# Energy-aware Resource Adaptation for Next-Generation Network Equipment

Raffaele Bolla, Roberto Bruschi, and Franco Davoli
Department of Communications, Computer and Systems Science (DIST)
University of Genoa
Via Opera Pia 13, 16145 Genova, Italy
phone: +39 010 353 2075; fax: +39 010 353 2154
e-mail: {raffaele.bolla, roberto.bruschi, franco.davoli}@unige.it

***Abstract*—In this contribution, we explore and try to evaluate the feasibility and the impact of power management policies, able to well suit a heterogeneous set of highly modular architectures, generally used for developing today's network equipment. The proposed policies aim at optimizing the power consumption of each device component with respect to its expected network performance. Finally, in order to provide an experimental evaluation of the proposed ideas, we applied such power management policies to a new generation SW router platform, and we evaluated it with real traffic traces.**

***Keywords- Green networks; Router Power Management; SW Router***

## I. INTRODUCTION

Today, fixed and mobile network infrastructures have enormous and heavily increasing requirements in terms of electrical energy. As shown in [1] and in [2], energy consumption of the Telecom Italia network in 2006 has reached more than 2TWh, increasing by 7.95% with respect to 2005, and by 12.08% to 2004. Another explanatory example is represented by British Telecom, which absorbed about 0.7% of the total UK's energy consumption in the winter of 2007, making it the biggest single power consumer in the nation [3].

As outlined in [4], similar trends can be generalized to a large part of the other telecoms and service providers, since they essentially depend on data traffic volume increase, which appears to follow the Moore's law, and new services being offered. Besides a more widespread sensitivity to ecological issues, the interest on energy efficient networking springs from heavy and critical economical needs, since both energy cost and network electrical requirements show a continuous growth, with an alarming trend over the past years.

It is well known that networks, links and devices are provisioned for busy or rush hour load, which typically exceeds their average utilization by a wide margin. While this margin is generally reached rarely and over short time periods [5], the overall power consumption in today's networks remains more or less constant with respect to different traffic loads. Against such flat energy wastes, the specific challenge for telecoms, network equipment manufacturers and the networking research community nowadays mainly regards the introduction of innovative criteria and technologies, able to save energy by dynamically adapting network capacities and resources to current traffic loads and requirements. Despite some interesting scientific contributions (e.g., [6], [7], [8] among others), green networking performance and optimization remains an open and very interesting issue.

Our ideas are based on the introduction, the exploitation and the control of power management capabilities (i.e., sleeping and rate adaptation) inside architectures and components of network equipment. In this respect, our approach starts by considering the two main kinds of power management hardware support, today available in the largest part of Components Off-The-Shelves (COTS) processors. These power management technologies respectively allow to minimize power consumption when no activities are performed (namely, "idle" optimizations), and to modify the trade-off between performance and energy when the hardware is active and performing operations (namely, "power state" optimizations). These kinds of power management support are generally realized at the hardware layer by powering off sub-components, or by changing the silicon operating frequency and voltage. Specific control applications, namely governors, are needed to dynamically configure such power profiles through the Advanced Configuration and Power Interface (ACPI) standard. In more detail, the specific objective of such SW governors is to optimize the configuration of network devices, in terms of operating frequency and voltage, with respect to their expected performance.

In [9], we evaluated and modeled the impact of power management capabilities on network performance of new generation Linux SW router (SR) platforms, founded on COTS multi-core processors and virtual I/O network interfaces [10]. The obtained results clearly highlight that power management mechanisms introduce a not linear trade-off between maximum forwarding performance and power consumption according to CPU/Cores frequency scaling. In [11], we proposed a preliminary and general optimization policy, which can be included in power management governors, specifically studied in order to exploit the equipment modularity for accurately and separately tuning the tradeoff between power consumption and forwarding performance for each device's component and functionality.

In this paper, we want to move significantly forwards these concepts by evolving the optimization policy in [11] through the exploitation of the empirical results and main results in [9]. In more detail, we will take into account the not linear trade-off between performance and power consumption, by introducing the empirical model obtained for the SR case in [9]. Moreover, in order to validate our approach, the proposed optimization policy has been developed and introduced into power management governors for multi-core SR platforms. Then, a detailed performance analysis has been performed by using real packet-level traffic traces.

