Packet Coalescing for Dual-Mode Energy Efficient Ethernet: A Simulation Study

Mehrgan Mostowfi
School of Mathematical Sciences
University of Northern Colorado
CB 120, 501 20th St
Greeley, Colorado 80369
USA
mehrgan.mostowfi@unco.edu

ABSTRACT

Energy Efficient Ethernet (EEE) is a standard that introduces a Low Power Idle (LPI) mode to Ethernet links to reduce their power consumption between packet transmissions. An EEE link can transition to LPI when there are no packets in its buffer to transmit – idle periods. Transition times to and from LPI are significantly high, which prevents EEE from taking full advantage of the link's idle periods. Packet coalescing is to collect multiple packets before sending them on a link as a burst of back-to-back packets. By coalescing packets into bursts, the overhead of transition times can be reduced and nearly energy-proportional operation can be achieved.

Only one LPI mode as described above is defined for Ethernet links with the capacity of 10 Gb/s and lower. However, two modes of low-power operation are defined for EEE links at 40 Gb/s and higher; Fast Wake and Deep Sleep. The Dual-Mode EEE can achieve significant savings that may not be achievable with a single LPI mode. In this paper, applying packet coalescing to Dual-Mode EEE is studied. Performance evaluation using simulation shows that packet coalescing can result is nearly energy-proportional operation in Dual-Mode EEE for 40 Gb/s and above, albeit with some modifications and trade-offs.

Keywords

Energy Efficiency, Networks, IEEE 802.3bj, Performance Evaluation, Simulation

1. INTRODUCTION

Conventional Ethernet links stay fully powered on all the time regardless of whether there is a packet in the link's buffer to transmit/receive or not. However, it is a known fact that there are a high number of "idle periods" for an Ethernet interface where there is no activity on the link. Even in environments with seemingly very high traffic load such as datacenter links, research has shown that the typical utilization of Ethernet links is surprisingly between 10 and 50% [2]. Energy Efficient Ethernet (EEE) is a standard to reduce the energy consumption of Ethernet links by powering down the link during the idle periods [8].

EEE defines a low-power mode for an Ethernet link called the Low-Power Idle Mode (LPI). In the LPI mode, the physical layer of an Ethernet interface is powered off and some elements in the receiver are stopped. As a result, the power consumption of the link is reduced to a fraction of its peak consumption.[1] The link can enter LPI when there are no packets to transmit and return to normal operation – the Active mode – once a packet arrives for transmission from upper layers. It is estimated that implementing EEE can result in significant energy savings – in the order of about $180 million per year if it were implemented in all current 1 Gb/s edge links in both residential and commercial buildings.

EEE has shown to be inefficient because of high transition times especially in cases where packets arrive to the link in single-packet or small batches. In these cases, the link wakes up to transmit a single packet and transitions back to LPI, with each transition taking multiple times that of a packet transition while consuming almost the same power as the link's peak power [14]. A method to reduce this inefficiency is to coalesce the outgoing packets into bursts and consequently decrease the number of necessary transitions to one per burst. This method, called Packet Coalescing, was introduced as an addition to EEE in 2010 ([3] and [15]) and was shown to be able to achieve nearly energy-proportional operation for EEE links. Packet Coalescing was inspired by a similar method called Receive-Side Coalescing which had been in use in many high-speed Ethernet interfaces – mostly on the receive side – to reduce CPU overhead for packet processing [11].

[1]Note that it may not be correct to use the verb "consume" for power. However, power and energy are sometimes used interchangeably in the context of Green Computing, which is, strictly speaking, incorrect. Energy is the work done in a system to generate the desired outcome (transmission of bits, calculation, etc.), while power is the rate at which energy is consumed (or produced). While special care has been taken in this paper not to confuse the two concepts, the verb "consume" is sometimes used for both power and energy, as is common in the literature. The term "power draw" is the correct term.

SIMUTOOLS 2015, August 24-26, Athens, Greece

DOI 10.4108/eai.24-8-2015.2261058

EEE was originally developed for 100 Mb/s, 1 Gb/s, and 10 Gb/s Ethernet, but not for any higher capacity. However, after the standardization of 40 Gb/s and 100 Gb/s in 2010, an optional EEE capability was added to the standard in 2014. EEE for 40 Gb/s and 100 Gb/s differs from that of 10 Gb/s and lower in that it contains an additional mode called Fast Wake. The reason to include this additional mode was likely to address the even longer transition times in the 40 Gb/s and 100 Gb/s EEE. It will be argued later in this paper that the additional mode can act as an intermediate low-power mode and enable additional energy savings for 40 Gb/s and 100 Gb/s EEE. The main question addressed in this paper is how Packet Coalescing can be implemented in Dual-Mode EEE – EEE with two modes of low-power operation – and what its trade-offs would be.

The contributions of this paper are proposing and developing a Packet Coalescing method for Dual-Mode EEE, and the performance evaluation of this method by simulation.

2. RELATED WORK

The first simulation study of EEE was performed in the pioneering work of Reviriego et al. [14] in 2009 where the inefficiency of EEE caused by relatively-long transition times was explored. Later, an analytical model of EEE was developed in 2011 [12]. Packet Coalescing for EEE was proposed and studied by simulation in [3] and [15] and was modeled analytically shortly after in [6] and [13].

