2012 26th IEEE International Conference on Advanced Information Networking and Applications

Greening the Internet using Multi-Frequency Scaling Scheme

Wei Meng*, Yi Wang*, Chengchen Hu†, Keqiang He*, Jun Li* and Bin Liu*
*Tsinghua National Laboratory for Information Science and Technology
Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China
Email: francismw08@gmail.com
†Department of Computer Science and Technology, Xi'an Jiaotong University, Xi'an 710049, China

***Abstract*—In this paper, we have designed a Multi-Frequency Scaling scheme for energy conservation of network devices, especially routers and switches. The frequency of components in a network device is scaled dynamically according to the real time workload. A Markov model is developed for performance analysis of this mechanism. We implement a prototype of this scheme in the data path of a general IPv4 router based on a real hardware platform - NetFPGA. Experimental results show excellent energy savings at the cost of a tolerable latency, under various ranges of traffic loads. Our work indicates the feasibility and possibility of deploying this mechanism into real network devices for energy saving.**

***Index Terms*—green networking; NetFPGA; dynamic frequency scaling**

I. Introduction

The number of Internet based services and Internet end users continues expanding over the last decade, which result in consistent and exponential increase in bandwidth demand as well as traffic volume. It is expected that the annual Internet traffic will reach Zettabytes (10^{21} bytes) in the next five years, while currently it is in Exabyte (10^{18} bytes) [1]. To keep pace with this growing trend, the number of routers and switches deployed globally needs to rise at a corresponding rate. A by-product of the growths in Internet traffic and the amount of network equipment is the increase of power consumption. It is estimated that the power consumption of the Internet is approximately in the range of $1\% - 4\%$ of total electricity consumption in a broadband enabled country now [2]. Among all kinds of network devices (not including PCs attached to network), switches and routers consume most of power consumption of the Internet [3] [2]. Although current link utilization is 30% in average and below 45% in peak [4], switches and routers operate at their full rates all the time (24 hours a day, 7 days a week) [5]. This causes huge waste in energy and meanwhile brings the opportunity for substantial conservation in energy consumption.

A number of power management methods have been proposed to reduce network power consumption, which can be generally divided into two categories: device sleeping mechanism [6] [7] and rate adaptation mechanism [8] [9]. Sleeping scheme powers off idle devices or components into sleep states [7] for a pre-estimated duration, and wakes up the sleeping devices or components when new packets arrive. However, this scheme is fragile against burst traffic [9].Rate adaptation or speed scaling scheme has high robustness against burst traffic. Nevertheless only Ethernet PHY in network devices supports it. Ethernet links only consume a small fraction of energy in switches and routers [5]. The scalability of rate transition is also limited by hardware, that it is difficult to configure the system to best adapt its operating rate to dynamic workload.

In this paper, we propose a *Multi-Frequency Scaling (MFS)* scheme for finer-grained rate adaptation on data paths of switches and routers [1], where the frequency of network devices and components is adapted to their actual workload indicated by the buffer occupancy[2] in our model. The frequency transition is triggered when the buffer occupancy crosses some pre-configured thresholds. In order to prevent unstable frequency switch, we employ dual-threshold for switching between two neighboring frequencies. To validate the design, we first theoretically analyze the performance of the proposed dual-threshold switching method by developing a Markov model, and then we test the performance by implementing a prototype based on NetFPGA [10]. Through the evaluations, we demonstrate that adaptive frequency scaling technique could achieve significant energy saving when deployed to other components of network devices in the data paths, without adversely affecting network performance. In addition, experiment results show that our scheme reduces power consumption of network devices under all kinds of utilization with gentle increase in packets delay. Especially, we make the following contributions in this paper.

- First, we propose MFS with dual-threshold switching for finer-grained rate adaptation as well as avoiding oscillation of rates.
- Second, we implement a real hardware prototype system supporting MFS on the NetFPGA platform [10] to validate our design. To our best knowledge, we are the first to build rate adaptation scheme into the data path of routers for energy conservation as real hardware system.
- Third, we derive a Markov model to investigate the effects of the distribution of available rates and thresholds on system performance. This model is different from

[1]In this paper, we use frequency and rate inter-changeably and assume that hardware supports working at multi-frequencies.

[2]Buffer occupancy denotes number of bytes currently in the buffer.

1550-445X/12 $26.00 © 2012 IEEE
DOI 10.1109/AINA.2012.119

the previous analysis in [8] and [9]. We show that the distribution of rates and thresholds have the most impact on performance.

The remainder of this paper is organized as follows. Section II reviews related work on network energy conservation. The MFS scheme and the power model are presented in Section III. We develop the Markov model of MFS scheme in Section IV, along with discussion about distribution of frequencies and thresholds configurations. Section V describes our prototype implementation on NetFPGA platform. Experiment results are presented in Section VI. Finally, Section VII concludes the paper.

II. Related Work

Today, networks are provisioned for the worst-case or busy-hour load, which typically exceeds their long-term utilization by a wide margin [9]. Meanwhile, most networking devices are dedicated for connection at the full processing speed, even when the utilization is dramatically low. Thus, the energy consumption of network equipment remains substantial even when the network is idle, and it results in a great waste in the energy of network devices.

