Dynamic Cloud Resource Scheduling in Virtualized 5G Mobile Systems

Ahmad Bilal*, Taleb Tarik†, Andras Vajda*and Bagaa Miloud†
* Ericsson, Finland
† Aalto University, Finland
bilal.x.ahmad@ericsson.com, talebtarik@ieee.org, andras.vajda@ericsson.com, miloud.bagaa@aalto.fi

***Abstract*—In virtualized networks, network functions are delivered as software running on generic hardware allowing service providers to dynamically allocate resources based on traffic and service demands. Network Function Virtualization (NFV) is becoming a key enabler and consequently a hot research topic. Dynamic scaling of resources in NFV is a highly important challenge towards its implementation in real-life networks. In this paper, we propose a method to predict the required resources in the appropriate time to sustain true elasticity in NFV. The capacity of different Virtualized Network Functions (VNFs) would increase/decrease in a way that the CPU utilization is maximized while the overall cost is minimized. In this paper, we present two strategies to predict the day-ahead CPU utilization. The first strategy is an offline scheduling method that helps managing elasticity in virtualized networks by predicting normal days events. The second one is an online scheduling approach that predicts the day-ahead CPU utilization during sudden peaks due to some unusual circumstances. In this paper, we also present new promising results that show the correlation between the control and data planes. Finally, we propose a hybrid algorithm that uses both strategies to efficiently handle elasticity in virtualized networks. The obtained results are encouraging and are all based on real-life data of mobile operator networks.**

I. Introduction

Network Function Virtualization (NFV) [1] promises to deliver network functions as pure software running in a virtualized environment with reduced cost and high deployment efficiency. Traditionally, network nodes are delivered preconfigured with highly optimized configurations and with specialized hardware and dedicated capacity. However, based on the authors' previous work [2], the overall load in these network nodes is only a fraction of installed capacity resulting in inefficient utilization of resources. In order to overcome this limitation, the use of NFV is suggested to increase cost efficiency. As the cloud users follow pay as-you-use business models, the virtualization with dynamic scaling of node size based on load is more cost efficient even if large virtualization overhead for data-plane is taken into account. We also observed from our previous work [2] that the resource utilization is different from a network to another but is highly correlated with time of day and follows almost the same pattern every day within the same network. Based on our previous observation, the prediction of required resources for every day can be done based only on the information gathered from previous days. This will help the different operators to increase the elasticity on their network and hence improve cost efficiency.

On the other hand, traffic patterns in real-life networks may be affected under special circumstances when there are abrupt peaks in utilization due to some major events. These events can be classified into two categories: i) the first category of events are those that can be predicted a priori, such as new year holidays; ii) the second category of events are those happening due to a sudden event and hence cannot be predicted. In this paper, we propose an offline mechanism to predict day-ahead CPU utilization and accordingly schedule required virtualized resources. However, the use of an offline mechanism can predict the day-ahead CPU utilization only when the first category of events is happening. The offline method cannot deal with the second type of events. The events that we cannot predict a priori can dramatically affect the system functionality by creating excessive loads on many components of the NFV system. In order to overcome this limitation, we propose an online approach to predict CPU utilization over short ranges of time. However, this method requires additional resources to continuously monitor the system and dynamically scale the resources.

Based on the resource utilization patterns of both the control plane (CP) and data plane (DP) on daily scale [2], we opt using time series models for forecasting the load a day-ahead, and also for assessing load transition tendency for both CP load and DP load [7]. We also show the correlation between CP and DP utilization. Furthermore, we propose an algorithm that makes use of day-ahead forecasts, for Virtual Machine (VM) allocation based on historical data, and current load for detecting any abnormal peaks due to some special happenings. To the best knowledge of the authors, this is the first VM scheduling models in the context of NFV which is based on historical data from real-life mobile operator networks.

The remainder of this paper is organized as follows. Section II presents some background literature related to NFV technologies. Section III presents our forecasting models. Section IV reports some observations on dynamic relationship between CP and DP. Finally, the paper concludes in Section V.

II. Related Work

NFV is foreseen as an important technology to enable the on-demand creation of cloud-based virtual mobile networks. Resource management is studied extensively in the context of cloud based applications to maximize resource utilization while meeting service level agreements (SLAs)[10]. Different

applications may have dynamic load demands and the resources can be scaled based on workload as cloud users follow pay as-you-use business models. Many researchers have proposed different algorithms for dynamic scaling of resources in cloud environment both at VM-level and resource-level using different techniques. The documents in [3-5] provide surveys on related research work dealing with resource allocation, VM management and VM placement in cloud environments.

