IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

SPECIAL SECTION ON DISTRIBUTED COMPUTING INFRASTRUCTURE FOR CYBER-PHYSICAL SYSTEMS

Received November 23, 2019, accepted December 8, 2019, date of publication December 18, 2019, date of current version December 31, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2960615

Fourth Industrial Revolution for Development: The Relevance of Cloud Federation in Healthcare Support

OLASUPO O. AJAYI, ANTOINE B. BAGULA, AND KUN MA
Department of Computer Science, University of the Western Cape, Bellville 7535, South Africa
Corresponding author: Olasupo O. Ajayi (olasupoajayi@gmail.com)

ABSTRACT Inefficient healthcare is a major concern among many African nations. It can be mitigated by building world-class infrastructure connecting different medical facilities for collaboration and resource sharing. Such infrastructure should support the exchange of medical data, enabling access to expertise not available locally. It should be equipped with technologies of the fourth industrial revolution, providing support to doctors thereby enabling African nations leapfrog from poorly equipped to medically prepared. Sadly, world-class healthcare facilities are a missing piece in African public health ecosystems. Medical facilities are either non-existent or prohibitively expensive. Being a collaborative model between Cloud providers, federated Clouds allow the execution of tasks on computing resources flexibly and cost efficiently. This paper aims to interconnect medical facilities across Africa by proposing a Cloud federation for healthcare using co-operative and competitive collaboration models. Simulations were carried out to test the efficiency of these models using two new allocation schemes: Genetic Algorithm-based VM Allocation (GAVA) and Stable Roommate Allocation (SRA). These schemes were bench-marked against First-Fit-Descending (FFD), Best-Fit-Descending (BFD), Binary-Search-Best-Fit (BSBF) allocation schemes; for both light and heavy workloads. Obtained results revealed that the co-operative model resulted in lower delays but higher resource utilization; while the competitive provided faster service delivery and better quality of service. Deployment considerations and potential business models for the African Cloud federation were also presented.

INDEX TERMS Africa, cloud computing, federated cloud, healthcare, workload allocation.

I. INTRODUCTION

Cloud computing is a key technology which plays a vital role when interfacing the physical and virtual worlds in most fields of the fourth industrial revolution (4IR). There are numerous definitions of Cloud computing in literature, however that of the NIST is arguably the most accepted. According to the NIST, Cloud computing is a model that enables pools of measurable computing resources be made available to users conveniently and ubiquitously [1]. One of the key characteristic of Cloud computing is elastic pool of resources, this implies a near infinitive resource scale. In actuality however, no Cloud Service Provider (CSP) is able to provide a limitless amount of resources to users. Beyond elasticity, Cloud resources need to be available at any time and from any location, globally. Though it is possible to achieve global coverage for a single site data center, users would however experience increased latency/delay and reduction in throughput as distance grows. Therefore, CSPs often have data centers located in multiple geographical areas to be as close to the users as possible - a concept known as multi-homing [2]. Similarly, there are situations whereby a CSP does not have sufficient resources to cater for all its users; such a situation might arise for example, during peak office hours (company websites), during promotions and sales (for e-commerce websites) or when students are resuming new academic sessions (for academic websites). Two potential solutions to this problem of resource shortage are resource scaling (either vertically or horizontally [9]) and collaboration with other CSPs. Resource scaling might however be extreme costly, especially if demand spikes are only for short duration of time. CSP collaboration on the other

The associate editor coordinating the review of this manuscript and approving it for publication was Zhen Ling.

hand might prove to be a more cost-effective solution. Cloud federation has emerged as a solution for CSP collaboration [3]. It is based on the economic model of federation game and one in which multiple CSPs combine their resources, in a way that allows for cross-utilization among themselves and improves the quality of services (QoS) rendered to users [8]. Cloud federation also provides CSPs with an extended reach, allowing them leverage on partner CSPs to reach disperse geographical locations. Cloud federation can be provided in one of three models [3], [8], which are: infrastructure pooling (where resources of multiple CSPs are aggregated together and appear as a single virtual infrastructure, similar to the disk striping or RAID0); hybrid federation, which combines resources across private and public Clouds and broker-based federation, wherein each CSP remains independent but conjoined by a single broker. The focus of this paper is on the third model. In this model, CSPs have the option of joining a federation or working independently.

A. CLOUD FEDERATION FOR HEALTHCARE SUPPORT IN AFRICA

It is widely recognized that developing nations have missed many of the opportunities offered by the first three industrial revolutions. It is also expected that, cognizant of this sad fact, many developing countries will take advantage of relevant technological offerings of the fourth industrial revolution to leapfrog from poorly equipped to technologically prepared countries. The specific 4IR use-case scenario being considered in this paper is the application of Cloud federation to healthcare and medicine across African countries. This would allow for collaboration and resource pooling across the continent for improved healthcare services. The justification for a federated Cloud for medicine in Africa are numerous, among which are: i. most African countries are either underdeveloped or developing. ii. access to world-class medical services is either non-existent or extremely expensive; however, there are a handful of African countries with good medical facilities, which can offer tele / cyber-health supports. iii. patients in many developing parts of Africa cannot afford the huge cost of flying abroad or to other African countries such as South Africa and Egypt for treatment. Cloud federation can therefore allow for collaboration, wherein resources can be pooled together for tasks such as X-Rays and CT Scans interpretations, remote testing and diagnosis, and possibly conference surgeries - where multiple experts monitor and observe surgical procedures. To put this in context, we would describe an application scenario. Currently, there are only about 75 Cloud data centers (DC) across the African countries according to [21] and Fig. 1 shows their distribution, with each bubble sized proportionality to the number of DCs in each country.

From Fig. 1 only six countries have more than five DCs, while eight countries have between one and three. This sums to only fourteen of the fifty-five countries in Africa. The other countries either do not have DCs or theirs' are below the DC standards as stipulated in [22], [23]. Building DCs and

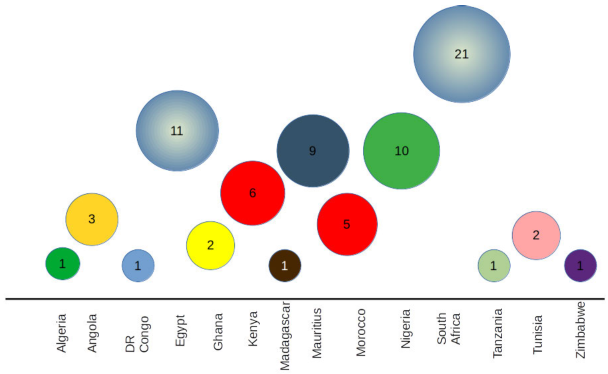


FIGURE 1. Data centers sizes across Africa.

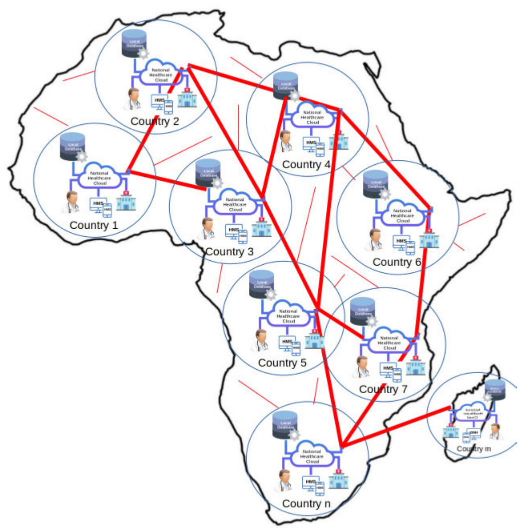


FIGURE 2. High-level conceptual cloud federation network for African Healthcare.

capacity is a very expensive and time consuming process. DCs are not a priority for many African countries, as they are often times encumbered with economic sustainability and survival challenges. Cloud federation can therefore be of immense value to these countries and the African continent in general. Fig. 2 shows a potential high-level Cloud federation network for medicine across Africa. Countries with multiple DCs are chosen as regional hubs and distributed as follows: Egypt to the North, South Africa to the South, Kenya to the East, Nigeria to the West and DRC at the center. Though DRC has only one DC, it has been selected as a hub because of its geographical position at the center of the continent. A high bandwidth, low latency network connection between these hub nations would serve as the backbone of the federated system, while the hub countries serve as regional gateways into the network.

The Federation can be done in one of two models. In the first, the CSPs agree to work together, forming a single virtualized resource pool; we refer to this as a co-operative federation model. Conversely, the CSPs can decide to work independently, we refer to this as the competitive federation model.

B. CONTRIBUTION AND OUTLINE

For this work, we considered five different workload allocations schemes to determine their effects on the co-operative or competitive Cloud federation. They are the heuristic models - First-Fit Descending, Best-Fit Descending and Binary-Search-Best-Fit; meta-heuristic model - Genetic Algorithm and the Stable Roommate Allocation economic model. Resource utilization, QoS and allocation delays were considered as performance metrics. The specific contributions of this paper are:

- A unique GA gene encoding scheme for the allocation of Cloud workloads to PMs and its implementation as contributed codes to the Cloudsim framework.
- An adapted Stable Roommate Allocation economic model, which guarantees the allocation of all Cloud workloads while using comparatively minimal Cloud resources. This was also added as an extension to Cloudsim.
- A detailed performance comparison of five different workload allocation schemes and how they affect various metrics in co-operative and competitive Cloud federations.
- Potential business models and considerations for the deployment of federated Clouds for healthcare in Africa.

