Modeling Temperature and Dissipation Behavior of an Open Multi-Frequency Green Router

[1]Alfio Lombardo, [2]Diego Reforgiato,[2]Vincenzo Riccobene, [1]Giovanni Schembra
[1]Dipartimento di Ingegneria Elettrica, Elettronica e Informatica (DIEEI)-University of Catania
[2]Lightcomm s.r.l.
{lombardo, diegoref,vincenzo.riccobene, schembra}@dieei.unict.it

Abstract— **Development of green routers aimed at saving energy when the input traffic is low has the positive side effect of reducing the working temperature of their internal components. This idea can encourage the realization of smaller scale routers, but this process has to be supported by the confidence that temperature remains in a given range. For this reason, the target of this paper is to propose an analytical discrete-time Markov model that allows green router designers to calculate in advance the temperature statistics of a router, for given input traffic statistics and energy saving law applied by the router Governor. At the same time the model evaluates the obtained energy saving gain. The proposed model is applied to a case study to show how it can be used to design the router Governor parameters to achieve the target of maintaining the mean temperature below a given threshold.**

Keywords-component; Temperature, Markov model, NetFPGA, Governor, Power Consumption

I. INTRODUCTION

Today networks are provisioned for worst-case or busy-hour load, and this load typically exceeds their long-term utilization by a wide margin; moreover, as shown in [1], current network nodes have a power consumption that is practically constant and does not depend on the actual traffic load they face. The implication of these factors is that most of the energy consumed in networks today is wasted [2-3].

A non-marginal side effect of high energy dissipation is the increment of the temperature of the ambient where network devices reside, with a consequent further waste of energy used by cooling machines to maintain the temperature of the local environment constant.

The steadily rising energy cost and the need to reduce the global greenhouse gas emission make this occurrence unsustainable: today, in fact, 37% of the total ICT emissions are due to telecommunications companies infrastructures and devices [4]. For this reason, addressing energy efficiency challenges in wireline networks is receiving considerable attention in the literature today [5-6]; moreover a number of research projects has been started on this topic (see for example [7-9]). Thus, some novel hardware devices, so-called "green routers", are expected in the near future to allow to enter different power states [10] according to the input traffic. Two approaches have been proposed to reduce the energy consumption in network components [6]. The first is based on putting network components to sleep during idle intervals, reducing energy consumed in the absence of traffic. The second is based on adapting the rate of network operations to the offered workload, reducing the energy consumed when actively processing packets. Rate adaptation in particular is usually achieved by scaling the processing power according to the data rate the router has to manage; at this purpose the clock frequency driving the router processes can be modified according to the input data rate [11]. The energy aware technique to be used in a green router depends on a number of parameters, including the role of the router in the network, the incoming traffic profile, the hardware complexity and the related costs with respect to the energy we can potentially save and the QoS we want to guarantee to the users [12].

Now, let us note that the introduction of green management techniques to make network routers green has an important consequence on the decreasing of working temperature of the device hardware. As known, temperature is one of the major factors which must be considered and addressed in the design and the manufacture of electronic devices, and specifically routers, since operating at higher temperature degrades system reliability, causes performance degradation and leads to higher cooling and packaging costs [13].

Moreover, "smaller and faster" are the chief demands driving today's electronic designs. These issues translate into high power densities, higher operating temperatures and lower circuit reliability. Therefore, greening a router can be considered as a leveraging approach to move towards this direction. In other words, reduction of the average temperature in green routers due to the application of algorithms aimed at reducing energy consumption will allow designers to modify hardware, reducing its size and the size of the passive and active cooling systems, since a package designed for the worst case is excessive. However, the above hardware modifications can make again router circuits heat beyond their designed thermal limits. For this reason, the working range of temperature becomes again an important issue in green router design.

With all this in mind, with the aim of optimizing green policies coupled with thermal management techniques while respecting requirements on both temperature, QoS and energy consumption, an analytical model is introduced. This model allows the designer to determine in advance if the device will operate within recommended thermal ranges when the green router governor uses a given energy saving policy. In addition,

the same model can be used to evaluate the achieved amount of energy saving.

The proposed model is a multi-dimensional discrete-time Markov model. A case study based on the open NetFPGA Reference router [14] is considered to show how the proposed model can be easily applied to a real case. At this purpose we modified the NetFPGA Reference Router leveraging on the facility of the NetFPGA platform that allows designers to reduce the clock rate from 125 to 62.5 MHz by changing the value of an ad-hoc hardware register.

