rates

(e.g. VoIP) would typically be transmitted on the CO subframe if implemented in the SNs. Larger packets are more likely found in the BE subframe. For this reason we analyzed two more cases, one with 500 bytes of average payload using the basic access procedure and another one with 1500 bytes of average payload and using the RTS/CTS procedure. In these two cases, U_{max} takes values greater than 0.85.

Our objective was to determine $U_{BEthreshold}$ which we determined to be equal to U_{max}. Once $U_{BEthreshold}$ is obtained we are able to determine an upper bound for U_{BEmax}, thus we can affirm that setting $U_{BEmax} \leq U_{BEthreshold} = U_{max}$ is enough to guarantee good performance in the BE Subframe. The lower U_{BEmax} is, the faster the hot-spot reaction is to changes in the load. Setting $U_{BEmax} = 0.79$ guarantees protection under any kind of traffic characteristics

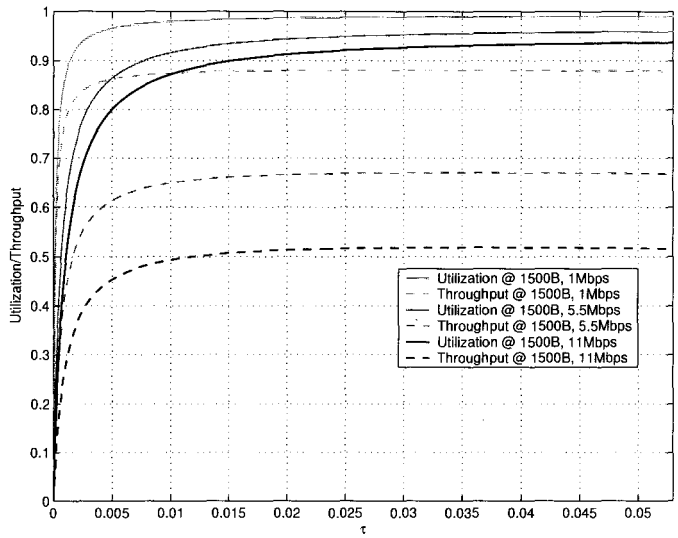


Figure 5.3: Saturation/Utilization Analysis for an average payload of 500 Bytes using RTS/CTS procedure for different mean Tx rates

(i.e. small payloads) but a greedy scheme might set $U_{BEmax} = 0.85$, considering a more realistic average payload size of 500 bytes. Performance evaluation of the system for different values of U_{BEmax} is shown in Chapter 6.

5.2 Power Consumption Model for the BE Subframe

In order to confirm the validity of our results, we introduce here a simple model for the power consumption on the BE subframe. Since the CO subframe behavior is almost deterministic (because it is controlled by the HC) the main problem resides in the BE subframe. Let us assume that the mean data frame size is $E[G]$ and the mean time required to transmit that data frame is $E[M]$.

	1-Byte payload, Basic Access		500-Byte payload, Basic Access		1500-Byte payload, RTS/CTS	
Mean Rate Mbps	U_{max}	S_{max}	U_{max}	S_{max}	U_{max}	S_{max}
1	0.8571	0.0081	0.9475	0.7603	0.9886	0.8797
5.5	0.8005	0.0023	0.8845	0.5011	0.9553	0.6686
11	**0.79**	0.0012	0.8537	0.3601	0.9314	**0.517**

Table 5.1: S_{max} and U_{max} for the BE Subframe

When the SNs are active and using the LAS or PLAS algorithm, they measure the average channel utilization U_{BE} over a T_W time period and then they decide based on this value if one of them should switch to ACTIVE or IDLE mode. Consequently, both the traffic load and U_{BE} are the main variables that affect the power consumption.

We define R as the number of packets arrived during the T_W period of time, then we can express the channel utilization as:

$$U = \frac{T_{Tx/Rx} R}{T_W} \tag{5.10}$$

where $T_{Tx/Rx}$ is the mean time that the channel is busy due to the transmission or reception of a single data frame and is given by the following equation:

$$T_{Tx/Rx} = E[M] + T_{PHY} + T_{MAC} + T_{RTS} + T_{CTS} + T_{ACK} \tag{5.11}$$

The above model does not consider channel utilization due to collisions and hidden STAs.

We can map these variables into the discrete time domain using T_W as a sampling interval. Additionally, to simplify the analysis we can assume that the traffic is uplink/downlink symmetric then we can express:

$$\begin{aligned} \lambda_{Tx}(k) &= \lambda_{Rx}(k) = \frac{\lambda(k)}{2}, \\ \lambda(k) &= \frac{R(k)}{T_W}, \\ U(k) &= \frac{R(k)T_{Tx/Rx}}{T_W}, \end{aligned} \tag{5.12}$$

where $\lambda(k)$ is the packet arrival rate. Using this Equations 5.12 and Equation 5.11 we can find an expression for the power consumption during the interval $[k-1, k]$:

$$\begin{aligned} P_{total}(k) &= \frac{\lambda}{2}(E[M] + T_{RTS} + T_{CTS} + T_{ACK})(P_{Tx} + P_{Rx}) \\ &\quad + P_{listening}, \\ P_{listening}(k) &= (\mu(k) - \lambda(k)(E[M] + T_{RTS} + T_{CTS} + \\ &\quad T_{ACK}))P_{Rx} \end{aligned} \tag{5.13}$$

where P_{Tx} and P_{Rx} are the power consumption values of the NICs of the SNs when transmitting and receiving respectively. $P_{total}(k)$ represents the total power consumption in the hot-spot and includes $P_{listening}(k)$ which is the power consumption associated with having the radio interface on but neither receiving nor transmitting frames. The value of $P_{listening}(k)$ depends on the number of ACTIVE SNs in the interval $[k-1, k]$ denoted by $\mu(k)$, which will vary according to the measured U_{BE}. This can be expressed as:

$$\mu(k) = \begin{cases} min\left(N_{SN}, \left\lceil \frac{U(k)}{U_{BEmax}} \right\rceil\right) & \mu(k) \geq \mu(k-1) \\ max\left(1, \left\lceil \frac{U(k)}{U_{BEmax} H_{factor}} \right\rceil\right) & \mu(k) < \mu(k-1) \end{cases} \tag{5.14}$$

Depending on the value of $U(k)$ in the interval $[k-1, k]$, the number of ACTIVE SNs is updated using the same criteria described in the algorithms presented in Chapter 4. The average power consumption of each SN can be simply obtained by dividing $P_{total}(k)$ by the number of SN, N_{SN}.

A comparison of this analytical model to results obtained through simulations is presented in Chapter 6.

5.3 Traffic Model for the BE Subframe

5.3.1 Self-Similar Traffic Model

The BE subframe is expected to be the main transport mechanism for transmitting frames for data applications. Traffic studies carried out in [23] demonstrated that traffic produced by data applications such as TELNET, FTP, WWW, etc. cannot be modeled used the classic Poisson Arrival Process mainly because it lacks the properties of these traffic classes. These applications typically exhibit self-similar characteristics (i.e. burstiness). This can be efficiently modeled using heavy-tailed distributions[1]. One distribution that has gained a lot of attention among other heavy-tailed distributions in the literature is the Pareto Distribution, this is due to its analytical tractability. According

[1] A R.V. Y is heavy-tailed if $Pr[Y \geq y] \sim t^{-\theta}$ as $t \to \infty$

to our literature survey, there are three main methods to produce self-similar traffic, in this thesis we will focus on the one described in [9]. The method generates self-similar traffic by generating packets with Pareto-distributed interarrival times between packets. The advantage of this method lies in its ease of implementation, thus the R.V. density can be expressed as:

$$f(t) = \frac{\psi \omega^{\psi}}{t^{\psi+1}} \tag{5.15}$$

with mean and variance given by:

$$\mu = \frac{\omega\psi}{\psi - 1},$$

$$\sigma^2 = \frac{\omega^2\psi}{(\psi-1)^2(\psi-2)} \tag{5.16}$$

Where the mean and variance are finite when $\psi > 2$, however, in this range the arrival process is not self-similar. On the contrary, self-similar properties are present for $\psi \leq 2$ where the variance is infinite, but the mean is infinite only for $\psi \leq 1$.

The degree of self-similarity in a process can be quantified using the Hurst parameter H which is bounded between $0.5 < H < 1$ for a Pareto process [31], where a larger value of H indicates more self-similarity. In [29] it was shown that the relation between H and ψ is given by:

$$H = \frac{3 - \psi}{2} \tag{5.17}$$

Therefore, once a value of H is selected it is easy to obtain ω just by specifying the mean interarrival time and substituting into Equation 5.16.

After all the parameters have been determined, one way to implement a packet sequence with Pareto distributed interarrival time is by applying inverse CDF methods [8] as

$$t = \frac{\omega}{x^{\frac{1}{\psi}}} \tag{5.18}$$

where x is an uniformly distributed R.V. with value in the range of $0 < x \leq 1$. Equation 5.18 is used in Chapter 6 to evaluate the system performance.

