A Batch Scheduling Algorithm for VPON Reconfiguration and BBU Migration in Hybrid Cloud-Fog RAN

Rodrigo Izidoro Tinini[1], Gustavo Bittencourt Figueiredo[2], Daniel Macêdo Batista[1]
Department of Computer Science – University of São Paulo – Brazil[1],
Department of Computer Science – Federal University of Bahia – Brazil[2]
rtinini, batista@ime.usp.br[1], gustavo@dcc.ufba.br[2]

***Abstract*—Hybrid Cloud-Fog Radio Access Network (CF-RAN) is a recent network architecture proposed to increase network coverage from CRAN while leveraging power consumption in future 5G networks. In CF-RAN, the processing of baseband signals from Remote Radio-Heads (RRHs) can be performed in virtualized BaseBand Units (vBBUs) located in the cloud or in fog nodes that are instantiated in function of the network demand. Through a Time-and-Wavelength Division Multiplexing Passive Optical Network (TWDM-PON), virtualized PONs (VPONs) can be dynamically created to support transmissions from RRHs to vBBUs. However, due to traffic fluctuations, the amount of necessary vBBUs and VPONs may change along a day. In this paper, we propose a batch scheduling algorithm based on Integer Linear Programming (ILP) to perform reconfiguration of VPONs and migration of vBBUs among processing nodes in function of fluctuation on traffic demands. Our results show that, in comparison to an incremental algorithm without reconfiguration of VPONs and vBBUs migration capacities, our algorithm reduces power consumption and bandwidth wastage by up to 28% and 57%, respectively, and also eliminates blocking probability.**

***Index Terms*—5G, Optical Networks, Cloud-Fog RAN, TWDM-PON, NFV**

I. Introduction

Hybrid Radio Access Networks (RAN) architectures are being proposed to increase user coverage in cloud-based RAN (CRAN) in future 5G networks [1]. In CRAN, the network CAPEX and OPEX can be decreased due to the centralization of baseband processing in a cloud. This is done by moving BaseBand Units (BBU) from cell sites and placing them in a BBU pool in the cloud. Hence, low-energy Remote Radio-Heads are left on cell cites to gather user equipment (UE) signals. In such architecture, an optical network called fronthaul is implemented to transmit digitized baseband signals between RRHs and the BBU pool under the Common Public Radio Interface (CPRI) [2].

While CRAN is able to reduce energy footprint, its coverage is limited by the fronthaul or cloud capacity. To increase the network coverage, a hybrid Cloud-Fog RAN (CF-RAN) [3] [4] was proposed to expand the baseband processing to fog nodes placed close to the RRHs [5] that can be dynamically activated to support the exhaustion of the fronthaul due to increase in network traffic.

978-1-7281-2522-0/19/$31.00 ©2019 IEEE

In CF-RAN, all CPRI processing is done in virtualized BBUs (vBBUs) by means of Network Functions Virtualization (NFV) [6]. So, vBBUs can be dynamically turned on or off in processing nodes and even processing nodes can be dynamically turned on or off to suit the network demand. Moreover, to meet CPRI latency and bandwidth requirements, the CF-RAN fronthaul is implemented under a Time-and-Wavelength Division Multiplexing Passive Optical Network (TWDM-PON) [7]. In TWDM-PON, virtualized PONs (VPONs) can be dynamically created to support multiple RRHs transmitting to a common processing node through a shared wavelength dedicated to this processing node [8].

However, the mobile traffic demand goes through many fluctuations over a day [9]. Those fluctuations can demand different numbers of vBBUs and VPONs to support incoming CPRI flows and may lead to an unbalancing distribution of load in VPONs and processing nodes. Hence, fog nodes can be kept consuming energy in moments when the better would be to have their workload migrated to the cloud. The ideal would be that, after a load balancing, lightly-loaded nodes could be turned off. Without loss of generality, VPONs used to transmit to these ligthly-loaded nodes could be reconfigured to transmit to the cloud and increase fronthaul availability.

In this paper we propose a load balancing mechanism through the migration of vBBUs between fog nodes and the cloud and reconfiguration of VPONs. Our approach relies on a batch scheduling algorithm based on an Integer Linear Programming (ILP) formulation used to find the lowest power consuming scheduling of CPRI flows across different traffic load patterns. Our algorithm seeks to balance the load on processing nodes and VPONs when a new RRH demands CPRI transmission or when a RRH is turned off due to traffic fluctuation. Compared to an incremental scheduling algorithm without load balancing, it was possible to eliminate network blocking probability. Results also showed that vBBUs migration and reconfiguration of VPONs play an important role on the power consumption reduction.

The rest of this paper is organized as follows: Section II discusses related works; in Section III, details of the CF-RAN architecture and the TWDM-PON fronthaul are presented; the concept of vBBUs migration and reconfiguration of VPONs is introduced in Section IV; Section V presents our batch

scheduling algorithm, the main contribution of the paper; numerical results obtained with experiments are shown in Section VI and Section VII concludes the paper.

II. Related Work

Scheduling of baseband processing and load balancing is a recurrent research topic in works related to CRAN and hybrid architectures like CF-RAN and Fog-RAN (F-RAN). Some of these works propose jointly solutions to both baseband processing scheduling and load balancing. Other works try to balance the load in fog nodes as traffic patterns change.

In [10], authors addressed the load balancing in VPON/CRAN jointly with the scheduling of sets of RRHs that participate in mitigation of interference through Coordinated Multi-Point (CoMP) techniques. Through an ILP formulation and a graph-based heuristic, it was possible to schedule adjacent RRHs into a common VPON to keep load balancing and reduce the processing latency. However, the the re-scheduling of RRHs in VPONs as traffic decreases was not addressed, which can incur additional latency to keep communication between RRHs scheduled in lightly-loaded VPONs. Moreover, load balancing was only considered in VPONs, in such a way that the load distribution in BBUs was not explored, which could incur additional latency due to communication among multiple BBUs of a processing node.

