Computer Communications 50 (2014) 175–186

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

Differential energy saving algorithms in a distributed router architecture

Andrea Bianco [a], Fikru Getachew Debele [a,*], Luca Giraudo [b]

[a] Dipartimento di Elettronica e Telecomunicazioni, Politecnico di Torino, Corso Duca degli Abruzzi, 24, 10129 Turin, Italy
[b] Google, 1667 Plymouth Street, Mountain View, CA 94043, USA

ARTICLE INFO

Article history:
Available online 20 March 2014

Keywords:
Energy efficiency
Distributed router
Heuristic algorithms

ABSTRACT

A distributed multistage software router (MSSR) is composed by several interconnected software routers running on personal computers (PCs). The MSSR architecture overcomes scalability and performance issues of single software router by providing parallel forwarding paths. Like many networking devices, a MSSR must be sized for peak traffic load, which implies energy inefficiency at low loads. Thus, we focus on energy saving schemes to improve the router energy efficiency by dynamically adapting the MSSR architecture to the currently offered load. We first introduce an optimal energy saving algorithm defined as a mixed integer linear programming (MILP) optimization model. Then, heuristic solutions, named differential algorithms are discussed. While the optimal approach provides higher energy savings, the heuristics avoid the complete MSSR reconfiguration, thus reducing forwarding delays and minimizing service interruption. The performance evaluation shows that the proposed heuristic algorithms, that gracefully modifies the internal MSSR configuration, preserve the load proportional energy demand characteristics of the optimal algorithm, with a minimal loss of efficiency, largely compensated by algorithm simplicity.

1. Introduction

Although ICT can make a major contribution to support energy savings and to face climate changes, it is responsible for roughly 2% of global carbon emissions – a figure close to the worldwide airline industry consumption. Since the ICT sector is growing at a faster rate [1], the CO_2 emission by ICT industry will reach an estimated of 6% by 2020 [2,3]. Telecom infrastructures and devices contribute to about 25% of the total ICT consumption.

Today the energy consumption in networking devices is proportional to the installed capacity rather than to traffic demands. Thus, dynamically resizing the system capacity to match traffic demands may significantly enhance network energy efficiency. Distributed router architectures, being composed by many independently powered components, are perfectly suited to provide the flexibility needed to dynamically resize the router architecture to match input traffic demands.

This research work focuses on the problem of reducing the energy consumption of distributed router architectures, with emphasis on a multistage software router (MSSR) architecture [4] shown in Fig. 1. The MSSR architecture is a distributed software router architecture proposed to overcome single PC-based software router performance limitations. The main benefits of the MSSR architecture, discussed in [4], include scalability, flexibility, programmability, enhanced performance and low cost.

The architecture exploits classical PCs as elementary switching elements to build a high-performance software router (SR). The MSSR architecture is organized in three stages: (i) a front-end stage exploiting layer-2 load balancers (LBs), either open-software or open-hardware based [5], that act as interfaces to the external networks and distribute IP packets to (ii) back-end personal computers (BEPCs), also named *back-end routers*, that provide IP routing functionality, and (iii) an interconnection network, based on Ethernet switches, that connects the two stages. A control entity, named *Virtual Control Processor* (virtualCP), running on a selected back-end router, controls and manages the overall architecture through a DIST protocol [4]. The virtualCP hides the internal details of the MSSR architecture to external network devices.

State-of-the-art PC-based routers can route few Gbps if packet processing is performed by the CPU [6,7] or few tens of Gbps if a specialized packet processing is implemented [8,9]. Therefore, the MSSR architecture might require several tens of BEPCs to build a high performance routing capability. This performance scale implies a high redundancy level at the back-end stage, which may translate into a source of energy waste during low traffic periods. Indeed, like most networking devices, the MSSR is typically designed to sustain peak traffic, thus dissipating a constant amount of power regardless of the actual traffic load. However, during low traffic periods, the routing task could be transferred to a subset

* Corresponding author.
E-mail addresses: andrea.bianco@polito.it (A. Bianco), fikru.debele@polito.it (F.G. Debele), lucagiraudo@google.com (L. Giraudo).

http://dx.doi.org/10.1016/j.comcom.2014.02.013
0140-3664/

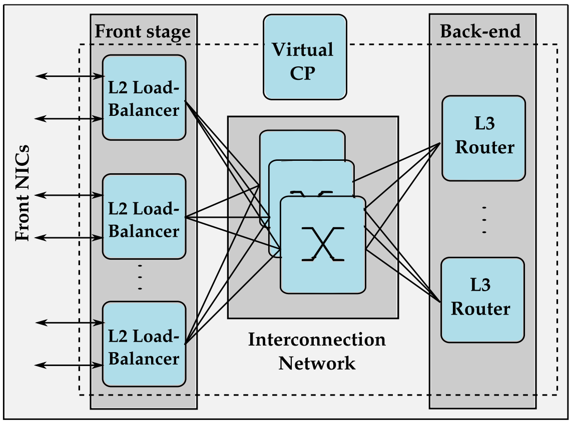

Fig. 1. Multistage software router architecture.

of back-end routers, by setting all un-needed back-end routers in low power state by switching them off to save energy.