In this article, we are only concerned with dynamic scaling of mobile network functions when delivered as virtualized instances on general purpose hardware in multi-tenant datacenters. In [6], a fine granular resource-aware VNF management is proposed for the initial deployment and runtime management of virtual network infrastructures. The limitation is that spinning of new resources would take time and can eventually effect performance. In our previous work [2], we studied the utilization of traditional network nodes which follows a repetitive pattern on daily basis. Utilizing historical time series of real network load, we could predict future demands. All models proposed in this article are evaluated using real network data which is described in our prior work [2]. The resource utilization of real network consists of normal load and few instances of peak load due to some unpredictable events. Our proposed algorithm was able to accurately predict day-ahead resource demand and to prevent impact due to traffic peaks which are rare.

III. TIME SERIES MODELS

In this section, we will describe our models to predict resource demand day-ahead based on historical data and load transition tendency five minutes ahead. We also make some remarks on the dynamic relationship between control plane and data plane utilization. For the sake of industrial applicability, our goal is to come up with an effective, yet simply implementable, model with a simple parameter setting.

A. OFFLINE Method

Dynamic scaling of VMs based on utilization can be done ONLINE or OFFLINE. OFFLINE scheduling has the advantage that it follows a pre-defined schedule, hence supporting normal operations of the networks. The mobile core network nodes which are to be virtualized follow utilization profiles with repetitive patterns on daily basis for both CP and DP [2]. Hence we can make use of time series models to predict day-ahead resource demand on historical utilization data. The utilization data under study are with five minutes granularity. We assumed time series with different window sizes w (i.e., number of days), type of day (weekday or weekend), and allocated different weights α_i to each day for predicting utilization day-ahead. The weight paratmeter α_i is given by equation 1.

$$\alpha_i = \frac{(1-\theta).\theta^i}{\theta(1-\theta^w)}, i = 1, 2, 3...w \tag{1}$$

whereby the parameter θ is called a skew factor. Setting θ to large values assigns uniform weights to each day while setting it to values close to 0 yields highly skewed weights [8]. In our scenario, we implemented the offline method by varying the window size for $w = 2, 4, 5, 7$ along with different types of days for forecasting utilization day-ahead.

1) **Model 1 ($w = 5$ for weekdays, $w = 2$ for weekends)** In this model, we set used $w = 5$ for consecutive weekdays and $w = 2$ for weekends (i.e., previous two weekends). This approach is motivated by the fact that the weekends have slightly different utilizations as compared to weekdays for CP utilization [2]. Weekday's' utilization can be predicted from previous weekdays and Saturday utilization can be predicted from last two Saturday utilizations. The model is evaluated on the basis of Prediction Error (PE), i.e. difference between predicted utilization and actual utilization, to keep underestimation as low as possible. In Model 1, the forecast of weekdays (wd) follow the following formula:

$$\begin{aligned} F_t(wd_1) =& \alpha_1(wd_1) + \alpha_2(wd_2) + \alpha_3(wd_3) + \\ & \alpha_4(wd_4) + \alpha_5(wd_5) \end{aligned} \tag{2}$$

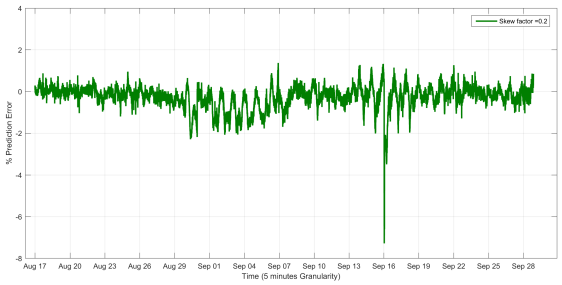
Similarly, the weekend day (we) forecast follows equation 3:

