7-2-2010

Improving the Energy Efficiency of IEEE 802.3az EEE and Periodically Paused Switched Ethernet

Mehrgan Mostowfi
University of South Florida

Improving the Energy Efficiency of IEEE 802.3az EEE and

Periodically Paused Switched Ethernet

by

Mehrgan Mostowfi

A thesis submitted in partial fulfillment
of the requirements of the degree of
Master of Science in Computer Science
Department of Computer Science and Engineering
College of Engineering
University of South Florida

Major Professor: Kenneth J. Christensen, Ph.D.
Miguel Labrador, Ph.D.
Jarred A. Ligatti, Ph.D.

Date of Approval:
July 2, 2010

Keywords: Green networks, power consumption, performance evaluation, simulation,
Packet Coalescing, adaptive shutdown, local area networks, LAN switches

to my mother

Acknowledgements

I would like to sincerely appreciate the great encouragement, support and guidance I received from my advisor Dr. Ken Christensen throughout this research. I would also like to thank Dr. Miguel Labrador and Dr. Jay Ligatti for agreeing to serve on my thesis committee and for their constructive comments.

My deepest gratitude goes to my mother Parvaneh, and my sister Mitra, for their lifelong love and support. This thesis would not have been possible without their constant encouragement and kindness.

Lastly, I offer my regards to all of those who supported me in any respect during the completion of this work.

Table of Contents

List of Tables		iii
List of Figures		iv
Abstract		vi
Chapter 1	Introduction	1
1.1	Background	1
1.2	Motivation	4
1.3	Contributions	6
1.4	Organization of this Thesis	6
Chapter 2	Background and Literature Review	8
2.1	Sleep and Wakeup	9
2.2	Rate Adaptation	13
2.3	Energy Proportionality	14
2.4	Energy Efficient Ethernet	15
2.5	Periodically Paused Switched Ethernet	21
2.6	Related Work in Wireless Networks	25
Chapter 3	Evaluation and Improvement of Energy Efficient Ethernet	27
3.1	Best and Worst Cases for EEE Overhead	27
3.2	EEE Simulation Model	29
3.3	Experiments on EEE	29
3.4	Results of EEE Experiments	31
3.5	Improving the Energy Efficiency of EEE by Packet Coalescing	33
3.6	Simulation Model of EEE with Packet Coalescing	33
3.7	Experiments on EEE with Packet Coalescing	38
3.8	Results of EEE with Packet Coalescing Experiments	39
3.9	Summary and Conclusions	48
Chapter 4	Periodically Paused Switched Ethernet	50
4.1	PPSE Fluid Flow Model	50
4.2	PPSE System Implementation	51
	4.2.1 Link Shutdown and Wakeup	51
	4.2.2 Energy Saving Opportunities and Mode Transitions	52
4.3	Simulation Model of Simple PPSE	53
4.4	Experiments on Simple PPSE	53
4.5	Results of Simple PPSE Experiments	55
4.6	Adaptive PPSE	57
4.7	Experiments on Adaptive PPSE	59

4.8	Results of Adaptive PPSE Experiments	62
	4.8.1 Poisson Traffic	62
	4.8.2 Bursty Traffic	68
4.9	Summary and Conclusions	74
Chapter 5	Economic and Environmental Benefits	77
5.1	Projected Benefits of EEE	77
5.2	Projected Benefits of PPSE	79
5.3	Summary and Overall Benefits of EEE and PPSE	81
Chapter 6	Summary and Future Work	82
6.1	Future Work	83
List of References		85

List of Tables

Table 2.1 – Proposed Wakeup, Sleep, Refresh and Quiet Times for EEE on 10GBase-T 16

Table 4.1 – Parameters of the Synthetic Packet Generator 61

Table 5.1 – Economic and Environmental Benefits of EEE 79

Table 5.2 – Economic and Environmental Benefits of PPSE 80

List of Figures

Figure 1.1 – 2006 U.S. Electricity Consumption (not to scale) [38]	2
Figure 2.1 – Ideal Power Consumption as a Function of Utilization	14
Figure 2.2 – Transitions between Active and Quiet Modes in EEE [27]	15
Figure 2.3 – FSM for EEE	17
Figure 2.4 – Energy Consumption of EEE as a Function of Load [46]	20
Figure 2.5 – High-Level Illustration of PPSE in a LAN Switch [6]	22
Figure 2.6 – FSM for Simple PPSE [6]	24
Figure 3.1 – Packet Transmission and Transitions in EEE – Best Case (not to scale)	28
Figure 3.2 – Packet Transmission and Transitions in EEE – Worst Case (not to scale)	28
Figure 3.3 – High-Level Illustration of EEE	29
Figure 3.4 – Power Consumption of EEE as a Function of Load	32
Figure 3.5 – FSM for EEE with Packet Coalescing	34
Figure 3.6 – Alternatives of A1 Transition	36
Figure 3.7 – High-Level Illustration of EEE with Packet Coalescing	37
Figure 3.8 – Power Consumption as a Function of Load for EEE with Packet Coalescing	39
Figure 3.9 – Average Packet Delay as a Function of Load for EEE with Packet Coalescing	41
Figure 3.10 – Histogram of Packet Delays (10% Load, Small Coalescer)	43
Figure 3.11 – Histogram of Packet Delays (10% Load, Large Coalescer)	43
Figure 3.12 – Histogram of Packet Delays (90% Load, Small Coalescer)	44
Figure 3.13 – Histogram of Packet Delays (90% Load, Large Coalescer)	44
Figure 3.14 – The Number of Packets Ahead in Downstream Queue (3% Load)	46

Figure 3.15 – The Number of Packets Ahead in Downstream Queue (6% Load) 46

Figure 3.16 – The Number of Packets Ahead in Downstream Queue (9% Load) 47

Figure 4.1 – Fluid Flow Model of Simple PPSE 51

Figure 4.2 – Average Packet Delay as a Function of Load – Simple PPSE 55

Figure 4.3 – FSM for Adaptive PPSE 58

Figure 4.4 – Power Consumption as a Function of Threshold – Adaptive PPSE 63

Figure 4.5 – Power Consumption as a Function of Load – Poisson Traffic 63

Figure 4.6 – Average Packet Delay as a Function of Load – Poisson Traffic 65

Figure 4.7 – Histogram of Packet Delays – Poisson Traffic (5% Load, Low Threshold) 66

Figure 4.8 – Histogram of Packet Delays – Poisson Traffic (5% Load, High Threshold) 66

Figure 4.9 – Histogram of Packet Delays – Poisson Traffic (25% Load, High Threshold) 67

Figure 4.10 – Power Consumption as a Function of Load – Bursty Traffic 69

Figure 4.11 – Average Packet Delay as a Function of Load – Bursty Traffic 70

Figure 4.12 – Histogram of Packet Delays – Bursty Traffic (5% Load, Low Threshold) 72

Figure 4.13 – Histogram of Packet Delays – Bursty Traffic (5% Load, High Threshold) 72

Figure 4.14 – Histogram of Packet Delays – Bursty Traffic (25% Load, High Threshold) 73

Improving the Energy Efficiency of IEEE 802.3az EEE and Periodically Paused Switched Ethernet

Mehrgan Mostowfi

Abstract

It is estimated that networked devices consumed about 150 TWh of electricity in 2006 in the U.S. which has cost around $15 billion and contributed about 225 billion lbs of CO_2 to greenhouse gas emissions. About 13.5% of this energy is consumed by network equipment such as switches and routers.

This thesis addresses the energy consumption of Ethernet, and designs and evaluates improvements on existing methods to reduce the energy consumption of Ethernet links and switches.

Energy Efficient Ethernet (EEE) is an emerging IEEE 802.3 standard which allows Ethernet links to sleep when idle. In this thesis, a performance evaluation of EEE is completed. This evaluation replicates previous work by Reviriego et al. in an independent manner. The performance evaluation shows that EEE overhead results in less energy savings than expected. A new method based on Packet Coalescing is developed and evaluated to improve the energy efficiency of EEE. Packet Coalescing bursts packets such that EEE overhead is minimized. The results show that EEE with Packet Coalescing for 10 Gb/s Ethernet can achieve very close to ideal (or energy proportional) performance at the expense of an insignificant added per packet delay.

Periodically Paused Switched Ethernet (PPSE) was previously proposed and prototyped by Blanquicet and Christensen in 2008. PPSE uses periodically sent notification packets to halt packet transmission into a LAN Switch and thus allowing the switch to sleep periodically. In this thesis, a first performance evaluation of PPSE is completed. The evaluation in this thesis shows that a PPSE for 10 Gb/s Ethernet LAN Switches achieves either significant energy savings at the expense of an excessive packet delay, or less than expected savings with a less than human response time added per-packet delay. An improvement to PPSE (Adaptive PPSE) is proposed and developed based on an adaptive policy. The adaptive policy considers past traffic load to determine whether to put the switch to sleep or not. The evaluation shows that Adaptive PPSE can achieve very close to ideal performance at the expense of an added average per packet delay which is less than half of the human response time.

Chapter 1: Introduction

One of the most challenging issues of the current century is the high consumption of energy which has lead to ever-increasing efforts to find and utilize new methods of conserving energy. The fact that even the most basic human activities directly or indirectly depend on energy turns shortages of energy into significant crises. Moreover, burning fossil fuels in order to produce energy not only releases several pollutant compounds, but also makes the main contribution to CO_2 production in the atmosphere. Although there are still doubts upon whether CO_2 emission is the real cause of global warming, but from scientific point of view it is mostly certain that CO_2 has properties that suggest it can contribute significantly to the warming of the planet [32]. It is estimated that global CO_2 emissions of ICT equipment (including PCs, printers, network devices, etc.) was about 2% of the estimated total emissions in 2007 [52]. Moreover, ICT "could deliver carbon savings five times larger than the total emissions from the entire ICT sector in 2020" [52] by enabling other sectors such as buildings and logistics to become more energy efficient. This work is concentrated on reducing the energy use of ICT devices; more specifically the energy consumption of computer networks.

1.1 Background

The Internet has grown rapidly during the past three decades. The number of hosts connected to it has increased from almost zero to approximately 680 million from early 1980s to July 2009 [29]. Consequently the electricity consumption of these hosts increased. As can be seen in Figure 1.1 shows that networked electronic devices consumed almost 4% (150 TWh) of all the electricity in the U.S. in 2006. About 13.5% of this 150 TWh is consumed by network equipment

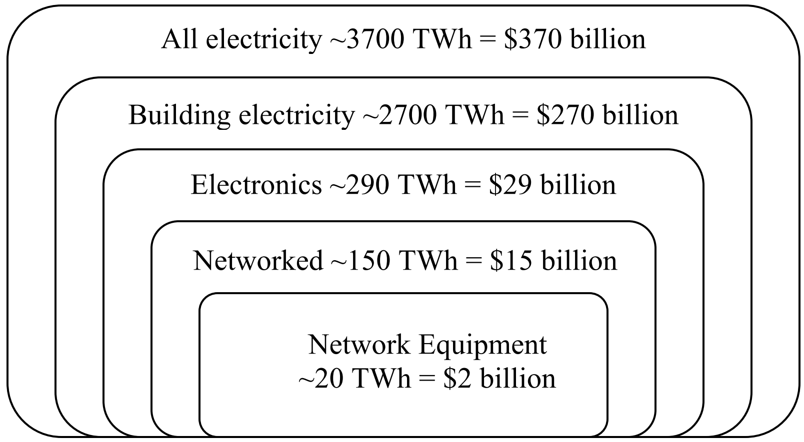

Figure 1.1 – 2006 U.S. Electricity Consumption (not to scale) [38]

such as switches and routers [38]. CO_2 emissions caused by burning fossil fuels to produce such amount of energy are significant too. A coal power plant emits about 1.5 lbs of CO_2 to produce 1 kWh of electricity ([7] and [11]) which makes the contribution of networked devices to greenhouse gas emissions about 225 billion lbs in 2006. [45] reports that the electricity consumption of data centers and servers in 2006 was 1.2% of the total U.S. electricity consumption which cost about $4.5 billion at that year. Additionally, network hosts which are not in data centers also consume a great amount of electricity. It is reported by [39] that enterprise or office PCs (including their monitors) consumed about 65 TWh/year, which is approximately equal to 5 percent of commercial building electricity.

Data centers and network hosts are each a part of some sort of a computer network which may or may not be connected to the Internet. However, the majority of computer networks are connected to the Internet nowadays to communicate and share resources. The energy consumption of computer networks is likely to increase drastically in the future for two major reasons. First, that more and more hosts get connected to the Internet everyday which makes it necessary for a host to become a part of a computer network first. Second, the practice of leaving the equipment fully powered-on all the time is being increasingly institutionalized for numerous reasons. These reasons include maintaining the ability of accessing the equipment remotely, sharing the resources on the computers while they are not in use and the inconvenience that a shutdown and powering-on causes.

Two types of energy consumption are considered for computer networks:

- Direct energy consumption; which is the direct electricity use of network equipment such as network adapters in PCs, switches, hubs, routers, etc.
- Induced energy consumption; in which computer networks cause other equipment such as a PC to consume energy. This type which is more subtle and complicated in terms of measurement and quantization, is the consumption caused by computer networks. This is when a device such as an office desktop is being left powered on, or its power management features is being disable, solely for the purpose of remaining present in the network. A common example of induced energy consumption by networks is a PC which is connected to an external Skype phone set. The PC must be left on all the time in order for the phone to ring whenever an incoming call is made to the Skype number.

This thesis concentrates on the direct energy consumption of computer networks and two improvements on the existing methods of reducing this consumption. Specifically, this thesis addresses the problem of the energy consumption of Ethernet.

Many approaches to reduce the energy consumption of both Ethernet switches and network adapters have been proposed, studied and evaluated. This thesis builds on and improves two existing methods; IEEE 802.3az Energy Efficient Ethernet (EEE) aimed for Ethernet link and Periodically Paused Switched Ethernet (PPSE) which is aimed for Ethernet LAN switches. A brief explanation of each method follows. Thorough explanation and details of each method will be presented in Chapter 2.

EEE is standard for twisted pair Ethernet by IEEE 802.3az workgroup [27] to enable the link to enter a low-power idle (LPI) mode when no traffic is on the link to transmit. By entering LPI mode, the physical layer is put to sleep which reduces the overall energy used by the link. The transition between Active and LPI modes takes an amount of time which is shown by Reviriego et al. [46] that is too high comparing to actual packet transmission time. The high transition times introduce an overhead on EEE which makes its energy efficiency less than ideal.

PPSE is a method which allows Ethernet LAN switches to periodically enter sleep mode in order to save energy. To prevent packet loss caused by packet transmissions while the switch is in sleep mode, packet transmission on all the links connected to the switch is halted before transition to sleep mode by means of a special notification packet. PPSE was proposed and prototyped by Blanquicet and Christensen in 2008 [6]. It was shown by their initial evaluation that PPSE is able to save significant amount of energy at the expense of an acceptable packet delay.

1.2 Motivation

The work in this thesis is motivated by the following two major factors:

- The network bandwidth is lightly utilized most of the time ([41] and [42]) while the energy consumption of links does not change significantly with the change in utilization [14].
- The absolute power consumption of Ethernet links is significant at present and will increase in the future as data rates increase to 10 Gb/s and higher.

Studies show that network links are typically under-utilized. Studies described in ([41] and [42]) showed that common edge links (links from edge devices like PCs to switches) work at much lower than their full capacity most of the time. The study in [41] which is done on the University of Toronto Ethernet network links in 1998 showed that the utilization of the network is approximately 1% of its capacity during a week period of time. In [8], traffic traces of top 100 users with the most traffic volume on the network are collected and studied. The study showed that the greatest utilization of the network capacity caused by the traffic volume of these users was 7.2%. Another study which is explained in detail in [42] showed that the average link utilization of LBNL enterprise network from December 2004 and January 2005 had been "2 to 3 orders of magnitude less than the capacity of the network". The study also mentioned that full utilization of the network capacity only occurred in rare cases in which swift bursts of data transfer happened.

It is reasonably expected that the energy consumption of an Ethernet link decreases when the link is lightly utilized. But it has been shown otherwise. The study in [14] which is done on a Cisco Catalyst 2970 network switch and Intel PRO 1000/MT Network Adapter shows that the difference between energy consumption of the Ethernet link under low and full utilization of the capacity is not significant.

The factors mentioned above lead to a question which is why to keep a link powered on all the time when it is used less that 10% of the time? It seems reasonable to think of ways to power-down or turn off the link when there is no traffic to transmit.

Another motivating factor of this work is the absolute power consumption of Ethernet links. As measurements of [14] show the increase in power consumption from 100 Mb/s to 1 Gb/s is about 4 W. This consumption increases between 10 to 20 W in case of 10 Gb/s. Considering the high number of deployed Ethernet links worldwide, the power consumption of the total links appears to be even more significant when 10 Gb/s start to become the default standard for desktops, laptops and other networked devices due to increase in data transfer volumes and the constant demand for more bandwidth among users. Therefore, although the methods studied in this work are applicable to all speed magnitudes of Ethernet over twisted pair medium, the simulation experiments are done on 10 Gb/s twisted pair (10GBASE-T) Ethernet links.

1.3 Contributions

The contributions of this work are as following:

1) An improvement on EEE called Packet Coalescing is designed and evaluated. The evaluation shows that EEE with Packet Coalescing for 10 Gb/s Ethernet can achieve very close to ideal performance at the expense of added packet delay of about 50 microseconds.
2) An improvement on PPSE is designed and evaluated which makes the decision of turning off the switch based on the past traffic on the links to the switch. The evaluation shows that Adaptive PPSE can achieve very close to ideal performance at the expense of an added average per packet delay of less than half of the human response time.

1.4 Organization of this Thesis

The chapters of this thesis are organized as follows:

- Chapter 2 presents a literature review of the past scientific research done on characterizing the high energy consumption of computer networks and methods to improve their energy efficiency. The related research done in wireless and wired networks are also reviewed in this chapter.
- Chapter 3 presents a performance and energy consumption evaluation of EEE and designs a method to improve it. The improved method is evaluated by means of simulation.
- Chapter 4 presents a first performance evaluation of PPSE and proposes an adaptive policy to improve its performance. The adaptive policy is evaluated by simulation in this chapter.

- Chapter 5 estimates the environmental and economic benefits of EEE and PPSE with the proposed improvements.
- Chapter 6 concludes this work and suggests possible related future work in the field.

Chapter 2: Background and Literature Review

Energy consumption of the Internet was not a primary concern prior to the 21st century since the global deployment of the Internet was at its early stages and the total energy consumed by the Internet was negligible. Prior to 2003, the research on energy consumption of network devices was confined to architecture and VLSI levels for the components of the device in which approaches to lower the energy consumption of hardware components of interconnection networks such as the microprocessor and switching fabric were explored ([53] and [21] for example). However, none of them addressed the energy consumption of networks as a whole.

In 1999, the opinion of the energy consumption of the transferring data over the Internet being too high was expressed in [23] by Huber. He estimated the energy required to transfer 2 MB of data over the Internet to be equivalent to burning 1 lbs of coal. However, the concern of the energy consumption of the Internet did not attract much attention until 2003 where Gupta and Singh used the absolute values of the energy consumed by various network devices to show that although the relative percentage of this consumption seemed low, the absolute energy consumption is too high to be neglected anymore [18]. They summarized the energy consumption of Hubs, LAN and WAN switches and routers in a table and calculated the total of 6.05 TWh for the total energy consumption of network devices in 2000. They also noted that this total consumption is almost 0.07 percent of total U.S. energy expenditure of 2000 which seems to be negligible. However, there are good reasons to still consider reducing this consumption. They brought forward three main reasons which were the current energy inefficiency, the opportunity for vaster deployment of the Internet and benefits in case of disaster. To show the significance of the first reason, they compared the energy consumption per byte between the current wired

Internet and the ideal consumption in wireless communication and concluded that wireless is between 1.25x and 10x times more efficient. The second reason is that lower energy consumption allows the Internet to be deployed in energy-scarce parts of the world and the third reason was that in case of power shortage, networking equipment can operate longer on their UPS batteries if they are made more energy efficient.

Given the above reasons to show the need for reducing the energy consumption of networks, the approaches to reduce the direct energy consumption of wired networks can be classified to two major categories as following:

- Sleep and Wakeup; in which all or some of the components of a device is put to sleep in order to save energy.
- Rate Adaptation; in which the rate at which the device works is lowered when possible in order to consume less energy.

The work done in this thesis belongs to Sleep and Wakeup category where improvements on two existing approaches to put Ethernet links and switches to sleep are studied.

In this chapter, the related work in Sleep and Wakeup and Rate Adaptation categories are reviewed first. Energy Proportionality which is considered the ideal target of energy consumption of the methods in this thesis will be described next. Since half of the contributions of this thesis are on improving EEE (Chapter 3) and the rest builds on PPSE (Chapter 4), two subsections are devoted to describing these two methods in detail. Section 2.4 explains EEE and its overhead in certain cases in detail and Section 2.5 explains PPSE and the past research which proposed and initially evaluated it. Finally, similar approaches taken in wireless networks are reviewed.

2.1 Sleep and Wakeup

Sleep and Wakeup approaches have been extensively explored for network switches and links. Gupta et al. [20] investigate the possibility of putting a LAN switch to sleep during periods of low utilization. The LAN traffic they collected showed that there are significant periods of inactivity which motivated them to propose an algorithm to put the link to sleep during these periods. Their algorithm estimates the mean time to arrival of the next packet. If the estimation shows a sleep opportunity that compensates for the energy wasted in transition from awake to sleep states, the link goes to sleep and is woken up upon arrival of the next packet. They addressed the possibility of losing the first arriving packet to a sleeping LAN switch by using Hardware Assisted Buffered Sleep (HABS) in which an incoming packet wakes the link up and is also buffered. Although HABS solves the problem of losing the first arriving packet, it consumes much more energy than when simple sleeping is used in which any packet received will wake the link up but is lost. They project 1.8 to 2.5x energy savings using HABS which results in no loss. However, more energy saving of between 2x and 4.5x is possible using their method provided that simple sleeping is used which results in about 7.5 percent of loss.

Gupta et al. [19] propose a method and an improvement over it (On/Off-1 and On/Off-2) to determine when and how to turn off the links of an Ethernet LAN switch based upon expiration of a timer or upon the load on the link exceeding a threshold. Based on their evaluation of the algorithm, 28 to 84% energy saving is possible using On/Off-2 method while results additional packet delay and packet loss especially when the traffic is highly bursty.

Tamura et al. [55] believe that the in-rush current caused by frequently changing the mode of the switch (as proposed by Gupta et al.'s On/Off algorithm) will destroy the circuitry of the switch and thus must be avoided to the maximum extent. They introduced their Extra Active Period (EAP) model according to which the link does not enter the sleep state even when the buffer is empty. Instead, it remains in active mode for another active period and if the buffer still remains empty after this extra period puts the switch to sleep. They evaluated their algorithm both numerically and analytically using M/M/1 and IPP/M/1 queueing models and concluded that assuming Poisson traffic on the link, their method improves the average number of turn-on instances while lowering the power reduction ratio.

Ananthanarayanan et al. [1] propose a similar approach to Gupta et al.'s On/Off scheme which eliminates the need for the ports to be on during the idle periods and takes better advantage of extended idle periods. They propose Time Windows Prediction (TWP) method which predicts the future traffic on each of the ports of the switch by observing the number of packets in a sliding window of time. If the number of packets observed is below a defined threshold the switch powers down the port. The packets arriving to the powered-down ports are being buffered by means of shadow ports. They report about 30% energy savings using their approach.

Rodriguez-Perez et al. [48] build on On/Off algorithm to remove two of its shortcomings. Their algorithm postpones sleeping if the buffer is not completely empty when On/Off algorithm decides to go to sleep. Instead, their algorithm waits until the buffer is drained and then determines the sleeping period. The second shortcoming they fix is that as opposed to On/Off their method does not try to maximize total sleeping time. Instead it determines if entering sleep state for the calculated period compensated for the cost to wakeup again or not. If it is determined not to be a "worthy" sleeping interval, the method chooses to keep the link active and idle than to put it to sleep. With these two improvements, they show that their method improves both on average delay and energy savings compared to On/Off algorithm. However, the degree of packet loss is slightly higher when their algorithm is used.

Buffer-and-Burst (B&B) method proposed by Nedevschi et al. [37] shapes the traffic at network edge routers in order to enable some of the routers inside the network to sleep. In this method all the edge routers shape the traffic into small bursts. Then they transmit the bursts at the same time to ingress routers thus the ingress routers will be able to process all the incoming burst with one sleep and wake transition. In other words, all the edge routers synchronize the time of their burst transmissions. Since synchronizing all the edge routers is not practically feasible, the practical method they devise is for each edge router to send its bursts at once as a single "train of bursts". As a result, bursts disperse as they get further from the originating edge router. Practical B&B showed significant overall energy savings in the network of up to 80%.