5.3.2 Bulk Traffic Model

A bulk traffic model can also be used to model traffic where large data transfers occur such as FTP file downloads or where WAN aggregation occurs. In this case, packets arrive in both the upstream and downstream directions following a compound Poisson distribution in which packets arrive in bursts. This burst arrival follows a Poisson arrival process and the burst size is geometrically distributed. The mean burst size is M packet, thus from the classic Poisson arrival model we can express the arrival rate as

$$\lambda = \frac{M\lambda' L}{C} \tag{5.19}$$

Where λ' is the mean burst arrival rate, C is the channel capacity and L is the mean packet size. Also, we assume symmetric traffic, thus $\lambda_{UL} = \lambda_{DL} = \lambda/2$. This model is also used in the simulation experiments in Chapter 6 to evaluate the system performance.

Chapter 6

Performance Results

In this chapter we present some experiments that help us to evaluate the performance of both the PLAS and LAS algorithms. Since the CO subframe behavior is mostly deterministic, we focus on the case of BE service using DCF for our experiments. We developed a detailed discrete event simulation software written in the C programming language and we used it to test the performance of the proposed solutions. There are two different experiments which are detailed in the next sections. We consider as mentioned in Chapter 3 only a single-hop performance of the system through the access NICs of the SNs. The delay and power consumption associated to the relaying/forwarding of packets through the ESS Mesh is not considered initially (we will consider later only the power consumption due to relaying packet to the ESS Mesh). We also considered the scenario where the packet interarrival time follows a Pareto process as explained in Section 5.3.1 because we are dealing with BE data traffic. This permits us to include bursty characteristics in the packet arrival process. The H parameter

we assume is 0.75 based on the results of several experiments conducted in [31]

6.1 LAS and PLAS Performance under Fixed Load

6.1.1 Fixed Load Experiment Description

We assume a set of 40 STAs distributed randomly in a 200m x 200m area. We consider 3 identical SNs providing coverage to such a service area. In each simulation step, the system supports fixed load (i.e the mean packet arrival rate is constant) for an interval of 300 seconds.

We also test both algorithms for different values of U_{BEmax}: 0.65, 0.75, and 0.85. As known from Section 5.1, 0.75 guarantees protection against small payload frame transmissions, but 0.65 makes the system even more reactive under load changes (peaks). On the contrary, 0.85 is a greedy setting for the system and is supposed to provide the lowest power consumption at the expense of the performance.

The measuring interval T_W is set to 2 seconds for this experiment. However, the PLAS algorithm balances the load (transferring stations) every 100 ms as recommended for the original algorithm ([10]). Table 6.1 shows some parameters used for this experiment.

Additionally, we also evaluate the system performance using the Bulk Traffic Model described in Section 5.3.2. In this way we compare the system performance under two different traffic models. The mean burst size M is set to 10,

25 and 50 for this experiment.

Parameter	**Value**
SOLAR NODES	3
STATIONS	40
PACKET SIZE	1500 Bytes
COVERAGE AREA	200 x 200 meters
TRANSMIT POWER	750 mW
RECEIVE POWER	500 mW
LISTENING POWER	500 mW
DOZE POWER	2 mW

Table 6.1: Simulation Parameters

6.1.2 Fixed Load Results

Fixed Load Experiment Results: Self-Similar Traffic Model

In Figure 6.1, the transmission delay versus normalized load is shown for different values of U_{BEmax}. The performance of both algorithms is advantageous in terms of delay (less than 50 ms). We can observe some peaks for both algorithms when the load is close to their respective U_{BEmax} setting. Every case presents 3 peaks, each one of them representing the state when the system operates at $\overline{U_{BE}} = nU_{BEmax}$ with $n = 1, 2, 3$. An explanation for this is that when the traffic load decreases both algorithms switch SNs to IDLE mode, but then the load suddenly increases (a burst) without having enough ACTIVE SNs to handle the load. Thus, the overall system delay increases too. This effect could be worse if the hysteresis provided by H_{factor} was not taken into consideration.

All the load balancing cases finally converge when the load is around 1.60, which is understandable since the theoretical capacity of the DCF channel is reached when the load $S_{max} = 0.517$ as shown in Table 5.1 in Chapter 5, and in this deployment there are only 3 SNs (i.e. $3 \times 0.517 \sim 1.6$). The case where no load balancing algorithm is used has the worst performance because the distribution of STAs among the SNs is uneven saturating the SNs with more STAs sooner, thus increasing the transmission delay. However, at lower load this same case shows the lowest transmission delay, mainly for two reasons: the first is because the channel of each SN is not highly utilized and also there are no enqueued/forwarded packets due to transferring STAs (handovers).

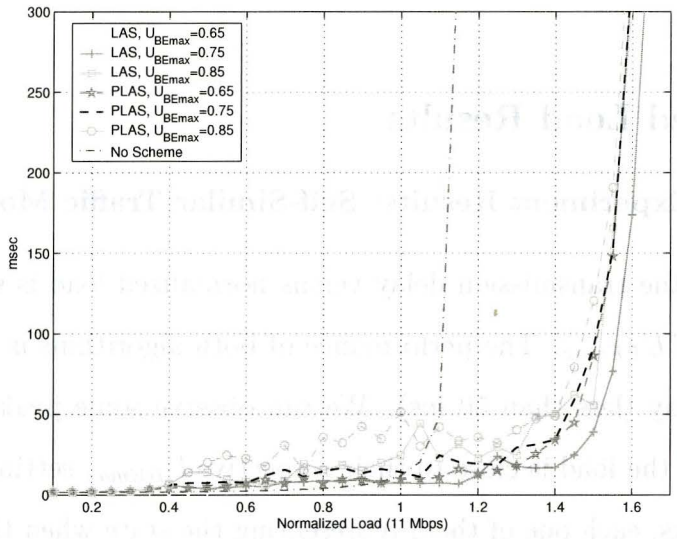


Figure 6.1: Mean Transmission Delay per packet vs. Normalized Load for Self-Similar Traffic

Another parameter measured was the mean number of collisions per packet. Figure 6.2 shows the results obtained. We can see that the LAS algorithm

slightly outperformed the PLAS algorithm in all the cases. We can also notice a significant improvement compared to the case where no load balancing/distribution algorithm is used at high load. However, similar to the transmission delay performance, when the load is low both algorithms increase the number of collisions.

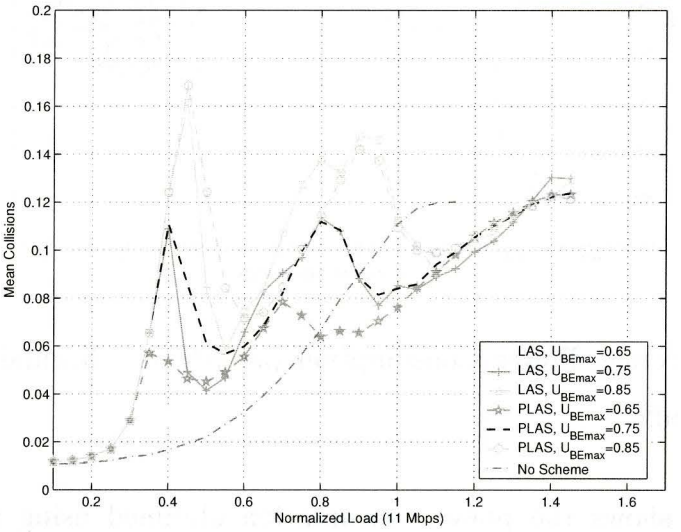


Figure 6.2: Mean Collisions per packet vs. Normalized Load for Self-Similar Traffic

Figure 6.3 shows the average power consumption per SN. We notice that the PLAS algorithm has a slightly lower power consumption in all the cases. We see that the LAS algorithm always wakes up SNs a little bit earlier than its PLAS counterpart and this is because the PLAS algorithm has more flexibility to balance the load and meet $\overline{U_{BE}} < U_{BEmax}$ range thus not requiring a SN to wakeup. We can also appreciate the power consumption improvements using load balancing compared to having the 3 SNs always ACTIVE (without using

load balancing).

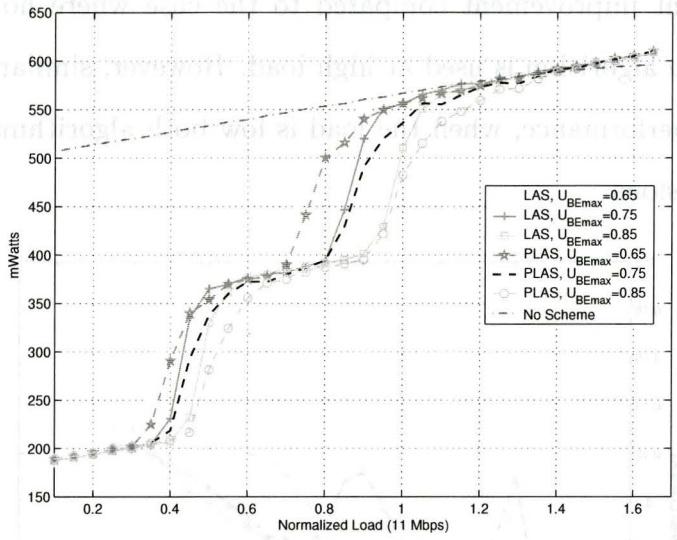


Figure 6.3: Average Power Consumption per SN vs. Normalized Load for Self-Similar Traffic

Figure 6.4 shows the power consumption obtained using the analytical power model developed in Section 5.2. We can see how the results are quite similar to those obtained through simulations (Figure 6.3). This result is very important because it confirms the validity of our simulation results. Some differences can be noticed between both plots because the analytical model does not include effects due to collisions.