The rest of this paper is organized as follows: following this introduction is a review of related work in section 2. In section 3, the Cloud federation models are presented, along side our proposed allocation schemes. Classical workload allocation schemes against which our proposed schemes are bench-marked are presented in section 4. Results of simulations done are presented and discussed in section 5, while deployment considerations and potential business models are presented in section 6. Section 7 concludes the paper with motivations for future works given.

II. RELATED WORK

With respect to collaboration across nations, a number of solutions already exist particularly in the academic and research domain. One such, is the African Research and Education Network (AfREN); which is a network established for collaboration and research in Universities and research centers across Africa [24]. It is a region based network which consists of ASREN covering the Northern and Mid-Eastern Africa, WACREN for the Western and Central Africa and UbuntuNet for the Eastern and Southern African countries. Similar networks also exists globally such as the Asia-Pacific Advanced Network (APAN) [25], GEANT [28] in Europe and internet2 [29] in the USA. A number of works have been done on providing infrastructure to support health across Africa. In the work of Bagula *et al.* [39], the authors proposed a multi-layered framework for Cyber-Physical Healthcare which combined IoT and machine learning techniques. IoT was used for the collection and muling of health data to the Cloud infrastructure, while machine learning techniques were used for patient triage. The potential advantage of this framework include better patient prioritization, better patient monitoring, cost and time savings. In a related work on IoT and healthcare, considerations for designing a full stack Remote Patient Monitoring system (RPM) for tele-medicine based on FiWARE was presented in [30]. FiWARE advocates openness and the authors proposed a solution inline with the FICHe guidelines. Critical considered to note when building such a system were given, some of which included design steps, device deployment; collection, muling, security and storage of data, as well as system integration. The authors in [4], considered the applications of Fog computing for securely storing sensitive health information in a Cyber-healthcare system and proposed the Multi-Phased Data Security and Availability (MDSA) protocol. The Fog networks helped cut down network latency, while the multi-phased security ensured end-to-end security coverage. In another related work, the authors in [40], proposed a Cloud-based medical triage service system. Upon collecting body vital signs from patients, the system analyses the information using either linear regression or k-means, bench-marking the obtained results again the WHO standard. In order to achieve collaborative healthcare system pan-African wide, standards have to be agreed upon for effective transmission and interpretation of patient medical / health records. This would foster interoperability between the various Cloud platforms spread across the continent. Lubamba and Bagula [41], had proposed a framework for the standardization of medical data. Their proposed model was based on the Health Level Seven (HL7) standard [42]. In their work, patient data had to be encoded into XML based HL7 format before being transmitted using HL7-CDA web service. From obtained results, the authors showed that their HL7 based model was able to transmit significantly more records, with minimal overhead when compared with the alternatives. Some of the ideas proposed in many of the works reviewed thus far could be applied in the implementation of healthcare kiosks in developing countries as suggested in [26], [27]. In the work done by Shimizu *et al.* [25], the authors presented medical use cases of combining the Asia-Pacific Research and Education Network (REN) with a Digital Video Transport System (DVTS). The DVTS allowed them obtain digital streams of images which could be transported via an IP network, while the REN provided a stable high-bandwidth network for transmission. A hundred different medical teleconferences were used as test, with images from live surgical sessions, endoscopy, transplants, nursing and healthcare etc. The authors in [31] also discussed the potentials and advantages of introducing tele-medicine in Africa. Some of these include: lowering medical cost, reducing geographical distance and cater for severe shortage of doctors across the African continent. Factors limiting the wide-spread adaptation of tele-medicine and possible future directions were also presented. Cloud computing has in the last two decades emerged as a reliable, robust and capable computing paradigm. It has grown beyond the single-site, single provider solution it once was to one in which multiple CSPs work together to achieve preset goals. Darzanos *et al.* [8], had proposed a model for economically evaluating Cloud federation. They focused on workload delays within federated Cloud systems. With time being the main metric, they therefore modeled each CSP as an M/M/1. In the work, the resources of the CSPs were pooled and user workloads could be served by resources belonging to any of the participating CSPs. They finally developed a model for allocation that maximized the profit of the collective whole. In a latter work [3], the authors extended on their earlier work by considering performance across three types of Cloud federation models- weak, strong and elastic. The strong being a co-operative model, the weak - a classic competitive model, while the elastic could be described as a dynamic competitive model. In this work however, profit was dependent on energy consumption and QoS adherence. Finally, the Shapley-value was used for profit sharing among the participating CSPs. With respect to our choice of allocation schemes, we can consider the allocation of workloads to Cloud resources as a bin-packing problem [16], which in itself is a NP-hard problem. This therefore necessitates the use of non-intrinsic methods to solve, such as heuristic and meta-heuristic models. In terms of the heuristic, the first-fit, best-fit and their variance are arguably the most common. For Cloud workload allocation, the Best-fit-descending (BFD) has been widely used by numerous researchers [11], [12], [15], it therefore makes an excellent choice for our selection. First-fit-descending (FFD) like BFD has been shown to use the same amount of bins, but much faster. With allocation speed being a core metric in this paper, we therefore considered FFD. In terms of meta-heuristic, the Genetic Algorithm (GA) has been widely used in many literature for workload allocation in Cloud computing environs. A few of these works are [10], [13], [14] with energy conservation, QoS and resource utilization considered as metrics. However, to the best of our knowledge GA has not been used for virtual machine (VM) migration in federated Clouds. Economic concepts have been widely applied in solving computing related problems. Coalition games and game theories have been used for problems relating and involving multiple participants - such as in VM migration in federated computing [7] and dynamic resource re-allocation in [6]. The stable marriage economic model has also been used for workload allocation in Cloud computing [5]. However, to the best of our knowledge, it has not been used to address VM migration issues in federated Cloud environments.

The focus of many of these reviewed works was either on medical collaboration via the Internet or various schemes for allocating workloads to Cloud resources. Unlike in those other works, this paper considers a Cloud federation system for improved user satisfaction and health service delivery across the African continent. Like the work of [3], this work also compares the different Cloud federation model, however, with the objective of determining which of the two models is best suited for specific requirements - light or heavy usage demands. To achieve this, we applied various workload schemes to a simulated federated Cloud, consisting of multiple CSPs working co-operatively or competitively to provide medical services across the continent.

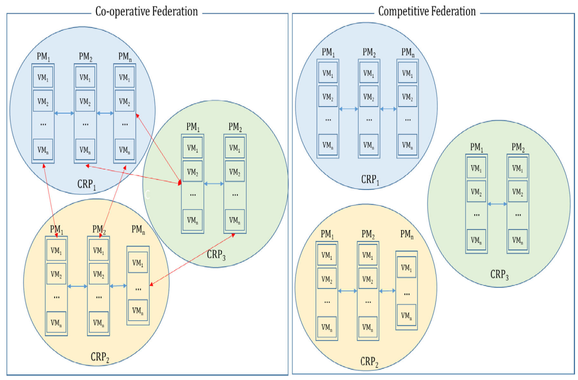


FIGURE 3. Co-operative vs competitive Cloud federation models.

III. CLOUD FEDERATION MODEL FORMULATION

Typically, a federated model for Cloud computing includes different Cloud providers collaborating by: i) sharing resources yet remaining independent entities with "thick walls" separating them; ii) having applications run "Cloud provider agnostic" through the use of virtual local networks between collaborating CSPs and iii) having Cloud providers totally disperse from each other in terms of cost and trust level.

When considering a federated Cloud environment, virtual machines allocated to tasks can be migrated either to physical resources within the same Cloud Resource Provider (CRP) or to resources of a different provider. Two federation models are being considered in this work; the competitive model, where virtual machines can only be migrated to other resources within the same provider's network and the co-operative, where migration can be to any provider. These are illustrated in Fig. 3, which reveals the migration of VMs between three CRPs in the co-operative federation. The right side of the figure, shows a localized migration for the competitive federation, with each CRP migrating its VMs within its own Cloud environment.

A. MATHEMATICAL FORMULATION

In our model, the resources which have been availed by a CRP j are expressed by $R_{pn}(j)$, while demand for resources by a VM i during migration are expressed by $D_{vn}(i)$. $\mathcal{P}$ and $\mathcal{V}$ respectively represent the set of CRPs and VMs. The federated Cloud computing problem consists of finding for each VM i in distress, a mapping to a physical resource provider j that maximizes a utility function $D(i,j)$ as defined below:

$$\max D(i,j) = \alpha(i,j) * (R_{pn}(j) - D_{vn}(i))$$
$$subject\ \ to\ \begin{cases} R_{pn}(i) \geq D_{vn}(j) & \forall i \in \mathcal{V},\ j \in \mathcal{P} \quad (1.a) \\ \alpha \in \{0, 1\} & \forall i \in \mathcal{V},\ j \in \mathcal{P} \quad (1.b) \end{cases} \tag{1}$$

Note that as expressed by equation $1.b$, $\alpha(i,j)$ is a binary parameter used in the model to differentiate between co-operative and competitive Cloud computing as expressed below:

$$\alpha(i,j) = \begin{cases} 1 & VM(i) \in PM(j) \quad \text{Co-operation} \\ 0 & VM(i) \notin PM(j) \quad \text{Competition} \end{cases} \tag{2}$$

Also note that, as expressed by 2, $\alpha(i,j)$ is used in the model to enable all participating providers be elected for VM migration under co-operative Cloud computing ($\alpha(i,j) = 1$) and prevents providers from participating in VM migration under competitive Cloud computing ($\alpha(i,j) = 0$) when such VMs do not belong to their clients.

