Power Scaling in Network Devices

Raffaele Bolla[(a)], Roberto Bruschi[(b)] and Alessandro Carrega[(a)]

[(a)] Department of Communications, Computer
and Systems Scie nce (DIST)
University of Genoa
Via Opera Pia 13, 16145 Genova, Italy
e-mail: {raffaele.bolla, alessandro.carrega}@unige.it

[(b)] National Inter-University Consortium for
Telecommunications (CNIT)
Research Unit of Genoa
Via Opera Pia 13, 16145 Genova, Italy
e-mail: roberto.bruschi@cnit.it

Abstract — ***The largest part of routers and switches, today deployed in production networks, has very limited energy saving capabilities, and substantially requires the same amount of energy both when working at full speed or when being idle. In order to dynamically adapt such energy requirements to the real device work load, current approaches foster the introduction of low power idle and power scaling primitives in entire devices, internal components and network interfaces. Starting from these considerations, we propose an analysis of the theoretical and technological limitations in adopting such kind of mechanisms. The results achieved show that the power scaling allows a linear trade-off between consumption and network performance, but the time to switch between two power states may cause a non negligible service interruption.***

I. Introduction

The energy efficiency issue is assuming ever-greater importance in most industrial sectors and research fields. Several studies suggested that the total energy consumption of electronics in the U.S., in 2006, was more than 70 trillion watt-hours per year (TWh/yr) of electricity, costing billions of dollars, and equivalent to at least 50 million metric tons of carbon dioxide emissions per year [1]. The energy wasted by telecommunication networks represents a non-negligible and continuously increasing share of such consumption.

The most important approaches to energy efficiency optimization in networking devices can be classified in two different categories: (*i*) minimizing the power consumption when no activities are performed (namely “idle” optimizations), and (*ii*) modifying the trade-off between network performance and energy when the hardware is active and performing operations (namely, "power state" optimizations). These two main kinds of power management policies are available in the largest part of COTS (Commercial Off-The-Shelf) processors and under rapid development in other hardware technologies (e.g., network processors, etc…). The IEEE 802.3az [2] task force considered both these mechanisms, but the current standard version includes only idle optimizations, given the technological and theoretical complexity of power scaling approaches. In fact, (*i*) changing frequency and/or voltage in silicon circuits generally requires a longer time period than entering low power idle states; and (*ii*) the adoption of power scaling mechanisms requires more control logic with respect to idle optimizations. However, the quest for power scaling mechanisms is still open and many aspects need to be further analyzed.

II. Power Saving Interface

A PC-based Software Router generally includes support of the Advanced Configuration and Power Interface (ACPI). It provides a well-known interface to support dialogue between the hardware and the software layers in order to hide the heterogeneous power management mechanisms and details of different processors. The ACPI standard abstracts the power-specific management capabilities of processors into two main different power saving mechanisms, namely performance and power states (P-States and C-States, respectively). These two mechanisms are essentially the ones described in the introduction, and P- and C-states correspond to power and idle optimization, respectively.

In this paper, we focus on P-states. The ACPI provides an equivalent frequency that represents the computing capacity of the core/processor given the specific P-state. Hence, the equivalent frequency is an abstraction behind which the operating energy is changed by altering the voltage, or throttling the clock. ACPI allows the tuning of performance through P-states transitions. Unfortunately, due to issues in silicon electrical stability, the transition time between different P-states is generally very slow, especially if compared with usual time scales in network dynamics.

The largest part of current CPUs can switch their performance state in about 2-3ms. Due to these large P-state transition times, any closed-loop policies with tight time constraints are not feasible and cannot be adopted for optimizing power consumption inside network device architectures. In this paper, we focus on the Intel Core i5 processor family that implements the latest power-saving mechanisms. The tests were performed considering the use of the Linux Software Router (SR). The SR architecture provides: (*i*) a number of CPUs/cores per Rx port that guarantee enough computational capacity to process incoming traffic; (*ii*) a number of Tx-Rings per interface equal to the number of CPUs/cores involved in traffic forwarding; and (*iii*) multiple Rx-Rings per high-speed interface (ideally equal to the number of CPUs/cores needed to achieve the maximum theoretical packet rate).

ACM CoNEXT Student Workshop, November 30, Philadelphia, USA.
Copyright 2010 ACM 978-1-4503-0468-9/10/11 ...$10.00.