Researchers have proposed a number of methods to reduce the energy consumption in the network during the last decade. *Dynamic Link Shutdown Mechanism* [7] and *Adaptive Link Rate* [8] are two approaches mainly focusing on energy conservation in Ethernet interfaces. However, other components of network devices are the major power consumer of certain devices, *e.g.* data processing units and queuing buffers in routers and switches [5] [11]. The authors of [8] firstly develop an *ALR Dual-Threshold Policy* which is very similar to our proposed scheme in the idea. They observe that the dual-threshold policy can oscillate between data rates which may cause great increase in packet delays, thus they developed another two policies instead of the dual-threshold policy. Wierman *et al*. [12] studies speed scaling in processor sharing systems. Their research results show that compared with halt-work scheme, dynamic speed scaling could significantly improve the robustness toward burst traffic, with the same performance on energy consumption saving. The authors of [9] consider both sleeping scheme and rate-adaptation for reducing network energy consumption. For rate adaptation, three distributions of rates are investigated: 10 rates and 4 rates uniformly distributed between 1Gbps to 10Gpbs, and 4 exponentially distributed rates (10Gbps, 1Gbps, 100Mbps and 10Mbps). By using these three rate distributions, the authors draw the conclusion that uniformly distributed rates is essential, in that their algorithm performs poorly for exponential distribution. However, the reliability of this result is questioned, since the exponentially distributed rates are in a different range from the two uniformly distributed rates. It is believed that the result would be quite different if the rates are distributed within the same range. Recently, Xinyu *et al*. proposed E-MiLi for energy conservation in wireles networks [13], which shared similar thought of downclocking during idle listening period with our work. E-MiLi focused on ensuring accurate packet detection and address filtering at low-speed model, while our work focused on analyzing the impact of system configurations on performance.

III. Model and Approach

This section describes the general power consumption model of network device, and proposes the Multi-Frequency Scaling scheme as well as methodology used for validation.

A. Objectives

In essence, there are two performance metrics for a rate adaptive scheme of network devices, and we present, analyze and evaluate our MFS scheme in accord to them.

- *Energy saving:* The amount of energy saving is linear to the reduction in rate (as explained in Section IV). Therefore, we use the *rate reduction* to measure the efficiency of energy saving. In particular, we use the average rate reduction which is more meaningful than instantaneous energy saving.
- *Packet delay:* Generally, more time to work on lower frequency, longer delay suffers. Moreover, variation of packet delay brought by frequency switch may have impact on routing protocols that update route using packet delay. Therefore, the rate adaptive scheme must incur limited increase in packet delay. The packet delay consists of *queuing delay* which is used as the metric for delay in this paper, and a constant amount of time for forwarding and switching.

Basically, a rate adaptive scheme needs to strike a balance between energy saving and packet delay even with highly dynamic traffic loads. In the following sections, we investigate, both in theory and in data result, the proposed MFS scheme with the above metrics. We show that it is feasible to achieve excellent energy saving with tolerable cost in traffic delay using multi-frequency.

B. Power Model

Generally, a network device works at two states: *active* state and *idle* state. When the device is actively processing traffic, it works at the active state. When there is no traffic for processing but the device is still powered on, it works at the idle state. Thus, the energy consumption of a general network device could be modeled as:

$$E = P_a T_a + P_i T i, \tag{1}$$

where T_a and T_i denote the time spent in active state and idle state respectively, P_a and P_i represent the power consumption in each mode. For both active and idle states, there is a static portion of power draw which is independent from the device's operating frequency, while the rest of power draw indeed depends on the operating frequency:

$$\begin{aligned} P_a(r) &= C + f(r), \\ P_i(r) &= C + \beta f(r), \end{aligned} \tag{2}$$

where $f(r)$ reflects the dynamic portion of energy consumption working at frequency r, and C denotes the static portion of

energy consumption. The parameter β represents the relative magnitudes of routine work incurred even in the absence of packets to the work incurred when actively processing packets [9]. In general, the dynamic portion of energy consumption depends on the operating frequency and voltage of the network device:

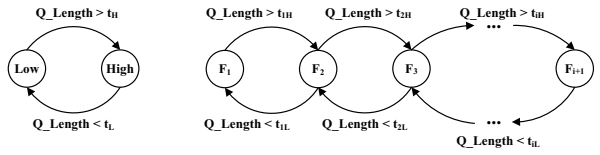
$$f(r) \propto rV_{dd}^2, \tag{3}$$

where V_{dd} is the voltage of the device [14]. Hence, energy consumption could be reduced for both states by scaling the device's operating frequency according to its workload.

C. Multi-Frequency Scaling Scheme

It is observed that buffers are commonly adopted in current design of network devices as queue for processing jobs. Intuitively, the actual buffer occupancy could be considered as indicator for real-time traffic load. Our MFS mechanism uses this indicator for frequency switch. We assume that the hardware supports working at several different rates. The operating rate of devices is dynamically switched according to the buffer occupancy. A range of thresholds are set to evaluate the relative workload of the devices.

To avoid rate oscillation of single threshold, Dual-Threshold scheme is proposed, which uses both high threshold t_H and low threshold t_L for rate switch, as illustrated in Fig. 1 (a). When component works at low rate mode, the buffer occupancy (denoted as Q_Length in Fig. 1) would rise with the increase of traffic load. The transition from low rate to high rate is induced when the buffer occupancy exceeds t_H. On the other hand, when component operate at high speed mode, the queue length would drop while the traffic load decreases. Similarly, a transition from high rate to low rate is caused when the queue length drops below t_L.



a) Dual-Threshold Mechanism

b) Multi-Dual-Threshold Mechanism

Fig. 1. State machine for buffer occupancy based frequency scaling mechanism.