B. ALGORITHMIC SOLUTIONS

In this work, we propose two solutions to the problem of finding suitable destination resource (PM) for VMs selected for migration. These solutions are the Genetic Algorithm based VM allocation (GAVA) and Stable Roommate Allocation (SRA). Both are described in the following subsections.

1) GENETIC ALGORITHM-BASED VM ALLOCATION (GAVA)

A number of researchers have applied GA to Cloud resource allocation, of particular note are [13], [19]. Like in those works, we also followed the steps of the classic GA described in [18], but with a different implementation. Our GAVA algorithm is as described in Algorithm 1.

We assumed each PM to be made up of two processing elements (PEs). PEs are synonymous to CPU cores. Each PM's PE is represented by gene, thus we have twice as many genes as PMs in our system. The algorithm starts off by setting all genes to 0 and iterates through the list of VMs. For each VM, a PM is randomly chosen and tested to determine if its PE can accommodate the VM's request. If it can, its corresponding gene value is set to 1, else it remains 0. This process is repeated until all VMs are assigned PMs. The obtained string of genes (chromosome) represents a potential VM-to-PM allocation solution. This process is illustrated in Fig. 4. The set of obtained chromosomes make up the population. Each chromosome has a fitness value. For this work, we took fitness value to be the total number of 0*s* in the chromosome. Therefore, the chromosome with the highest number of 0*s* (least number of 1*s*) is selected as the best. This translates to a solution which uses the least amount of PMs to cater for all Vms.

For our mutation step, we performed partial mutation and only changed 0*s* to 1*s*. This is because changing a 1 to 0 would require de-allocating all VMs currently assigned to such a PE and then searching for alternate VMs to allocate. Rather than doing this, we instead simply created a different chromosome.

Algorithm 1 GA VM Allocation (GAVA)

procedure GENE ENCODING $genes[size(PmList)] \rightarrow 0$
  for each $vm \in Vmlist$
  while true **do**
    $p \leftarrow random(PmList)$
    if $vm.getRequestedCapacity \leq p.getPE_0CapacityRemaining$ **then**
      $gene[p] \leftarrow 1$
      $p.setPE_0CapacityRemaining = (p.getPE_0CapacityRemaining - vm.getRequestedCapacity)$
      break
    if $vm.getRequestedCapacity \leq p.getPE_1CapacityRemaining$ **then**
      $gene[p+1] \leftarrow 1$
      $p.setPE_1CapacityRemaining = (p.getPE_1CapacityRemaining - vm.getRequestedCapacity)$
      break
  population.add(genes)
procedure MUTATION
  $m \rightarrow 0$
  while $m \leq mutationCount$ **do**
    $c \leftarrow random(population)$
    $i \leftarrow random(population[c].length)$
    if $c[i] == 0$ **then**
      $c[i] \leftarrow 1$
    else
      continue
procedure CROSS-OVER
  $x \leftarrow random(population[c].length)$
  From *population*, select the best two chromosomes f and s
  Swap the first x genes in f with those of s
procedure FITNESS VALUE
  $min \leftarrow genes[0].size$
  for each $g \in population$
  $fv \leftarrow CountOnes(g)$
  if $fv \leq min$ **then**
    $min \leftarrow fv$

2) STABLE ROOMMATE ALLOCATION (SRA)

For this work, the stable roommate algorithm was adapted for application in Cloud workload allocations. The stable roommate is a version of stable marriage wherein one party is allowed to have multiple partners or a room is allowed to have multiple occupants. In applying SRA, we took PMs to represent rooms/men while VMs represented the user workloads to be allocated onto PMs. A PM can have multiple VMs assigned to it, but each VM can only be assigned to a single PM. Our implementation process is shown in Algorithm 2.

Our algorithm starts off by setting up preference lists for both VMs and PMs. VMs however build a second list called the suitor list. Each PM then propose to all VMs in its preference list. VMs do not accept the proposal(s) immediate, instead each proposal is added to a suitor list. Finally, at the allocation phase, each VM cross references the content of its suitor list with its preference list and only accepts a single proposal from the PM with the highest values.

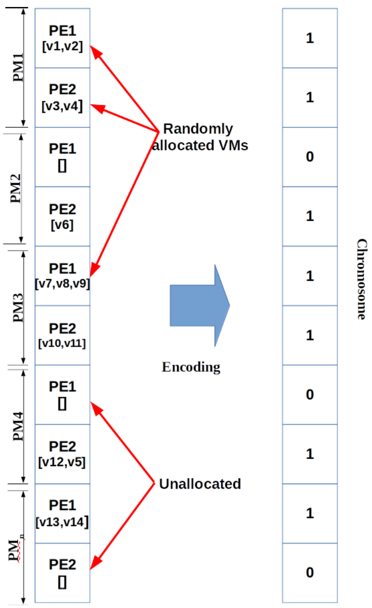


FIGURE 4. Gene encoding for GAVA.

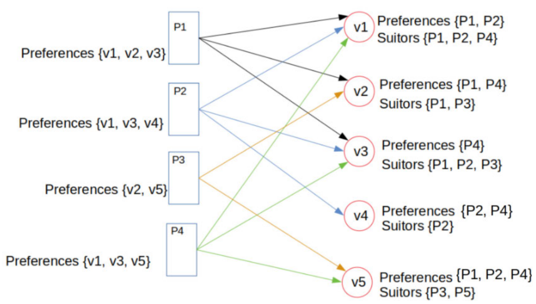


FIGURE 5. Illustration of Stable Roommate Allocation.

Algorithm 2 Stable Roommate Allocation (SRA)

procedure BUILD VM PREFERENCE LISTS
  for each $vm \in Vmlist$
  $vm.vmPrefList[] \leftarrow 0$
  $vm.vmSuitorList[] \leftarrow 0$
  for each$p \in PmList$
  if $vm.getRequestedCapacity < p.getAvailableCapacity$ **then**
    $vm.vmPrefList.add(p)$
procedure BUILD PM PREFERENCE LISTS
  for each $p \in PmList$
  $p.pmPrefList[] \leftarrow 0$
  for each $vm \in Vmlist$
  if $p.getRemainingCapacity \geq vm.getRequestedCapacity$ **then**
    $p.pmPrefList.add(vm)$
    $p.setRemainingCapacity = (p.getRemainingCapacity - vm.getRequestedCapacity)$
procedure PM PROPOSAL
  for each $p \in PmList$
  for each $vm \in p.pmPrefList$
  $p.proposeTo(vm)$
  $vm.suitorList.add(p)$
procedure WORKLOAD ALLOCATION
  for each $vm \in Vmlist$
  $best = null$
  for each $p \in vm.vmPrefList$
  if $vm.suitorList.contains(p)$ AND $p.getRemainingCapacity \geq best.getRemainingCapacity$ **then**
    $best = p$

For this algorithm, we are only concerned with ensuring that all VMs are allocated; hence, it is possible for some PM(s) not to be matched at all. This is the main difference between our work and [5]. In our model, it is desirable for some PMs to be unmatched, as this implies better resource utilization and lower energy consumption. Fig. 5 shows an illustration of the SRA with four PMs, $p_1 \ldots p_4$ and five VMs, $v_1 \ldots v_5$. The figure also shows that each VM has two lists - preference and suitor list.

C. CLOUD MIGRATION PROCESS

Fig. 6 shows the processes involved in workload migration in federated Clouds. The process starts with the allocation of workloads to Cloud resources (physical machines). The allocation is done in a way that the size of the Cloud resource j ($R_{pn}(j)$) meets or exceeds the workload i's requirement ($D_{vn}(i)$). With continuous allocation, the Cloud resource j might become unable to meet workload requirements as shown in 1, hence the need to migrate workloads to other viable resources. A monitor within the scheduler handles this process.

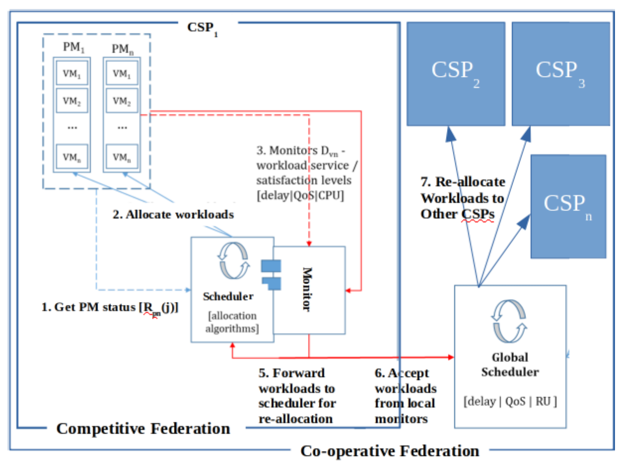


FIGURE 6. Top view of migration process in federated Cloud.

In the competitive federation, the workloads selected for migration are forwarded to the scheduler for re-allocation to other resources. This process is depicted in CSP_1 of Fig. 6. In the co-operative federation however, the selected workloads are forwarded to the global scheduler for re-allocation into a different Cloud resource within the same or different CSP. This is as illustrated on the right of Fig. 6.