The paper is organized as follows. The optimization policy to be used in the power management governor is introduced in Section II, while Sections III and IV show the applied analytical model and the optimization procedure, respectively. Section V reports numerical results. Conclusions are in Section VI.

## II. THE POWER MANAGEMENT POLICY

We assume that the distributed router is composed by $C$ HW components, each one supposed to manage a certain share of the overall forwarded traffic. Moreover, components are assumed to independently switch on different power states, which provide a different tradeoff between performance and power consumption for each HW element. Finally, every element is obviously supposed to increase its maximum forwarding capacity, as a more power intensive state is selected. The main objective of the optimization policy is to minimize the power consumption of a network device, while maintaining a certain performance level. In this respect, the methodology to be used implies the definition of a suitable cost function, to capture the tradeoff between performance and power consumption, which must then be minimized with respect to the operating parameters.

Since the time scales at which the hardware can be switched among different clock frequencies are typically longer than those at the packet- and flow-level, the optimization cannot be realized as a closed-loop control with tight timing constraints. Therefore, we will treat the problem as a parameter-adaptive optimization one, where the expected value of the cost function is periodically minimized over a finite horizon, on the basis of updated information on average values of traffic volumes and requirements.

We suppose to divide a day in different time slices, during which the link traffic loads have an almost similar statistical behavior. During each time slice, the power management governor is thought *i*) to estimate the statistical features of incoming traffic for each device component (e.g., by using the data collected in the same time slice of previous days), and, then, *ii*) to adopt a suitable power configuration for each device element to optimize its trade-off between the expected forwarding performance and energy waste.

In this way, the overall optimal configuration of the modular device is achieved by optimally and individually setting the working frequency values of each HW component $f_c \in F_c$ (where $F_c$ is the set of admissible working frequencies for the element $c$).

We introduce a cost function that represents the overall power consumption of the distributed router, and that can simply be expressed as the sum of power consumptions of the individual elements.

$$\widetilde{\Phi}(f_1, \dots, f_C) = \sum_{c=1}^{C} \widetilde{\Phi}_c(f_c)\, \Phi(f_1, \dots, f_C) = \sum_{c=1}^{C} \Phi_c(f_c) \quad (1)$$

We consider as the cost function of our optimization problem, and we minimize its values in a constrained domain, where the minimum performance bounds we want to assure to forwarded traffic are satisfied. We fix a single set of performance constraints regarding the maximum values of packet loss rates $p_c^*$ for each router component. Given the distributed equipment architecture, these values can be easily used to determine the loss probability of forwarded traffic flows (as will be shown in sub-section III.C). Thus, we can formulate our optimization problem as follows:

$$\begin{cases} \min_{f_1,\dots,f_C} \widetilde{\Phi}(f_1, \dots, f_C) \\ p_c(f_c) \le p_c^* \;\; \forall c \in [1, C] \end{cases} \quad (2)$$

In order to find the optimal router configuration, we have to find the frequency array $\{\hat{f}_1, \dots, \hat{f}_C\}$, which guarantees the minimum value of that respects the performance bounds.

## III. THE ANALYTICAL MODEL

Our approach does not aim to describe equipment architecture and performance in detail, but to provide a generic analytical framework, which well suits in a large set of equipment/ networking scenarios. We provide an overall model composed by two simple sub-models that represent traffic and single components' behavior, respectively. The traffic sub-model represents the traffic offered to the equipment in terms only of average traffic matrix on a per port basis, related variance and maximum deviation. The single components' model takes into account how the forwarding capacity of a component scales according to the working frequencies and traffic loads. Obviously, to correctly apply the overall optimization policy, the equipment architecture and the components' interaction must be given.

### A. Traffic model

We decided to adopt a simple modeling approach that works on data and parameters that can be easily provided to routers. Moreover, owing to the 24 hours' time scale variability of traffic load dynamics on Internet links, in many cases rush hours and low utilization time bands can be easily identified.