As an example, let us examine the design of a MSSR architecture comparable to a Juniper T320 core router that supports up to sixteen 10 Gbps ports with a 160 Gbps forwarding capacity. The T320 router has a nominal power consumption of 2.8 kW [10]. Suppose the following internal components are available to build the MSSR architecture:

- back-end routers with 5.5 Gbps forwarding capacity and equipped with a single 10 Gbps interface;
- LBs with two (one internal and one external) 10 Gbps interfaces;
- a commercial Ethernet switch (or a set of interconnected switches) with enough capacity to interconnect LBs and back-end routers.

To design a 160 Gbps capable MSSR, 16 LBs, 1 switch and 29 back-end routers are needed. Assuming that PCs used as LBs and back-end routers have a power consumption of 80 W[1] and that the interconnecting switch consumes 200 W, the MSSR architecture dissipates roughly 3.8 kW at maximum load. However, during very low traffic periods, one (or few) back-end router(s) may be enough to provide the required service, thanks to input traffic consolidation to few BEPCs while setting all other BEPCs to low power state. Thus, it would be possible to dramatically reduce the MSSR power consumption to 1.5 kW, including the power consumption of LBs and the switch. Thus, the MSSR architecture is competitive, in terms of power consumption, with commercially available routers also in this worst case scenario. However, the real advantage of the MSSR architecture relies on the ability to scale the router configuration to adapt to traffic fluctuations. A commercial router is sized for peak traffic and, today, it is not easily re-sizeable. Since the average traffic load on 24 h is much less than the peak load, having a scalable architecture leads to significant energy savings, as demonstrated in this paper.

In the above example, the LBs and the switch contribute to about 34% of the total energy consumption. We do not consider energy saving features for these two stages because they act, respectively as external interfaces (which must stay active to guarantee MSSR external connectivity) and internal interconnection network (which must be active to guarantee internal connectivity). As such, saving energy by switching off LBs is only possible when operating at the network level, as in [11], where the whole network power consumption is optimized by redirecting the traffic over a subset of routers. Therefore, in this paper we propose algorithms that resize the number of active BEPCs, on the basis of their power consumption and routing capacity, to adapt the overall MSSR capacity to the incoming traffic demand so as to minimize the architecture power consumption. The proposed algorithms gracefully reconfigure the back-end stage when needed without causing service disruption or reconfiguration delays. Preliminary results were presented in [12]. The main novel contributions, besides more extensive results, are: (i) the extension of previously proposed algorithms with other heuristic solutions and (ii) the analysis under unsplittable traffic assumption.

The remainder of the paper is organized as follows: related works in energy savings are presented in Section 2. In Section 3 we present the MSSR energy saving problem and give the formal problem formulation. Proposed heuristic solutions are discussed in Section 4. Section 5 compares the performance of the proposed heuristics with respect to the optimal solution. Finally, Section 6 concludes the paper.

2. Related work

Rising energy cost and increasing environmental standards urged researchers and industries to draw attention to energy footprint aspect of data networking. Starting from a position paper by Gupta et al. [13], IT researchers started focusing on energy saving issues also on data network. In this section, we focus on research efforts related to energy optimization in electronic devices, not limiting our attention to networking devices but including also data centers and server clusters.

Chase and Doyle [14] proposed an energy-conscious request switching paradigm to reduce energy usage for server cluster during low load periods. The switch monitors cluster load and concentrates traffic on the minimal set of servers that can satisfy the load with a specified utilization and latency levels. The remaining idle servers are put to a low-power state. The proposal basically extends the load-balancing switches with an energy-conscious routing policy that leverages the power management features of the back-end servers.

A similar approach is proposed by Pinheiro et al. [15]. In this case a system that dynamically turns cluster nodes either on, to handle a load increase, or off, to save energy during low load periods, is studied. A control-theoretic and load distribution algorithm makes cluster reconfiguration decision by considering the cluster total load, the power and performance implication of changing the current configuration. The technique saves up to 38% of energy.

Power-aware request distribution [16] is a method of scheduling service requests among servers in a cluster so that energy consumption is minimized, while maintaining a particular level of performance. Energy efficiency is obtained by powering down some servers when the desired quality of service can be met with fewer servers.

All the above mentioned schemes [14–16] operate only at the coarse-granularity of the entire server and/or only homogeneous servers are considered. However, in the MSSR architecture back-end routers are heterogeneous both in capacity and power consumption. Furthermore, managing one or more Network Interface Cards (NICs) in the routers adds another dimension to the optimization problem. Observe that a 10 Gbps NIC may consume a non negligible amount of power, roughly 20 W [17], one fourth of a standard PC consumption.

Heath et al. [18] designed a cooperative Web server for a heterogeneous cluster exploiting an optimization model to minimize the energy per request. The approach saves 45% more energy than an energy-conscious server that was proposed for homogeneous clusters. Their approach is similar to the optimal solution of the MSSR design as defined in this paper. However, both solutions

[1] This value could be further reduced for LBs if they are designed in open hardware exploiting field programmable gate array (FPGA).