Finally, in Figure 6.5 the handover rate is shown. This parameter is very important not only because some NICs and/or applications are not able to support re-association (3.3.2), but also because delay sensitive (e.g. voice or video) applications might experience packet losses or call terminations. By analyzing the graph, it is obvious that the PLAS algorithm has a very high

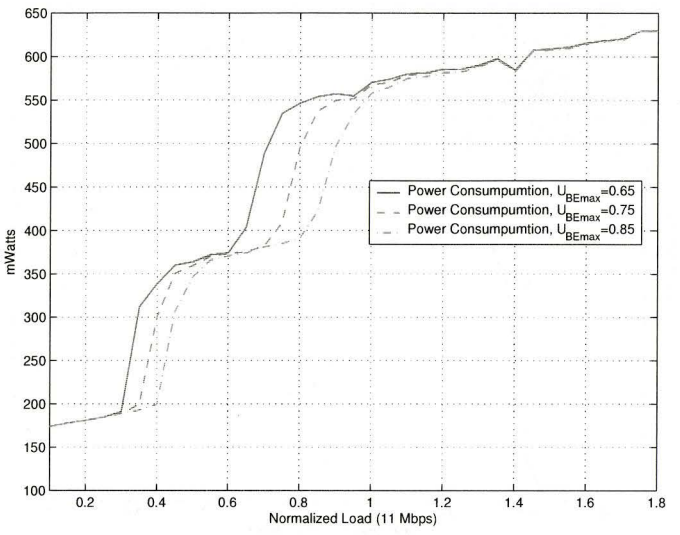


Figure 6.4: Analytical Average Power Consumption per SN vs. Normalized Load

cost in terms of handovers compared to the LAS algorithm.

Fixed Load Experiment Results: Bulk Traffic Model

We can see in Figure 6.6 that the delay performance is quite similar in shape to the case where Pareto distributed inter-arrival time was used (Figure 6.1). However, the overall delay is higher and unlike the self-similar case we can notice higher peaks when the normalized load is around 0.5 which corresponds to the second SN activation.

In Figure 6.7 we can see the high number of collisions per packet. Contrary to the Pareto arrivals case (Figure 6.2), in which the collisions increase with the load, in this case the collisions decrease. This could be interpreted as the repetitive event where a burst arrives and the medium is idle (due to the low

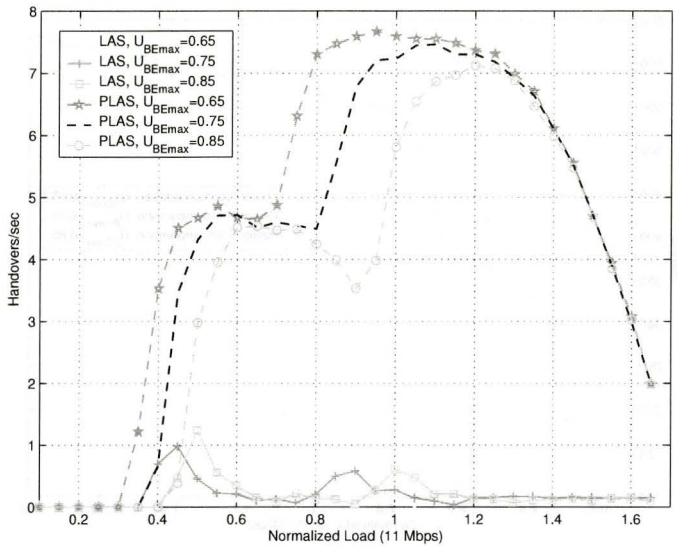


Figure 6.5: Handover rate vs. Normalized Load for Self-Similar Traffic

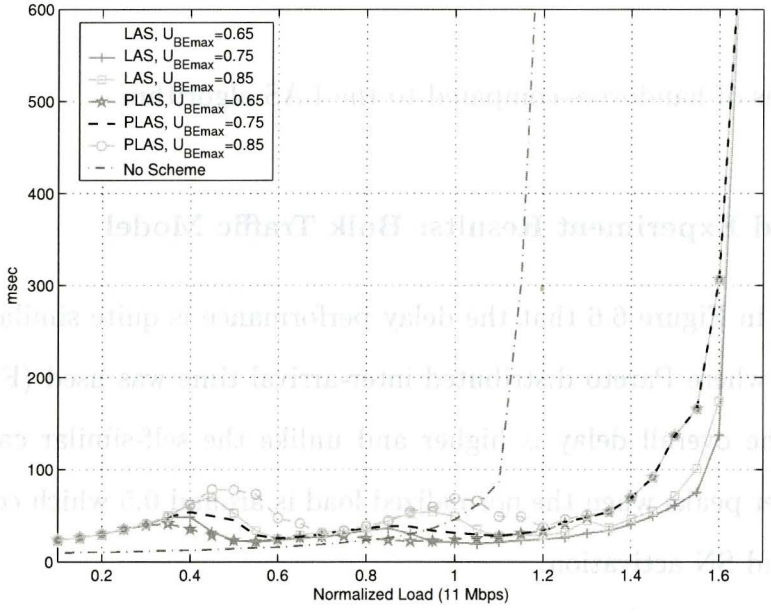


Figure 6.6: Mean Transmission Delay per packet vs. Normalized Load for Bulk Traffic $M = 10$

load), therefore all the STAs attempt to transmit packets at the same time. At lower rates, there is only one SN, therefore, all the STAs are contending on the same channel. When the load increases there are two reasons why the number of collision decreases: the first is that that there are more SNs thus decreasing the number of STAs contending on each channel (this only applies to cases that use load distribution algorithms); the second is that when the load increases the channel starts being busy more frequently, therefore when a burst arrives it is more probable that the medium is already busy due to packet transmissions (in this case the STAs defer access using the standard 802.11 CSMA/CA Backoff procedure explained in Section 2.1.2).

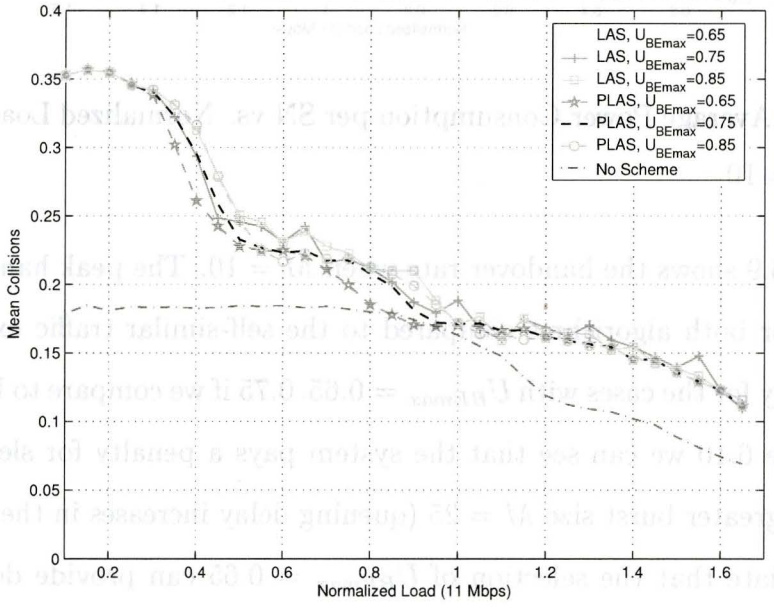


Figure 6.7: Mean Collisions per packet vs. Normalized Load for Bulk Traffic $M = 10$

Figure 6.8 shows the power consumption for this case. The results are very

similar in this case to the self-similar traffic case in Figure 6.3.

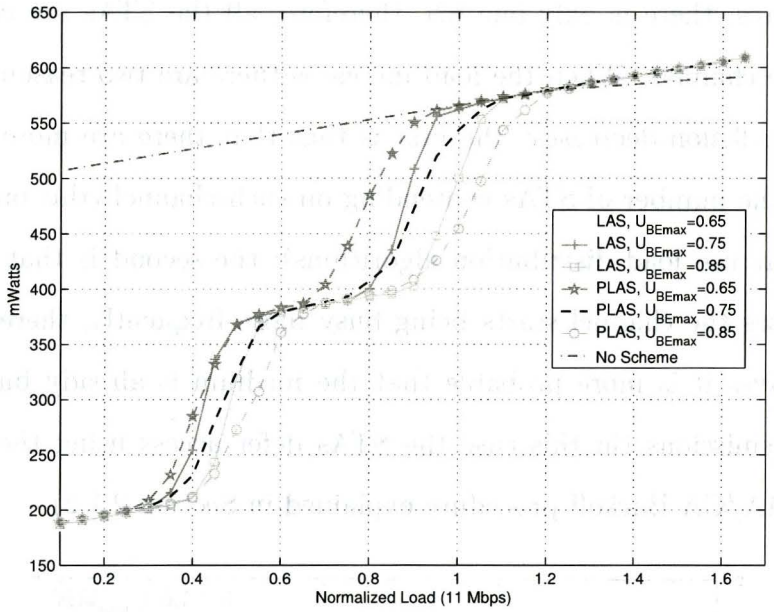


Figure 6.8: Average Power Consumption per SN vs. Normalized Load for Bulk Traffic $M = 10$

Figure 6.9 shows the handover rate when $M = 10$. The peak handover rate increases for both algorithms compared to the self-similar traffic experiment, but specially for the cases with $U_{BEmax} = 0.65, 0.75$ if we compare to Figure 6.5.