To better adapt the operating frequency to the real workload of the network device, we propose Multi-Dual-Threshold policy in this paper, as illustrated in Fig. 1 (b). There is a dual-threshold pair between every two neighboring frequencies. Theoretical analysis of Multi-Dual-Threshold policy of MFS is discussed in Section IV, using a Markov model.

IV. Theoretical Analysis

This section defines the Multi-Dual-Threshold policy for MFS, along with a Markov model for performance analysis. Then, the impact of rates and thresholds configuration on performance is investigated by following the criteria defined in Section III.

A. Multi-Dual-Threshold policy

We assume that packet arrival and service times follow Poisson distribution. The Multi-Dual-Threshold policy could be modeled as a single server queue, where the service rate depends on state the server working at. To avoid potential rate oscillation and system instability, dual-threshold is introduced into the rate adaptation system.

Defining the packets arrive at a rate λ, and the system is capable of serving at $M+1$ different rates, from μ_1 to μ_{M+1} ($\mu_j < \mu_{j+1}, 1 \le j \le M+1$). The utilization could be derived as $\rho_j = \lambda/\mu_j$, respectively. In the Multi-Dual-Threshold policy, $2M$ buffer occupancy thresholds are defined for rate transition. The M low-thresholds are denoted as TL_j ($1 \le j \le M$), while the M high-thresholds are denoted as TH_j ($1 \le j \le M$), where $TL_j < TH_j < TL_{j+1}, 1 \le j \le M-1$. A buffer occupancy level dropping below a low threshold TL_j causes a service rate switch from μ_{j+1} to μ_j. A buffer occupancy level exceeding or equaling a high threshold TH_j causes the service rate switches from μ_j to μ_{j+1}.

In the rest of this section, we use a Markov model for the Multi-Dual-Threshold policy. Then we present expression for the steady-state probability of n customers (packets) in the system (buffer), as P_n. The denotation π_n is used to represent the steady-state probability of state n, which is different from P_n, and should be noted specially.

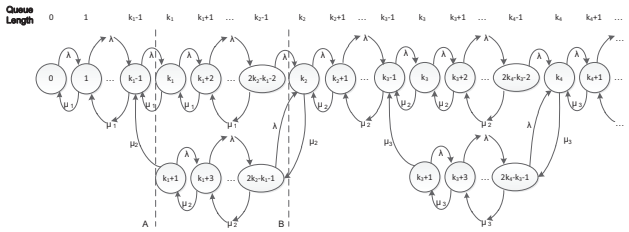


Fig. 2. State machine for Multi-Frequency Scaling mechanism.

B. Steady-State Probabilities

Fig.2 shows the Markov chain for our Multi-Dual-Threshold policy. The heading line represents the buffer occupancy corresponding to the states below. Parameters used in the Markov model are defined or presented as follows:

- λ is arrival rate of packets.
- M is number of high/low buffer occupancy thresholds.
- μ_i is one of the service rates ($1 \le i \le M+1$).
- $\rho_i = \lambda/\mu_i$ is utilization corresponding to μ_i.
- k_{2i-1} represents low buffer occupancy threshold TL_i ($1 \le i \le M$).
- k_{2i} represents high buffer occupancy threshold TH_i ($1 \le i \le M$).
- π_n is the steady-state probability of state n. For buffer occupancy between k_{2i+1} and $k_{2i+2}(0 \le i < M)$, *e.g.* $k_{2i+1}+j$, $\pi_{k_{2i+1}+2j}$ denotes the probability of state in the upper chain while $\pi_{k_{2i+1}+2j+1}$ denotes the probability of state in the lower chain.
- P_n is the steady-state probability of buffer occupancy n. For the case n corresponding to only one state, $P_n = \pi_n$,

for the case n corresponding to two states, P_n is the sum of steady-state probabilities of the two states, *e.g.* $P_{k_1} = \pi_{k_1} + \pi_{k_1+1}$.

- T_i is the time proportion of service rate μ_i.

Specially, k_0 denotes buffer occupancy 0, and $P_{k_{2i}}$ represents the steady-state probability of high buffer occupancy threshold TH_i $(1 \leq i \leq M)$.

State transitions in the left side of cut (A) is similar to an M/M/1 queue, thus

$$P_n = \pi_n = \rho_1^n P_0, 0 \leq n < k_1. \tag{4}$$

Partitioning the chain on cut (A) yields

$$\pi_{k_1} = \rho_1 \pi_{k_1-1} - \frac{\rho_1}{\rho_2}\pi_{k_1+1}. \tag{5}$$

From the balance equations for state transitions between cut (A) and cut (B), we derive

$$\begin{aligned} \pi_{k_1+2j} &= \rho_1^{j+1}\pi_{k_1-1} - \tfrac{\rho_1(1-\rho_1^{j+1})}{\rho_2(1-\rho_1)}\pi_{k_1+1} \quad (0 \leq j < k_2 - k_1), \\ \pi_{k_1+2j+1} &= \tfrac{1-\rho_2^{j+1}}{1-\rho_2}\pi_{k_1+1} \quad (0 \leq j < k_2 - k_1). \end{aligned} \tag{6}$$

Partitioning the chain on cut (B) yields

$$\pi_{k_2} = \rho_2 P_{k_2-1} = \rho_2(\pi_{2k_2-k_1-2} + \pi_{2k_2-k_1-1}). \tag{7}$$

On the other hand, the equilibrium equation of state $(2k_2 - k_1 - 1)$ yields

$$\pi_{k_2} = \frac{1-\rho_2^{k_2-k_1+1}}{1-\rho_2}\pi_{k_1+1}. \tag{8}$$

From (7) and (8), we have

$$\pi_{k_1+1} = \frac{\rho_2(1-\rho_1)\rho_1^{k_2-k_1}}{1-\rho_1^{k_2-k_1+1}}\pi_{k_1-1}. \tag{9}$$