IV. CLASSIC WORKLOAD ALLOCATION SCHEMES

We have described the allocation of workloads to Cloud resources as a bin packing problem, and also proffered two model solutions (GAVA and SRA). It is imperative to bench-mark these schemes against other workload allocation schemes that exist in literature. In this section, we present three classic workload allocation schemes considered in this work for "packing" workloads into servers. They are described as follows:

1) **Best-Fit Descending (BFD):** BFD is a greedy heuristic algorithm that has been shown to use $(11/9 * optimalBins) + 1$ bins [16]. When applied in Cloud computing, virtual machines (VMs) are considered items to be put in bins while the physical machines (PMs) are considered as the bins. Both the VMs and PMs are of heterogeneous sizes. The allocation speed of BFD can be increased if the PMs are sorted in order of their capacity. In this work only the CPU is considered, thus the PMs are sorted in decreasing order of CPU. This is done to allow for a uniform basis of comparison across all the different workload allocation schemes.
2) **First-Fit Descending (FFD):** FFD is another variant of the greedy heuristic algorithm but unlike BFD, it assigns VMs to the first PM it finds that can accommodate it. The performance of this algorithm can also be significantly improved if the PMs are sorted in descending order.
3) **Binary-Search Best-Fit (BSBF):** This is an algorithm proposed in [17], with the main objective of speeding up the PM search time. Rather than the linear PM search used by BFD and FFD, it instead builds a Red-Black Tree (RBT) based on PM capacities (available CPU). Being a RBT, in theory it has a worse case search time complexity of $log_2 n$ which is faster than BFD and FFD with complexities of at least n. However, there is an additional time required to build and update the RBT which also needs to be taken into consideration. Despite this additional time, BSBF was reported by the authors to still be significantly faster and conserves resources better than the other algorithms. It is for these reasons that BSBF was considered in this paper.

V. RESULTS AND DISCUSSION

For this work, simulations were carried out using Cloudsim [20] with a data center consisting of a number of heterogeneous PMs; similar to that used in [11], [15], [17], [19]. These PMs were of two categories with specifications and power consumption models based on bench-marked data from real servers [34] and given as follows: category one had 2 CPU cores clocked at 1,860MHz and 4GB of memory, while the second category had similar configuration but with CPUs clocked at 2,600MHz. To model the co-operative federated Cloud: a data center with a total of 300 PMs was setup in Cloudsim. User workloads were executed on any of four types of VMs, viz.: single core @ 2500MHz, single core @ 2000MHz, single core @ 1000MHz and single core @ 500MHz. Data used for this experiment were extracted from anonymized workload traces of VMs submitted to a Google cluster and PlanetLab. A total of 168 workload traces were used for each experiment and distributed as follows:

a) To simulate light user demands, the smallest 56 traces from the Google cluster TraceVersion1 [33] were used.
b) For heavy demands, 56 of the largest traces were extracted from PlanetLab dataset of 12th April, 2011 [32] and used.
c) For the medium demands, the 56 traces used were made up of a mix of large traces from Google cluster and light traces from PlanetLab dataset.

For the competitive federation, three data centers were set up to simulate the countries with the most number of DCs as shown in Fig. 1. For a fair and consistent result, we assigned equal number of PMs to the countries, at 100 each. Similar to the co-operative model, user workloads were split into light, medium and heavy and ran on VMs with similar configuration. In presenting the results, the performance of both federation models under light and heavy workloads were compared. Those of the medium workloads were omitted for space conservation. Six metrics were considered and obtained results are presented in subsequent subsections:

A. LIGHT WORKLOAD

1) ALLOCATION DELAY

This is a measure of how long users have to wait before processing begins on their submitted workloads. Two delays are considered in this work, pre-processing delay and average delay. In our experiments, we simply used the system time in nanoseconds to obtain the value of these delays by subtracting the start time from the end time.

a) **Pre-processing Delay:** The pre-processing delay is a measure of the time spent by each algorithm before allocating the first workload (VM). For BFD and FFD, it is the time spent sorting all PMs in descending order of available CPU. For BSBF, it is the time spent sorting the PMs in descending order plus the time spent on building the binary search tree. For GAVA, it's the time spent encoding genes, building up a population of chromosomes and iterating through 200 generations to find the best individual (VM-to-PM mapping). For SRA, it's the time spent building the preference list for both PMs and VMs as well as the time it takes each PM to propose to all its preferred VMs. Only the pre-processing times of BSBF, BFD and FFD are reported. This is because GAVA and SRA had their pre-processing done offline as they took significantly longer time to complete compared to the others. The results of pre-processing times are shown in Fig 7. From the figure the algorithms had varied pre-processing times under the two federation models. BSBF has the longest pre-processing delay for both the co-operative and competitive models at 3,943,250ns and 4,253,600ns

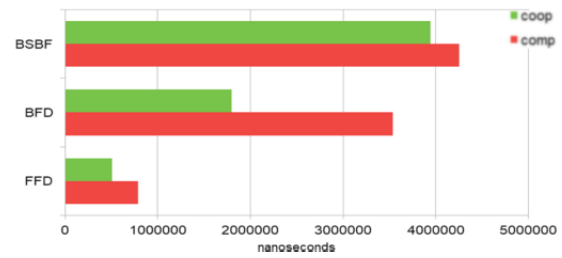


FIGURE 7. Comparison of Pre-processing Delay.

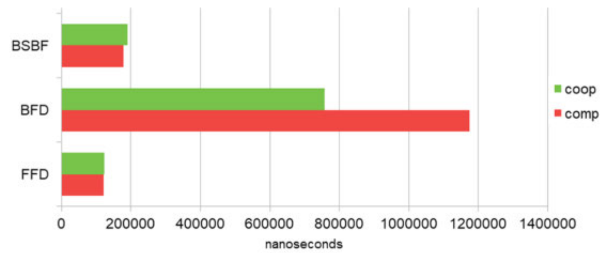


FIGURE 8. Comparison of average workload allocation Delay.

respectively. This is due to the extra time spent building the binary search tree. BFD was second, at 1,795,000ns for co-operative and 3,533,900ns for competitive. FFD was the fastest of the three at 509,300ns for co-operative federation and 785,833ns for the competitive. Cumulatively, pre-processing delays were higher in competitive than in the co-operative federation.

b) **Average Delay:** This is a measure of the average time taken to allocate a VM to a PM. The results are shown in Fig. 8. BFD took the longest time, across both federation models, at 756,459.00ns for the co-operative and 1,174,501.67ns for the competitive federation. Conversely, FFD reported the least allocation delay. For both federation models, FFD and BSBF gave almost equal delays with the competitive federation being marginally faster (less than 3,000ns) in both cases. This significant difference in speed between the algorithms can be attributed to their mode of operation. BFD searches through the entire list of PMs for one which best fits a given workload, while FFD assigns the workload to the first capable PM it finds. The benefit of the binary search tree used by BSBF is most evident here, as values of BSBF are much lower than those of BFD and almost at par with FFD. This observation is in line with results reported in [17].

2) EXECUTION TIME

In this paper, execution time is taken to mean the total time spent by a PM while serving user workloads. Equation 3 describes execution time.

$$\begin{aligned} &max \sum_{i=1}^{n} ExeV(i)p, \quad p \in P \\ &ExeT(V, P) \leq TimeLimit \end{aligned} \tag{3}$$

FIGURE 9. Comparison of overall execution time.

where $ExeV(i)p$ is execution time of a user's workload $V(i)(i \in n)$ on a PM $p(p \in P)$. This execution time should not exceed the $TimeLimit$ agreed upon in the SLA.

Fig. 9 shows a comparison of execution times of the different allocation schemes for both federation models when light workloads are submitted. For the co-operative federation, BFD resulted in the shortest execution time, followed by SRA, FFD, BSBF and finally GAVA. For the competitive federation, SRA and GAVA were the quickest, followed by FFD, BSBF and BFD. It is important to note that these time difference are only in fractions of seconds. For all algorithms, workload execution took shorter time to complete in the competitive federation than in the co-operative federation.

3) RESOURCE UTILIZATION

The five allocation algorithms were compared to determine how well they utilized resources when allocating workloads to PMs. Two results are presented, the first being resource utilization immediately after allocating workloads to Cloud resources and the second being after optimizing the allocation. Optimizing the allocation aims to reduce the number of resources used by consolidating workloads into fewer number of PMs. Utilization values were obtained by subtracting the number of idle PMs from the total number of PMs. This is simply expressed in 4

$$\begin{aligned} &min|P| - \sum_{p=1}^{n} P_{idle}(p), \quad p \in P \\ &P_{idle}(p) = \begin{cases} 1 & \text{\textit{if} } PM(p) \text{ is idle} \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{4}$$

From the results shown in Fig. 10, for both co-operative and competitive federation equal number of resources were used across all but GAVA and SRA. For GAVA, the co-operative federation was slightly better with 120 PMs versus 123 in the competitive federation; similar results were obtained for SRA with 67 PM for co-operative and 69 for the competitive federation. Across all the allocation schemes compared, BSBF resulted in the best matching of VMs to PMs and utilized only 63 PMs. BFD and FFD followed closely with 66 PMs, while GAVA resulted in the worst. It must however be noted that, this result is based on a fitness function being set to 65% utilization and 200 epochs; a lower fitness function and more

FIGURE 10. Comparison of resource utilization before consolidation.