Two modes of low-power operation was first introduced as a potential option for 100 Gb/s EEE in a presentation at an IEEE 802.3bj meeting in 2012 [1]. In this presentation, a limited performance study of EEE with two modes with burst transmission of packets where a few packets are coalesced and sent as a batch on the link was presented. The main focus of this presentation was to show the possibility of this approach, not a detailed performance evaluation, so the simulation model was simplistic, the coalescing method was not documented and the results were preliminary. To the best of the author's knowledge, no thorough investigation of Packet Coalescing for Dual-Mode EEE, without any modifications to what is defined in the standard nor with modifications similar to what proposed here, has ever been performed.

3. DUAL-MODE EEE

In 2008, the IEEE 802.3 working group determined that higher capacities than 10 Gb/s are needed to address the growing demands of packet switched networks, and formed the IEEE P802.3ba 40 Gb/s and 100 Gb/s Ethernet Task Force to draft a standard for Ethernet at these capacities. The final result of this effort was the amendment of the IEEE Std 802.3ba 2010 40 Gb/s and 100 Gb/s Ethernet to the IEEE Std 802.3 2008 Ethernet standard in 2010. In another amendment to the standard, the IEEE Std 802.3bj, the optional EEE capability to 40 and 100 Gb/s Ethernet was defined.

EEE for 40 Gb/s and 100 Gb/s Ethernet as defined in IEEE 802.3bj contains a Deep Sleep mode which is identical to LPI. In addition to Deep Sleep, a second mode called Fast Wake is also defined which does not yield any power savings; this mode is only used to keep the sender and receiver in alignment while the receiver transitions to Deep Sleep, and is practically the same as the Active mode. The reason for introducing Fast Wake is likely that with only one LPI mode the transition time to and from the mode would be impractically high due to the long times needed for removing and reapplying power, as well as realigning the line.

Even though Fast Wake is not defined as a low-power mode, it seems that only minimal functionality of the interface is needed in this mode to continue sending LPI signals, so some components can potentially be turned off or powered down. As suggested in [1], the second low-power mode can be without clock stopping yielding limited power reduction, while having a short transition time compared to Deep Sleep. The changes in the circuitry of an EEE interface to accommodate two low-power modes is beyond the scope of this paper and needs a separate research effort. However, based on the discussions in [1] and the definition of Fast Wake in [7], reducing the power consumption of the interface in Fast Wake seems feasible.

Assuming that both Fast Wake and Deep Sleep would reduce the power consumption, the functionality of Dual-Mode EEE would be as follows:
The link transmits all the packets in its buffer until the buffer is empty. Then, it transitions to and stays in Fast Wake until either a certain time passes or a packet arrives to the buffer, whichever occurs earlier. If a packet arrives, the link transitions back to the Active mode, transmits the buffered packets and goes back to Fast Wake. Otherwise, if the certain amount of time passes without any packet arrivals, the link transitions to Deep Sleep where it stays until a packet arrives, in which case it wakes up to transmit it, and this cycle repeats. The energy savings come from the difference in the power consumption in the Active, Fast Wake, and Deep Sleep modes. The opportunity of sleeping in both Fast Wake and Deep Sleep is made by the gaps between packet arrivals. The main trade-off would be the added packet delay due to the packets waiting in the buffer until the link wakes up if they arrive while the link is in Fast Wake or Deep Sleep. Transitions to and from Deep Sleep would naturally be relatively much longer than to and from Fast Wake as defined in the standard [7].

4. PACKET COALESCING FOR DUAL-MODE EEE

As is expected and will be confirmed later by the results of the experiments, the Dual-Mode EEE described above suffers from long transition times similar to the conventional EEE and would be far from energy-proportional. Packet Coalescing can improve its energy efficiency by minimizing the long transition times effect.

The Packet Coalescing method proposed here combines two types of coalescing similar to the method in [3]; count-based and time-based. In count-based coalescing, a certain number of packets are grouped in one batch and transmitted in one burst. In time-based coalescing, packets are coalesced for a certain period of time and all the packets that are grouped during the predetermined time (regardless of their count) are transmitted in one burst. When combined, packets will not remain in the coalescing buffer forever when the load is low (they would if only count-based coalescing is used)

