Steady-State Energy Consumption Evaluation in BaseBand Units Pool in Cloud Radio Access Network

Maroua Idi[1][a], Sana Younes[1][b] and Riadh Robbana[2][c]
[1] Faculty of Sciences of Tunis, University of Tunis El Manar, Tunis 2092, Tunisia
[2] National Institute of Applied Sciences and Technology, University of Carthage, Tunis 1080, Tunisia

Keywords: C-RAN, Energy Efficiency, Hysteresis, MRM, CTMC, CSRL.

Abstract: Cloud Radio Access Network (C-RAN) have been proposed as a fifth-generation (5G) cellular network solution for high spectral and energy efficiency. In the C-RAN architecture, which leverages cloud computing technology, the baseband processing is performed in the cloud. In fact, the BaseBand Units (BBUs) are located in the cloud and generate Virtual Machines (VMs) to serve User Equipment (UE) calls. This paper performs a quantitative analysis of the energy consumption computed over two schemes named Virtual Machine Hysteresis Allocation Strategy (VMHAS) and Virtual Machine Allocation Strategy (VMAS) for 5G C-RAN. The first, VMHAS, uses the hysteresis mechanism to minimize energy consumption by adjusting the number of VMs in BBUs according to the traffic load. It consists of switching the idle VMs to sleep mode to save energy. The second, VMAS, allocates VMs without considering the sleep mode. We use the Markov Reward Model (MRM) to evaluate measures related to energy consumption in the proposed schemes. Modeling and performance measures specification are achieved by Continuous-Time Markov chains (CTMCs) and Continuous Stochastic Reward Logic (CSRL). We quantify the steady-state performance measures by checking CSRL formulas using the PRISM model checker. The obtained results demonstrate that the scheme with the hysteresis mechanism, VMHAS, achieves an enhanced energy performance compared to VMAS.

1 INTRODUCTION

Cloud Radio Access Network (C-RAN) architecture (Checko et al., 2014) has been proposed as a one of the fifth generation (5G) cellular networks for providing high spectral efficiency and conserving energy. By leveraging cloud computing technology, C-RAN implements the functional separation of traditional Base Station into two parts: the BaseBand Unit (BBU) and the Remote Radio Head (RRH). C-RAN moved all the BBUs to a central unit called the BBU pool. Meanwhile, the RRHs are left off on the cell sites. The connection between the BBUs and RRHs, respectively responsible for baseband and radio functionalities, is referred to as the fronthaul links and is done via an optical transport network.

Several researchers have focused on improving the energy efficiency of C-RAN using the ON-OFF switching (also known as sleep mode) technique. Considering that the RRHs only solely function as transmitters/receivers with minimal energy consumption, the energy conservation achieved by deactivating (sleep mode) RRHs would be restricted (Feng et al., 2017). Therefore, it is highly appealing to investigate the ON-OFF switching for BBUs. Many works (Sigwele et al., 2017) (Aldaeabool and Abbod, 2017) (Sahu et al., 2017) have studied the deactivating mechanism for BBUs in C-RAN. Nonetheless, deactivating a BBU affects all its associated RRHs and User Equipment (UE) connected to those RRHs. Hence, the activation and deactivation conditions of BBUs are interconnected with the mappings between BBU-RRH and RRH-UE. Since, in the C-RAN architecture, all BBU functions are implemented on standard hardware and executed on Virtual Machines (VMs) (Yu et al., 2017) which serve UEs. Therefore, a more accurate adjustment of the number of VMs in a BBU that can support would improve the network capacity and the energy efficiency which is the target for this paper.

This paper continues the work proposed in (Idi et al., 2022), in which a new Call Admission Con-

[a] https://orcid.org/0000-0002-6467-8887
[b] https://orcid.org/0000-0002-4883-3381
[c] https://orcid.org/0000-0001-5736-4137

Idi, M., Younes, S. and Robbana, R.
Steady-State Energy Consumption Evaluation in BaseBand Units Pool in Cloud Radio Access Network.
DOI: 10.5220/0012629400003687

In *Proceedings of the 19th International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE 2024)*, pages 99-109
ISBN: 978-989-758-696-5; ISSN: 2184-4895

trol (CAC) scheme was proposed. This scheme, called Virtual Machine Hysteresis Allocation Strategy (VMHAS), aims to adjust the number of active VMs in BBUs by making non-used VMs in sleep mode. To achieve this, VMHAS utilizes the hysteresis mechanism by dividing the VMs of BBUs into levels. Each level will be activated when the number of used VMs attains an activated threshold. Similarly, the level is deactivated when the number of used VMs is less than a deactivation threshold. Only the call-blocking probability was evaluated without considering the main performance measure related to the energy consumption that the scheme should enhance. It is the main contribution of this paper in which we focus on evaluating energy consumption. To show the effectiveness of the VMHAS scheme, we compare the system's energy consumption under this allocation strategy with a new traditional scheme, called Virtual Machine Allocation Strategy (VMAS), that does not consider the VM sleep mode.

We use the Markov Reward Model (MRM) to evaluate energy consumption measures in this work's proposed schemes. MRM (Katoen et al., 2005) is a mathematical model used to describe and analyze the behavior of a system over time. It combines a continuous-time Markov model and a set of reward functions. We model CAC schemes by Continuous-Time Markov Chains (CTMCs) (Kulkarni, 2016) and specify performance measures by Continuous Stochastic Reward Logic (CSRL) (Haverkort et al., 2002). We quantify the steady-state performance measures by checking CSRL formulas using the PRISM model checker.

The work contains the following contributions:

1. We perform a quantitative analysis of the energy consumption computed over an MRM of the hysteresis-based CAC scheme VMHAS.
2. We model a new traditional scheme VMAS with an MRM and quantify the energy consumption relative to this scheme.
3. We use the PRISM model checker to perform modeling, specification, and quantification of the steady-state reward measures of CAC schemes (VMHAS and VMAS) by checking CSRL formulas.
4. We perform a comparative analysis of the energy consumption between VMHAS and VMAS schemes. Results show that the VMHAS scheme performs better than VMAS in saving energy under low and medium traffic.

The remainder of this paper is organized as follows. Section 2 is devoted to the related work. In section 3, we introduce the MRM and CSRL logic. Section 4 provides a formal modeling of CAC schemes. Section 5 presents a formal specification of steady-state reward requirements. Section 6 gives numerical results. Finally, section 7 concludes the paper.