In figure 6.10 we can see that the system pays a penalty for sleeping SNs under this greater burst size $M = 25$ (queuing delay increases in the SNs). We can appreciate that the selection of $U_{BEmax} = 0.65$ can provide delay below 100ms, while the other values cannot.

Figure 6.11 shows the mean collisions per packet for $M = 25$. Compared to results with $M = 10$ (Figure 6.7) this variable is slightly lower. This could be explained as follows: when the upstream burst increases in size, successive

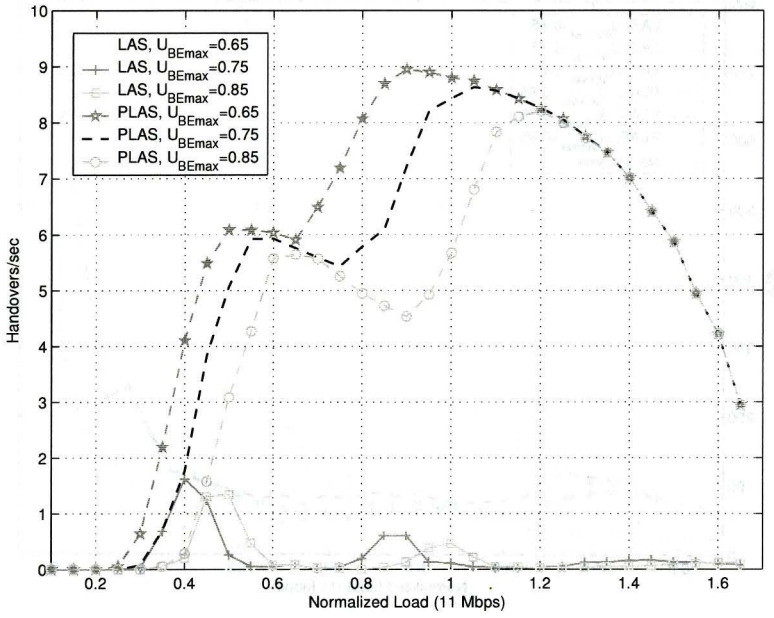


Figure 6.9: Handover rate vs. Normalized Load for Bulk Traffic $M = 10$

packets after the first one arriving in the burst are only enqueued and do not increase the collision count, which is the opposite of the previous case where the STAs sensed more often that the medium was idle.

Figure 6.12 shows the average average power consumption for this case. We notice how the shape of the curves changes for the three levels (200mW, 400mW 550mW). The tangent lines to this curves in the "transition regions" have lower slope than in the previous cases because the larger burst size has increased the "transition range" between levels (i.e.each level is equivalent to the number of ACTIVE SNs).

Figure 6.13 shows the handover rate for this case. As the burst size increases in size, the handover rate does so for both algorithms comparing to $M = 10$ (Figure 6.9).

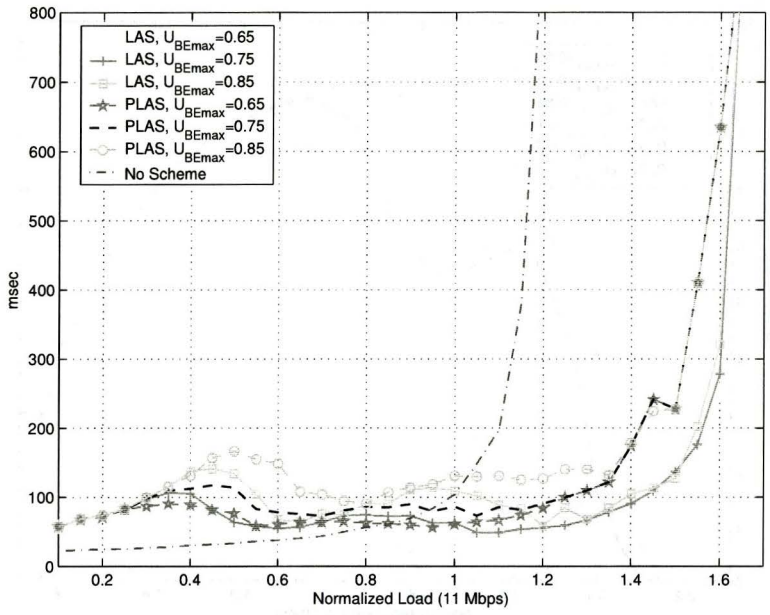


Figure 6.10: Mean Transmission Delay per packet vs. Normalized Load for Bulk Traffic $M = 25$

In figure 6.14 we can see that this burst size causes a deterioration of the delay performance of these algorithms. The case where LAS is employed for a value of $U_{BEmax} = 0.65$ provides the lowest delay among the load balancing/distribution schemes with a peak delay around 180ms before reaching maximum capacity.

Figure 6.15 shows the mean collions per packet for $M = 50$. As expected from the previous case (Figure 6.11) this parameter decreases in this case.

Figure 6.16 shows the average average power consumption for $M = 50$. We can notice how the shape of the curves approximate more to straight lines, this is caused by continuous changes of state in the SNs (IDLE, ACTIVE) which is a consequence of the big burst size.

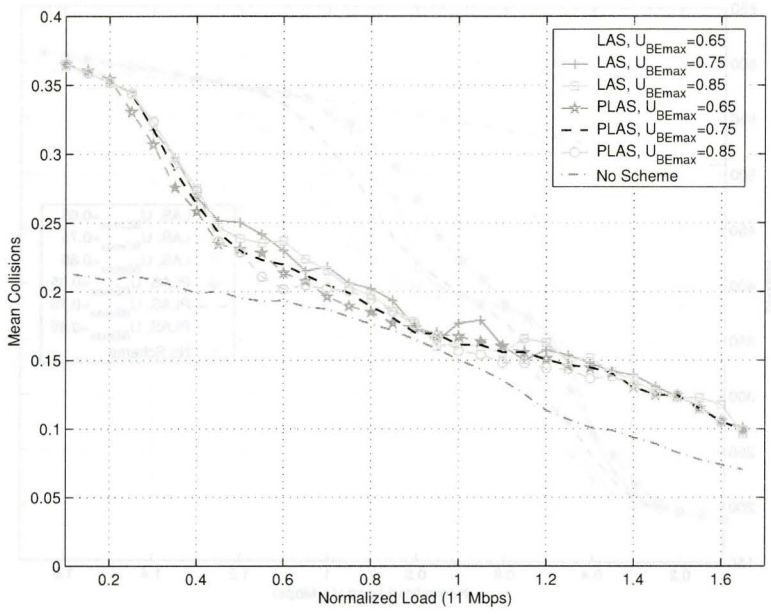


Figure 6.11: Mean Collisions per packet vs. Normalized Load for Bulk Traffic $M = 25$

Figure 6.17 shows the handover rate for $M = 50$. We find in this case the highest handover rate for the LAS algorithm, but surprisingly this variable decrease for the PLAS algorithm compared to the case $M = 25$ (Figure 6.13).

6.2 LAS and PLAS Evaluation under a Traffic Profile

6.2.1 Traffic Profile Experiment Description

In this experiment, we test the performance of both the LAS and PLAS algorithms for a given traffic profile. In [4], Kotz studied the Verizon Wi-Fi hot-spot

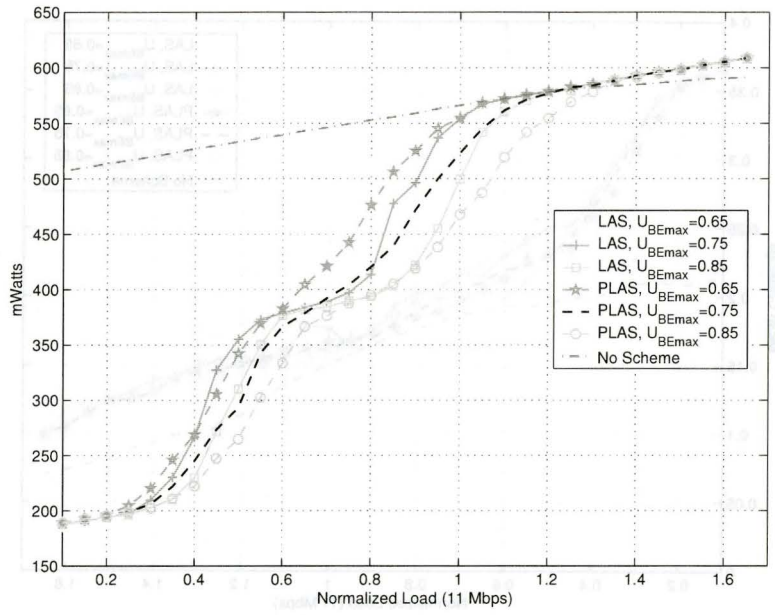


Figure 6.12: Average Power Consumption per SN vs. Normalized Load for Bulk Traffic $M = 25$

network in Manhattan, USA for over 5 weeks. Several traffic statistics were gathered in this study including the hourly traffic load. That deployment was intended for indoor/outdoor service, we used the data to create a traffic profile for our scenario as we expect some indoor users to make use of our hot-spot. The traffic profile for this experiment is shown in Figure 6.18. The hourly traffic load from [4] was scaled to obtain a profile with a highest peak load of approximately 13 Mbps (more than enough to justify the deployment of 3 SNs). In this experiment, U_{BEmax} is set to 0.65, 0.75 and 0.85 and the measuring/algorithm-execution interval T_W is set to 5 seconds for the LAS algorithm. For the PLAS algorithm the measuring interval for waking up and sleeping is also 20 seconds, but the execution interval for balancing the load is 100 ms as recommended