The paper is structured as follows. Section II introduces the NetFPGA Reference Router architecture. Section III shows the measurements that we have conducted on both temperature and energy consumption in the modified version of the reference router. Section IV describes the Markov model that we have defined. All the results of our analysis are shown in Section V. Finally, Section VI ends the paper with authors' conclusions and future directions.

II. NETFPGA ROUTER ARCHITECTURE

The NetFPGA [8] is an accelerated network hardware that augments the functions of a standard computer. A user-programmable FPGA (with two PowerPC processors) is hosted on the board together with SRAM, DRAM, and four 1 Gbps Ethernet ports[1].The FPGA directly handles all data-path switching, routing, and processing operations of Ethernet frames and Internet packets, leaving software to handle control-path functions only [15].

The open router we consider in this paper is the reference router [14] implemented on the NetFPGA platform (as it is in the version 1Gbps). The reference pipeline consists of the user data path, eight receive queues and eight transmit queues (each port has a CPU and a MAC queue).

It is possible to change the NetFPGA clock rate from 125 MHz to 62.5 MHz and vice versa by changing the value of the CPCI_CNET_CLK_SEL_REG hardware register. This clock frequency variation affects only a part of the NetFPGA board. The physical ports still run at 125 MHz. However, there are shallow (2KB) FIFOs between the two clock domains that manage the clock domain crossing. This clock variation facility constitutes the base of our work.

As stated in [16], when the reference router works at 62.5 MHz, it is able to forward traffic without loss only when the bitrate is less than or equal to 2 Gbps. The switching we have developed so far is performed by the host computer accessing the hardware register via software. For this reason, during each frequency switching interval, the NetFPGA will lose all the incoming packets. We are currently working on the hardware version of the frequency switch where the use of some buffer and an adequate policy will prevent packet loss during switch; anyway, the reader notices that our analytical model is general and does not depend on whether the frequency switch is realized in hardware or software.

[1] 10 Gbps Ethernet ports are already available in the newer version.

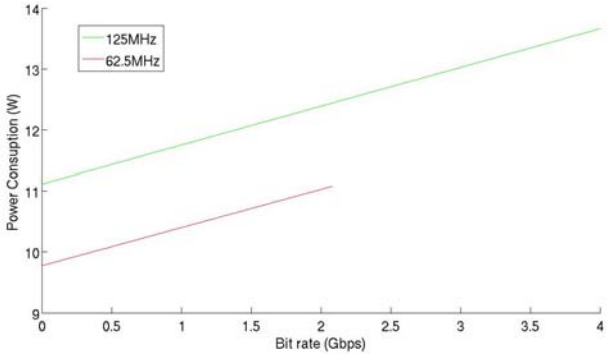

Figure 1. Power consumption model for NetFPGA platform

In order to decrease energy consumption, but at the same time control the QoS maintaining it acceptable, we introduced a Governor, an entity with the aim of changing the clock frequency of the CPU to avoid waste of energy when the input traffic is less than 2 Gbps. To this purpose, it receives the run-time value of the input bit-rate by the Traffic Monitor module, and decides the CPU clock frequency according to the power management policy described below. This policy is defined taking into account that a clock frequency variation, as said so far, causes loss of traffic entering the router during the switching interval. So the policy is defined as follows:

- if the clock frequency was previously set to 62.5 MHz and the current input bit rate is greater than 2 Gbps, then the clock frequency is switched to 125 MHz, in order to avoid losses due to the low processing power;
- if the clock frequency was previously set to 125 MHz and the current input bit rate is lower than 2 Gbps, then the clock frequency is switched to 62.5 MHz with a probability $p_G(B_{IN})$, which is function of the current input bit rate. Of course, the greater the distance between B_{IN} and the threshold of 2 Gbps ($B_{IN} < 2$ Gbps), the less the risk of a needed new frequency switch to set the clock frequency to 125 MHz again, and therefore the higher the value we can use for $p_G(B_{IN})$. On the contrary, the less the value of $p_G(B_{IN})$, the rarer the use of the clock frequency 62.5 MHz, and then the less the energy saving.

Of course, the probability term $p_G(B_{IN})$ plays a very important role in the router performance in terms of both QoS, temperature and energy saving. The design of this function will be assisted by the analytical model that will be described in Section IV.