Another approach to reduce the energy consumption of the switches is called Periodically Paused Switched Ethernet (PPSE) which is first introduced by Blanquicet and Christensen [6] with a different name (Pause Power Cycle). PPSE will be studied in detail later in this chapter and in Chapter 4. In brief, PPSE starts a timer, sends a Pause Notification to all the links connected to the switch to force them not to send any traffic for the same period of it requested traffic stop, and powers down. When the timer expires, it exits sleep state and resumes normal activity for a period of time, and the procedure repeats.

In addition to Ethernet LAN switches, sleep and wakeup methods have also been devised for Ethernet links. The first sleep and wakeup method for Ethernet links is proposed by Gupta et al. [17]. This method, called Dynamic Ethernet Link Shutdown (DELS), determines if putting the link to sleep is feasible or not based on the number of packets already buffered in the link's transmission queue as well as the mean inter-arrival time of the packets. If putting the link to sleep is determined to be feasible, the length of the sleep is computed based on a given maximum sleep interval and a maximum buffer size. Arriving packets to the link when in sleep mode are buffered to be transmitted when the sleep period is over. The method shows to save significantly energy for light link loads of less than 5%, but fails to save any significant energy for higher loads. The increase in delay caused by this method is reported to be of the order of less than 1 ms.

Energy Efficient Ethernet (EEE) [27] is another sleep and wakeup approach for Ethernet links. EEE will be covered in detail both in this chapter and the next. In brief, EEE is a method to bring the energy consumption of Ethernet links closer to proportional to link's utilization by putting the link to a defined low-power mode when there is no packet to transmit and quickly take it back to active mode as soon a packet arrives to the link. The main difference between EEE and DELS is that EEE puts the link to sleep only when the link's buffer is empty and wakes up upon packet arrival.

2.2 Rate Adaptation

Rate Adaptation methods are the approaches that bring the link's rate closer to its utilization without putting it to sleep in order to reduce the energy consumption. The first Rate Adaptation method for wired networks was Adaptive Link Rate (ALR) method which was proposed in 2005 [40] by B. Nordman and K. Christensen. ALR is evaluated both analytically and by simulation for Ethernet links and switches in [13], [14] and [15]. ALR switches the rate of the link between high and low based on the queued packets in the link's buffer. The simplest rate change policy of ALR (Dual Threshold Policy [13]) uses two thresholds on the link's output buffer; Low and High Thresholds. If the number of packets in the queue exceeds the high threshold, the rate is changed to high. If it decreases below the Low Threshold the rate is changed to low provided that both sides of the link agree. This policy causes rate oscillation under certain traffic loads which increased response time and variability. To minimize link rate oscillation another policy is proposed for ALR (Utilization-Threshold policy [15]) according to which the number of bytes counted during a time period is used in order to make the rate change decision.

ALR method is inspired by the power reduction approach taken in new versions of Asynchronous Digital Subscriber Line (ADSL), ADSL2 and ADSL2+. In these versions, three power modes are defined; L0 in which the ADSL link is fully functional and consumes maximum power, L2 in which the link is still active but works at a reduced rate and consumes less power, and L3 in which the link is idle and powered off [10]. A link transitions from L0 to L2 when the link's transmission buffer is empty or almost empty and to L3 mode when the user is not online [57].

The most recent Rate Adaptation method is presented in [37] called practRA in which a similar approach to ALR is taken with a few differences. Most importantly, practRA uses just one threshold to determine rate change and avoids oscillation by using the packet arrival history to predict the rate in near future and only changes the rate when made sure that the future link utilization is not more than the high rate.

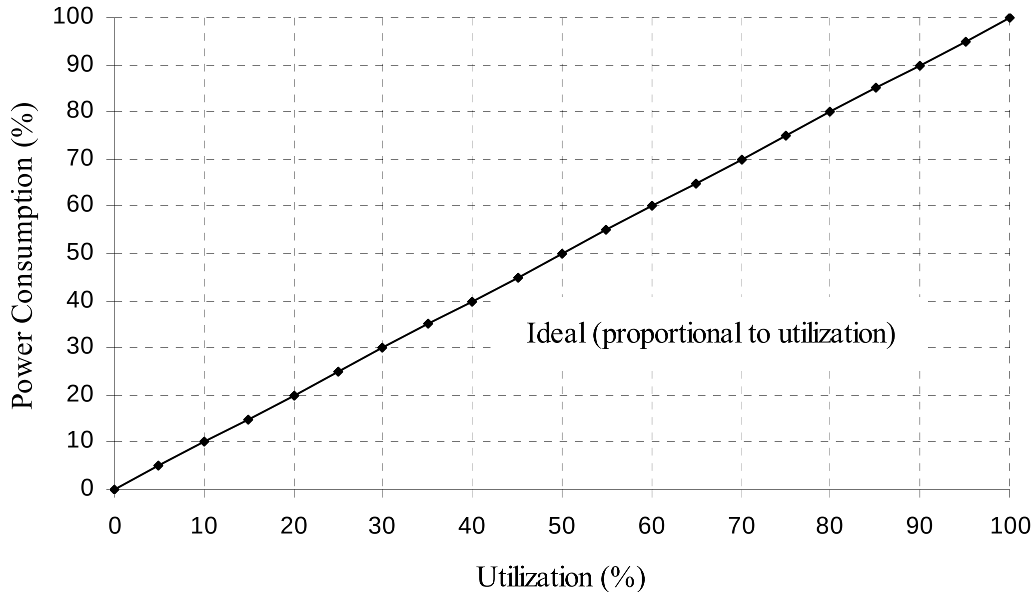

Figure 2.1 – Ideal Power Consumption as a Function of Utilization

2.3 Energy Proportionality

Energy Proportionality or Energy Scaling is the capability of a system to consume just as much energy as it needs to complete its workload. An energy proportional system consumes no energy when not being utilized and its energy consumption grows in proportion to its utilization [5] and [56].

The ideal energy conservation methods presented in this work is to bring the consumption as close as possible to the utilization on the device. Utilization here is solely determined by the traffic load on the device (or the link) and not by any other factor such as computations needed to determine the route on a switch to which a packet will be forwarded or decomposition and analyzing packet headers. Therefore, if for example the aggregate packet load on a 10 Gb/s Ethernet switch is 2 Gb/s, the utilization of the switch is 20%, and the ideal energy consumption of it under such load is considered to be 20% of the device's peak energy consumption. The proportionality of power consumption to utilization is depicted in Figure 2.1.

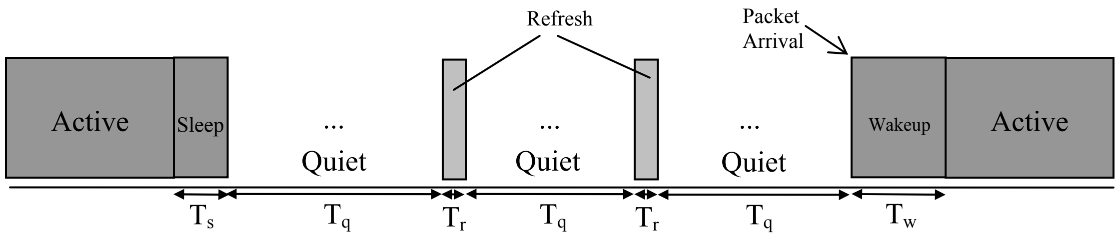

Figure 2.2 – Transitions between Active and Quiet Modes in EEE [27]

2.4 Energy Efficient Ethernet

Energy Efficient Ethernet (EEE) [25] is an emerging IEEE 802.3 standard to reduce the energy consumption of Ethernet links and bring it closer to the ideal consumption which is proportional to the utilization of the link.

Two modes are defined in EEE; Active mode and Quiet mode. In Active mode the link is powered-on to transmit packets, but as soon there is no more data to send the link enters a low-power mode called Quiet mode. In Quiet mode, the physical layer is powered off and the elements in the receiver are stopped. Therefore, energy can be saved while in Quiet mode. As soon as a packet arrives to the link, the link is woken up quickly in a few microseconds in order to resume packet transmission. The states and transitions occurring in EEE are depicted in Figure 2.1.

In Figure 2.1, T_s is the time needed for the link to enter Quiet mode, T_w is the time needed for the link to exit Quiet mode and go back to Active mode again. The link spends T_q time in Quiet mode which is the time during which energy is saved. Finally, in order to refresh the receiver and to maintain the alignment of receiver elements to channel conditions the link has periodic activity during T_r.

It is only during T_w, T_s and T_r that the link consumes significant power while during T_q only almost 10% of the links full consumption is needed. Although different depending on the manufacturer, but it can be conservatively assumed that the link consumes the same amount of power as when active during transitions. So, the power consumption during T_w, T_s and T_r is assumed 100% hereafter. The proposed wakeup, sleep, refresh and quiet times in EEE as appeared in [27] are presented in Table 2.1.

Table 2.1 shows the minimum and maximum (if applicable) values of T_s, T_w, T_q and T_r for 10GBase-T links. Note that since T_r is fixed, its minimum and maximum are the same. Also, the minimum and maximum of T_q are the same provided that no packets are generated during a T_q. Arrival of a packet breaks a T_q.

Table 2.1 – Proposed Wakeup, Sleep, Refresh and Quiet Times for EEE on 10GBase-T

Min T_w (μs)	Min T_s (μs)	Max T_s (μs)	Min and Max T_r (μs)	Min and Max T_q (μs)
4.48	2.88	3.2	1.28	39.68

The functionality of EEE can be explained as a Finite State Machine (FSM). Note that this FSM as well as all other FSMs used in this thesis follow the below syntax for states and transitions:

- States are shown by vertical lines. The state's name is mentioned above the state. Time passes by when inside a state.
- Transitions are shown with arrows from the source state to the destination state. The condition under which the transition is made is stated above the arrow and the action taken upon the transition being made is stated below the arrow. Transitions take place instantaneously. Each transition is given a label to be used to reference the transition which is stated at the bottom of the beginning of the arrow (or right of the arrow if it is vertical).

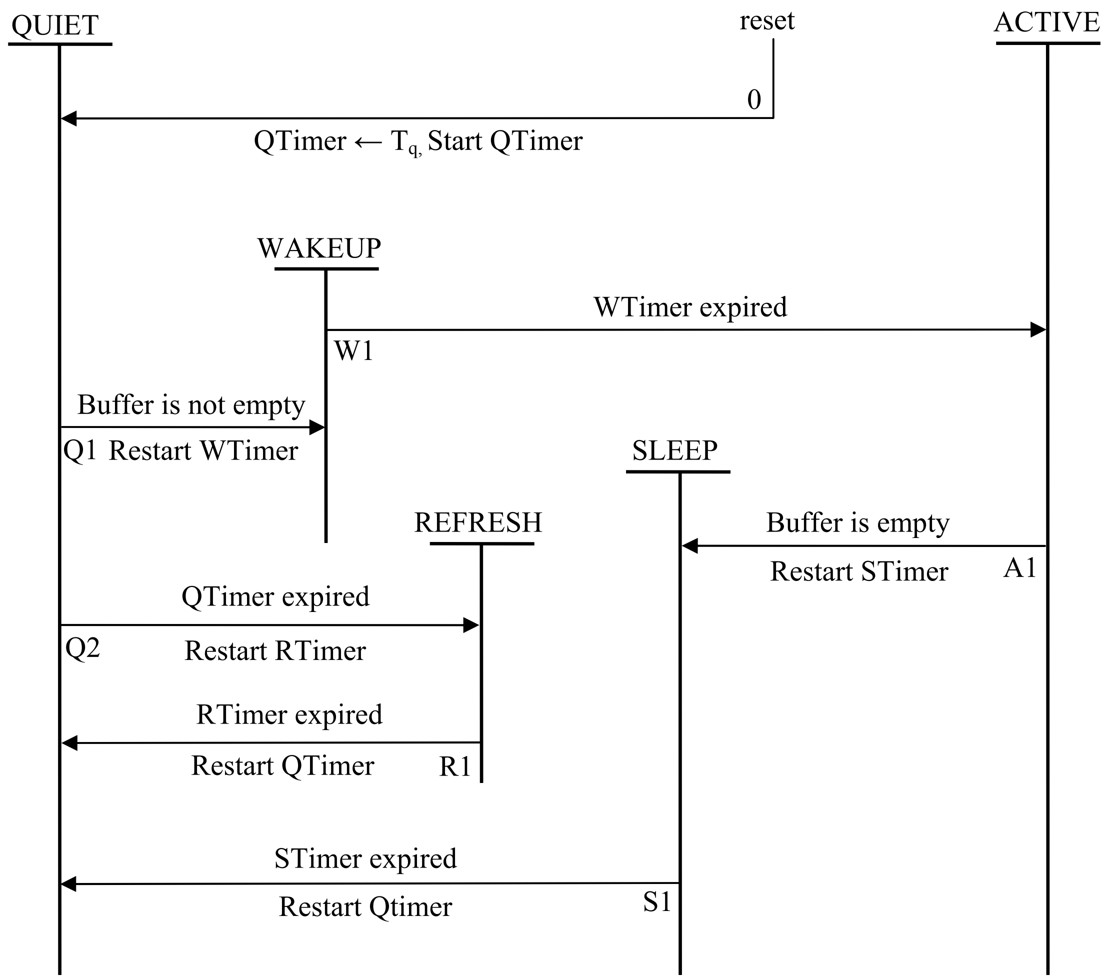

Figure 2.3 – FSM for EEE

Figure 2.3 shows the FSM for EEE. The FSM has five states defined as following:

- ACTIVE: The link is fully powered-on and operational. Packets are queued in the buffer and transmitted.
- QUIET: The link is in low-power or quiet mode.
- WAKEUP: The link is powering-on.
- SLEEP: The link powering-down.
- REFRESH: The link is being refreshed.

Four timers are defined as following:

- STimer: Keeps time spent in SLEEP state.
- WTimer: Keeps time spent in WAKEUP state.
- RTimer: Keeps time spent in REFRESH state.
- QTimer: Keeps time spent in QUIET state.

These timers are initialized to the following values upon restart respectively:

- T_s: Time to remain in SLEEP state.
- T_w: Time to remain in WAKEUP state.
- T_r: Time to remain in REFRESH state.
- T_q: Time to remain in QUIET state.

Note that when a timer is restarted, it means it is set to its initial value and starts to count down to zero. When the value of the timer reaches zero it expires. Also note that Buffer is a FIFO queue in which the packets are stored and served in the same order they arrived to the queue. A detailed explanation for each of the FSM transitions follows:

- Transition 0: The link starts in QUIET mode. QTimer is set to its initial value and starts to count down to simulate the time spent in quiet mode.
- Transition Q1: When one or more packet(s) are generated so the buffer is not empty, the link wakes up to transmit the packet by entering WAKEUP state. WTimer is restarted to simulate the time spent to wake the link up.
- Transition Q2: QTimer is expired. The link is refreshed by entering REFRESH state. RTimer is restarted to simulate the time spent to refresh the link.
- Transition W1: WTimer is expired. The link enters ACTIVE state.
- Transition A1: There link has transmitted all the packets in the buffer. The buffer is empty, so the link starts to power down by entering SLEEP state. STimer is restarted to simulate the time spent to power the link down.

- Transition S1: STimer is expired and no packets are generated while the link was being powered-down so the buffer is empty. The link enters QUIET state. QTimer is restarted to simulate the time spent in quiet mode.
- Transition R1: RTimer is expired. The link enters QUIET state. QTimer is restarted to simulate the time spent in quiet mode.

As can be seen in Table 2.1, $T_q >> T_r$. Therefore, although possible, but it is very unlikely that the link stays in quiet mode for an entire T_q and so needs to be refreshed. For this reason, the refresh times are ignored hereafter in the FSM based on EEE's FSM as well as simulation models for simplicity. By ignoring link refresh, there is no need for RTimer and so is removed. Moreover, QTimer is no longer needed and removed because the link will be allowed to stay in quiet mode as long as no packets are generated. Other states and transitions are the same except for REFRESH state which is removed.

The wakeup and sleep transitions introduce an overhead to EEE which results in poor energy saving gains compared to ideal. This inefficiency was first studied by Reviriego et al. [46]. They used the following example in order to show the inefficiency of EEE.

Let us assume that a single packet is sent each time the link wakes up, and all packets are of the size 1500 bytes (Maximum Transmission Unit in Ethernet), so it takes 1.2 µs to transmit a single packet (T_{pkt}). Using minimum T_s and T_w values from Table 2.1 and ignoring refresh times, the ratio of the time devoted to actual packet transmission over the overall time required to transmit the packets including transitions can be computed as following:

$$\frac{T_{pkt}}{T_{overall}} = \frac{T_{pkt}}{T_w + T_{pkt} + T_s} = \frac{1.2\mu s}{4.48\mu s + 1.2\mu s + 2.88\mu s} = 0.14$$

Only 14% of the time required to transmit a packet (8.56 µs) is devoted to actual packet transmission (1.2 µs) and the remaining is spent in transition. In other words, 86% of time and thus energy consumed by the link is wasted in transition in this example.

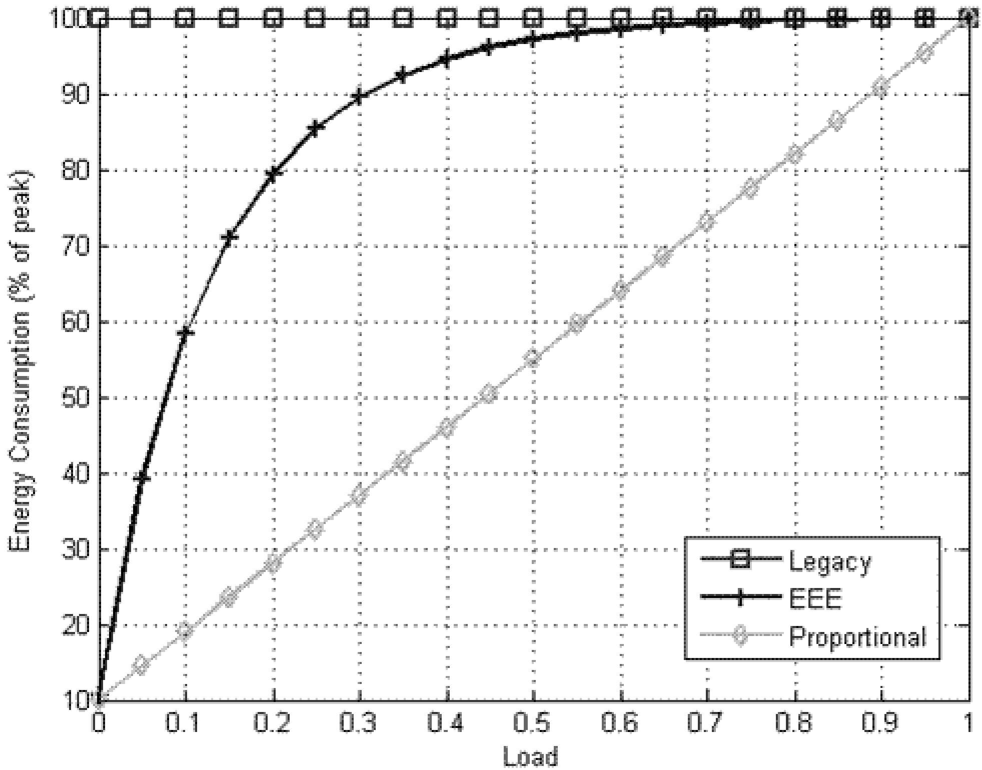

Figure 2.4 – Energy Consumption of EEE as a Function of Load [46]

Reviriego et al. evaluated the power consumption of an EEE 10GBase-T link by means of simulating packet arrivals to a link using MATLAB and compared it to proportional power consumption [48]. Figure 2.4 shows their results for 10 Gb/s links. In this figure the difference between the proportional power consumption and the consumption resulted by using EEE demonstrates the inefficiency.

One approach to decrease the overhead of transitions on EEE, is grouping packets and sending then as one burst of back-to-back packets. As a result, only the time and energy of one wakeup and sleep transition is wasted for transmitting a number of packets which reduces the transition overhead. This idea is proposed and initially evaluated by Reviriego et al. [47].

2.5 Periodically Paused Switched Ethernet

Periodically Paused Switched Ethernet (PPSE) is a method of saving energy in Ethernet LAN switches proposed and prototyped by Blanquicet and Christensen in 2008 [6] under a different name (Pause Power Cycle). In PPSE, all the links connected to a LAN Switch are stopped from sending any traffic at the same time for a fixed period of time. Therefore, the entire switch (or at least the majority of its components) can be turned off during this period of time. PPSE is specifically intended for edge network switches since these are the most lightly utilized switches in the network with many idle periods. Therefore, there is a good opportunity of saving energy on these switches while keeping the adverse effect on the performance of the network as low as possible.

PPSE seems similar to and is indeed motivated by Gupta et al.'s On/Off algorithm. However, the two have a fundamental difference. In On/Off algorithm the future sleeping opportunities are predicted and the method tries to exploit them by putting the switch to sleep during these times. In PPSE, the sleeping opportunities are "artificially" made by the switch (by sending Pause Notifications). In other words, PPSE knows the idle periods while On/Off guesses them. This difference leads to eliminating the event of packet loss during off times in PPSE since it is already made sure that none of the links sends any traffic during the idle periods. Moreover, since PPSE stops all the links at the same time, more components of the switch can be powered off compared to On/Off and other approaches described earlier.

Figure 2.5 shows a high-level view of how PPSE works. A notification message (referred to as Pause Notification hereafter) indicating that the device connected to a link must not send any traffic for an arbitrary interval is sent by the switch on all the links connected to it. The switch then enters a low-power state in which a number of its components are powered off (OFF state). When the interval elapses, the switch resumes to fully operational state and resumes servicing packets (ON state).The time that the switch spends in OFF and ON states are called T_{off} and T_{on}

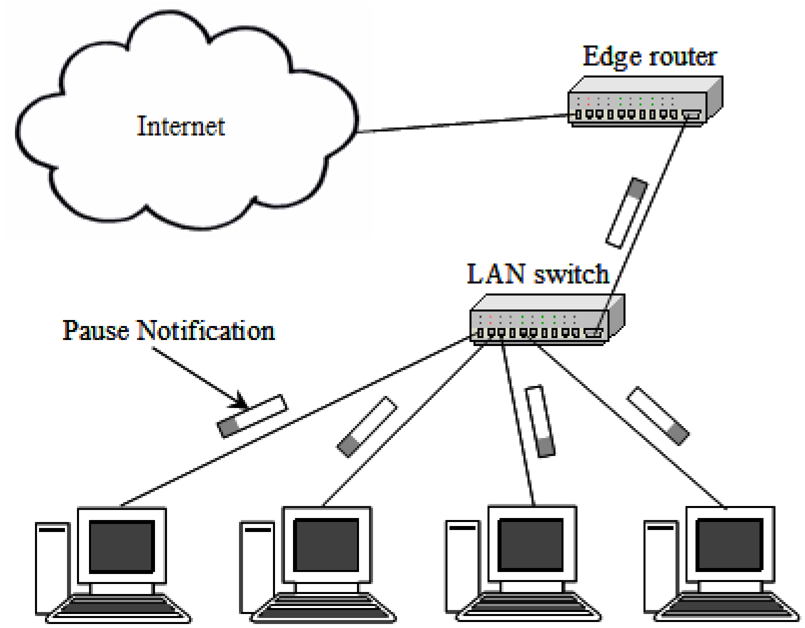

Figure 2.5 – High-Level Illustration of PPSE in a LAN Switch [6]

respectively. With respect to T_{off} and T_{on}, a parameter called Duty Cycle (D) is defined as follows:

$$D = \frac{T_{on}}{T_{on} + T_{off}} \quad \text{(Eq. 2.1)}$$

D determines the percentage of time the switch spends in ON state. If transition times and power consumption in transitions are neglected, the power consumption of the switch can be determined by D.

By fixing T_{off} and D, T_{on} can be determined using Equation 2.2:

$$T_{on} = \frac{D \times T_{off}}{1 - D} \quad \text{(Eq. 2.2)}$$

Equation 2.2 will be used in performance evaluation of PPSE where T_{on} is determined using D and T_{off}.

One of the mechanisms that could be used to notify a Network Interface Card (NIC) to stop sending any traffic for a period of time is the optional flow control operation known as "PAUSE" frames which is defined in Ethernet standard. PAUSE frames notify the NIC to temporarily stop the flow of traffic (except for MAC Control frames) for a certain period of time [27]. PAUSE frames are intended to allow an end of a connection to recover from congestion state by stopping the other end from transmitting more traffic temporarily. PAUSE frames can be exploited by PPSE in order to make sure that no traffic will be received and powering off would not cause packet loss.

