On the Analytical Characterization of a Real Life Virtual Network Function: The Italtel Virtual Session Border Control

Sergio Montagna and Pietro Paglierani
Italtel S.p.A.: Product Unit Smart Networks
Settimo Milanese, Italy
sergio.montagna@italtel.it; pietro.paglierani@italtel.it

Abstract— **In this paper, we analyze a real-life application of virtualization: the Italtel Virtual Session Border Controller (VSBC). The measurements obtained in *ad hoc* loading experiments show that the VSBC performance is not linear with respect to variations in the call rate. Such a behavior is not in accordance with the theoretical results predicted by standard statistical tools based on queuing theory. As a consequence, particular attention must be paid to accurately assess the VSBC performance, because inaccurate estimates could lead to undue costs, or under-performing solutions. To overcome this problem, a novel approach to accurately predict the VSBC performance is proposed, which allows optimizing the system behavior and minimizing its costs.**

Keywords-Virtualization; Telephony; SBC; RTP; Erlang.

I. INTRODUCTION

In the last decade, many efforts have been devoted by the telecommunication industry to develop in software some fundamental network functions, which could previously be provided only by specialized hardware equipment. Recently, however, the rapid advances in virtualization technologies and parallel computation have made the software implementation of network functions not only feasible, but also very attractive to network providers, as an effective alternative to proprietary hardware-based applications.

The adoption of Virtual Network Functions (VNF) [1] can significantly reduce the costs of network equipment. VNF, in fact, typically run on commercial servers, produced in high volumes and with large economies of scale to satisfy the huge demand originated by the Information Technology market. The use of a common hardware platform to implement a variety of different applications can also greatly simplify the network infrastructure, and therefore reduce its maintenance costs.

Finally, it is widely acknowledged that the use of VNF will enable scalability, rapid re-configuration and optimal allocation of network resources; hence it will give "elasticity" and "openness" to the network infrastructure, now "ossified" by the deployment of a *pletora* of closed appliances based on proprietary hardware architectures [1].

In this paper, we present a simple technique to analyze and predict the performance (i.e., measurement of the virtual machine load, defined in the following [2]) of a complex VNF. As a case study, we consider the problem of assessing the performance of a virtual Internet Protocol Telephony function, namely a Session Border Controller (SBC), implemented and commercialized by Italtel. An SBC [3] operates at the edge of two separate networks, both on the control plane and on the media plane. On the control plane, it performs load balancing and call-control; on the media plane, the SBC provides media adaptation capabilities, i.e., it can adjust in real time the coding format of the speech signals transmitted by the users.

In this paper, we discuss the main problems encountered in the experimental characterization of the VSBC. In particular, we have observed that the standard performance analysis based on classical queuing theory [2] can provide inaccurate results. To overcome such a problem, we present a novel analytical framework, which allows predicting and optimizing the overall VSBC performance in an accurate way. The presented analytical solution can be easily extended to any VNF.

The structure of the paper is as follows. In the next section we briefly describe the VSBC. In Section III we report the experimental performance results observed in the lab in a number of *ad hoc* experiments. Finally, we propose the analytical solution and present our conclusions.

II. THE VIRTUAL SBC MODEL

In Fig. 1, we show a simplified scheme of the virtual SBC implemented by Italtel to handle up to **2K** Erlang.

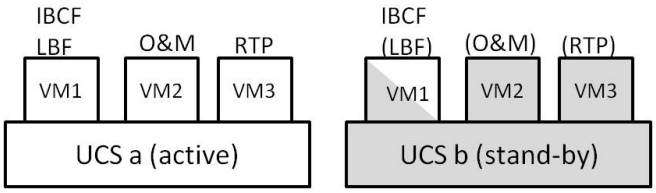

Figure 1. A simplified block-scheme of the virtual SBC architecture. In white, the active VM's, in grey, with the names in brackets, the stand-by VM's.

 ISBN: 978-1-61208-271-4

The used hypervisor is VMWare vSphere Hypervisor 5.1, the VM are based on the Linux operating system. The VSBC runs on two CISCO UCS B200 servers, with hyper-threading enabled, that in the following will be referred to as UCS a and UCS b, respectively. Each server runs three Virtual Machines (VM) implementing three different functions.

a) A first VM, operating on the control plane, implements the Load Balancing Function (LBF) and the Border Control Function (BCF). Such a VM runs on the active server UCS a. Two virtual Central Processing Units (vCPU's) are assigned to this VM, which will be indicated as VM_{LBiBCF}. A second VM operating on the control plane runs on UCS b. This second VM, however, performs only the BCF, while the LBF is in stand-by, thus protecting the LBF running on UCS a in case of fault according to an active/stand-by protection scheme. We will indicate this second VM as VM_{iBCF}.