2 RELATED WORK

Several works are related to the energy efficiency in C-RAN. In (Sigwele et al., 2015), authors proposed a green intelligent Traffic and Resource Elastic Energy CAC scheme called iTREE. In iTREE, the number of used BBUs is reduced to equal the correct amount of the traffic load (Idle BBUs can then turn to sleep mode). To minimize the energy consumption in C-RAN, the authors proposed an approximation heuristic bin-packing algorithm. The work in (Sigwele et al., 2017) is an extension of (Sigwele et al., 2015), in which authors proposed an energy reduction model for C-RAN architecture that considers workload consolidation of BBUs located in the cloud. The idea of the model is to act with a fixed amount of BBUs and, according to the demand, deactivate the idle BBUs and reactivate them only in case of overloading. The proposed model reduces the total consumption of energy and saves resources. In (Aldaeabool and Abbod, 2017), authors proposed a strategy for switching a BBU between On/Off modes according to the traffic load in the associated RRH. In fact, they proposed a host server in the BBU pool that hosts a Modified Best Fit Decreasing (MBFD) algorithm. They formulated an optimization problem of reducing the number of BBUs with low loads by transferring them to neighboring BBUs with the available capacity. Simulation results demonstrated that the MBFD algorithm performs better than traditional ones by minimizing the number of active BBUs and the power consumption for normalized traffic load. The authors studied in (Zhang et al., 2016) the BBU pool energy consumption problem under a tidal traffic scenario. By developing heuristic algorithms, the number of active BBUs is minimized. In (Sahu et al., 2017), a scheme based on the graph method to reduce the energy consumption in the BBU pool is proposed. Simulation results show that the proposed algorithm reduces energy consumption by about 20%. In (Alhumaima et al., 2018), authors introduced the problem of allocating an optimal number of VMs to the cloud server. They used Monte Carlo-based evolutionary algorithm to reach the suboptimal number of VMs that optimizes C-RAN energy efficiency. In (Mai et al., 2023), the authors proposed an optimization model for system energy efficiency that jointly allocates multiple resources for C-RAN downlink transmission.

3 MARKOV REWARD MODEL

MRM (Katoen et al., 2005) is a mathematical model used to describe and analyze the behavior of a system over time. It combines the principles of Markov chains and rewards to quantify the expected rewards associated with different states in the system. In this work, we consider a continuous-time Markov reward model, which is a combination of a CTMC with a reward function associated with states of the chain. Formally, a CTMC $\mathcal{M}$ is a tuple $(S, \mathbf{R})$ where S is a set of *states* and $\mathbf{R} : S \times S \rightarrow \mathcal{R}^+$ is the *rate matrix* (Kulkarni, 2016).

For a CTMC, there are two types of state probabilities: transient probabilities, where the system is considered at time t, and steady-state probabilities, when the system reaches an equilibrium if it exists. Let $\Pi_s^{\mathcal{M}}(s')$ be the steady-state probability to be in state s' starting from the initial state s. If $\mathcal{M}$ is ergodic, $\Pi_s^{\mathcal{M}}(s')$ exists, and it is independent of the initial distribution that we will denote by $\Pi^{\mathcal{M}}(s')$. Let $\Pi^{\mathcal{M}}$ be the steady-state probability vector. For $S' \subseteq S$, we denote $\Pi^{\mathcal{M}}(S')$ the steady-state probability to be in states of S', $\Pi^{\mathcal{M}}(S') = \sum_{s' \in S'} \Pi^{\mathcal{M}}(s')$.

In order to specify and check performance requirements over a RMM, we use CSRL (Haverkort et al., 2002), which is an extension of Continuous Stochastic Logic (CSL) (Aziz et al., 2000) by adding constraints over rewards. CSL is a temporal logic that provides ample means to specify state and path-based performance measures for CTMC but does not support reward formulas. Therefore, we use the CSRL, which contains reward operators that refer to the stationary and transient behavior of the system under consideration. This paper concentrates on model-checking procedures for steady-state performance measures. Indeed, we will use the steady-state reward formula $\mathcal{E}_J(\phi)$, which asserts that the long run reward rate in S_ϕ (States that satisfy the formula ϕ) lies in J (J is an interval of real numbers).

Let $\rho : S \rightarrow \mathcal{R}^+$ be a *reward structure* that assigns to each state $s \in S$ a reward value $\rho(s)$. The verification of this reward formula $\mathcal{E}_J(\phi)$ requires the computation of the steady-state distribution $\Pi_s^{\mathcal{M}}$ of the considered $\mathcal{M}$.

$$s \models \mathcal{E}_J(\phi) \quad \text{iff} \sum_{s' \in S_\phi} \Pi^{\mathcal{M}}(s') \cdot \rho(s') \in J \qquad (1)$$

4 FORMAL MODELING OF THE CONSIDERED CAC SCHEMES

This paper considers a C-RAN system with a cell containing a certain number of RRHs and a centralized BBU pool of K BBUs. Each BBU comprises a set of VMs; it could serve each UE by generating a VM. We suppose that each UE has its corresponding VM i.e. a VM can only serve one UE. We also assume that all BBUs are identical: They have the same performance and they support the same number of VMs.

In the following, we will formally describe and model the two CAC schemes (VMHAS and VMAS).

4.1 Formal Model of VMHAS Scheme

In this subsection, we recall from (Idi et al., 2022) the basic concept of VMHAS and give its corresponding Markov chain rate matrix.

4.1.1 VMHAS Description

The VMHAS scheme uses the hysteresis mechanism to save energy in the C-RAN system. In fact, since the energy consumption of an idle VM (activated VM waiting for a call) is 60 to 80 percent of that of a busy VM (VM occupied by a call) (Duan et al., 2015). Therefore we use the hysteresis mechanism to divide each BBU into three levels of VMs. A VM can be in three modes: sleep, idle, or busy, as shown in Fig.1. A VM in sleep mode is a deactivated VM and consumes the lowest energy compared to its consumption in the other modes.

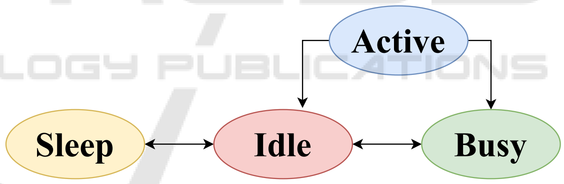


Figure 1: VM modes.