Authors in [8] proposed a RRH scheduling algorithm in CRAN to balance load in BBUs in a cloud. In this work, heuristics were proposed to distribute the load of multiple VPONs into the least number of BBUs. However, the bandwidth capacity of the fronthaul was not considered, providing a non-realistic scenario where the TWDM-PON capacity was never exhausted. Similarly, the study in [11] explored load balancing in VPONs to decrease handovers in CRAN. By placing several RRHs involved on the mitigation of interference of UEs traversing different cell sites into the same VPON and BBU, latency of interference mitigation was decreased and the throughput was optimized.

Authors in [12] proposed a load balancing mechanism in CRAN to reduce latency in applications of an Internet of Things (IoT) scenario. Metrics such as the size of the BBUs processing queue, lowest waiting time on the cloud and the least aggregated load in BBUs was took into account in the proposed algorithms. Results showed that the scheduling of applications on BBUs with the lowest waiting time for processing provided reductions on the processing latency.

Load balancing during content access in F-RANs was proposed by authors in [5]. In this work, content can be provided both by cloud servers or in cache copies in fog nodes. The proposed algorithm seeks to balance the load in the backhaul, providing access to cached content when the backhaul is stressed. It was proposed an optimization model based in clustering to schedule multicast transmissions between RRHs and groups of UEs that demand the same content. The clustering model was able to reduce power consumption, balance the backhaul load through the provisioning of cached content and optimize the Quality of Service (QoS).

In our previous work [13], we proposed an optimal placement of baseband processing functions in CF-RAN through an ILP formulation. Results showed that CF-RAN brings huge reductions in power consumption in comparison with traditional Distributed RANs (DRAN) and increases network coverage in comparison with CRAN. We also proposed an optimal dimensioning of wavelengths in CF-RAN in [3], where a set of limited VPONs were dimensioned among fog nodes in order to increase power efficiency and avoid signals collisions in optical links. Finally, in [4] we proposed a graph-based heuristic in order to place baseband processing and create VPONs in a low latency and energy-efficient manner in dynamic traffic scenarios. However, none of these works addressed the migration of vBBUs and reconfiguration of VPONs among the cloud and fog nodes. So, the contribution of this work relies on the jointly scheduling and load balancing of RRHs into vBBUs and VPONs, while considering bandwidth capacity constraints in the fronthaul and how the VPONs can be reconfigured to deal with vBBUs migrations between fog nodes and the cloud.

III. CF-RAN Architecture

The CF-RAN architecture and its operations is summarized in Fig. 1. The CF-RAN architecture is composed of a cloud and fog nodes that implement the virtualized baseband processing in vBBUs. Each processing node has a dedicated server where a set of Virtual Digital Units (VDU) is implemented. A VDU is a container where vBBUs can be instantiated to perform baseband processing. A TWDM-PON fronthaul is used to connect RRHs both to fog nodes and the cloud. RRHs are connected to an Optical Network Unit (ONU), responsible for transmitting CPRI traffic in a wavelength. Each processing node is equipped with an Optical Line Terminal (OLT), responsible to receive CPRI traffic from ONUs. Dedicated PONs called VPONs can be created to support transmission from ONUs that share the same wavelength to transmit to a common processing node, sharing the same optical channel in a Time Division Multiplexing (TDM) manner. The total fiber extension from RRHs to the cloud is of 40 km, providing a maximum round-trip time of CPRI flows of about 19μs, which is very below the fronthaul propagation latency of 250μs [4].

The CPRI traffic transmitted in a VPON is received by the OLT of a processing node and forwarded to a transceiver called Line Card (LC). Each OLT has a set of LCs to receive the traffic for each available wavelength. Each LC is directly connected to a VDU and the data terminated in a LC are forwarded to its VDU. The LCs and VDUs of a processing node has the same cardinality $|W|$, where W is the set of wavelengths of the TWDM-PON. However, if the workload of a VPON is greater than the processing capacity of the VDU associated to its LC, an auxiliary VDU (associated to other LC) can be used to receive the surplus workload of the original VDU. In that case, an internal switch, responsible to interconnect different VDUs, can be activated to switch traffic of a single VPON among different VDUs.

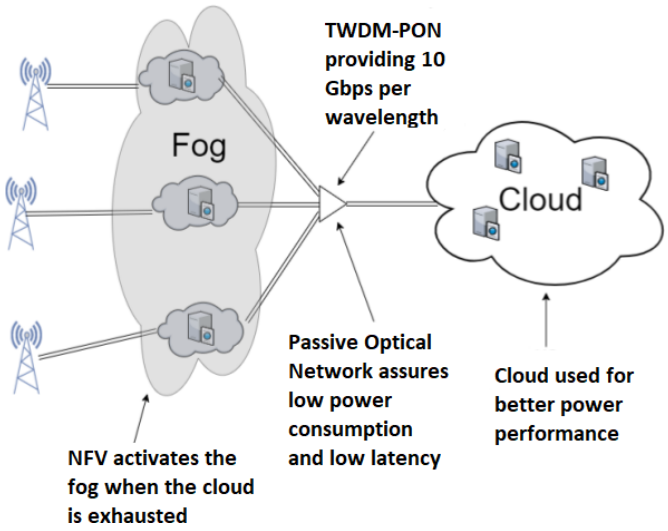

Fig. 1. Overview of CF-RAN architecture

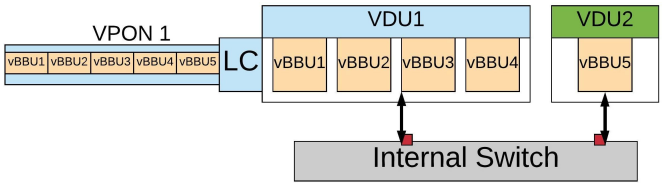

Fig. 2. Internal switch used to switch traffic between VDUs

As shown in Fig. 2, VPON1 is transmitting the data from 5 different vBBUs. However, its VDU has capacity to process only 4 vBBUs. In this case, the excess vBBU is instantiated in other VDU and its data is forwarded from its VPON to this auxiliary VDU through the internal switch (without creating a new VPON for the auxiliary VDU). Note that the utilization of the switch will incur additional latencies to the baseband processing and the activation of its ports will also incur additional power consumption.