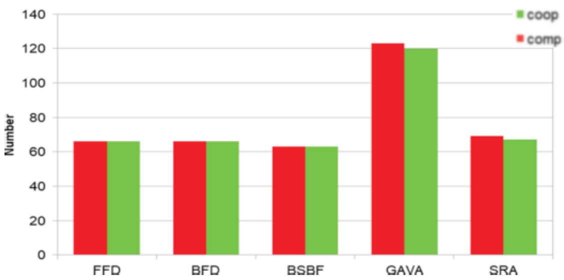


FIGURE 11. Comparison of resource utilization after consolidation.

FIGURE 12. Comparison of energy consumption before consolidation.

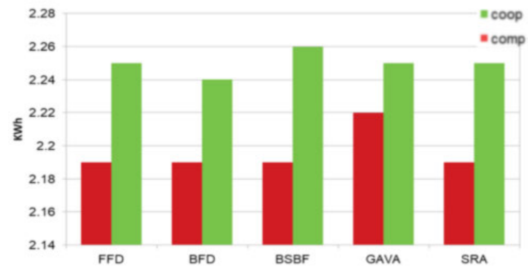


FIGURE 13. Comparison of energy consumption after consolidation.

epochs might have resulted in lowered values, though at the cost of an even longer training time. Cumulatively, the co-operative federation model was slightly better as it utilized an average of 76 units of resource compared to the 78 used in the competitive federation.

Fig. 11 shows the utilization after optimizing the allocation (consolidating workloads into fewer PMs). From the results and across all algorithms, the co-operative federation was marginally better than the competitive federation with an average of 76 versus 77 units of resource.

The main purpose for considering the resource utilization after consolidation is to determine how well each algorithm performed in terms of packing workloads into PMs. The lower the change in number of resources utilized between ‘‘before consolidation’’ and ‘‘after consolidation’’, the better the algorithms is at packing.

4) ENERGY CONSERVATION

Beyond effective resource utilization, conservation of energy is also very vital to CSPs. This is because there is a global drive to reduce energy consumption and carbon emissions for the purpose of a greener earth. Equation 5 gives a description of energy consumption of all PMs in a DC.

$$P_{tot} = \sum_{p=1}^{n} kP_{max}(p) + (1-k)P_{max}(p)U(p) \tag{5}$$

where P_{max} is the maximum power a PM p can consume in Watts. P_{tot} = Total energy of the DC. $k = 0.7$ is the fraction of power used by an idle PM. n = number of active PMs in the DC. $U(p)$ is current utilization level of a p.

Comparisons of the five algorithm with respect to energy conservation for both federation models are shown in Fig. 12 and Fig. 13. In Fig. 12 energy consumption levels were almost similar across all algorithms and for both federation models. This is in line with the resource utilization levels shown in Fig. 10. Overall, energy consumption in co-operative federation, were slightly better than values in the competitive federation model. Furthermore, BSBF with 34.8KWh conserved the most energy across both federation models. It was followed by BFD and FFD both at 35.4KWh; SRA at 36.9KWh (competitive), 35.9KWh (co-operative). GAVA had the most energy consumption at 67.4KWh and 65.0KWh for competitive and co-operative federated Clouds respectively. Results of energy utilization after consolidation are shown in Fig. 13. From the graph, the co-operative Cloud federation model resulted in higher energy consumption compared to the competitive. This held true for all the five workload allocation schemes.

5) QUALITY OF SERVICE VIOLATION

This is a measure of the ‘‘dissatisfaction index’’ of users to the allocation and services rendered to them. It is often referred to as Service Level Agreement (SLA) violation. For this work, the SLA metric used was similar to that used in [11], [15], [17] and measured as the total duration of time for which a PM is unable to satisfactory serve its assigned workload(s). Equation 6 describes the QoS violation metric.

$$Q = \frac{1}{n} * \sum_{p=1}^{n} \frac{T_{over}(p)}{T_{total}(p)} \tag{6}$$

where $T_{over}(p)$ is the duration of time during which a PM p is unable to serve its assigned workload(s) and $T_{total}(p)$ is the total duration of time during which p is active.

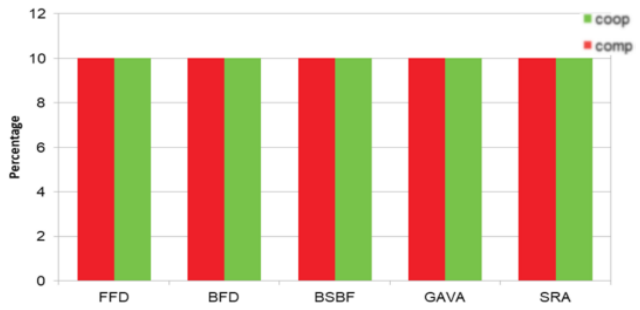


FIGURE 14. Comparison of SLA violations due to consolidation.

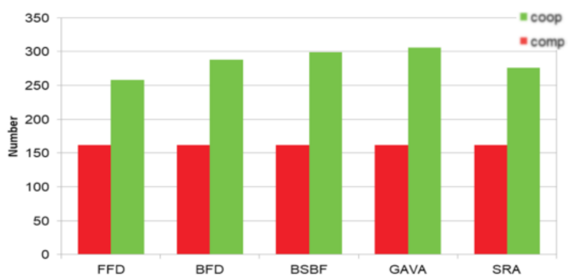


FIGURE 15. Comparison of migration counts.

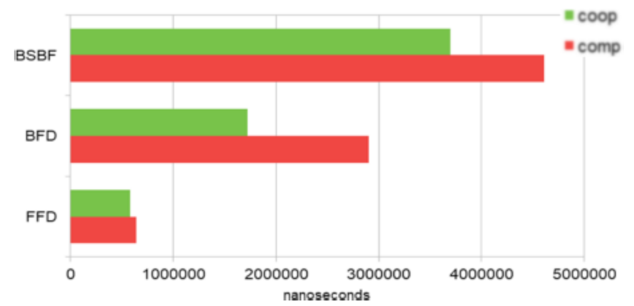


FIGURE 16. Comparison of Pre-processing Delay.

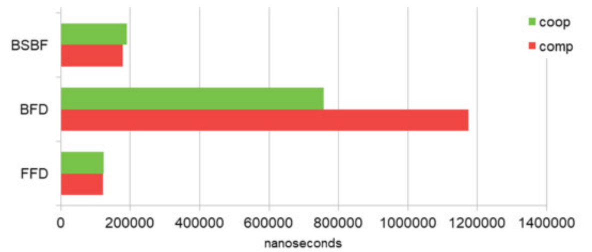


FIGURE 17. Comparison of average workload allocation Delay.

Fig. 14 shows a comparison of the average SLA violations for each of the algorithms and across the two Cloud federation models for light workloads. From the figure, violation percentage remained equal across all the algorithms and federation models. This result might be attributed to the fact that the workloads requirements were light and the PMs had more than sufficient capacity to serve them with minimal SLA violations.

6) NUMBER OF VM MIGRATIONS

The last metric considered is the migration count. It is a measure of the number of times user workloads were interrupted and moved to different PMs, either for consolidation purposes or to reduce SLA/QoS violations. The results in Fig. 15 show that user workloads were migrated more often in the co-operative federated Cloud than in the competitive federation. An explanation for this is that there are less resources (PMs) in the competitive than in the co-operative model, hence limited migration options. The number of migrations were equal for all allocation schemes under the competitive federation model. For the co-operative however, FFD resulted in the least number of migrations (258), followed by SRA with 276, BFD with 288, BSBF with 299 and GAVA with 306.

B. HEAVY WORKLOADS

1) ALLOCATION DELAY

a) **Pre-processing Delay:** From Fig. 16 and similar to Fig. 7 BSBF had the longest pre-processing delay for both the competitive and co-operative federation models at 4,611,366.67ns and 3,699,650.00ns respectively. BFD was second, at 2,905,150.00ns for competitive and 1,726,300.00ns for co-operative federation. FFD was the fastest of the three at 639,500.00ns for the competitive federation and 579,800.00ns for co-operative federation. Similar to results obtained with the light weight workloads, pre-processing delays were higher in the competitive federation model than in the co-operative model.

b) **Average Delay** Fig. 17 shows that BFD resulted in longest delay at 756,459.00ns for co-operative and 1,299,219.00ns for competitive. FFD was the fastest at 123,338.50ns for the co-operative and 111,508.00ns for competitive federation. Finally, BSBF was much faster than BFD but not as fast as FFD with 191,440.67ns for the co-operative federation model and 194,762.00ns for the competitive federation model. As observed with the light weight, workloads experienced lower delays in the co-operative versus the competitive Cloud federation model.

2) EXECUTION TIME

Fig. 18 shows a comparison of execution times of the different allocation schemes for both federation models when heavy workloads are submitted. In both the co-operative and competitive models, BSBF resulted in the quickest execution time and was closely followed by BFD. For the competitive, GAVA was the third fastest, followed by FFD and SRA; while for the co-operative federation, SRA was the third fastest, followed by FFD and GAVA. As stated above, these time differences are only in thousandth of seconds and might not be overly significant in life environments. In general and similar to the light weight workloads, execution took shorter time to complete in the competitive federation than in the co-operative federation.

FIGURE 18. Comparison of overall execution time.

FIGURE 19. Comparison of resource utilization before consolidation.

FIGURE 20. Comparison of resource utilization after consolidation.