Initially, for all BBUs, we suppose that VMs in the first level are activated and are in idle mode, while the remaining VMs belonging to the other levels are in sleep mode. When a call arrives, it will be assigned to the available VM in the least-load BBU; therefore, it passes from idle to busy mode. By default, when the current traffic load is the same in all BBUs, the first BBU will serve the incoming call. When the number of VMs in busy mode reaches the first hysteresis-activated threshold (Vm_1), the second level will be activated (VMs of the second level pass from sleep mode to idle mode). Similarly, the third level in all BBUs will be activated when the number of busy VMs reaches the second hysteresis-activated threshold (Vm_2). The hysteresis mechanism supposes that the deactivation thresholds differ from the activation thresholds to avoid redundant activation/deactivation of levels containing only VMs in idle mode. These thresholds: T_1 for the deactivation of the second level and T_2 for the deactivation of the third level, are respectively less than their corresponding activation levels (Vm_1 and Vm_2).

Let us recall that V_{max} is the maximum number of VMs in a BBU, and K is the total number of BBUs. We assume that the call arrival process follows the Poisson distribution with the following mean rates: λ. We suppose that the holding time of VM is exponentially distributed with mean $1/\mu$ and independent from the arrival process.

Based on these assumptions, the VMHAS scheme (Idi et al., 2022) is modeled by a multidimensional homogeneous CTMC. In the case of K-BBUs, a state of the CTMC is represented by the tuple $(i_1, i_2 \cdots, i_k, \cdots, i_K, l)$, where i_k represents the number of VMs in the busy mode in the kth BBU, where $1 \leq k \leq K$. l represents the activated level of VMs which can be equal to 1, 2, or 3.

4.1.2 Markov Chain for Two BBUs

To represent the VMHAS model, we use a labeled Markov model in the case of two BBUs ($K = 2$), as VMHAS is modeled by CTMC consisting of a large number of components, where each dimension represents the number of active VMs in the BBU. We recall from (Idi et al., 2022) the Markov state space of VMHAS. We then present their rate matrix in the particular case of two BBUs.

Markov Chain State Space. In the case of two BBUs, the VMHAS scheme is modeled by a three-dimensional CTMC. The obtained CTMC is composed of three blocks relative to the number of levels in the hysteresis mechanism. Therefore, the state space S' is composed of three subsets: S'_1, S'_2 and S'_3. The subset S'_1 (resp. S'_2) (see Eq.2 and Eq.3) contains states relatives to the activation of the first (resp. the second) level of VMs. S'_3 (see Eq.4) contains states relatives to the activated VMs of the third block.

Therefore, the whole state space is given by:

$$S' = S'_1 \cup S'_2 \cup S'_3$$

In state (i, j, l), i (resp. j) represents the number of busy VMs in BBU_1 (resp. BBU_2), and l represents the activate level of VMs (1, 2 or 3):

$$S'_1 = \{(i,j,1); 1 \leq i \leq Vm_1 \ \ and \ \ 0 \leq j \leq i-1\} \cup \{(i,j,1); 0 \leq j \leq Vm_1 - 1 \ and \ 0 \leq i \leq j\} \quad (2)$$

$$S'_2 = \{(i,j,2); T_1 \leq i \leq Vm_2 \ \ and \ \ 0 \leq j \leq i-1\} \cup \{(i,j,2); T_1 \leq j \leq Vm_2 - 1 \ \ and \ \ 0 \leq i \leq j\} \quad (3)$$

$$S'_3 = \{(i,j,3); T_2 \leq i \leq V_{max} \ \ and \ \ 0 \leq j \leq i-1\} \cup \{(i,j,3); T_2 \leq j \leq V_{max} \ and \ 0 \leq i \leq j\} \quad (4)$$

Markov Chain Rate Matrix. Transition rates $\mathbf{R}'(s; (\bar{i}, \bar{j}, \bar{l}))$ from the state $s = (i, j, l)$ to the state $(\bar{i}, \bar{j}, \bar{l})$ are defined as:

- Markov Rates due only to the arrival of calls.
  - Rates when the first level is activated.
    - ∗ A call arrives and assigned to a VM in the first BBU.

      $\mathbf{R}'(s = (i,j,1); (i+1,j,1)) = \lambda$ if $\{0 \leq i \leq j \leq Vm_1 - 1\}$
    - ∗ A call arrives and assigned to a VM in the second BBU.

      $\mathbf{R}'(s = (i,j,1); (i,j+1,1)) = \lambda$ if $\{1 \leq i \leq Vm_1 - 1; 0 \leq j \leq i-1\}$ **or** $\{i = Vm_1; 0 \leq j \leq Vm_1 - 2\}$
    - ∗ A call arrives and assigned to a VM in the second BBU and the second level of VMs is activated.

      $\mathbf{R}'(s = (i,j,1); (i,j+1,2)) = \lambda$ if $\{i = Vm_1; j = Vm_1 - 1\}$
  - Rates when the first and the second level are activated.
    - ∗ A call arrives and assigned to a VM in the first BBU.

      $\mathbf{R}'(s = (i,j,2); (i+1,j,2)) = \lambda$ if $\{0 \leq i \leq j; T_1 \leq j \leq Vm_2 - 1\}$
    - ∗ A call arrives and assigned to a VM in the second BBU.

      $\mathbf{R}'(s = (i,j,2); (i,j+1,2)) = \lambda$ if $\{T_1 \leq i \leq Vm_2 - 1; 0 \leq j \leq i-1\}$ **or** $\{i = Vm_2; 0 \leq j < Vm_2 - 2\}$
    - ∗ A call arrives and assigned to a VM in the second BBU and the third level of VMs is activated.

      $\mathbf{R}'(s = (i,j,2); (i,j+1,3)) = \lambda$ if $\{i = Vm_2; j = Vm_2 - 1\}$
  - Rates when all levels are activated.
    - ∗ A call arrives and assigned to a VM in the first BBU.