III. TEMPERATURE AND POWER CONSUMPTION MEASUREMENT

In this section we extend measurements presented in [17] to calculate the power consumed by the NetFPGA Reference Router when both 125 and 62.5 MHz are used as clock frequencies. Power consumption has been evaluated using the following configuration: the Reference Router project was loaded on the NetFPGA card along with SCONE (the NetFPGA control software). Ultraview PCI Smart Extender

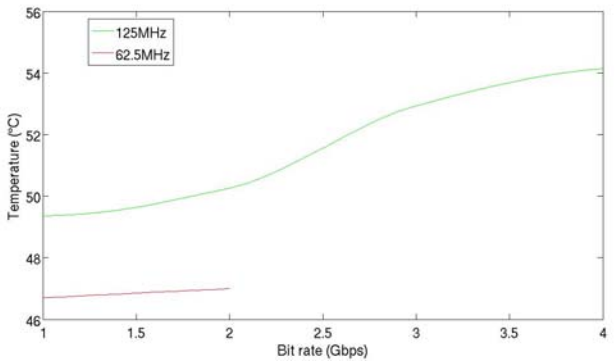

Figure 2. Steady values of temperature for different values of the bitrate.

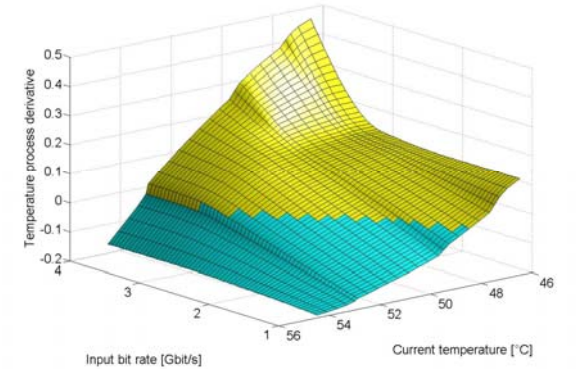

Figure 3. Temperature process derivative within 125 MHz

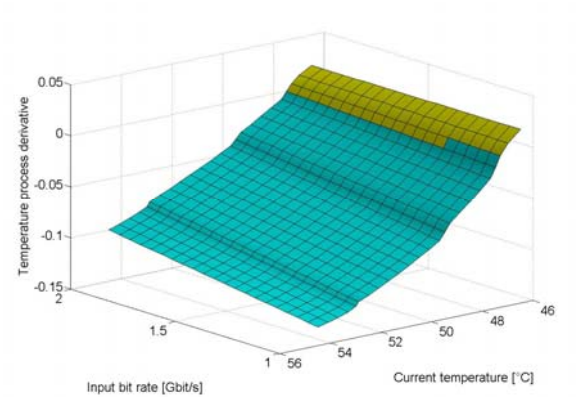

Figure 4. Temperature process derivative within 62.5 MHz

PCIEXT-64UB card has been used to read the power consumption of the NetFPGA card, by isolating it from the consumption of the NetFPGA host computer. An Agilent MSO7054A series 7000 InfiniiVision oscilloscope has been used for power measurements. Finally, we used the IXIA, an high-precision commercial-grade hardware traffic generator with sophisticated capabilities for configuring traffic profiles, synchronizing traffic on multiple ports, and accumulating statistics based on pattern filters. In Fig. 1 we show the power consumption model which characterizes the current NetFPGA Reference Router varying the input bit rate and for the two considered clock frequencies [16]. Specifically, when the bit rate is lower than or equal to 2 Gbps (in such a case the clock frequency 62.5 MHz may be used), the consumed power is less than 11 W; for bit rates higher than 2 Gbps, the clock frequency of 125 MHz has to be used, and the power consumption goes up to 14 W. More details about such measurements produced by our analysis can be found in [16].

To measure the temperature, we have used a thermal diode. The temperature we have measured depends on both the network traffic processed and the clock frequency. We have calculated, for each value of the clock frequency and for different values of the input bit rate, the steady temperature, shown in Fig. 2. Moreover, for each pair (bit rate *b*, temperature *t*), we measured the derivative of the temporal temperature behavior to reach the steady temperature value of *b* from *t*. Figs. 3 and 4 show such values when the clock frequency is 62.5 and 125 MHz, respectively.