$$F_t(we_1) = \alpha_1(we_1) + \alpha_2(we_2) \tag{3}$$

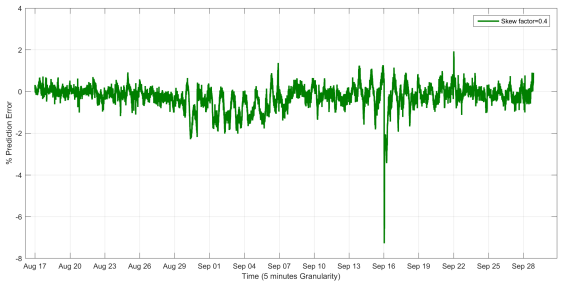
The weights are allocated in such a way that when predicting for day x, we allocate more weights to previous day (x-1). From Fig.1, it can be observed that the prediction error remains between 2% for normal load, which is an acceptable range that can be reflected in the actual resource deployment without much impact on the overall cost. From Fig.1, we also observe that there is one peak when PE reaches -7 $\%$. This peak in PE is due to a sudden peak in traffic load generated during a sudden event. This shows that the proposed offline model is unable to predict such peaks: certainly, these peaks can be predicted during pre-known events such as Football world cup, religious holidays, and New Year eve, but in other cases, they cannot be easily predicted earlier. From Fig. 1, we also observe that the obtained results in terms of PE are slightly better when giving more weight to immediately previous days in the load forecast i.e., setting the skew factor to smaller values e.g., $\theta = 0.2$.

2) **Model 2 ($w = 2$ for weekdays irrespective to day type)** In this model, we used $w = 2$ to forecast load demand day a- head using the utilization data of the same days from two previous weeks. This model uses the same approach of Model 1 applied to weekends. This model is simplest: the load at time t is forecasted by simply using the load at time t, from the same days of the previous two weeks. Similarly, the any day (d) forecast follows equation 4:

$$F_t(d_1) = \alpha_1(d_1) + \alpha_2(d_2), where \alpha_1 + \alpha_2 = 1 \tag{4}$$



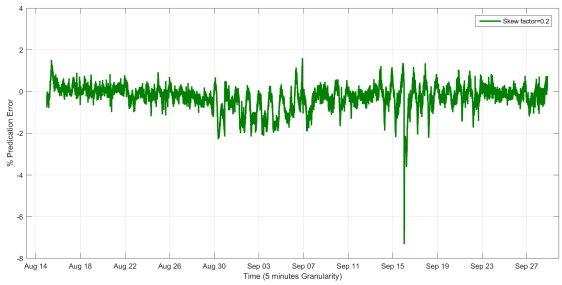
(a) Skew factor $\theta = 0.2$



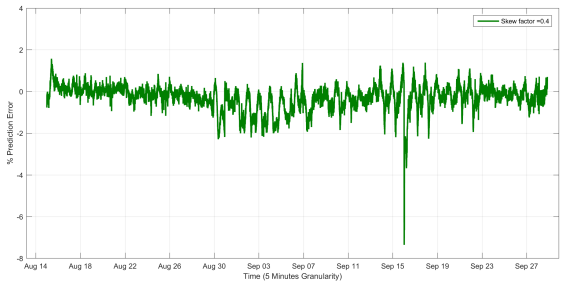
(b) Skew factor $\theta = 0.4$

(c) Skew factor $\theta = 0.8$

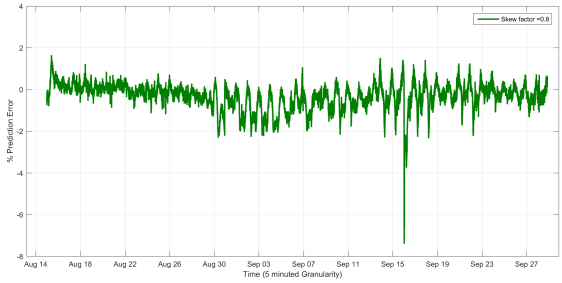
Fig. 1: Prediction errors experienced in Model 1 for different skew factors.



(a) Skew factor $\theta = 0.2$



(b) Skew factor $\theta = 0.4$



(c) Skew factor $\theta = 0.8$

Fig. 2: Prediction errors experienced in Model 2 for different skew factors.