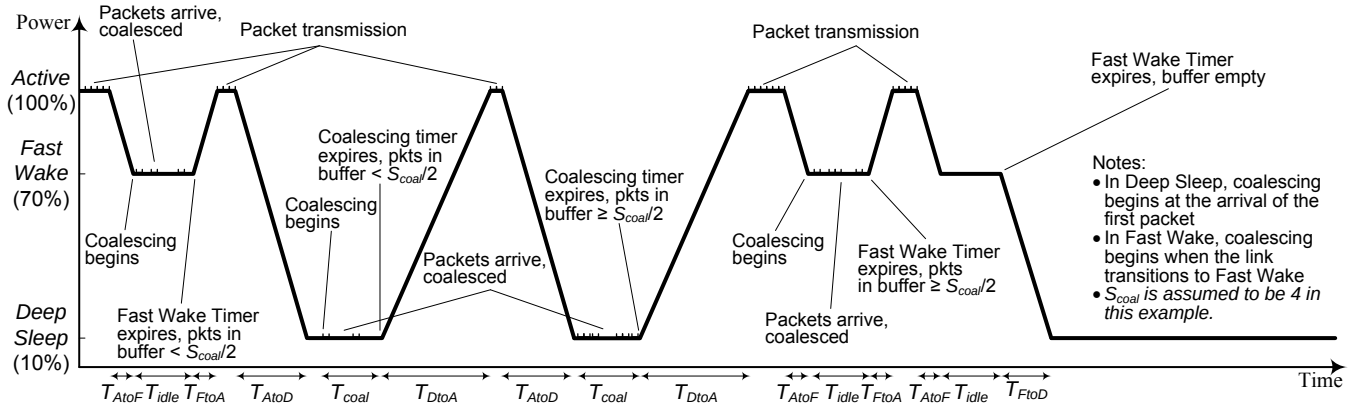

Figure 1: Dual-Mode Energy Efficient Ethernet with Packet Coalescing

Table 1: Constants and Variables in Dual-Mode EEE with Packet Coalescing (some are used in Fig. 1)

Const/Var	Description	Value(s) in sim model
T_{AtoF}	Transition time from Active to Fast Wake	0.18 μs
T_{FtoA}	Transition time from Fast Wake to Active	0.34 μs
T_{AtoD}	Transition time from Active to Deep Sleep	0.90 μs
T_{DtoA}	Transition time from Deep Sleep to Active	5.50 μs
T_{FtoD}	Transition time from Fast Wake to Deep Sleep	0.72 μs
T_{idle}	Time to spend in Fast Wake before either going to Deep Sleep or waking up	3.00 μs
T_{coal}	Coalescing time in Deep Sleep	3.00 μs and 30.00 μs
S_{coal}	Coalescing count in Deep Sleep	10 pkts and 100 pkts

and the size of bursts would be kept to a reasonable level when the load is high (it would not be if only time-based coalescing is used).

A schematic view of power consumption versus time in Dual-Mode EEE with Packet coalescing is depicted in Figure 1. The power consumption in Fast Wake is arbitrarily assumed to be 70% of the link's peak consumption, although it is about the same power reduction suggested in [1] based on the power breakdown numbers for clock-only reduced-power scenario. The power consumption in Deep Sleep is assumed to be 10%, which is the same as that of LPI in conventional EEE as shown in practice [16]. The constants and variables used in Figure 1 along with the values used for them in the simulation model and the experiments later are enumerated in Table 1.

As can be seen in Figure 1, the link transmits all the buffered packets and transitions to the Fast Wake mode (transition duration T_{AtoF}). It then stays in Fast Wake for a fixed and predetermined coalescing period (duration T_{idle}), while buffering the arriving packets. Once the time period ends, the link transitions to the Active mode again to transmit the burst of the coalesced packets (transition duration T_{FtoA}). Before waking up, the link determines whether to go straight to Deep Sleep after transmitting the burst, or transition to Fast Wake again by looking at the number of coalesced packets and comparing to a predetermined threshold. This is a simplistic mechanism to determine if the load is likely to be low in the next burst. If it is (determined by the number of coalesced packets falling below the threshold), then it makes sense to directly transition to Deep Sleep. Otherwise, it may be better to transition to Fast Wake to keep more packets from experiencing the transition delay from Deep Sleep to Active at the end of coalescing. The threshold was set here to half the coalescing count in Deep Sleep since it showed in the simulation trials that this threshold yields improvements in the added delay to the packets while not compromising the energy savings significantly. Once in Deep Sleep, time-based and count-based coalescing both begin by the arrival of the first packet to the coalescing buffer. When a certain time passes (duration T_{coal}) or a certain number of packets are buffered (the coalescing count, S_{coal}), coalescing ends by the link transitioning to Active to transmit the buffered packets in one burst (transition duration T_{DtoA}). Determining the next mode after the transmission of the burst is done the same way as described for the Fast Wake mode. Note that if by the end of the coalescing time in Fast Wake no packets arrive to the buffer, the link transitions directly to Deep Sleep (transition duration T_{FtoD}).

The values for transition times in the simulation model were taken from the standard [7]. The focus in this paper is on 40 Gb/s EEE but the developed simulation model can be extended to other link capacities with minimal modifications. As such, the transition times correspond to those of 40 Gb/s EEE. A T_{idle} of a few times larger than T_{FtoA} was chosen to compensate for the power consumed during transitions. Also, two types of coalescer were used by setting T_{coal} and S_{coal} accordingly to study the effects of the coalescing count and time on the response variables, which will be explained in more detail in Section 6.