b) A second VM is dedicated to providing the SBC Operation & Management (O&M) functions. Four vCPU are assigned to such a VM. Also the O&M function is implemented by adopting the active/stand-by redundancy scheme.

c) The third VM is equipped with 4 vCPU. This VM will be referred to as VM_{codec}; it performs Real Transport Protocol (RTP) [4],[5] media packet processing, both in the so-called Network Address Translation (NAT) scenario, and in the transcoding scenario. In the NAT scenario, only the media packet network address is modified, while the RTP header and payload are left unmodified; conversely, in the transcoding scenario the RTP header and payload are processed, so as to change the adopted coding scheme when forwarding the RTP packets from one network to the other. An equivalent stand-by VM is present on the USC b server, to provide redundancy.

A scheme that summarizes the basic call flow is shown in Fig. 2. One can observe that:

a) The basic call includes seven Session Initialization Protocol (SIP) messages, {Invite,100 trying, 180 Ring, 200 OK, ACK, bye, 200 OK}

b) In this scenario, the call is processed both by VM_{LBiBCF} (on UCS a) and by VM_{iBCF} (on UCS b).

c) When the calling and called users adopt different speech codecs, namely G.711 and G.729 [4],[5], the scenario is labeled as Transcoding otherwise as NAT.

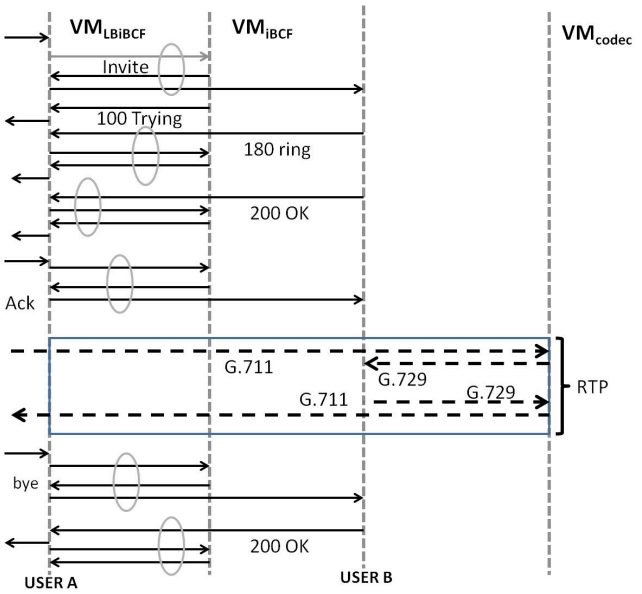

Figure 2. Control plane: SIP/RTP message flow

III. RESULTS

Fig. 3 shows the overall CPU load of VM_{codec}, ($\rho_{VMcodec}$) when a NAT call is considered, as a function of the "call per second" (cps) parameter. We can deduce the following conclusions:

a) Effects due to the RTP packet processing on VM_{codec}: at equal call rates, the load observed on VM_{codec} can result different, due to the difference of offered Erlang values (**520** Erlang; **260** Erlang).

b) From the observed results, we can obtain the experimental values for $\{T_1;h_1^{Erl}(ptime)\}$, which represent the cost of the single call, and the contribution to the load due to the traffic expressed in Erlang in the NAT scenario, respectively. Such values can be used to estimating the load of VM_{codec} through the expression [2]:

$$\rho_{teor}^{VM} = \rho_{base} + \sum_{i=1}^{2}\left[\lambda_i * T_i + \lambda_i * hold_i * h_i^{Erl}(ptime)\right] \quad (1)$$

where:

$\boldsymbol{\rho_{teor}^{VM}}$: is the offered load of $\mathbf{VM_{codec}}$

$\boldsymbol{\rho_{base}}$: is the load of $\mathbf{VM_{codec}}$ without traffic

$\boldsymbol{\lambda_1}$: is the offered call rate to handle NAT

$\boldsymbol{\lambda_2}$: is the offered call rate to handle transcoding

$\mathbf{T_1}$: is the cost of the call to handle NAT

$\mathbf{T_2}$: is the cost of the call to handle transcoding

$\mathbf{hold_1}$: is the length (in seconds) of the RTP phase of the call to handle NAT

$\mathbf{hold_2}$: is the length (in sec.) of the RTP phase of the call to handle transcoding