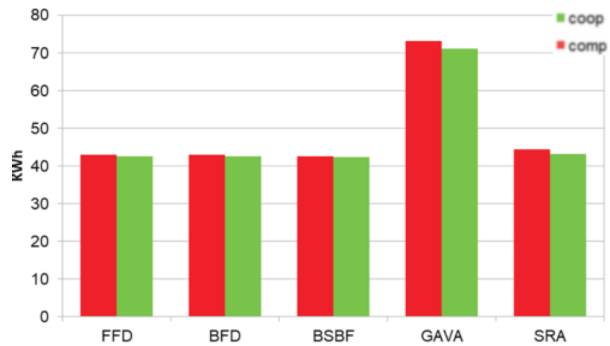


FIGURE 21. Comparison of energy consumption before consolidation.

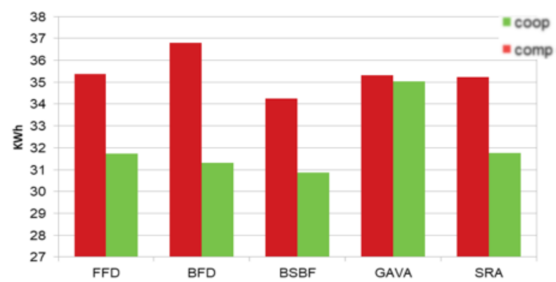


FIGURE 22. Comparison of energy consumption after consolidation.

3) RESOURCE UTILIZATION

From the results shown in Fig. 19 for both co-operative and competitive federation equal number of resources were used across all but SRA. For SRA, 67 PMs were used in the co-operative model compared to 69 in the competitive model. Like with the light weight workloads and across all allocation schemes, BSBF also resulted in the best matching of VMs to PMs and utilized the least number of PMs. BFD and FFD were second with 66 PMs, while GAVA utilized 119 PMs. Comparatively, the co-operative federation model was slightly better as it utilized an average of 76.2 units of resource compared to the 76.6 used in the competitive federation model.

Fig. 20 shows the utilization of resources after workload consolidation. BSBF utilized the least amount of resources for both federation models. BFD's initial allocation was consolidated to 18 for the competitive federation and 24 for the co-operative federation. FFD and SRA gave 24 and 26 for competitive and co-operative federations respectively, while GAVA resulted in 28 and 21 for competitive and co-operative federations respectively. Cumulatively, resource utilization was tied across both federation models.

4) ENERGY CONSERVATION

Fig. 21 and 22 show comparisons of the energy consumed when the five allocation schemes were used to allocate heavy workloads. In Fig. 21 for the competitive federation, FFD and BFD both gave similar consumption values at 42.87KWh; while BSBF consumed 42.45KWh; SRA, 44.4KWh and GAVA, 73.14KWh. For the co-operative federation, BFD resulted in the consumption of 42.42KWh of energy; FFD, 42.47KWh; BSBF, 42.25KWh, SRA, 43.1KWh and GAVA, 71.09KWh. Overall, less energy was consumed in the co-operative federation model versus the competitive model. Results of energy utilization after consolidation are shown in Fig. 22. From the graph, it can be seen that for each scheme, significantly more energy was consumed under the competitive model than in the co-operative federation model. The only exception was GAVA where the values were closer at 35.31KWh for the competitive model and 35.03KWh in the co-operative federation model.

5) QUALITY OF SERVICE VIOLATION

Fig. 23 shows a comparison of the average SLA violations for each of the schemes and across the two Cloud federation models, when heavy workloads are considered. They all performed poorly, with at least 30% SLA violation. This was expected as a large proportion of the workloads required resources that could only be provided by the PMs running

FIGURE 23. Comparison of SLA violations due to consolidation.

FIGURE 24. Comparison of migration count.

at 2,600MHz. Thus only 50% of the resources in the data center were utilized.

BSBF resulted in the least violation of all the algorithms for both the competitive and co-operative federation at 30.76% and 30.85% respectively. This was followed by GAVA with 35.05% for competitive and 40.62% for co-operative. The other results are as shown in the figure. Overall, workloads experienced higher violation in the co-operative federation than in the competitive federation.

6) NUMBER OF VM MIGRATIONS

The results in Fig. 24 show that user workloads are migrated more often in the competitive than in the co-operative Cloud federation.

C. SUMMARY OF RESULTS

Tables 1 and 2 provide a summary of the obtained results in a concise manner. On Table 1, a comparison of the two Cloud federation models is shown, alongside their performances for the various metrics considered. From the table workloads experienced lower delays in the co-operative Cloud federation but slower overall execution time compared to the competitive federation. In terms of resource utilization, the co-operative model was the better option to use for lighter workloads while the competitive was best suited for heavier workloads. When energy consumption was considered, the co-operative federation was better for heavier workloads, while the competitive was better for light weight workloads. In terms of providing satisfactory services, the competitive model was better overall, as it resulted in lower QoS violations for heavy workloads, while remaining at par with the co-operative federation for lighter workloads.

Table 2 shows the performance of the five allocation schemes across the various metrics, workload types and federation models. From the table, FFD resulted in the shortest delay, followed by BFD and BSBF. This is understandable as BFD seeks through the entire PM list for the best-fit, while BSBF has to create and constantly update its BST during the allocation process. For overall execution time, BFD was the fastest while GAVA was the slowest for the co-operative federation. For the competitive federation model, GAVA was fastest for light weight workloads while BSBF was fastest for heavier workloads. With respect to resource utilization, BSBF was the most effective and was closely followed by BFD. This can be attributed to both algorithms seeking to allocate workloads to resources that fit the best. Similar trends are observed for energy conservation (before consolidation). For energy conservation after consolidation, BSBF was better for all but light weight workloads in the co-operative federation where it marginally lost to BFD. Across both federation models, all the allocation schemes resulted in similar SLA violations for the light weight workloads. For the heavier workloads however, BSBF resulted in the least violation and was followed by GAVA, SRA, FFD and BFD. Finally, in terms of VM migrations, for the co-operative federation the SRA economic model resulted in the least number of migrations for both heavy and light weight workloads. It performed equally well in the competitive model being only slightly outperformed by BFD. BSBF and GAVA on the other hand resulted in the highest number of migrations.

VI. BUSINESS MODELS AND DEPLOYMENT CONSIDERATIONS

Often times, authors only focus on technical concepts and models without considering how such models can be actualized. Deployment constraints, sustainability and/or profitability of proposed models are often ignored. As an example and to the best of our knowledge, despite the large number of published articles on Cloud healthcare, none have looked at the mapping of medical workloads as well as business models for African healthcare. In this section, we go beyond the technical and present a mapping of health information and activities to Cloud workloads; business considerations, including opportunities and potential hurdles; and possible deployment strategies.

a) **Mapping of Medical Workloads:** The authors in [35], [36] had presented a number of ways in which Cloud computing could be applied to medicine. Some of these application areas are: preservation of medical data, medical training, medical imagery, online billing systems, medical inventory management systems etc. These are services that should be available in all standard medical facilities, however, this is not the case for hospitals in developing countries across Africa. As stated in the introductory section of this paper, Cloud federation and collaboration can help improve the quality of medical services in Africa. To put this in perspective and tie it to the models

TABLE 1. Comparison of federation model.

	Competitive Federation		Co-operative Federation	
	Light Weight Workloads	Heavy Weight Workloads	Light Weight Workloads	Heavy Weight Workloads
Pre-processing Delay	High	High	Low	Low
Average Allocation Delay	High	High	Low	Low
Execution Time	Fast	Fast	Slow	Slow
Resource Utilization	Used more resources	Used more resources	Used less resources	Used less resources
Energy Conservation	High	Low	Low	High
Quality of Service	Equal	Less violations	Equal	More violations
VM Migrations	Low	High	High	Low

TABLE 2. Performance of workload allocation schemes in federation models.

	Competitive Federation		Co-operative Federation	
	Light Weight Workloads	Heavy Weight Workloads	Light Weight Workloads	Heavy Weight Workloads
Pre-processing Delay	First: FFD; Second: BFD; Third: BSBF			
Average Allocation Delay	First: FFD; Second: BSBF; Third: BFD			
Execution Time	First: GAVA; Second: SRA, BSBF, FFD; Fifth: BFD	First: BSBF; Second: GAVA, BFD, FFD; Fifth: SRA	First: BFD; Second: SRA, FFD; Fourth: BSBF; Fifth: GAVA	First: BFD; Second: BSBF Third: SRA; Fourth: FFD; Fifth: GAVA
Resource Utilization (Before Consolidation)	First: BSBF; Second: BFD, FFD; Fourth: SRA; Fifth: GAVA			
Resource Utilization (After Consolidation)	First: BSBF; Second: BFD, FFD; Fourth: SRA; Fifth: GAVA	First: BSBF; Second: BFD; Third: FFD, SRA; Fifth: GAVA	First: BSBF; Second: BFD, FFD; Fourth: SRA; Fifth: GAVA	First: BSBF; Second: GAVA; Third: BFD; Fourth: FFD,SRA
Energy Conservation (Before consolidation)	First: BSBF; Second: BFD, FFD; Fourth: SRA; Fifth: GAVA			
Energy Conservation (After consolidation)	First: BSBF, BFD, FFD, SRA; Fifth: GAVA	First: BSBF; Second: GAVA; Third: SRA, FFD, BFD	First: BFD; Second: FFD, SRA, GAVA, BSBF	First: BSBF; Second: BFD; Third: FFD, SRA, GAVA
Quality of Service	Equal	First: BSBF; Second: GAVA; Third: SRA, FFD, BFD	Equal	First: BSBF; Second: GAVA; Third: SRA, FFD, BFD
VM Migrations	Equal	First: BFD; Second: SRA; Third: FFD; Fourth: BSBF; Fifth: GAVA	First: SRA; Second: FFD; Third: BFD; Fourth: BSBF; Fifth: GAVA	First: SRA; Second: FFD; Third: BFD; Fourth: GAVA; Fifth: BSBF

TABLE 3. Mapping of medical workloads.