As the baseband processing is virtualized in CF-RAN, using NFV technology, each VDU can be instantiated only when an RRH is activated and demands for a vBBU, otherwise, unused VDUs are kept off to save energy. So, as soon as an RRH becomes active, the network operator must find a suitable processing node with a VDU with enough capacity to instantiate a new vBBU to this RRH. To save energy, as long as there is free capacity on the cloud, only its VDUs are used to instantiate new vBBUs. After the vBBU is instantiated, a suitable VPON must be find to support the CPRI transmission between the RRH and the new deployed vBBU. If there are available VPONs on the host processing node of the vBBU, they are used for a better bandwidth usage. if not, an available wavelength, i.e., that was not used to create a VPON, must be allocated to a processing node in order to create a new VPON to receive CPRI transmission in that node. Only when the capacity of the cloud is exhausted, the fog nodes begin to be activated so new vBBUs and VPONs can be created.

IV. vBBUs Migration and VPONs Reconfiguration

Due to UEs mobility, the network traffic load shows different patterns regarding time of day. For instance, the central business district of a city shows peak rates of traffic load among 10a.m. and 2p.m.. On the other hand, on the first hours of the morning and during the night, the workload is reduced. The network traffic load will define the number of vBBUs and VPONs that will be necessary to support the network demand. Hence, to different hours in the day, a different number of vBBUs and VPONs will be necessary.

Considering the CF-RAN architecture, in times of high load, the fog nodes will be necessary to support the overall CPRI traffic. However, when the traffic fluctuates and the network demand is decreased, RRHs are deactivated and resources allocated to them are freed from use. Hence, the network load can become unbalanced regarding the active fog nodes and VPONs. So, if the cloud has enough capacity when fog nodes become unbalanced, the workload of fog nodes can migrate to it. Similarly, a VPON that was used to transmit to the fog node can be reconfigured to transmit to the cloud, increasing fronthaul capacity, or even be deactivated if the VPONs previous allocated to the cloud have enough bandwidth to support the migrated traffic. When an RRH has its vBBU migrated to a new VPON, it is necessary to reconfigure its ONU to be tuned to the wavelength of the new VPON.

Fig. 3 shows an example of migration and VPON reconfiguration. In Fig. 3 a), the workload from fog nodes 1 and 2 is the same as the available capacity of the cloud. Then, as show in Fig. 3 b), their workload are both migrated to the cloud and they are deactivated. Also, note that, VPON1 from fog node 1 was reconfigured to transmit to the cloud and the traffic from VPON2 was migrated to it. After that, VPON2 was deactivated as well and the load was balanced.

Note that for the migration to be possible, it is necessary that the cloud has enough capacity both in its VDUs and in its internal switch, besides the fronthaul capacity provided by

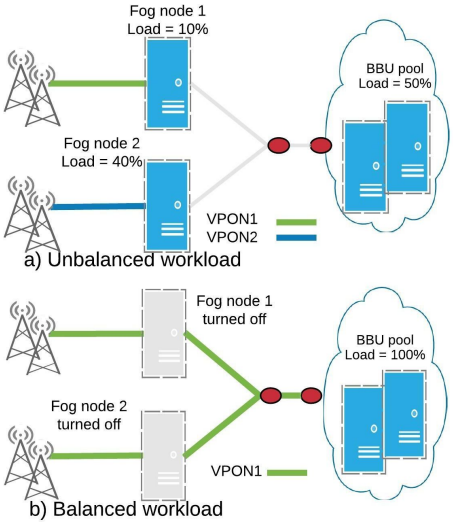

Fig. 3. Migration of vBBUs and VPONs from fog nodes to the cloud

its VPONs. For instance, if a vBBU migrated to the cloud will be placed in an auxiliary VDU from an existing VPON, the internal switch must have enough bandwidth to switch the traffic from the operating VDU of the VPON to the auxiliary VDU where the migrated vBBU will be placed. If the cloud has not enough capacity both in the fronthaul, in its VDUs or even in the internal switch, the migration can not be performed.

Besides the unbalanced load among cloud and fog nodes, an internal unbalanced load in processing nodes may lead to power and bandwidth wastage and even to additional switching delays between VDUs. As shown in Fig. 4 a), after a traffic fluctuation, VPON 1 is transmitting RRHs 1 and 2 to VDU 1 and using the internal switch to transmit RRH 6 to VDU 2. As for VPON 2, it is transmitting RRHs 7 and 8 to its associated VDU 2. However, suppose that VDU 1 can handle all of this load. So, a load balancing needs to be performed in order to decrease the active elements. In Fig. 4 b), VPON 2 and VDU 2 are deactivated and their traffic is moved to VPON 1 and VDU 1, shutting down the internal switch and removing its power consumption and the switching delay from the total baseband processing latency. Finally, power consumption from VPON 2 and VDU 2 are mitigated. In next section we will present our proposed batch mechanism to instantiate vBBUs and VPONs and perform load balancing.