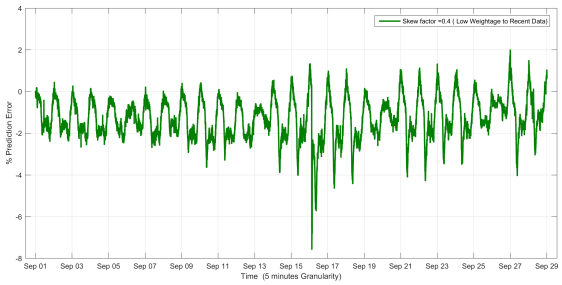
The recent data are given high weight than the older data. Similar to Model 1, Fig. 2 also indicates that PE using Model 2 lies within 2% for normal load and that Model 2 is unable to accurately predict the peak load. From these figures, it becomes apparent that Model 1 is more efficient in terms of load underestimation. We also observe that the skewing factor does not have a significant impact on the prediction accuracy.

3) **Model 3 ($w = 4$ for weekdays irrespective to day type)**
In this model, we set $w = 4$ to predict load demand day ahead using previous four weeks data. For instance, predicting load demand for Monday would utilize previous four Mondays utilization data. This model is an enhancement of Model 2 with regard to increasing historical data for prediction. We also evaluate the model allocating high weight to recent data than the old ones. From Fig.3 (a), we observe that allocating high weight to old data has a negative impact on the prediction accuracy. We then evaluate the model assigning high weight to recent data. Fig.3 (b) shows the efficiency of this strategy in terms of reducing the prediction error. Forecast for any day is given by equation 5:

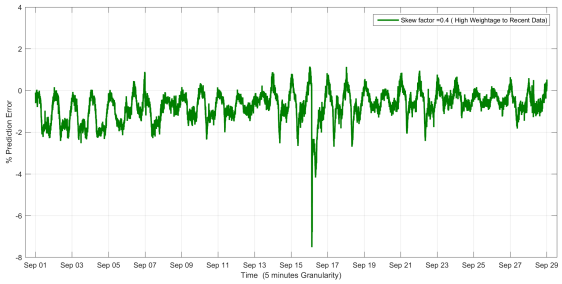
$$F_t(d_1) = \alpha_1(d_1) + \alpha_2(d_2) + \alpha_3(d_3) + \alpha_4(d_4) \quad (5)$$

The skew factor does not yield any significant change in prediction that is why only plots for $\theta = 0.4$ are included in this paper for this model. The PE for this model are more than - 2% due to large w size of historical data for prediction.

Based on the obtained values of PE and amount of time for under estimation of resource utilization experienced during the prediction period, we selected Model 1 for our OFFLINE scheme. Although the skew factor does not significantly change the prediction accuracy, with a $\theta = 0.2$ the results are slightly better. This model will be



(a) Skew factor =0.4 with low weight (Skewness in opposite direction) to recent data



(b) Skew factor =0.4 with high weight to recent data.

Fig. 3: Prediction errors experienced in Model 3 for skew factor = 0.4.

used for predicting day-ahead resource utilization during normal load conditions.

B. Load transition tendency (ONLINE METHOD)

In general, an ONLINE method relies on current or most recent data for load prediction. This method has the advantage that it can detect peaks in resource utilization quickly but at the cost of operational complexity, i.e., additional resources for online monitoring and real-time decision making. The results discussed above reveal that there are instances when peaks in resource utilization cannot be predicted through OFFLINE models. It will be indeed interesting to know a while earlier if the utilization is increasing or decreasing so that resources are allocated in time to ensure high quality of experience [9]. Similar in spirit to [7], the Exponential Smoothing Average (ESA) method is used to detect the load transition tendency so that resource utilization peaks can be detected quickly and the amount of resources needed can be allocated before the load exceeds the capacity level of the node. The choice of exponential smoothing average for predicting load transition tendency is driven by the fact that it is easy to implement and requires minimal computational load. The exponential smoothing average method is formally defined through the following equation equation 6:

$$F_n = F_p + \alpha(A - F_p) \tag{6}$$

where Fn is the next forecasted value for time $t+1$, Fp is the previous forecasted value for time t and A is the actual