Thus, our approach consists in dividing a day in different time slices, in order to collect and to separately estimate traffic loads, and to calculate an optimized router configuration for each one of such time periods. For the sake of simplicity and without loss of generality, we drop the time slice index in the rest of this paper. In detail, we assume to collect periodical samples, for each network interface (or at least for each router component), inside the same time slice of the instantaneous traffic offered load $\lambda_c^{(i)}(t)$. Then, for each network port $i$ of router component $c$, such offered load samples are used to estimate three main statistical parameters, namely: $\lambda_c^{(i)}$, the average value of traffic offered load; $\sigma_c^{2\,(i)}$, the variance of traffic offered load; $\pi_c^{(i)}$, the peak value of traffic offered load[1]. We assume the per port instantaneous traffic loads $\lambda_c^{(i)}(t)$ to be random variables independent among themselves. We can write the average value of overall traffic offered load to the router component *c* as follows:

$$\lambda_c = \sum_{i \in I_c} \lambda_c^{(i)} \tag{3}$$

where $I_c$ is the set of network interfaces of the component *c*.

From the independence assumptions, we can also express the overall average variance of traffic offered load crossing the router element as follows:

$$\sigma_c^2 = \sum_{i \in I_c} \left[\sigma_c{}^{(i)}\right]^2 \tag{4}$$

We additionally define the parameter *M* as the maximum deviation from the average traffic load of flows crossing the component *c*.

### *B. Equipment component model*

Since our main aim is to provide a high level model for network equipment components supporting power saving mechanisms, we only model some basic aspects of the router component, which can easily be adapted to different component HW technologies and architectures.

We assume that each equipment component *c* can work at different internal clock frequencies $f_c \in F_c$, where the set $F_c$ includes a limited number of frequency values at which component HW circuits can correctly work[2]. Each component is supposed to rise its packet processing capacity and its power consumption, as the working frequency value increases. Moreover, we take into account idle optimizations, which allow reduce power consumption when no activities or operations are performed.

The performance and the power saving support of the router element *c* can be completely characterized by considering the following three parameters:

- $\mu_c(f_c)$: the maximum service rate when working at $f_c$;
- $\Phi_{idle}(f_c)$: average power consumption when no activity is performed inside the router element working at $f_c$;
- $\Phi_{active}(f_c)$: average power consumption when the router element *c* performs operation at the clock frequency $f_c$.

Note that these parameters strictly depend on the specific HW implementation of the network equipment component. However, $\mu_c$, $\Phi_{idle}$, and $\Phi_{active}$ are supposed to be monotonic increasing functions with respect to the frequency $f_c$. The limit scenarios, where $\Phi_{idle}$ or $\Phi_{active}$ are constant according to $f_c$, correspond to the lack of "idle" or "power state" management mechanisms, respectively.

Therefore, we can write the steady-state power consumption $\Phi_c$ of a network equipment component, working at $f_c$, as the weighted sum of $\Phi_{idle}$, and $\Phi_{active}$:

$$\Phi_c(f_c) = \tilde{p}_{idle}\Phi_{idle}(f_c) + (1 - \tilde{p}_{idle})\Phi_{active}(f_c) \tag{5}$$

The "idle probability" $\tilde{p}_{idle}$ corresponds to the steady-state probability that the component's packet processing unit is not running any forwarding operations.

As shown in [9], $\tilde{p}_{idle}$ can be related to the router component's utilization, but it does not show a linear behavior with respect to $\lambda_c$ as supposed in [11]. [9] demonstrates that, in the case of energy-aware SW routers, the non-linear behavior of $p_{idle}$ is essentially due to packet handling operations, whose efficiency increases (in a non-linear way) with respect to the traffic offered load.

In more detail, following the empirical model in [9], we can express the instantaneous value of $p_{idle}$ at the time *t* as follows:

$$p_{idle}(\mathrm{t}) = 1 - \chi\left(\frac{\lambda_c(t)[1 - p_c(f_c, t)]}{\mu_c(f_c)}\right) \tag{6}$$

where $\lambda_c(t)$ and $p_c(f_c, t)$ are the instantaneous values of traffic offered load and of loss probability, while $\chi(\cdot)$ is the normalized form function introduced in [9]. Note that in the case of SW router platforms, $\chi(\cdot)$ has been derived as a 7th-order polynomial fitting function:

$$\chi(t) = \sum_{i=0}^{i=7} a_i t^i \tag{7}$$

Assuming that $\lambda_c(t)$ has a normal probability distribution with mean equal to $\lambda_c$, and variance to $\sigma_c^2$, we can obtain the steady-state estimation of $p_{idle}$ as follows:

$$\tilde{p}_{idle} = \mathrm{E}\{p_{idle}(\mathrm{t})\} = \tag{8}$$

$$= 1 - \frac{1}{\sigma_c\sqrt{2\pi}} \int_0^{\lambda_c + M_c} \chi\left(\frac{[1 - p_c(f_c, t)]}{\mu_c(f_c)} t\right) \mathrm{e}^{-\frac{(\mathrm{t} - \lambda_c)^2}{\sigma_c^2}} \mathrm{d}t$$