III. Trade-Off Between Performance and Power Consumption

The evaluations regard the performance and power consumption with the specific hardware introduced in Section II. For each equivalent frequency, tests were run by analyzing the performance of forwarding in terms of throughput with their consumption. Figure 1 reports the results of our tests with CBR (Constant Bit-Rate with packets of 64 bytes) traffic for each different equivalent frequency. With low offered load there are no differences by changing the frequency. Indeed, when the offered load is above 50% the thourghput is aways the same with all the frequencies. Considering the possible maximum throughput, the difference became purposeful passing 50% of the offered load. Figure 2 shows the power consumption with the variation of the offered load according to the same scenario of Figure 1. In detail, these results suggest that it is possible reduce the power consumption by lowering the operating equivalent frequencies. However, we can note how the achievable energy savings mainly depend also on incoming traffic volumes. This is because C-state optimizations were enabled in the i5 processor under test: when the CPU goes idle, it enters the C1 state and saves about 70% of its energy. If the C1 optimization had been disabled, Figure 2 would have shown horizontal lines, representing power requirements of a CPU when active at a certain equivalent frequency. Figure 2 demonstrates that it is possible to obtain energy savings by using only the C-State. If one wishes to obtain more energy savings, it is necessary to integrate these mechanisms with power optimization.

IV. Power Scaling Optimization

As shown in Figures 1 and 2, power optimization can allow obtaining significant power saving without compromising performance, by considering the traffic load dynamics on the Internet, where low utilization time bands can be easily identified. The main difficulties are how and when changing the equivalent frequency. Since the time required for the frequency scaling is non-zero, performance may be adversely affected if the change is done very frequently. In particular, the time spent on frequency scaling consists of two parts: (a) hardware time necessary to change the frequency; (b) time necessary to choose the next frequency. These two delays can affect performance in terms of increased latency. Some cases of frequency scaling are shown in Figure 3.

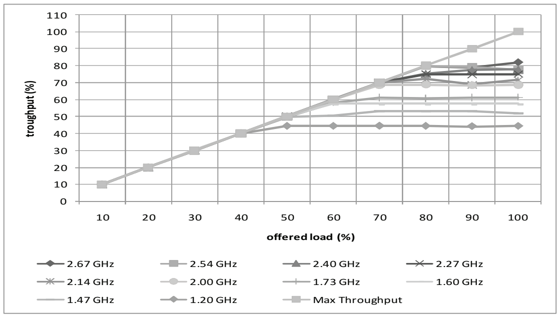

Figure 1. Throughput values according to different equivalent frequencies.

The results are based on the tests done with the proposed architecture and related to the ones of Figures 1 and 2. It is interesting to observe how the latency changes based on what is the equivalent frequency before the scaling. For example, the result of frequency scaling from 2.00 GHz to 2.40 GHz is higher than the opposite. The period of frequency setting is done in the forwarding core and it causes the interruption of the service affecting the network performance. It is possible to reduce the period of service interruption with the multi-queuing network interface. This way, during the period of frequency setting for a specific core/processor, the associated forwarding queue of the NIC is remapped to a different core/processor. The service interruption caused by the frequency scaling is minimized. Packet latency values during the frequency change.

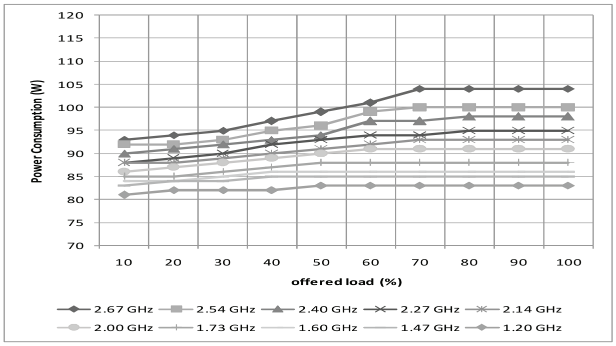

Figure 2. Power consumption values according to different equivalent frequency.

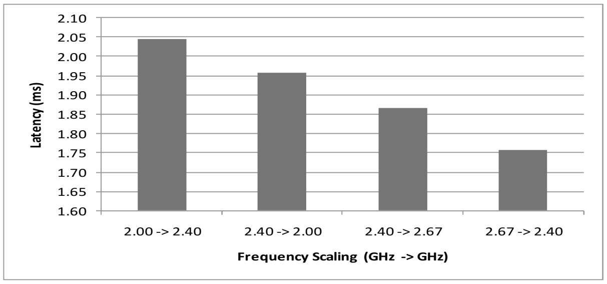

Figure 3. Packet latency values during the frequency change.

V. Conclusions and Future Works

In this short paper, we focused on the theoretical and technological limitations in adopting a power scaling mechanism. In detail, we compute several tests to identify the technological limitations in COTS hardware with power capabilities. The results show that the power consumption grows with the increase of the frequency. Power scaling allows a linear trade-off between consumption and network performance. Future work will aim to extend these studies by including a more detailed analysis about the idle state, and on how it is possible to use power scaling in a device with idle power mechanism without reducing performance.

References

[1] Research News, Berkeley Labs: ”Berkeley Lab Researchers Are Developing Energy-Efficient Digital Network Technology”, URL: http://www.lbl.gov/Science-Articles/Archive/EETD-efficient-networks.html.

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