The power consumption of the link after working for a cer-

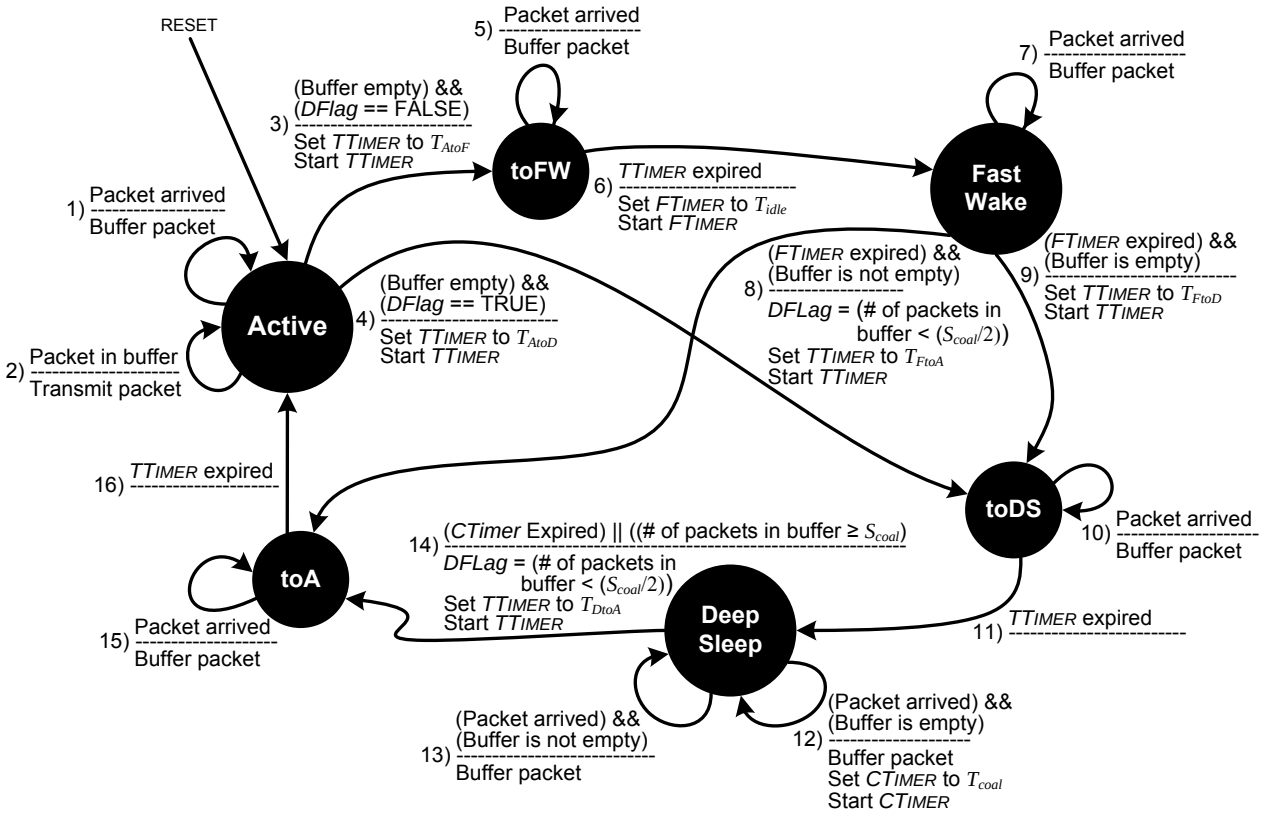

Figure 2: The FSM according to which the simulated EEE link with two modes of low-power operation works

tain period of time can be obtained by using an equation. Let P denote the power consumption of the link in a total period of time as a percentage of the link power consumption if it were active the entire time. P can be obtained by

$$\begin{aligned} P &= t_A P_A + t_F P_F + t_D P_D + \\ &\quad t_{AtoF} P_{AtoF} + t_{FtoD} P_{FtoD} + t_{AtoD} P_{AtoD} + \\ &\quad t_{FtoA} P_{FtoA} + t_{DtoA} P_{DtoA}, \end{aligned} \quad (1)$$

where t_A, t_F, and t_D are the fractions of total time that the link spends in Active, Fast Wake, and Deep Sleep modes, respectively, and t_{AtoF}, t_{FtoD}, t_{AtoD}, t_{FtoA}, and t_{DtoA} are the fractions of total time that the link spends in transition from Active to Fast Wake, Fast Wake to Deep Sleep, Active to Deep Sleep, Fast Wake to Active, and Deep Sleep to Active, respectively. P_A, P_F, and P_D are the power consumption of the link as a ratio of the link's peak power consumption in Active, Fast Wake, and Deep Sleep modes, respectively.

5. THE SIMULATION MODEL

A unidirectional EEE link with Active, Fast Wake, and Deep Sleep modes and Packet Coalescing as described in the previous section was simulated using the CSIM simulation library in C [4]. The Ethernet link was simulated by a CSIM server facility with the service rate of the link's full capacity when active. Figure 2 shows the Finite State Machine (FSM) that controlled the server and was implemented in the sending side of the Ethernet link. In the FSM, the circles denote state in which the FSM can be at any time, and the arrows denote transitions. The event which triggers a transition is listed on top of the corresponding transition arrow, and the resulting action performed at the same time as the transition occurs is listed below the arrow. Time passes when the FSM is in a state, but not during transitions. The FSM in Figure 2 contains 6 states. In the Active state, the link is fully powered-on and operational. Packets are queued in the link's buffer and transmitted in a FIFO order. In the Fast Wake and Deep Sleep states, the link is sleeping and packets are coalesced in the link's buffer to be transmitted later when the link is active. The toFW, toDS, and toA states represent the transition of the link into Fast Wake, Deep Sleep, and Active, respectively. Three timers and a variable are defined in the FSM – Table 2 summarizes them with a brief description for each. They will be explained in detail with the FSM's transitions below.