Supporting Ethernet PAUSE frames is optional for the NICs that implement full-duplex protocol. Since 10 Gb/s Ethernet is full-duplex, PAUSE frames cannot be devised as a reliable mechanism to stop the incoming traffic from the NICs. However, for the NICs that support PAUSE frames, the MAC Control Parameters field could be set to T_{off} value (with proper conversion to 4096-bit time) to stop the traffic from the link. The intention of presenting PAUSE frames here is to show that it is possible to stop traffic on a link for a period of time. In any case, the implementation of PPSE in real switches requires some sort of Pause Notification to be supported by MAC or PHY layers of NICs.

The simplest version of PPSE (the version proposed in [6]) is when the switch stays in ON state for a fixed period of time and then enters the OFF state and stays there for a fixed period of time and the process repeats. The function of Simple PPSE can be shown by a Finite State Machine (FSM) which is depicted in Figure 2.6. The FSM has two states defined as following:

- ON: The switch is fully operational
- OFF: The links connected to the switch are paused and the switch is powered down

Two timers are defined as following:

- TON: Keeps time spent in ON state.
- TOFF: Keeps time spent in OFF state.

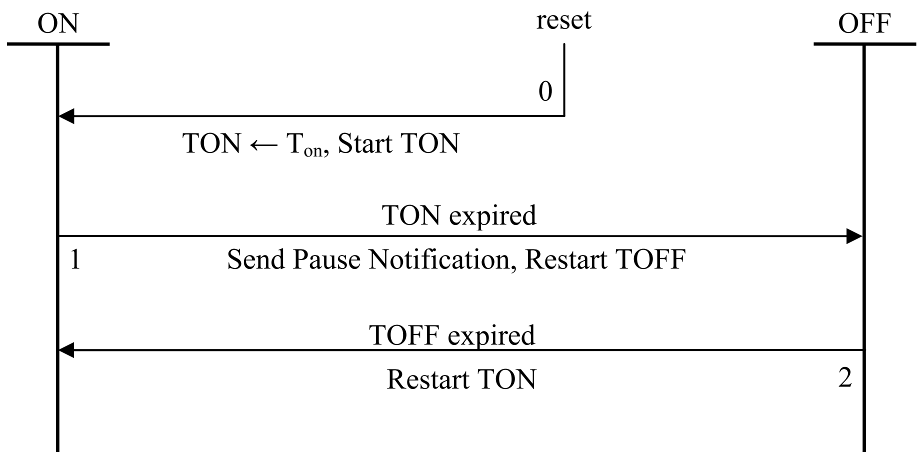

Figure 2.6 – FSM for Simple PPSE [6]

These timers are set to the following values upon restart respectively:

- T_{on}: Time to remain in ON state
- T_{off}: Time to remain in OFF state

An explanation for each of the FSM transitions follows:

- Transition 1: When Simple PPSE starts or resets, the switch enters the ON state, the timers TON and TOFF are set to the initial timer values T_{on} and T_{off} and the TON timer starts.
- Transition 2: When TON expires, Pause Notifications are sent on all the links connected to the switch, TOFF is reset and starts to count down and the switch enters the OFF state.
- Transition 3: Upon expiration of TOFF, TON is reset to its initial value, T_{on} and starts to count down, the switch returns to ON state and the entire procedure is repeated.

This FSM will be modified in Chapter 4 in order to develop an improvement over Simple PPSE which is called Adaptive PPSE.

Although it appears that significant energy savings can be gained using PPSE, it comes with a tradeoff in performance. An increased delay in the devices connected to the switch is expected due to lack of the switch's service during time spent in OFF state. The performance of Simple and Adaptive PPSE will be evaluated in Chapter 4 by means of simulation.

2.6 Related Work in Wireless Networks

Energy efficiency in wireless networks has been subject to excessive research. A comprehensive survey of the research done in this field presented in [2]. The most significant related work done in wireless networks are briefly reviewed.

Wireless interfaces are used in mobile devices and sensor networks which are both battery powered. The wireless network interface consumes a significant portion of the battery power in these devices, so improving their energy consumption will have a great impact on the battery life [54].

Power management methods in wireless devices have been studied in different network layers which include transport layer, network layer (energy efficient routing algorithms) and in MAC layer. Only the research done in MAC layer is relevant to the work presented here and will be reviewed briefly. Energy efficiency in MAC layer explores methods that enable the wireless network interface to sleep while there is no data to transmit or receive while at the same time maintain the devices presence in the network.

Methods such as Sparse Topology and Energy Management (STEM) [49] and Power Aware Multi-Access protocol (PAMAS) [51] are among the most significant approaches to increase the time a wireless network interface spends in stand-by mode in order to conserve energy.

STEM uses one radio for transmitting and receiving wakeup signals and another radio for packet transmission. The data transmission radio is in stand-by mode and only turns on when the wakeup radio receives a wakeup beacon from a neighbor node. The wakeup radio is also in stand-by mode but is periodically turned on to listen to beacons from other nodes. Therefore the neighbor that needs to initiate a connection send a stream of beacons to ensure that at least one is received by the wakeup radio.

PAMAS schedules the times of packet transmission for each node and powers down the node when it is not scheduled to transmit. Using PAMAS, nodes scheduled to transmit data if the do not overhear transmission. Otherwise they are shut off since existing transmission means that other nodes in the neighborhood cannot transmit data without causing collision.

Another approach which is widely used as part of the 802.11 standard [25] is a beaconing method in which the wireless device informs the access point that it will put the network interface to stand-by. While the interface is in stand-by mode, the access point buffers the packet destined to it and sends a beacon packet periodically. The interface wakes up periodically to listen to the beacon from the access point and receive its packets.

Another solution is presented in [16] (Low-Power Distributed MAC) which proposes that the main radio is turned off when not transmitting or receiving while another ultra low-power receiver listens to control messages and wakes the main radio up whenever there are packets to receive.

Synchronized sleep and wakeup scheduling methods are also devised for wireless networks; methods such as Fully Synchronized Pattern [33]. In this method, all the nodes in the network are put to sleep at the same time for a fixed period. Upon expiration of this period all the nodes wake up and stay on for a fixed period and the process repeats. Although this approach assures that all the nodes are active at the same time, it has two caveats. First, turning on some of the nodes may not be necessary during all the active periods and second, time synchronization among all the nodes is difficult. However, this approach is very trivial to implement.

Chapter 3: Evaluation and Improvement of Energy Efficient Ethernet

In this chapter, the best and worst cases of EEE transition overhead are explained first. The simulation model used to evaluate EEE is then presented. Experiments are then designed using this simulation model of which results are presented next. Second, a method called Packet Coalescing is presented in order to reduce the energy inefficiency of EEE. The simulation model used to evaluate EEE with Packet Coalescing is explained next which is used to design experiments. The results of these experiments are explained next followed by summary and conclusions.

3.1 Best and Worst Cases for EEE Overhead

The inefficiency of EEE which is caused by the overhead of Sleep and Wakeup transitions of EEE and results in poor energy saving gains are described in the previous chapter. A best and a worst case can be considered for the transition overhead that EEE introduces. When packets arrive back-to-back, the NIC is able to wakeup once, transmit them all in one large burst of packets and then enter the quiet mode and stay there until the next burst arrives. In this scenario, only one sleep and one wakeup transition is required per burst which makes the percentage of time the NIC spends in full-power mode very close to the utilization. Figure 3.1 illustrates this scenario. The worst case happens when packets arrive with a fixed inter-arrival time of exactly $T_w+T_{pkt}+T_q$. In this case, as soon as the link finishes a sleep transition, a packet arrives to the link and the link wakes up to transmit the packet. So no time would be spent in quiet mode while only a small fraction of time is spent transmitting the packet and the rest is wasted in transition. In this scenario, a sleep and a wakeup transition is required for each packet and the link does not spend

Figure 3.1 – Packet Transmission and Transitions in EEE –
Best Case (not to scale)

any time in quiet mode. Therefore, the percentage of time the NIC spends in full-power mode is 100% while the utilization of the link is very low. Figure 3.2 illustrates this worst case scenario.

The best and worst cases can happen in real Internet traffic. The best case often occurs in the form of file downloads using TCP where large blocks of data are burst onto a link (for example, from a server to a client) at a very high rate. In this case only one sleep and wakeup transition is required to transmit the entire burst. However, a close-to-worst case traffic occurs in the form of TCP ACKs being returned from the downloading client to the server. These TCP ACKs are typically small packets and are spaced-out evenly. In this case an overhead is inserted per ACK packet.

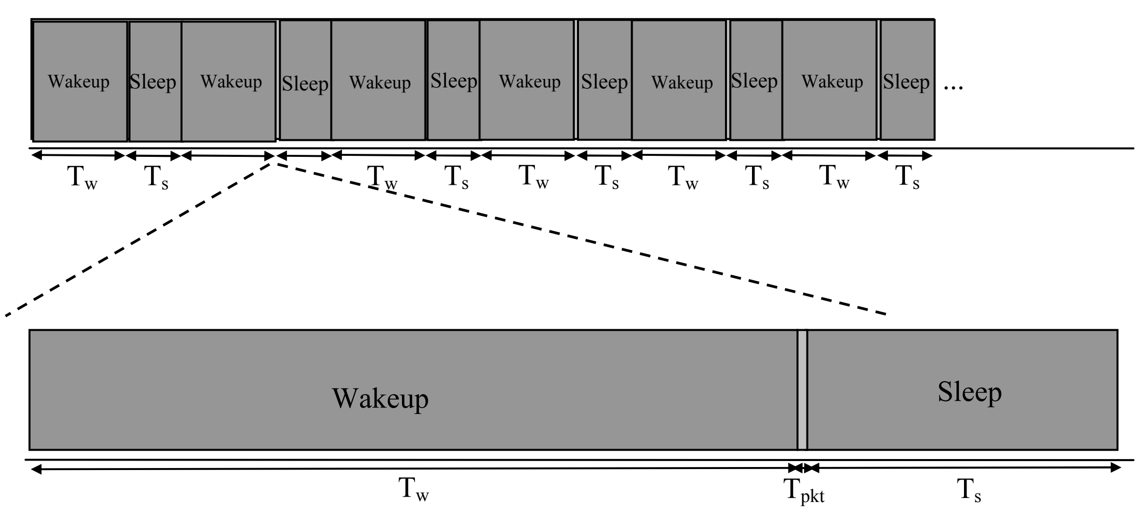

Figure 3.2 – Packet Transmission and Transitions in EEE –
Worst Case (not to scale)

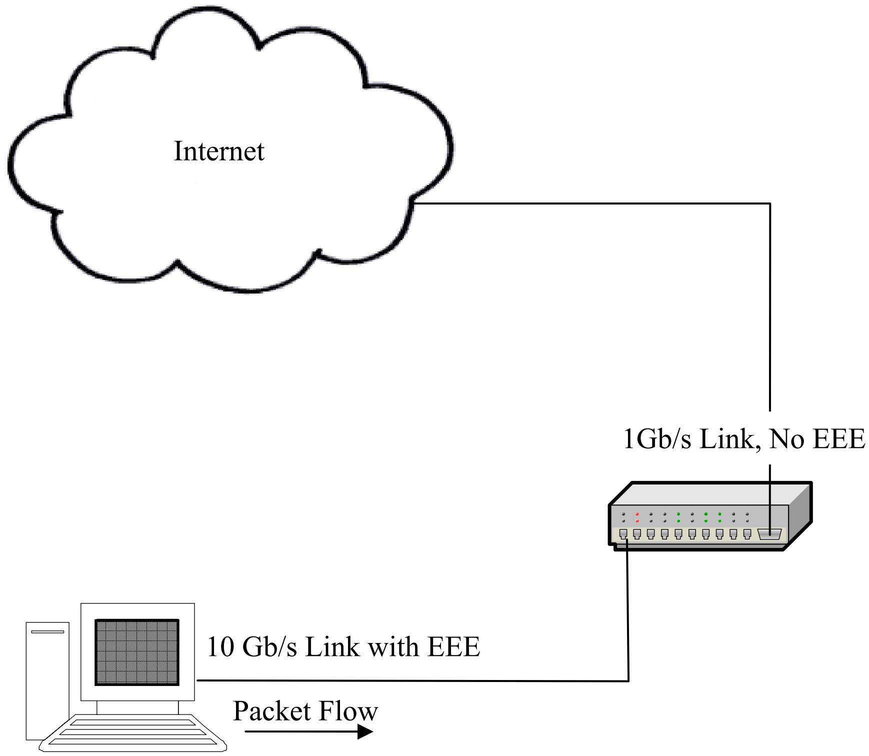

Figure 3.3 – High-Level Illustration of EEE

3.2 EEE Simulation Model

In order to further study the inefficiency of EEE and study an approach to reduce it in the subsequent sections, a simulation model of EEE is developed and validated using CSIM simulation library [50]. The model simulates the system view depicted in Figure 3.3. Packets are generated at the PC and arrive to the 10 Gb/s EEE link which works according to the FSM of EEE depicted in Figure 2.4. In the model, simulated packets arrive to a single server queue (the server simulates the link) which works according to the FSM of Figure 2.4.

3.3 Experiments on EEE

The simulation model described in section 3.1 is used in this section to design experiments in order to evaluate the performance of EEE and study its drawbacks. In the experiments, the response variables of interest are:

- Power consumption of the link; which is computed as a percentage of the link's consumption when working at full power using Equation 3.1:

$$P_{link} = P_{quiet} \times T_{quiet} + P_{active} \times (T_{active} + Tr_s + Tr_w) \qquad \text{(Eq. 3.1)}$$

  in which T_{quiet}, T_{active}, Tr_s and Tr_w are the percentages of times spent in quiet mode, active mode, sleep transition and wakeup transition respectively. P_{quiet} is assumed 10% according to the estimation made by different NIC manufacturers during the standardization process of EEE. The power consumption in active mode is obviously 100% of the link's consumption. The power consumption during transitions (P_{active}) is also assumed 100% based on the reasons given in the description of EEE in the previous chapter.

- Average packet delay

The factors in these experiments are:

- Transition times, T_s and T_w; set to their minimums as presented in Table 2.1.
- Load; a percentage of the link's capacity. This factor is varied between 0 and 95%.
- Distribution of packet arrivals and packet size; set to Poisson distribution with fixed packet size of 1500 B.

The first factor is independent of the other two factors. However, Load and Distribution of the packets as well as the size of the packets are dependant factors. For instance, by varying the Load, the inter-arrival time in Poisson distribution must be changed accordingly. Or for a fixed load, when the packet size changes the inter-arrival time in Poisson distribution must be changed also to obtain the same load. The transition times, distribution and size of the packets are the two fixed factors in the experiments performed in this section. However, the Load factor is varied to determine the effects of it on the response variables. In other words, a Simple Experiment [30] is designed by varying one factor and fixing the others to determine the effects on the response variables.

Note that it is well-know that the distribution of the packets is not Poisson in a real link. However, a Poisson distribution is still a valid first-order approximation of the distribution and shape of aggregated traffic on a link in some cases [31] and also serves as an intermediate case between the best and worst cases that were explained earlier in section 3.1. Additionally, 1500 B represents the packet size of which the increasingly growing large files are being transferred over the Internet. Therefore Poisson distribution and fixed 1500 B packet sized will be suitable approximation for the evaluation of EEE here.

The results of the above experiment will be presented in the next section. The experiment is done on large enough number of packet arrivals to obtain a 95% confidence level on the mean average delay of the packets. The confidence interval within which the mean point falls is smaller than 10% of the mean value in all cases. Note that the Average Packet Delay will not be presented in this section but it will be in section 3.7 where the Average Packet Delay caused by EEE is compared to the delay caused by Packet Coalescing.

3.4 Results of EEE Experiments

Figure 3.4 shows the power consumption of the link as a function of the load on the link. Also shown in Figure 3.4, are the current power consumption of the links – 100% all the time – and the ideal consumption which is proportional to the load. Note that these results are not a first look and the performance of EEE, but replicate the results published in [46] which are explained earlier in Chapter 2. The proportional consumption is computed using Equation 3.2, in which P_{quiet} and P_{active} are the power consumption in the low-power and active modes which as in Equation 3.1 are substituted with 10% and 100% respectively.

$$P = P_{quiet} \times (1 - Load) + P_{active} \times Load \qquad \text{(Eq. 3.2)}$$

As illustrated in Figure 3.4, the link currently consumes 100% power regardless of how much load is on the link. Using EEE, the power consumption decreases significantly, especially

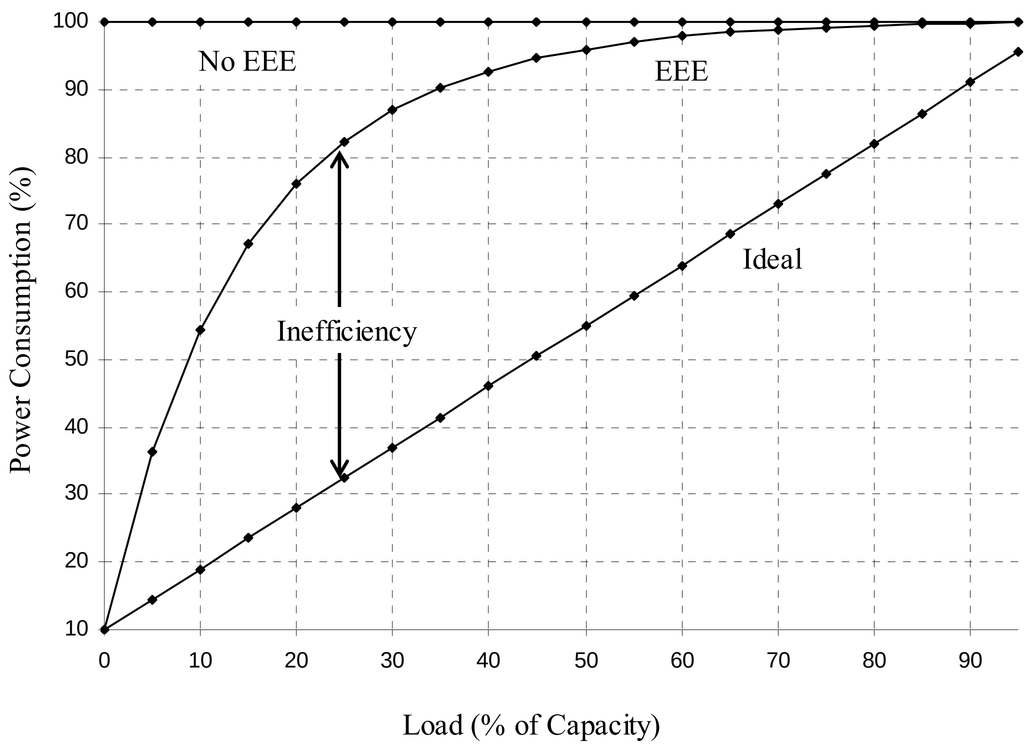

Figure 3.4 – Power Consumption of EEE as a Function of Load

when the load is low. However, it is still far from proportionality. For instance, when there is 10% load on the link, it consumes about 53% power, whereas the proportional power consumption for this load is 19%. The inefficiency is labeled in Figure 3.4. The reason for this inefficiency is the large values of the sleep and wakeup transition times compared to the actual time required to transmit a packet.

According to the results shown in this section, the transition overhead of EEE increases its power consumption compared to the proportional power consumption which is our ideal target. In the next section, a method to decrease this inefficiency is presented and evaluated.

3.5 Improving the Energy Efficiency of EEE by Packet Coalescing

An explained in Chapter 2, in order to decrease the overhead of transitions on EEE, packets can be "coalesced" into groups and then be sent as one burst of back-to-back packets. As a result, only one wakeup and sleep transition is used for transmitting a number of packets which reduces the transition overhead.

The idea of coalescing packets for EEE is similar to Receive Side Coalescing (RSC) [36] to some extent. RSC is a software-transparent mechanism which combines a number of packets belonging to the same TCP flow into a single large packet before forwarding it to the upper layers in TCP/IP stack. This is to keep the header of the first packet in the each flow (and changing it accordingly) and remove the rest of packets' headers and combine all the payloads together. As a result, the interrupt overhead on the processor is reduced. However, Packet Coalescing method is different in two aspects. First, the coalescing here is on transmission side. Second, coalescing of the packets here does not mean combining the packets into one large packet. It means buffering the packets while not modifying their headers or their payloads and then transmitting them back to back as a large burst of distinct packets. Nevertheless, the existence of Receive Side Coalescing shows the technical feasibility of implementing coalescing for send side in the NICs.

In the next section, the simulation model used to simulate an EEE enabled link which coalesces packets is explained.

3.6 Simulation Model of EEE with Packet Coalescing

Figure 3.5 shows the FSM for EEE modified to coalesce packets before transmission. This FSM is a modified version of the FSM presented for EEE in Figure 2.7. While the states of the FSM remain the same, a timer, a variable, and a constant are added to the FSM as following. Note that refresh times are ignored and therefore the REFRESH state in Figure 2.7 along with its corresponding transitions are removed before modifying the FSM for coalescing.

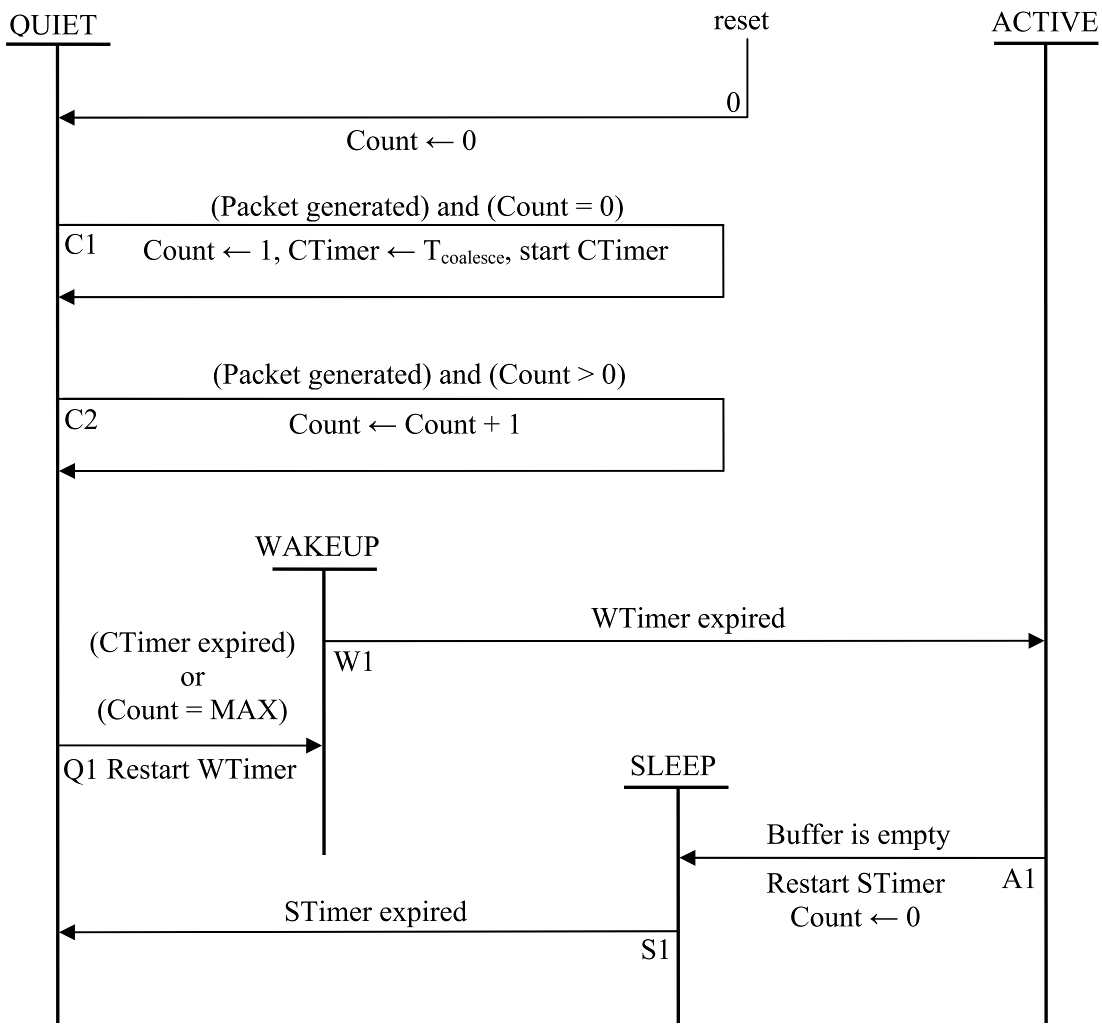

Figure 3.5 – FSM for EEE with Packet Coalescing

- CTimer: The timer which keeps the Packet Coalescing time. Coalescing resets when this timer expires. CTimer is set to $T_{coalesce}$ when restarted.
- Count: The variable which keeps the number of coalesced packets.
- MAX: Maximum number of packets that are allowed to be coalesced before transmission.