 ISBN: 978-1-61208-271-4

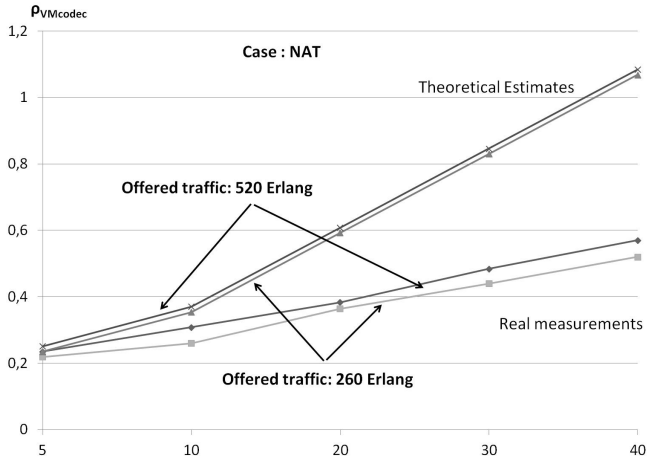

Figure 3. VM_{codec} load as a function of the offered traffic (call per second)

$\mathbf{h_1}^{\mathbf{Erl}}$ (*ptime*): is the cost to manage one NAT Erlang. This value depends on the packetization time (*ptime*) of the codec.
$\mathbf{h_2}^{\mathbf{Erl}}$ (*ptime*): is the cost to manage one transcoding Erlang. This value depends on the packetization time (*ptime*) of the codec.

The results shown in Fig. 3 highlight that the theoretical load predicted by (1) can be significantly underestimated. For a given load value, for instance $\boldsymbol{\rho_{VMcodec}}$ equal to 0.6, the theoretical estimate predicts an actual call rate of 20 call per second (caps), while the actual call rate that can be successfully processed is 40 caps.

IV. ANALYTICAL SOLUTION

The last observation suggests that a different approach must be adopted to predict the virtual SBC load in an accurate way. In Fig. 4, we show a simplified scheme that summarizes the new approach to estimate the load proposed in this paper. The new approach is based on the use of a reduction factor **f** applied to the theoretical estimate $\boldsymbol{\varrho_{teor}}^{\mathbf{VM}}$ of the load provided by (1). This quantity is lower bounded by the VM load without traffic, i.e., ρ_{base}; the upper bound can be theoretically infinite. In our application, we assume that the upper bound is determined by the number of vCPU (N_{vcpu}) dedicated to the considered VM. Thus, the range for the load can be defined as $\rho_{base} \leq \rho_{teor}^{VM} \leq N_{vcpu}$.

We assume that the reduction factor **f** depends on the offered load (ρ_{teor}^{VM}) and on the value N_{vcpu}. Furthermore, we also assume that the reduction function **f** exhibits a linear behavior in the range $1 \leq \mathbf{f} \leq N_{vcpu}$, with $N_{vcpu} \geq 2$. As shown in Fig. 4, the method requires that a load measurement is performed at low traffic, so that the observed load (ρ_{meas}) can be considered close to the base load (ρ_{base}).

This measurement is performed in order to estimate the processing cost of the single call on the considered VM. The estimated VM processing cost thus results equal to:

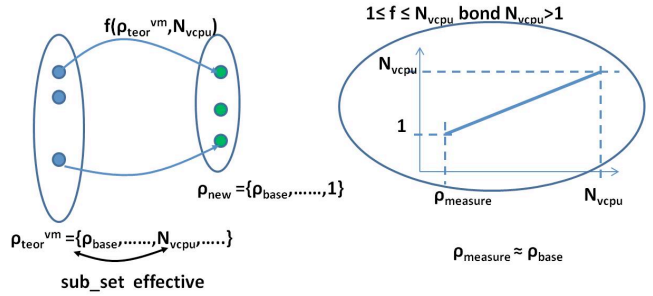

Figure 4 Simplified scheme of the proposed approach to estimate the VM load

$$\rho_{new} = \frac{\rho_{teor}^{VM}}{f} = \frac{\rho_{teor}^{VM} * (N_{vcpu} - \rho_{meas})}{(N_{vcpu} - 1) * (\rho_{teor}^{VM} - \rho_{meas}) + (N_{vcpu} - \rho_{meas})} \quad (2)$$

The application of (2) to the results previously discussed is shown in Fig 5.