      $\mathbf{R}'(s = (i,j,3); (i+1,j,3)) = \lambda$ if $\{0 \leq i \leq j; T_2 \leq j \leq V_{max}\}$

  * A call arrives and assigned to a VM in the second BBU.
    $\mathbf{R}'(s=(i,j,3);(i,j+1,3)) = \lambda$ if $\{T_2 \leq i \leq V_{max}; 0 \leq j \leq i-1\}$ **or** $\{i = V_{max}; 0 \leq j < V_{max}-1\}$

- Markov Rate due only to the departure of calls.
  - Rates when the first level is activated.
    * A VM is released from the first BBU.
      $\mathbf{R}'(s=(i,j,1);(i-1,j,1)) = i\mu$ if $\{0 < i \leq Vm_1; 0 \leq j \leq Vm_1 - 1\}$
    * A VM is released from the second BBU.
      $\mathbf{R}'(s=(i,j,1);(i,j-1,1)) = j\mu$ if $\{0 < i \leq Vm_1; 0 \leq j \leq Vm_1 - 1\}$
  - Rates when the second level is activated.
    * A VM is released from the first BBU.
      $\mathbf{R}'(s=(i,j,2);(i-1,j,2)) = i\mu$ if $\{0 < i \leq j; T_1 \leq j \leq Vm_2 - 1)\}$ **or** $\{T_1 < i \leq Vm_2; 0 \leq j \leq i-1\}$
    * A VM is released from the second BBU.
      $\mathbf{R}'(s=(i,j,2);(i,j-1,2)) = j\mu$ if $\{T_1 \leq i \leq Vm_2; 0 < j \leq i-1\}$ **or** $\{0 \leq i \leq j; T_1 < j \leq Vm_2 - 1\}$ **or** $\{i = T_1; j = T_1\}$
    * Following the departure of a call, the second level of VMs is deactivated.
      $\mathbf{R}'(s=(i,j,2);(i,j-1,1)) = j\mu$ if $\{0 \leq i < T_1; j = T_1\}$
      $\mathbf{R}'(s=(i,j,2);(i-1,j,1)) = i\mu$ if $\{i = T_1; 0 \leq j < T_1\}$
  - Rates when all levels are activated.
    * A VM is released from the first BBU.
      $\mathbf{R}'(s=(i,j,3);(i-1,j,3)) = i\mu$ if $\{0 < i \leq j; T_2 \leq j \leq V_{max}\}$ **or** $\{T_2 < i \leq V_{max}; 0 \leq j \leq i-1\}$
    * A VM is released from the second BBU.
      $\mathbf{R}'(s=(i,j,3);(i,j-1,3)) = j\mu$ if $\{i = T_2; j = T_2\}$ **or** $\{T_2 \leq i \leq V_{max}; 0 < j \leq i-1\}$ **or** $\{0 \leq i \leq j; T_2 < j \leq V_{max}\}$
    * Following the departure of a call, the third level of VMs is deactivated.
      $\mathbf{R}'(s=(i,j,3);(i,j-1,2)) = j\mu$ if $\{0 \leq i < T_2; j = T_2\}$
      $\mathbf{R}'(s=(i,j,3);(i-1,j,2)) = i\mu$ if $\{i = T_2; 0 \leq j < T_2\}$

4.2 Formal Model of VMAS Scheme

In order to assess the effectiveness of the VMHAS scheme, we have chosen to compare it with the VMAS model, which does not consider VM sleep mode. Indeed, in the VMAS scheme, all the VMs in all BBUs are always activated. Therefore, when a call arrives, it will be assigned to the available VM in the least-load BBU. By default, when the current traffic load is the same in all BBUs, the first BBU will serve the incoming call.

4.2.1 Algorithmic Description

The algorithmic description of the VMAS scheme in the case general of K-BBUs is presented in Algorithm 1. Notes that the arrival of different calls and the departure of ongoing calls cannot trigger simultaneously due to the Markovian hypothesis.

4.2.2 Markov Chain of VMAS

In the case of K-BBUs, a state of the CTMC is represented by $(i_1, i_2 \cdots, i_k, \cdots, i_K)$, where i_k represents the number of VMs in the busy mode in the kth BBU, $1 \leq k \leq K$. Let us remark that a state in VMAS does not contain the l component (as for VHMAS) because a BBU is considered with one level of VMs that are always activated.

Similarly to the VMHAS scheme, we formally present the Markov chain of the VMAS in the case of two BBUs. It is presented in Fig.2 and defined with the following state space:

$$\underline{S'} = \{(i,j); 1 \leq i \leq V_{max} \ and \ 0 \leq j \leq i-1\} \cup \{(i,j); 0 \leq j \leq V_{max} \ and \ 0 \leq i \leq j\} \quad (5)$$

For VMAS, the transition rates $\underline{\mathbf{R}'}(s;(\bar{i},\bar{j}))$ from the state $s=(i,j)$ to the state $(\bar{i},\bar{j})$ are defined as:

- A call arrives and assigned to a VM in the first BBU.
  $\underline{\mathbf{R}'}((i,j);(i+1,j)) = \lambda$ if $\{0 \leq i \leq j < V_{max}\}$
- A call arrives and assigned to a VM in the second BBU.
  $\underline{\mathbf{R}'}((i,j);(i,j+1)) = \lambda$ if $\{1 \leq i \leq V_{max}; 0 \leq j \leq i-1\}$
- A VM is released from the first BBU.
  $\underline{\mathbf{R}'}((i,j);(i-1,j)) = i\mu$ if $\{0 < i \leq V_{max}; 0 \leq j \leq V_{max}\}$
- A VM is released from the second BBU.
  $\underline{\mathbf{R}'}((i,j);(i,j-1)) = j\mu$ if $\{0 \leq i \leq V_{max}; 0 < j \leq V_{max}\}$

Algorithm 1: Proposed algorithm of VMAS scheme in the case of K-BBUs.

K: Total number of BBUs;
V_{max}: Max number of VMs in a BBU;
V_k: Number of VMs in busy mode in the k^{th} BBU where ($1 \le k \le K$);
/* All BBUs are empty. */
for $(1 \le k \le K)$ **do**
  $V_k = 0$
/* Two call verification methods: arrival and departure. */
$CallA = CallArrived()$;
$CallD = CallDeparture()$;
while $(CallA \;\; or \;\; CallD)$ **do**
  if $CallA$ **then**
    /* Put the call in the first BBU. */
    if $(\forall\ 1 \le k \le K, V_1 = V_2 ... = V_K)$ **then**
      $V_1 = V_1 + 1$;
    else
      /* Put the call in the least loaded BBU. */
      if $(\forall\ 1 \le k \le K, V_k < V_{max})$ **then**
        $V_l \ = \ min\, V_k$;
        $V_l = V_l + 1$;
      else
        /* Call rejected: all VMs in all BBUs occupied $(\forall\ \ 1 \le k \le K, V_k = V_{max})$. */
        Reject call ;
  if $CallD$ **then**
    /* Function returning a BBU from which a call departed. */
    $k = departedcall()$;
    $V_k = V_k - 1$;
  $CallA = CallArrived()$;
  $CallD = CallDeparture()$;