Additionally, three transitions are added and two are modified as following:

- Transition 0: Modified to initialize Count.

- Transition C1: When the first packet is generated, Count is set to 1 and CTimer is restarted which means that CTimer is set to $T_{coalesce}$ and starts to count down.
- Transition C2: Upon generation of the following packets, Count is incremented.
- Transition Q1: This transition is modified to be made when either Count has reached MAX or CTimer has expired. In this case WTimer is restarted and the FSM enters WAKEUP state. When restarting WTimer, it is set to its initial value of T_w and starts to count down.
- Transition S1: This transition is modified to reset Count to 0 upon returning to QUIET state.

The added and modified transitions result in the FSM to buffer packets in Buffer until either CTimer expires or Count reaches MAX when it wakes up and start transmitting the buffered packets back-to-back in the same order they entered the buffer. The FSM remains in ACTIVE state until all the buffered packets and the packets buffered during transmission of the buffered packets are transmitted and Buffer is empty. Thus, more than Count packets can be sent each time the ACTIVE state is entered.

There are two alternatives to transition A1. These alternatives are shown in Figure 3.6. The first transition shows an alternative to transition A1 which limits the number of packets transmitted to LIM regardless of whether the packets are buffered while the link is coalescing packets or the arrived while transmitting the coalesced packets thus confining the size of each burst to the number of packets. Note that LIM is a constant which can be set equal to or greater than MAX. This alternative would be an appropriate choice over the current transition if it is known that the size of the receiving side buffer is limited to LIM packets and more than this number of packets in a burst may result in packet loss. However, it is assumed here in the simulation in this section that the receiving side buffer is much larger than the coalescing buffer which is mostly the case when the EEE enabled link is connected to a switch or a router.

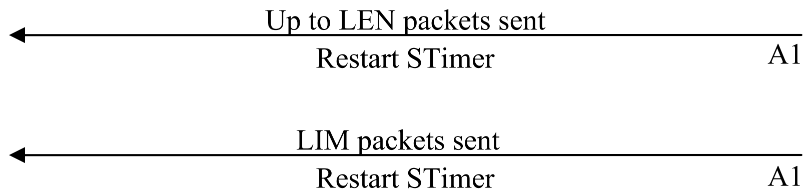

Figure 3.6 – Alternatives of A1 Transition

The second transition shows another alternative to transition A1 which limits the number of packets to only the ones that are buffered while in QUIET state. LEN is the number of packets buffered at transition. This alternative would be an appropriate choice over the current transition if two separate queues are used for Packet Coalescing and packet transmission. One queue can be used to buffer the packets, while the packets buffered in the other queue are transmitted. The two queues are switched at transition Q1. This is more of an implementation preference which is not the focus here. Note that in order to use either of these alternatives, the (Buffer is Empty) condition should be removed from A1 transition and Count should not be set to 0 anymore since some packets might still be remaining in the buffer upon exiting the ACTIVE state. Instead, as each packet from the buffer is sent in the ACTIVE state, Count should be decremented.

In addition to the reasons not to choose any of the alternatives, there are two reasons to choose A1 transition in Figure 3.5 over its alternatives; 1) it makes the largest bursts, and 2) it does not delay the packets generated while in ACTIVE state as opposed to the alternatives which do.

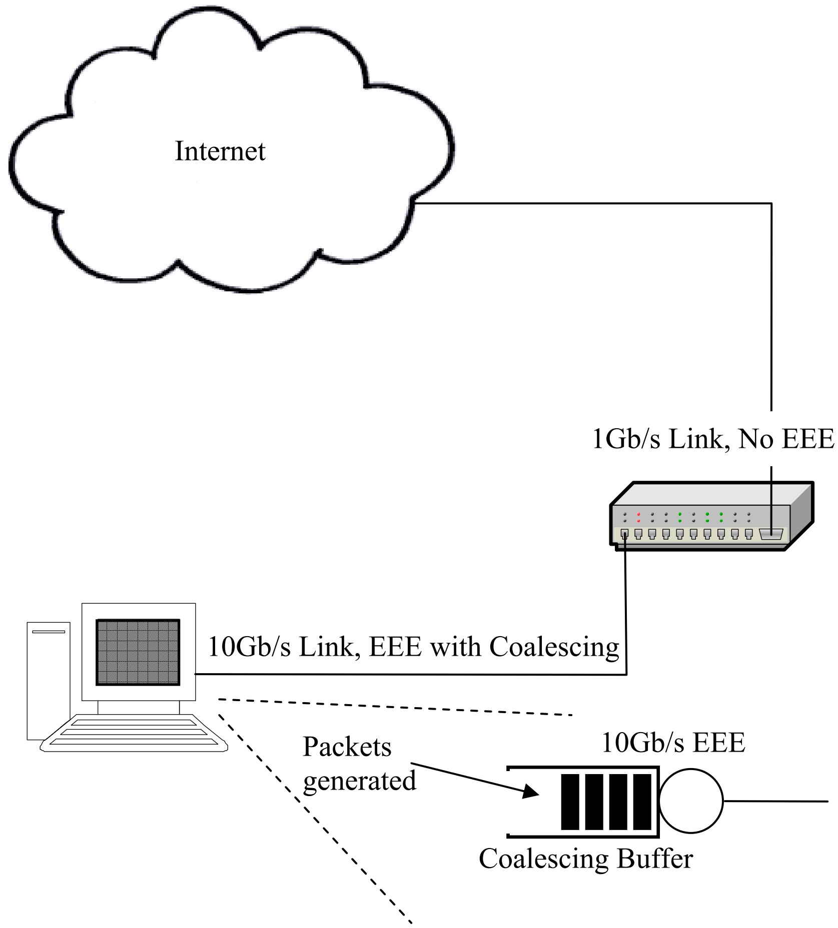

Figure 3.7 – High-Level Illustration of EEE with Packet Coalescing

A high-level illustration of EEE with Packet Coalescing is depicted in Figure 3.7. In this model, a PC is connected to an Ethernet Switch with a 10 Gb/s EEE with Packet Coalescing link. The switch is connected to the Internet with a 1 Gb/s ordinary Ethernet link. It is reasonable to assume that this setup will appear in homes or small offices in near future very often where a 10 Gb/s LAN is used locally and the connection to the Internet is made by a slower link of 1 Gb/s. Packets are generated at the PC and arrive to the 10 Gb/s EEE link with Packet Coalescing which works according to the FSM depicted in Figure 3.5. After being transmitted, packets are received by the switch and are then transmitted at 1 Gb/s (without EEE) to the Internet. Since the capacity of the link from the switch to the Internet is 10 times lower than the one connecting the PC to the switch, it is expectable for the packets to be queued prior to transmission at the switch. This queue will be used to evaluate the increased burstiness caused by Packet Coalescing. In the simulation model, simulated packets arrive to a single server queue (the server simulates the link) which works according to the FSM in Figure 3.5. The downstream queue is modeled with another single server queue which services the packet in order in which they are received.

3.7 Experiments on EEE with Packet Coalescing

In the experiment designed to evaluate Packet Coalescing in this section, the response variables of interest are the same as experiments for EEE explained in Section 3.3 except for the following added response variable which is used in order to study burstiness:

- The number of packet ahead of each arriving packet to the downstream queue; as an indicator of burstiness.

Moreover, the coalescing parameters are added to the factors:

- Coalescing Timer, $T_{coalesce}$ and Coalescing buffer size, MAX; two sizes of a coalescer are simulated here by setting these parameters accordingly; "Small Coalescer" and "Large Coalescer". For the Small Coalescer, 12 µs and 10 packets are used for these factors respectively. For the Large Coalescer, 120 µs and 100 packets are used. Note that the values for $T_{coalesce}$ are chosen to be similar to the ones that are already in use for Receive Side Coalescing which is described in section 3.3. The common timer values for RSC are 25 µs to 125 µs [36] . The value of MAX is chosen to be equal to the number of 1500 B packets transmitted in one $T_{coalesce}$ time.

The load factor and $T_{coalesce}$ and MAX are varied to determine the effects of them on the response variables. In other words, a Fractional Factorial Experiment [30] is designed by varying two factors for certain factor levels and fixing the other factors to determine the effects on the response variables. Two experiments are done with the Small and Large Coalescer of which results will be presented and explained in the next section. The experiments are done on large

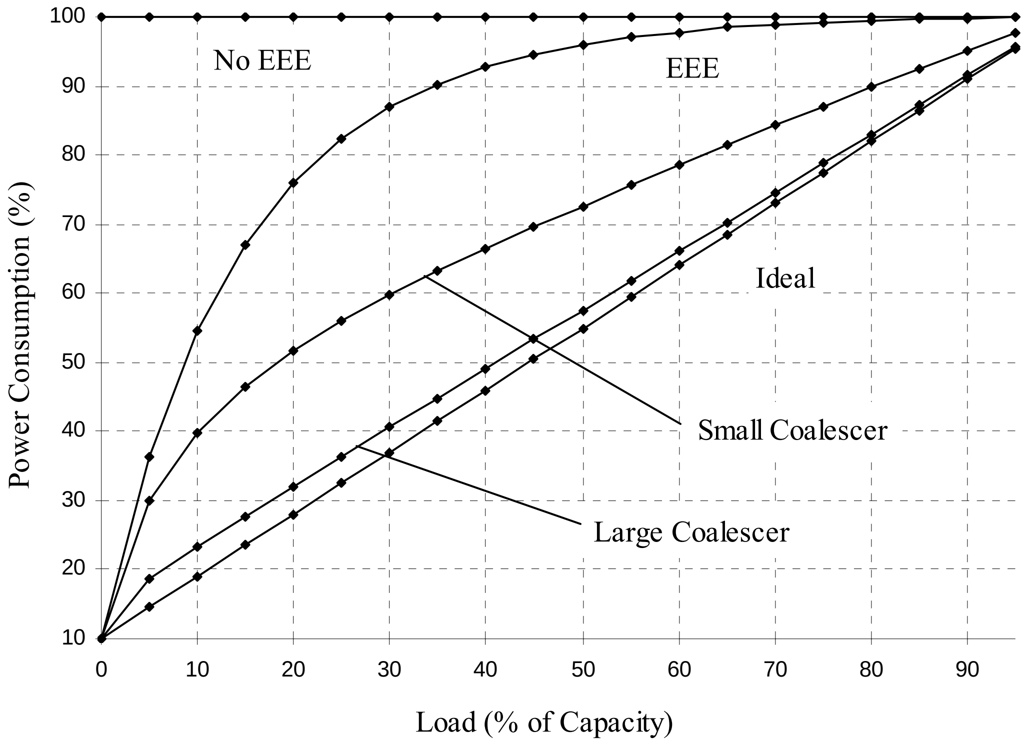

Figure 3.8 – Power Consumption as a Function of Load
for EEE with Packet Coalescing

enough number of packet arrivals to achieve a 95% confidence level on the mean average delay of the packets. The confidence interval within which the mean point falls is smaller than 10% of the mean value in all cases.

3.8 Results of EEE with Packet Coalescing Experiments

Figure 3.8 shows the power consumption of the link as a function of load. The two traces labeled as Small Coalescer and Large Coalescer show the power consumption of the link when using the Small and Large Coalescer. The current power consumption of the links – 100% all the time – the ideal consumption which is proportional to the load, and the power consumption of EEE without coalescing are also shown again in the figure for comparison.

The use of the Small Coalescer improves the energy efficiency almost “half-way” between EEE and ideal (proportional). The Large Coalescer brings the power consumption very close to proportional. For instance, at 15% load the power consumption using the Small Coalescer is about 45% of the full consumption which is over 20% less than what EEE without coalescing yields. The Large Coalescer yields about 27% power consumption which is only about 4% more than the proportional power consumption.

This improvement in energy efficiency has two drawbacks; increase of packet delay, and increase of the relative burstiness of traffic sent by the Ethernet link.

Packet coalescing results in increased per-packet delay since each packet must remain in the coalescer for some time before being transmitted. Figure 3.9 shows the average packet delay for no EEE, EEE, EEE with Small Coalescer, and EEE with Large Coalescer. It can be seen in this figure that EEE adds a small delay to each packet and the larger the coalescer becomes, the more will the packet delay be. For instance, at 15% load the average packet delay of EEE, EEE with Small Coalescer, and EEE with Large Coalescer are about 5 µs, 12 µs and 67 µs respectively.

The delay caused by EEE, which is shown in Figure 3.9, is obviously because of the sleep and wakeup transitions. As can be seen in this figure, the average packet delay caused by EEE is slightly smaller than the sum of T_s and T_w which is 7.36 µs. The small difference is caused by the packets which are transmitted without a transition since they are already queued in the interface when the transmission of the current packet ends.

As seen in Figure 3.9, the packet delay introduced by the Small Coalescer is between 10 µs and 14 µs for any load, which is almost the same as Coalescing Timer ($T_{coalesce}$). Instrumentation of the simulation model shows that the majority of the bursts occur due to expiration of TIMER when the load is low (up to 40% of capacity). As the load increases from 5% to 40%, the number of single packet bursts decreases while the number of multiple-packet-bursts increases which results in relatively lower delay. The delay is the lowest for moderate loads but increases with increasing load. When the load is higher (70% and higher) the opposite case happens; most of the

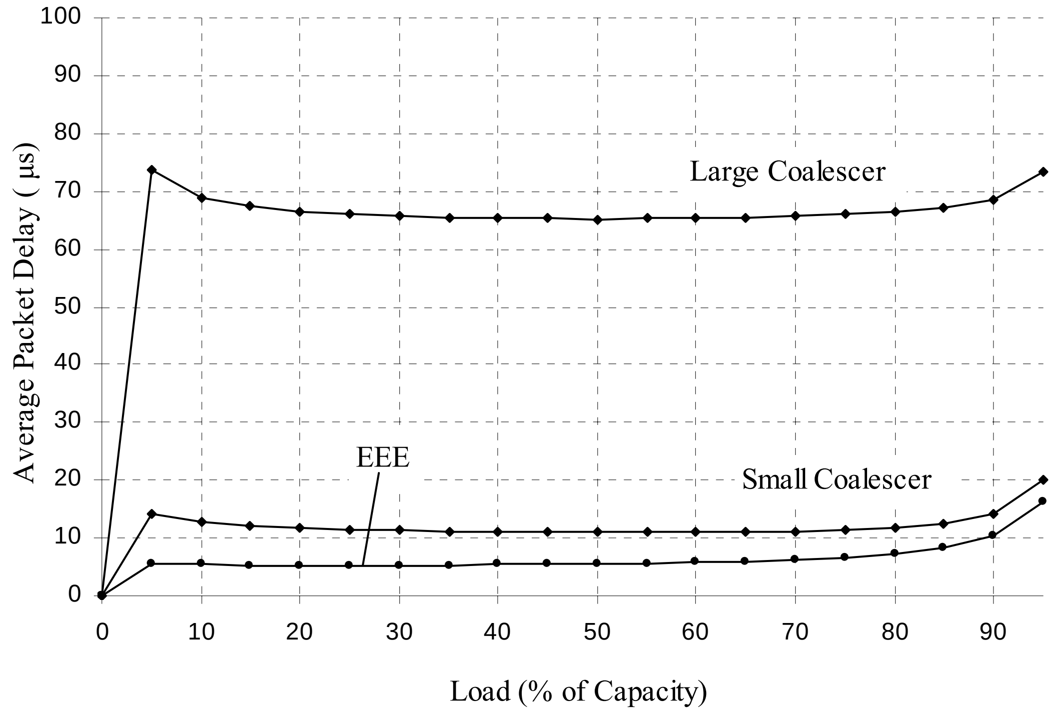

Figure 3.9 – Average Packet Delay as a Function of Load for EEE with Packet Coalescing

bursts are due to the buffer being filled with packets before the timer expires. The delay in this case is caused by 10 packets being transmitted after a 4 μs wakeup time and 1.2 μs multiplied by the number of packets ahead of them in the burst. In the best case, all 10 packets arrive to the burst buffer at time 0 so each wait an average of 4.16 μs + 6.6 μs. In the worst case, 9 packets arrive at time 0 and the 10th arrive at time 12 μs. In this case about 21 μs delay for each packet is possible.

In case of the Large Coalescer which is labeled as "Large Coalescer" in Figure 3.9, instrumentation of the simulation model shows that even for very high loads of 90% and higher, the bursts occur due to timer expiration not the buffer being filled with packets. But the number of packets in each burst is high. Moreover, $T_{coalesce}$ is much higher than T_s and T_w. Since there are a number of packets in each burst, some wait the entire burst timer to be transmitted whereas some are just arrived before the timer expires and wait less. Therefore, in average each packet in a burst is delayed for $T_{coalesce}/2$ (60 µs) plus an additional 5 to 15 µs which is due to the time the packets waiting for the ones ahead of them to be transmitted.

So far, it shown that EEE with either Small or Large Coalescing adds at most a few tens of microseconds of delay to a packet in average. The significance of the added delay will be explained more in Section 3.9. However, besides the average, the distribution of the packet delay is also one of the interesting characteristics of the delay since the distribution of packet delay determines how the delay is distributed among all the packets. To study these points the distribution of the packet delay caused by the Small and Large Coalescer are shown in Figures 3.10 to 3.13 for two samples of light and heavy loads. Note that the reason why the y-axis range in Figure 3.10 only is 50% whereas in the other three figures the y-axis range is 10% is that the value of bin 18 in Figure 3.10 is about 41% while there are no values larger than 10% in all other figures. Therefore, the difference in y-axis range is chosen in order to show the distributions clearer.

Figure 3.10 and 3.11 show the distribution of packet delay caused by the Small and Large Coalescer respectively for a light load (10%). It can be seen in Figure 3.10 that about 41% of the packets are delayed for between 17 to 18 µs. This spike happens because with 10% load, all bursts are due to timer expiration. The average burst size is about 2.5 packets, so the number of bursts is approximately 40% of the number of packets. Obviously, each burst has a first packet which is delayed for 17.68 µs ($T_{coalesce} + T_w + T_{pkt}$). Similarly, in Figure 3.11 the peak is at bin 26 where 9.1% of the packets are delayed for 125 to 130 µs. The same case happens in Figure 3.13 where although the load is high, but so is the coalescing buffer. So the majority of bursts occur due to timeout. The rest of the delays in these figures are uniformly distributed since the arrival of the packets follow a Poisson distribution and they arrive in a fixed interval of $T_{coalesce}$. Under such circumstances, the joint distribution of the arrivals will be uniform over the $T_{coalesce}$ interval [34]. Note that a delay less than 1 µs is impossible since T_{pkt} is 1.2 µs.

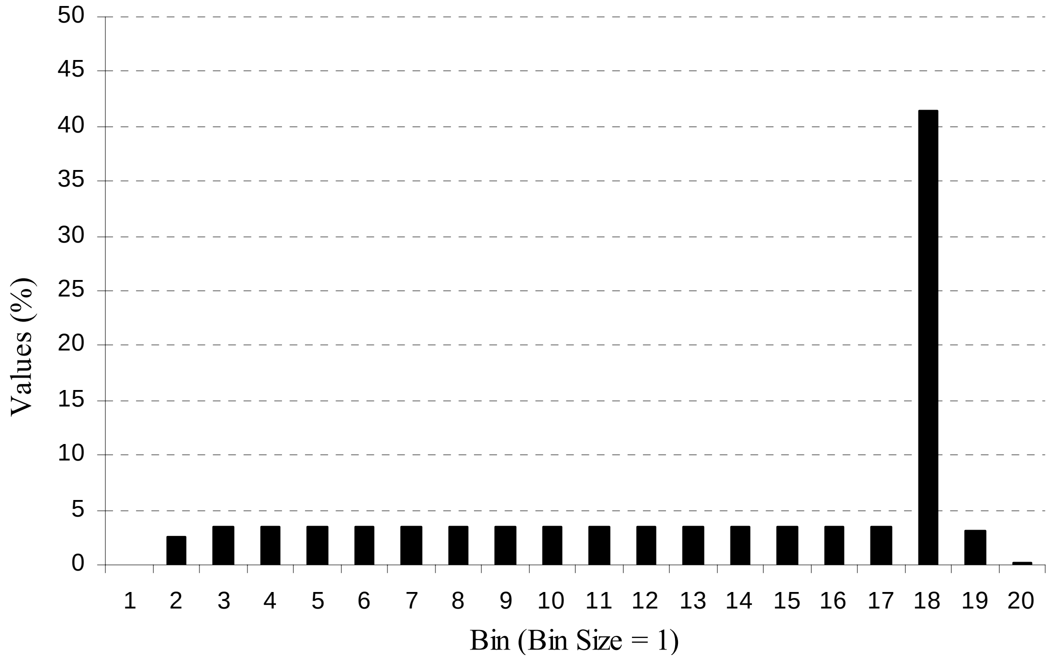

Figure 3.10 – Histogram of Packet Delays (10% Load, Small Coalescer)

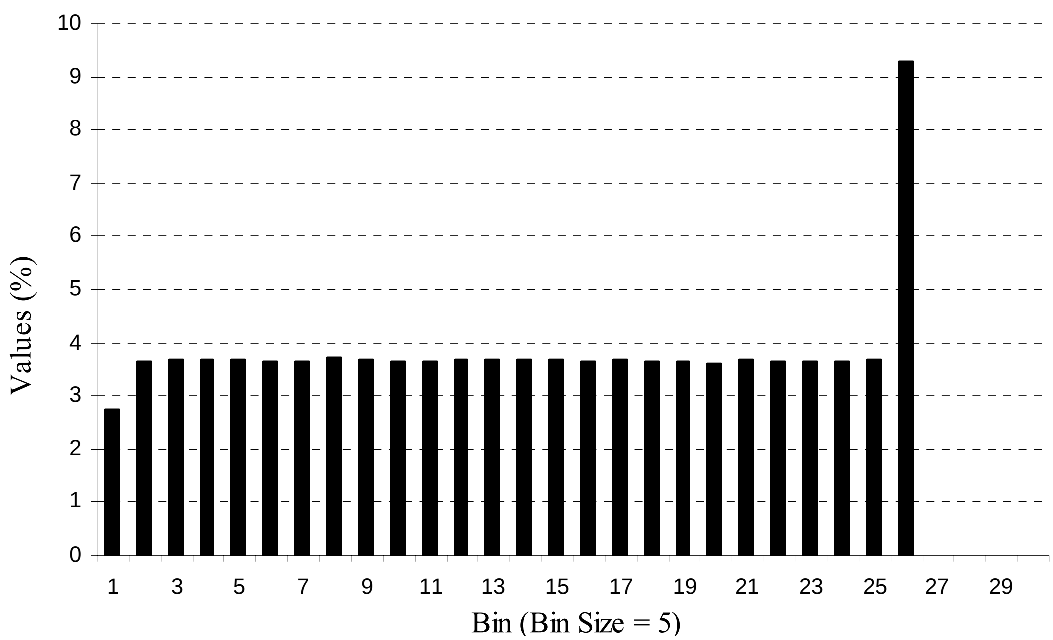

Figure 3.11 – Histogram of Packet Delays (10% Load, Large Coalescer)

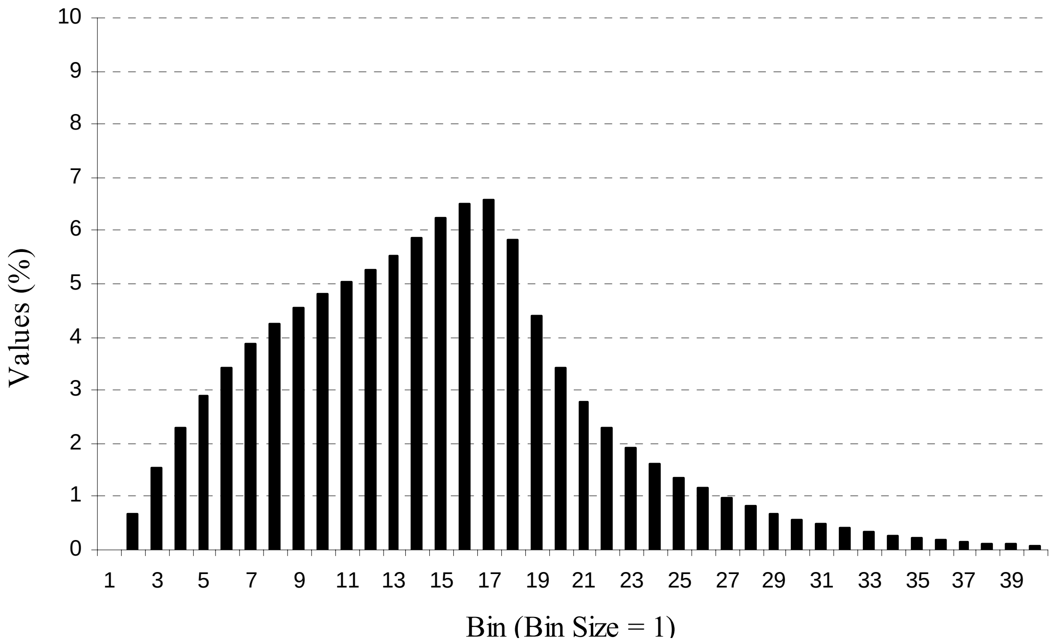

Figure 3.12 – Histogram of Packet Delays (90% Load, Small Coalescer)