According to the steady temperature value shown in Fig. 2, the slope could be either positive (if the starting temperature *t* is lower than the steady value) or negative (if the starting temperature is higher than the steady value). For example, if the starting temperature is 46.7 °C and the bit rate is of 4 Gbps, then the current temperature will have to grow up to the steady value of the state with bit rate equal to 4 Gbps, i.e. 54.14°C. Conversely, in the same clock frequency and bitrate conditions, if the starting temperature is 55 °C, then the current temperature will have to decrease in order to reach the same steady temperature value.

IV. MARKOV MODEL

In this section we define a discrete-time model of the temperature behavior of the CPU residing on the considered router. Since it depends on both the clock frequency and the input traffic bit rate, we use a 4-dimensional Markov model whose state is $S^{(\Sigma)}(n) = \left(S^{(C)}(n), S^{(I)}(n), S^{(T)}(n), S^{(S)}(n)\right)$, where $S^{(C)}(n) \in \Im^{(C)}$ is the clock frequency process at the generic slot *n*; $S^{(I)}(n) \in \Im^{(I)}$ represents the quantized input traffic bit rate at the generic slot *n*; $S^{(T)}(n) \in \Im^{(T)}$ is the temperature at the generic slot *n*; $S^{(S)}(n) \in \Im^{(S)}$ is the indicator variable of a switch at the generic slot *n*: $S^{(S)}(n) = 1$ if a clock switch occurred in the slot *n*. The sets $\Im^{(C)}$, $\Im^{(I)}$, $\Im^{(T)}$ and $\Im^{(S)}$ are the state spaces for the four above state variables.

In the NetFPGA router case, $\Im^{(C)} = \{62,125\}$ MHz; the set $\Im^{(S)}$ is, by definition, $\Im^{(S)} = \{0,1\}$. The set $\Im^{(I)}$ contains the considered quantized input traffic values (in the case study described in Section V $\Im^{(I)} = \{0.5,1.0,1.5,2.0,2.5,3.0,3.5,4.0\}$ Gbits/s). The set $\Im^{(T)}$ is constituted by evenly spaced values ranging from the minimum to the maximum values the temperature can assume when the router works. These values are chosen according to the measures presented in Section III.

Now, in order to define the system model, we have to decide the slot duration and the time diagram of each slot. As far as the slot duration is concerned, we use the duration of a clock frequency switch, and we will indicate it as Δ. In order to

define the model time diagram, we consider two generic states: $s_{\Sigma 1} = (s_{C1}, s_{I1}, s_{T1}, s_{S1})$ in the slot n, and $s_{\Sigma 2} = (s_{C2}, s_{I2}, s_{T2}, s_{S2})$ in the slot $n+1$. We assume the following event sequence:

1. The first action at the beginning of the slot $n+1$ is the evaluation of the new value of the input traffic bit rate. We assume that this new value will be maintained for the whole duration of the slot; this assumption is realistic if a buffer is used to compensate traffic fluctuations during the slot, if any.
2. Then, according to the new value of the input traffic bit rate, the Governor decides the clock frequency for the new slot. As discussed so far, if a clock frequency modification occurs, since a slot interval is needed to apply the clock frequency switch, the new clock frequency will be actually operative from the slot $n+2$; during the slot $n+1$ the router is frozen and all the input traffic arriving in this slot is lost.
3. Then, at the end of the slot $n+1$, the system state variables are observed.

Now we can define the generic element of the state transition probability matrix as follows:

$$Q^{(\Sigma)}_{[s_{\Sigma 1}, s_{\Sigma 2}]} = \text{Prob}\left\{S^{(\Sigma)}(n+1) = s_{\Sigma 2} \middle| S^{(\Sigma)}(n) = s_{\Sigma 1}\right\} = \\ = Q^{(I)}_{[s_{I1}, s_{I2}]} \cdot \eta^{(C)}_{[s_{C1}, s_{C2}]}(s_{I2}) \cdot Q^{(T)}_{[s_{T1}, s_{T2}]}(s_{C2}, s_{I2}) \cdot \vartheta^{(S)}_{[s_{S2}]}(s_{C1}, s_{C2}) \quad (1)$$

where:

- $\vartheta^{(S)}_{[s_{C1}, s_{C2}]}$ is a clock switch indicator function. It is defined as follows:

$$\vartheta^{(S)}_{[s_{S2}]}(s_{C1}, s_{C2}) = \begin{cases} 1 & \text{if } (s_{C2} \neq s_{C1} \text{ and } s_{S2} = 1) \\ & \text{or } (s_{C2} = s_{C1} \text{ and } s_{S2} = 0) \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

  Its goal is to set (1) to 0 when the values of the states s_{C1} and s_{C2} are not compatible with the state s_{S2} (i.e. they are equal to each other, but $s_{S2} = 1$, or are different but $s_{S2} = 0$).