Since our objective is to characterize the router component performance at a very aggregated level, we assume packet loss to be caused only by a limited packet processing capacity. To this purpose, we decided to not calculate the loss probability (since this will potentially lead us to introduce further assumptions on component architecture). Thus, we provide and work with upper and lower bound approximations for the loss probability, namely $p_c^{max}(f_c)$ and $p_c^{min}(f_c)$, respectively. Thus:

$$p_c^{max}(f_c) \geq p_c(f_c) \geq p_c^{min}(f_c) \tag{9}$$

In order to obtain $p_c^{min}(f_c)$, we consider the packet processing unit as a generic G/G/1/N queuing model, which is suitable for a large set of scenarios, owing to the lack of assumptions on both service and packet inter-arrival time probability distributions. Starting from these assumption we can easily obtain the following relationship:

$$p_c^{min}(f_c) = \max\left\{0, \frac{\lambda_c - \mu_c(f_c)}{\lambda_c}\right\} \tag{10}$$

Regarding the upper-bound approximation of loss probability, we decided to use Bernstein's inequality [12], which belongs to the Chernoff-family theorems that characterize the upper-bound of sums of stochastic variables. In particular, we recall from sub-section III.A that: (*i*) the traffic load offered to the component *c* is the sum of the flows incoming from its I/O interfaces, (*ii*) such flows have a stochastic load with expected average value equal to $\lambda_c^{(i)}$, variance equal to ${\sigma_c^{(i)}}^2$, maximum value equal to $\pi_c^{(i)}$, (*iii*) the

---

[1] The maximum peak rate value is always limited by the bandwidth of the incoming link.

[2] The HW available frequency values are generally a certain multiple of a "base frequency".

instantaneous offered load of each flow has a generic probability distribution. Then, we can apply Bernstein's inequality as follows:

$$P\{\lambda_c(t) \geq \mu_c(f_c)\} \leq e^{-\frac{[\mu_c(f_c)-\lambda_c]^2}{2\sigma_c^2+\frac{2}{3}M_c[\mu_c(f_c)-\lambda_c]}} \tag{11}$$

where $\lambda_c(t)$ is the instantaneous traffic load offered to the router component *c* at the time *t*. Since $P\{\lambda_c{}^{(i)}(t) \geq \mu_c(f_c)\}$ corresponds to the long-term packet loss probability of an unbuffered queueing system (i.e., *N*=0), we can conclude that:

$$p_c(f_c) \leq P\{\lambda_c(t) \geq \mu_c(f_c)\} \tag{12}$$

and, then, we can fix the $\mathrm{p}_c^{\max}(\mathrm{f}_c)$ as follows:

$$p_c(f_c) \leq p_c^{max}(f_c) = e^{-\frac{[\mu_c(f_c)-\lambda_c]^2}{2\sigma_c^2+\frac{2}{3}M_c[\mu_c(f_c)-\lambda_c]}} \tag{13}$$

While the $p_c^{max}$ parameter is useful to estimate the performance constraint in Eq. 2, the $p_c^{min}$ one can be used to find an upper bound for the average power consumption $\Phi_c$. In particular, starting from Eq. 5, we can determine the following upper bound for the idle probability:

$$\tilde{p}_{idle} = 1 + \tag{14}$$

$$-\frac{1}{\sigma_c\sqrt{2\pi}}\left[\int_0^{\mu_c(f_c)} \chi\left(\frac{t}{\mu_c(f_c)}\right) \mathrm{e}^{-\frac{(\mathrm{t}-\lambda_c)^2}{\sigma_c^2}} \mathrm{d}t + \chi(1) \int_{\mu_c(f_c)}^{\lambda_c+M_c} \mathrm{e}^{-\frac{(\mathrm{t}-\lambda_c)^2}{\sigma_c^2}} \mathrm{d}t\right]$$

and, consequently, also for the estimated power consumption:

$$\widetilde{\Phi}_c(f_c) = \Phi_{idle}(f_c) + \tilde{p}_{idle}[\Phi_{active}(f_c) - \Phi_{idle}(f_c)] \tag{15}$$

Finally, in order to solve the optimization problem defined in Section II, we can use Eq. 18 for evaluating $\widetilde{\Phi}_c(f_c)$, and we can approximate $p_c(f_c)$ with its upper-bounds $p_c^{max}(f_c)$.