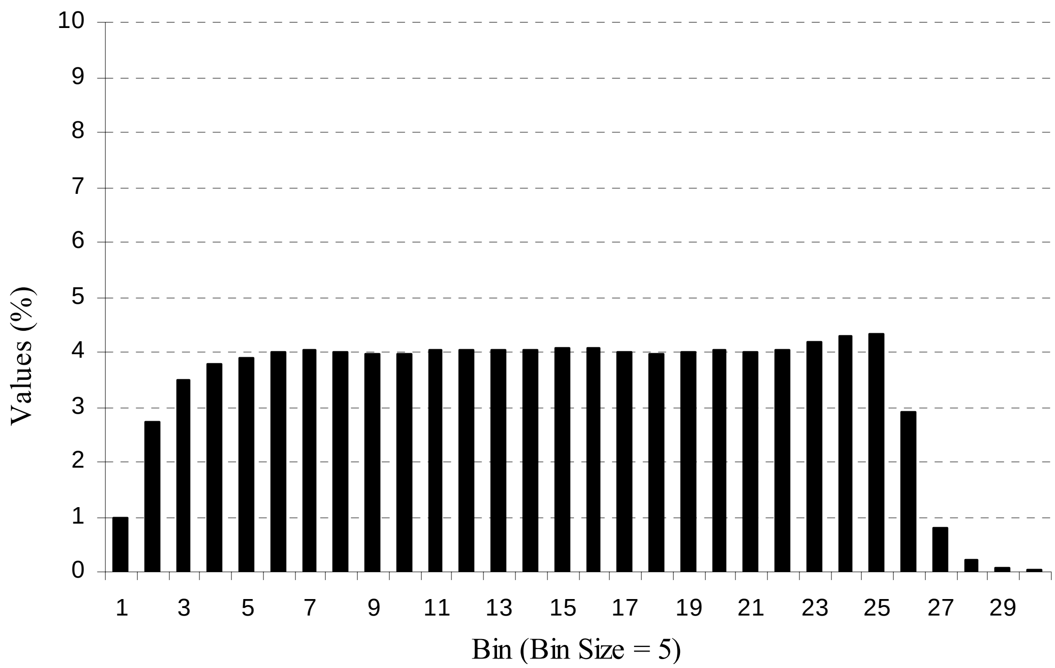

Figure 3.13 – Histogram of Packet Delays (90% Load, Large Coalescer)

Figure 3.12 shows the distribution of packet delay caused by the Small Coalescer for the heavy load of 90%. Since the load in this case is heavy almost all the bursts occur due to the buffer being filled with packets. The inter-arrival time of the packets is exponentially distributed therefore the delays increase until around 16 μs which is the time required to transmit a full buffer plus a T_w to wakeup the link. Delays from this point to the end belong to the packets that arrive while the FSM is in ACTIVE state and the burst is being transmitted. In this state the link works as a non-coalescing link which causes the delays to be distributed as a Poisson process.

The above discussions and figures show that in most cases a high percentage of the packets are for ($T_{coalesce} + T_w + T_{pkt}$) and the rest of the delays are distributed uniformly which lowers the chances of certain packets being delayed for too long while others are transmitted instantaneously.

In addition to the delay and its distribution, there is another issue with Packet Coalescing which is of interest. Packet Coalescing increases the burstiness of the packets departing the link. This is so since the coalescer creates bursts by grouping packets in the buffer before sending them all back-to-back. Burstiness in general, adversely affects the flow of packets in a connection as it results in packet loss due to buffer overflow in the intermediate nodes. Packet loss must either be neglected (as in UDP) which results in less quality of service or must be recovered by other mechanism such as TCP which results in less performance. The increased burstiness caused by coalescing will be studied here by studying the effect of coalescing on the simulated downstream queue showed in Figure 3.7.

One of the response variables of the experiments is the number of packets ahead of each arriving packet to the downstream queue which is an indicator of burstiness. It is obvious that coalescing causes more packets to see more number of packets ahead of them when they arrive to the downstream queue, but the question is how much coalescing and the size of the coalescer add to this number. Figures 3.14, 3.15 and 3.16 show the number of packets each arriving packet sees ahead of itself upon entering the downstream queue for three different loads of 3%, 6% and 9% of

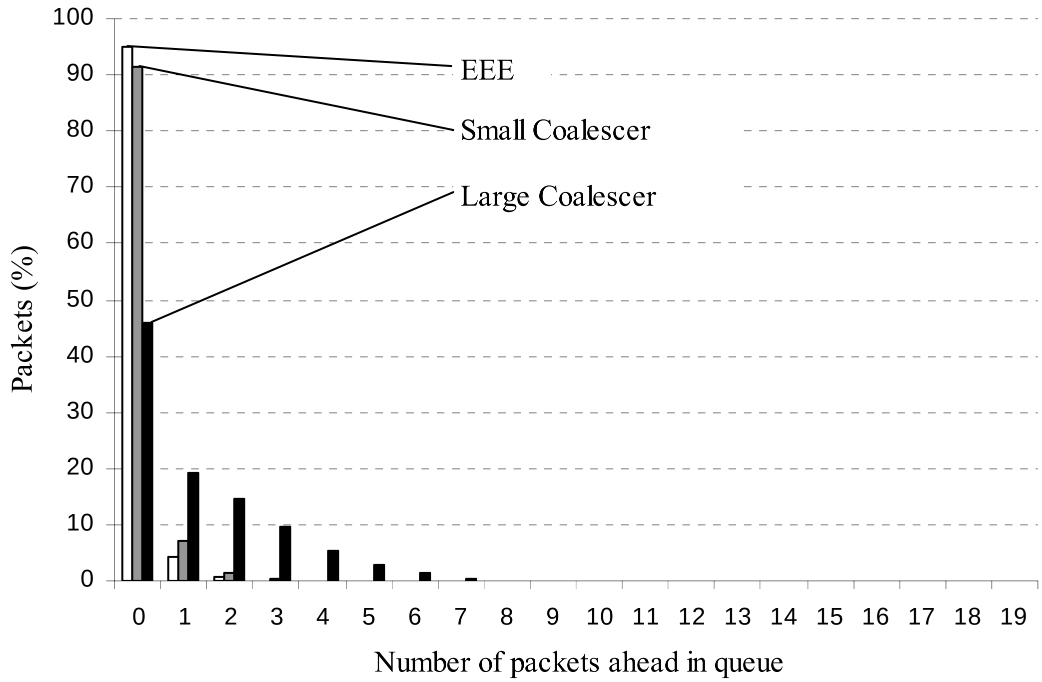

Figure 3.14 – The Number of Packets Ahead in Downstream Queue (3% Load)

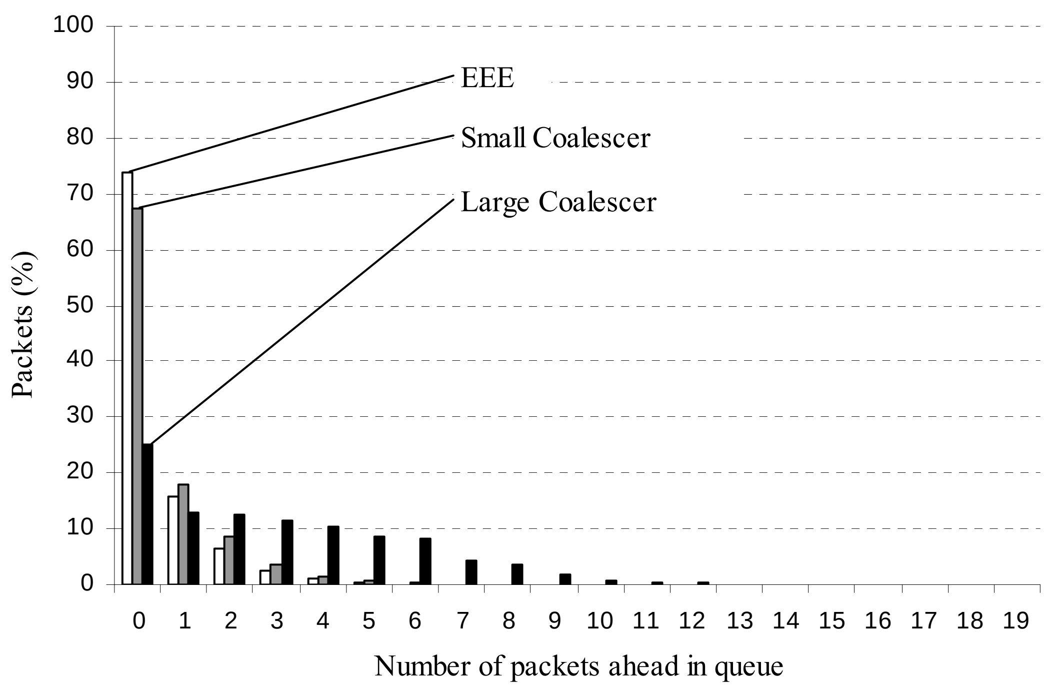

Figure 3.15 – The Number of Packets Ahead in Downstream Queue (6% Load)

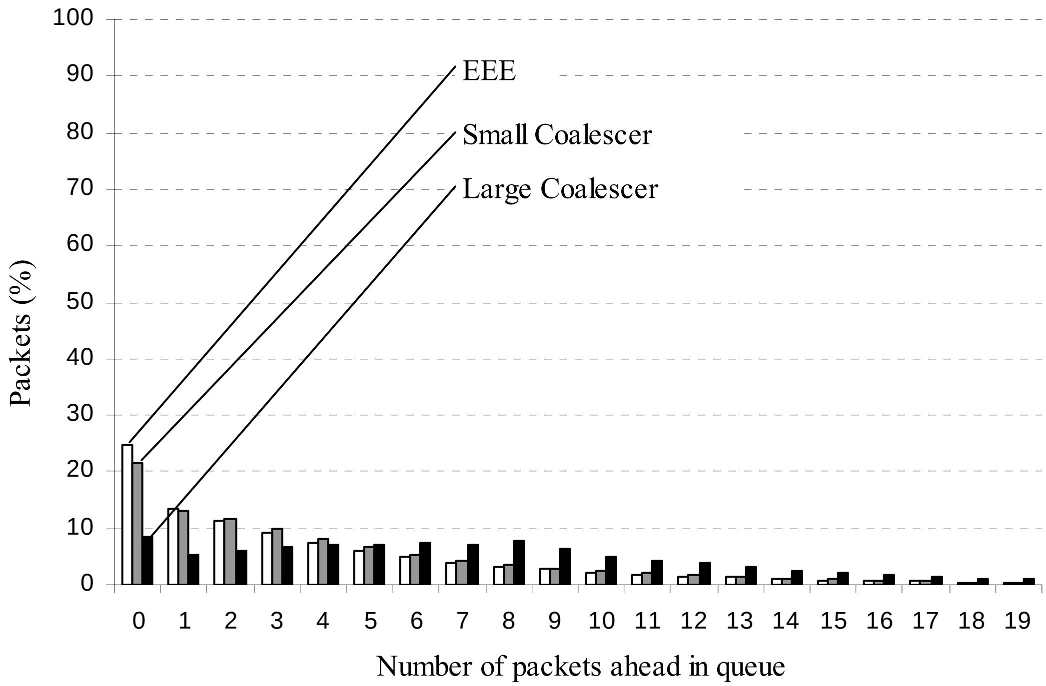

Figure 3.16 – The Number of Packets Ahead in Downstream Queue (9% Load)

the 10 Gb/s link to represent low, moderate and high loads on the downstream link. Note that since the capacity of the downstream link is 10% of the 10 Gb/s, loads more than 10% of capacity mean more than 100% capacity of the downstream link and so make its queue to grow infinitely. In each of the three figures, values for EEE with no coalescing, EEE with Large Coalescer and EEE with Small Coalescer are labeled. For instance, it is shown in Figure 3.14 (3% load) that about 95, 91 and 46 percent of the arriving packets to the downstream queue see no packets ahead of them in the queue when EEE with no coalescing, EEE with small coalescer and EEE with Large Coalescer are in use respectively.

As can be observed in these figures, the Small Coalescer does not add much to the number of packets seen in front of each packet. In fact the increase in any number of packets ahead in the downstream queue is not more than 10%. However, the Large Coalescer increases the number of packets in the downstream queue much more, especially for the light loads (3% for instance).

Therefore, the Large Coalescer shows to add more burstiness to the departing traffic. While it appears that this increase in burstiness does not affect the overall performance of the network significantly, the issues it may cause for TCP congestion control mechanism should be carefully examined. The significance of the added burstiness and its possible effects on TCP will be studied in section 3.9.

3.9 Summary and Conclusions

In this chapter, the overhead of EEE transitions was studied which showed that the power consumption of EEE is much higher than proportional. The reason is that sleep and wakeup transitions of EEE are so long comparing to the actual packet transmission time.

Then Packet Coalescing was studied as a means to decrease this inefficiency. The simulation model of EEE with Packet Coalescing was then used to perform experiments to examine the energy efficiency of EEE with Packet Coalescing and study its drawbacks. The results showed that Packet Coalescing can bring the energy efficiency of EEE very close to proportional.

Nevertheless, Packet Coalescing comes with two drawbacks. First, it increases the per-packet-delay. The results from simulation experiments showed that the average delay added to each packet is of the order of tens of microseconds at most. The significance of a few microseconds per packets delay is very likely to be low when compared to tens of milliseconds end-to-end delay of Internet connections. However, even this much increase in packet delay in a data center is considered significant, but the additional energy savings gained in a data server may be able to justify a reasonable delay per packet.

The increase in burstiness of the packets being transmitted from the link was also studied using a downstream queue simulation. The increase in burstiness requires the routers and NICs to have larger buffers to prevent loss. But this is not the main issue of burstiness since the future routers, switches and NICs will be equipped with tens of megabytes of queueing buffer which makes them tolerate much greater degrees of queueing. The more serious concern with burstiness will be its effects on upper layer protocols such as TCP. For instance, coalescing may increase the Round Trip Time in some scenarios and may also cause the slow start phase of TCP's congestion control mechanism to end sooner. These issues are studied to some extent in [9] using an ns-2 model and it is concluded that the effects of coalescing on TCP will be small provided that: 1) the coalescing buffer size is much smaller than the router and NIC buffers and the TCP window, and 2) the coalescing timer is much smaller than the RTT.

Chapter 4: Periodically Paused Switched Ethernet

In this chapter, additional explanation of PPSE method is presented using a fluid flow model followed by its technical details such as the mechanism of halting the transmission of the links, shutting down the links and waking them up again. The simulation model of PPSE is described next. Experiments on Simple PPSE are then designed using this simulation model of which results are presented and explained next. The improved version of PPSE, called Adaptive PPSE is presented next. The performance of Adaptive PPSE is then evaluated. Finally the results and discussions of PPSE are summarized and concluded.

4.1 PPSE Fluid Flow Model

The functionality of PPSE can be explained by means of a fluid flow model of the queue forming at the outgoing buffers of the links connected to the switch. Figure 4.1 depicts this model. The bits can be considered as the fluid flowing into and out of the buffer. The bits that arrive during T_{off} fill the buffer up to L_{max} level at which a T_{on} period starts and bits flow out of the buffer. Assuming that the bits arrive to the buffer and depart it at the constant rates of λ and μ respectively, L_{max} can be determined using Equation 4.1:

$$L_{\max} = \lambda \times T_{off} \quad \text{(Eq. 4.1)}$$

The time to empty the buffer completely is:

$$T_{empty} = \frac{L_{\max}}{\mu - \lambda} \quad \text{(Eq. 4.2)}$$

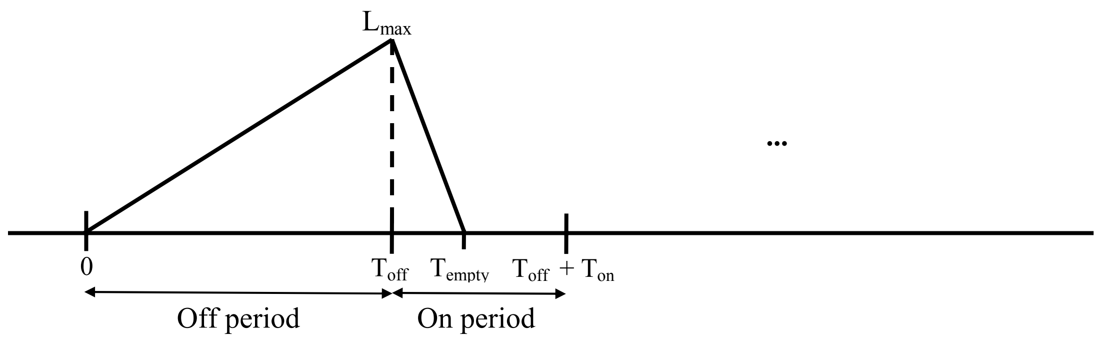

Figure 4.1 – Fluid Flow Model of Simple PPSE

The stability condition of the queue is $T_{on} > T_{empty}$. The mean length of the queue which is the area of the two triangles in Figure 4.1 divided by the total time of a cycle can be determined by Equation 4.3:

$$L_{mean} = L_{max} \times \frac{T_{off} + T_{empty}}{2(T_{off} + T_{on})} \qquad \text{(Eq. 4.3)}$$

4.2 PPSE System Implementation

In this section two technical aspects of PPSE implementation will be explained. These technical aspects are how the links are shut down and being woken up again, and the hardware components that can be powered-off during OFF states.

4.2.1 Link Shutdown and Wakeup

Assuming that upon receiving a Pause Notification the link stops sending any traffic for the designated T_{off} time, the link must enter a low-power mode in order to save energy. The question is how this low-power mode works and what the mechanism of putting the link in this low-power mode is.

EEE has already answered the above question. When a link is EEE-enabled, it enters Quiet mode whenever there it has no packets to send. Now assume that a Pause Notification is sent to a link. The link knows that it must not send any packets for a known period which directly translates to having nothing to send for the mentioned period. Therefore, the link enters the Quiet mode after finishing the Sleep transition.

The wakeup process happens likewise. When the pausing time period ends the NIC is able to start sending the packets in its buffer (provided that there are packets in the buffer) which means that it now has some packets to send. The link enters Wakeup transition and consequently becomes active to send these packets.

4.2.2 Energy Saving Opportunities and Mode Transitions

In order to assess the opportunities that PPSE provides to save energy in a switch, two factors must be considered; the components that can be put to sleep and the time and power they require to transition between sleep and active modes.

The major components of a modular switch can be categorized into four main parts; chassis, switching fabric and ports. The components of the chassis include cooling equipment, power supply, etc. The switching fabric performs tasks such as switching traffic between two fabric ports.

Using PPSE, all the ports can be put to sleep when the switch is in OFF state since it is known that the switch will not be needed to forward any packets for the next T_{off} time. Moreover, due to the same reason, the switching fabric can be put to sleep when in OFF state and the chassis can be powered-down. The higher number of components that can be put to sleep is one of the advantages of PPSE over the other approaches that reduce the power consumption of the switch such as the work done in [1] and [18]. Synchronization of the idle periods on all ports using Pause Notifications is what enables PPSE to put more number of components to sleep.

The transition times vary greatly from each component to another. Moreover, sleeping states for many of the components are not yet supported by the manufacturers. This variety and lack of support in some cases has made it common for the transition times and consumptions to be takes from the wireless community and assuming that they would be similar to a great extent for Ethernet switches too. Based on the estimates in [20] it appears to be reasonable to assume transition times of between 1 ms to 10 ms and transition power consumptions of 0.1 W to 2 W. [1] also assumes similar values. To avoid unnecessary complicities the transition times and powers are completely ignored in the performance evaluation sections of this chapter. The power consumption is assumed to be equal to the time spent in OFF state over the total time of simulation as if the transitions occurred instantaneously and thus consumed no power. Then in Chapter 5 where energy savings of PPSE are estimated, an overall percentage is added to the energy consumption to compensate for neglecting transition times and power.

4.3 Simulation Model of Simple PPSE

In order to further evaluate the performance of Simple PPSE and its improvement in the subsequent sections, a simulation model is developed and validated using CSIM simulation library [50]. The model simulates the system view depicted in Figure 2.6. In this model, simulated packets arrive to a single server queue (the server simulates the switch) which works according to the FSM of Figure 2.7. Note that instead of simulating the traffic on multiple links, the aggregated traffic of all the links are simulated as one link connected to the server.

4.4 Experiments on Simple PPSE

The simulation model explained in section 4.3 is used in this section to design experiments in order to evaluate the performance of Simple PPSE. In these experiments, the response variables of interest are:

- Power consumption of the switch; which is equal to Duty Cycle for Simple PPSE. It is assumed that transitions between ON and OFF states occur instantaneously.
- Average packet delay

The factors in the experiments are:

- T_{off}; is set to 100 ms which represents human reaction time. Moreover, it is reported in [6] that using this value for T_{off} along with a proper Duty Cycle barely affects user experience.
- Duty Cycle: Two duty cycles of 10% and 50% are considered. The reason for using the former is the light utilization of the links which is typically between 3 to 7% [41]. Therefore, higher Duty Cycles will unnecessarily keep the switch on and waste energy. The latter Duty Cycle is set to match what presented in the initial evaluation of PPSE in [6]. T_{on} is computed using Equation 4.2 and substituting T_{off} and D with the values stated above.
- Load; a percentage of the link's capacity. This factor is varied between 0 and 30%. More than 30% offered load is very unlikely and not of interest.
- Distribution of packet arrivals and packet size; set to Poisson distribution with fixed packet size of 1500 B.

T_{off} and Duty Cycle factors are dependent to each other and independent of all the other factors. However, Load and Distribution of packets as well as the size of the packets are dependant factors. The distribution and size of the packets are the two fixed factors in the experiments performed in this section. However, Load and Duty Cycle factors are varied for the levels mentioned to determine the effects of them on the response variables. In other words, a Fractional Factorial Experiment [30] is designed by varying two factors for certain factor levels and fixing the other factors to determine the effects on the response variables. Two experiments are done with the 10% and 50% Duty Cycles of which results will be presented and explained in the next section. The experiments are done on large enough number of packet arrivals to achieve a 95% confidence level on the mean average delay of the packets. The confidence interval within which the mean point falls is smaller than 10% of the mean value in all cases.

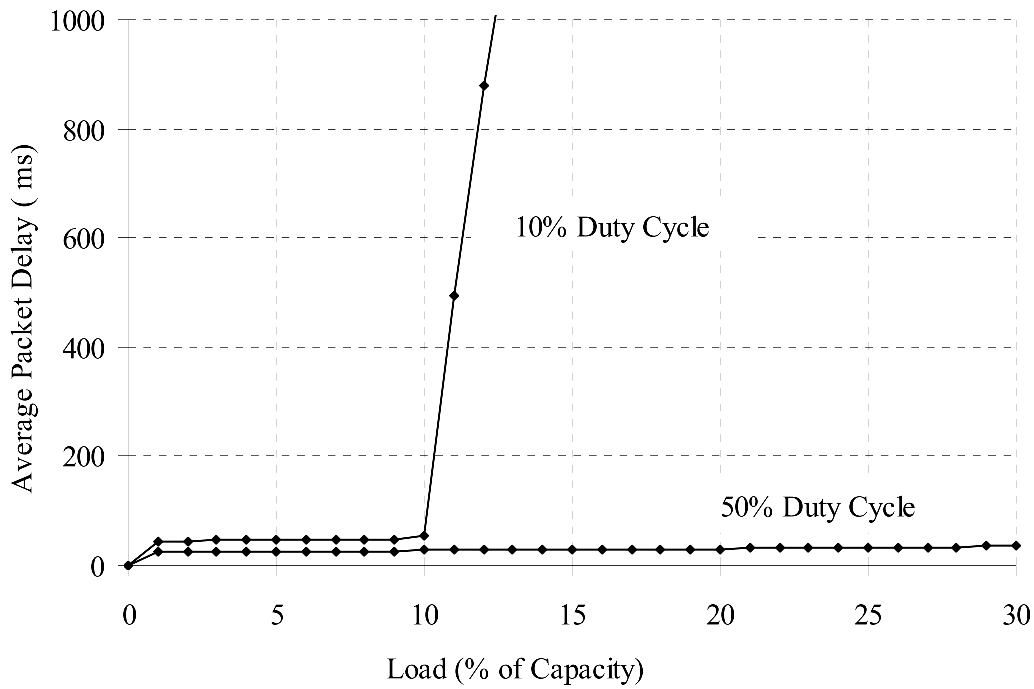

Figure 4.2 – Average Packet Delay as a Function of Load – Simple PPSE

4.5 Results of Simple PPSE Experiments

Since the power consumption of the link in case of Simple PPSE is assumed to be the same as the Duty Cycle, the power consumption of the switch is 10% and 50% for duty cycles of 10% and 50% respectively regardless of the other factors. This is one of the valuable characteristics of Simple PPSE; the power that the switch consumes can be determined solely by setting the Duty Cycle. However, although the duty cycle can be set arbitrarily to achieve any level of energy savings, the per packet delay resulting from setting a low duty cycle will adversely affect the performance of the switch.