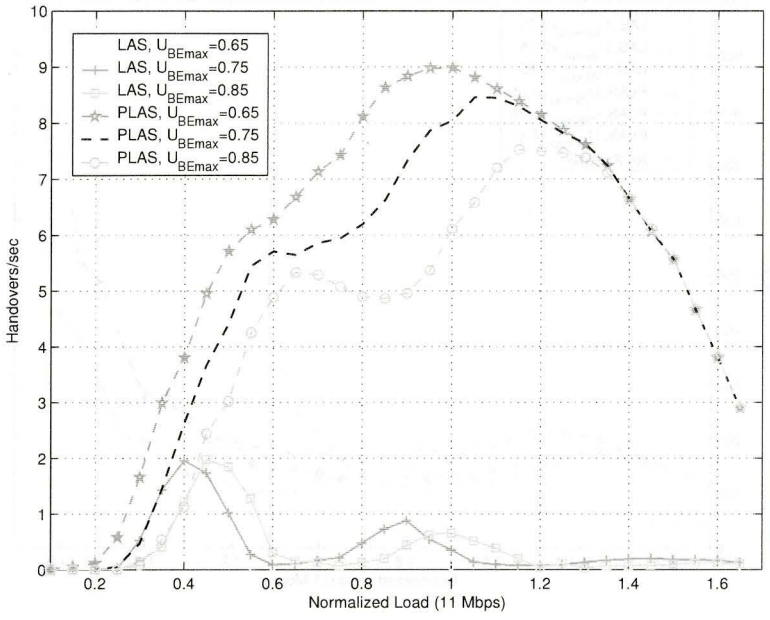


Figure 6.13: Handover rate vs. Normalized Load for Bulk Traffic $M = 25$

in [10]. In this experiment we consider that the packet arrival follows a Pareto process, as explained in Section 5.3.1, with $H = 0.75$. The rest of the parameters for this experiment are the same as those used in the previous experiment and are shown in Table 6.1.

6.2.2 Traffic Profile Experiment Results

Figures 6.19 and 6.20 show the load handled by each SN when $U_{BEmax} = 0.75$. We can appreciate that when more than one SN is ACTIVE the PLAS algorithm is always trying to equalize the load among the SNs, while in the LAS algorithm the SNs try to equalize the utilization only when $\overline{U_{BE}} > U_{BEmax}$. This behavior is predictable because it follows the procedures specified in Chap-

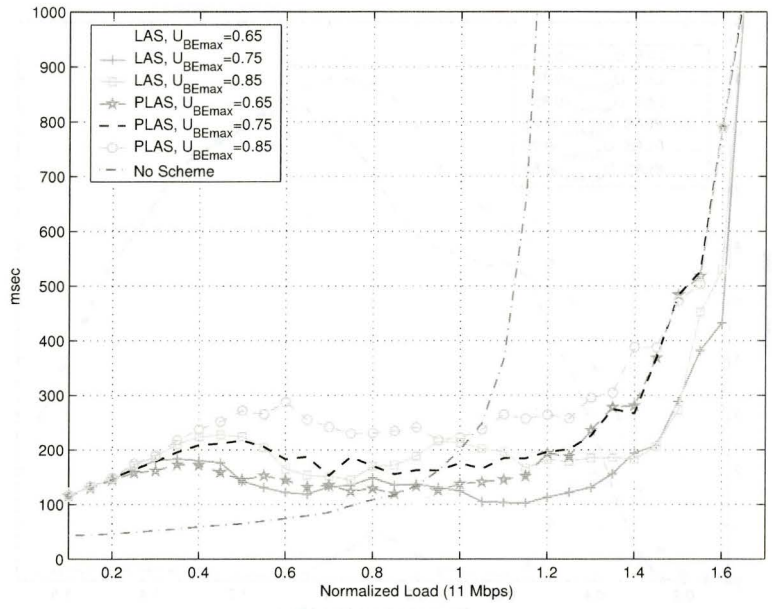


Figure 6.14: Mean Transmission Delay per packet vs. Normalized Load for Bulk Traffic $M = 50$

ter 4.

Figure 6.21 shows the average power consumption per SN in the hot-spot versus the simulation time for the LAS algorithm with $U_{BEmax} = 0.75$. We notice how the use of both algorithms brings down the power consumption when the traffic load decreases. Clearly, in terms of power consumption, the performance is similar for both solutions.

Even though both schemes decrease the delay at high load (close to hot-spot capacity) and save power by sleeping SNs, the proposed solutions suffer from a disadvantage: Figure 6.22 shows the average transmission delay per packet measured every minute. When the load is low, the PLAS and LAS have higher delay compared to the case when load balancing/distribution schemes are not

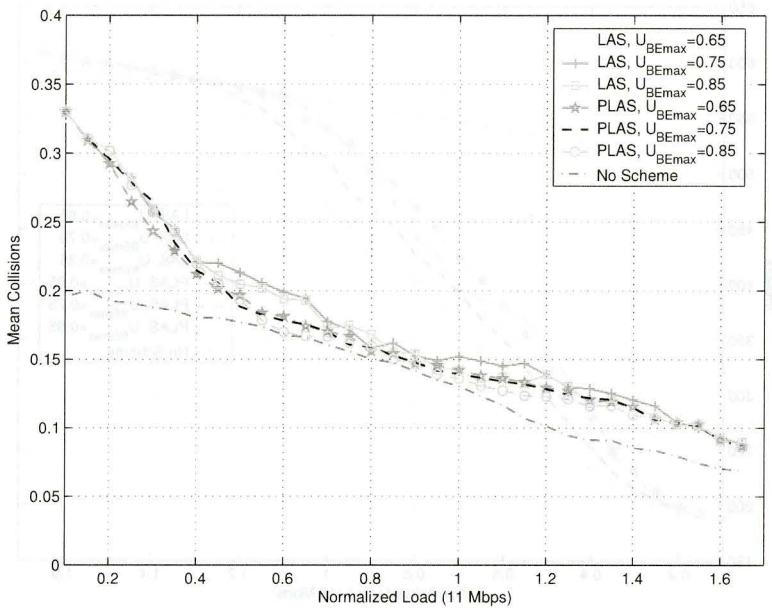


Figure 6.15: Mean Collisions per packet vs. Normalized Load for Bulk Traffic $M = 50$

used. The reason behind this is that when using these algorithms, we force the SNs to operate at higher channel utilization and when the load increases, the transmission delay increases too as shown previously in Figure 6.1. Moreover, transferring STAs from SN to SN also introduces more delay (packets are buffered and forwarded from SN to SN). Additionally, we notice several delay peaks which are mainly related to traffic bursts that increased the channel utilization before the system could react to them (the maximum time the system can take to awake a node is given by T_W for the PLAS and $3T_W$ for the LAS from Chapter 4). However, we can see that from 10 to 11 AM ($[36000, 39600]$ sec) the lack of a load balancing scheme affected the performance in the hotspot, while the PLAS and LAS schemes handled this situation of reducing the

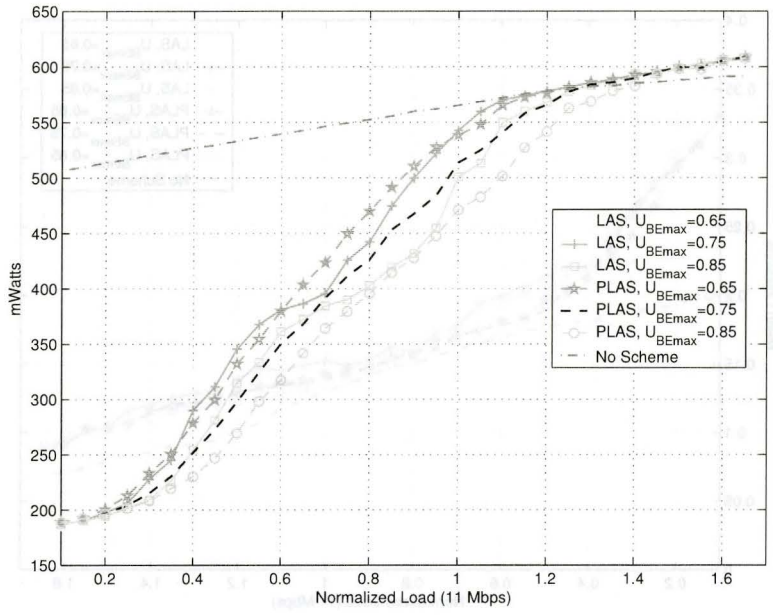


Figure 6.16: Average Power Consumption per SN vs. Normalized Load for Bulk Traffic $M = 50$

delay without major problems.

Table 6.2 shows a summary of the statistics obtained from this experiment for different values of U_{BEmax}. We can see that power saving achieved with the PLAS and LAS algorithms is around 35% when $U_{BEmax} = 0.85$. We also see as expected from the previous experiment that the PLAS algorithm has a high handover rate compared to the LAS. Therefore, the mean delay is also higher for the PLAS scheme.