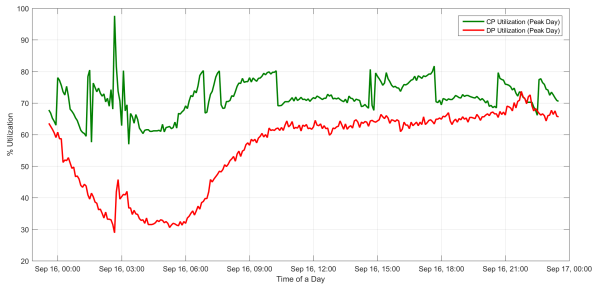


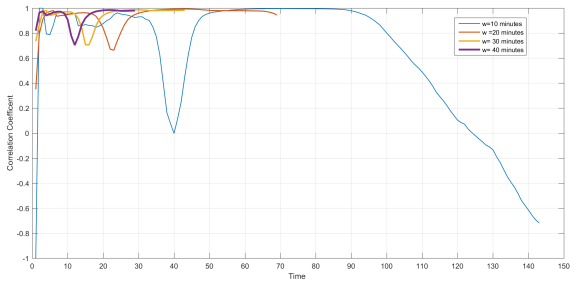
Fig. 4: CP and DP utilization when VMs are allocated based on the Exponential Smoothing Average Method.

value at time t. The value $\alpha = 0.9$ is a damping factor which gives high weight to forecast error. The abrupt change in utilization due to some happening takes some time to reach highest value. During the course of sudden peak, VMs can handle the load for a specified time until the tendency of increase or decrease is predicted in the next forecast. Then, the allocated resources would be scaled up/down based on prediction to meet the required load. ESA predicts this increase soon both in CP and DP before the utilization of allocated VMs increases from 80% to 100%. ESA may be unable to prevent the peak at the very first instant when it hits the node but it soon detects and adjusts the number of resources required to reduce the magnitude of the traffic peak impact on the overall network performance. As depicted in Fig. 4, in the analyzed real network data, we had a case of sudden peak in CP and DP which is soon detected by the ONLINE method and the average VM utilization remains below the recommended 80 % for the rest of the time. With a fine granular data, this method can easily support peak loads much more efficiently and complement existing OFFLINE prediction. Alternatively, we can compare current utilization with some thresholds and allocate more resources when the current utilization exceeds the predefined thresholds.

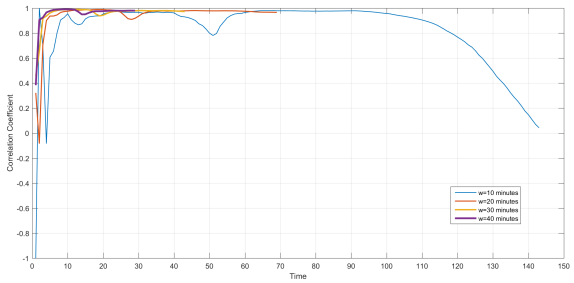
C. Dynamic Correlation Between CP And DP

Based on our previous work [2], we noticed a high correlation between the patterns of the CP and DP resource utilizations over time. We particularly observed that a peak in CP is usually followed by a peak in DP. Based on this observation, a possible correlation between CP and DP can be quantified. The interest in this correlation is to prevent the peak in DP based on previous information from CP. As the time series of the CP and DP resource utilizations are correlated with the time of the day, we used a technique called moving windowed-cross correlation between CP and DP utilization to find the dynamic relationships between the two planes.

Both time series were windowed into different time windows ($w = 2 * 5min, 4 * 5min, 6 * 5min, 8 * 5min$) and the correlation coefficient was calculated using the equation equation 7 for each time window.



(a) Weekday



(b) Weekend

Fig. 5: Windowed Moving Cross Correlation between CP and DP for varying Window Sizes

$$r(A,B) = \frac{1}{N-1}\sum_{i=1}^{N}\left(\frac{A_i-\mu_A}{\sigma_A}\right)\left(\frac{B_i-\mu_B}{\sigma_B}\right) \quad (7)$$

Where μ_A and σ_A are the mean and standard deviation of time series A, respectively. μ_B and σ_B denote the mean and standard deviation of time series B. Different nodes and different days are used for finding cross-correlation between CP and DP. As evident from the plots, on weekends the correlation is much stronger than weekdays because of increase or decrease rate in traffic is slower on weekends as compared to weekdays. We observed that there is a good correlation between CP and DP for $w > 20$ minutes as it can be observed from Fig. 5. Note that the data under study are obtained with five minutes granularity. Based on our study, the correlation between CP and DP would be much stronger and finer if we had much fine granular data. With much fine granular data, we can reveal the accurate relationship equation between CP and DP and hence the results can be used for dynamic scaling of DP based on CP.