Application Areas	Description	Requirements	Workload Category
Picture Archival Communication System	Storage and communication of medical images including X-Rays, CT scans, digital pathology	IaaS providing large storage and processing system.	Heavy
Medical Inventory Management System	An information system for managing digital requisition, storage, disbursement of medical related inventories.	SaaS - software solution.	Light
Online Billing System	A system for managing the hospital's billing system.	SaaS - software solution	Light
Medical Training Courses	This includes learning materials, video tutorials formatted in high definition, e-books, presentation slides, lecture notes etc.	IaaS/SaaS	Heavy
Patient Information System	Digital repository of patient's medical records.	SaaS - software solution	Light
Tele-medicine, Tele-surgery and collaborative surgery	Usually involves streaming very high definition audio/visuals	IaaS/PaaS	Heavy
Laboratory Management System	Specimen management and result processing system	SaaS - software solution	Light/Heavy
Emergency and Ambulance Management Services	A system that incorporates emergency call centers, ambulance dispatch, route planning, and triage.	PaaS - a platform to incorporate multiple systems.	Light

and results presented in this paper, these aforementioned Cloud medical applications can be grouped into heavy and light weight workloads based on our perception of data size and system (resource) requirements. Table 3 shows some potential application areas of Cloud computing in healthcare and their mappings to corresponding workload categories.

b) **Deployment Considerations:** In considering new projects, products or process, the SWOT analysis is often used by organizations as it easily identifies potential

TABLE 4. SWOT analysis of cloud federation for Healthcare in Africa.

Strengths	Weaknesses
• Improvement in level of healthcare services across the African continent. • Cost savings for patients. • Cost savings for the hospitals and country in general. • Collaboration and team work among medical practitioners across Africa. • Potential reduction in mortality rate in developing countries across Africa.	• High cost of purchasing and installing communication facilities. • There is the need to educate / train medical and support staff to use the facilities. • Human inherent resistance to change.
Opportunities	**Threats**
• Potential for economic growth of African countries. • State-of-the-art medical and technological facilities. • High scalability. • Potential to apply machine learning and artificial intelligence to discover hidden pattern which could translate to improved medical services.	• Security - a security breach could result in exposure of sensitive patient information. • Over-reliance on network and communication facilities - an outage or network downtime could be fatal especially in emergency situations or during a surgical procedure. • Diverse polices and information usage acts across counties.

TABLE 5. Potential business models for cloud federation for Healthcare.

Perspective	Federation Model	Business Model	Description	Benefits	Challenges and Limitations
Between Cloud Service Providers (Business-to -Business)	Competitive Federation	Product-based: subscription model	A CSP has to pay a monthly or annual subscription fee for access to the network of other CSPs	Easy and customizable payment plans	i. CPSs are lock-in for the duration of the contract. ii. difficult to migrate date to another CSP
		Product-based: product-as-a-service model	Payment is made each time the service is used.	Vendor lock-in is avoided	Constant data migration might compromise data integrity and confidentiality.
	Co-operative Federation	Service-based: Support model	The platform is made available to potential CSPs at a little entry fee, but revenue is made from service and support provided.	i. Low entry requirements. ii. Revenue sharing model can easily be agreed upon as there are tools that can accurately measure the resource utilization of each member CSP.	i. Profitability is dependent on the amount of service provided to users. ii. Security can be a challenge, as a security breach on a member CSP can spread across the entire co-operative network.
Between Hospitals and Cloud Service Providers	Hybrid Cloud (On-premise Cloud + Public Cloud)	Product-based: product-as-a-service model	The hospital pays each time the service is used.	If optimally used, this model can result in higher margin for the CSP and lower cost for the hospital	i.Difficult to move data to other CSPs ii. Patients suffer the most if there is a dispute between the hospital and CSP
Between Patients and Hospital	NA	Product-based: subscription model	Patients pay a monthly or annual subscription. The fee could be included as part of medical insurance bills.	i. Easy payment system. ii. Customizable payment options	i. Patient are unaware of the back-end Cloud provider and their QoS levels. ii. Patients have no way of ensuring that the hospital pays the CSP.

weaknesses and threats. It also sheds light on the unique advantages of their product as well as potential opportunities. In Table 4, various aspects of the SWOT analysis of Cloud federation for healthcare in Africa are itemized.

c) **Business Models** A number of business models for Cloud computing and related technologies have been discussed in [37], [38]. This section presents some of these business models that can be applied to Cloud federation based on a number of perspective. These models are shown on Table 5.

VII. CONCLUSION

Malnutrition, epidemic diseases and high human mortality rate are common in many African countries. Many of these are associated with the high level of poverty and poor state of infrastructure, especially those related to health. However, there are a few countries in Africa with better than average or world-class healthcare facilities. Thus an imbalance exist across African nations in terms of healthcare. A solution to this would be to build world-class hospitals in every cities across the continent, but this is prohibitively expensive. An alternative solution is to leverage on technology and Cloud computing in particular. Cloud computing has emerged as a computing paradigm that converts computing from a product to a paid service. By leveraging on the Cloud, medical expertise can be ''imported'' at a comparatively cheaper cost. Among the offering of Cloud computing is on-demand access to computing resources and cost savings. These features are however not often achievable by a single Cloud Service Provider (CSP) without adverse effect on service quality, which is pertinent to healthcare. In a bid to achieve these without compromising quality, CSPs have to collaborate and form Cloud federations. Cloud federation across the African continent can prove to be an effective solution to some of the healthcare infrastructural challenges. In this paper, two Cloud federation models were considered - the co-operative and competitive. Two new workload allocation schemes Genetic Algorithm VM Allocation (GAVA) and Stable Roommate Allocation (SRA) were presented. These schemes were compared with three class workload allocation schemes First-Fit-Descending (FFD), Best-Fit-Descending (BFD) and Binary-Search-Best-Fit (BSBF); to determine their performance and effect on co-operative federation, where participating CSPs pool resources together and on competitive federation, where participants utilize their resources independently. Service delay, resource utilization, energy conservation and adherence to Service Level Agreements (SLA) were metrics considered and experimental simulations were conducted on both light and heavy workloads. Obtained results show that the co-operative federation resulted in the least allocation delays and utilized resources better, while the competitive federation was faster in completing user tasks with lower violations on agreed service level. With respect to the allocation schemes, FFD was the fastest overall, while BSBF was the most effective for resources utilization, energy conservation and service adherence. Finally, this paper presented deployment considerations for federated Cloud for healthcare across Africa as well as various potential business models. For future works, the effect of cost and penalties associated with SLA violations might be considered as well as a hybrid combination of these algorithms in a bid to find an optimal solution. Government policies, ethical considerations and most importantly a robust network architecture for this trans-national Cloud federation for healthcare could also be looked into.

REFERENCES

[1] P. Mell and T. Grance, "The NIST definition of cloud computing," Nat. Inst. Standards Technol., Gaithersburg, MD, USA, 2009, vol. 53, no. 6, p. 50.

[2] A. Akella, B. Maggs, S. Seshan, A. Shaikh, and R. Sitaraman, "A measurement-based analysis of multihoming," in *Proc. Conf. Appl. Technol. Archit. Protocols Comput. Commun.*, Aug. 2003, pp. 353–364.

[3] G. Darzanos, I. Koutsopoulos, and G. D. Stamoulis, "Economics models and policies for cloud federations," in *Proc. IFIP Netw. Conf. IFIP Netw. Workshops*, May 2016, pp. 485–493.

[4] S. B. Akintoye, A. Bagula, and O. E. Isafiade, "Towards fog-based cyber-healthcare data storage security and availability," in *Proc. IST-Africa Conf.*, 2018.

[5] H. Xu and B. Li. *Anchor A Stable Matching Framework for Managng Cloud Resources*. Accessed: May 18, 2019. [Online]. Available: http://iqua.ece.toronto.edu/papers/matching-TR.pdf

[6] D. Irwin, J. Chase, L. Grit, L. A. Yumerefendi, D. Becker, and K. G. Yocum, "Sharing networked resources with brokered leases," in *Proc. USENIX ATC*, 2006, p. 18.

[7] L. Grit, D. Irwin, A. Yumerefendi, and J. Chase, "Virtual machine hosting for networked clusters: Building the foundations for 'Autonomic' orchestration," in *Proc. 2nd Int. Workshop Virtualization Technol. Distrib. Comput.*, 2006, p. 7.

[8] G. Darzanos, I. Koutsopoulos, and G. D. Stamoulis, "A model for evaluating the economics of Cloud federation," in *Proc. IEEE 4th Int. Conf. Cloud Netw. (CloudNet)*, Oct. 2015, pp. 291–296.