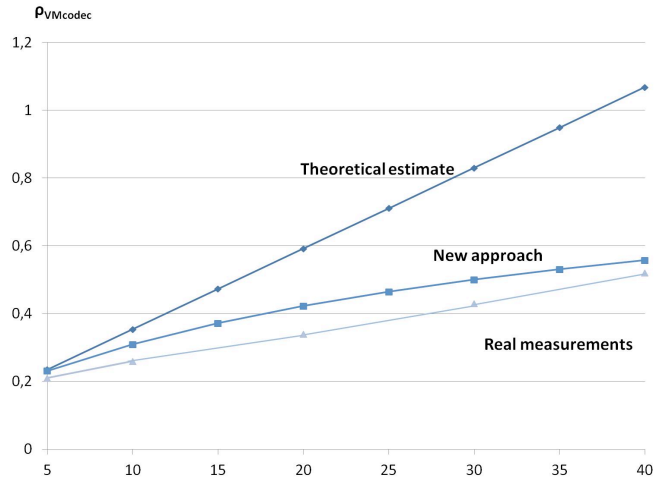

Figure 5. Load of VM_{codec} vs offered call rate (call per second)

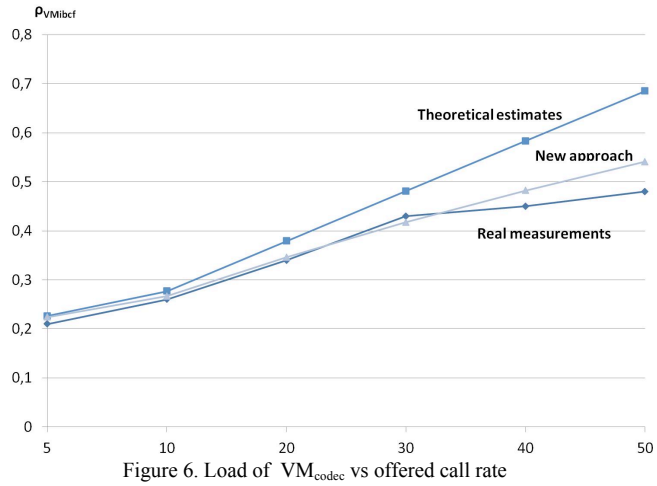

Figure 6. Load of VM_{codec} vs offered call rate

The performance estimates achieved with (2) result significantly more accurate than those provided by (1),

which are linear with respect to the offered traffic.

In Fig. 6 we show the load of VM_{ibcf} as a function of the call rate. In these tests, the quantity "offered Erlang" has been kept constant, equal to **260** Erlang; only the NAT scenario is considered.

The use of (1) provides over-estimated performance values; conversely, the performance predicted by the new approach results more accurate. The obtained measurements also indicate that the gradient of the load curve decreases for increasing traffic. This means that VNF's tend to optimize the cost of processing telephony traffic. Thus, when traffic increases, vCPU's dedicate more time to traffic handling.

V. CONCLUSIONS

The Italtel Virtual Session Border Controller (VSBC) has been developed to handle up to 2K Erlang of Voice over IP sessions (and in the near future, up to 4K Erlang). It has been implemented by adopting the concept of virtualization. The measurements carried out in our laboratory have shown that the VSBC performance (processing load of the virtual machines) is non-linear with respect to variations in the rate of processed calls. It has been observed that the virtual system tends to optimize the processing cost of the calls; as a consequence, the overall performance results better than the one predicted by linear models. We have also proposed a strategy aimed at matching experimental data with analytical predictions. The main result originated by this effort is a technique which allows to accurately dimensioning the deployed solutions, reducing their cost.

The proposed analytical method allows to reliably predicting the number of virtual CPU's that must be assigned to the VSBC, to achieve the target performance. This way, the cost per Erlang of the VSBC can be minimized, thus increasing the competitiveness of the system.

ACKNOWLEDGMENT

The authors wish to thank their colleagues Roberto Porfidio, who carried out the experiments and measurements discussed in this work, and Marco Beccari, for his support on the VSBC.

REFERENCES

[1] M. Chiosi, *et al.*, "Network Functions Virtualization," presented at the "SDN and OpenFlow World Congress", October 22-24,2012, Darmastadt, Germany.

[2] Leonard Kleinrock, "Queuing Systems: Volume I, Theory", John Wiley and Sons, New York, 1975.

[3] IETF RFC 5853 "Requirements from Session Initiation Protocol (SIP) – Session Border Control (SBC) deployement," April 2010.

[4] Paglierani, P.; Petri, D.; "Uncertainty Evaluation of Objective Speech Quality Measurement in VoIP Systems," I*nstrumentation and Measurement, IEEE Transactions on* , vol.58, no.1, pp.46-51, Jan. 2009

[5] IETF RFC 3551, "RTP Profile for Audio and Video Conferences with Minimal Control"

 ISBN: 978-1-61208-271-4