- $\eta^{(C)}_{[s_{C1}, s_{C2}]}(s_{I2})$ is a Boolean function depending on the clock frequency switching law used by the governor to decide the clock according to the input traffic bit rate and the current temperature. Its goal is to set (1) to 0 when the Governor policy chooses a clock frequency different from s_{C2} when its input values are s_{I2} and s_{C1}. More specifically, $\eta^{(C)}_{[s_{C1}, s_{C2}]}(s_{I2}) = 1$ if the Governor policy decides to use the clock frequency s_{C2} when the clock frequency and the input traffic bit rate are s_{C1} and s_{I2}, respectively. Otherwise, $\eta^{(C)}_{[s_{C1}, s_{C2}]}(s_{I2}) = 0$. In our case study, according to the Governor policy described in the previous section, we have:

$$\eta^{(C)}_{[s_{C1}, s_{C2}]}(s_{I2}) = \\ = \begin{cases} 1 & \text{if } s_{C2} = 125\,\text{MHz} \text{ and } s_{I2} \geq 2\text{Gbit/s} \\ 1 - p_G(s_{I2}) & \text{if } s_{C1} = s_{C2} = 125\,\text{MHz} \text{ and } s_{I2} < 2\text{Gbit/s} \\ p_G(s_{I2}) & \text{if } s_{C1} = 125\,\text{MHz} \text{ and } s_{C2} = 62.5\,\text{MHz} \\ & \text{and } s_{I2} < 2\text{Gbit/s} \\ 1 & \text{if } s_{C1} = 62.5\,\text{MHz} \text{ and } s_{C2} = 62.5\,\text{MHz} \\ & \text{and } s_{I2} < 2\text{Gbit/s} \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

  The term $p_G(s_{I2})$ is the probability that the Governor decides to switch down the clock frequency from 125 MHz to 62.5 MHz when the input traffic is less than 2 Gbit/s, and depends on its current value s_{I2}.

- $Q^{(I)}$ is the state transition probability matrix for the quantized input traffic. It is an input of the problem, because it characterizes the traffic crossing the router;
- $Q^{(T)}(s_{C2}, s_{I2})$ is the state transition probability of the temperature. It will be calculated below.

Let $s_{\Sigma 1} = (s_{C1}, s_{I1}, s_{T1}, s_{S1})$ be the state at the end of the previous slot, and let us indicate the array containing the derivative of the temporal temperature behavior when the system state is $s_{\Sigma 1}$ as $\gamma(s_{C2}, s_{I2}, s_{T1})$. Thus, the new CPU temperature in the next slot $n+1$ is:

$$T_2 = s_{T1} + \gamma(s_{C2}, s_{I2}, s_{T1}) \cdot \Delta \quad (4)$$

Therefore, the new state of the temperature, s_{T2}, will be one of the two most adjacent states to T_2 belonging to $\Im^{(T)}$. Let us indicate the most adjacent state with a temperature greater than T_2 as $\lceil T_2 \rceil$, and the most adjacent state with a temperature lower than T_2 as $\lfloor T_2 \rfloor$. The new temperature state s_{T2} will be either $s_{T2} = \lceil T_2 \rceil$ or $s_{T2} = \lfloor T_2 \rfloor$ with a probability dependent on the distance between the real temperature calculated as in (4) and the temperature associated to the adjacent states $\lceil T_2 \rceil$ and $\lfloor T_2 \rfloor$. More specifically:

$$S^{(T)}(n+1) = \begin{cases} \lceil T_2 \rceil & \text{with prob: } (T_2 - \lfloor T_2 \rfloor)/(\lceil T_2 \rceil - \lfloor T_2 \rfloor) \\ \lfloor T_2 \rfloor & \text{with prob: } (\lceil T_2 \rceil - T_2)/(\lceil T_2 \rceil - \lfloor T_2 \rfloor) \end{cases} \quad (5)$$

Now we can calculate the generic element of the state transition probability of the temperature as follows:

$$Q^{(T)}_{[s_{T1}, s_{T2}]}(s_{C2}, s_{I2}) = \begin{cases} (T_2 - \lfloor T_2 \rfloor)/(\lceil T_2 \rceil - \lfloor T_2 \rfloor) & \text{if } s_{T2} = \lceil T_2 \rceil \\ (\lceil T_2 \rceil - T_2)/(\lceil T_2 \rceil - \lfloor T_2 \rfloor) & \text{if } s_{T2} = \lfloor T_2 \rfloor \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

where T_2 is calculated as in (2).