IV. THE PROPOSED DYNAMIC CLOUD RESOURCE SCHEDULING FRAMEWORK

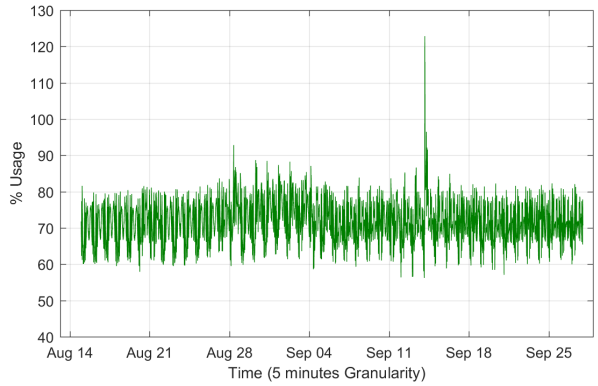
So far, we have studied, in isolation, different concepts and models that can be used for dynamic scheduling of cloud resources in virtualized environment. An intelligent framework that copes with load peaks and supports normal daily loads in a cost-efficient manner is required. For example, the framework could trigger the OFFLINE method for normal day's prediction and when required, triggers the online method to handle sudden peaks due to some unexpected happenings. Effectively, an operator can use our proposed framework to schedule resources in the evening for the next day and in case of peak load it can add more VMs accordingly. The algorithm of the framework is described as follows.

Data: Let V_p be the number of VMs needed from Offline forecast method at time t, V_c be the number of VMs adjusted for current actual utilization at time t. P_u = Prediction load based on offline method. C_u = Current measured load based on predicted VMs. $D = P_u - C_u$, V_a = Allocate VMs at time t
initialization (V_a = V_p);
while *(1)* **do**
  OFFLINE METHOD , V_a = V_p ;
  if $D < -10\%$ *AND* $C_u > 90\%$ **then**
    $V_a = V_p + 1$;
    Modify V_p of OFFLINE METHOD for $t+1$;
  end
end

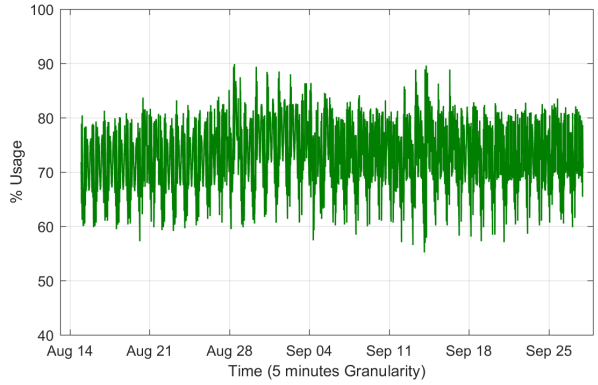
Algorithm 1: PROPOSED algorithm

The algorithm works in a way that during normal load conditions, the required VMs as predicted from the OFFLINE method, support the load demand. However, whenever there is a peak in the traffic load, the algorithm then adds more VMs to cope with the load peak. The added VMs are instantiated as long as they are needed. We used a threshold of 90% for ONLINE method due to granularity of data at hand. Otherwise the above-described load transition tendency method can be used for fine granular data. We tested our proposed algorithm on the measured data from nodes. The data consist of some instances when there is a peak in the load. It was observed that the online scheduling is triggered only when there is a peak in the load. For data with 5 minutes granularity measured during a one and half month, online scheduling is triggered only 0.18% of the times. This means the ONLINE method is triggered only when there is abnormal load and does not impact the operations of the OFFLINE method during normal load conditions. In Fig.6 (a), one can observe that the OFFLINE method assigned a number of VMs that are unable to support the abnormal peak conditions and the resource utilization exceeded the recommended maximum threshold of 90%.