By substituting from (9), (6) could be rewritten as

$$\begin{aligned} \pi_{k_1+2j} &= \tfrac{\rho_1^{k_1+j}-\rho_1^{k_2}}{1-\rho_1^{k_2-k_1+1}}P_0 \quad (0 \leq j < k_2 - k_1), \\ \pi_{k_1+2j+1} &= \tfrac{\rho_1^{k_2-1}(1-\rho_1)\rho_2(1-\rho_2^{j+1})}{(1-\rho_1^{k_2-k_1+1})(1-\rho_2)}P_0 \quad (0 \leq j < k_2 - k_1). \end{aligned} \tag{10}$$

Similarly, the steady-state probabilities of states between k_{2i+1} and k_{2i+2} are

$$\begin{aligned} \pi_{k_{2i+1}+2j} &= \tfrac{\rho_{i+1}^{k_{2i+1}+j}-\rho_{i+1}^{k_{2i+2}}}{1-\rho_{i+1}^{k_{2i+2}-k_{2i+1}+1}}P_{k_{2i}} \\ &\quad (0 \leq j < k_{2i+2} - k_{2i+1}, 0 \leq i < M), \\ \pi_{k_{2i+1}+2j+1} &= \tfrac{\rho_{i+1}^{k_{2i+2}-1}(1-\rho_{i+1})\rho_{i+2}(1-\rho_{i+2}^{j+1})}{(1-\rho_{i+1}^{k_{2i+2}-k_{2i+1}+1})(1-\rho_{2i+2})}P_{k_{2i}} \\ &\quad (0 \leq j < k_{2i+2} - k_{2i+1}, 0 \leq i < M), \end{aligned} \tag{11}$$

where

$$P_{k_{2i+2}} = \frac{(1-\rho_{i+2}^{k_{2i+2}-k_{2i+1}+1})(1-\rho_{i+1})\rho_{i+2}\rho_{i+1}^{k_{2i+2}-1}}{(1-\rho_{i+2})(1-\rho_{i+1}^{k_{2i+2}-k_{2i+1}+1})}P_{k_{2i}}, \quad (0 \leq i < M). \tag{12}$$

The final closed form equations for the steady-state probabilities are

$$P_n = \begin{cases} \rho_{i+1}^{n-k_{2i}}P_{k_{2i}} & (k_{2i} \leq n < k_{2i+1}, 0 \leq i < M), \\ \pi_{k_{2i+1}+2(n-k_{2i+1})} + \pi_{k_{2i+1}+2(n-k_{2i+1})+1} & \\ \qquad (k_{2i+1} \leq n < k_{2i+2}, 0 \leq i < M), & \\ \rho_{M+1}^{n-k_{2M}}P_{k_{2M}} & (n \geq k_{2M}). \end{cases} \tag{13}$$

Since the number of the different service rates is nondeterministic, it is really hard to deduce the analytic expression for P_0. However, the integral of P_n from 0 to infinity equals to 1. By giving a specific value of M, we could derive the approximation of P_0.

The time proportion of service rates could be adopted as a scale to evaluate the energy savings. Specifically, according to (11), the time proportion T_i of service rate μ_i is described by the following expression

$$T_i = \begin{cases} \sum_{n=0}^{k_1-1}\pi_n + \sum_{j=0}^{k_2-k_1-1}\pi_{k_1+2j} & (i=1), \\ \sum_{j=0}^{k_{2i-2}-k_{2i-3}-1}\pi_{k_{2i-3}+2j+1} + \sum_{n=k_{2i-2}}^{k_{2i-1}-1}\pi_n & \\ + \sum_{j=0}^{k_{2i}-k_{2i-1}-1}\pi_{k_{2i-1}+2j} & (2 \leq i \leq M), \\ \sum_{j=0}^{k_{2M}-k_{2M-1}-1}\pi_{k_{2M-1}+2j+1} + \sum_{n=k_{2M}}^{\infty}\pi_n & (i=M+1). \end{cases} \tag{14}$$

Then rate reduction is described as

$$RateReduction = 1 - \sum_{i=1}^{M+1} T_i \frac{\mu_i}{\mu_{M+1}} / \sum_{i=1}^{M+1} T_i. \tag{15}$$

Queuing Delay is described as

$$\begin{aligned} Delay = & \sum_{i=0}^{M-1}\sum_{n=k_{2i}}^{k_{2i+1}-1}\pi_n\tfrac{n}{\mu_{i+1}} + \sum_{n=k_{2M}}^{\infty}\pi_n\tfrac{n}{\mu_{M+1}} \\ & + \sum_{i=0}^{M-1}\sum_{n=k_{2i+1}}^{k_{2i+2}-1} n\left(\tfrac{\pi_{2n-k_{2i+1}}}{\mu_{i+1}} + \tfrac{\pi_{2n-k_{2i+1}+1}}{\mu_{i+2}}\right) \end{aligned} \tag{16}$$

C. Rates and thresholds configuration

To analyze the impact of distribution of rates and thresholds, three kinds of rates and thresholds distribution are taken into consideration for analysis, which are

1) *Uniform rates*, N rates uniformly distributed between $1/N$ of full rate $(1/NGbps)$ to full rate $(1Gbps)$, with $N-1$ high thresholds uniformly distributed, denoted as *URUT-N*.
2) *Expo rates & uniform thresholds*, $(L = log_2 N + 1)$ rates exponentially distributed between $1/NGbps$ to $1Gbps$, with $log_2 N$ high thresholds uniformly distributed, denoted as *ERUT-L*.
3) *Expo rates & expo thresholds*, $(L = log_2 N + 1)$ rates exponentially distributed between $1/NGbps$ to $1Gbps$, with $log_2 N$ high thresholds exponentially distributed as well, denoted as *ERET-L*.