### C. The equipment architecture

In order to correctly apply the optimization policy for each component, we have to know, and to estimate the statistical features (in terms of $\lambda_c$, $M_c$ and $\sigma_c^2$) of the traffic share incoming to that element. To this purpose, the overall device architecture and how components exchange traffic among themselves must be known. This because if the device's data plane is composed by one or more components working in series, the loss probability of the first components in the chain may affect the $\lambda_c$ (and, consequently, the $M_c$ and $\sigma_c^2$) of the last ones.

This is certainly the case of architecture based on switching matrix and line-cards (usually adopted by high-end commercial equipment). In fact, each line-card can be thought at least as a single device component, and the forwarding process for a single packet usually involves two line-cards (i.e., the reception and the transmission ones). In such scenario, the solution to be used in order to correctly evaluate the $\lambda_c$ parameters consists in a simple recursive optimization procedure, which is diffusely adopted in "multi-stage" architectures. For this reason and for the paper's space limit, we do not deepen such procedure here.

New generation SW routers, founded on multi-core processors and COTS hardware [12], deploy a rather different architecture with respect to their commercial cousins. In more detail, as outlined in [9] and [10], each Core included in a SW router can be considered as an independent component that entirely process a certain share of the incoming traffic (i.e., the traffic incoming from the network interfaces bounded to that Core). In this respect, a SW router can be thought like a set of Cores working in a parallel and fully independent manner. In such case, the $\lambda_c$ of a component will not be affected by other ones, and no recursive optimization procedure is needed. Therefore, we can decompose our optimization problem on a per-component basis:

$$\min_{f_1,\dots,f_C} \Phi(f_1,\dots,f_C) = \sum_{c=1}^{C} \min_{f_c} \Phi_c(f_c) \tag{16}$$

## IV. MINIMIZATION PROCEDURE

For both distributed architectures introduced in sub-section 3.3, the core step of the procedure is the power consumption minimization of each single component.

Thus, in order to find the optimal component configuration, we select the sub-set $\tilde{F}_c$ of working frequencies $f_c$ that respect the performance constraint in Eq. 2. To determine $\tilde{F}_c$, we exploit Eq. 12 as follows:

$$\tilde{F}_c = \{f_c \in F_c\colon\ p_c^{max}(f_c) < p_c^*\} \tag{17}$$

To this purpose, by inverting Eq. 12, and by substituting the $p_c^{max}$ with the loss constraint $p_c^*$, we can easily obtain the minimum capacity $\mu_c^*$ that assures the constraint fulfillment:

$$\mu_c^* = \lambda_c - \frac{1}{3}M_c\ln(p_c^*) + \frac{1}{3}\sqrt{M_c{}^2\ln^2(p_c^*) - 18\sigma_c^2} \tag{18}$$

Exploiting Eq. 21, we can select the $\tilde{F}_c$ set as:

$$\tilde{F}_c = \{f_c \in F_c\colon \mu_c(f_c) < \mu_c^*\} \tag{19}$$

where the $\mu_c(f_c)$ are supposed to be known from the router component data-sheet, and $\mu_c^*$ can be obtained from Eq. 21.

Thus, in general, we have to find the optimal frequency $f'_c$, which guarantees the minimum value of $\widetilde{\Phi}_c(f_c)$, by numerically evaluating Eq. 14 for $\forall f_c \in \tilde{F}_c$. This corresponds in solving the minimization problem through a "brute force" approach (exhaustive search). However, since the number of working frequencies $|F_c|$ is generally very low (i.e., it does not exceed 10 values in the largest part of HW components and technologies), and since the proposed model is characterized by a very low computational complexity, finding the minimum is feasible even by performing an exhaustive search.

However, in the specific case of SR platforms, as shown in [9], the $\chi(\cdot)$ function as well as the $\Phi_{idle}(f_c)$ and $\Phi_{active}(f_c)$ values are so, that the minimum power consumption always corresponds to the minimum $f_c$ value in $\tilde{F}_c$. Therefore, in the specific case of SR platforms, no exhaustive search of the optimal configuration is needed.