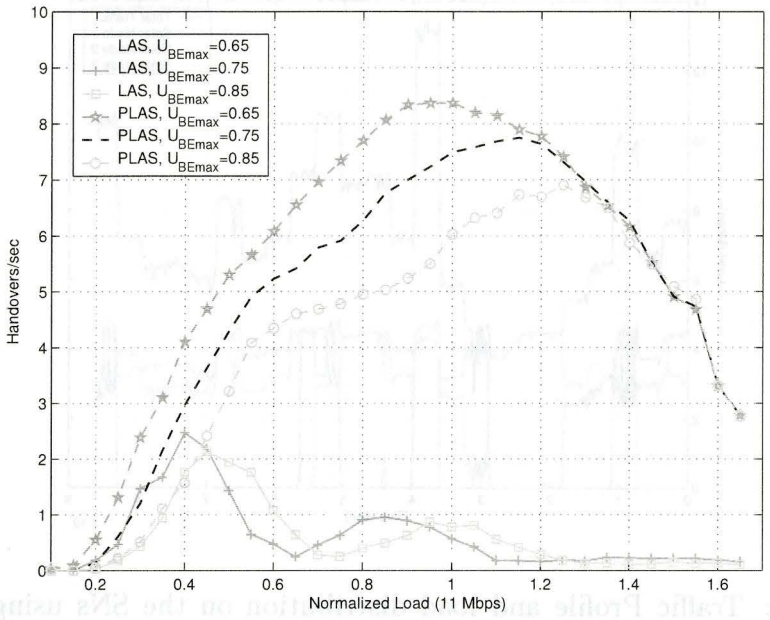


Figure 6.17: Handover rate vs. Normalized Load for Bulk Traffic $M = 50$

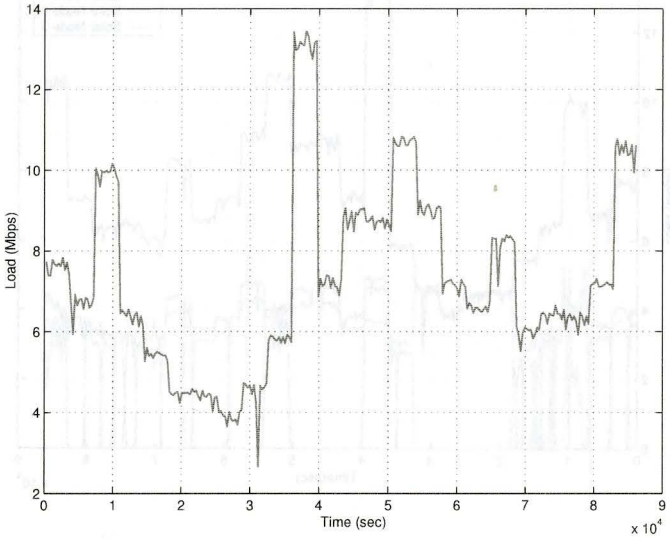


Figure 6.18: Traffic profile over a 24h window

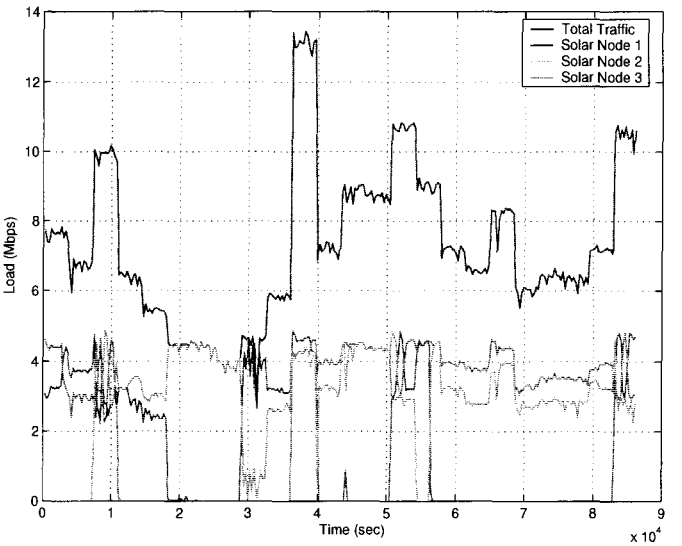


Figure 6.19: Traffic Profile and load distribution on the SNs using the LAS Algorithm

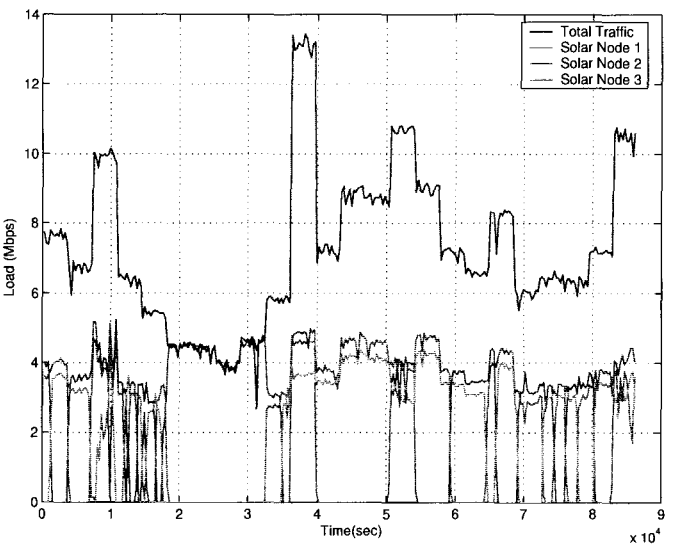


Figure 6.20: Traffic Profile and load distribution on the SNs using the PLAS Algorithm

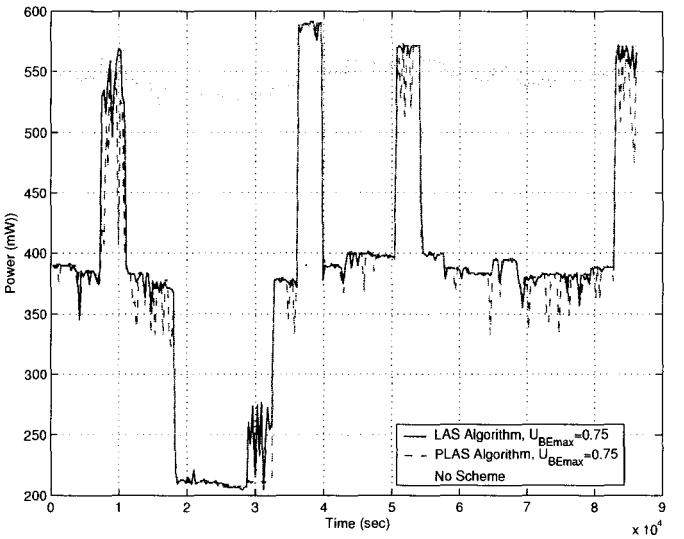


Figure 6.21: Power Consumption Comparison

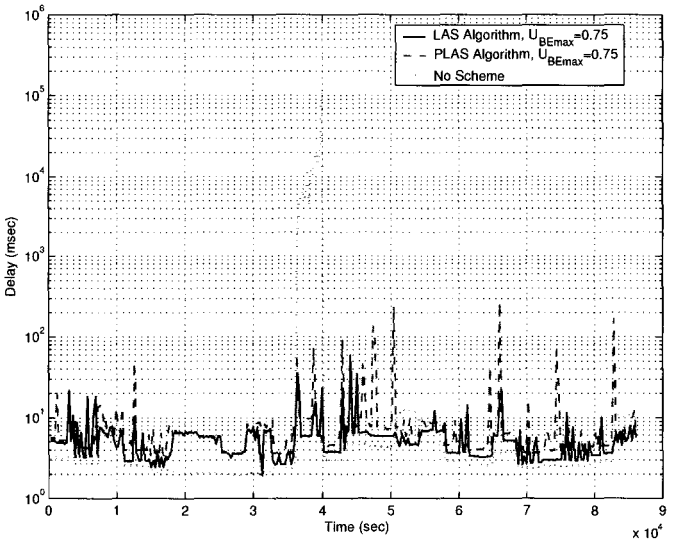


Figure 6.22: Delay Comparison measured over a 5 sec. period

6.2.3 Solar Node Design under a Traffic Profile

In this section, we aim to introduce the design and resource allocation procedure for the SNs, while assuming the traffic profile shown in Figure 6.18. This section

	Mean Power per SN (mW)	**Power Saving Ratio (%)**	**Handovers per sec**	**Mean Delay (ms)**	**Collisions per Packet**
PLAS, $U_{BEmax} = 0.65$	418.41	23.71	5.221	8.59	0.055
LAS, $U_{BEmax} = 0.65$	427.16	22.11	0.06	4.07	0.052
PLAS, $U_{BEmax} = 0.75$	380.69	30.59	4.195	12.05	0.072
LAS, $U_{BEmax} = 0.75$	387.62	29.32	0.051	6.33	0.075
PLAS, $U_{BEmax} = 0.85$	357.07	34.89	3.486	22.89	0.088
LAS, $U_{BEmax} = 0.85$	365.62	33.33	0.054	10.67	0.093
No scheme	548.43			1287.76	0.057

Table 6.2: Summary of Results for the Traffic Profile Experiment

is based entirely on the work in [6].