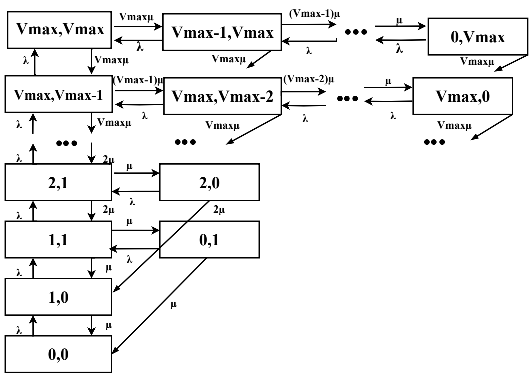


Figure 2: CTMC of the proposed VMAS.

5 FORMAL SPECIFICATION OF STEADY-STATE REWARD REQUIREMENTS

We use $\mathcal{E}_{=?}(true)$ formula, which belongs to CSRL logic (Haverkort et al., 2002), to express QoS requirements. Hence, we enrich CTMC models that we developed on PRISM with reward functions. The verification of the reward formula $\mathcal{E}_J(\phi)$ requires the computation of the steady-state distribution $\Pi^{\mathcal{M}}$ of the considered CTMC (see Eq.1).

5.1 Specification for VMHAS Scheme

In order to check steady-state reward formulas to quantify the steady-state energy consumption (for BBU_1, BBU_2, and the whole system), the mean number of idle VMs, and the mean number of sleep VMs: we enrich PRISM with the following reward functions. These reward functions are written for K-BBUs and 2-BBUs.

5.1.1 Energy Consumption

We note by E_s the energy consumption of a sleepy VM, E_i the energy consumption of an idle VM, E_b the energy consumption of a busy VM, and E_a the additional energy consumption of a VM caused by its activation.

In the case of K-BBUs, the energy consumption function E for the proposed VMHAS is defined by: $E : S \to R_{\geq 0}, \forall s = (i_1, i_2 \cdots, i_k, \cdots, i_K, l) \to E(s)$, where $E(s)$ associated to s is equal to:

$$E(s) = \begin{cases} E_b \sum_{k=1}^{K} i_k + E_i(KVm_1 - \sum_{k=1}^{K} i_k) \\ \qquad + \ E_s K(V_{max} - Vm_1) \text{ if } \ l = 1 \\ E_b \sum_{k=1}^{K} i_k + E_i(KVm_2 - \sum_{k=1}^{K} i_k) \\ \qquad + \ E_s K(V_{max} - Vm_2) \ \text{ if } \ l = 2 \\ E_b \sum_{k=1}^{K} i_k + E_i(KV_{max} - \sum_{k=1}^{K} i_k) \ \text{ if } \ l = 3 \end{cases}$$

In order to evaluate the effect of activating VMs in each level, we associate the activation transitions of level 2 and level 3 with the cost in terms of energy. For the activation transition of level 2, we associate $KE_a(Vm_2 - Vm_1)$, and for the activation transition of level 3, we associate $KE_a(V_{max} - Vm_2)$. These additional costs are counted when evaluating the energy consumption.

In the case of 2-BBUs, the energy consumption E' of the proposed VMHAS is given as follows: $E' : S' \to R_{\geq 0}, \forall s = (i, j, l) \to E'(s)$ where $E'(s)$ is associated to the state s defined by:

$$E'(s) = \begin{cases} E_b(i+j) + E_i(2Vm_1 - (i+j)) \\ \quad + E_s 2(V_{max} - Vm_1) \text{ if } l = 1 \\ E_b(i+j) + E_i(2Vm_2 - (i+j)) \\ \quad + E_s 2(V_{max} - Vm_2) \text{ if } l = 2 \\ E_b(i+j) + E_i(2V_{max} - (i+j)) \text{ if } l = 3 \end{cases} \tag{6}$$

We associate the cost $2E_a(Vm_2 - Vm_1)$ (resp. $2E_a(V_{max} - Vm_2)$) for the activation transition of level 2 (resp. level 3).

Additionally, we are interested in evaluating the energy consumption in each BBU. Therefore we define two functions E'_1 and E'_2. $E'_1 : S' \to R_{\geq 0}, \forall s = (i,j,l) \to E'_1(s)$ which associates to each state s the energy consumption of BBU_1 in this state. Similarly, $E'_2 : S' \to R_{\geq 0}, \forall s = (i,j,l) \to E'_2(s)$ which associates to each state s the consumption energy of BBU_2 in s. These values $E'_1(s)$ and $E'_2(s)$ are defined as following:

$$E'_1(s) = \begin{cases} E_b i + E_i(Vm_1 - i) + E_s(V_{max} - Vm_1) \\ \qquad \text{if } l = 1 \\ E_b i + E_i(Vm_2 - i) + E_s(V_{max} - Vm_2) \\ \qquad \text{if } l = 2 \\ E_b i + E_i(V_{max} - i) \\ \qquad \text{if } l = 3 \end{cases} \tag{7}$$

$$E'_2(s) = \begin{cases} E_b j + E_i(Vm_1 - j) + E_s(V_{max} - Vm_1) \\ \qquad \text{if } l = 1 \\ E_b j + E_i(Vm_2 - j) + E_s(V_{max} - Vm_2) \\ \qquad \text{if } l = 2 \\ E_b j + E_i(V_{max} - j) \\ \qquad \text{if } l = 3 \end{cases} \tag{8}$$

5.1.2 Number of Idle VMs

In order to evaluate the number of idle VMs, we define the reward function related to this measure. We first give this function in the case of K-BBUs, I. Then we write the function in the case of 2-BBU, I'.