Figure 4.2 shows the average packet delay as a function of load. Note that the packets are delayed in the NICs connected to the switch. The reason is that the switch is halting the transmission from the NICs during T_{off} periods while the packets are still being generated by the applications at the devices connected to the switch and are buffered at the NICs. The two labeled traces show the average packet delay when using PPSE with 10% and 50% duty cycles. The average packet delays are composed of the delays for the packets buffered in OFF state which are transmitted immediately after the switch turns on plus the packets that arrive after all the buffered packets are transmitted. The joint distribution of the arrival of Poisson distributed packets to fixed period of time is uniform over the fixed period [34]. Therefore, the joint distribution of the packets arrive to the T_{off} period is uniform over T_{off}. So it can be imagined that in average they are each transmitted with a $T_{off}/2$ delay. The packets arrive while in ON state and after all the buffered packets from OFF state are transmitted get no delay (neglecting the 1.2 μs transmission delay). Therefore, for example in the case of 50% duty cycle and 10% load, half of the packets get 50 ms delay while the other half get no delay which makes the average about 25 ms. The 2 ms difference comes from the increased delay caused by a small number of packets that arrived in ON period to wait for the buffered packets to be transmitted first. The ratio of the number of packets arriving during ON state to the number arriving in OFF period does not change with load, so the load does not affect the average of these packets. The reason for the slight increase in packet delay with increasing load is that the number of packets arriving in T_{off} depends on the load. Consequently, the number of packets that arrive when the packets buffered from OFF state are being serviced increases. This leads to more delay for these packets, which increases the average delay slightly.

Moreover, it can be observed in Figure 4.2 that when the load is more than 10% for 10% duty cycle, the average delay starts to increase very fast due to instability of the queue. In other words, not all the packets buffered in OFF state are transmitted in one ON period which makes the average delay increase gradually. The same phenomenon happens for any duty cycle as well as if there is a sudden burst in the traffic although the overall load is less than the duty cycle.

To overcome this shortcoming, the duty cycle can be increased, but so will the power consumption. Bearing in mind that the typical utilization of a link is low, increasing the utilization will result in unnecessary power consumption of the switch under low utilizations. A better approach to this problem would be to "predict" the occasional high utilizations or bursts so the switch stays on under these circumstances only and not under normal low utilizations. In the next section, a threshold is introduced to PPSE in order to make it more adaptive to the utilization without losing most of the possible energy savings.

4.6 Adaptive PPSE

The use of Simple PPSE results in large increases in per packet delay of the traffic especially when the aggregate load on the link is heavy and/or when a sudden burst of packets flow to the switch. To reduce this effect, a modification of Simple PPSE is proposed which regulates the transition to OFF state based on the number of packets received while in ON state. Simple PPSE with this modification is called Adaptive PPSE.

In Adaptive PPSE, a counter is used to count the sum of the number of packets received from the links while in each ON state. If this number is below a certain threshold, the transition to OFF state occurs. Otherwise, the TON timer is reset and the switch remains in ON state for another T_{on}. In other words, Adaptive PPSE makes a guess on the future traffic based on the recent traffic. If the recent traffic is high, it is likely to remain high for the next T_{on} period. Otherwise, energy can be saved by entering OFF state. This guess does not always turn out to be correct. As an example, consider the TON period when the last few packets of a large burst is being transmitted by a link to the switch. The number of packets exceeds the threshold and the switch remains in ON state for another T_{on} period while the burst is over and no more packets are left to be transmitted. This fault in guessing the amount of future traffic will result in a decrease in energy savings gained by Adaptive PPSE compared to Simple PPSE which will be studied in detail later in this chapter.

The functionality of Adaptive PPSE can be shown by the FSM depicted in Figure 4.3. The FSM has the same states as the FSM for Simple PPSE depicted in Figure 2.7. A new array and a new variable are defined as following:

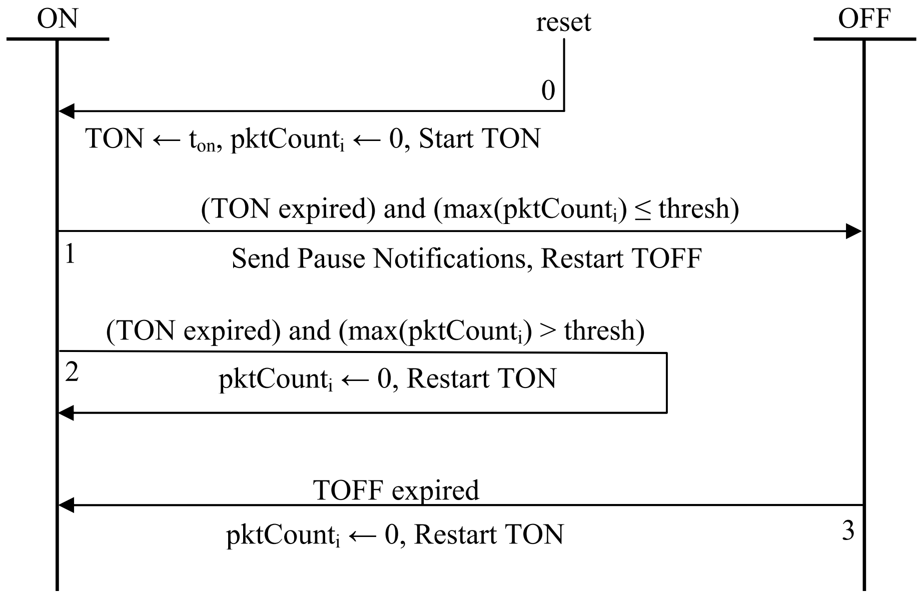

Figure 4.3 – FSM for Adaptive PPSE

- pktCount: An array of the size equal to the number of links connected to the switch. Each index of pktCount stores the number of packets received from the corresponding link.
- Threshold: The threshold which is compared to the maximum of all $pktCount_i$'s to determine if transition to OFF state should be made.

A detailed explanation for each of the FSM's transitions follows:

- Transition 0: As for Simple PPSE, when Adaptive PPSE starts or resets, the switch enters the ON state, TON is set to its initial value T_{on} and starts to count down. Also, all cells of pktCount are set to 0.
- Transition 1: When TON expires and the maximum over all cells of pktCount is greater than or equal to threshold, Pause Notification are sent on all the links connected to the switch, TOFF is reset to its initial value T_{off}, and starts to count down and the switch enters the OFF state.

- Transition 2: If the maximum over all cells of pktCount is less than threshold when TON expires, TON is reset to its initial value T_{on} and starts to count down, all cells of pktCount are set to 0 and the switch remains in ON state.
- Transition 3: Upon expiration of TOFF, TON is reset to its initial value T_{on} and starts to count down, all cells of pktCount are set to 0, the switch returns to ON state and the entire procedure is repeated.

In the simulation model of PPSE in this work, the aggregated traffic on all the links connected to the switch is simulated as one link. Therefore, pktCount can be considered as a variable instead of an array and the actual value of pktCount is used instead of its maximum over all pktCount_i's.

4.7 Experiments on Adaptive PPSE

The simulation model presented in section 4.6 is used in this section to design experiments in order to evaluate the performance of Adaptive PPSE. In these experiments, the response variables of interest are:

- Power consumption of the switch; which is assumed to be equal to the Actual Duty Cycle of the switch. The Actual Duty Cycle (D_A) is computed by Equation 4.4. It is assumed that transition between ON and OFF states occur instantaneously.

$$D_A = \frac{A_{on}}{A_{on} + A_{off}} \qquad \text{(Eq. 4.4)}$$

  where A_{on} and A_{off} are the overall actual times spent in ON and OFF states respectively. Remember that some T_{off} periods can be missed due to exceeding the threshold. Therefore, D_A is always equal or greater than the Duty Cycle.

- Average packet delay

The factors in the experiments are:

- T_{off}; is set to 100 ms which represents human reaction time.

- Duty Cycle; 10% Duty Cycle is chosen. The reason for using this duty cycle is the light utilization of the links. Therefore more than 10% Duty Cycle seems to be unnecessary. The delay caused by occasional higher link utilizations or packet bursts are predicted and handled using the threshold. T_{on} is computed using Equation 4.2 and substituting T_{off} and D with the values stated above. So T_{on} for T_{off} =100 ms and D=10% is approximately 11 ms.
- Threshold; varied between 0 and 12000 packets in the first experiment. Then two thresholds of 1000 and 5000 packets are considered for the rest of the experiments. The reason to choose the former is that is that 1000 packets in 11 ms translates to almost 10% of the links capacity (equal to the Duty Cycle) if fixed size packets of 1500 B are considered. The latter is almost half of the link's capacity which represents a very high load and PPSE can only be considered effective if handles occasional high loads well.
- Load; a percentage of the link's capacity. This factor is varied between 0 and 30%. More than 30% offered load is very unlikely and not of interest.
- Distribution of packet arrivals and packet sizes; two distributions are used. First, Poisson arrivals with fixed packet size of 1500 B and second, synthetic bursty traffic with Bounded Pareto packet sizes and exponential inter-arrival times. The characteristics of synthetic traffic are explained below.

It is well-known that the traffic in real Internet links is very bursty ([35] and [43]). In [15], a synthetic traffic generator is developed and shown to produce very similar traffic traces to the real Ethernet traffic. The sizes of the bursts generated by this traffic generator follow Bounded Pareto distribution and the inter-burst times between packets follow exponential distribution. This traffic generator is borrowed here to produce traces of bursty traffic in order to evaluate the performance of PPSE with traffic which is similar to the real links. The list of tunable parameters of the traffic

Table 4.1 – Parameters of the Synthetic Packet Generator

Parameter	Value	Represents
Minimum Packet Length (B)	64	Minimum Ethernet packet length
Maximum Packet Length (B)	1500	Maximum Transmission Unit in Ethernet v2
Minimum Burst Size (KB)	10	Sudden transfer of a small picture file
Maximum Burst Size (MB)	100	Sudden transfer of a small video clip
Pareto Alpha value	1.5	bursty traffic with small bursts [15]
Link Utilization (%)	0 to 30%	Load factor of the experiment

generator and the values to which each of them is in the experiments set are presented in Table 4.1.

T_{off} and Duty Cycle factors are dependent to each other and independent of all the other factors and are chosen to be fixed in these experiments based on the results obtained from experiments on Simple PPSE. Two factor levels for threshold which is completely independent of all other factors are chosen. Load and Distribution of packets as well as the size of the packets are dependant factors.

An experiment is done with Poisson traffic to determine the dependency of the behavior of PPSE to the threshold. To do so, a Fractional Factorial Experiment [30] is designed by varying the threshold from 0 to very high (12000 packets) for three levels of load (10%, 20% and 30%) and plotting the power consumption. It seems trivial that Adaptive PPSE with a low enough threshold is as if PPSE is not used at all, and with a very high threshold is equivalent to Simple PPSE. However, the pattern of change in the in-between cases can lead to a better understanding of the behavior of PPSE and also support the reason behind choosing High and Low Thresholds in the rest of the experiments.

The above experiment is followed by two sets of experiments which are done by varying Distribution and size of the packets; the first set with fixed-size, Poisson distributed traffic and the second set with variable-size, bursty traffic which is produced by a synthetic traffic generator. Therefore, a Fractional Factorial Experiment [30] is designed by varying three factors for certain factor levels and fixing the other factors to determine the effects on the response variables. The experiments are done on large enough number of packet arrivals to achieve a 95% confidence level on the mean average delay of the packets. The confidence interval within which the mean point falls is smaller than 10% of the mean value in all cases.

4.8 Results of Adaptive PPSE Experiments

The results obtained from Adaptive PPSE experiments will be presented and described in this section. This section is divided to two subsections; the results from Poisson traffic experiments and the results from bursty traffic experiments.

4.8.1 Poisson Traffic

Figure 4.4 shows the power consumption or the Actual Duty Cycle of the switch as a function of threshold. The low thresholds cause PPSE to keep the switch in ON state 100% of the time. It is at 1000 packets when the Actual Duty Cycle starts to decrease and eventually get stable at between 3000 and 8000 when increase in the number of packets arriving during a T_{on} period does not change the number of times that threshold is exceeded. For very high thresholds, Adaptive PPSE will have the functionality of Simple PPSE where the power consumption is the same as Duty Cycle which makes excessive per-packet delays inevitable.

Figure 4.5 shows the power consumption or the Actual Duty Cycle of the switch as a function of load. It can be seen in this figure that using the High Threshold, the power consumption of the switch is close to (less than 10% different than) the ideal consumption which is proportional to the link utilization. However, the power consumption is more when the Low Threshold is used.

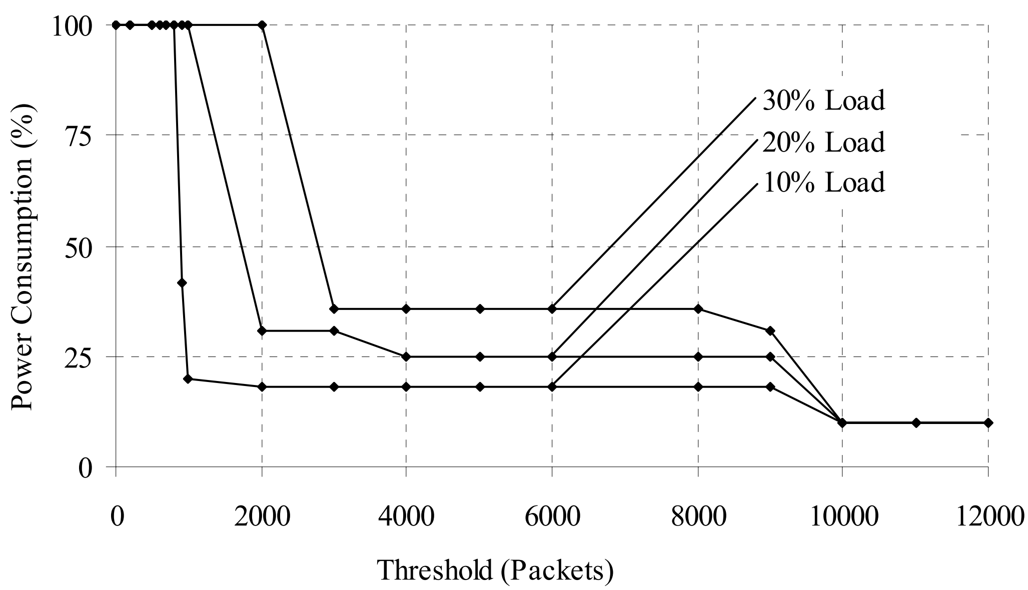

Figure 4.4 – Power Consumption as a Function of Threshold –
Adaptive PPSE

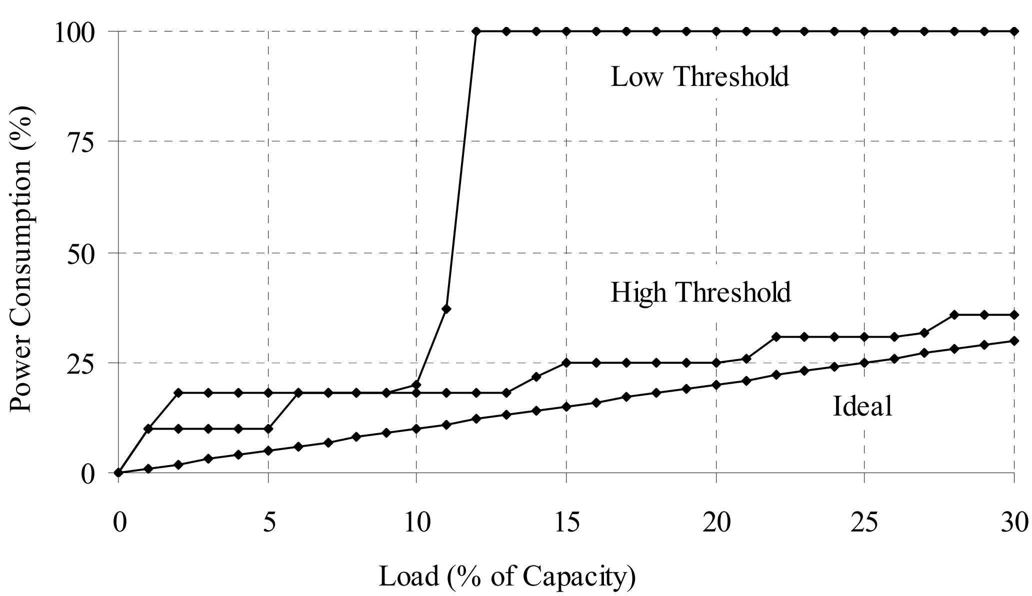

Figure 4.5 – Power Consumption as a Function of Load –
Poisson Traffic

For example, when the utilization is 11%, the power consumption is about 18% and 37% for High and Low Thresholds respectively.

Using the Low Threshold, when the load exceeds the duty cycle (10%) the power consumption ascends quickly to 100% and stays at this level as the load increases. Instrumentation of the simulation model shows that this sudden increase in power consumption is due to the number of arriving packets during all T_{on} periods exceeding thresh. The steps seen in both traces are also because of the number of packets during T_{on} period exceeding thresh. At the points that power consumption increases to the next step, the number of packets that arrive to some of the T_{on} periods exceed thresh. Then for the next few loads, while the number of packets increase, it is not to the extent that causes any more number of packets during T_{on} periods to exceed the threshold. So the power consumption stays the same for these “in-between” utilizations until the next step in which the increase in the number of packets cause more number of packets arriving during T_{on} period to exceed the threshold.

Although using PPSE brings the power consumption close to proportional, it obviously increases the per packet delay since the switch is sometimes in OFF state and unable to service packets while the packets keep arriving to the devices attached to the links connected to the switch. Therefore, packets arriving during the time in which the switch is in OFF state are queued and delayed. Figure 4.6 shows the average packet delay caused by Adaptive PPSE. The average packet delay when using High and Low Thresholds are labeled in this figure. Both the thresholds cause an average packet delay of between 40 ms and 50 ms. For instance, when the load is 5% the average packet delay using High and Low Thresholds are around 47 and 43 ms respectively.

As can be seen in Figure 4.6, the traces have “steps” as the load increases. Each step corresponds to an power consumption step in Figure 4.5. The reason is the same as what described for the power consumption. As the load increases, more packets arrive to a T_{on} period. At some point this number exceeds the threshold which causes PPSE to remain in ON state for another T_{on} period and service packets. This is the point where a new step begins. An increase in

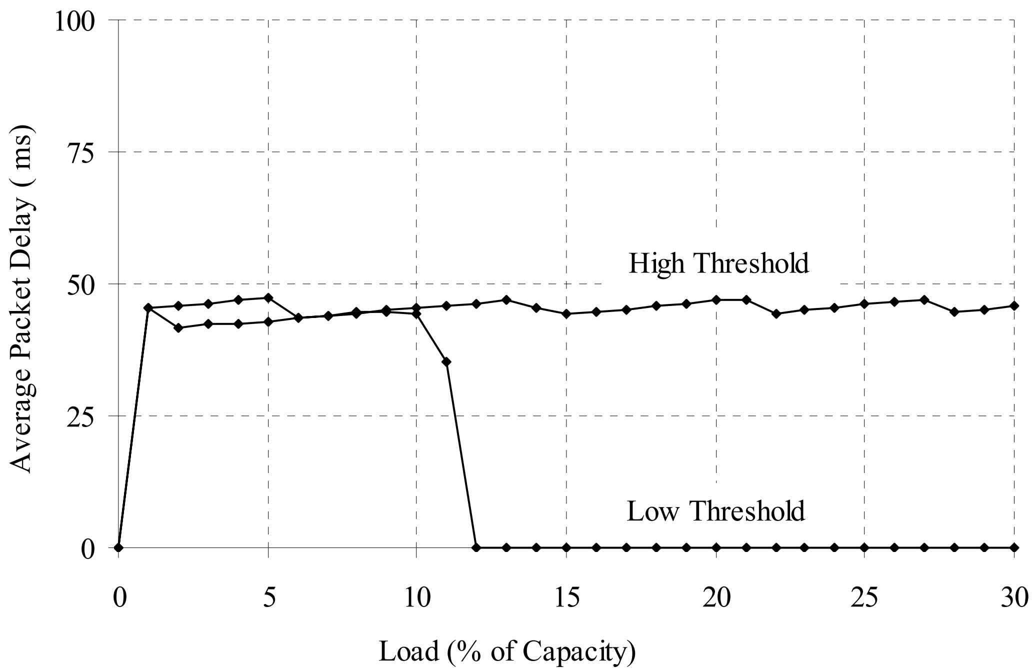

Figure 4.6 – Average Packet Delay as a Function of Load –
Poisson Traffic

power consumption and a drop in packet delay are seen at this point due to more actual T_{on} time from the switch.

The sudden drop in delay at 10% load when using the Low Threshold happens when number of packets arriving to all T_{on} periods exceeding the threshold which causes the switch to stay constantly in ON state and service the packets immediately. Thus a zero delay (neglecting transmission delay of 1.2 µs) is seen soon after the load exceeds the Duty Cycle of 10%.