Table 2: Variables and Timers in the FSM of Fig. 2

Variable/Timer	Description
CTimer	Timer to keep time since colaescing started
FTimer	Timer to keep time spent in Fast Wake
TTimer	Timer to keep transition times
DFlag	Boolean variable to determine if link should transition to Deep Sleep or Fast Wake after transmitting the currently buffered packets

The FSM starts and resets into the Active state. The variable DFlag is set to false and all timers are stopped upon reset. The link remains in the Active state as long as packets arrive. Arriving packet when the link is in the Active state are buffered and transmitted in order (transitions 1 and 2). When the link transmits all the packets in the buffer (buffer is empty), the link either transitions into the Fast Wake state or Deep Sleep state, depending on the value of DFlag. DFlag is a boolean variable which determines whether the next state to go to after Active is Fast Wake or Deep Sleep. The case where DFlag is false is considered first and the case where it is true is explained at the end of the FSM description.

If DFlag is false, the link transitions to Fast Wake via the toFW state (transition 3). Upon entering toFW, the timer TTimer is set to T_{AtoF} and starts ticking down to zero to simulate the transition time from Active to Fast Wake. While in toFW, arriving packets are buffered (transition 5). Once TTimer expires, the link transitions to the Fast Wake state. The timer FTimer is set to T_{idle} and starts ticking down to keep the time for which the link stays in Fast Wake before waking up to transmit the buffered packets, if any (transition 6). While in Fast Wake, arriving packets are buffered (transition 7) – they are held in the buffer, or coalesced, until the link wakes up. When FTimer expires, the link's buffer is either empty or not. If it is not empty, the link wakes up to transmit the packets by transitioning to Active via the ToA state (transition 8). At transition 8, if the number of buffered packets is above a predetermined threshold (here, the half the coalescing count, $S_{coal}/2$), DFlag is set to false so that the link goes to Fast Wake after transmitting all the buffered packets in the Active state. Otherwise it is set to true so that the link goes to Deep Sleep after transmitting all the buffered packets. Upon entering toA, TTimer is set to T_{FtoA} and starts ticking down to zero to simulate the transition time from Fast Wake to Active. If the link's buffer is empty when FTimer expires while in Fast Wake, the link enters Deep Sleep via the ToDS state (transition 9). Upon entering toDS, TTimer is set to T_{FtoD} and starts ticking down to zero to simulate the transition time from Fast Wake to Deep Sleep. As can be inferred from the FSM functionality, coalescing in the Fast Wake state is time-based only; the link holds all the packets received in a T_{idle} period and either wakes up to transmit them all in one batch, or goes to Deep Sleep if there is none.

The link enters Deep Sleep from the ToDS state (transition 11) when the transition time passes. When the first packet arrives to the buffer while the link is in Deep Sleep, the timer CTimer is set to T_{coal} and starts ticking down to zero to simulate coalescing time while in Deep Sleep (transition 12). Other arriving packets are buffered while in Deep Sleep (transition 13) until coalescing ends by either CTimer expiring (time-based coalescing) or the number of buffered packets exceed the coalescing count, T_{coal} (count-based coalescing). If either case happens, the link wakes up to transmit the coalesced packets by entering the Active state via the ToA state (transition 14). Similar to how DFlag was set in Fast Wake, if the number of buffered packets is above a predetermined threshold, $S_{coal}/2$, DFlag is set to false so that the link goes to Fast Wake after transmitting all the buffered packets in the Active state. Otherwise, it is set to true so that the link goes to Deep Sleep after transmitting all the buffered packets. When TTimer expires in the toA state to simulate the transition time from Deep Sleep to Active, the link wakes up by entering the Active state (transition 16). Note the difference between coalescing in Fast Wake and Deep Sleep; coalescing in Fast Wake is time-based only and the coalescing time is measured from the point the link enters Fast Wake. However, coalescing in Deep Sleep is both time-based and count-based and the coalescing time begins when the first packet arrives to the link in Deep Sleep.

Another way to enter the Deep Sleep state is directly from Active when the buffer becomes empty in the Active state and DFlag is true. DFlag is set to true when in either Fast Wake or Deep Sleep the number of coalesced packets is lower than a predetermined threshold. This threshold is arbitrarily set to half of S_{coal} in the experiments here, but it really is a simplistic mechanism to determine if the load is likely to be low, so that powering down the link straight to Deep Sleep may add an insignificant delay to the average per-packet delay. Entering Deep Sleep from Active without going to Fast Wake happens by transition 4 where the link transition to Deep Sleep via toDS. Upon entering toDS, the timer TTimer is set to T_{AtoD} and starts ticking down to zero to simulate the transition time from Active to Deep Sleep.

In the simulation model, the link's capacity, transition times T_{FtoA}, T_{AtoD}, T_{DtoA}, T_{FtoD}, coalescing parameters T_{idle}, T_{coal}, S_{coal}, the distribution by which the packets arrived to the link, the size of packets, and the power consumption in each state as a percentage of the link's peak consumption were variables adjusted based on the experiments. The rationale behind the chosen values for these variables is explained next. The model was instrumented to measure the average packet delay, and the overall power consumption as a percentage of the link's peak power consumption – the power consumption of the link if it were active all the time.

6. EXPERIMENTS

Dual-Mode EEE with Packet Coalescing was evaluated by experiments performed on the simulation model. In the experiments, the link's capacity was set to 40 Gb/s. The transition times were set to the values stated in Table 1 which are their minimums in the standard [7]. The coalescing time in the Fast Wake mode, T_{idle}, was set to 3.00 μs, about three times T_{FtoA} based on the discussion in Section 4. The power consumption of the link during each experiment was calculated by Equation 1 with P_A, P_F, and P_D set to 100%, 70%, and 10% of the link's peak consumption, respectively. All the experiments were done for a large enough number of packet arrivals to achieve a 95% confidence on the mean average delay of packets. The packet size was fixed to 1500 bytes, the maximum transmission unit of Ethernet. Two sets of experiments were designed.