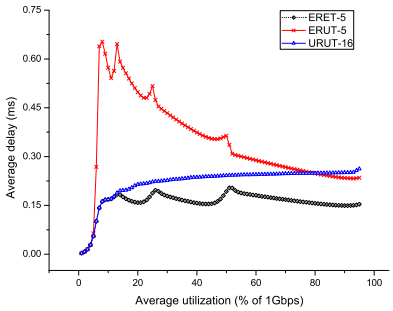


Fig. 3. Packet delay for policy A, when N=16.

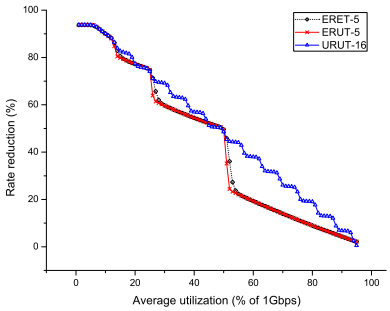


Fig. 4. Rate reduction for policy A, when N=16.

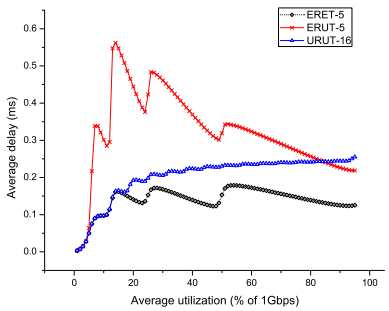


Fig. 5. Packet delay for policy B, when N=16.

The low thresholds all depend on their corresponding high thresholds. Two is used as exponent since it is easiest to be implemented in hardware. The buffer size is configured as 32KB for the following evaluation.

First, we set the low thresholds close to their corresponding high thresholds, namely $TL_i \approx TH_i \& TL_i < TH_i, 1 \leq i \leq M$. This monitors the situation of single-thresholds, and is referd to as policy A. Fig. 3 and Fig. 4 show the average packet delay and rate reduction of policy A when $N = 16$, respectively. ERET-5 has the lowest packet delay among the 3 distributions. The packet delay of ERUT-5 is much higher than the other two distributions especially when the utilization is low. This is due to that the distribution of thresholds of ERUT-5 is not proportional to the distribution of frequencies. For rate reduction, URUT-16 is much more proportional to utilization, while the curves of ERET-5 and ERUT-5 are very close.

Then we consider the impact of difference between the high thresholds and low thresholds on system performance. We set $TL_i \approx TH_{i-1} \& TH_{i-1} < TL_i, 1 < i \leq M$. This monitors the situation that there are large buffers between adjacent rates to reduce the rate switch frequency, and is refered to as policy B. The average packet delay and rate reduction of policy B are presented in Fig. 5 and Fig. 6 respectively. Compared to Fig. 3 and Fig. 4, the packet delays of all three distributions are decreased, while rate reductions of three distributions are all almost proportional to utilization. Since the dual-chains in the Markov model become much longer due to the large difference between high thresholds and low thresholds, system with policy B has more chance working at relative high operational rates than system with policy A. As a result, the average packet delay reduces, which also suggests dual-threshold switching has better performance than single-threshold switching.

Third, we consider the impact of granularity of available frequencies on system performance. Fig. 7 and Fig. 8 show the average packet delay and rate reduction when $N = 8$ using policy B, respectively. Compared to Fig. 5 and Fig. 6, the packet delay of ERUT drops, as well as rate reduction of the three distributions under low utilizations. The reason is that the lowest available frequency becomes higher, that the system would work at higher rate compared to system when $N = 16$. However, the rate reduction is still significant under low utilizations. Thus it is not necessary to set too many levels of available frequencies.

To conclude, the distribution of rates and thresholds are most important to performance. The difference between a low threshold and a high threshold, which form a pair, should be as large as possible. Also, the distribution of thresholds should be proportional to the distribution of frequencies. The lowest available rate or the granularity of rates has influence on power saving under low utilization, but not very much. Thus it is not needed to set many levels of available frequencies. Buffer size influences the average packet delay. Generally, smaller the buffer is, lower packet delay is in average. Among the 3 kinds of distributions, ERET has the lowest average delay, and is very easy to be implemented in hardware. As a result, it is feasible to apply MFS into data path of router for energy conservation. To validate this, we implement a real hardware prototype system based on NetFPGA platform supporting for MFS, which is further discussed in Section V.

V. Implementation

This section presents the implementation details and some engineering choices. We begin by introducing the platform of NetFPGA that we use. Then, we give the detailed description about our implementation of MFS on NetFPGA.

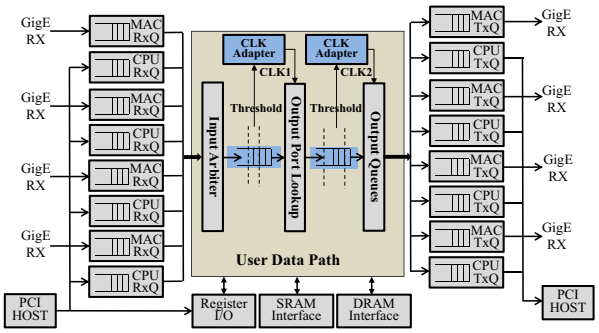


Fig. 9. Reference pipeline of NetFPGA.