## V. NUMERICAL RESULTS

In order to evaluate the proposed optimization mechanism, we decided to use the multi-Core SW router architecture already adopted in [12]. In detail, the used HW platform includes two dual core Xeon processors. The choice of validating the proposed optimization policy on a SW router platform was simply driven by the fact that only such king of HW platforms already include power management capability.

Each core can independently work at four operating frequencies, namely 3.0, 2.667, 2.333 and 2.0 GHz. Table I shows $\Phi_{idle}$, $\Phi_{active}$ and $\mu_c$ values at different available $f_c$ referred to a single Core the selected HW platform. The SW router includes 4 Gigabit Ethernet interfaces. Each Core processes the packets incoming from a single Gigabit port.

TABLE I. POWER CONSUMPTION AND SERVICE RATE PER CORE AND $f_c$

| $f_c$ [MHz] | $\Phi_{idle}$ [W] | $\Phi_{active}$ [W] | $\mu_c$ [kpps] |
|---|---|---|---|
| 3000 | 37.5 | 82.75 | 1010 |
| 2667 | 29.0 | 66.0 | 890 |
| 2333 | 21.5 | 50.5 | 780 |
| 2000 | 13.5 | 38.0 | 670 |

Regarding the traffic load generation, we used the Agilent N2X router tester to emulate the traffic offered to each ingress link.

Regarding the $\chi(\cdot)$ function, we adopted the one proposed in [9], which was obtained empirically by measuring the performance and the network performance also of the selected HW platform, and which has been introduced in Eq. 7. The $a_i$ parameters have been configured as follows: $a_0$=0, $a_1$=0.47, $a_2$ =15.01, $a_3$ =-89.86, $a_4$ =223.07, $a_5$ =-256.47, $a_6$ =130.47, $a_7$=-21.70.

The rest of this section is devoted to validate the model adopted inside our optimization approach, and to analyze the overall energy savings that the optimization procedure can obtain.

### *A. Model Validation*

In this section, we perform different tests with the aim of validating the model that we adopted for capturing the device behavior, and the trade-off between network- and energy-aware performance.

Regarding the offered traffic load, we assume to have a fixed $\sigma_c^2$ value equal to 400, while $M_c$ depends on the offered load $\lambda_c$: $M_c = 200 - \frac{2}{100}\lambda_c$.

In detail, $M_c$ takes into account that, when the average value of offered load rises, the maximum deviation (in term of maximum gap between the peak and the mean traffic load) obviously decreases. Under such assumptions, we calculated from Eq. 14 the upper-bound of the cost function $\widetilde{\Phi}_c(f_c)$.

Fig. 1 reports the estimated $\widetilde{\Phi}_c$ values only for all the operating frequencies, and outlines how the maximum power consumption is achieved and kept constant when processing capacity is saturated (i.e., $\forall\lambda_c \geq \mu_c(f_c) \rightarrow p_{idle}$=0).

Observing Fig. 1, we can underline how a rise in operating frequency corresponds to larger power consumptions for all the traffic offered loads.

However, this effect fundamentally depends on the $\chi(\cdot)$ function, as well as on how the $\Phi_{idle}$, the $\Phi_{active}$ values change according to $f_c$, and it cannot be generalized to all the ACPI-capable platforms. This because, in general, once fixed the traffic load $\lambda_c$, when the operating frequency decreases, the router component must spend larger time periods in the "active" state in order to process all the traffic packets. On the contrary, when a higher frequency is selected, the $p_{idle}$ value rises and the component remains in idle state for longer periods. Therefore, this outlines how, under certain conditions, selecting a higher operating frequency may lead to a lower energy waste, since it allows better exploiting "idle" power management mechanisms.

Fig. 2 reports the percentage error between the estimated power consumptions $\widetilde{\Phi}_c(f_c)$ and the ones measured, when Constant Bit Rate (CBR) traffic is offered to the SR. The shown error values underline the good level of accuracy provided by the model (the maximum error is lower than 0.05%).