V. Proposed Batch Scheduling Algorithm

Our proposed algorithm is responsible to jointly instantiate vBBUs and VPONs as RRHs are activated and perform migrations as the traffic load changes. The objective of the algorithm is to maintain the load on processing nodes and VPONs always balanced. Hence, every time a new RRH becomes active, the algorithm searches for a new optimal scheduling to this incoming RRH and for all the others already being processed. Similarly, every time an RRH is deactivated due to traffic load fluctuation, the algorithm searches for a new scheduling for the remaining active RRHs, always seeking

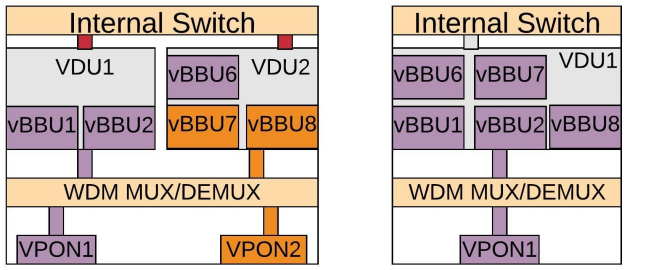

Fig. 4. Reconfiguration of VPONs inside a processing node

to minimize the number of active processing nodes, VDUs and VPONs. Hence, if a new scheduling, with a less power consumption, is found through the migration of vBBUs and VPONs to the cloud, the network resources allocation are reconfigured to this new optimal scheduling and the migrations are performed.

The proposed algorithm is based on batch scheduling of RRHs. An ILP model is proposed to schedule batches of RRHs every time an RRH is activated or deactivated. Since ILPs are commonly used to perform scheduling in static traffic scenarios, we propose an algorithm to invoke the ILP and decide about the migration of vBBUs and VPONs, allowing the use of the ILP in dynamic scenarios

A. ILP Formulation

Input Variables

R: set of RRHs r; N: set of processing nodes n; F_{rn}: set of binary variables representing fog nodes n connected to RRH r; V_{wn}: set of binary variables representing the availability of each wavelength/VPON w to be placed in processing node n; W: set of available wavelengths and VDUs w; B_r: bandwidth demand of RRH r; B_w: bandwidth capacity of wavelength w; I_w: processing capacity of VDU w; B_{e_n}: bandwidth capacity of internal switch e in node n; C_n: power cost of node n; C_{lc}: power cost of a LC; C_e: power cost of the switch activation; B: a very large positive number; α, β, and ρ: weights used to prioritize the minimization of different network elements in the objective function.

Decision Variables

x_{rwn}: = 1 if the vBBU of RRH r is instantiated in node n using VPON w to transmit its CPRI traffic; u_{rwn}: = 1 if the vBBU from RRH r is instantiated in VDU w in node n; y_{rn}: = 1 if r was placed in node n; x_n: = 1 if node n was activated; z_{wn}: = 1 if VPON w is transmitting to node n; k_{rn}: = 1 if the vBBU of RRH r was allocated in an auxiliary VDU w in node n; r_{wn}: = 1 if VDU w was activated to receive an excess vBBU in node n; s_{wn}: = 1 if VDU w is activated in node n; e_n: = 1 if the switch e was activated in node n; g_{rwn}: auxiliary variable that equals 1 if the vBBU from RRH r was placed in the auxiliary VDU w in node n.

Objective Function

Objective function (1) aims to minimize the active processing nodes, VPONs and the switching of traffic of vBBUs

among VDUs. Weights β and ρ are used to prioritize the minimization of active VPONs or the switching of CPRI traffic among VDUs through the internal switch with the objective of evaluating the network performance under different minimization objectives. If β is 1, the minimization of active VPONs is prioritized (this case will be referred in the rest of the paper as *minVPON*). Otherwise, if ρ is 1, the minimization of the use of the internal switch is prioritized (this case will be referred as *minRedir*). In this paper α will always be 1 because the minimization of active processing nodes will always be prioritized to reduce power consumption.

$$(1) Min.\ \ \alpha.(\sum_{n=1}^{N} x_n.C_n) + \beta.(\sum_{w=1}^{W}\sum_{n=1}^{N} z_{wn}.C_{lc}) + \rho.(\sum_{n=1}^{N} e_n.C_e)$$

Constraints

$$(2)\sum_{w=1}^{|W|}\sum_{n=1}^{|N|} x_{rwn}=1, \forall r\in R, (3)\sum_{w=1}^{|W|}\sum_{n=1}^{|N|} x_{rwn}=1, \forall r\in R$$

$$(4)\sum_{r=1}^{|R|} x_{rwn}\geq 0, \forall w,n\in W,N, (5)\sum_{n=1}^{|N|} y_{rn}=1, \forall r\in R$$

$$(6)\sum_{n=1}^{|N|} z_{wn}\leq 1, \forall w\in W, (7) z_{wn}\leq V_{wn}, \forall w,n\in W,N$$

$$(8) y_{rn}\leq F_{rn}, \forall r,n\in R,N, (9)\sum_{r=1}^{|R|}\sum_{n=1}^{|N|} x_{rwn}.B_r\leq B_w, \forall w\in W$$

$$(10)\sum_{r=1}^{|R|}\sum_{n=1}^{|N|} x_{rwn}\leq I_w, \forall w\in W, (11)\sum_{r=1}^{|R|} k_{rn}.B_r\leq B_{e_n}, \forall n\in N$$

$$(12) B.x_n\geq\sum_{r=1}^{|R|}\sum_{w=1}^{|W|} x_{rwn}, \forall n\in N, (13) x_n\leq\sum_{r=1}^{|R|}\sum_{w=1}^{|W|} x_{rwn}, \forall n\in N$$

$$(14) B.z_{wn}\geq\sum_{r=1}^{|R|}\sum_{n=1}^{|N|} x_{rwn}, \forall w\in W, (15) z_{wn}\leq\sum_{r=1}^{|R|}\sum_{n=1}^{|N|} x_{rwn}, \forall w\in W$$

$$(16) B.y_{rn}\geq\sum_{w=1}^{|W|} x_{rwn}, \forall r,n\in R,N, (17) y_{rn}\leq\sum_{w=1}^{|W|} x_{rwn}, \forall r,n\in R,N$$