A. NetFPGA platform

The NetFPGA is a line-rate, flexible and open platform for gigabit-rate network switching and routing, which enables students and researches to build high-performance networking systems using field-programmable logic array (FPGA) hardware. The core data processing functions are implemented in a modular style, enabling researches to design and build their own functional components independently without modifying original codes [15]. We use NetFPGA-1G card for hardware

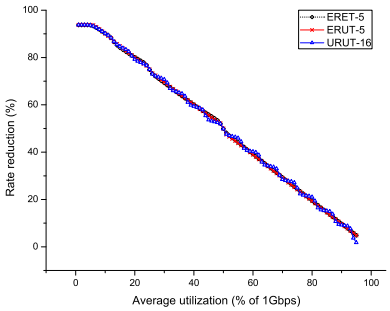


Fig. 6. Packet delay for policy B, when N=16.

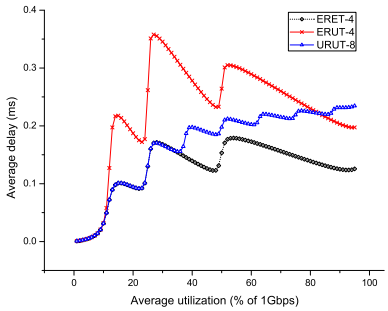


Fig. 7. Packet delay for policy B, when N=8.

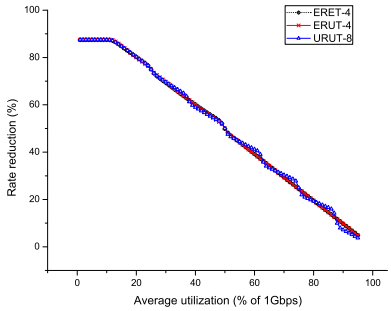


Fig. 8. Packet delay for policy B, when N=8.

prototype, which contains one Xilinx VirtexII-Pro 50 FPGA, The core clock of NetFPGA-1G card could be configured to operate at either 125 MHz or 62.5 MHz. Two SRAMs work synchronously with this core FPGA In our implementation, the core clock rate is set as 125 MHz, which is the default setting of NetFPGA-1G card.

The reference pipeline of NetFPGA-1G consists of receive/transmit queues and the user data path assembled with multiple modules, as shown in Fig. 9. The *CLK Adapter* is our module for frequency tuning. The pipeline in the user data path is 64-bit wide, allowing 8 Gbps peak bandwidth through the data path [15]. It should be noticed that for the 8 receive/transit queues of NetFPGA, 4 are MAC interfaces while the other 4 are CPU-DMA interfaces. Thus, the highest supported incoming traffic rate of NetFGPA is only 4 Gbps, not 8 Gbps. All the internal module interfaces use standard request-grant First-In-First-Out protocol, which works for buffering incoming processing packets. Designers could add and connect their modules to the user data path using FIFOs.

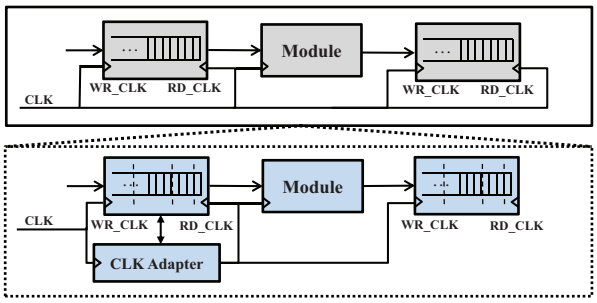


Fig. 10. Adaptive frequency scaling module.

B. Rate Adapter Implementation

The *Rate Adapter* (*i.e.*, *CLK Adapter* in Fig. 10) is our main component for rate adaptation. The CLK Adapter reads FPGA's core clock as input, and generates a range of exponentially distributed frequencies based on the input, by simply using clock cycle counters. The buffer is used for frequency switch decision. Frequency transition is achieved by switching to another frequency according to the input clock, and set it as the output. In our implementation, the frequency switch does not involve much overhead, which suggests frequency switch has negligible impact on packet delay. The output is used for driving the modules connected to the output of buffer, as shown in Fig. 10. To avoid misalignment between input and output clock frequencies, D flip-flop is adopted for buffering the generated output clock. Thus, the actual frequency transition happens one clock cycle after the decision of transition is made.

In our implementation, the CLK Adapters are embedded between *Input Arbiter* and *Output Port Lookup*, and into some sub-modules of *Output Queues*. Six frequencies distribute from 3.096 MHz to 125 MHz in an exponential order of two. The core processing rates are in accord from 250 Mbps to 8 Gbps. Since not all components in NetFPGA support operating at a frequency different to the default value, it is difficult to directly measure the reduction in energy saving of components supporting frequency scaling. Energy conservation could only be estimated using equation (3) in Section 3. However, the rate reduction could be measured by exploiting the *Register System* of NetFPGA. The register interfaces allow software running on host system to send data to and receive data from the hardware modules. A few registers are used for recording operating time of each frequency in our prototype, so that we could estimate average rate reduction. We delay the choices of parameters to the next section.

VI. Experiments

In this section, we investigate the performance of our MFS scheme on NetFPGA and study the results by comparing them with those in Section IV. Our analysis follows the criteria for performance defined in Section III: energy saving and packet delay. In particular, we need to ask the following critical questions: What is the variation of average delay and energy saving in face of different traffic load?