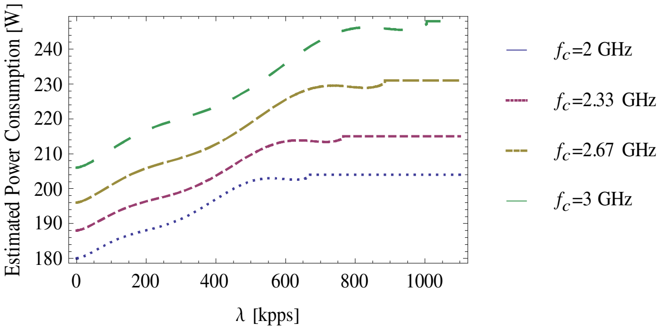

Figure 1. Average power consumption according to different values of operating frequency and traffic offered load.

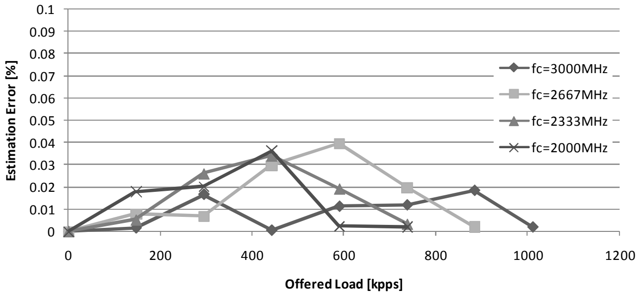

Figure 2. Estimation error between Average power consumption according to different values of operating frequency and traffic offered load.

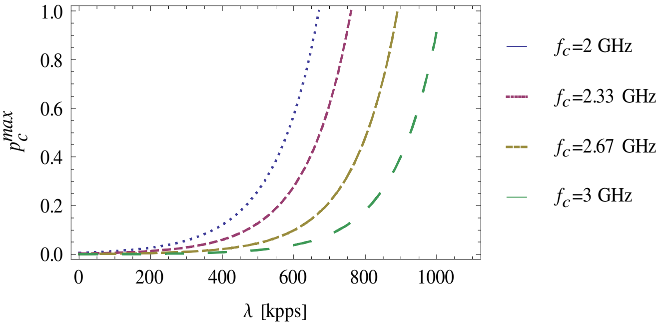

Figure 3. Loss probability values estimated with Eq. 12 with respect to different frequencies and traffic loads.

Regarding the loss probability constraints, Fig. 3 shows the loss probability values estimated with Eq. 12 for each available frequency and according to different traffic loads, and underlines how Eq. 13 provides a very conservative upper-bound for $p_c(f_c)$. Measured packet loss probabilities are much lower than the ones in Fig. 3.

### *B. Performance evaluation of the optimization algorithm*

In order to thoroughly emulate such offered load, we exploited statistical features collected from real traffic traces, captured by monitoring the GARR network [14] during a 30 day period. The optimization framework works on 24 daily time slices, each one with duration equal to 1 hour. Fig. 4 shows the average values, the deviation, and some "real" samples of traffic load offered to a single Gigabit link, calculated on a per hour basis.

Finally, by considering all the 4 cores in the SW router and related traffic load statistics, Fig. 5 summarizes the energy saving obtained with respect to a case without power management mechanisms (where power consumption is always 310 W). Such values were reported both for the power consumptions obtained with the proposed model, and for the ones really measured by emulating the SW router behavior for 10 days.

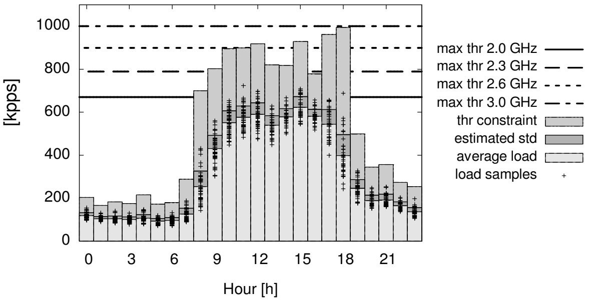

Figure 4. Offered load daily statistics (in terms of traffic volume samples, average value and deviation), minimum throughput respecting the loss probability constraint and maximum capacities at the available frequencies.

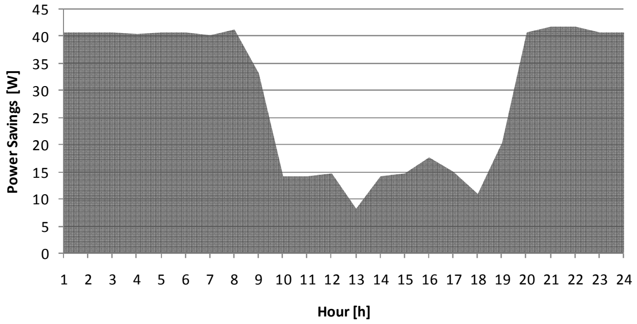

Figure 5. Measured power saving obtained with the proposed framework with respect both to an equivalent SW router without power management mechanisms.