For K-BBU, $I : S \to R_{\geq 0}, \quad \forall s = (i_1, i_2 \cdots, i_k, \cdots, i_K, l) \to I(s)$. $I(s)$ is defined as following:

$$I(s) = \begin{cases} KVm_1 - \sum_{k=1}^{K} i_k & \text{if } \quad l = 1 \\ KVm_2 - \sum_{k=1}^{K} i_k & \text{if } \quad l = 2 \\ KV_{max} - \sum_{k=1}^{K} i_k & \text{if } \quad l = 3 \end{cases}$$

For 2-BBUs, $I' : S' \to R_{\geq 0}, \forall s = (i,j,l) \to I'(s)$. $I'(s)$ is defined as following:

$$I'(s) = \begin{cases} 2Vm_1 - (i+j) & \text{if } \quad l = 1 \\ 2Vm_2 - (i+j) & \text{if } \quad l = 2 \\ 2V_{max} - (i+j) & \text{if } \quad l = 3 \end{cases} \tag{9}$$

5.1.3 Number of Sleep VMs

To evaluate the mean number of sleep VMs, we define the reward function related to this measure. We give the formal definition of this function in the case of K-BBUs (L) and in the case of 2-BBUs (L').

In the case of K-BBUs, L is defined by: $L : S \to R_{\geq 0}$, $\forall s \to L(s)$ that associates to each state $s = (i_1, i_2 \cdots, i_k, \cdots, i_K, l)$, the number of sleep VMs in s:

$$L(s) = \begin{cases} K(V_{max} - Vm_1) & \text{if } \quad l = 1 \\ K(V_{max} - Vm_2) & \text{if } \quad l = 2 \\ 0 & \text{if } \quad l = 3 \end{cases}$$

For 2-BBUs and by replacing K by 2, $L' : S' \to R_{\geq 0}, \forall s = (i,j,l) \to L'(s)$. $L'(s)$ is defined by:

$$L'(s) = \begin{cases} 2(V_{max} - Vm_1) & \text{if } \quad l = 1 \\ 2(V_{max} - Vm_2) & \text{if } \quad l = 2 \\ 0 & \text{if } \quad l = 3 \end{cases} \tag{10}$$

5.2 Specification for VMAS Scheme

To check the steady-state reward formulas to evaluate the mean energy consumption and the mean number of idle VMs for the VMAS scheme, we enrich the model of VMAS, implemented in PRISM, with the following reward functions. We define these functions for K-BBUs and 2-BBUs.

5.2.1 Energy Consumption

The function related to the energy consumption for VMAS scheme in the case of K-BBUs is defined by:

$\underline{E} : \underline{S} \to R_{\geq 0}$, $\forall s \to \underline{E}(s)$ where $\underline{E}(s)$ is associated to $s = (i_1, i_2, \cdots, i_K)$ as following:

$$\underline{E}(s) = E_b \sum_{k=1}^{K} i_k + E_i(KV_{max} - \sum_{k=1}^{K} i_k)$$

In the case of 2-BBUs the reward function is:

$\underline{E'} : \underline{S'} \to R_{\geq 0}$, $\forall s \to \underline{E'}(s)$ where $\underline{E'}(s)$ is associated to $s = (i,j)$:

$$\underline{E'}(s) = E_b(i+j) + E_i(2V_{max} - (i+j)) \tag{11}$$

5.2.2 Number of Idle VMs

The reward function associated to the evaluation of the mean number of idle VMs is defined for K-BBUs and 2-BBUs as following.

We define, for K-BBUs, the reward function $\underline{I} : \underline{S} \to R_{\geq 0}$, $\forall s \to \underline{I}(s)$ that associates to each state $s = (i_1, i_2, \cdots, i_K)$ the number of idle VMs:

$$\underline{I}(s) = KV_{max} - \sum_{k=1}^{K} i_k$$

For 2-BBUs, the reward function is $\underline{I}' : \underline{S}' \to R_{\geq 0}$, $\forall s \to \underline{I}'(s)$ that associates to each state $s = (i, j)$ the number of idle VMs:

$$\underline{I}'(s) = 2V_{max} - (i + j) \tag{12}$$

6 NUMERICAL RESULTS

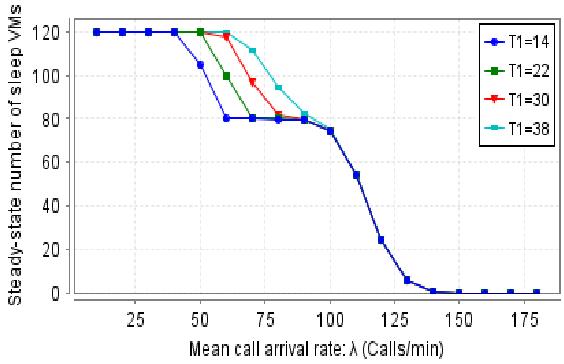
In this section, we present numerical results to show the effectiveness of VMHAS scheme. These results are obtained by verifying CSRL formulas under the VMHAS model and compared with VMAS (a model which does not consider the sleep mode). In order to construct and solve the studied models, we use the probabilistic model checker PRISM (Kwiatkowska et al., 2011). This tool is a high-level modeling language, and formulas are checked automatically. The numerical results in this section are obtained with the parameters presented in Table 1.

6.1 VMHAS Performance

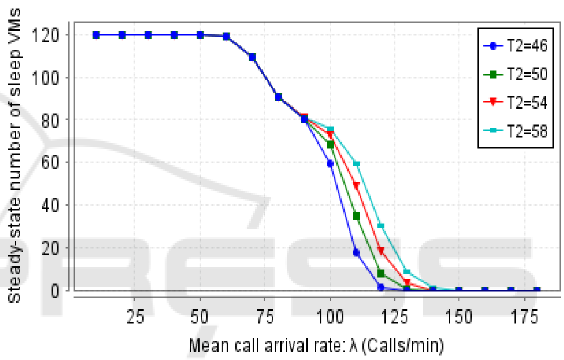
This subsection is devoted to evaluating the performance of VMHAS scheme in terms of the steady-state number of sleep VMs and the steady-state energy consumption of each BBU.

Fig.3 shows the steady-state number of sleep VMs depending on the hysteresis deactivating thresholds (T_1 and T_2) and the traffic load λ. The corresponding reward function to this measure is L', presented in Eq.10.