A. Experimental Settings

We have introduced the details of NetFPGA platform used in our experiment in Section V. In particular, we used NetFPGA-1G with core clock rate set as 125MHz, the default value of NetFPGA-1G. To get the energy saving and the cost in delay, we implemented two routers using NetFPGA, one with MFS scheme and one without MFS for reference. We refer to them as MFS-R and R-R respectively. CLK Adapter is used in MFS-R for deriving frequencies from the core clock frequency. The buffer size was set as 8 KB in all experiments. To generate different traffic loads, we used a Spirent SmartBits 600 network performance analyzer [16] which was connected

directly to four 1GE-ports of the NetFPGA. We tuned the incoming traffic from 5% to 95% of 4 Gbps, the highest incoming traffic rate supported by NetFGPA router. Each experiment lasted for 30 seconds which was long enough to feed the system and collect average results (*e.g.* delay and energy saving).To make the results trustworthy, all tests were run for 10 times on both routers to average result.

B. Experiment Results

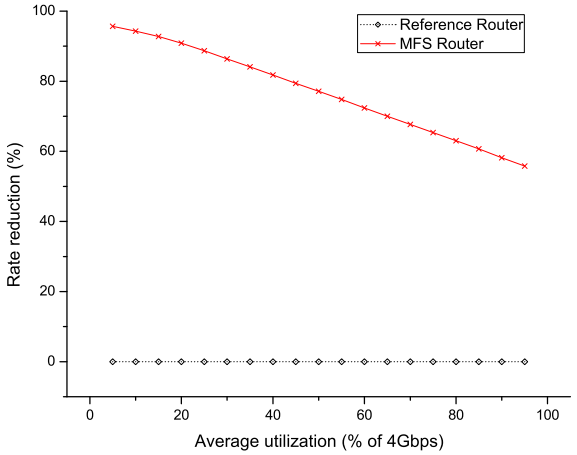


Fig. 11. Rate reduction of R-R and MFS-R.

Fig. 11 presents the average rate reduction of MFS-R and R-R. For R-R, there is no rate reduction. For MFS-R, when the traffic is close to zero, the rate reduction is very close to 100%, suggesting that the components working at very low operating frequencies. However, when the utilization is close to 100%, the rate reduction is still nearly 50%, which is quite different from our analytical result. It is because that the highest supported incoming traffic rate is only 4 Gbps, while the full processing rate is 8 Gbps, as mentioned in Section V.

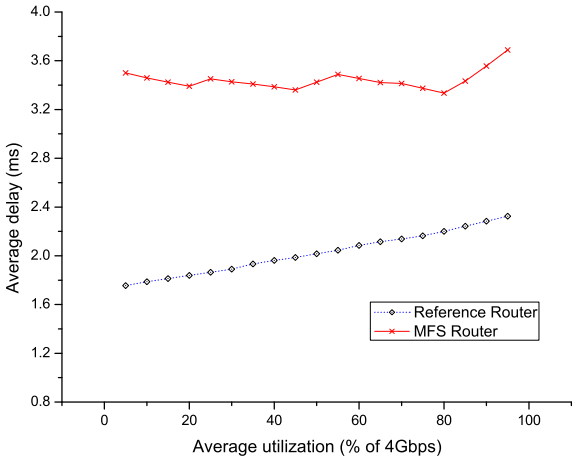


Fig. 12. Average response time of R-R and MFS-R.

Fig. 12 shows the mean response time under different incoming traffic loads. We see that the actual response time is $2ms$ larger than the reference router. This value is quite small compared to the general Internet transmission delay which is at the order of 10 or mostly 100 milliseconds. Since not all the routers on the path are needed to deploy our MFS mechanism, especially those high workload core routers, the total increase in packet delay introduced by MFS should remain within an acceptable level. However, the increase in delay of MFS Router is much larger than the analytical result, even by using smaller buffer and higher operating rate. We believe that there are three possible causes for this. First, the real packets arrivals may not correspond to Poisson distribution exactly. Second, the frequency transition cost is not really negligible, which needs our further investigation. Last, the evaluation of packet delay in Section IV-C is per module while in MFS Router there are several modules supporting MFS, which may be the real reason for the increase in average response time. It shows that with the introduction of rate adaptation into real system, the average response time increases within an acceptable level.

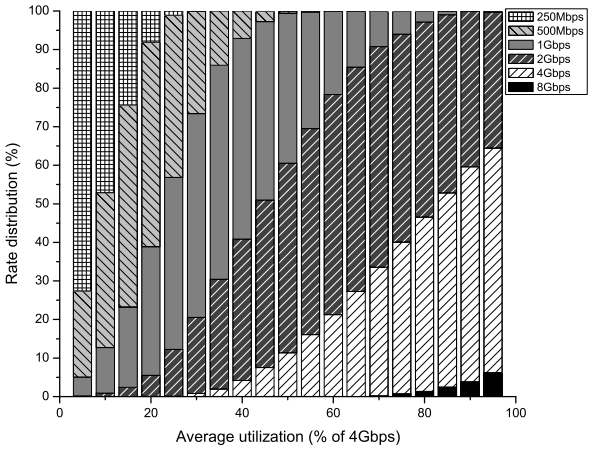


Fig. 13. Rate distribution of MFS Router.