In the first set, called the Smooth Traffic Experiments, packets arrived to the link according to a Poisson distribution. Although it is known that network traffic is likely to be bursty [10], a Poisson distribution remains a reasonable first-order approximation in cases where the traffic is highly aggregated (such as in datacenter links) and the traffic sample is taken in small (sub-second, typically) time spans [9]. In these experiments, the arrival rate, λ, was varied to simulate

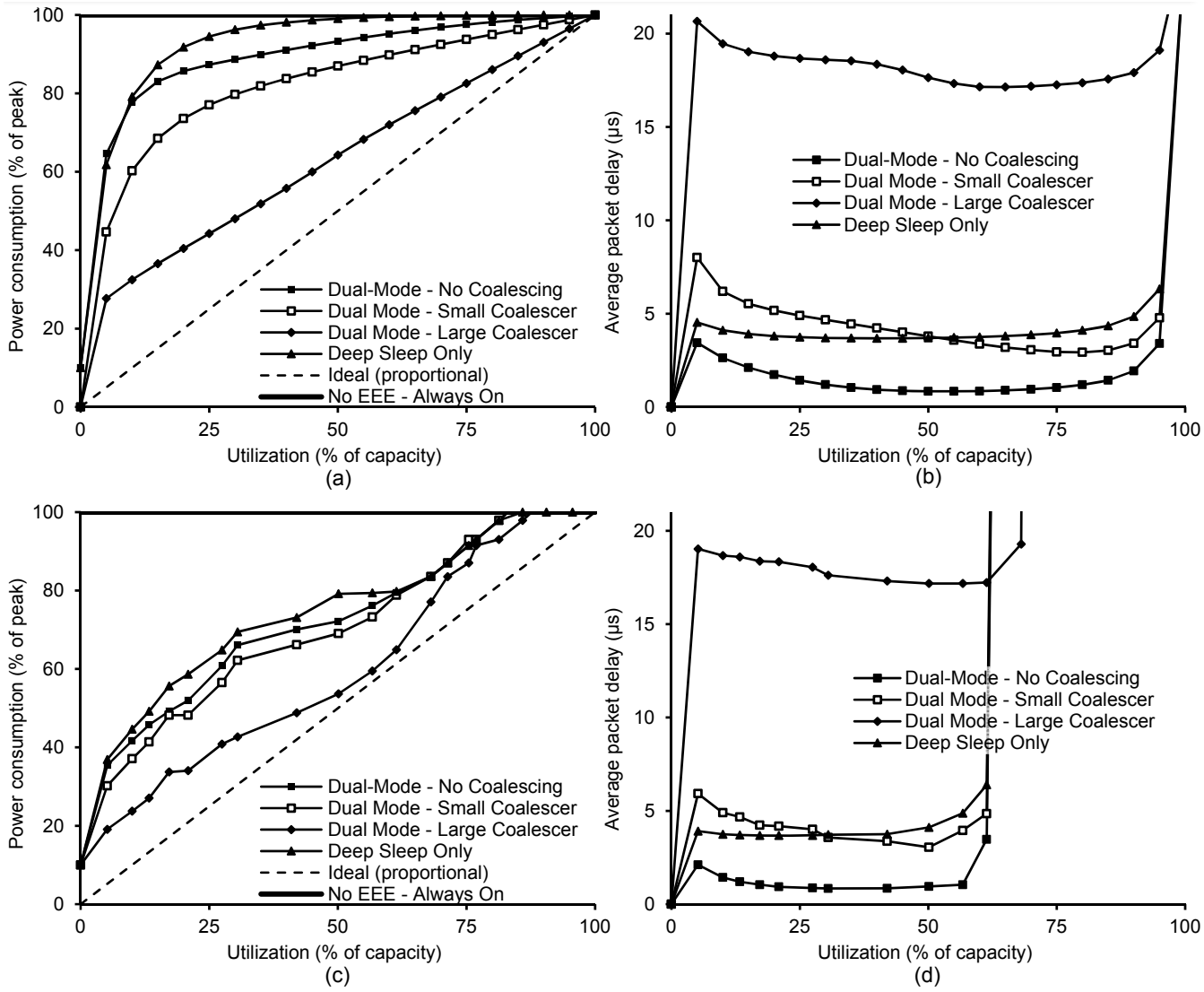

Figure 3: Experiment results – Power consumption and average packet delay versus utilization in Smooth Traffic (a and b) and Bursty Traffic (c and d) experiments

loads ranging from 5% to 95%.

In the second set, called the Bursty Traffic Experiments, packets arrived to the link according to an Interrupted Poisson Process (IPP). IPP is among the distributions commonly used to approximate bursty traffic of various types [5]. The IPP distribution has two states, On and Off. In the On state, packets are generated with the rate λ, and in the Off state no packets are generated. Transitioning from On to Off state and vice versa occurs with rates α and β, respectively. which were both set to 10, to simulate short bursts of large number of packets. λ was adjusted to yield loads of 5% to 95% in the long run.