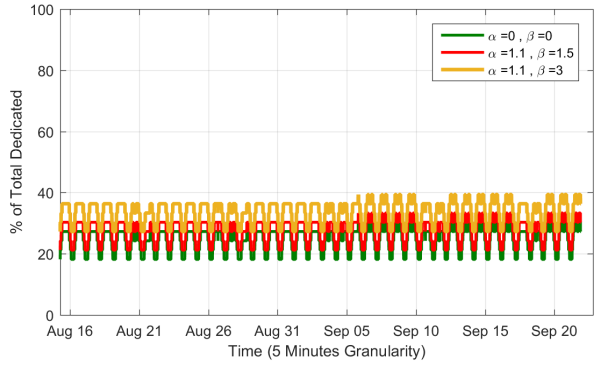
The proposed framework activates the ONLINE method to add resources when the network operates under abnormal peak conditions. As it can be observed from Fig.6 (b), using this framework, the utilization remains below 90 % even during load peak scenarios. We can also observe from Fig.6 (c) that NFV with dynamic scaling is more cost-efficient than its native counterpart, with only 20-40% of the native resources being required, in the presence of a suitable dynamic scaling algorithm.



(a) Actual Utilization based on VMs allocated only by OFFLINE METHOD



(b) Utilization based on the proposed framework



(c) % of dedicated resources allocated by the framework taking into account the virtualization overheads of CP and DP ,α and β ,respectively [2].

Fig. 6: Efficieny of Proposed framework

V. CONCLUSIONS

In this work, we studied different ONLINE and OFFLINE models for allocating VMs to VNFs based on workload prediction. The OFFLINE method is good for allocating cloud resources when the network operates under normal workload, whilst the ONLINE methods are used only when there is a peak in the traffic load due to some major events. The OFFLINE method provides operational flexibility while the ONLINE method ensures detecting load peaks which may be unpredictable. The dynamic relationship between control plane utilization and data plane utilization was also investigated. It was observed that there is a strong correlation between CP and DP. With data measured at a finer granularity, this correlation is expected to be further stronger, yielding more accurate results in predicting load and accordingly required cloud resources. Based on the above and for the sake of industrial applicability, we proposed a lightweight framework that decides when to use only the offline method and when to trigger the online one, while ensuring low computational resources. The use of the OFFLINE resource scheduling approach improves the QoE and limits the operational impact as VMs needed for the next instant in time is known one day earlier. Meanwhile, the ONLINE approach copes with sudden peaks. The obtained results demonstrate the efficiency of proposed framework, requiring less than 40% of the resources compared to native deployments.

REFERENCES

[1] ETSI, *"Network Functions Virtualisation: An Introduction, Benefits, Enablers, Challenges & Call for Action,"* NFV - Introductory white paper, Oct. 2012

[2] A. Bilal, A. Vajda and T. Taleb *"Impact of Network Function Virtualization : A Study based on Real-Life Mobile Network Data ,"* in Proc. IEEE IWCMC'16, Paphos, Cyprus, Sep. 2016.

[3] B. Jennings and R. Stadler, *" Resource Management in Clouds : Survey and Research Challenges, "* Journal of Network and Systems Management, pp. 1-53, 2014.

[4] B.H Bhavani and H.S. Guruprasad *"Resource Provisioning Techniques in Cloud Computing Envirnment : A Survey,"* International Journal of Research in Computer and Communication Technology , Vol.3, no.3, pp.395-401,2014.

[5] M.H. Mohamaddiah , A. Abdullah ,S. Subramaniam and M. Hussin, *"A Survey on Resource Allocation and Monitoring in Cloud Computing, "* International Journal of Machine Learning & Computing, vol.4, no. 1, pp. 31-38, 2014.

[6] F.Z. Yousaf and T. Taleb, *"Fine Granular Resource-Aware Virtual Network Function Management for 5G Carrier Cloud,"* to appear in IEEE Network Magazine.

[7] T. Taleb, A. Jamalipour, Y. Nemoto, and N. Kato, *"DEMAPS: A Load-Transition Based Mobility Management Scheme for an Efficient Selection of MAP in Mobile IPv6 Networks,"* in IEEE Trans. on Vehicular Technology journal, Vol. 58, No. 2, Feb. 2009. Pp. 954-965

[8] T. Taleb, N. Kato, and Y. Nemoto, *"Neighbors-Buffering Based Video-on-Demand Architecture"*, Signal Processing: Image Communication, Vol. 18, No. 7, Aug. 2003, pp 515-526.

[9] S. Dutta, T. Taleb, and A. Ksentini, *" QoE-aware Elasticity Support in Cloud-Native 5G Systems,"* in IEEE ICC16, Kuala Lumpur, Malaysia, May 2016.

[10] ETSI, *"NFV Virtualization Requirements,"* , Oct. 2013.