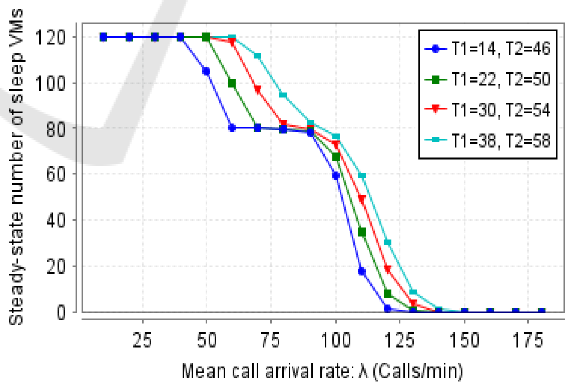
We can first remark that all curves are decreasing because when the arrival of calls increases, VMs pass from sleep mode to an activated mode to serve calls, and therefore the number of sleep VMs decreases. In Fig.3(a), we can observe that the highest curve corresponds to the highest value of T_1. This result is explained by the fact that when we increase T_1 (becomes near to Vm_1), the second level will close frequently, and therefore VMs of the second level frequently pass to the sleep mode. The same results can be seen through Fig.3(b) when varying T_2. In Fig.3(c), we illustrate the mean number of sleep VMs when we vary T_1 and T_2. We can observe that the highest curve is related to the highest values of T_1 and T_2, which confirms the results of Fig.3(a) and Fig.3(b).



(a) Varying T_1.



(b) Varying T_2.



(c) Varying T_1 and T_2.

Figure 3: Steady-state number of sleep VMs.

In order to evaluate the steady-state energy consumption in BBU_1 and in BBU_2 by considering different traffic loads, we vary the arrival rate λ of calls from 10 to 180 calls/min. The corresponding reward function to these measures are E'_1 for BBU_1 (see Eq.7) and E'_2 for BBU_2 (see Eq.8).

Table 1: Experimental Parameters.

Parameters	Value
K: Total number of BBUs.	2
$Vmax$: Max number of VMs in a BBU.	100
Vm_2: Hysteresis Level 3 activating threshold.	60
Vm_1: Hysteresis Level 2 activating threshold.	40
T_2: Hysteresis Level 3 deactivating threshold.	56
T_1: Hysteresis Level 2 deactivating threshold.	36
$1/\mu$: Mean VM holding time (per minute).	1
E_s: Energy consumption of a sleeping VM(per mJ).	0.2
E_i: Energy consumption of an idle VM (per mJ).	0.4
E_b: Energy consumption of a busy VM (per mJ).	0.5
E_a: Additional energy consumption of a VM caused by activation (per mJ).	2

Table 2: Steady-state energy consumption in each BBU.

	Steady-state energy consumption (mJ)	
Call arrival rate: λ (calls/min)	BBU_1	BBU_2
10.0	28.521550765802747	28.478449234200447
20.0	29.022981186839626	28.977018814554246
30.0	29.52356702449307	29.476432992294825
40.0	30.023902006144166	29.976125945301177
50.0	30.550625554609844	30.502443296489755
60.0	32.77501023996112	32.72654758056934
70.0	42.53185589385637	42.48318793333326
80.0	45.26724652233278	45.21842171268469
90.0	40.17329401118137	40.12434544337346
100.0	47.250198654799775	47.20114993720736
110.0	67.23262000827796	67.18348857666923
120.0	71.25853230740609	71.20933140571053
130.0	57.790673358614654	57.74141328345368
140.0	49.39830475913209	49.34899371437636
150.0	47.79158282943648	47.74222902769638
160.0	48.04006386928369	47.99069960088592
170.0	48.5055002528775	48.45631940730138
180.0	48.931289400383555	48.882897731887205

It can be observable through Table 2 that when λ does not exceed 60 calls, the energy consumption increases slightly in each BBU. This is expected because only the first level of VMs in two BBUs is activated to accept calls. However, the remaining two levels are deactivated because they are not needed. When $\lambda = 70$, the energy consumption shows a remarkable increase; this is explained by the energy consumption due to the opening of the second level of VMs (see the first two boxes: $\lambda = 70$ and $\lambda = 80$). In addition, each BBU consumes the maximum energy when the VMs of the third level are activated (see the second two boxes: $\lambda = 110$ and $\lambda = 120$).

As observed, the energy consumption of BBU_2 is slightly lower than the energy consumption of BBU_1. This slight difference is explained by the fact that when the call arrives, it will be assigned to the available VM in the least-load BBU. By default, when the current traffic load is the same in the two BBUs, the first BBU will serve the incoming call. All that shows that VMHAS schemes ensures the load balancing between BBUs.

6.2 Performance Comparison Between VMHAS and VMAS

In this subsection, we perform a comparative analysis of the steady-state number of idle VMs and the energy consumption between VMHAS and VMAS schemes.

Fig.4 illustrates the steady-state number of idle VMs (activated VMs waiting for calls) in VMHAS and VMAS schemes under different traffic loads. Their corresponding reward functions are defined respectively by I' (see Eq.9) and $\underline{I}'$ (see Eq.12). It can be seen that the curve of VMAS is always higher than that of VMHAS and is decreasing because it is a model that does not consider sleep mode. VMAS is a scheme in which VMs can be only in two modes: busy (VMs occupied by calls) or idle; therefore, this explains the negative slope of its corresponding linear curve.

For VMHAS, we can see that when the traffic load in the system does not exceed 60 calls/min, we obtain a decreasing curve because one level of VMs in two BBUs can support all UEs. The increase of idle VMs twice (see the two peaks) is caused by the activation of VMs of the second (resp. the third level). Once level three is activated, the two CAC schemes will have the same number of idle VMs because there are no VMs in sleep mode in VMHAS. Clearly, the VMHAS scheme outperforms VMAS because it adjusts the number of VMs needed to support the traffic load in the system.

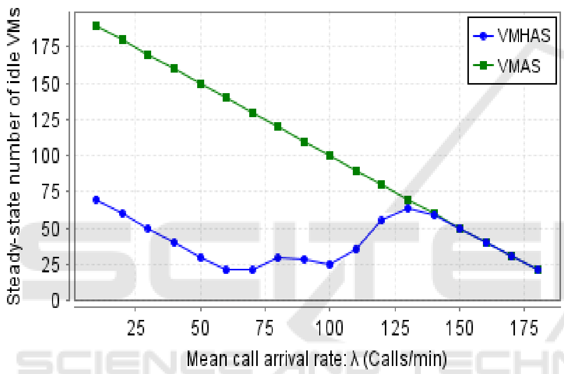


Figure 4: Steady-state number of idle VMs.