$$(18) B.y_{rn}\geq\sum_{w=1}^{|W|} x_{rwn}, \forall r,n\in R,N, (19) y_{rn}\leq\sum_{w=1}^{|W|} x_{rwn}, \forall r,n\in R,N$$

$$(20) B.s_{wn}\geq\sum_{r=1}^{|R|} x_{rwn}, \forall w,n\in W,N, (21) s_{wn}\leq\sum_{r=1}^{|R|} x_{rwn}, \forall w,n\in W,N$$

$$(22) B.k_{rn}\geq\sum_{w=1}^{|W|} g_{rwn}, \forall r,n\in R,N, (23) k_{rn}\leq\sum_{w=1}^{|W|} g_{rwn}, \forall r,n\in R,N$$

$$(24) B.r_w\geq\sum_{r=1}^{|R|}\sum_{n=1}^{N} g_{rwn}, \forall w\in W, (25) r_w\leq\sum_{r=1}^{R}\sum_{n=1}^{|N|} g_{rwn}, \forall w\in W$$

$$(26) B.e_n\geq\sum_{r=1}^{|R|} k_{rn}, \forall n\in N, (27) e_n\leq\sum_{r=1}^{|R|} k_{rn}, \forall n\in N$$

$$(28) g_{rwn}\leq x_{rwn}+x_{rwn}, \forall r,w,n\in R,W,N,$$

$$(29) g_{rwn}\geq x_{rwn}-x_{rwn}, \forall r,w,n\in R,W, N$$

$$(30) g_{rwn}\geq x_{rwn}-x_{rwn}, \forall r,w,n\in R,W, N$$

$$(31) g_{rwn}\leq 2-x_{rwn}-x_{rwn}, \forall r,w,n\in R,W, N$$

$$(32)\beta\neq\rho$$

Constraints 2 to 5 assure that RRH r will be placed in a single processing node, VPON and VDU. Constraint 6 assures that each VPON is allocated to transmit traffic to a single processing node per time. Constraint 7 verifies that each RRH r can only be processed in the cloud or in a fog node connected to it. Constraint 8 designates in which processing nodes a free wavelength can be allocated to create a VPON. Constraints 9, 10 and 11 assure that capacities of VPONs, processing nodes and the switch will be respected. Constraints 11 to 14 assure the activation of processing nodes n and VPONs w when vBBUs are allocated to them. Constraints 15 to 31 assure the activation of the internal switch and auxiliary VDUs if it is necessary. Finally, constraint 32 assures that the minimization of active VPONs and traffic switching among VDUs can not be performed together.

The *batchILP* algorithm, formally described in Algorithm 1, invokes the ILP as soon as new RRHs are activated or deactivated to re-schedule the network when there is the opportunity of vBBUs and VPONs migration. When a new RRH r is activated, a batch containing r and previously allocated RRHs is created (lines 2 to 4). After that, the ILP is executed to find an optimal solution to this batch (function `ILP(.)`, line 5). If an optimal scheduling is found, the network state is updated with the new solution (function `nfvUpdate(.)` in line 7), else, requisition r is blocked (line 9). Concomitantly, every time an RRH j is deactivated (line 10), a batch is created containing the remaining active RRHs (line 12). The ILP is executed to found a new optimal scheduling in terms of power consumption in comparison to the current network state (line 13). If a new optimal solution is found, the network state is update with the new scheduling (line 15). Else, if the returned solution is not more efficient than the current network scheduling, the network is not re-scheduled, i.e., the migration and reconfiguration of vBBUs and VPONs are not performed (line 17).

Algorithm 1 batchILP

Input: Requisition of RRH r, Deactivated RRH j, Set of previously allocated RRHs *B*, Network state and current scheduling of the network N_{state}

Output: Optimal allocation of r and migration and reconfiguration of vBBUs and VPONs to the cloud

1: **while** True **do**
2: **if** RRH r was activated **then**
3: Create a *batch* containing new requisition r and previously allocated RRHs
4: *B'*←*B* + r
5: Execute *ILP(B')*
6: **if** Is there an optimal scheduling to *B'*? **then**
7: nfvUpdate(N_{state})
8: **else**
9: Blocks r
10: **if** RRH j was deactivated **then**
11: Create a *batch* containing the remaining activated RRHs
12: *B'*←*B*
13: Execute *ILP(B')*
14: **if** Is there a new optimal scheduling to *B'*? **then**
15: nfvUpdate(N_{state})
16: **else**
17: Do not perform any vBBU migration or VPON

reconfiguration

VI. Performance Evaluation

To assess the efficiency of the proposed algorithm, an ad-hoc event-driven simulator was developed in Python[1]. CPLEX V12.8.0 was integrated into our simulator by the DOCPLEX library to solve dynamic traffic scenarios instances. The computer used to run all the simulations was an Intel i7 2.2GHz, 16GB of RAM running Ubuntu 18.04.1.

The simulated CF-RAN is composed of 1 cloud node, 2 fog nodes and 64 RRHs. Each fog node is connected to 32 RRHs. The TWDM-PON has 4 wavelengths with 10Gbps of capacity. The cloud and fog nodes VDUs can process up to 8 and 4 RRHs, respectively.

The traffic load follows a central business district pattern from a period of 24 hours [9]. At the beginning of the simulation, all RRHs are deactivated and begin to be activated following a Poisson process whose mean is equal to $(e/60)$, where e is the erlang for a given hour during the simulation. The service time of each RRH is uniformly taken from (25 min., 1 hour). The proposed batch scheduling algorithm was compared to an incremental execution of the ILP model, which will be referenced as *incILP*, without load balancing capacities. The *incILP* algorithm process and searches the optimal solution for each newly activated RRH without support to batch analysis.