Fig. 13 shows the distribution of the six frequencies under all the traffic loads. It can be observed that with the increase of incoming traffic, the components spend more time working at higher operating frequencies. Thus, MFS mechanism indeed adapts the device's operating rate to its workload, which may result in significant energy saving if the average utilization of the device is very low. Since most network devices are underutilized, significant amount of energy would be saved if these devices support MFS.

The power saving percentages of dynamic portion of energy consumption of modules supporting MFS in our prototype are estimated by using (3), and are presented in Fig. 14. The reason that the minimum saving percentage is around 50% is also because that the full processing rate is 8 Gbps. Since frequency only has influence on dynamic power consumption, the total energy saving may be less than the results presented in Fig. 14. However, the dynamic power consumption is much higher than the static power consumption generally, when there is no rate adaptation in hardware. Thus, MFS would achieve great energy saving in modules supporting it, especially when the average utilization is low. Based on the results presented above, we draw the conclusion that the MFS mechanism is

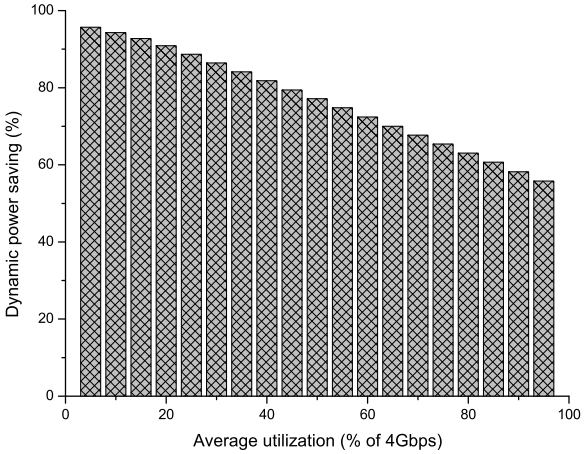


Fig. 14. Dynamic power savings of modules supporting MFS in MFS Router.

feasible and effective for energy conservation in real systems.

VII. Conclusion

In this paper, we have proposed *Multi-Frequency Scaling Scheme*, aiming at energy conservation in data path of routers and switches. Specially, the mechanism employs a Multi-Dual-Threshold policy for frequency switch. The configurations of parameters on system performance are well investigated by using a Markov model. The proposed mechanism is further implemented on the NetFPGA platform for validation with moderate modifications on the reference router. Experiment results indicate that the proposed mechanism effectively cuts off the power consumption of the hardware components inside a router with slight increase in average packet delay. Consequently, MFS is both feasible and effective for energy conservation in real systems.

Acknowledgment

The authors would like to thank the support from NSFC (61073171, 60873250), Tsinghua University Initiative Scientific Research Program, China Postdoctoral Science Foundation (023230012), and the Specialized Research Fund for the Doctoral Program of Higher Education of China (20100002110051).

References

[1] Cisco-Systems, "Approaching the zettabyte era," http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-481374.pdf.

[2] J. Baliga, K. Hinton, and R. Tucker, "Energy consumption of the internet," in *Proceedings of COIN-ACOFT 2007.*

[3] M. Gupta and S. Singh, "Greening of the internet," in *Proc. of ACM SIGCOMM*, 2003.

[4] http://arstechnica.com/old/content/2008/09/what-exaflood-net-backbone-shows-no-signs-of-osteoporosis.ars, 2008.

[5] J. Chabarek, J. Sommers, P. Barford, C. Estan, D. Tsiang, and S. Wright, "Power awareness in network design and routing," in *Proc. of IEEE INFOCOM*, 2008.

[6] M. Gupta, "A feasibility study for power management in lan switches," in *Proceedings of the 12th IEEE International Conference on Network Protocols*, 2004.

[7] M. Gupta and S. Singh, "Dynamic ethernet link shutdown for energy conservation on ethernet links," in *IEEE International Conference on Communications, 2007.*

[8] C. Gunaratne, K. Christensen, B. Nordman, and S. Suen, "Reducing the energy consumption of ethernet with adaptive link rate (alr)," *IEEE Transactions on Computers*, vol. 57, pp. 448–461, 2008.

[9] S. Nedevschi, L. Popa, G. Iannaccone, S. Ratnasamy, and D. Wetherall, "Reducing network energy consumption via sleeping and rate-adaptation," in *Proc. of NSDI, 2008.* Berkeley, CA, USA: USENIX Association, pp. 323–336.

[10] "Netfpga program." [Online]. Available: http://www.netfpga.org/

[11] H. sheng Wang, L. shiuan Peh, and S. Malik, "A power model for routers: Modeling alpha 21364 and infiniband routers," *IEEE MICRO*, vol. 23, no. 1, pp. 26–35, 2003.

[12] A. Wierman, L. L. H. Andrew, and A. Tang, "Power-aware speed scaling in processor sharing systems," in *Proc. of INFOCOM*, 2009.

[13] X. Zhang and K. G. Shin, "E-mili: energy-minimizing idle listening in wireless networks," in *Proc. of MobiCom*, 2011, pp. 205–216.

[14] B. Zhai, D. Blaauw, D. Sylvester, and K. Flautner, "Theoretical and practical limits of dynamic voltage scaling," in *DAC 04: Proceedings of the 41st annual conference on Design automation*, 2004.

[15] J. Lockwood, N. McKeown, G. Watson, G. Gibb, P. Hartke, J. Naous, R. Raghuraman, and J. Luo, "Netfpga–an open platform for gigabit-rate network switching and routing," in *IEEE International Conference on Microelectronic Systems Education, 2007.*

[16] "Spirent smartbits." [Online]. Available: http://www.spirent.com/Solutions-Directory/Smartbits.aspx