We compare the steady-state energy consumption between VMHAS and VMAS schemes by considering different traffic loads. Their corresponding reward functions are respectively defined by E' (see Eq.6) and $\underline{E'}$ (see Eq.11). It can be observable through Fig. 5 that when the traffic load increases, the mean energy consumption for both VMHAS and VMAS schemes increases. This is trivial because when the number of calls increases, the number of busy VMs increases too. The VMHAS scheme consumes less energy than the VMAS scheme due to the utilization of fewer number of VMs during the low-traffic periods (λ of calls from 10 to 65 calls per minute). On the other hand, the VMAS scheme consumes more energy than expected because all VMs in the two BBUs are always active. It is also remarkable that in the curve of the VMHAS scheme, the first peak of energy consumption is shown when the second level of VMs activated; its energy consumption stays better than the VMAS scheme. Nevertheless, as the offered traffic increases (arrival rate λ of calls is from 110 to 120), the second peak of energy consumption is shown. As a result, the curve of VMHAS becomes above that of VMAS; this is justifiable by the additional energy caused by the activation of 80 VMs. Then, from $\lambda = 150$, both schemes have the same energy consumption because all VMs are activated.

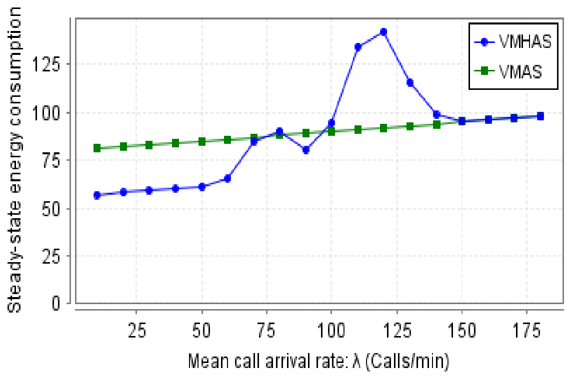


Figure 5: Steady-state energy consumption.

7 CONCLUSION

In this paper, we have performed a quantitative analysis of the energy consumption computed over two schemes named Virtual Machine Hysteresis Allocation Strategy (VMHAS) and Virtual Machine Allocation Strategy (VMAS) for C-RAN. The VMHAS scheme consists of switching the idle VMs to sleep mode to save energy, but the VMAS scheme allocates VMs without considering the sleep mode. We modeled schemes with MRMs to evaluate measures related to energy consumption. We used the PRISM model checker to perform modeling, specification, and quantification of the steady-state reward measures of VMHAS and VMAS schemes by checking CSRL formulas. Results show that the VMHAS scheme performs better than the VMAS scheme in saving energy under low and medium traffic. Furthermore, it allowed load balancing between the BBUs. In the future, we will extend this work by using Discrete-Time Markov Decision Processes (MDPs) to enhance the performance and energy efficiency in the resource allocation scheme using the hysteresis and the migration mechanisms.

REFERENCES

Aldaeabool, S. R. and Abbod, M. F. (2017). Reducing power consumption by dynamic BBUS-RRHS allocation in C-RAN. In *25th Telecommunication Forum (TELFOR)*, pages 1–4. IEEE.

Alhumaima, R. S., Ahmed, R. K., and Al-Raweshidy, H. S. (2018). Maximizing the energy efficiency of virtualized C-RAN via optimizing the number of virtual machines. *IEEE Transactions on Green Communications and Networking*, 2(4):992–1001.

Aziz, A., Sanwal, K., Singhal, V., and Brayton, R. (2000). Model-checking continuous-time markov chains. *ACM Transactions on Computational Logic (TOCL)*, 1(1):162–170.

Checko, A., Christiansen, H. L., Yan, Y., Scolari, L., Kardaras, G., Berger, M. S., and Dittmann, L. (2014). Cloud RAN for mobile networks—A technology overview. *IEEE Communications surveys & tutorials*, 17(1):405–426.

Duan, L., Zhan, D., and Hohnerlein, J. (2015). Optimizing cloud data center energy efficiency via dynamic prediction of CPU idle intervals. In *IEEE 8th International Conference on Cloud Computing*, pages 985–988. IEEE.

Feng, M., Mao, S., and Jiang, T. (2017). Base station ON-OFF switching in 5G wireless networks: Approaches and challenges. *IEEE Wireless Communications*, 24(4):46–54.

Haverkort, B., Cloth, L., Hermanns, H., Katoen, J.-P., and Baier, C. (2002). Model checking performability properties. In *Proceedings International Conference on Dependable Systems and Networks*, pages 103–112. IEEE.

Idi, M., Younès, S., and Robbana, R. (2022). Performance evaluation of call admission control strategy in cloud radio access network using formal methods. In *International Conference on Software and Data Technologies*.

Katoen, J.-P., Khattri, M., and Zapreevt, I. (2005). A markov reward model checker. In *Second International Conference on the Quantitative Evaluation of Systems (QEST'05)*, pages 243–244. IEEE.

Kulkarni, V. G. (2016). *Modeling and analysis of stochastic systems*. Chapman and Hall/CRC.

Kwiatkowska, M., Norman, G., and Parker, D. (2011). PRISM 4.0: Verification of probabilistic real-time systems. In *International conference on computer aided verification*, pages 585–591. Springer.

Mai, Z., Chen, Y., Xie, Y., and Chen, G. (2023). An energy efficiency optimization jointing resource allocation for delay-aware traffic in fronthaul constrained C-RAN. *Wireless Networks*, 29(1):353–368.

Sahu, B. J., Dash, S., Saxena, N., and Roy, A. (2017). Energy-efficient BBU allocation for green C-RAN. *IEEE Communications Letters*, 21(7):1637–1640.

Sigwele, T., Alam, A. S., Pillai, P., and Hu, Y. F. (2017). Energy-efficient cloud radio access networks by cloud based workload consolidation for 5G. *Journal of Network and Computer Applications*, 78:1–8.

Sigwele, T., Pillai, P., and Hu, Y. F. (2015). iTREE: Intelligent traffic and resource elastic energy scheme for Cloud-RAN. In *2015 3rd International Conference on Future Internet of Things and Cloud*, pages 282–288. IEEE.

Yu, N., Song, Z., Du, H., Huang, H., and Jia, X. (2017). Multi-resource allocation in cloud radio access networks. In *IEEE International Conference on Communications (ICC)*, pages 1–6. IEEE.

Zhang, J., Ji, Y., Xu, X., Li, H., Zhao, Y., and Zhang, J. (2016). Energy efficient baseband unit aggregation in cloud radio and optical access networks. *Journal of Optical Communications and Networking*, 8(11):893–901.