We evaluated the following metrics: blocking probability, power consumption, the number and the probability of vBBU migrations, down time of vBBU services, probability of finding the vBBU services interrupted, average traffic redirection latency, VDUs usage, bandwidth wastage, and the execution time of the algorithm. The vBBU migration probability is given by M_{vBBUs}/A_{vBBUs}, where M_{vBBUs} is the amount of migrated vBBUs and A_{vBBUs} is the amount of vBBUs being processed. The probability of vBBU services interruption is given by M_{time}/P_{time}, where M_{time} is the average time to migrate the vBBUs and P_{time} is the average time that RRHs were activated. The bandwidth wastage is defined as $1-(T_{cpri}/T_{vpons})$, where T_{cpri} is the total CPRI traffic being transmitted and T_{vpons} is the total amount of available bandwidth, given in function of the number of activated VPONs. Results show average values obtained from 50 executions of each scenario with a confidence level of 95%.

Fig. 5 a) shows the blocking probabilities provided by our proposed *batch* algorithm in comparison to *incILP*. Regardless the resources activation minimization policy (*minRedir* or *minVPON*), blocking probability is completely mitigated in all erlangs by the *batch* scheduling. In the hours of the day (x axis) with lowest load, the performance of the two algorithms is similar, however, a significant increasing in blocking probability can be observed in peak hours for *incILP*. This occurs because, when each RRH is individually scheduled, *incILP* algorithm tends to centralize most of the VPONs in the cloud, exhausting other wavelengths to be allocated in fog nodes more quickly. Hence, the optimal use of the network resources decreases in function of the traffic growth.

[1]The simulator is available at https://github.com/rodrigo-tinini/5gSim under GPL license.

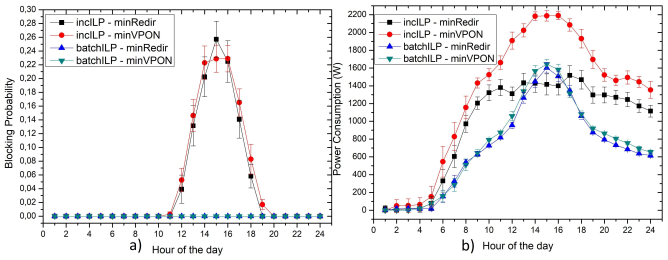

Fig. 5. a) Blocking probability b) Power consumption

Regarding power consumption, as shown in Fig. 5 b), *batchILP* shows the same behaviour for both minimization policies *minRedir* and *minVPON*. Note that *batchILP* outperforms *incILP* with *minRedir* and *minVPON* policies in at most 31% and 28% at peak hours, respectively. Note that when using the batch algorithm, *minRedir* consumes up to 5% less power than *minVPON*. This occurs because, as *minRedir* minimizes the redirection of traffic among VDUs, it activates the internal switch less often, minimizing the power cost from the use of the internal switch. Regarding *incILP*, note that there is a significant reduction of at most 37% in the power consumption by *minRedir* policy. This is explained because *minRedir* creates as many as possible VPONs in a processing node to decrease the switching latency between VDUs when using the internal switch (Section III, Fig. 2). Hence, as most wavelengths are allocated in the cloud, fog nodes can not be activated due to lack of bandwidth, and power consumption tends to be small, but with higher blocking probabilities as shown in Fig. 5 a).

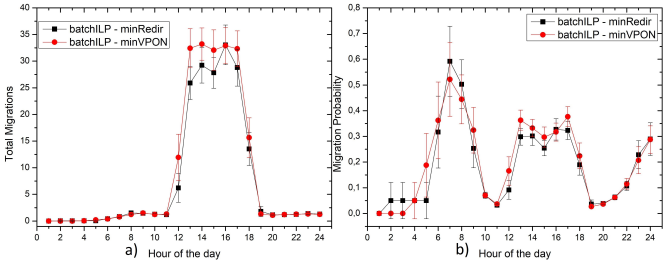

Fig. 6. a) Number of vBBU migrations b) Migrations probability

Figs. 6 a) and b) show the number and probabilities of migrations performed by *batchILP*. Note that most of migrations are done in peak load hours, which shows that the vBBUs migration plays an important role in decreasing power consumption as shown in Fig. 5 b). Policy *minRedir* provides a reduction of at most 20% on the amount and probability of migrations than *minVPON* at peak hours. As *minRedir* will create more VPONs in the cloud before activating fog nodes, this will imply in a bigger number of vBBUs placed in the cloud, which will decrease the future occurrence of vBBUs migrations to the cloud.

The average down time of vBBUs services in each hour of the day is presented in Figure 7. The time of vBBUs services interruption comes from the time to migrate a specific amount of vBBUs from a fog node to the cloud by means of Live Migration. Note that the average time that the vBBUs will be interrupted is very much small and a peak of down time is experienced by the *minVPON* policy at the peak hour of the day, specifically between 13 and 16 hours. Comparing the two minimization policies, it can be observed that the minimization of VPONs by *minVPON* doubles the down time at the peak times in comparison to *minRedir*. This shows an interesting trade off between the minimization of used wavelengths and the time to migrate traffic from fog nodes to the cloud, which can be explained by the fact that, when minimizing VPONs, more vBBUs are activated in each fog node, which leads to more vBBUs being migrated by the time the batch scheduling algorithm is executed.