So far, it shown that Adaptive PPSE with either High or Low Threshold adds at most a few milliseconds of delay to a packet in average. The significance of the added delay will be explained further in Section 4.9. But besides the average, the distribution of the packet delay is also one of the interesting characteristics of the delay since the distribution of packet delay determines how the delay is distributed among all the packets. To study these points the distribution of the packet delay caused by Adaptive PPSE with High and Low Thresholds are

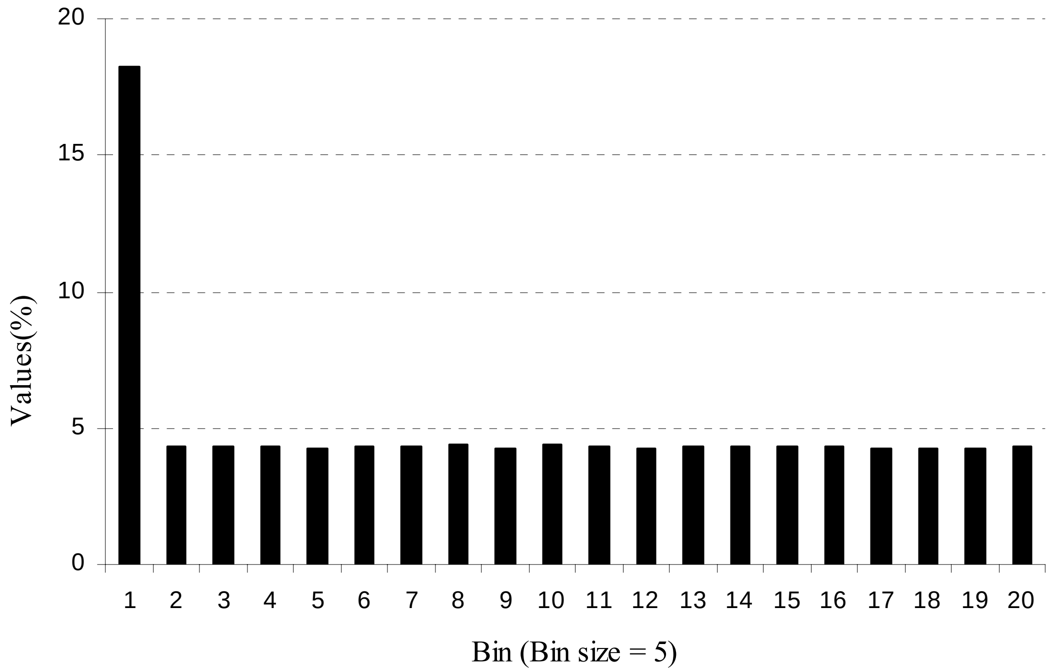

Figure 4.7 – Histogram of Packet Delays – Poisson Traffic
(5% Load, Low Threshold)

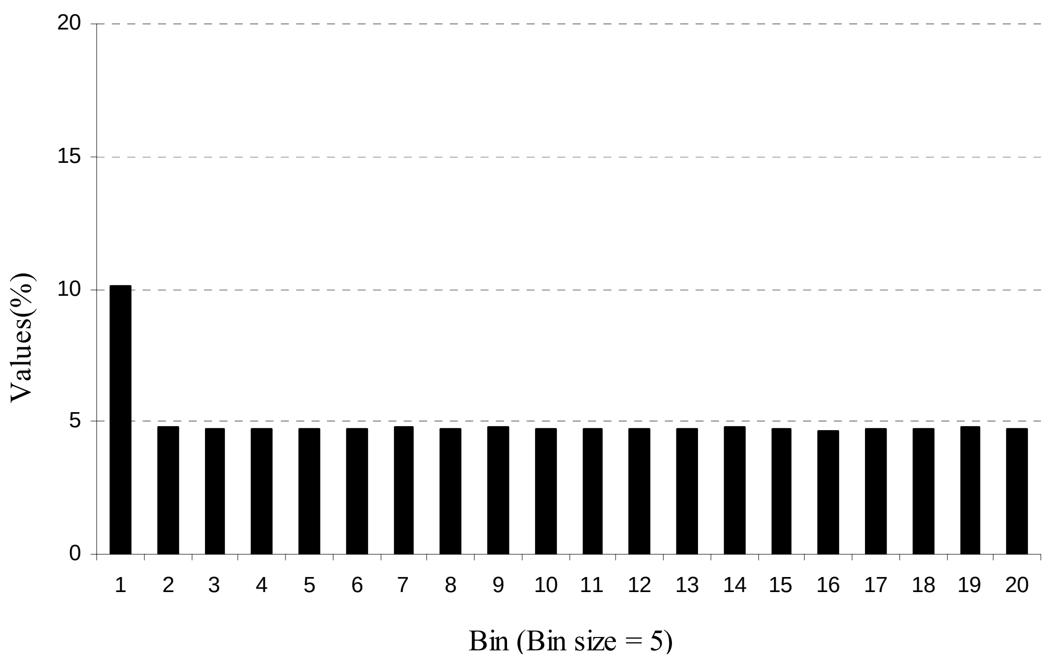

Figure 4.8 – Histogram of Packet Delays – Poisson Traffic
(5% Load, High Threshold)

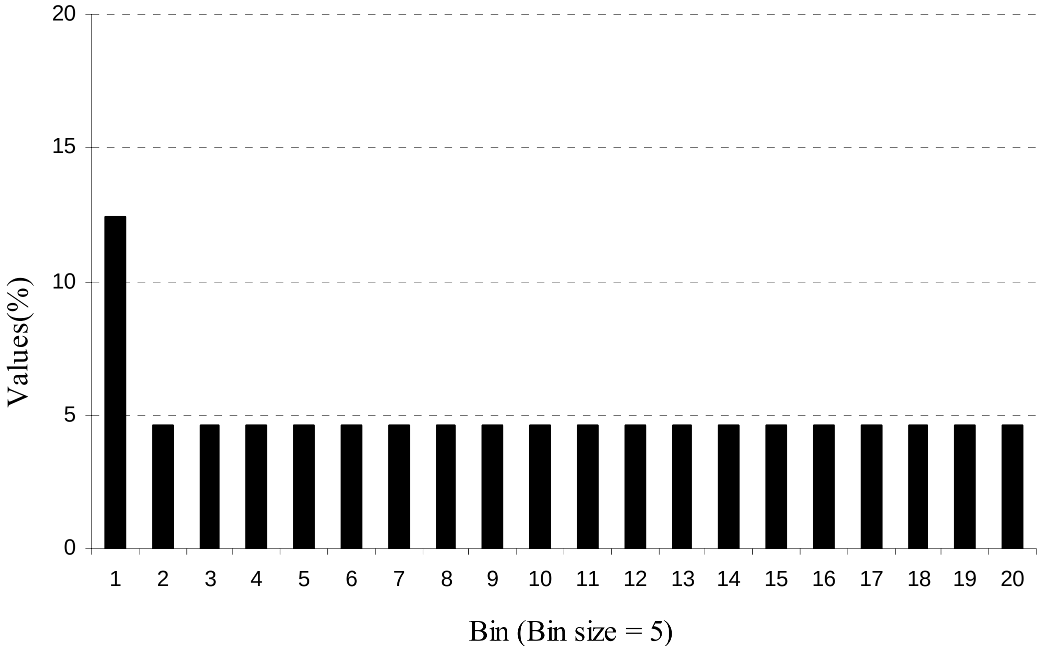

Figure 4.9 – Histogram of Packet Delays – Poisson Traffic
(25% Load, High Threshold)

shown in Figures 4.7 to 4.9 for two samples of light loads and one sample of a heavy load. Note that as said earlier in this subsection, the delay for high loads when using the Low Threshold is almost 0 (neglecting the 1.2 μs transmission delay of each packet).

Figure 4.7 and 4.8 show the distribution of packet delay caused by Adaptive PPSE with Low and High Threshold respectively when the load is light (5%). It can be seen that in both cases the percentage of packets that belong to the first bin (the spike) are about the same as the Actual Duty Cycle for the corresponding load and threshold. The number of packets arrive during each T_{on} is almost the same as the Actual Duty Cycle. The ones among these packets that arrive after all packets queued from the previous T_{off} period are serviced, are delayed for 1.2 μs which fall into the first bin. The rest of the packets which arrive during a T_{on} but are not serviced immediately are delayed for a few milliseconds. These packets still belong to the first bin since the load is low and their delay is not more than 5 ms. Therefore in both these figures the number of packets that get the least delay and belong to the first bin is almost the same as the Actual Duty Cycle. The rest of the delays are uniformly distributed since arrival of these packets follow a Poisson distribution and they arrive in a fixed interval of T_{off}. Under such circumstances, the joint distribution of the arrivals will be uniform over the T_{off} interval [34].

As seen in Figure 4.9, the number of packets in the first bin (about 12%) is lower than the Actual Duty Cycle (about 30%). In this case still the percentage of packets arriving during a T_{on} period is about the same as the Actual Duty Cycle. However, the packets arriving to T_{on} periods wait much more due to the high load and consequently a high number of queued packets is serviced prior to them. This is why that relative to low loads, a less percentage of the packets are delayed low enough to belong to the first bin. The uniformity of the rest of the delays is due to the same reason noted for low loads.

The above discussions and figures show that in most cases a high percentage of the packets (about the same as Actual Duty Cycle) are delayed for less than 5 ms and the rest of the delays are distributed uniformly which lowers the chances of certain packets being delayed for too long while others are transmitted instantaneously. This conclusion will be explained more in section 4.9.

4.8.2 Bursty Traffic

Figure 4.10 shows the power consumption or the Actual Duty Cycle of the switch as a function of load when the traffic on the link to the switch is bursty. Similar to power consumption for Poisson traffic, the power consumption of the switch is very close to (less than 10% higher than) the ideal consumption when the High Threshold is in use. Also, the power consumption is more when the Low Threshold is used. For example, when the utilization is 10%, the power consumption is about 18% and 23% for high and Low Thresholds respectively.

As expected, the power consumption pattern of the switch with PPSE when synthetic bursty traffic (bursty with bursts Bounded Pareto distributed and inter-burst times Poisson distributed) is

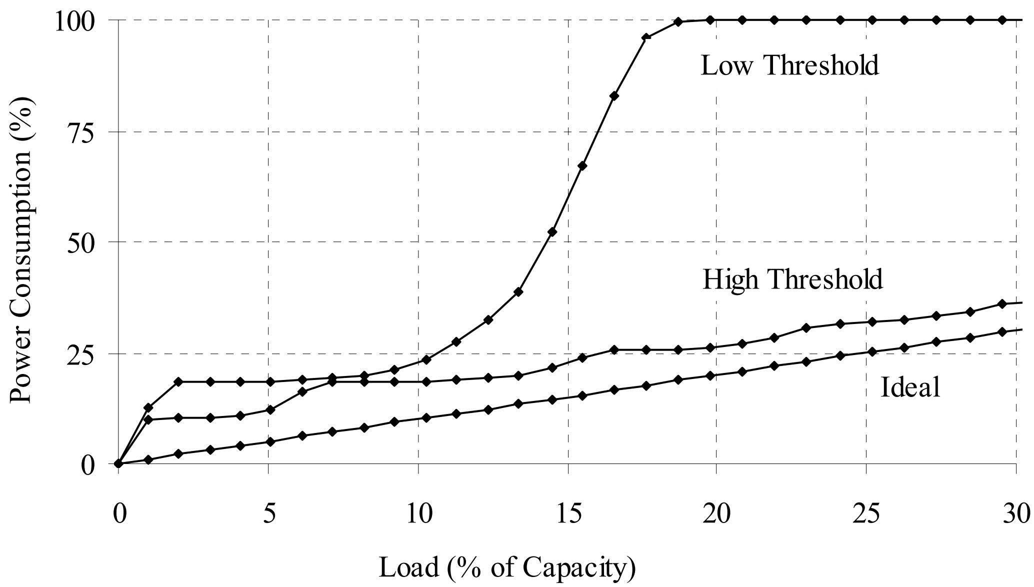

Figure 4.10 – Power Consumption as a Function of Load –
Bursty Traffic

on the link shows to share some characteristics with when the traffic is smooth (with Poisson inter-arrival times). The “steps” seen before in Figure 4.5 can be seen here too, but the transition between steps occurs less suddenly. In other words, while the power consumptions in Figure 4.5 in the “in-between” utilizations in each step are exactly the same until the sudden increase at the end of the step which takes the power consumption to the next step, a gradual increase is seen in the in-between utilizations in Figure 4.10. These slight increases are due to characteristics of the synthetic traffic traces. While the average size of the bursts as well as the number of them does not change with 1% more utilization, the bursts come slightly closer together to increase the overall utilization. In other words, the inter-burst time is reduced in order to slightly increase the utilization. This increased proximity of bursts makes the packets transmitted during some T_{on} periods to exceed the threshold and consequently the switch to stay on for another T_{on} period which increases the power consumption slightly. An example illuminates this phenomenon more. Let us consider the slight (less than 1%) increase in the power consumption between 5% and 6%

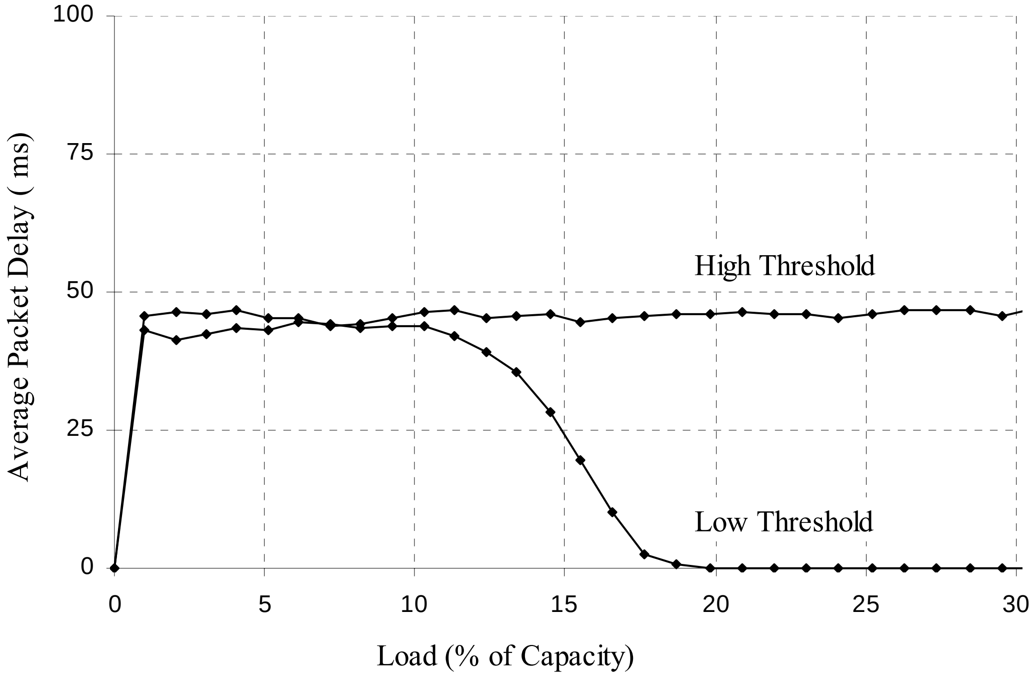

Figure 4.11 – Average Packet Delay as a Function of Load –
Bursty Traffic

utilizations. Instrumentation of the traffic generator shows that the average burst size and the number of bursts are the same for both these utilizations (237455 and 30416 B respectively). However, the average inter-burst time decreases slightly from 5% to 6% utilization (462 μs to 381 μs) which makes the number of packets during T_{on} periods to exceed the threshold from 268 to 598 times when the High Threshold is effective and from 987 to 1091 times when the High Threshold is effective and consequently increases the power consumption for just a fraction of a percent.

Figure 4.11 shows the average packet delay caused by Adaptive PPSE when the traffic on the link is bursty. Similar to Poisson traffic, the per-packet delay is around 45 ms for all the utilizations when the High Threshold is used and slightly lower for low utilizations when the Low Threshold is in use. For instance, when the load is about 10%, the average packet delay using High and Low Thresholds are around 46 and 43 ms respectively. Using the Low Threshold, the delay starts to decrease quickly from 10% load (Same as the Duty Cycle) until it is reduced to just a few microseconds of transmission delay at around 18% utilization.

The pattern which both the traces of High and Low Thresholds follow is similar to what was seen for Poisson traffic in Figure 4.6. The difference is that with bursty traffic the transitions between steps are smoother. In other words, slight changes (less than 1 ms) are made when the utilization changes 1% as opposed to the case of Poisson traffic in Figure 4.6 where the delay for in-between utilization levels stays the same until a sudden transition of 5 to 10 ms takes the delay level to the next step. This smooth pattern is due to the same reason described for smooth changes of power consumption levels earlier. With each 1% increase in utilization the bursts come slightly closer together in average. More proximity among the bursts results in a few more number of packets arriving during a T_{on} period to exceed the threshold which consequently keeps the switch in ON state for another period. Therefore the switch services more packets and the delay decreases slightly. On the other hand the increase in the overall number of packets makes the average delay a few microseconds higher compared to the previous utilization. In the in-between utilization in each step the increase in delay caused by more packets exceed the decrease caused by more staying in T_{on} period which results in slight increase in average per-packet delay. The delay increases until the reverse happens; the decrease caused by staying in more T_{on} periods exceeds the increase caused by more packets which results in gradual decrease in average per-packet delay that consequently forms the transition to the next step.

The distribution of the delay will be studied here too to determine how the delay is distributed among the packets. To study this point, the distribution of the packet delay caused by Adaptive PPSE with High and Low Thresholds are shown in Figures 4.12 to 4.14 for one sample of light loads (5%) and one sample of heavy loads (25%). Note that as shown in Figure 4.11 in this subsection, the delay for high loads when using the Low Threshold is almost 0 (neglecting the 1.2 μs transmission delay and a few microseconds of queueing delay caused by occasional

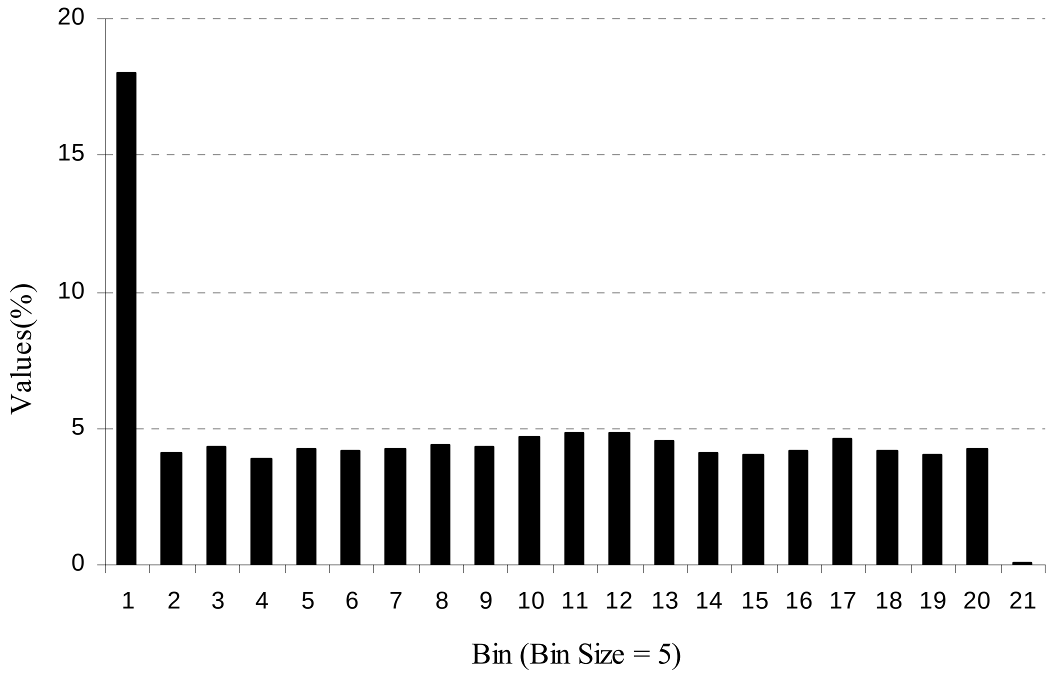

Figure 4.12 – Histogram of Packet Delays – Bursty Traffic
(5% Load, Low Threshold)

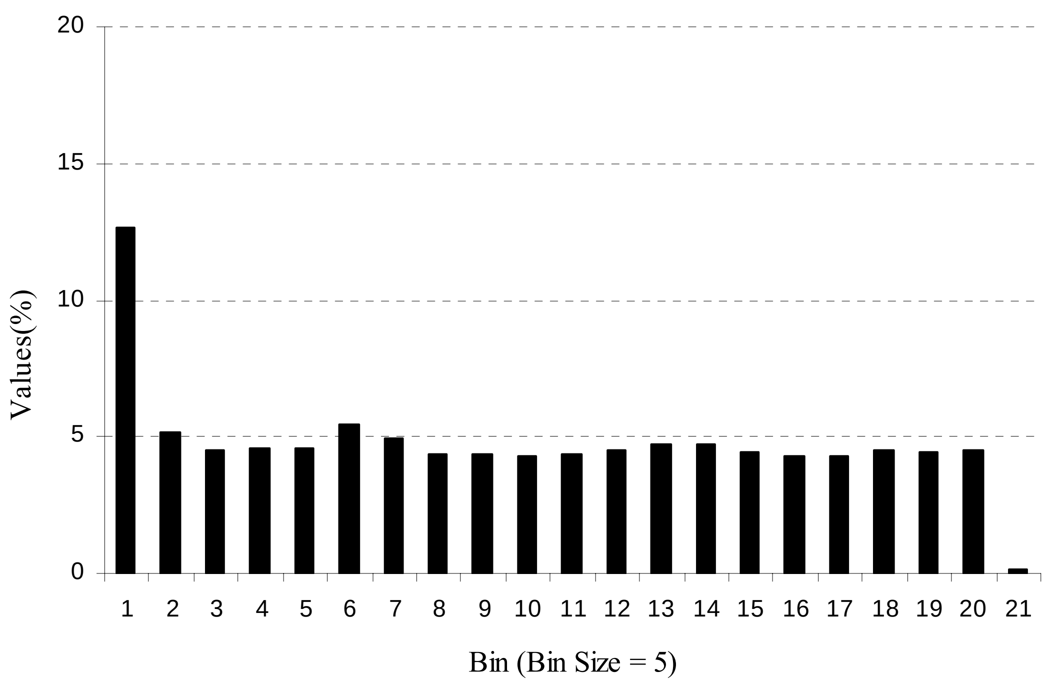

Figure 4.13 – Histogram of Packet Delays – Bursty Traffic
(5% Load, High Threshold)

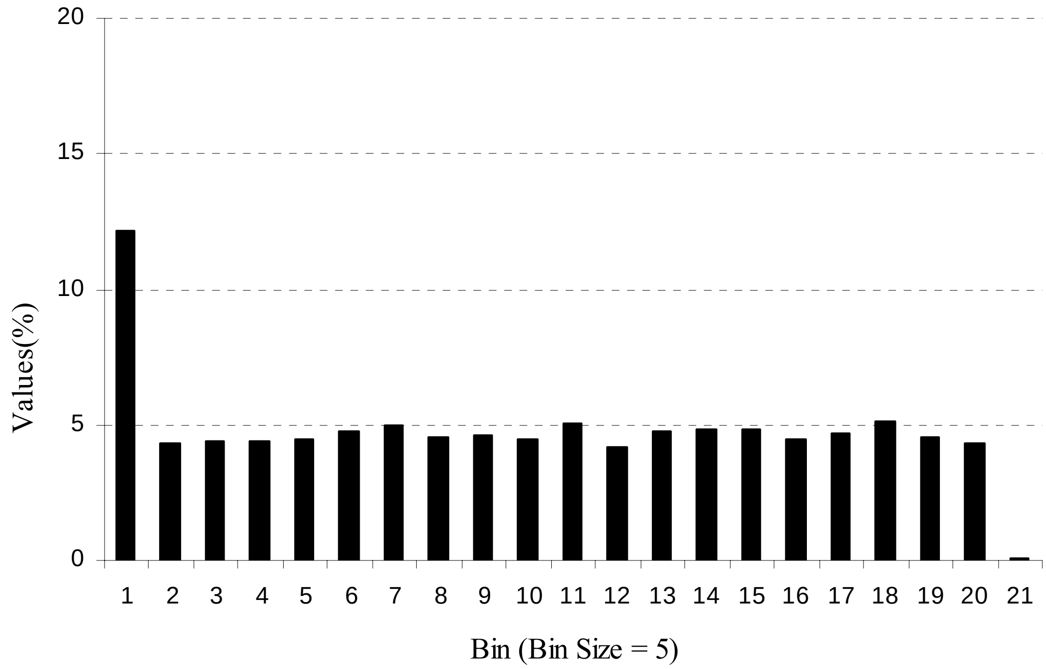

Figure 4.14 – Histogram of Packet Delays – Bursty Traffic
(25% Load, High Threshold)

large bursts) since the switch is always on under such traffic, and no queue is formed at the devices connected to it..

Figure 4.12 and 4.13 show the distribution of packet delay caused by the Adaptive PPSE with Low Threshold and High Threshold respectively when the load is light (5%) and Figure 4.14 shows the distribution of packet delays for High Threshold for a sample of heavy loads (25%). The patterns seen in delay distributions of Poisson traffic are repeated here with some differences. The main reason for the similarity between these patterns is that although in bursty traffic the size of the bursts follows a Bounded Pareto distribution, the inter-burst times are still Poisson distributed. So the joint distribution of the individual bursts arriving to a fixed T_{off} period is uniform [34]. Therefore, it is seen that except for the first bin, the delays are almost uniformly distributed. However, the variety of the number of packet delays belonging to each bin is more comparing to Figure 4.7 to 4.9 where the traffic was smooth. This slight variety is caused by the size of the bursts which are not Poisson. So, depending on the number of bursts generated during each T_{off} period, the delay can be varied to some extent. In other words, presence of bursts disturbs the uniformity of the distribution of delay. However, it is observed that since the bursts are generally small, the variety is a few percents and does not exceed 5% for the sample loads that are depicted in these figures.

The level of spikes in Figures 4.12 and 4.14 are almost the same as the spikes seen in the corresponding figures for Poisson traffic. Similar to Poisson traffic, for low loads, the number of delays belonging to the first bin is almost the same as the Actual Duty Cycle. It is true that not all the packets arriving in a T_{on} period are serviced immediately; but even in this situation, the delay caused by waiting for the packets ahead is not enough to take their delay to the next bin.

The spike in the packet distribution for the sample of high load with High Threshold (Figure 4.13) is about 13% while the actual duty cycle for this load and threshold is about 31%. While in this case also the percentage of packets arriving during a T_{on} period is roughly the same as the Actual Duty Cycle, the packets arriving to T_{on} periods wait much more due to the high load. Consequently a high number of queued packets are serviced prior to them which results in a less percentage of the packets that are delayed low enough to belong to the first bin.