For each set of experiments, two sizes of a coalescer were simulated by setting the T_{coal} and S_{coal} parameters. For the "Small Coalescer", 3.00 μs and 10 were assigned to these parameters, respectively, and for the "Large Coalescer", 30.00 μs and 100 were used. Using coalescer sizes is similar to what was done in [3] to determine the effects of the coalescer parameters on the response variables and the trade-offs.

7. RESULTS

Figure 3 shows the results of the experiments. For the two sets of experiments, the power consumption of the link which was calculated by equation 1 (total times spent in each mode or transition was obtained by instrumenting the simulation model) and the average per packet delay (again, obtained by instrumenting the simulation model) as a function of utilization on the link (offered load) are shown. Also shown for comparison are the power consumption and average packet delay when no coalescing is used in Dual-Mode and when only a single mode of low-power operation, Deep Sleep, is used. Dual-Mode with no coalescing means when the link enters the Fast Wake mode after transmitting all the buffered packets, it stays there for T_{idle} and if no packet arrives meanwhile transitions to Deep Sleep. If a packet ar-

rives in Fast Wake, the link immediately breaks Fast Wake, wakes up and transmits all the buffered packets and enters Fast Wake again. Similarly, arrival of the first packet while in Deep Sleep results in the link wakeup to Active. EEE with Deep Sleep only is equivalant to the current EEE (without coalescing. of course) where the link directly transitions to the only defined low-power mode after transmitting all the buffered packets in the Active mode (transition time T_{AtoD}). In Figure 3a and c where the power consumption is shown, the power consumption when no EEE is used (always 100% of the peak) as well as the ideal power consumption (proportional to load) are also shown.

Smooth Traffic Experiments. Figure 3a shows the power consumption of the link as a function of utilization for Dual-Mode EEE with large and small coalescers. It can be seen that for small loads up to about 10%, Dual-Mode with no coalescing and a single-mode EEE with Deep Sleep mode only yield about the same power consumption, although it is very high compared to the load (about 60% for 5% load, and 80% for 10% load). This is expected due to the fact that transitioning to Deep Sleep takes a long time compared to that of Fast Wake, and was assumed to consume 100% of the link's peak consumption. Each packet arrival in Deep Sleep causes a T_{DtoA} transition and therefore it is very wasteful of power. It is the same reason that causes Deep Sleep only and Dual-Mode with no coalescing to yield a very high power consumption as the load increases from low loads to high loads. As can be see in the figure, the power consumption is within 10% of the link's peak consumption even when the load is moderate around 25%. High transition times of EEE was known since 2009 [14], as was indeed the motivation of introducing Packet Coalescing in [3] and other related studies that followed it. Coalescing significantly decreases the effect of high transition times, as can be seen in the figure. The small coalescer almost consistently drops the power consumption by about 10 to 20 percentage points for every load level. The reason, as expected and confirmed by instrumenting the simulation model, is that coalescing causes more packet arrivals per wakeup transition. It also causes the link to stay in both Fast Wake and Deep Sleep for longer on average, especially when the load is low. The effect of coalescing was even more significant when the large coalescer was used. The power consumption of EEE with the large coalescer is very close to the ideal consumption, which is proportional to load. Note that the results here reinforces what was concluded in [14] for single-mode EEE with Packet coalescing for capacities of less than 10 Gb/s.

Dual-Mode EEE – with and without Packet Coalescing – indeed comes with a trade-off. As can be seen in Figure 3b and is expected, Dual-Mode EEE without coalescing causes the least added delay overall. The reason is that while in Fast Wake or Deep Sleep, an arriving packet causes the link to wake up so it experiences minimal delay compared to when coalesced with other packets and transmitted later. What is counterintuitive is that the average delay decreases as the load increases. The reason is that when the load increases, the probability of a packet arriving while the link is still in Fast Wake and has not transitioned to Deep Sleep rises. This results in the link transition more often from Fast Wake to Active to transmit a packet, which has a transition time of much shorter than Deep Sleep to Active. Naturally, when the load is extremely high (80% an more) the average delay starts to rise again dues to increased queuing delay.

Packet Coalescing has a larger impact on the average packet delay. The average packet delay starts at 5 to 7 μs for low loads and drops to 4 to 5 μs for higher loads when the small coalescer is in use. The reason is intuitive. The coalescer holds a number of packets before releasing them; the larger the coalescer, the longer the time the packets are held in the coalescer on average, and the higher the average packet delay would become. For instance, the small coalescer causes an average of 5 μs of delay while the large coalescer causes about 19 μs at 30% load.

Bursty Traffic Experiments. When the traffic is bursty, large groups of packets arrive to the link with no, or very small gaps between one another. However, the gaps between consecutive bursts of traffic is likely to be relatively long. Compared to the smooth traffic experiments, all methods show improvement in power consumption. When Deep Sleep only is used, the gaps between bursts allow the link to transition to, and stay in Deep Sleep for extended periods. When Dual-Mode with no coalescing is in use the link transitions to Deep Sleep between gaps while still taking advantage of the few and occasional inter-packet gaps within bursts to make a quick transition to Fast Wake and back, yielding a slightly lower power consumption than Deep Sleep only.