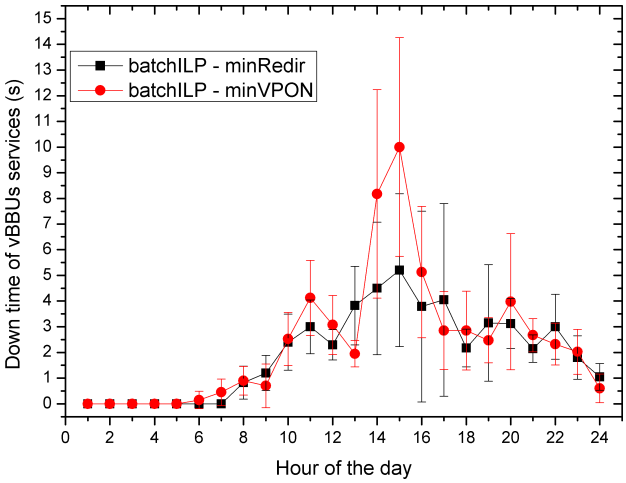

Fig. 7. Average down time of vBBUs services

The fact that vBBUs migration and VPONs reconfiguration promotes less interruption of services is also stated at Figure 8, where the probability of having the vBBUs services found interrupted is shown. Note that, in general, both policies produces similar probabilities of service interruption. Policy *minVPON* tends to produce higher probabilities in peak hours. Nevertheless, note that the probabilities are very small and at peak hours the probability of finding the services interrupted is only 1%. It is interesting to see that the probabilities tend to decrease as traffic grows. This happens because as more vBBUs are already placed in the network, less migrations possibilities can be found.

Figure 9 shows the average latency values from redirecting traffic among VDUs. Note that the redirection begins when traffic begins to grow, specifically between 10 and 18 hours. As expected, *minRedir* produces less latency than *minVPON*, being able to reduce latency in at most 1.5 times. Although there is a growth in the latency experienced in the network when traffic is redirected between VDUs, these values are very below the round-trip latency constraint of the fronthaul

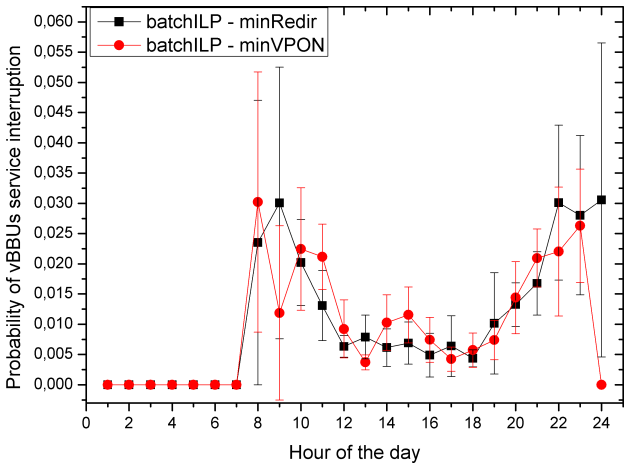

Fig. 8. Probability of vBBUs services to be found interrupted

as defined by CPRI.

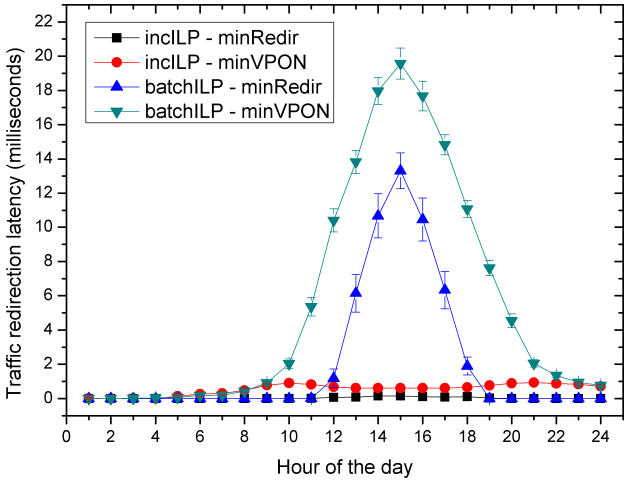

Fig. 9. Average traffic redirection latency (milliseconds)

The average usage of the VDUs processing resources is shown in Figure 10. It can be observed that the migration of vBBUs and reconfiguration of VPONs play an important role in optimizing the usage of the processing resources. The VDUs usage is optimized in at most 28% by the batch algorithm with *minRedir* policy in comparison to *incILP*. Regarding *batchILP* with *minRedir* and *minVPON* policies, the minimization of traffic redirection by *minRedir* is able to optimize the VDUs usage in at most 17% in comparison to *minVPON*. This happens because, in order to reduce redirection of traffic, more vBBUs are placed in the same VDUs.

Regarding bandwidth usage, in Fig. 11 a) it is possible to observe that *batchILP* algorithm provides the lower rate of bandwidth wastage. Note that *minVPON* is able to reduce bandwidth wastage in at most 57% in comparison to *incILP*, as *minRedir* only reduces it up to 25%. This happens because *minRedir* maximizes the amount of VPONs in processing nodes to decrease VDUs intercommunications, decreasing the overall CPRI traffic in each VPON. On the other hand, when

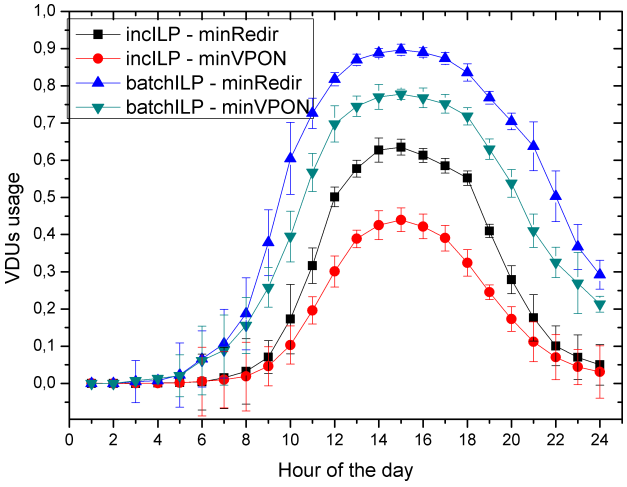

Fig. 10. Average VDUs processing resources usage

minimizing VPONs in each processing node, *minVPON* uses more efficiently the available bandwidth by increasing the overall CPRI traffic in each VPON.