The study of the distribution of packet delays when the traffic is bursty showed that the delay is uniformly distributed among the packets except the very low delays of less than 5 ms which affects 2 to 3 times more packets than other delays. So, packets get less than 5 ms delay with more probability than other delays and the probability of getting other (higher) delays is almost the same for all delays.

4.9 Summary and Conclusions

In this chapter, a fluid flow model of Simple PPSE [6] was described to demonstrate the functionality of PPSE. Next, the technical feasibility of implementing PPSE on a switch is described. It is explained that notifying all the devices connected to the links to stop transmitting traffic is possible by using a defined flow control packet called PAUSE packet or defining new types of packets. Additionally, the components of the switch that can potentially be put to sleep are considered. It was noted that switching ports had been studied before as components that are possible to sleep. PPSE also enables the switch to the switching fabric to sleep too, an opportunity which has not been there before due to possibility of incoming traffic on active ports. This problem does not occur when using PPSE since it is made sure than no traffic will be on the link for a known fixed period of time. Moreover, the mechanism to put the links to sleep is described. In fact, the mechanism is nothing but EEE which was studied earlier in Chapter 3. When the links are stopped from transmitting traffic they have nothing to send which is the triggering condition for EEE.

The performance of Simple PPSE was evaluated next. It was seen that this version either imposes a high level of per-packet delay if a too low Duty Cycle is chosen, or does not save much energy if the Duty Cycle is chosen too high. Therefore a modification to PPSE was introduced which uses a threshold to determine if it is proper to stop the link and power down of another OFF period or not. This version which "adapts" to the load on the link is called Adaptive PPSE. Adaptive PPSE was first explained in this chapter using an FSM and then was evaluated.

In order to evaluate Adaptive PPSE, a simulation model consisting of a link connected to a switch was used. Using this model, Adaptive PPSE was evaluated with Poisson and bursty traffic traces. The bursty traffic was generated using a synthetic traffic generator developed in [15] which has shown to generate traffic traces similar to the traffic on real Internet links. The parameters of the traffic generator were set in a way that represented small burst on the link. Evaluation results showed that by setting proper values for Duty Cycle and the threshold power consumption close to proportional are obtainable while the average delay stays less than half of the human response time (100 ms). This level of delay does not have much negative effect most of the time [44] but it is where issues of performance trade-offs start to rise which needs in-depth study and analysis. Also, the distribution of delays showed that the delay is distributed fairly and smoothly among all the packets.

Based on the experiments presented in this chapter and their results, it is concluded that Adaptive PPSE can be considered as a significant method to reduce the energy consumption of Ethernet switches without imposing excessive added delays. Nevertheless, the impact of Adaptive PPSE on network protocols such as TCP still needs thorough investigation and study which is future work.

Chapter 5: Economic and Environmental Benefits

In this chapter the economic and environmental benefits of both EEE and PPSE are studied. The benefits are projected first in the form of the absolute amount of energy that could potentially be saved using each of these methods as well as the combination of them. This amount of energy is then converted to monetary value using the current conventional energy costs which will be called Economic Benefits. Moreover, producing such energy causes certain volume of CO_2 emissions which impacts the environment by contributing to global warming. The reduction of CO_2 by this volume using EEE and PPSE will be called Environment Benefits.

5.1 Projected Benefits of EEE

EEE standard is scheduled to become final in September 2010 [9]. Some hardware manufacturers such as ASUS have already introduced EEE to some of their products [3] and it is expected that other manufacturers also incorporate EEE in their future products [9]. Initially, the savings gained from adopting EEE in devices will be low since most of them will be connected to legacy devices which are not able to exploit the efficiency that EEE offers. The connection between a non-EEE and an EEE device means that the EEE device must deactivate its EEE capability to stay compatible with the legacy device. The savings will increase as soon as more devices adopt EEE until it becomes the standard for the majority of devices. It is then that true benefits of EEE will appear in networks.

Since the outreach of EEE is difficult to determine now, the estimates that appear in this section use the current number of 10 Gb/s links although most of them are not adopted EEE yet. In other words, the estimations assume that all the current 10 Gb/s links have turned into EEE links and calculates the benefits gained. Note that the estimations here make very similar assumptions and techniques in [9].

The number of currently active 10 Gb/s Ethernet links is approximately 65 million in the U.S. [9]. These links mostly consist of data center links and uplinks from rack-based switches. So it can be assumed that each of these links is utilized slightly more than an edge link of around 3% in average. It is assumed that such link consumes 5.5 W of power and being constantly powered on throughout the year. At 3% utilization, about 72% savings can be gained using EEE which translates to approximately 2.25 TWh of energy per year. Additionally, an 18% more saving is gained using Packet Coalescing (Large Coalescer) which is about 0.41 TWh of energy per year. The reduction in total energy consumption caused by adopting EEE and adding Packet Coalescing to EEE on the links saves about $225 million and $41 million a year respectively in the U.S. at the current average electricity cost of about $0.10 per kWh [4]. Totally, EEE with Packet Coalescing will save about 266 TWh of energy per year which costs about $266 million and is equivalent to about 3.4 billion lbs of CO_2 emissions per year. The estimations above are summarized in Table 5.1. Also given in Table 5.1 is the ideal savings for reference.

The estimations and calculations presented above are not absolutely accurate since many other factors contribute to energy savings resulted by adopting EEE. The number of links compatible, the characteristics of traffic on the links and the time that each individual link is powered on throughout the year are among other factors that change the estimations. Moreover, the cost of energy and amount of CO_2 emissions to which it translates depend highly on the source that is converted to electricity, the efficiency of conversion, proximity to the power plant, etc. Nevertheless, the estimations presented above will still give a picture of the potential impact that EEE can have on the economy and the environment.

Table 5.1 – Economic and Environmental Benefits of EEE

	EEE	Packet Coalescing	Total	Ideal
Assumptions				
Consumption per Link (W)	5.5	5.5	–	5.5
Link Utilization (%)	5%	5%	–	5%
Active Links (millions)	65	65	–	65
Electricity Cost ($/kWh)	0.1	0.1	–	0.1
CO2 Emissions (lbs/kWh)	1.5	1.5	–	1.5
Benefits				
Savings per Link (%)	72%	18%	–	88%
Savings per Link (W)	3.96	0.71	4.67	4.84
Total (MW)	257.4	46.3	303.7	314.6
Total (TWh/year)	2.25	0.41	2.66	2.76
Econ. Benefits (million $/year)	225	41	266	276
Environ. Benefits (billion CO2 lbs/year)	3.4	0.6	4.0	4.1

5.2 Projected Benefits of PPSE

PPSE is still in its initial proposal, evaluation and study phases and the full deployment of it takes time, discussion, effort and hardware/software modifications provided that it is first embraced as an effective method by the manufacturers. Fortunately, the backward compatibility issue is of less significance in case of PPSE since the devices connected to a PPSE enabled switch must only support some kind of a Pause Notification. Although optional, but a type of Pause Notification is already standardized as PAUSE frames which makes the process of full support of Pause Notifications much easier and faster. However, using the current data and making reasonable assumptions, an initial estimation of how much energy could be saved using PPSE can be made which supports the idea of deploying PPSE on switches. So, similar to the estimations made for EEE, the estimates in this section will also use the current number of switches and provide calculations on energy saving with the assumption that all the current edge switches have adopted PPSE. The estimates here use the current power consumptions of 100 Mb/s switches which are common now in households and small offices. However, if PPSE is ever deployed, it will be on future 10 Gb/s and above switches which will definitely consume more energy.

Table 5.2 – Economic and Environmental Benefits of PPSE

	Adaptive PPSE	Ideal
Assumptions		
Consumption per Switch – ON (W)	10	10
Consumption per Switch – OFF (W)	5	5
Switch Utilization (%)	10	10
Active Switches (millions)	100	100
Electricity Cost ($/kWh)	0.1	0.1
CO2 Emmisions (lbs/kWh)	1.5	1.5
Benefits		
Savings per Switch (%)	80	90
Savings per Switch (W)	4	4.5
Total (MW)	400	450
Total (TWh/year)	3.5	3.9
Econ. Benefits (million $/year)	350	394
Environ. Benefits (billion CO2 lbs/year)	5.3	5.9

Based on data in [58], the number of housing units in the U.S. was about 130 million in 2008. Eliminating about 20% of these units (being vacant of not having an internet connection) and assuming the rest have a broadband internet connection with a small Ethernet switch installed, the number of switches with the potential for deployment of PPSE is around 100 million.

Considering that each switch has multiple ports on each of which the traffic load is very low, the aggregate load on all of the links is assumed to be 7 to 10% in average. At these utilizations, about 80% savings can be gained on each switch using PPSE according to the results shown in Figure 4.6.

To estimate the approximate power usage of a conventional Ethernet LAN switch, a Linksys WRT54G is connected to a Kill-A-Watt Power Meter. With 4 active Ethernet links connected to the switch, an average 10 W power is drawn by the switch. Considering the number of components that can be turned off using PPSE along with transition times and power consumption which were ignored in Chapter 4, it can be reasonably estimated that the switch consumes 50% (5 W) of its full power consumption when in OFF state.

Having explained the basis of the numbers used for estimations in this section, the potential energy savings that can be obtained by deploying Adaptive PPSE on all the currently active switches is approximately 3.5 TWh/year which is about $350 million a years in cost in the U.S. at the current average electricity cost. 3.5 TWh/year of electricity also translates to about 5.3 billion lbs of CO_2 emissions per year. These estimations are summarized in Table 5.2. Also given in Table 5.2 is the ideal savings for reference. Although not absolutely accurate, but the estimations presented above show the potential economic and environmental benefits that deployment of PPSE can achieve.

5.3 Summary and Overall Benefits of EEE and PPSE

In this chapter, the economic and environmental benefits of EEE and PPSE were studied separately. Some assumptions have been made in order to estimate these benefits.

For EEE, it is assumed that all the current 10 Gb/s links are EEE enabled so the current number of active links can be used in the estimations. For PPSE, it is assumed that all the current edge switches are PPSE enabled so the current number of active edge switches can be used in the estimations. Moreover, the average utilizations of 3% and 10% are assumed for edge links and switches respectively.

With the above assumptions, it was estimated that EEE and PPSE could save about 2.6 TWh/year and 3.5 TWh/year respectively. Therefore, the overall electricity savings of EEE and PPSE combined is about 6 TWh/year. This amount of electricity translates to about $600 million of economic benefits at the current average rate of electricity in the U.S. Additionally, 6 TWh/year savings in electricity means around 9 billion lbs less CO_2 emissions which is the environmental benefit of the combination of EEE and PPSE. The performance trade-off of EEE is a few microseconds of per-packet-delay which seems to be negligible. For PPSE, the performance trade-off is a few milliseconds of average per-packet delay which is about half of the human response time.

Chapter 6: Summary and Future Work

Evaluation and improvements of two existing methods to reduce the energy consumption of Ethernet were presented in this thesis.

An independent performance evaluation of EEE which replicates previous work by Reviriego et al. [46] is presented first. Based on this evaluation, it was showed that the energy efficiency of EEE is less than proportional to the utilization of the link. The reason is the high transition times compared to actual packet transmission time. An approach to minimize this waste is Packet Coalescing which groups the packets together and sends each group as a burst of packets. This approach is first proposed and initially evaluated by Reviriego et al. [47]. Packet Coalescing reduces the number of transitions to the number of bursts instead of the number of packets. The results of the experiments on EEE with Packet Coalescing showed that it brings the energy consumption of the link to about the same level as proportional to the link's utilization at the trade-off of a few microseconds of per-packet delay.

The second method was Periodically Paused Switched Ethernet (PPSE) which is intended to reduce the power consumption of Ethernet switches. The idea of PPSE is presented in [6] by Blanquicet and Christensen. PPSE is specifically applicable to edge switches since the traffic on these switches is very light so periods in which the switch is idle are plenty. PPSE method half the transmission of all the links connected to the switch for a fixed period of time and then powers down (OFF state). By the time this period is expired, the switch powers up again and enters the normal operation mode (ON state) to service the incoming packets and the procedure repeats. Two versions of PPSE were studied, Simple PPSE and Adaptive PPSE. Simple PPSE switches between ON and OFF states periodically regardless of the load on the links. Adaptive

PPSE on the other hand determines if the next OFF state should be entered or not based on the comparison between the load and a pre-defined threshold. PPSE predicts the future traffic based on the current traffic and tries to adapt to it. Performance evaluation results showed that Adaptive PPSE is able to achieve the energy consumption level of close to proportional while keeping the average packet delay level to around half of the human response time. This level of delay will be likely not to have excessive negative effect on network performance and average user experience.

The combination of PPSE on the edge switches and EEE on Ethernet links can save about 6 TWh/year of electricity, or about $600 million in the U.S.

6.1 Future Work

The research presented in this thesis can be extended in many directions. The below are a number of further works that can be done to improve the work presented in this research:

- EEE with Packet Coalescing should be studied more in context of the Internet. Its effects on other protocols such as TCP/IP congestion control mechanism should be further investigated. This has been done to some extent in [9] but the full investigation of EEE is future work.
- It is possible to synchronize the periods in which a number of connected switches can enter OFF state. By synchronizing the sleeps periods much more energy than individual sleeps can be saved while maintaining the delay at reasonable levels. However, this conversion will require modifications to PPSE as well as defining a synchronization mechanism among the switches.
- The low-utilization periods prediction mechanism of PPSE which is now based on a single threshold can be improved by adopting a more sophisticated approach such as exponentially weighted moving average method presented in [24]. An initial study of a power management approach based on this method is done in [12].

- It is necessary that all aspects of the effects of PPSE on Internet protocols such as TCP/IP are studied in detail. To this end, implementing PPSE on a well known network simulator is future work. ns-2 can be used as the simulation environment in which PPSE is implemented.
- The functionality of PPSE can be analytically modeled as a single server queue with periodic server breakdowns and deterministic repair times. Solving for such queues which is future work would also be a contribution to the research in queueing theory.

List of References

[1] G. Ananthanarayanan, R. H. Katz, "Greening the Switch," *HotPower 2008*, December 2008.

[2] G. Anastasi, M. Conti, M. Di Francesco, A. Passarella, "Energy Conservation in Wireless Sensor Networks: a Survey," *Ad Hoc Networks*, Vol. 7, May 2009.

[3] "ASUS to Deliver New IEEE Standard on Motherboards," *World Energy Media,* April 2010.

[4] "Average Retail Price of Electricity to Ultimate Customers by End-Use Sector, by State," *Electric Power Monthly*, U.S. Energy Information Administration, February 2010.

[5] L. A. Barroso and U. Holze, "The Case for Energy-Proportional Computing," *IEEE Computer*, vol. 40, no. 12, pp. 33–37, December 2007.

[6] F. Blanquicet, "PAUSE Power Cycle: A New Backwards Compatible Method to Reduce Energy Use of Ethernet Switches," White Paper, Version 1.0, Ethernet Alliance, April 2008.

[7] "Carbon Dioxide Emissions from the Generation of Electric Power in the United States," U.S. Energy Information Association.
URL: http://www.eia.doe.gov/electricity/page/co2_report/co2report.html

[8] K. Christensen, C. Gunaratne, B. Nordman, and A. George, "The Next Frontier for Communications Networks: Power Management," *Computer Communications*, Vol. 27, No. 18, pp. 1758-1770, December 2004.

[9] K. Christensen, P. Reviriego, B. Nordman, M. Bennett, M. Mostowfi, and J. A. Maestro, "IEEE 802.3az: The Road to Energy Efficient Ethernet," *submitted to IEEE Communications,* March 2010.

[10] G. Ginis, "Low Power Modes for ADSL2 and ADSL2+," Broadband Communications Group, Texas Instruments, January 2005.
URL: http://focus.ti.com/lit/an/spaa021/spaa021.pdf.

[11] Greenhouse Gas Equivalencies Calculator, EPA.
URL: http://www.epa.gov/cleanenergy/energy-resources/calculator.html.

[12] C. Gunaratne, and K. Christensen, "A New Predictive Power Management Method for Network Devices," *IEE Electronics Letters*, Vol. 41, No. 13, pp. 775-777, June 2005.

[13] C. Gunaratne, K. Christensen, and S. Suen, “Ethernet Adaptive Link Rate (ALR): Analysis of a Buffer Threshold Policy,” *Proceedings of IEEE GLOBECOM,* November 2006.

[14] C. Gunaratne, K. Christensen, and B. Nordman, “Managing Energy Consumption Costs in Desktop PCs and LAN Switches with Proxying, Split TCP Connections, and Scaling of Link Speed,” *International Journal of Network Management*, Vol. 15, No. 5, pp. 297-310, September/October 2005.

[15] C. Gunaratne, K. Christensen, S. Suen, and B. Nordman, “Reducing the Energy Consumption of Ethernet with an Adaptive Link Rate (ALR),” *IEEE Transactions on Computers,* Vol. 57, No. 4, pp. 448-461, April 2008.

[16] C. Guo, L. Zhong, and J. Rabaey, “Low Power Distributed MAC for Ad Hoc Sensor Radio Networks,” *Proceedings of IEEE GLOBECOM 2001*, pp. 2944-294, November 2001.

[17] M. Gupta and S. Singh, “Dynamic Ethernet Link Shutdown for Power Conservation on Ethernet Links,” *Proceedings of the IEEE International Conference on Communications*, June 2007.

[18] M. Gupta and S. Singh, “Greening of the Internet,” *Proceedings of the 2003 ACM SIGCOMM*, pp. 19-26, August 2003. Revised version (2004) URL: http://www.cs.pdx.edu/~singh/ftp/sigcomm03.pdf.

[19] M. Gupta and S. Singh, “Using Low-power Modes for Energy Conservation in Ethernet LANs,” *Proceedings of the IEEE INFOCOM,* pp. 2451–2455, 2007.

[20] M. Gupta, S. Grover, and S. Singh, “A Feasibility Study for Power Management in LAN Switches,” *Proceedings of the 12th IEEE International Conference on Network Protocols (ICNP’04)*, pp. 361-371, October 2004.

[21] W. Hang-Sheng, P. Li-Shiuan, S. Malik, “A Power Model for Routers: Modeling Alpha 21364 and InfiniBand Routers,” *Proceeding of 10th Symposium on High Performance Interconnects,* 2002.

[22] G. Held, “Emerging Technology: The Price of Power Consumption,” *Network Magazine*, September 2001.

[23] P. Huber and M. Mills, “Dig More Coal – The PCs Are Coming,” *Forbes*, May 1999.

[24] Hwang, C. and Wu C., “A Predictive System Shutdown Method for Energy Saving of Event-Driven Computation,” *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, Vol.5 No. 2, pp. 226-241, April 2000.

[25] “IEEE 802.11 Wireless LAN Medium Access Control and Physical Layer Specification,” URL: http://standards.ieee.org/getieee802/802.11.html.

[26] IEEE 802.3az Energy Efficient Ethernet Task Force, URL: http://www.ieee802.org/3/az/index.html

[27] "IEEE P802.3az Energy Efficient Ethernet Standard Draft 2.1," November 2009.

[28] International Energy Agency, "Gadgets and Gigawatts: Policies for Energy Efficient Electronics," First Edition, OECD Publishing, 2009.

[29] "Internet Host Count History," Internet Systems Consortium, URL: http://www.isc.org/solutions/survey/history

[30] R. Jain, "The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling," Wiley-Interscience, New York, NY, April 1991.

[31] T. Karagiannis, M. Molle, M. Faloutsos, and A. Broido, "A Nonstationary Poisson View of Internet Traffic," *Proceedings of IEEE INFOCOM 2004*, November 2004.

[32] C. Keller, "Global Warming: A Review of this Mostly Settled Issue", *Stochastic Environmental Research and Risk Assessment*, August 2009.

[33] A. Keshavarzian, H. Lee, L. Venkatraman, "Wakeup Scheduling in Wireless Sensor Networks," *Proceedings of ACM MobiHoc*, May 2006.

[34] L. Kleinrock, "Queueing Systems. Volume 1: Theory," Wiley-Interscience, 1975.

[35] W. Leland , M. Taqqu , W. Willinger , D. Wilson, "On the Self-Similar Nature of Ethernet Traffic," IEEE*/ACM Transactions on Networking (TON)*, Vol. 2. pp.1-15, February 1994.

[36] S. Makineni, R. Iyer, P. Sarangam, D. Newell, L. Zhao, R. Illikkal and J. Moses., "Receive Side Coalescing for Accelerating TCP/IP Processing," *Proceeding of International Conference on High Performance Computing (HiPC)*, pp 289-300, December 2006.

[37] S. Nedevschi, L. Popa, G. Iannaccone, S. Ratnasamy, and D. Wetherall, "Reducing Network Energy Consumption via Rate-Adaptation and Sleeping," *Proceedings Of NSDI*, April 2008.

[38] B. Nordman, Lawrence Berkeley National Laboratories

[39] B. Nordman and K. Christensen, "Greener PCs for the Enterprise," *IEEE IT Professional*, Vol. 11,No. 4, pp. 28-37, July/August 2009.

[40] B. Nordman and K. Christensen, "Reducing the Energy Consumption of Network Devices," tutorial, *IEEE 802 LAN/MAN Standards Committee Plenary Session,* San Francisco, July 2005.

[41] A. Odlyzko, "Data Networks are Lightly Utilized, and Will Stay That Way," *Review of Network Economics*, Vol. 2, No. 3, pp. 210-237, September 2003.

[42] R. Pang, M. Allman, M. Bennett, J. Lee, V. Paxson, and B. Tierney, "A First Look at Modern Enterprise Traffic," *ACM SIGCOMM/USENIX Internet Measurement Conference*, October 2005.

[43] V. Paxson, Floyd S., " Wide-Area Traffic: The Failure of Poisson Modeling," *IEEE/ACM Transactions on Networking,* Vol.3, No.3, pp. 226-244, June 1995.

[44] V. Paxson, “Some Perspectives on the Performance Impact of Link-Speed Switching Outages,” *Presentation at IEEE 802.3 Meeting,* July 2007.

[45] “Report to Congress on Server and Data Center Energy Efficiency,” EPA ENERGY STAR Program, *U.S. Public Law 109-431*, August 2007.

[46] P. Reviriego, J. Hernandez, D. Larrabeiti, and J. Maestro, “Performance Evaluation of Energy Efficient Ethernet,” *IEEE Communications Letters*, Vol. 13, No. 9, pp. 1-3, September 2009.

[47] P. Reviriego, J. Hernadez, D. Larrabeiti, J. Maestro, “Burst Transmission in Energy Efficient Ethernet,” *IEEE Internet Computing*, March 2010.

[48] Rodríguez-Pérez, M., Herrería-Alonso, S., Fernández-Veiga, M., and López-García, C., “Improved opportunistic sleeping algorithms for LAN switches,” *Proceedings of the 28th IEEE Conference on Global Telecommunications* November/December 2009.

[49] C. Schurgers, V. Tsiatsis, M. B. Srivastava, “STEM: Topology Management for Energy Efficient Sensor Networks,” *IEEE Aerospace Conference*, March 2002.

[50] H. Schwetman, “CSIM19: A Powerful Tool for Building System Models,” *Proceedings of the 33rd Winter Simulation Conference,* pp. 250-255, December 2001.

[51] S. Singh and C. Raghavendra, “PAMAS: Power Aware Multi-Access Protocol with Signaling for Ad Hoc Networks,” *Computer Communications Review*, Vol. 28, No. 3, pp. 5-26, July 1998.

[52] “SMART 2020: Enabling the Low Carbon Economy in the Information Age,” The Climate Group on behalf of the Global eSustainability Initiative (GeSI), 2008.

[53] V. Soteriou, P. Li-Shiuan, “Dynamic Power Management for Power Optimization of Interconnection Networks Using On/Off Links,” *Proceedings of 11th Symposium on High Performance Interconnects*, pp. 15-20, August 2003.

[54] M. Stemm and R. Katz, “Measuring and Reducing the Energy Consumption on Network Interfaces in Handheld Devices,” *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Science,* pp. 1125-1131, August 1997.

[55] H. Tamura, Y. Yahiro, Y. Fukuda, K. Kawahara, Y. Oie, “Performance Analysis of Energy Saving Scheme with Extra Active Period for LAN Switches” *Global Telecommunications Conference,* 2007

[56] N. Tolia, Z. Wang, M. Marwah, C. Bash, P. Ranganathan, and X. Zhu, “Delivering Energy Proportionality with Non Energy-Proportional Systems – Optimizing the Ensemble,” *HotPower*, December 2008.

[57] M. Tzannes, “ADSL2 Helps Slash Power in Broadband Designs,” *CommsDesign*, January 2003. URL: http://www.commsdesign.com/design_corner/OEG20030130S0008.

[58] USA QuickFacts from the U.S. Census Bureau,
URL: http://quickfacts.census.gov/qfd/states/00000.html