Observing Fig. 5, which reports the measured power savings provided by our framework with respect to the same SR router platform without ACPI optimization, we can note how the proposed framework allows to effectively reduce the overall power wasting of the SW Router (on a per day basis, we obtain a saving of about 25%-30%).

## VI. CONCLUSIONS

In this contribution, we demonstrated the feasibility and showed an explanatory example of green equipment for next generation networks. In more detail, we introduced and evaluated a governor policy that can effectively be adopted to dynamically optimize power consumption of a modular network device with respect to its expected forwarding performance. We showed that the proposed approach suits properly different equipment architectures (especially multi-core SW routers). The benchmarking results, obtained with a multi-core COTS SW router and real link traffic statistics, point out that the proposed optimization mechanism provides interesting levels of power saving (about 25%-30% with respect to the case of no ACPI mechanisms).

## REFERENCES

[1] C. Bianco, F. Cucchietti, G. Griffa, "Energy consumption trends in the next generation access network - a telco perspective," *Proc. 29th Int. Telecomm. Energy Conf (INTELEC 2007)*, Rome, Italy, Sept. 2007, pp.737-742.

[2] Telecom Italia Website, "The Environment", URL: http:// www.telecomitalia.it/sostenibilita2006/English/B05.html

[3] BT Press, "BT announces major wind power plans," Oct. 2007, http://www.btplc.com/News/Articles/Showarticle.cfm? ArticleID=dd615e9c-71ad-4daa-951a-55651baae5bb.

[4] ITWales, "Green evangelist to call for big changes in computer use to aid environment at ITWales conference," Nov 2007, http://www.itwales.com/997539.htm.

[5] Sprint IP Monitoring Project. URL: http://ipmon.sprint.com/.

[6] Noguera, J.; Kennedy, I.O., "Power Reduction in Network Equipment through Adaptive Partial Reconfiguration," *Proc. of the 2007 Int. Conf. on Field Programmable Logic and Applications (FPL 2007)*, Aug. 2007, pp. 240-245.

[7] M. Gupta, S. Singh, "Using Low-Power Modes for Energy Conservation in Ethernet LANs," *Proc. of the 26th Annual IEEE Conf. on Computer Communications (IEEE INFOCOM 2007)*, Anchorage, Alaska, USA, May 2007.

[8] S. Nedevschi, L. Popa, G. Iannaccone, D. Wetherall, S. Ratnasamy, "Reducing network energy consumption via sleeping and rate-adaptation," *Proc. 5th USENIX Symp. on Networked Systems Design and Implementation*, San Francisco, CA, 2008, pp. 323-336.

[9] R. Bolla, R. Bruschi, A. Ranieri, "Green Support for PC-based Software Router: Performance Evaluation and Modeling", Proc. *2009 IEEE International Conference on Communications* (ICC 2009), Dresden, Germany, June 2009. Winner of the best paper award.

[10] R. Bolla, R. Bruschi, "PC-based Software Routers: High Performance and Application Service Support," *Proc. of the ACM Sigcomm Workshop on Programmable Routers for Extensible Services of Tomorrow* (PRESTO'08), Seattle, WA, USA, Aug. 2008, pp. 27-32.

[11] R. Bolla, R. Bruschi, F. Davoli, "Energy-aware Performance Optimization for Next Generation Green Network Equipment", *Proc. of the ACM SIGCOMM 2009 Workshop on Programmable Routers for Extensible Services of Tomorrow* (PRESTO 2009), Barcelona, Spain, Aug. 2009.

[12] S.N.Bernstein, "*On several modifications of Chebyshev's inequality*", vol. 4,22, Doklady Akad, Nauk SSSR, 17, n. 6 (1937), pp. 275-277.

[13] K. Argyraki, *et al*, "Can Software Router Scale?," *Proc. of the ACM Sigcomm Workshop on Programmable Routers for Extensible Services of Tomorrow* (PRESTO'08), Seattle, WA, USA, Aug. 2008, pp. 21-26.

[14] The GARR NoC, https://www.noc.garr.it/ GINS/svg/weathermap.php