Fig. 11 b) shows the execution times of the algorithms. Note that, for both *minRedir* and *minVPON* policies, *incILP* algorithm has the lowest execution time, but at the cost of worst network performance. The proposed *batchILP* algorithm shows a relatively high execution time to obtain optimal solutions for *minRedir* policy. However, when *minVPON* policy is used, the execution time is drastically reduced in at most 93% in high loaded hours. So, it is possible to note that, although providing slightly higher migration probabilities, the minimization of the created VPONs brings significant gains on the bandwidth usage and small execution times even for high network erlangs.

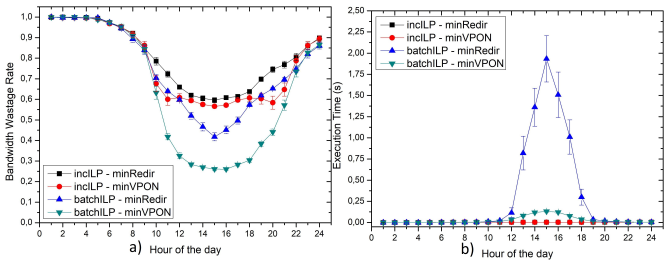

Fig. 11. a) Bandwidth wastage rate b) Execution time of the algorithms

VII. Conclusion

In this work we proposed a batch scheduling algorithm able to reconfigure VPONs and migrate vBBUs among processing nodes to support transmission and processing of baseband radio signals in a CF-RAN architecture. Our algorithm performs the placement of vBBUs and the sizing of wavelengths in a power-efficient way. Through simulations, we observed in our results that there is a strong trade-off between the minimization of intercommunication between vBBUs in a processing node and the blocking probability. Our proposed algorithm significantly outperforms an incremental scheduling algorithm and significantly reduces power consumption, blocking probability and wastage of bandwidth when reconfiguration of VPONs and migration of vBBUs are performed. We also observed that the migration of vBBUs and VPONs promotes better utilization of processing resources and that very small service interruptions are provided by our algorithm in order to promote the reconfiguration of processing and network resources. In future works we will develop heuristics to decrease the execution time and the migration probabilities of the proposed batch scheduling algorithm.

VIII. Acknowledgement

Work funded by CAPES - Finance Code 001, the INCT of the Future Internet for Smart Cities (CNPq 465446/2014-0, CAPES 88887.136422/2017-00, and FAPESP 14/50937-1 and 15/24485-9) and CNPq 311608/2017-5, 420907/2016-5 and 312324/2015-4.

References

[1] X. Wang, A. Alabbasi, and C. Cavdar, "Interplay of energy and bandwidth consumption in cran with optimal function split," in *IEEE ICC*, 2017, pp. 1–6.

[2] A. de la Oliva, J. A. Hernández, D. Larrabeiti, and A. Azcorra, "An overview of the cpri specification and its application to c-ran-based lte scenarios," *IEEE ComMag*, vol. 54, no. 2, pp. 152–159, 2016.

[3] R. I. Tinini, D. M. Batista, and G. B. Figueiredo, "Energy-efficient vpon formation and wavelength dimensioning in cloud-fog ran over twdm-pon," in *IEEE ISCC*, 2018, pp. 521–526.

[4] R. I. Tinini, D. M. Batista, G. B. Figueiredo, M. Tornatore, and B. Mukherjee, "Low-latency and energy-efficient bbu placement and vpon formation in virtualized cloud-fog ran," *IEEE/OSA JOCN*, vol. 11, no. 4, pp. B37–B48, 2019.

[5] D. Chen, S. Schedler, and V. Kuehn, "Backhaul traffic balancing and dynamic content-centric clustering for the downlink of fog radio access network," in *IEEE 17th SPAWC*, 2016, pp. 1–5.

[6] H. Hawilo, A. Shami, M. Mirahmadi, and R. Asal, "Nfv: state of the art, challenges, and implementation in next generation mobile networks (vepc)," *IEEE Network*, vol. 28, no. 6, pp. 18–26, 2014.

[7] Y. Luo, X. Zhou, F. Effenberger, X. Yan, G. Peng, Y. Qian, and Y. Ma, "Time-and wavelength-division multiplexed passive optical network (twdm-pon) for next-generation pon stage 2 (ng-pon2)," *Journal of Lightwave Technology*, vol. 31, no. 4, pp. 587–593, 2013.

[8] X. Wang, S. Thota, M. Tornatore, H. S. Chung, H. H. Lee, S. Park, and B. Mukherjee, "Energy-efficient virtual base station formation in optical-access-enabled cloud-ran," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 5, pp. 1130–1139, 2016.

[9] C. Peng, S.-B. Lee, S. Lu, H. Luo, and H. Li, "Traffic-driven power saving in operational 3g cellular networks," in *Proceedings of the 17th annual international conference on Mobile computing and networking*. ACM, 2011, pp. 121–132.

[10] G. B. Figueiredo, X. Wang, C. C. Meixner, M. Tornatore, and B. Mukherjee, "Load balancing and latency reduction in multi-user comp over twdm-vpons," in *IEEE ICC*, 2016, pp. 1–6.

[11] X. Wang, L. Wang, C. Cavdar, M. Tornatore, G. B. Figueiredo, H. S. Chung, H. H. Lee, S. Park, and B. Mukherjee, "Handover reduction in virtualized cloud radio access networks using twdm-pon fronthaul," *Journal of Optical Communications and Networking*, vol. 8, no. 12, pp. B124–B134, 2016.

[12] C. Tsai and M. Moh, "Load balancing in 5g cloud radio access networks supporting iot communications for smart communities," in *IEEE ISSPIT*, 2017, pp. 259–264.

[13] R. I. Tinini, L. C. M. Reis, D. M. Batista, G. B. Figueiredo, M. Tornatore, and B. Mukherjee, "Optimal placement of virtualized bbu processing in hybrid cloud-fog ran over twdm-pon," in *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*, Dec 2017, pp. 1–6.