[9] K. Hwang, Y. Shi, and X. Bai, "Scale-out vs. Scale-up techniques for cloud performance and productivity," in *Proc. IEEE 6th Int. Conf. Cloud Comput. Technol. Sci.*, Dec. 2014, pp. 763–768.

[10] P. Daharwal and V. Sharma, "Energy efficient cloud computing Vm placement based on genetic algorithm," *J. Comput. Trends Technol.*, vol. 44, no. 1, pp. 15–23, 2017.

[11] A. Beloglazov and R. Buyya, "Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in Cloud data centers," *Concurrency Comput. Pract. Exper.*, vol. 24, no. 13, pp. 1397–1420, 2012.

[12] S. Banerjee, M. Adhikari, S. Kar, and U. Biswas, "Development and analysis of a new cloudlet allocation strategy for QoS improvement in cloud," *Arabian J. Sci. Eng.*, vol. 40, no. 5, pp. 1409–1425, 2015.

[13] O. Sharma and H. Saini, "Performance evaluation of VM placement using classical bin packing and genetic algorithm for cloud environment," *Int. J. Bus. Data Commun. Netw.*, vol. 13, no. 1, pp. 45–57, 2017.

[14] J. Liu, X. Luo, X. Zhang, F. Zhang, and B. Li, "Job scheduling model for Cloud computing based on multi-objective genetic algorithm," *Int. J. Comput. Sci. Issues*, vol. 10, no. 1, pp. 134–139, 2013.

[15] T. H. Nguyen, M. Di Francesco, and A. Yla-Jaaski, "Virtual machine consolidation with multiple usage prediction for energy-efficient cloud data centers," *IEEE Trans. Serv. Comput.*, to be published.

[16] W. F. de la Vega and G. Lueker, *Bin Packing Can be Solved Within 1 + ϵ Linear Time*, vol. 1. Berlin, Germany: Springer, Dec. 1981, pp. 349–355.

[17] O. O. Ajayi, F. A. Oladeji, and C. O. Uwadia, "Multi-class load balancing scheme for QoS and energy conservation in cloud computing," *West African J. Ind. Academic Res.*, vol. 17, no. 1, pp. 28–36, 2017.

[18] S. Chen, J. Wu, and Z. Lu, "A cloud computing resource scheduling policy based on genetic algorithm with multiple fitness," in *Proc. IEEE 12th Int. Conf. Comput. Inf. Technol.*, Oct. 2012, pp. 177–184.

[19] A. Mosa and N. W. Paton, "Optimizing virtual machine placement for energy and SLA in clouds using utility functions," *J. Cloud Comput.*, vol. 5, no. 1, p. 17, 2016.

[20] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. F. De Rose, and R. Buyya, "Cloudsim: A toolkit for modeling and simulation of Cloud computing environments and evaluation of resource provisioning algorithms," *Softw. Practical Exper.*, vol. 41, no. 1, pp. 23–50. 2011.

[21] *Data Center Research (n. d.) Data Center Map*. Accessed: Jul. 29, 2019. [Online]. Available: https://www.datacentermap.com/africa/

[22] W. Turner, J. H. Seader, and W. Renaud, *Data center site infrastructure tier standard: Topology*. Washington, DC, USA: Uptime Institute, 2010.

[23] C. DiMinico, and J. Jew, *Telecommunications Infrastructure Standard For Data Centers*, Standards ANSI/TIA-942. 2016.

[24] M. Foley. (2016). The role and status of National Research and Education Networks (NRENs) in Africa. SABER-ICT Technical Paper Series. World Bank, Washington, DC, USA. Accessed: Oct. 9, 2019. [Online]. Available: https://openknowledge.worldbank.org/handle/10986/26258

[25] S. Shimizu, N. Nakashima, K. Okamura, and M. Tanaka, "One hundred case studies of Asia–Pacific telemedicine using a digital video transport system over a research and education network," *Telemed. E-Health*, vol. 15, no. 1, pp. 112–117, 2009.

[26] M. Mandava, C. Lubamba, A. Ismail, H. Bagula, and A. Bagula, "Cyber-healthcare for public healthcare in the developing world," in *proc. IEEE Symp. Comput. Commun. (ISCC)*, Messina, Italy, 2016, pp. 14–19.

[27] M. Bagula, H. Bagula, M. Mandava, C. K. Lubamba, and A. Bagula, "Cyber-healthcare kiosks for healthcare support in developing countries," in *Proc. Africomm*, Dakar, Senegal, Nov. 2018, pp. 185–198.

[28] *Geant Project Home*. Accessed: Aug. 28, 2019. [Online]. Available: https://geant3plus.archive.geant.net/

[29] *Internet*. Accessed: Aug. 28, 2019. [Online]. Available: https://www.internet2.edu

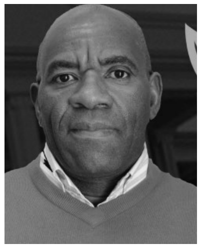
[30] A. Celesti, M. Fazio, F. G. Márquez, A. Glikson, H. Mauwa, A. Bagula, F. Celesti, and M. Villari, "How to develop IoT Cloud e-health systems based on FIWARE: A lesson learnt," *J. Sensor Actuator Netw.*, vol. 8, no. 1, p. 7, 2019.

[31] M. Mars and L. Erasmus, "Telemedicine can lower health care costs in Africa," *Innovate*, no. 7, pp. 32–33, 2012.

[32] K. Park and V. S. Pai, "Comon: A mostly-scalable monitoring system for planetlab," *SIGOPS Operating Syst. Rev.*, vol. 40, no. 1, pp. 65–74, 2006.

[33] J. Wilkes and C. Reiss. *Google Cluster Usage Traces: Format + Schema of Google Workloads*. Accessed: Mar. 19, 2019. [Online]. Available: http://code.google.com/p/googleclusterdata/
[34] D. Rice, J. Glick, D. Cercy, C. Sandifer, and B. Cramblitt. *SPECPowerssj2008*. Accessed: Jun. 24, 2019. [Online]. Available: https://www.spec.org/power_ssj2008/results/
[35] R. Daman, M. Tripathi, and S. Mishra, ''Cloud computing for medical applications & healthcare delivery: Technology, application, security and swot analysis,'' in *Proc. ACEIT Conf.*, 2016, pp. 155–159.
[36] L. Wang and C. Alexander, ''Medical applications and healthcare based on cloud computing,'' *Int. J. Cloud Comput. Services Sci.*, vol. 2, no. 4, p. 213, 2013.
[37] A. Osterwalder, Y. Pigneur, and C. Tucci, ''Clarifying business models: Origins, present, and future of the concept,'' *Commun. Assoc. For Inf. Syst.*, vol. 16, no. 1, p. 1. 2005.
[38] S. Turber, J. Brocke, O. Gassmann, and E. Flesich, ''Designing business models in the era of Internet of Things,''in *Proc. Int. Conf. Design Sci. Res. Inf. Syst.*, Miami, FL, USA, May 2014, pp. 17–31.
[39] A. Bagula, M. Mandava, and H. Bagula, ''A framework for healthcare support in the rural and low income areas of the developing world,'' *J. Netw. Comput. Appl.*, vol. 120, pp. 17–29, Oct. 2018.
[40] A. Bagula, C. Lubamba, M. Mandava, H. Bagula, M. Zennaro, and E. Pietrosemoli, ''Cloud based patient prioritization as service in public health care,'' in *Proc. ITU Kaleidoscope ICTs Sustain. World (ITU WT)*, Nov. 2016, pp. 1–8.
[41] C. Lubamba, and A. Bagula, ''Cyber-healthcare Cloud computing interoperability using the HL7-CDA standard,'' in *Proc. IEEE Symp. Comput. Commun. (ISCC)*, Jul. 2017, pp. 105–110.
[42] J. Quinn, ''An hl7 (health level seven) overview,'' *J. Amer. Health Inf. Manage. Assoc.*, vol. 70, no. 7, p. 32, 1998.

OLASUPO O. AJAYI received the B.Sc., M.Sc., and Ph.D. degrees in computer science from the University of Lagos, Nigeria, in 2008, 2012 and 2017, respectively. Before his research fellowship, he was a Lecturer with the University of Lagos. He is currently a Postdoctoral Research Fellow with the University of the Western Cape, Cape Town, South Africa, under the supervision of Prof. A. B. Bagula. He has published a number of articles around these research interests in journals and proceedings of international conferences. His research interests include but are not limited to cloud/fog/edge computing, ICT4D, and the Internet-of-Things.



ANTOINE B. BAGULA received dual M.Sc. degrees in computer engineering from the Universite catholique de Louvain (UCL), Belgium, and in computer science from the University of Stellenbosch (SUN), South Africa, and the Ph.D. degree in communication systems from the Royal Institute of Technology (KTH), Sweden. He is currently a Full Professor and the Head of the Department of Computer Science, University of the Western Cape (UWC), where he also leads the Intelligent Systems and Advanced Telecommunication (ISAT) Laboratory. He is a well published scientist in his research field which include interests in the Internet-of-Things, cloud/fog computing, and next generation networks including 4G/5G. For the last ten years, he has applied a combination of these research interests to the design and implementation of cyber physical systems frameworks for health, energy, water, and pollution.

KUN MA received the bachelor's degree in computer science and the M.Sc. degree in computer science in 2008 and 2012, respectively. He is currently pursuing the Ph.D. degree in computer science with the University of the Western Cape, Cape Town, South Africa. His thesis' research focus is on resource management in cloud and fog computing.