It can be seen in Figure 3c that using coalescing results in less power consumption for all load levels when the traffic is bursty compared to when the traffic is smooth in Figure 3a. This is due to a more opportunities of sleeping during gaps when the traffic is bursty. In bursty traffic, the gaps between packets are typically very small, but the idle periods between bursts are relatively large, especially when the traffic load is low. Packet Coalescing further enhances the opportunity of sleeping by consolidating the idle periods, both in and between bursts, into larger ones. The large coalescer almost yields energy-proportional operation, although with a more significant delay trade-off. Note that the delay for utilization levels of more than about 65% is not shown since the link's buffer would overflow as a result of very large packet bursts and start dropping packets.

A fundamental question is if the added delay caused by Packet Coalescing is significant and if it is, to what extent. The answer depends on many factors. For some delay-insensitive applications (such as large file transfers or backups) a few microseconds per packets delay is very likely to be low compared to tens of milliseconds end-to-end delay of typical Internet connections. However, even this much increase in packet delay in a data center or real-time applications can be considered significant, but the additional energy savings gained in a data server may justify a reasonable per-packet delay.

8. CONCLUSIONS AND FUTURE WORK

In this paper, it was shown by simulation that Packet Coalescing for Dual-Mode EEE can be an effective method of extending idle periods between packet arrivals in order to maximize the opportunity for sleeping. Using a large enough coalescer, the power consumption of the link can be brought very close to proportional to load, which is the ideal con-

sumption. The trade-off, the increased packet delay, can be negligible or significant depending on the type of traffic and applications. The energy savings gained by this method may justify the trade-off.

Accurately calculating the potential energy savings by implementing this method – in monetary terms, for instance – is difficult since 40 and 100 Gb/s links are just recently being deployed limitedly in datacenters and are still under development. Additionally, a more detailed study is needed to determine the technical feasibility of the required modifications to EEE in order to implement Dual-Mode EEE as explained here and also Packet Coalescing.

The intention is to extend this work is the following directions:

- The technical feasibility and implications of Dual-Mode EEE with Packet Coalescing needs to be thoroughly investigated.
- Changes to other aspects of packet delay besides average need to be studied to obtain a better understanding of the packet delay trade-off. For instance, the distribution of the packet delay, is also one of the interesting characteristics of the delay.
- Packet coalescing may cause other effects that were not studied here. For instance, an increased burstiness of the traffic is possible which can have adverse effects on the downstream network hops such as routers and switches.
- Studying Dual-Mode EEE with Packet Coalescing under traffic traces from datacenter links is also an important direction which would help understand the impacts of the method on real traffic.

9. REFERENCES

[1] H. Barrass, "Options for EEE in 100G," presentation at *IEEE P802.3bj meeting*, January 2012.

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

[3] K. Christensen, P. Reviriego, B. Nordman, M. Bennett, M. Mostowfi, and J. Maestro, "IEEE 802.3az: The Road to Energy Efficient Ethernet," *IEEE Communications Magazine*, vol. 48, no. 11, pp. 50–56, November 2010.

[4] CSIM 20: Development Toolkit for Simulation and Modeling, Mesquite Software Inc. URL: http://www.mesquite.com/products/csim20.htm.

[5] W. Fischer and K. Meier-Hellstern, "The Markov-Modulated Poisson Process Cookbook," *Performance Evaluation*, vol. 18, pp. 149–171, September 1993.

[6] S. Herreria-Alonso, M. Rodriguez-Perez, M. Fenandez-Veiga, and C. Lopez-Garcia, "A Power Saving Model for Burst Transmission in Energy-Efficient Ethernet," *IEEE Communications Letters*, vol. 15, no. 5, pp. 584–586, May 2011.

[7] "IEEE 802.3bj-2014 Amendment 2: Physical Layer Specifications and Management Parameters for 100 Gb/s Operation Over Backplanes and Copper Cables," IEEE Computer Society, June 2014.

[8] IEEE P802.3az Energy Efficient Ethernet Task Force. URL: http://www.ieee802.org/3/az/index.html.

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

[10] W. Leland, M. Taqqu, W. Willinger, and D. Wilson, "On the Self-Similar Nature of Ethernet Traffic," *IEEE/ACM Transactions on Networking*, vol. 2, pp.1–15, February1994.

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

[12] M. Marsan, A. Anta, V. Mancuso, B. Rengarajan, P. Vasallo, and G. Rizzo, "A Simple Analytical Model for Energy Efficient Ethernet," *IEEE Communications Letters*, vol. 15, no. 7, pp. 773–775, July 2011.

[13] M. Mostowfi and K. Christensen, "An Energy-Delay Model for a Packet Coalescer," in *Proceedings of the IEEE SoutheastCon*, March 2012.

[14] P. Reviriego, J. Hernandez, D. Larrabeiti, and J. Maestro, "Performance Evaluation of Energy Efficient Ethernet," *IEEE Communications Letters*, vol. 13, pp. 697–699, September 2009.

[15] P. Reviriego, J. Maestro, J. Hernandez, and D. Larrabeiti, "Burst Transmission for Energy-Efficient Ethernet ," *IEEE Internet Computing*, vol. 14, no. 4, pp. 50–57, August 2010.

[16] P. Reviriego, K. Christensen, J. Rabanillo, and J. A. Maestro, "An Initial Evaluation of Energy Efficient Ethernet," *IEEE Communications Letters*, vol. 15, no. 5, pp. 578–580, May 2011.