Now, from the matrix $Q^{(\Sigma)}$ we can derive the system steady-state probability array $\underline{\pi}^{(\Sigma)}$ by solving the following system:

$$\pi^{(\Sigma)} Q^{(\Sigma)} = \pi^{(\Sigma)} \quad \text{and} \quad \pi^{(\Sigma)} \cdot \underline{1}^T = 1 \quad (7)$$

where $\underline{1}^T$ is a column array with all the elements equal to one.

Its generic element, $\pi^{(\Sigma)}_{[s_\Sigma]}$, is the steady-state probability of the state $\underline{s}_\Sigma = (s_C, s_I, s_T, s_S)$.

We can now derive the marginal steady-state probability array for the temperature process and its mean value:

$$\pi^{(T)}_{[s_T]} = \sum_{\forall s_C \in \Im^{(C)}} \sum_{\forall s_I \in \Im^{(I)}} \sum_{\forall s_S \in \Im^{(S)}} \pi^{(\Sigma)}_{[s_C, s_I, s_T, s_S]} \quad \text{and}$$
$$E\{\underline{\pi}^{(T)}\} = \sum_{\forall s_T \in \Im^{(T)}} s_T \cdot \pi^{(T)}_{[s_T]} \tag{8}$$

Likewise, we can calculate the mean energy consumption:

$$E\{p\} = \sum_{\forall s_C \in \Im^{(C)}} \sum_{\forall s_C \in \Im^{(C)}} p_{[s_c, s_I]} \sum_{\forall s_I \in \Im^{(I)}} \sum_{\forall s_S \in \Im^{(S)}} \pi^{(\Sigma)}_{[s_C, s_I, s_T, s_S]} \tag{9}$$

where $p_{[s_c, s_I]}$ is the power consumed when the clock frequency is s_C and the input bit rate is s_I.

Let us now calculate the probability of loss occurring during the switching slots. It is defined as:

$$P^{(Loss)} = \lim_{m \to +\infty} \frac{L(m)}{m} \frac{m}{V(m)} = \frac{\overline{L}}{\overline{V}} \tag{10}$$

where $L(m)$ and $V(m)$ are the cumulative number of lost bits and arrived bits in m consecutive slots. The term $\overline{V}$ is the mean value of arrived bits per slot; the term $\overline{L}$ represents the mean value of bits lost per slot. They can be calculated as follows:

$$\overline{V} = \sum_{s_\Sigma \in \Im^{(\Sigma)}} s_I \pi^{(\Sigma)}_{[s_\Sigma]} \quad \text{and} \quad \overline{L} = \sum_{s_C \in \Im^{(C)}} \sum_{s_I \in \Im^{(I)}} \sum_{s_T \in \Im^{(T)}} s_I \pi^{(\Sigma)}_{[s_C, s_I, s_T, 1]} \tag{11}$$

V. NUMERICAL RESULTS

Let us now apply the proposed analytical model to a case study, in order to evaluate the behavior of the green router in terms of both temperature and energy saving.

To this purpose we consider an input traffic quantized in eight different bit rate levels, ranging from 0.5 Gbits/s to 4 Gbit/s with steps of 0.5 Gbit/s. From measurements of real traffic flowing on the DIEEI Lab at the University of Catania, we obtained a tri-diagonal transition probability matrix whose main diagonal, superior and inferior pseudo-diagonals are shown in Table I. Moreover, Figs 5 and 6 show, respectively, the related probability density function and the autocorrelation function.

Then, with the aim of evaluating the behavior of the CPU temperature, we have calculated the relationships between temperature statistics, loss probability and energy saving gain. Moreover, in order to analyze the impact of the traffic correlation on the achieved performance, we considered two more cases of input traffic, characterized by transition probability matrices derived from the one listed in Table II by multiplying the terms of the pseudo-diagonals by a coefficient $\alpha \in \{10^{-2}, 10^{-4}\}$. The terms of the main diagonals are calculated such that the sum of each row is equal to one. In this way first-order statistics remained unchanged, while traffic becomes more correlated for decreasing values of α.