Until now we have considered the power consumption associated with transmitting and receiving packets using a single NIC, which is the access radio power consumption. We will now factor in the power consumption of the rest of the circuits (e.g. motherboard) in the SNs shown in Section 3.2. We will assume that the relay links are also implemented using IEEE 802.11 interfaces and that the average power consumption of the motherboard is about 400 mW in ACTIVE mode and 30mW in IDLE mode. Using the above information, we are able to obtain the total average power consumption per SN for each hour of the day, assuming the traffic profile in Figure 6.18. This is shown in Figure 6.23 (this case corresponds to $U_{BEmax} = 0.75$).

Previously in Section 3.2, we showed a simplified block diagram of a SN

Figure 6.23: Average Power Consumption per hour under a traffic profile ($U_{BEmax} = 0.75$)

(Figure 3.1). The solar panel and battery are connected to the SN through a charge controller which performs functions such as battery over-/under-charge protection. In the following energy flow model, we define $\mathcal{E}_{panel}(k)$ to be the energy produced in the solar panel over the time increment $[(k-1)\Delta, k\Delta]$, where Δ is the time-step length considered. In photovoltaic systems designed using publicly available meteorological data, data collection and modeling is done in discrete time, and more than sufficient accuracy is obtained using 1 hour Δ increments. The solar panel size is given by $\mathbf{S}_{panel}$, and is usually rated in watts at peak solar insolation. We also define $\mathcal{B}(k)$ to be the residual battery energy stored at time $k\Delta$, and Bc is defined to be the total battery capacity. If we assume that $\mathbf{L}(k)$ is the load energy demand over the time

duration $[(k-1)\Delta, k\Delta]$, then we can write that [26],

$$B(k) = \min\{\max[\mathcal{B}(k-1) + \mathcal{E}_{panel}(k) - \mathbf{L}(k), \mathbf{B}_{outage}], \mathbf{B}_{max}\} \quad (6.1)$$

Where $k = 0, 1, \ldots, K_{max}$ ranges over the entire set of solar irradiation samples taken for a given geographic location. Available data of this kind typically spans several decades of continuous measurements. In Equation 6.1, $\mathbf{B}_{outage}$ is the maximum allowed depth of discharge, based on safety and battery life considerations [18]. When $B(k) < \mathbf{B}_{outage}$, the charge controller will automatically disconnect the SN load and it will experience a radio outage. When the SN is designed for a zero outage target, outages of this kind should not be permitted.

Using publicly-available solar insolation data for the USA and Canada [20][19], and the energy flow model, a worst-case design can be performed. This is done by assuming that $\mathbf{L}(k) = \overline{P}_{SN}\Delta$ for all k, where $\overline{P}_{SN}$ is the maximum power consumption of a SN. Using Equation 6.1 and for a given traffic energy profile a discrete-time simulation of the SN can be done over the solar insolation history of the desired location. Using this method it is straightforward to produce a set of solar panel versus battery capacity contours of constant outage probability[1] (P_{outage}).

Using the available data and the software tool developed by the Wireless Networking Group at McMaster University [6] to design SNs, we were able to determine the required combinations of solar panel size and battery capacity to operate under our assumed traffic energy profile for a given outage probability. Using this software, for a given P_{outage}, combinations of battery/panel

[1] The outage probability is defined as the total SN outage duration for a given time period.

sizes can be obtained. The data generated is based on over 20 years of hourly meteorological data, and is available for over 100 locations in USA and Canada. Figure 6.24 represents the outage probability contour plots for different combinations of panel/battery sizes for the city of Toronto, Canada, and in Figure 6.25 for the city of Phoenix, USA.

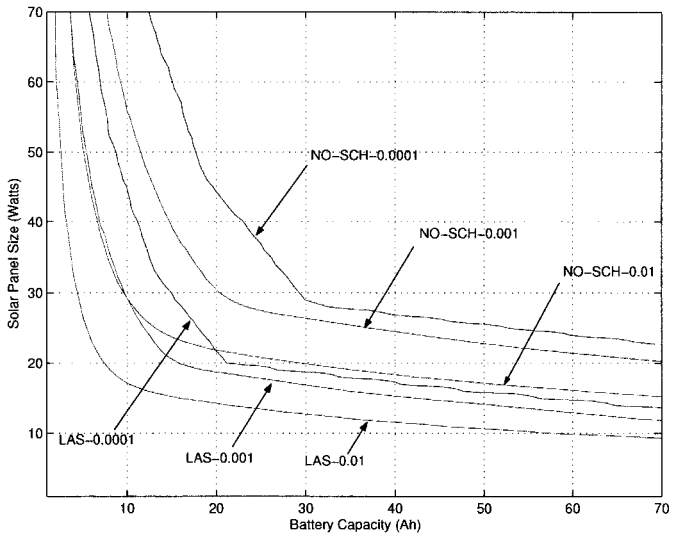


Figure 6.24: Solar Panel size vs. Battery Capacity combinations for the city of Toronto (CAN), for different P_{outage} values

Additionally, Figures 6.26 and Figure 6.27 show contours for $P_{outage} = 0.0001$ and different values of U_{BEmax}. The reductions in solar panel sizes and battery capacities are clear when we use the LAS algorithm as opposed to the case when load balancing is not implemented. As a result of this, the production cost of SNs can be reduced significantly. The cost optimal selection of battery capacities and solar panel sizes is obtained by super-imposing lines of constant cost and slope on the outage graphs. We make use of the linear

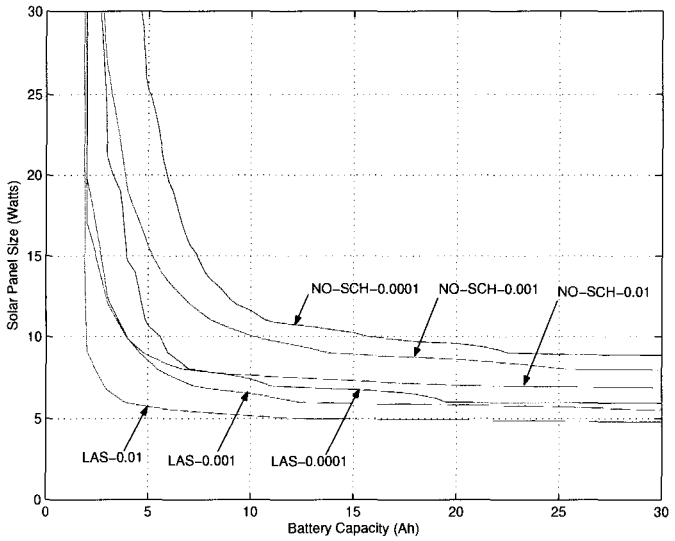


Figure 6.25: Solar Panel size vs. Battery Capacity combinations for the city of Phoenix (USA), for different P_{outage} values

relationship between cost and both panel and battery size (which is usually the case). Table 6.3 shows optimum values for both cities based on this cost function. The advantages provided by sleeping SNs is clear, especially for cities like Toronto where the solar radiation varies significantly from Winter to Summer seasons.

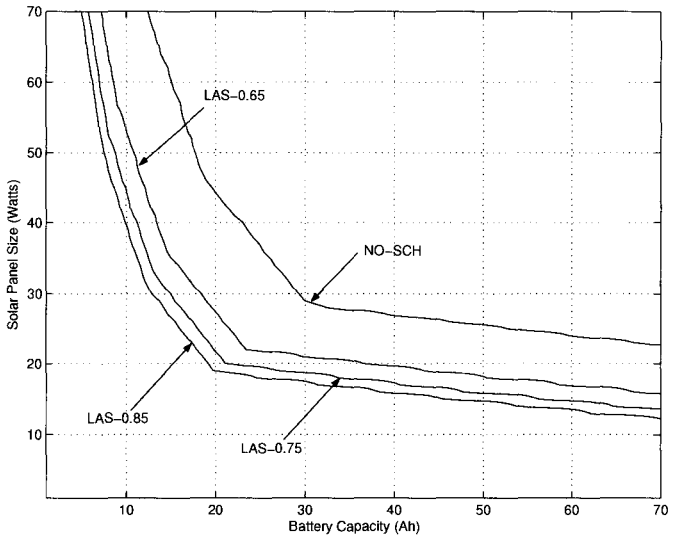


Figure 6.26: Solar Panel size vs. Battery Capacity combinations for the city of Toronto (CAN), for $P_{outage} = 0.0001$

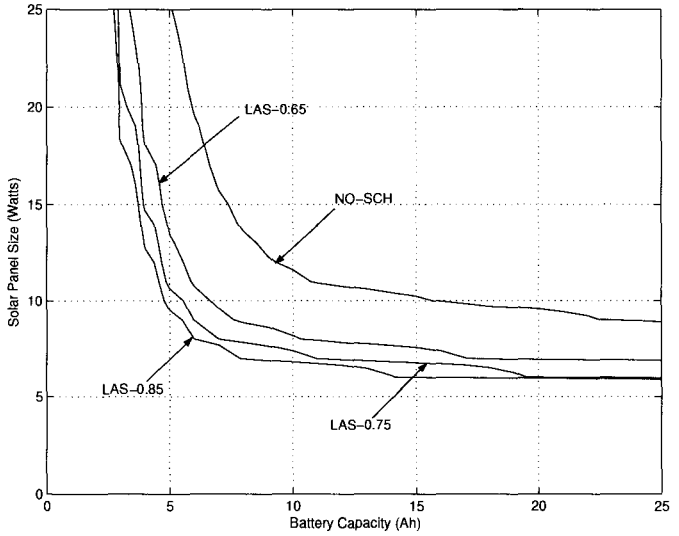


Figure 6.27: Solar Panel size vs. Battery Capacity combinations for the city of Phoenix (USA), for $P_{outage} = 0.0001$