INFERIOR PSEUDO-DIAGONAL		MAIN-DIAGONAL		SUPERIOR PSEUDO- DIAGONAL	
Pos	**Value**	**Pos**	**Value**	**Pos**	**Value**
		(1,1)	9.9990e-001	(1,2)	1.0000e-004
(2,1)	3.1569e-005	(2,2)	9.9993e-001	(2,3)	3.5098e-005
(3,2)	6.7811e-006	(3,3)	9.9994e-001	(3,4)	4.8774e-005
(4,3)	4.1255e-005	(4,4)	9.9995e-001	(4,5)	6.3636e-006
(5,4)	1.9848e-005	(5,5)	9.9994e-001	(5,6)	3.8975e-005
(6,5)	3.8314e-005	(6,6)	9.9992e-001	(6,7)	3.8609e-005
(7,6)	1.3970e-005	(7,7)	9.9990e-001	(7,8)	8.6030e-005
(8,7)	1.4286e-004	(8,8)	9.9986e-001		

Table I: Non-null elements of the Input traffic transition probability matrix

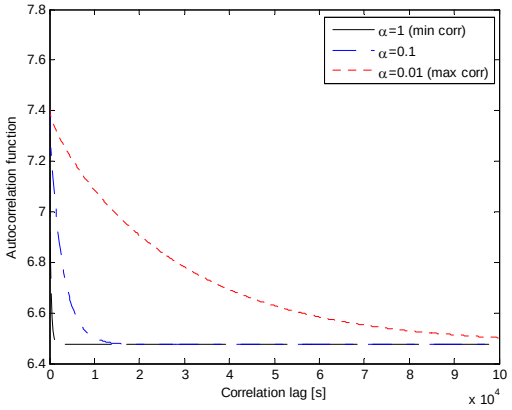

Figure 5. Probability density function

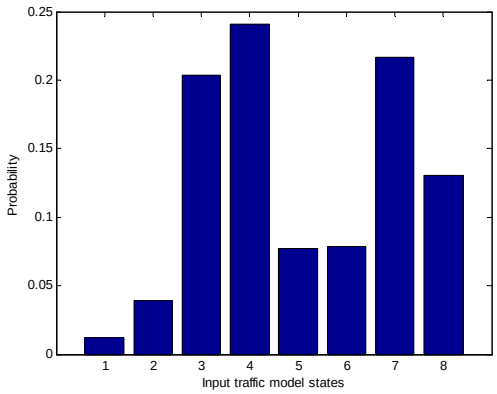

Figure 6. Autocorrelation function

Probability distribution array of the considered traffic has a mean value of 2.54 Gbit/s and a standard deviation of 0.965 Gbit/s. The first step of our analysis consisted in evaluating loss probability and energy saving gain for all the possible combinations of $p_G(s_I) \in \{1 \cdot 10^{-9}, 1 \cdot 10^{-7}, 1 \cdot 10^{-5}, 1 \cdot 10^{-3},\}$, for each $s_I \in \{0.5, 1.0, 1.5, 2.0\}$ Gbit/s. Energy saving gain is defined as follows:

$$\rho = \frac{P_{\max} - E\{p\}}{P_{\max}} \cdot 100\% \tag{12}$$

where P_{max} is the maximum power consumed if no saving policy is applied by the Governor, while $E\{p\}$ is the mean power evaluated in (11).

Results are shown in Figs. 7 and 8 for the three input traffic examples. We can observe that Fig. 7 shows the loss probability with respect to the temperature mean value. For increasing values of the loss probability the temperature mean value decreases. The rationale behind this is that if we want to keep the temperature low, we have to accept some packet loss due to frequency switch. Fig. 8 shows the relation between the loss probability and the energy saving gain. Clearly, when the former increases, the energy saving gain increases as well because higher loss probability means that we are operating in lower frequency clock domains which tend to a lowering of the temperature. These results allow us to design the Governor law to maintain the temperature within a known interval range; this leads us to design the hardware components by considering lower values of temperature which the devices can reach with respect to their current design where they are manufactured for the worst case scenario for what the temperature is concerned.

VI. Conclusions and Future Work

In this paper we have proposed an analytical discrete-time Markov model that allows green router designers to calculate in advance the temperature statistics of a router, for given input traffic statistics and energy saving law applied by the router Governor. Moreover, the model allows the manufacturers to evaluate the energy saving gain which is possible to obtain. In the case study we have shown how the model can be used to design the router Governor parameters to achieve the target of maintaining the mean temperature below a given threshold.

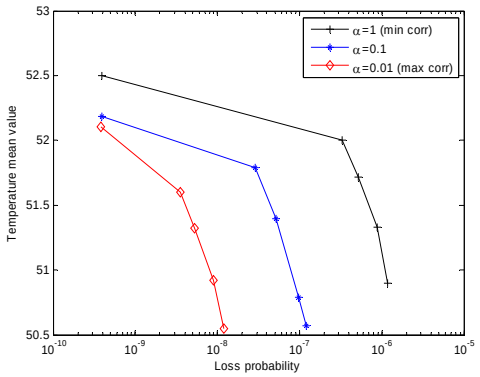

Figure 7. Temperature statistics vs. the target energy saving gain

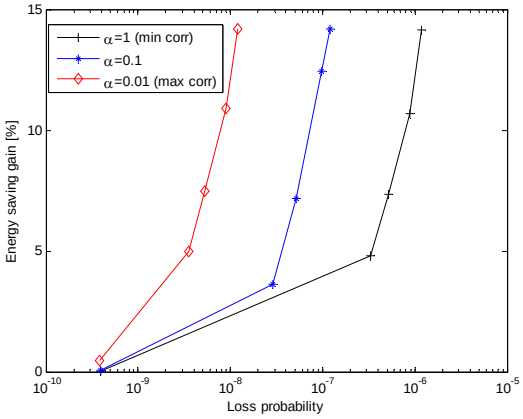

Figure 8. Loss probability vs. the target energy saving gain

Acknowledgement

The work described in this paper is performed with the support of the Econet project (low Energy Consumption NETworks), funded by the EU through the FP7 call.

References

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

[2] Ipmon Sprint, "The applied research group. http://ipmon.sprint.com/,"2007.

[3] A. P. Jardosh, *et alii*, "Towards an energy-star wlan infrastructure," in Proc. of the 8th IEEE Workshop on Mobile Computing Systems and Applications, Washington, DC, USA 2007, pp. 85–90.

[4] The Climate Group, "The climate group: Global e-sustainability initiativereport, smart 2020 enabling low carbon economy in the informationage, http://www.smart2020.org/publications/," 2008.

[5] M. Gupta and S. Singh, "Greening of the Internet," in Proc.of the 2003 conference on Applications, technologies, architectures,and protocols for computer communications, SIGCOMM '03, New York, NY, USA:

[6] S. Nedevschi, *et alii*, "Reducing network energy consumption via sleeping and rateadaptation," in Proc. of the 5th USENIX Symposium on Networked Systems Design and Implementation, 2008.

[7] Econet, "http://www.econet-project.eu/," 2010.

[8] Trend, "http://www.fp7-trend.eu/," 2010.

[9] Greentouch, "http://www.greentouch.org/," 2011.

[10] Cisco, "Cisco energy wise,http://www.cisco.com/," 2009.

[11] R. Bolla, *et alii*, "Energy efficiencyin the future internet: A survey of existing approaches and trends inenergy-aware fixed network infrastructures," in IEEE Comm. and Tutorials (COMST) 2011.

[12] DC. Hu, C. Wu, W. Xiong, B. Wang, J. Wu, and M. Jiang, "On thedesign of green reconfigurable router toward energy efficient internet, in Commun. Magazine, IEEE, v. 49, n. 6, 2011, pp. 83–87.

[13] Massoud Pedram Shahin Nazarian, "Thermal Modeling, Analysis and Management in VLSI Circuits: Principles and Methods", Proceedings of the IEEE, vol 94, Issue 8, pages 1487-1501, 2006.

[14] http://netfpga.org/foswiki/bin/view/NetFPGA/OneGig/ReferenceRouter Walkthrough

[15] G. Gibb, J. W. Lockwood, J. Naous, P. Hartke, N. McKeown, "NetFPGA – An Open Platform for Teaching How to Build Gigabit-Rate Network Switchs and Routers", IEEE Transactions on Education, Vol. 51, N. 3, August 2008.

[16] A. Lombardo, C. Panarello, D. Reforgiato, G. Schembra, "Power control and management in the NetFPGA Gigabit Router," accepted to Future Network & Summit 2012, 4 - 6 July 2012, Berlin, Germany.

[17] V. Sivaraman, A. Vishwanath, Z. Zhao, and C. Russel, "Profiling perpacket and per-byte power consumption in the netfpga gigabit router," IEEE INFOCOM Workshop on Green Communications and Networking (GCN), 2011.