	TORONTO		PHOENIX	
	Panel Size (W)	Battery Capacity (AH)	Panel Size (W)	Battery Capacity (AH)
NO SCHEME	29.00	29.93	11.00	11.73
LAS $U_{BEmax} = 0.65$	22.00	23.46	9.00	7.60
LAS $U_{BEmax} = 0.75$	20.00	21.15	8.00	7.00
LAS $U_{BEmax} = 0.85$	19.00	19.75	7.00	7.87

Table 6.3: Optimum solar panel and battery capacity for $\mathbf{P}_{outage} = 0.0001$

Chapter 7

Conclusions and Future Work

7.1 Conclusions

This thesis has presented two different algorithms that address the multi-coverage problem in solar/battery powered IEEE 802.11 WLAN networks. The schemes proposed guarantee the uninterrupted performance of delay and packet loss sensitive applications. In addition, when data traffic -which is modeled with self-similar and bulk traffic models- is present, the LAS algorithm showed very good performance in terms of power-saving, delay and handover rate. On the other hand, the PLAS algorithm provided slightly improved power-saving for the SNs, but at the expense of a very high handover rate. The LAS algorithm has a low handover rate making it suitable for real-time applications. The LAS algorithm is completely compatible with legacy APs and can be used in cases where all of them provide uninterrupted service all the time. Moreover, the LAS algorithm supports the new IEEE 802.11e standard. Additionally, a

CAC functionality has been defined at the APs for overlapping deployments to support HCCA based services. It was also demonstrated through experiments using a real traffic profile that both the LAS and PLAS algorithms saved on average between 22% and 35% of power depending on the selection of U_{BEmax}. Finally, using the hourly power consumption under this traffic profile, we were able to reduce significantly the optimum solar panel and battery size requirements compared to the case where no algorithms were used.

7.2 Future Work

There are several possible extensions to this work. One such extension is to make use of the power-saving capabilities in SNs as defined in [1], thus reducing the power consumption when SNs does not fully utilize their channels. Another extension is the integration of capacity-deficit control mechanisms defined in [6] to overcome unexpected traffic patterns that might occasion service outages.

Additionally, the use of beamforming directional antennas can be used by SNs to dynamically reconfigure Joint Coverage Areas according to the hot-spot location and traffic demand. A new scheme can also be proposed to modify the U_{BEmax} parameter dynamically based on characteristics of the traffic (i.e. mean interarrival time, mean packet size), this might allow for efficient use of the channel capacity and enhanced protection of real-time sessions.

Finally, a slight modification to the current IEEE 802.11 standard would permit a reduction of the handover time if STAs were able to know beforehand which SNs are willing to accept them through information included in the

Disassociation Notification Messages.

Bibliography

[1] Zhang F., Todd T.D., Zhao D. and Kezys, V., "Power Saving Access Points for IEEE 802.11 Wireless Network Infrastructure," *Wireless Communications and Networking Conference, 2004. WCNC IEEE*, vol. 1, March 21-25 2004.

[2] A. Balachandran, P. Bahl and G. M. Voelker, "Hot-Spot Congestion Relief in Public-area Wireless Networks," *Proc. of 4th IEEE Workshop on Mobile Computing Systems and Applications*, June 2002.

[3] G. Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordination Function," *IEEE Journal on Selected Areas in Communications*, March 2000.

[4] D. P. Blinn, T. Henderson, and D. Kotz, "Analysis of a Wi-Fi Hotspot Network," *Proceedings of the International Workshop on Wireless Traffic Measurements and Modeling (WiTMeMo '05)*, June 2005.

[5] Cisco Systems Inc., "Cisco Aironet Access Points Technical Documents," *http://www.cisco.com/global/AP/aironet/prod_ftr.shtml*, September 2003.

[6] A. Farbod, "Design and Resource Allocation for Solar-Powered ESS Mesh Networks," Master's thesis, McMaster University, 1280 Main St. West, Hamilton, Ontario, Canada L8S 4K1, August 2005.

[7] M. R. Garey and D. S. Johnson, *Computers and Intractability; A Guide to the Theory of NP-Completeness.* New York, NY, USA: W. H. Freeman & Co., 1990.

[8] E. Gentle, *Random Number Generation and Monte Carlo Methods, Second Edition.* New York, NY, USA: Springer-Verlag, 2003.

[9] J. Gordon, "Pareto Process as a model of self-similar packet traffic," *Global Telecommunications Conference, 1995. GLOBECOM '95., IEEE*, Nov 1995.

[10] Hector Velayos, Victor Aleo and Gunnar Karlsson, "Load balancing in overlapping wireless cells," *IEEE ICC 2004, Paris, France*, June 2004.

[11] D. S. Hochbaum, ed., *Approximation algorithms for NP-hard problems.* Boston, MA, USA: PWS Publishing Co., 1997.

[12] IEEE 802.11 Task Group S, "IEEE 802.11 ESS Mesh Networking PAR," *http://standards.ieee.org/board/nes/projects/802-11s.pdf*, August 2004.

[13] IEEE Computer Society LAN MAN Standards Committee, *IEEE 802.11: Wireless LAN Medium Access Control and Physical Layer Specifications.* IEEE Press, 1999.

[14] IEEE Computer Society LAN MAN Standards Committee, *IEEE 802.11: Wireless LAN Medium Access Control and Physical Layer Specifications:*

Medium Access Control (MAC) Quality of Service (QoS) Enhancements. IEEE Press, 2005.

[15] ITU, Telecommunication Standardization Sector Of, "ITU-T Recommendation G.114: One-way Transmission Time," tech. rep., International Telecommunication Union, May 2003.

[16] LAN/MAN Standards Committee of the IEEE Computer Society, *IEEE Recommended Practice for Multi-Vendor Access Point Interoperability via an Inter-Access Point Protocol Across Distribution Systems Supporting IEEE 802.11 Operation.* IEEE Press, 2003.

[17] McMaster University Wireless Networking Group, "The SolarMESH Network," *http://owl.mcmaser.ca/solarmesh/*, 2004.

[18] L. Narvarte and E. Lorenzo, "On the Usefulness of Stand-Alone PV Sizing Methods," *Progress in Photovoltaics: Research and Applications, Prog.Photovolt: Res.Appl.*, vol. 8, pp. 391–409, 2000.

[19] National Climate Data and Information Archive, "http://www.climate.weatheroffice.ec.gc.ca/." The Meteorological Service of Canada, Canada., 2004.

[20] National Solar Radiation Data Base, "http://rredc.nrel.gov/solar/." National Renewable Energy Laboratory (NREL), U.S. Department of Energy, 2004.

[21] Nortel Networks, "WLAN Security Switch 23XX," *http://products.nortel.com.*

[22] P. Bahl, V. N. Padmanabhan, and A. Balachandran, "A Software System for Locating Mobile Users: Design, Evaluation and Lessons," *Technical Report MSR-TR-2000-12, Microsoft Research*, February 2000.

[23] V. Paxson and S. Floyd, "Wide Area Traffic: The Failure of Poisson Modeling," *IEEE/ACM Transactions on Networking*, vol. 3, no. 3, pp. 226–244, 1995.

[24] Proxim Wireless Networks, "Orinoco Access Point Family," *http://www.proxim.com/learn/library/datasheets/AP–2000_A4.pdf.*

[25] S. Wang, L. Cuthbert and J. Bigham, "Agent-based Load Balancing of WLAN in In-door Usage," *3rd IASTED International Conference on Wireless and Optical Communications (WOC2003), Banff, Canada*, July 2003.

[26] F. Safie, "Probabilistic Modeling of Solar Power Systems," in *Reliability and Maintainability Symposium, 1989. Proceedings., Annual*, pp. 425–430, 1989.

[27] S.-T. Sheu and C.-C. Wu, "Dynamic Load Balance Algorithm (DLBA) for IEEE 802.11 Wireless LAN," *Tamkang Journal of Science and Engineering*, vol. 2, no. 1, pp. 45–52, 1999.

[28] O. Tickoo and B. Sikdar, "Queueing Analysis and Delay Mitigation in IEEE 802.11 Random Access MAC based Wireless Networks," *Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies INFOCOM*, 2004.

[29] B. Tsybakov and N. Georganas, "On Self-Similar Traffic in ATM Queues: Definitions, Overflow Probability Bound and Cell Delay Distribution," *IEEE/ACM Transactions on Networking*, June 1997.

[30] H. Velayos and G. Karlsson, "Techniques to Reduce the IEEE 802.11b Handoff Time," *Proceedings of the IEEE International Conference on Communications*, June 2004.

[31] W.E. Leland, M.S. Taqqu, W. Willinger and D. V. Wilson, "On the Self-Similar Nature of Ethernet Traffic (extended version)," *IEEE/ACM Transactions on Networking*, vol. 2, no. 1, pp. 1–15, February 1994.

[32] J. Zou and D. Zhao, "Admission Control with Load Balancing in IEEE 802.11-Based ESS Mesh Networks," *The Second International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks (QShine)*, August 2005.